PLOS Computational Biology

METHODS

Cell type annotation for scATAC-seq via DNA large language model and graph domain adaptation

Yan Liu[1], Sheng Guan[1], He Yan[2], Long-Chen Shen[3], Ji-Peng Qiang[1]*, Guo Wei[4]*

1 School of Information and Artificial Intelligence, Yangzhou University, Yangzhou, Jiangsu, China, **2** College of Information Science and Technology & Artificial Intelligence, Nanjing Forestry University, Nanjing, Jiangsu, China, **3** School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing, Jiangsu, China, **4** School of Computer Science and Information Engineering, Bengbu University, Bengbu, Anhui, China

* jipengqiang@yzu.edu.cn (J-PQ); weiguo@smail.nju.edu.cn (GW)

OPEN ACCESS

Citation: Liu Y, Guan S, Yan H, Shen L-C, Qiang J-P, Wei G (2026) Cell type annotation for scATAC-seq via DNA large language model and graph domain adaptation. PLoS Comput Biol 22(4): e1014226. https://doi.org/10.1371/journal.pcbi.1014226

Editor: Mingfu Shao, The Pennsylvania State University, UNITED STATES OF AMERICA

Received: October 13, 2025

Accepted: April 10, 2026

Published: April 30, 2026

Data availability statement: The datasets used in this study are publicly available at Synapse (https://www.synapse.org/Synapse:syn52559388/files/). The source code for reproducing the analyses and results is available at GitHub (https://github.com/sheng-guan-2001/scLLMDA).

Abstract

Single-cell ATAC-seq (scATAC-seq) enables the exploration of chromatin accessibility at single-cell resolution, offering critical insights into gene regulation. Accurate cell type annotation is a fundamental prerequisite in scATAC-seq analysis. While cross-modality annotation methods leverage scRNA-seq data for label transfer, they often suffer from modality mismatch and signal distortion. Intra-modality annotation, which utilizes only scATAC-seq reference data, has gained attention for its biological consistency. However, existing methods are limited by insufficient sequence representation and lack of neighborhood modeling during domain adaptation. To address these limitations, we propose scLLMDA, a novel framework for scATAC-seq cell type annotation via DNA large language model and graph-based domain adaptation (GDA). scLLMDA uses a pretrained DNA-specific language model to generate contextual embeddings of peak sequences, which are then integrated with accessibility information to represent individual cells. We construct similarity-based cell graphs for both source and target datasets, and apply a graph neural network to align domains while preserving local structural context. Our approach captures rich sequence semantics and neighborhood dependencies, enabling more accurate and robust cell type annotation across datasets. Extensive experiments on multiple benchmarks demonstrate that scLLMDA outperforms existing methods in accuracy. The source code and implementation of scLLMDA are publicly available at: https://github.com/sheng-guan-2001/scLLMDA.

Author summary

Single-cell ATAC-seq provides unique insights into gene regulation, but accurate cell type annotation remains challenging. Existing approaches often suffer from

Funding: This work was supported by the National Natural Science Foundation of China (Grant No. 62306142 to HY), the Jiangsu Funding Program for Excellent Postdoctoral Talent (Grant No. 2023ZB224 to HY), the Natural Science Foundation of the Higher Education Institutions of Jiangsu Province of China (Grant No. 24KJB520041 to YL), and the Scientific Research Project of the Anhui Provincial Department of Education (Natural Sciences Category – Youth Project, Grant No. 2025AHGXZK40444 to GW). The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing interests: The authors have declared that no competing interests exist.

modality mismatch or limited feature representation. We introduce scLLMDA, a framework that combines a DNA large language model with graph-based domain adaptation. By capturing sequence semantics and neighborhood structure, scLLMDA achieves more accurate and robust cell type annotation across datasets.

Introduction

scATAC-seq has emerged as a powerful technique for profiling chromatin accessibility at single-cell resolution [1], offering unique insights into gene regulatory landscapes across diverse cell types [2]. A key step in scATAC-seq data analysis is accurate cell type annotation, which enables downstream analyses such as trajectory inference, differential accessibility analysis, and integration with other single-cell omics datasets. However, due to the inherent sparsity and noise in scATAC-seq data, automated and robust annotation remains a significant challenge.

Existing cell type annotation methods for scATAC-seq can be broadly categorized into two groups based on the omics modality of the reference dataset: (1) Cross-modality annotation [3–5], which involves transferring cell type labels from scRNA-seq or other omics modalities through data integration techniques, has emerged as a widely used strategy in single-cell analysis. However, substantial challenges remain due to inherent differences in data structure, sparsity, and modality-specific biases, which complicate accurate alignment across omics layers. (2) Intra-modality annotation [6], which involves using well-annotated scATAC-seq datasets as references to directly annotate cells from new scATAC-seq experiments, has gained traction as a modality-specific alternative to cross-omics approaches. Several representative methods illustrate the diversity of strategies in this domain. For example, Cellcano [6] uses a two-stage supervised learning approach: an MLP [7] is first trained on reference data to predict target labels, followed by self-distillation on high-confidence "anchor" cells to refine predictions. scATAnno [8] extracts embeddings via SnapATAC2 [9], aligns reference and query data using Harmony [10], and assigns labels through a KNN classifier [11]. EpiAnno [12] adopts a Bayesian neural network [13] to incorporate uncertainty estimation, enhancing robustness and classification accuracy in cell type annotation. With the increasing availability of high-quality manually annotated scATAC-seq datasets, intra-modality annotation has gained more attention due to its reduced modality gap and improved biological consistency. In this work, we focus on the intra-modality scATAC-seq cell type annotation task, where the goal is to assign cell labels to unannotated cells using only scATAC-seq reference data.

Despite recent progress, current intra-modality annotation methods suffer from two main limitations: (1) Limited Sequence Representation: Most methods rely on convolutional neural networks (CNNs) to extract features from DNA sequences of accessible regions [14,15]. However, CNNs are constrained by fixed receptive fields and lack the ability to capture long-range dependencies or contextual biological signals embedded in the genomic sequence. This restricts the richness of the learned representations and may compromise annotation accuracy. (2) Neglect of Neighborhood

Context in Domain Adaptation [16]: When adapting knowledge from a labeled source dataset to an unlabeled target dataset, current domain adaptation strategies typically align global feature distributions without explicitly modeling the graph structure or local neighborhood relationships between cells. This can lead to suboptimal domain alignment, especially in heterogeneous or complex cell populations.

To address these challenges, we propose a novel framework entitled scLLMDA, the schematic is shown in Fig 1. Our method leverages a DNA-specific pretrained language model to obtain rich, context-aware representations of DNA sequences. These representations are combined with peak accessibility information to learn expressive embeddings for each cell. Furthermore, we construct cell-cell graphs based on embedding similarity and employ graph neural networks to incorporate local and global structure into both the feature learning and domain adaptation processes. By integrating sequence-level semantics and graph-aware domain alignment, our approach achieves more accurate and robust cell type annotation for scATAC-seq datasets. In summary, our contributions are threefold:

