A TCN-CNN Hybrid Network for the Inference of Single-cell Gene Regulatory Networks

Siying Du, Kui Jin, Mingjing Tang, Yanqiong Duan, Wei Gao

Abstract—**Gene regulatory networks (GRNs) elucidate the mechanisms underlying gene expression regulation and play a pivotal role in understanding cellular functions and organismal development. The emergence of single-cell transcriptome sequencing technology has significantly improved the accuracy of GRN inference and the ability to analyze cell type-specific characteristics. However, the inherent sparsity, noise, and dropout events in single-cell transcriptome data present challenges in accurately identifying regulatory relationships. To address these limitations, we propose TCGRN, a hybrid deep learning framework combining Temporal Convolutional Networks (TCN) and Convolutional Neural Networks (CNN) for single-cell GRN inference. Our method initially preprocesses raw gene expression data into correlation vectors and images, which are subsequently fed into TCN to model temporal dependencies and CNN to extract spatial features, respectively. Subsequently, the attention mechanism dynamically integrates these multi-modal features for robust regulatory relationship prediction. We compare TCGRN with unsupervised and supervised methods on mouse hematopoietic stem cell datasets of various lineages and scales, and apply TCGRN on human datasets of different scales for experiments. The experimental results demonstrate that TCGRN outperforms existing approaches in prediction accuracy while demonstrating a certain degree of generalizability.**

Index Terms—**Gene regulatory networks, single-cell RNA sequencing, hybrid network, deep learning**

I. Introduction

Gene regulatory networks (GRNs) consist of gene interactions within a cell or genome. These networks define regulatory relationships between transcription factors (TFs) and their target genes, governing transcriptional activity and modulating cellular processes [1]. Deciphering these biological GRNs enables the extraction of critical gene interaction patterns, which are fundamental for elucidating molecular mechanisms underlying key biological events such as cell proliferation, DNA repair, and apoptosis [2-4]. With advancements in single-cell RNA sequencing (scRNA-seq) technology, numerous analytical approaches have been developed and applied to scRNA-seq data analysis [5-7]. These approaches primarily aim to resolve gene expression heterogeneity at single-cell resolution, thereby uncovering cell type- and state-specific GRNs. Nevertheless, due to inherent high dimensionality, technical noise, and data sparsity in scRNA-seq datasets, the precise reconstruction of GRNs remains a substantial challenge.

GRNs serve as a critical tool for characterizing gene-gene interactions and transcriptional regulation processes [8]. The accurate reconstruction of GRNs plays a pivotal role in elucidating gene functions and cellular mechanisms [9, 10]. To overcome existing limitations in network inference, numerous computational methods have been developed for GRN reconstruction from gene expression data [11-15], primarily falling into unsupervised and supervised learning categories. Among these, unsupervised methods that deduce regulatory relationships solely from gene expression patterns [16-19] have gained widespread adoption in GRN inference. Specifically, information-theoretic approaches are widely used due to their simplicity, low sampling requirements, and minimal need for additional data preprocessing. They commonly utilize mutual information as a metric, such as the correlation network model proposed by Butte et al. [20], which employed mutual information to quantify gene associations. The NARROMI algorithm [21] integrates recursive optimization based on ordinary differential equations with mutual information, surpassing many existing methods in performance. However, since correlation measures inherently capture bidirectional relationships, networks inferred through these approaches frequently yield undirected relationships, as seen in PIDC [12] and similar methods. To overcome parameter optimization difficulties, machine learning-based solutions have been proposed for network inference, primarily employing regression and classification techniques like those implemented in GENIE3 [11] and GRNBoost2 [22].

Compared to unsupervised approaches, supervised methodologies can leverage known interactions for training, often outperforming unsupervised approaches [23, 24]. In recent years, substantial progress in deep learning for natural language processing has spurred the emergence of numerous innovative deep learning frameworks [25-27]. Specifically, CNNC [23] represents gene pairs as images—in the form of histograms—and applies convolutional neural network (CNN) to infer the relationships between the varying expression levels illustrated in these images. However, it overlooks the role of neighboring images. DeepDRIM [28] builds on CNNC by incorporating neighborhood images for

Manuscript received December 3, 2024; revised July 19, 2025.

This work is supported by the National Natural Science Foundation of China (No. U2002204); the Key Project of Basic Research in Yunnan Province (No. 202501AS070007).

Siying Du is a graduate student of school of Information Science and Technology, Yunnan Normal University, Kunming 650500, China (e-mail: 1960089397@qq.com).

Kui Jin is a graduate student of school of Information Science and Technology, Yunnan Normal University, Kunming 650500, China (e-mail: kui_jin@163.com).

Mingjing Tang is a professor of school of Information Science and Technology, Yunnan Normal University, Kunming 650500, China (corresponding author to provide e-mail: tmj@ynnu.edu.cn).

Yanqiong Duan is a teacher of school of Information Science and Technology, Yunnan Normal University, Kunming 650500, China (e-mail: 2548750708@qq.com).

Wei Gao is a teacher of school of Mathematics, Hohai University, Nanjing 210098, China (e-mail: gaowei@hhu.edu.cn).

each gene pair to reduce false positives arising from transitive interactions between genes, but it requires substantial computational resources and a high-performance experimental environment. Zhao et al. [29] develop DGRNS, a hybrid deep learning framework that combines gated recurrent units (GRU) with CNN to process encoded raw data and identify putative gene interactions. However, feeding GRU outputs into CNN may result in the loss of certain biological information contained in the original data. Additionally, for CNN frameworks that learn spatial features, image-formatted data may prove more suitable.