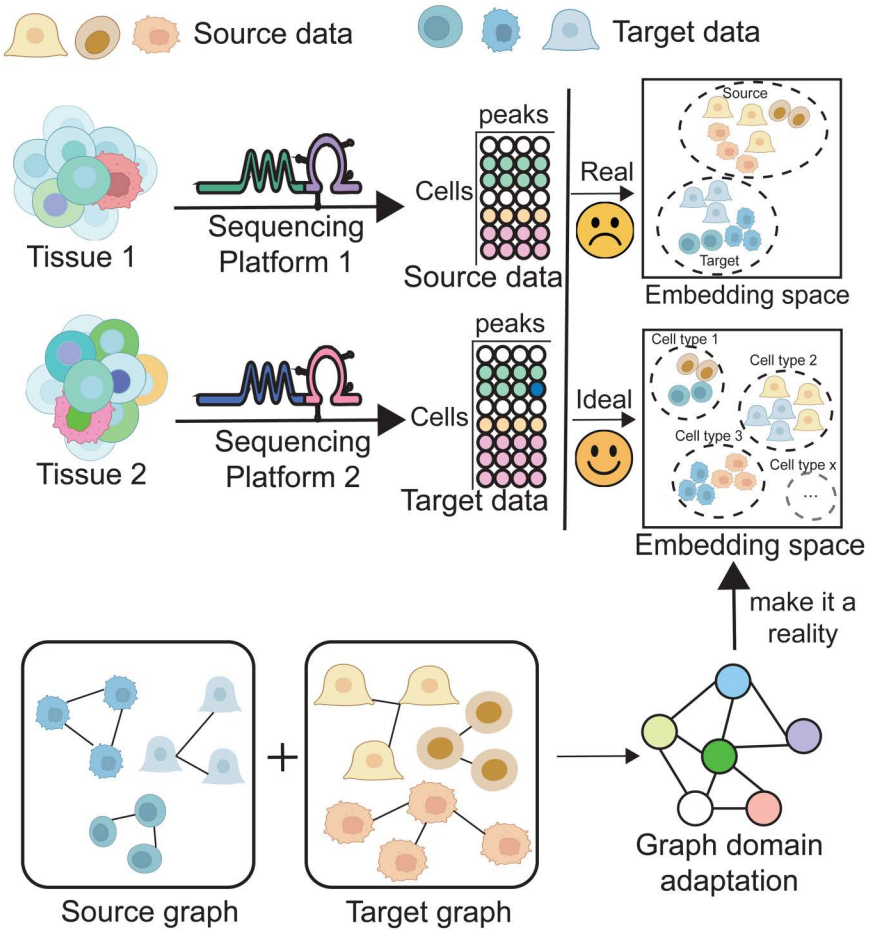

Fig 1. Overview of batch effect and graph-based domain adaptation in scATAC-seq cell type annotation. Cells from different tissues are sequenced using different platforms, resulting in source and target datasets with potential batch effects. The observed embedding space (top right) often separates cells by batch rather than by biological identity, due to sequencing-induced biases. The ideal scenario should organize cells according to true cell types, regardless of platform or batch. To achieve this, we construct source and target graphs based on intra-dataset similarity, and apply a graph domain adaptation strategy to align the two domains. This framework enables batch-aware alignment and improves cross-platform cell type annotation accuracy.

https://doi.org/10.1371/journal.pcbi.1014226.g001

- We introduce DNA large language model into the scATAC-seq annotation pipeline to enhance sequence-level feature representation;
- We design a graph-based domain adaptation strategy that captures neighborhood structure during source-target alignment;
- We demonstrate superior performance over existing methods on multiple benchmark datasets, showing strong generalizability across domains.

Materials and methods

Benchmark datasets

In this study, we utilized several publicly available single-nucleus ATAC-seq (snATAC-seq, A sequencing platform of the ATAC-seq technique) datasets to comprehensively evaluate the performance of the proposed method. These datasets cover different tissue regions and sequencing platforms. Specifically, MosA1, MosM1, and MosP1 were derived from chromatin accessibility profiling of distinct functional regions within the cerebral cortex [17,18], annotated based on the GRCm38 reference genome, and are available from the GEO database (accession number: `GSE126724`). The Mouse Brain (10x) dataset was generated using the 10x Genomics platform with `mm10` as the reference genome. WholeBrainA and WholeBrainB are atlas-style datasets [19] produced using sciATAC-seq [20] (A sequencing platform of the ATAC-seq technique) and annotated with the `mm9` reference genome; they are accessible via the GEO database (accession number: `GSE111586`).

Benchmark methods

To evaluate the effectiveness of our proposed approach, we compare it against seven representative baseline methods. scNym [21], scJoint [3], and Cellcano [6] rely solely on gene peak features without incorporating DNA sequence information. For scJoint, we use scATAC-seq data during both the training and testing phases. Cellcano is evaluated using its original settings: when the target cell number in the test set is small, results from the first training round are used; otherwise, results from the second round are applied. In contrast, SANGO [15] integrates both DNA sequence and peak-based features for joint representation learning. annATAC [22] is a language model–based framework for automatic scATAC-seq cell type annotation, leveraging pre-training, fine-tuning, and prediction to achieve accurate labeling. AtacAnnoR [23] is a reference-based scATAC-seq cell type annotation method that employs a two-round strategy to transfer labels from annotated scRNA-seq data and refine predictions using genome-wide chromatin accessibility features. MINGLE [24] is a mutual information–based interpretable framework for scATAC-seq cell type annotation that integrates cellular similarity and topological structure modeling to improve robustness and enable identification of rare or novel cell types.To ensure fairness, all training data are preprocessed following the same pipeline as used in SANGO.

Problem definition

We consider the task of intra-modality cell type annotation for scATAC-seq data, where both the reference (source domain) and query (target domain) datasets originate from the same modality. Although both datasets are collected using scATAC-seq, they may exhibit significant batch effect due to variations in sequencing platforms(e.g., snATAC-seq, 10x, sciATAC-seq), tissue sources, or experimental settings [25]. These discrepancies can lead to distributional shifts in the feature space between the source and target domains, which pose challenges to direct label transfer and necessitate effective domain adaptation strategies [26]. Formally, let:

- $\mathcal{D}_s = \{(x_i^s, y_i^s)\}_{i=1}^{N_s}$ denote the source domain, where x_i^s is the DNA sequence and peak signal representation for cell i, and $y_i^s \in \{1, 2, \ldots, C\}$ is the corresponding cell type label.

- $\mathcal{D}_t = \{x_j^t\}_{j=1}^{N_t}$ denote the target domain, where labels are unavailable and need to be inferred.

Our objective is to learn a function $f : x \to y$ such that for each $x_j^t \in \mathcal{D}_t$, the predicted label $\hat{y}_j^t = f(x_j^t)$ is as accurate as possible. To this end, we define a two-stage solution: Step 1: Feature Extraction from Genomic Sequences; Step 2: Cell Type Annotation via Graph Domain Adaptation.The overall framework of our proposed method is illustrated in Fig 2.

Feature extraction from genomic sequences

To extract biologically meaningful sequence features, we utilize the pretrained DNA language model [27] (i.e., DNABERT-2 [28]) to perform contextualized encoding of each peak sequence. In contrast to conventional convolutional

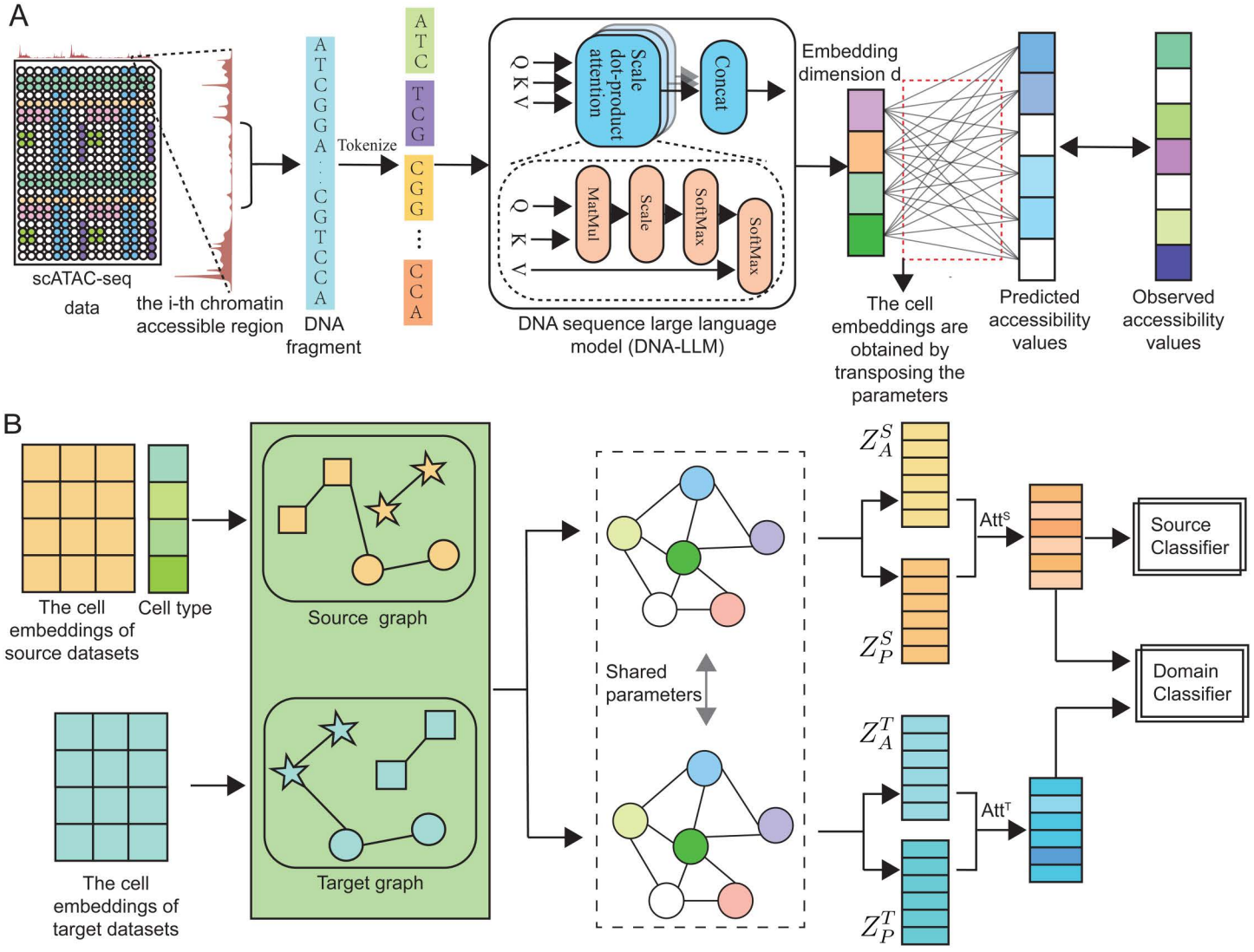

Fig 2. Overview of the proposed scLLMDA framework for cell type annotation from scATAC-seq data. (A) *Feature extraction from genomic sequences*: Chromatin-accessible regions are tokenized and encoded using a pretrained DNA language model (DNA-LLM), generating contextualized embeddings that are used to predict accessibility profiles and derive cell embeddings. **(B)** *Cell type annotation via graph domain adaptation*: Source and target cell embeddings are used to construct corresponding graphs. A domain adaptation module with attention mechanisms and shared parameters enables effective knowledge transfer and cell type classification across datasets.

https://doi.org/10.1371/journal.pcbi.1014226.g002

approaches, DNABERT-2 captures intricate regulatory syntax and long-range dependencies, including transcription factor binding patterns and motif co-occurrence. The resulting sequence embeddings integrate both local sequence motifs and higher-order regulatory semantics, thereby offering a comprehensive and biologically enriched feature representation.

Let seq_i denote the DNA sequence corresponding to the i-th peak in cell i. Prior to encoding, each peak sequence is tokenized strictly following the official DNABERT-2 tokenizer. Concretely, we first clean the raw peak sequence by converting it to uppercase and removing non-canonical characters (only A/T/C/G are retained; other symbols such as N are filtered or replaced). We then apply the SentencePiece tokenizer trained with Byte Pair Encoding [29] (BPE) to segment the sequence into a list of variable-length sub-sequence tokens (subwords) according to the learned vocabulary (size $2^{12} = 4096$). Next, special tokens required by the Transformer are appended (e.g., a start token and an end/separator token), and the resulting token sequence is mapped to integer IDs to form the model input (optionally with padding/truncation to the maximum length supported by DNABERT-2). These token IDs are finally fed into the Transformer encoder to obtain contextualized embeddings.

The model outputs:

$$\mathbf{z}_i = \text{DNABERT}(\text{seq}_i) \in \mathbb{R}^{d'}, \tag{1}$$

where $\mathbf{z}_i$ is a high-dimensional embedding of the peak. These peak embeddings $\{\mathbf{z}_i\}_{i=1}^{N_p}$ are then used to predict the binary accessibility of each peak across N_{cell} cells using a fully connected output layer [30]:

$$\hat{\mathbf{Y}} = \sigma(\mathbf{Z}\mathbf{W}_o + \mathbf{b}_o), \tag{2}$$

where $\mathbf{Z} \in \mathbb{R}^{N_p \times d}$ is the matrix of all peak embeddings, $\mathbf{W}_o \in \mathbb{R}^{d \times N_{\text{cell}}}$ is the learnable weight matrix of the output layer, and $\sigma(\cdot)$ is the element-wise sigmoid activation. The predicted values $\hat{\mathbf{Y}} \in [0,1]^{N_p \times N_{\text{cell}}}$ represent the estimated peak accessibility across all cells. The model is trained by minimizing the binary cross-entropy (BCE) loss between predicted and observed accessibility:

$$\mathcal{L}_{\text{BCE}} = -\frac{1}{N_p \cdot N_{\text{cell}}} \sum_{i=1}^{N_p} \sum_{j=1}^{N_{\text{cell}}} \Big[y_{ij} \log(\hat{y}_{ij}) + (1 - y_{ij}) \log(1 - \hat{y}_{ij}) \Big]. \tag{3}$$

After training, we extract the column-wise weight vectors from $\mathbf{W}_o$ as the cell embeddings for the source and target datasets, respectively. Each column $\mathbf{w}_j \in \mathbb{R}^d$ of $\mathbf{W}_o$ serves as the d-dimensional representation of cell j.

Cell type annotation via graph domain adaptation

Graph construction. Given the learned embeddings of all cells by the Step 1, we construct two k-nearest neighbor graphs based on similarity in the embedding space:

- Source domain graph: $G_s = (\mathcal{V}_s, \mathcal{E}_s, \mathbf{X}_s)$, where $\mathcal{V}_s$ are labeled source cells, $\mathcal{E}_s$ are edges computed using the KNN algorithm, and $\mathbf{X}_s$ is the feature matrix.
- Target domain graph: $G_t = (\mathcal{V}_t, \mathcal{E}_t, \mathbf{X}_t)$, similarly defined for unlabeled target cells.

Capture the local consistency relationship of each graph. To perform local structural representation learning for both the source graph $G_s = (\mathcal{V}_s, \mathcal{E}_s, \mathbf{X}_s)$ and the target graph $G_t = (\mathcal{V}_t, \mathcal{E}_t, \mathbf{X}_t)$, we introduce a customized graph convolutional module, AgConv, built upon CachedGCNConv—a computationally efficient variant of the GCN originally

proposed by Kipf and Welling [31]. This design enhances the model's capacity to capture neighborhood-level interactions while maintaining computational scalability.

The forward propagation rule for AgConv at the l-th layer is defined as:

$$\mathbf{H}^{(l+1)} = \phi\left(\tilde{\mathbf{D}}^{-1/2}\tilde{\mathbf{A}}\tilde{\mathbf{D}}^{-1/2}\mathbf{H}^{(l)}\mathbf{W}^{(l)} + \mathbf{b}^{(l)}\right), \tag{4}$$

where $\tilde{\mathbf{A}} = \mathbf{A} + \mathbf{I}$ is the adjacency matrix augmented with self-loops, and $\tilde{\mathbf{D}}$ is the corresponding degree matrix with diagonal entries $\tilde{D}ii = \sum_j \tilde{A}ij$. Here, $\mathbf{H}^{(l)}$ denotes the input cell features at layer l, while $\mathbf{W}^{(l)}$ and $\mathbf{b}^{(l)}$ are the learnable weight matrix and bias vector, respectively. The function $\phi(\cdot)$ denotes a ReLU activation function.

AgConv is applied independently to the source and target graphs, yielding the localized feature representations $\mathbf{Z}_A^S$ and $\mathbf{Z}_A^T$ for the source and target domains, respectively.

Capture the global consistency relationship of each graph. To overcome the representational limitations of relying solely on first-order neighborhood information and to improve the model's capacity to capture higher-order structural dependencies [32], we introduce a Positive Pointwise Mutual Information (PPMI)–based Graph Convolution module, referred to as PgConv. This mechanism is applied to both the source domain graph $G_s = (V_s, E_s, X_s)$ and the target domain graph $G_t = (V_t, E_t, X_t)$.

Given an input graph $G = (V, E, X)$, where V denotes the set of nodes, $E \subseteq V \times V$ the set of edges, and X the cell feature matrix, we perform random walks [33] of a maximum length L from each node $u \in V$. For every other node $v \in V$ visited during these walks, we record the visitation count as $C(u, v)$. This leads to the empirical conditional probability of observing node v given node u, defined as:

$$P(v \mid u) = \frac{C(u, v)}{\sum_{v' \in V} C(u, v')} \tag{5}$$

This probability reflects how frequently node v appears in the local neighborhood of node u based on the random walk trajectory.

To quantify the global occurrence of node v across all cell contexts [34], we define the marginal probability as:

$$P(v) = \sum_{u \in V} P(v \mid u) \tag{6}$$

The Pointwise Mutual Information (PMI) for each cell pair (u, v) is computed as:

$$\mathrm{PMI}(u, v) = \log\left(\frac{P(v \mid u)}{P(v)}\right) \tag{7}$$

To retain only statistically significant node associations, we define the PPMI as:

$$\mathrm{PPMI}(u, v) = \max\left(0, \log\left(\frac{P(v \mid u)}{P(v)} \cdot \frac{|V|}{L}\right)\right) \tag{8}$$

where $|V|$ denotes the number of cells in the graph, and L is the maximum walk length, serving as a normalization factor. Based on these scores, a new weighted edge set is formed:

$$E' = \left\{(u, v, w_{uv}) \mid w_{uv} = \mathrm{PPMI}(u, v),\ w_{uv} > 0\right\} \tag{9}$$

This process is applied to both the source and target graphs, yielding the corresponding PPMI-weighted adjacency matrices P_s and P_t, respectively.

Given a PPMI matrix P (either P_s or P_t), we perform graph convolution using the following propagation rule:

$$H^{(l+1)} = \phi\left(\tilde{D}^{-1/2}P\tilde{D}^{-1/2}H^{(l)}W^{(l)} + b^{(l)}\right) \tag{10}$$

Here, $\tilde{D}$ denotes the degree matrix of P, $W^{(l)}$ and $b^{(l)}$ are the learnable weights and biases of the l -th layer (shared with the AgConv module), and $\phi(\cdot)$ represents a non-linear activation function, such as ReLU. This operation yields the global node embeddings for the source and target graphs, denoted by Z_P^S and Z_P^T, respectively.

Feature fusion via attention. To integrate both local and global information of cells within the graph, we design an attention-based feature fusion module [35,36], which is applied independently to the source domain graph and the target domain graph. This module unifies the features extracted from AgConv (local structure) and PgConv (global semantics) and performs weighted fusion to obtain more expressive cell representations.

Specifically, for a given graph $G = (V, E, X)$, let Z_A denote the node features extracted via AgConv and Z_P denote the features from PgConv based on PPMI encoding.

First, we concatenate the two feature matrices:

$$H = \text{stack}(Z_A, Z_P) \tag{11}$$

Next, we introduce three shared linear projections to transform the stacked features into the query, key, and value matrices:

$$Q = HW_Q, \quad K = HW_K, \quad V = HW_V \tag{12}$$

We then compute attention weights across different feature types and obtain the attention-enhanced representation:

$$H_{\text{att}} = \text{softmax}\left(\frac{QK^\top}{\sqrt{d}}\right)V \tag{13}$$

Finally, we aggregate the representations to produce the fused cell features:

$$Z = \text{Aggregate}(H_{\text{att}}) \tag{14}$$

This fusion procedure is applied separately to both source and target graphs, i.e., to Z_A^S, Z_P^S and Z_A^T, Z_P^T, resulting in the final fused representations Z^S and Z^T, respectively.

Loss function

In our model, the training objective jointly considers classification accuracy on the source domain and domain alignment between the source and target domains. Let Z^S and Z^T denote the learned cell embeddings for the source and target domain graphs, respectively.

The overall loss function is defined as:

$$\mathcal{L} = \mathcal{L}_{\text{cls}} + \lambda \cdot \mathcal{L}_{\text{grl}} \tag{15}$$

where $\mathcal{L}_{\text{cls}}$ is the supervised classification loss on the source domain, $\mathcal{L}_{\text{grl}}$ is the domain adversarial loss [37] based on a gradient reversal mechanism, and λ is a hyperparameter balancing the two terms.

The classification loss is defined as the average multi-class negative log-likelihood [38] over all labeled source domain nodes:

$$\mathcal{L}_{\text{cls}} = -\frac{1}{V_s}\sum_{i=1}^{V_s}\sum_{c=1}^{C}\mathbb{I}(y_i^S = c)\cdot \log p_c^S(i) \tag{16}$$

where V_s is the number of source domain nodes, C is the number of classes, y_i^S is the ground truth label of the i-th source node, $p_c^S(i)$ is the predicted probability that node i belongs to class c, and $\mathbb{I}(\cdot)$ is the indicator function that returns 1 if the condition holds and 0 otherwise.

To align the source and target domains, we introduce a gradient reversal layer and train a domain discriminator $f_{\text{dom}}(\cdot)$ that encourages the node embeddings to be domain-invariant. The domain adversarial loss is defined as a binary cross-entropy loss over both source and target nodes:

$$\mathcal{L}_{\text{grl}} = -\frac{1}{V_s}\sum_{i=1}^{V_s}\log p_0^S(i) - \frac{1}{V_t}\sum_{j=1}^{V_t}\log p_1^T(j) \tag{17}$$

where $p_0^S(i)$ is the predicted probability that the domain discriminator classifies source node i as "source," and $p_1^T(j)$ is the predicted probability that target node j is classified as "target," with V_t being the number of target domain nodes.

Parameter settings

During the cell representation learning phase, we use a MLP with hidden dimensions of 512, 256, and 128, respectively. The initial learning rate is set to 1e-3, and the model is trained for 400 epochs. For the graph domain adaptation stage, all nodes from both the source and target graphs are jointly fed into the model for training. A unified learning rate of 3e-3 is used in this stage, and the GCNs for both source and target domains consist of two hidden layers ($L = 2$) with a network architecture of 100–16. The number of training epochs varies depending on the source graph: 55 epochs when the source graph is MosP1, MosA1, or MosM1; 122 epochs for Mouse Brain (10x); and 22 epochs for WholeBrainA or WholeBrainB. The only exception is when WholeBrainB is used as the source and MosM1 as the target, where training is conducted for 138 epochs.

Results

Intra-platform cell type annotation

As shown in Table 1, all evaluated methods exhibit strong performance on the intra-platform cell type annotation tasks, such as MosA1→MosM1, MosM1→MosP1, and MosA1→MosP1, where the reference and query datasets are derived from the same sequencing platform and share similar experimental conditions. In these cases, the performance gap between different methods is marginal (e.g., most methods achieve accuracy above 0.95), indicating that batch effects and distributional shifts are relatively mild in intra-platform scenarios. This also suggests that simpler models relying primarily on peak-based features may suffice when the domain discrepancy is small.

Although our method incorporates a batch effect correction module specifically designed for cross-platform cell type annotation, it still achieves comparable or even slightly superior annotation accuracy on intra-platform datasets.

Cross-platform cell type annotation

To investigate the robustness of cell type annotation across sequencing platforms, we conducted comprehensive evaluations involving both snATAC-seq and sciATAC-seq datasets. As shown in Table 2, we performed bi-directional transfer

Table 1. Performance comparison (Accuracy and F1-score) of intra-platform cell type annotation.

Method	R: WholeBrainA Q: WholeBrainB		R: WholeBrainB Q: WholeBrainA		R: MosA1 Q: MosM1		R: MosM1 Q: MosA1	
	Acc	F1	Acc	F1	Acc	F1	Acc	F1
scLLMDA	0.9351	0.7673	0.9184	0.8277	0.9682	0.9618	0.9780	0.9757
SANGO	0.9358	0.7899	0.9150	**0.8625**	**0.9780**	**0.9750**	0.9763	0.9746
scNym	0.9125	0.8152	0.8970	0.7776	0.9522	0.9434	0.9637	0.9560
scJoint	**0.9425**	**0.8414**	**0.9203**	0.8014	0.9730	0.9682	**0.9794**	**0.9761**
Cellcano	0.8847	0.7261	0.8658	0.7008	0.9207	0.9005	0.9226	0.9063
annATAC	0.8837	0.6942	0.7812	0.4904	0.947	0.9372	0.9650	0.9605
AtacAnnoR	0.8655	0.6124	0.9044	0.832	0.8871	0.8747	0.8767	0.8679
MINGLE	0.9051	0.8294	0.9139	0.7984	0.9493	0.9421	0.9503	0.9525
Method	**R: MosA1 Q: MosP1**		**R: MosP1 Q: MosA1**		**R: MosM1 R: MosP1**		**Q: MosP1 Q: MosM1**	
	Acc	**F1**	**Acc**	**F1**	**Acc**	**F1**	**Acc**	**F1**
scLLMDA	0.9688	0.9632	**0.9804**	0.9776	0.9713	0.9635	0.9706	0.9590
SANGO	**0.9770**	**0.9777**	0.9773	**0.9791**	**0.9796**	**0.9748**	0.9625	0.9616
scNym	0.9552	0.9523	0.9644	0.9581	0.9587	0.9456	0.9583	0.9453
scJoint	0.9659	0.9684	0.9738	0.9719	0.9765	0.9726	**0.9733**	**0.9662**
Cellcano	0.9231	0.8969	0.9234	0.9056	0.9257	0.8991	0.9247	0.9003
annATAC	0.9266	0.9365	0.9607	0.9549	0.9638	0.9563	0.9456	0.9434
AtacAnnoR	0.8791	0.8325	0.8844	0.8748	0.8876	0.8421	0.8621	0.845
MINGLE	0.9539	0.9468	0.9405	0.9416	0.9341	0.9067	0.9494	0.9476

https://doi.org/10.1371/journal.pcbi.1014226.t001

experiments between snATAC-seq datasets (MosA1, MosM1, MosP1) and sciATAC-seq datasets (WholeBrainA, WholeBrainB). Table 3 further complements the analysis by including the 10x Genomics-based MouseBrain(10x) dataset as the reference or query in cross-platform scenarios.

Across both tables, our proposed method, scLLMDA, consistently achieves superior performance compared to all baseline methods in terms of both accuracy and macro-F1 score, with the exceptions of the WholeBrainB→MouseBrain(10x) and WholeBrainA→MosP1 transfer scenarios. Notably, scLLMDA achieves the best performance in most evaluated directions, demonstrating strong generalization when transferring annotations between platforms with differing technical properties and cell coverage. For example, in the challenging transfer from WholeBrainA to MosA1 (Table 2), scLLMDA achieves an accuracy of 0.7691 and a macro-F1 of 0.3993, outperforming scJoint and scNym by substantial margins. Similar trends are observed in transfers involving MouseBrain(10x), where scLLMDA obtains the best F1 score (0.4249) in the WholeBrainB→MouseBrain(10x) direction (Table 3).

These results highlight two key observations. First, peak-only methods such as scNym, scJoint, and Cellcano show moderate performance but struggle to generalize across technologies, particularly in reverse transfer settings. Second, sequence-aware models like SANGO offer improvements, but integrating DNA sequence features with domain adaptation (as done in scLLMDA) further boosts performance significantly.

In summary, scLLMDA demonstrates strong cross-platform transferability, particularly in scenarios involving heterogeneous sequencing protocols (snATAC-seq vs. sciATAC-seq) and variable cell type granularity. This validates the advantage of combining local chromatin accessibility patterns with global regulatory context via sequence-based representation learning.

Table 2. Cell type annotation between snATAC-seq and sciATAC-seq platforms.

Method	Ref: MosA1 Q: WholeBrainA		R: WholeBrainA Q: MosA1		R: MosP1 Q: WholeBrainA		R: WholeBrainA Q: MosP1		R: MosM1 Q: WholeBrainB		R: WholeBrainB Q: MosM1	
	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1
SANGO	0.6508	0.5868	0.6964	0.2981	0.7065	0.6077	0.6062	0.2940	0.6693	0.6035	0.6525	0.3034
scNym	0.6817	0.5843	0.7471	0.3108	0.6937	0.5986	0.6643	0.2932	0.6739	0.6075	0.5877	0.3101
scJoint	0.6452	0.5678	0.7569	0.3403	0.5572	0.5237	**0.7081**	0.3494	0.6574	0.5878	0.7422	0.4118
Cellcano	0.6416	0.5388	0.5335	0.3035	0.6409	0.5350	0.4686	0.2952	0.5970	0.4913	0.4173	0.2803
annATAC	0.6342	0.5578	0.6654	0.2966	0.6320	0.5127	0.5228	0.1938	0.6010	0.5252	0.390	0.2052
AtacAnnoR	0.4523	0.3544	0.5442	0.2317	0.4658	0.371	0.3878	0.2333	0.6648	0.5956	0.2827	0.2025
MINGLE	0.7143	0.6256	0.7095	0.32	0.7097	0.6243	0.6459	0.2973	0.6665	0.6022	0.7033	0.3756
scLLMDA	**0.7228**	**0.6525**	**0.7691**	**0.3993**	**0.7114**	**0.6253**	0.7044	**0.3630**	**0.7012**	**0.6490**	**0.7583**	**0.4580**

https://doi.org/10.1371/journal.pcbi.1014226.t002

Table 3. Cell type annotation between MouseBrain(10x) and sciATAC-seq platforms.

Method	R:MouseBrain(10x)	R: WholeBrainA	R: MouseBrain(10x)	R: WholeBrainB
	Q: WholeBrainA	Q: MouseBrain(10x)	Q: WholeBrainB	Q: MouseBrain(10x)
	Acc / F1	Acc / F1	Acc / F1	Acc / F1
SANGO	0.7309 / 0.6307	0.7019 / 0.3614	0.7006 / 0.6270	0.7003 / 0.3601
scNym	0.6636 / 0.5466	0.7568 / 0.3661	0.6631 / 0.5754	**0.7794** / 0.3712
scJoint	0.7097 / 0.6154	0.7453 / 0.3541	0.6813 / 0.6131	0.7611 / 0.3576
Cellcano	0.6324 / 0.5245	0.6978 / 0.3705	0.6050 / 0.5094	0.6801 / 0.3635
annATAC	0.6735/ 0.5392	0.7527/ 0.3934	0.6659 / 0.5690	0.7469 / 0.3349
AtacAnnoR	0.6648 / 0.5956	0.5942 / 0.2813	0.5885 / 0.4776	0.3782 / 0.2837
MINGLE	0.7256 / 0.6255	0.721 / 0.4069	0.6938 / 0.6196	0.7696 / 0.3999
scLLMDA	**0.7423** / **0.6616**	**0.7606** / **0.4141**	**0.7359** / **0.6703**	0.7423 / **0.4249**

https://doi.org/10.1371/journal.pcbi.1014226.t003

UMAP visualization comparison across methods

To further provide an intuitive comparison of annotation quality across methods, we visualize the predicted labels and corresponding ground-truth labels using UMAP embeddings under the MosA1→WholeBrainA transfer task. The visualizations of annATAC, AtacAnnoR, Cellcano, MINGLE, SANGO, scJoint, scNym, and scLLMDA are shown in Fig 3.

For annATAC, it is important to note that the model does not directly output cell-level embeddings. Instead, annATAC produces sequence-level representations with shape (N, L, 200), where N denotes the number of cells and L corresponds to the peak dimension. Since no officially defined cell embedding is provided, we obtained approximate cell-level representations by averaging along the peak dimension, yielding (N, 200) features for UMAP visualization. This post-processing step may partially affect the geometric structure of annATAC embeddings and explains the atypical visualization pattern observed in Fig 3.

From the UMAP comparisons, several observations can be made. First, most baseline methods exhibit notable mixing between cell populations, indicating limited separability under cross-platform transfer. In particular, methods such as scNym, scJoint, Cellcano, MINGLE, and AtacAnnoR show substantial overlap among neuronal subtypes and glial populations, reflecting challenges in capturing discriminative regulatory patterns across technologies. Second, sequence-aware methods such as SANGO demonstrate improved clustering structure but still display fragmented or dispersed clusters for certain cell types.

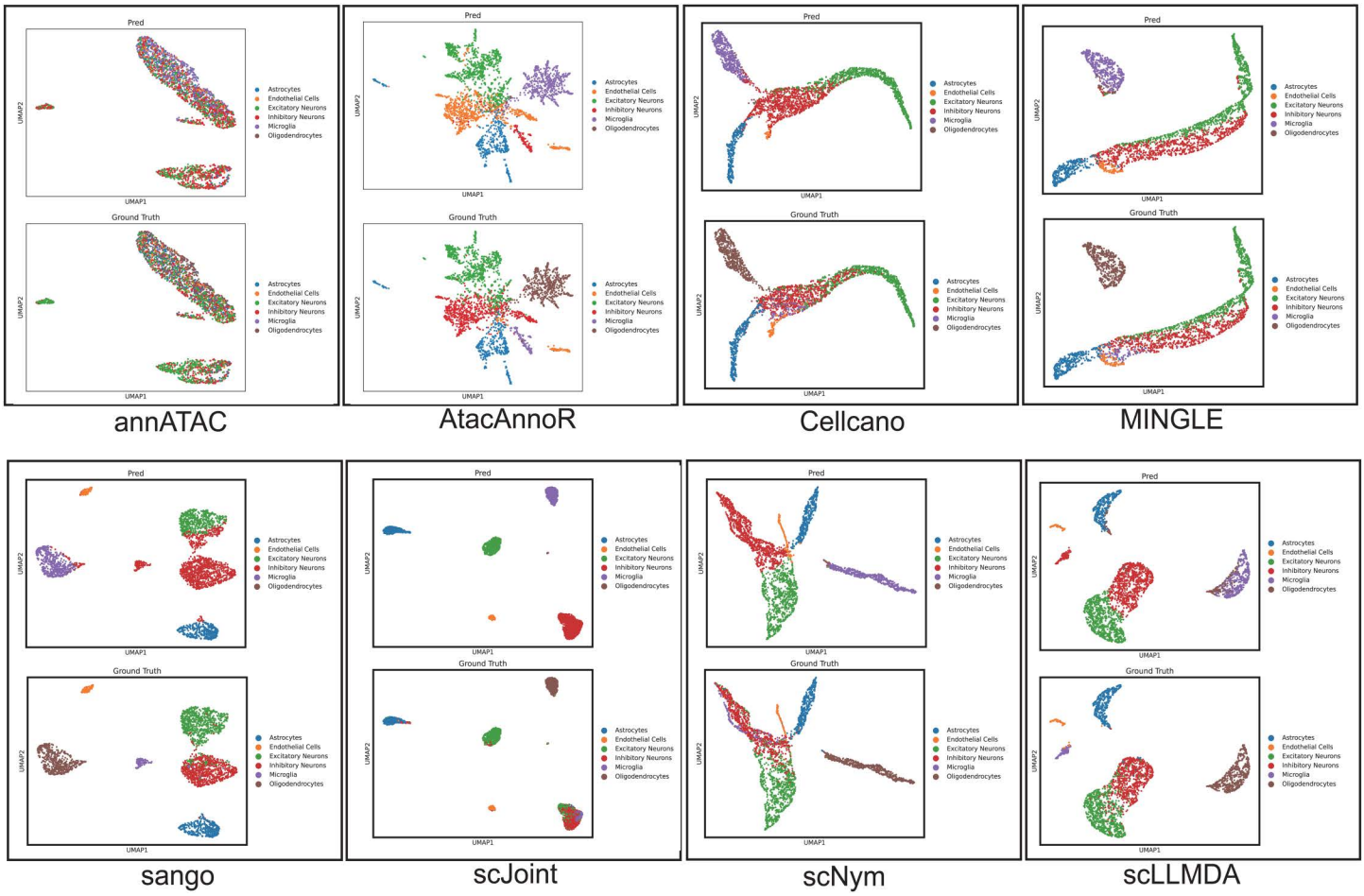

Fig 3. UMAP visualization comparison of cell type annotation results under the MosA1→WholeBrainA transfer task. Predicted labels (top row) and ground-truth labels (bottom row) are shown for each method, including annATAC, AtacAnnoR, Cellcano, MINGLE, SANGO, scJoint, scNym, and scLLMDA.

https://doi.org/10.1371/journal.pcbi.1014226.g003

A particularly challenging case is the Oligodendrocyte population. As shown in the prediction panels, nearly all baseline methods fail to recover a coherent cluster for this cell type, resulting in extremely low recognition rates. In contrast, scLLMDA successfully identifies a subset of Oligodendrocyte cells and forms a partially separable cluster consistent with the ground-truth distribution. This improvement highlights the benefit of incorporating sequence-informed regulatory representations together with domain adaptation to capture subtle chromatin accessibility signatures.

Overall, the UMAP visualization corroborates the quantitative results by demonstrating that scLLMDA produces more structured and biologically meaningful embedding spaces, leading to improved cross-platform cell type discrimination.

Computational efficiency and scalability

Scalability is a major concern for single-cell omics methods, especially under cross-dataset annotation scenarios. To evaluate scalability, we conducted systematic computational efficiency benchmarks in terms of running time and GPU memory usage on two representative cross-dataset annotation tasks, namely MosA1→WholeBrainA and

MouseBrain(10x)→WholeBrainA. We compared scLLMDA with several representative baselines, including sequence-based and expression-based methods. The results are summarized in Table 4.

For scLLMDA, the overall pipeline consists of three stages: (1) DNABERT-2 feature extraction, (2) cell embedding learning, and (3) graph construction with domain adaptation. On the MosA1→WholeBrainA task, the total running time is 53min37s, with 15min19s, 37min40s, and 38s for the three stages, respectively. The corresponding GPU memory consumption is 990MiB, 954MiB, and 1186MiB (peak ~1.2GB). Similar behavior is observed on the MouseBrain(10x)→WholeBrainA task, with a total running time of 39min19s and a comparable peak memory footprint.

Compared to the sequence modeling baseline SANGO, scLLMDA achieves substantially lower computational overhead while providing improved annotation performance. For example, on the MosA1→WholeBrainA task, SANGO requires 127min10s and 1656MiB, whereas scLLMDA completes in 53min37s with ~1.2GB peak GPU memory, reducing runtime by over 50%. Although scLLMDA is slower than some expression-based methods (e.g., scNym, scJoint) due to the additional DNABERT-2 sequence modeling, it remains significantly more memory-efficient than methods such as annATAC (around 22GB). Overall, the benchmarks demonstrate that scLLMDA maintains a favorable balance between efficiency and accuracy among sequence-informed approaches, indicating good scalability in practical GPU settings.

Ablation study

Sensitivity analysis of the balancing hyperparameter λ. We selected four representative cross-platform annotation tasks to evaluate the sensitivity of scLLMDA to the balancing hyperparameter λ. As shown in Fig 4, with the gradual increase of λ, the model's performance in terms of accuracy and F1 score generally exhibits a rising-then-falling trend, with the optimal performance typically occurring around $\lambda \approx 1.0$. This observation suggests that introducing a moderate domain adversarial signal effectively enhances the model's adaptability to distributional shifts between platforms. However, setting λ too large introduces an overly strong adversarial signal, which suppresses the learning of task-discriminative features in the main classification objective and leads to degraded overall performance. This degradation is especially prominent in challenging scenarios with significant platform discrepancies, such as *MouseBrain→WholeBrainB*. Therefore, careful tuning of λ is crucial for achieving accurate and robust transfer annotation. Based on multiple experiments, we recommend setting λ within the range of 0.9 to 1.1, where scLLMDA strikes a good balance between accuracy and stability, demonstrating strong generalization and cross-platform transfer performance.

Effectiveness of GDA module

To evaluate the effectiveness of the proposed GDA module in the scLLMDA framework, we conducted an ablation study comparing two variants: (i) DNABERT+GDA, the full pipeline where cell embeddings extracted in the first stage are used

Table 4. Computational efficiency benchmarks (time and memory) on two cross-dataset annotation tasks. For CPU-only methods, we report "CPU" in the memory column.

MosA1→WholeBrainA			MouseBrain(10x)→WholeBrainA		
Method	**Time**	**Memory**	**Method**	**Time**	**Memory**
SANGO	126min20s+50s = 127min10s	1656MiB	SANGO	116min12s+43s = 116min55s	1656MiB
scNym	8min5s+11s = 8min16s	1094MiB	scNym	5min35s+10s = 5min45s	1094MiB
scJoint	4min24s	2742MiB	scJoint	4min11s	2742MiB
Cellcano	8min16s+1min5s = 9min21s	CPU	Cellcano	4min19s+1min7s = 5min26s	CPU
scLLMDA	15min19s+37min40s + 38s = 53min37s	1186MiB	scLLMDA	15min42s+23min4s + 33s = 39min19s	1186MiB
annATAC	116min21s+56s = 117min17s	22528MiB	annATAC	50min51s+58s = 51min49s	22728MiB
AtacAnnoR	3min36s	CPU	AtacAnnoR	6min56s	CPU
MINGLE	1min6s	13352MiB	MINGLE	1min2s	13998MiB

https://doi.org/10.1371/journal.pcbi.1014226.t004

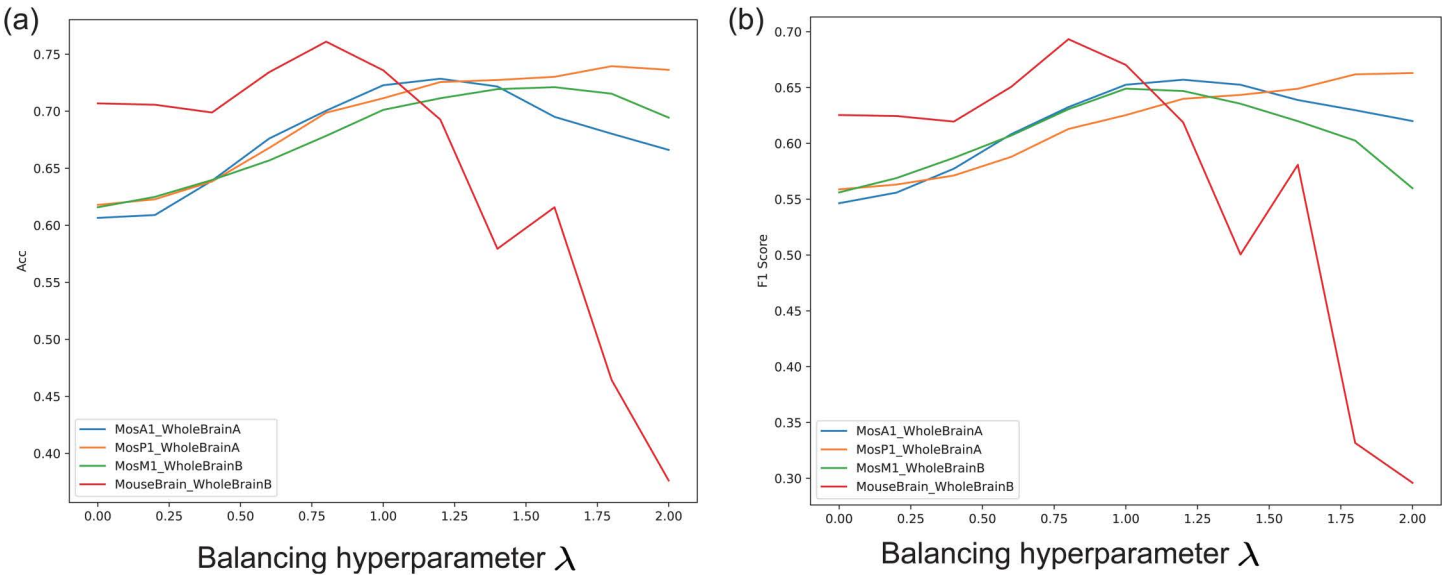

Fig 4. Effect of the balancing hyperparameter λ on cross-platform annotation performance of scLLMDA. (a) Accuracy; **(b)** F1-score.

https://doi.org/10.1371/journal.pcbi.1014226.g004

for annotation through the GDA module; and (ii) DNABERT+KNN, where the GDA module is replaced with a *k*-nearest neighbor classifier ($k = 6$), serving as a baseline.

The experiment is performed across 10 transfer tasks involving different sequencing platforms (e.g., MosA1, MouseBrain(10x), WholeBrainA/B), with performance measured by accuracy and macro-F1 score. As shown in Fig 5, DNABERT+GDA consistently outperforms DNABERT+KNN across both metrics. The boxplots illustrate that DNABERT+GDA yields higher median accuracy and macro-F1, while black dots represent per-task results, confirming the robustness of GDA across diverse settings.

In particular, GDA demonstrates significant advantages in more challenging scenarios involving domain shifts, suggesting its ability to align feature distributions and enhance the recognition of minority or ambiguous cell types. This ablation study highlights that domain adversarial training is crucial for improving generalization in cell type annotation tasks based on scATAC-seq data.

Discussion

In this study, we proposed scLLMDA, a novel framework that combines DNA language model and graph-based domain adaptation for accurate scATAC-seq cell type annotation. Experimental results show that scLLMDA consistently outperforms existing methods across both intra- and cross-platform settings. The key strengths of our method lie in its ability to capture rich sequence semantics and structural relationships between cells. By integrating contextual DNA embeddings with local and global graph information, scLLMDA effectively mitigates domain shifts and improves annotation robustness.

Despite its effectiveness, our framework has limitations. First, the computational cost of using large-scale pretrained DNA language models can be non-trivial [39], especially for datasets with hundreds of thousands of peaks. Efficient adaptation or compression of such models for scATAC-seq is a promising direction. Second, while we consider both local and global graph structure, current edge construction is based purely on similarity in embedding space. Incorporating biological priors—such as known enhancer-promoter interactions or transcription factor binding networks [40]—may further improve interpretability and performance.

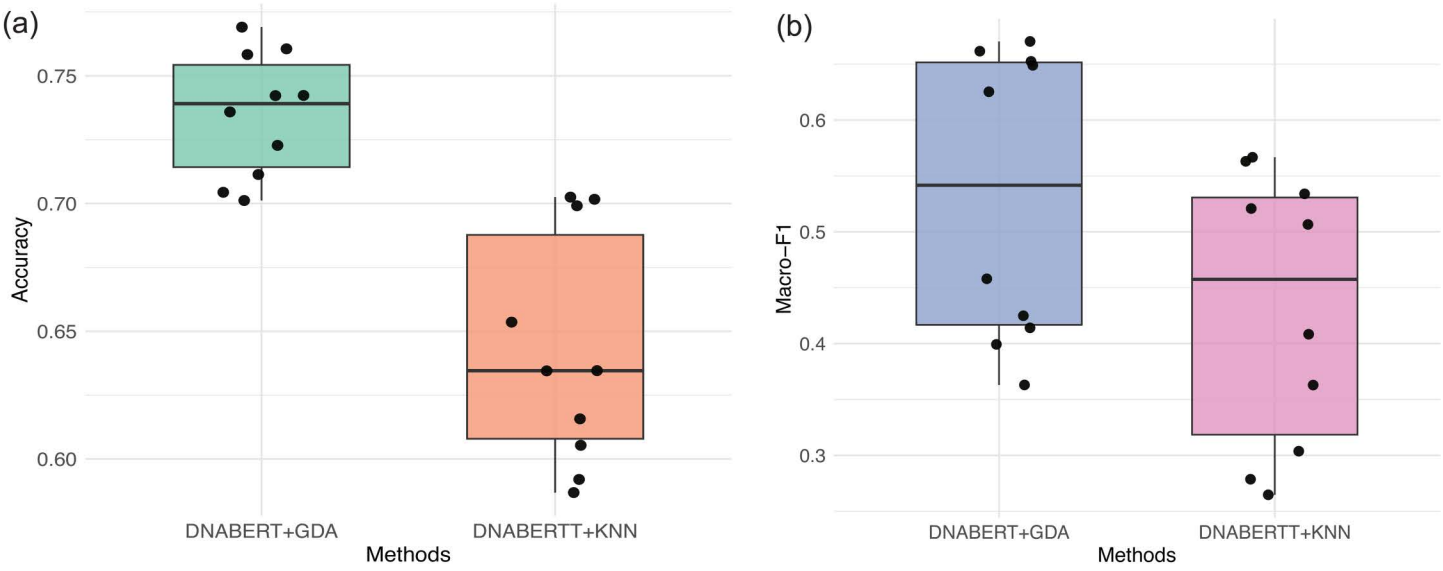

Fig 5. Ablation study comparing DNABERT+GDA and DNABERT+KNN across 10 cross-dataset annotation tasks. The boxplots display distributions of **(a)** Accuracy and **(b)** Macro-F1 score.

https://doi.org/10.1371/journal.pcbi.1014226.g005

Future work may also extend scLLMDA to multi-modal integration, where chromatin accessibility is combined with transcriptomic or epigenomic profiles [41] at the single-cell level. Additionally, applying scLLMDA to disease-related datasets or clinical samples could provide insight into pathological regulatory programs [42,43] and cell-type perturbations.

Conclusion

We presented scLLMDA, a framework that combines DNA language models with graph-based domain adaptation for accurate scATAC-seq cell type annotation. By capturing sequence semantics and cell–cell structural relationships, scLLMDA achieves robust performance across datasets. Future work will focus on improving efficiency, integrating biological priors, and extending the framework to multi-modal and clinical applications.

Author contributions

Conceptualization: He Yan, Guo Wei.

Data curation: Sheng Guan, Guo Wei.

Formal analysis: Sheng Guan.

Funding acquisition: He Yan, Guo Wei.

Methodology: Yan Liu.

Project administration: Long-Chen Shen.

Resources: He Yan, Long-Chen Shen.

Software: Yan Liu.

Supervision: Long-Chen Shen.

Validation: Ji-Peng Qiang.

Visualization: Yan Liu.

Writing – original draft: Yan Liu.

Writing – review & editing: Ji-Peng Qiang, Guo Wei.

References

1. Jin W, Ma J, Rong L, Huang S, Li T, Jin G, et al. Semi-automated IT-scATAC-seq profiles cell-specific chromatin accessibility in differentiation and peripheral blood populations. Nat Commun. 2025;16(1):2635. https://doi.org/10.1038/s41467-025-57931-2 PMID: 40097444
2. Baek S, Lee I. Single-cell ATAC sequencing analysis: From data preprocessing to hypothesis generation. Comput Struct Biotechnol J. 2020;18:1429–39. https://doi.org/10.1016/j.csbj.2020.06.012 PMID: 32637041
3. Lin Y, Wu T-Y, Wan S, Yang JYH, Wong WH, Wang YXR. scJoint integrates atlas-scale single-cell RNA-seq and ATAC-seq data with transfer learning. Nat Biotechnol. 2022;40(5):703–10. https://doi.org/10.1038/s41587-021-01161-6 PMID: 35058621
4. Yan X, Zheng R, Chen J, Li M. scNCL: transferring labels from scRNA-seq to scATAC-seq data with neighborhood contrastive regularization. Bioinformatics. 2023;39(8):btad505. https://doi.org/10.1093/bioinformatics/btad505 PMID: 37584660
5. Zhang Z, Yang C, Zhang X. scDART: integrating unmatched scRNA-seq and scATAC-seq data and learning cross-modality relationship simultaneously. Genome Biol. 2022;23(1):139. https://doi.org/10.1186/s13059-022-02706-x PMID: 35761403
6. Ma W, Lu J, Wu H. Cellcano: supervised cell type identification for single cell ATAC-seq data. Nat Commun. 2023;14(1):1864. https://doi.org/10.1038/s41467-023-37439-3 PMID: 37012226
7. Popescu MC, Balas VE, Perescu-Popescu L, Mastorakis N. Multilayer perceptron and neural networks. WSEAS Trans Circuit Syst. 2009;8(7):579–88.
8. Jiang Y, Hu Z, Lynch AW, Jiang J, Zhu A, Zeng Z, et al. scATAnno: automated cell type annotation for single-cell ATAC sequencing data. bioRxiv. 2023;2023-06.
9. Zhang K, Zemke NR, Armand EJ, Ren B. A fast, scalable and versatile tool for analysis of single-cell omics data. Nat Methods. 2024;21(2):217–27. https://doi.org/10.1038/s41592-023-02139-9 PMID: 38191932
10. Korsunsky I, Millard N, Fan J, Slowikowski K, Zhang F, Wei K, et al. Fast, sensitive and accurate integration of single-cell data with Harmony. Nat Methods. 2019;16(12):1289–96. https://doi.org/10.1038/s41592-019-0619-0 PMID: 31740819
11. Taunk K, De S, Verma S, Swetapadma A. A brief review of nearest neighbor algorithm for learning and classification. In: 2019 International Conference on Intelligent Computing and Control Systems (ICCS). IEEE; 2019. p. 1255–60.
12. Chen X, Chen S, Song S, Gao Z, Hou L, Zhang X, et al. Cell type annotation of single-cell chromatin accessibility data via supervised Bayesian embedding. Nat Mach Intell. 2022;4(2):116–26. https://doi.org/10.1038/s42256-021-00432-w
13. Goan E, Fookes C. Bayesian neural networks: An introduction and survey. Case Studies in Applied Bayesian Data Science: CIRM Jean-Morlet Chair, Fall 2018. 2020. p. 45–87.
14. Yuan H, Kelley DR. scBasset: sequence-based modeling of single-cell ATAC-seq using convolutional neural networks. Nat Methods. 2022;19(9):1088–96. https://doi.org/10.1038/s41592-022-01562-8 PMID: 35941239
15. Zeng Y, Luo M, Shangguan N, Shi P, Feng J, Xu J, et al. Deciphering cell types by integrating scATAC-seq data with genome sequences. Nat Comput Sci. 2024;4(4):285–98. https://doi.org/10.1038/s43588-024-00622-7 PMID: 38600256
16. Dziugaite GK, Roy DM, Ghahramani Z. Training generative neural networks via maximum mean discrepancy optimization. arXiv preprint arXiv:150503906. 2015.
17. Fang R, Preissl S, Li Y, Hou X, Lucero J, Wang X, et al. Comprehensive analysis of single cell ATAC-seq data with SnapATAC. Nat Commun. 2021;12(1):1337. https://doi.org/10.1038/s41467-021-21583-9 PMID: 33637727
18. Preissl S, Fang R, Huang H, Zhao Y, Raviram R, Gorkin DU, et al. Single-nucleus analysis of accessible chromatin in developing mouse forebrain reveals cell-type-specific transcriptional regulation. Nat Neurosci. 2018;21(3):432–9. https://doi.org/10.1038/s41593-018-0079-3 PMID: 29434377
19. Cusanovich DA, Hill AJ, Aghamirzaie D, Daza RM, Pliner HA, Berletch JB, et al. A single-cell atlas of in vivo mammalian chromatin accessibility. Cell. 2018;174(5):1309–24.
20. Cusanovich DA, Daza R, Adey A, Pliner HA, Christiansen L, Gunderson KL, et al. Multiplex single cell profiling of chromatin accessibility by combinatorial cellular indexing. Science. 2015;348(6237):910–4. https://doi.org/10.1126/science.aab1601 PMID: 25953818
21. Kimmel JC, Kelley DR. Semisupervised adversarial neural networks for single-cell classification. Genome Res. 2021;31(10):1781–93. https://doi.org/10.1101/gr.268581.120 PMID: 33627475
22. Cui L, Wang F, Li H, Liu Q, Zhou M, Wang G. annATAC: automatic cell type annotation for scATAC-seq data based on language model. BMC Biol. 2025;23(1):145. https://doi.org/10.1186/s12915-025-02244-5 PMID: 40437567
23. Tian L, Xie Y, Xie Z, Tian J, Tian W. AtacAnnoR: a reference-based annotation tool for single cell ATAC-seq data. Brief Bioinform. 2023;24(5):bbad268. https://doi.org/10.1093/bib/bbad268 PMID: 37497729
24. Li S, Huang Y, Chen S. MINGLE: a mutual information-based interpretable framework for automatic cell type annotation in single-cell chromatin accessibility data. Genome Biol. 2025;26(1):162. https://doi.org/10.1186/s13059-025-03603-9 PMID: 40500763

25. Haghverdi L, Lun ATL, Morgan MD, Marioni JC. Batch effects in single-cell RNA-sequencing data are corrected by matching mutual nearest neighbors. Nat Biotechnol. 2018;36(5):421–7. https://doi.org/10.1038/nbt.4091 PMID: 29608177
26. Chen H, Lareau C, Andreani T, Vinyard ME, Garcia SP, Clement K, et al. Assessment of computational methods for the analysis of single-cell ATAC-seq data. Genome Biol. 2019;20(1):241. https://doi.org/10.1186/s13059-019-1854-5 PMID: 31739806
27. Liu J, Yang M, Yu Y, Xu H, Li K, Zhou X. Large language models in bioinformatics: applications and perspectives. arXiv preprint arXiv:240104155. 2024.
28. Zhou Z, Ji Y, Li W, Dutta P, Davuluri R, Liu H. Dnabert-2: Efficient foundation model and benchmark for multi-species genome. arXiv preprint arXiv:230615006. 2023.
29. Sennrich R, Haddow B, Birch A. Neural machine translation of rare words with subword units. In: Proceedings of the 54th annual meeting of the association for computational linguistics (volume 1: long papers). 2016. p. 1715–25.
30. Nair S, Kim DS, Perricone J, Kundaje A. Integrating regulatory DNA sequence and gene expression to predict genome-wide chromatin accessibility across cellular contexts. Bioinformatics. 2019;35(14):i108–16. https://doi.org/10.1093/bioinformatics/btz352 PMID: 31510655
31. Kipf TN, Welling M. Semi-supervised classification with graph convolutional networks. arXiv preprint arXiv:160902907. 2016.
32. Li Q, Han Z, Wu X. Deeper Insights Into Graph Convolutional Networks for Semi-Supervised Learning. AAAI. 2018;32(1). https://doi.org/10.1609/aaai.v32i1.11604
33. Perozzi B, Al-Rfou R, Skiena S. Deepwalk: Online learning of social representations. In: Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining. 2014. p. 701–10.
34. Qiu J, Dong Y, Ma H, Li J, Wang K, Tang J. Network embedding as matrix factorization: Unifying deepwalk, line, pte, and node2vec. In: Proceedings of the eleventh ACM international conference on web search and data mining. 2018. p. 459–67.
35. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN. Attention is all you need. Adv Neural Inf Process Syst. 2017;30.
36. Zhang Z, Cui P, Zhu W. Deep Learning on Graphs: A Survey. IEEE Trans Knowl Data Eng. 2022;34(1):249–70. https://doi.org/10.1109/tkde.2020.2981333
37. Ganin Y, Ustinova E, Ajakan H, Germain P, Larochelle H, Laviolette F. Domain-adversarial training of neural networks. J Mach Learn Res. 2016;17(59):1–35.
38. Bishop CM, Nasrabadi NM. Pattern recognition and machine learning. vol. 4. Springer; 2006.
39. Xu C, McAuley J. A Survey on Model Compression and Acceleration for Pretrained Language Models. AAAI. 2023;37(9):10566–75. https://doi.org/10.1609/aaai.v37i9.26255
40. Fulco CP, Nasser J, Jones TR, Munson G, Bergman DT, Subramanian V, et al. Activity-by-contact model of enhancer-promoter regulation from thousands of CRISPR perturbations. Nat Genet. 2019;51(12):1664–9. https://doi.org/10.1038/s41588-019-0538-0 PMID: 31784727
41. Lim J, Park C, Kim M, Kim H, Kim J, Lee D-S. Advances in single-cell omics and multiomics for high-resolution molecular profiling. Exp Mol Med. 2024;56(3):515–26. https://doi.org/10.1038/s12276-024-01186-2 PMID: 38443594
42. Satpathy AT, Granja JM, Yost KE, Qi Y, Meschi F, McDermott GP, et al. Massively parallel single-cell chromatin landscapes of human immune cell development and intratumoral T cell exhaustion. Nat Biotechnol. 2019;37(8):925–36. https://doi.org/10.1038/s41587-019-0206-z PMID: 31375813
43. Sundaram L, Kumar A, Zatzman M, Salcedo A, Ravindra N, Shams S, et al. Single-cell chromatin accessibility reveals malignant regulatory programs in primary human cancers. Science. 2024;385(6713):eadk9217. https://doi.org/10.1126/science.adk9217 PMID: 39236169