Some of the aforementioned supervised learning methods do not take into account the time characteristics of the data, while others rely on a single channel to extract both time and spatial information, which may result in the mutual overwriting of some information. To address these issues, we introduce TCGRN, a hybrid network that combines Temporal Convolutional Network (TCN) with CNN for inferring single-cell gene regulatory networks from transcriptomic data. First, raw data undergoes dual processing: (1) transformation into correlation vectors preserving temporal dynamics, and (2) conversion to image representations augmented with neighborhood contexts (images of gene pairs sharing common genes) to encapsulate spatial relationships. Second, a dual-channel architecture simultaneously processes these modalities: the TCN channel extracts temporal patterns from correlation vectors, while the CNN channel deciphers hierarchical spatial features from genomic images. These features are then adaptively fused via the attention mechanism for precise GRN inference. Third, comprehensive benchmarking across multi-species, multi-lineage, and multi-scale datasets validates TCGRN's superior performance against state-of-the-art methods. The key contributions of this paper are outlined below:

1) The original data is processed from two perspectives: on one hand, it is converted into correlation vectors to serve as input for learning temporal features; on the other hand, it is transformed into images, and gene pair images (neighborhood images) that share a common gene with the target pair are incorporated as input for learning spatial features.
2) A dual-channel model is employed, utilizing TCN to capture time features and CNN to extract spatial features. Subsequently, the attention mechanism is used to effectively integrate these features, and hence infer the gene regulatory network.
3) Experimental results demonstrate the effectiveness of TCGRN in GRN inference by comparing it with other methods across datasets of varying species, lineages, and sizes.

II. Materials and Methods

This section provides a detailed exposition of the TCGRN framework, encompassing its architectural design and methodological implementation pipeline.

A. Datasets

The experimental data were curated from scRNA-seq benchmarks in the BEELINE framework [30], with original data sourced from [31]. Our evaluation is based on four authentic single-cell transcriptomic datasets: three lineage-specific datasets of mouse hematopoietic stem cells (mHSC), representing the erythroid (E), lymphoid (L), and granulocyte-macrophage (GM) lineages, and one human embryonic stem cell (hESC) dataset [32], which are used to assess cross-species generalizability.

For each independent dataset, we use a standard network based on non-specific ChIP-seq data [33-35]. Genes expressed in fewer than 10% of cells are filtered out, followed by variance stabilization and P-value calculation (using a threshold of $P < 0.01$), and then sorted in ascending order by P-value. From these datasets, we extract the top 500 and 1000 most highly variable genes for downstream analysis. Datasets are annotated using the convention 'original dataset name-network scale'. Detailed information about each dataset, including the number of genes and cells, is provided in Table 1.

B. Overview

We propose a hybrid TCN-CNN network, TCGRN, for inferring single-cell gene regulatory networks. The framework begins by transforming raw single-cell transcriptomic data into both correlation vectors and image representations, enabling comprehensive feature extraction from temporal and spatial perspectives. The architecture employs parallel TCN and CNN branches to process these distinct data modalities, subsequently utilizing an attention mechanism for feature fusion and GRN prediction. TCGRN frames GRN inference as a binary classification task to determine whether gene pairs have regulatory relationships. Fig. 1 illustrates the overall TCGRN framework, which comprises four main steps: (1) data preprocessing; (2) time feature learning through TCN; (3) spatial feature extraction using CNN; and (4) feature fusion and prediction.

C. Data Preprocessing

TCGRN preprocesses data from two distinct perspectives. On one hand, the original data is transformed into correlation vectors; on the other hand, each gene pair is converted into histogram images that are combined with neighborhood images for subsequent spatial feature extraction. Since the data we use is closely tied to the cell differentiation process, and each cell's differentiation stage reflects underlying time information, we employ pseudo-time to describe the relative progression of cells during differentiation. Pseudo-time analysis typically integrates a variety of computational methods and algorithms to infer cell differentiation trajectories. In this paper, we employ SlingShot [36] to compute pseudo-time; it maps cells into a low-dimensional space, constructs trajectories within that space, and thereby clearly determines each cell's position in the differentiation process.

In processing the input data for the model, we are inspired by the data handling methods in DGRNS [29]. We extract a series of ordered correlation vectors that capture the relationships between gene pairs, using statistical criteria to quantify their correlations. A sliding window mechanism is employed to preserve pseudo-time information. This is elaborated in Equations (1)-(5).

$$Z_{u,m} = \left\{ Z_{u,ml_u+1}, Z_{u,ml_u+2}, ..., Z_{u,ml_u+s} \right\} \quad (1)$$

$$Z_{v,n} = \left\{ Z_{v,ml_u+nl_v+1}, Z_{v,ml_u+nl_v+2}, ..., Z_{v,ml_u+nl_v+s} \right\} \quad (2)$$

Table 1. Detailed Information of Single-Cell Transcriptomic Datasets

Datasets	Genes	Cells	Number of TFs	Regulation	Network density
mHSC-GM-500	108	889	17	154	0.0839
mHSC-L-500	114	847	22	147	0.0586
mHSC-E-500	169	1071	21	279	0.0786
hESC-500	188	758	22	236	0.0571
mHSC-GM-1000	321	889	48	566	0.0367
mHSC-L-1000	403	847	63	845	0.0333
mHSC-E-1000	355	1071	45	677	0.0424
hESC-1000	541	758	55	808	0.0272

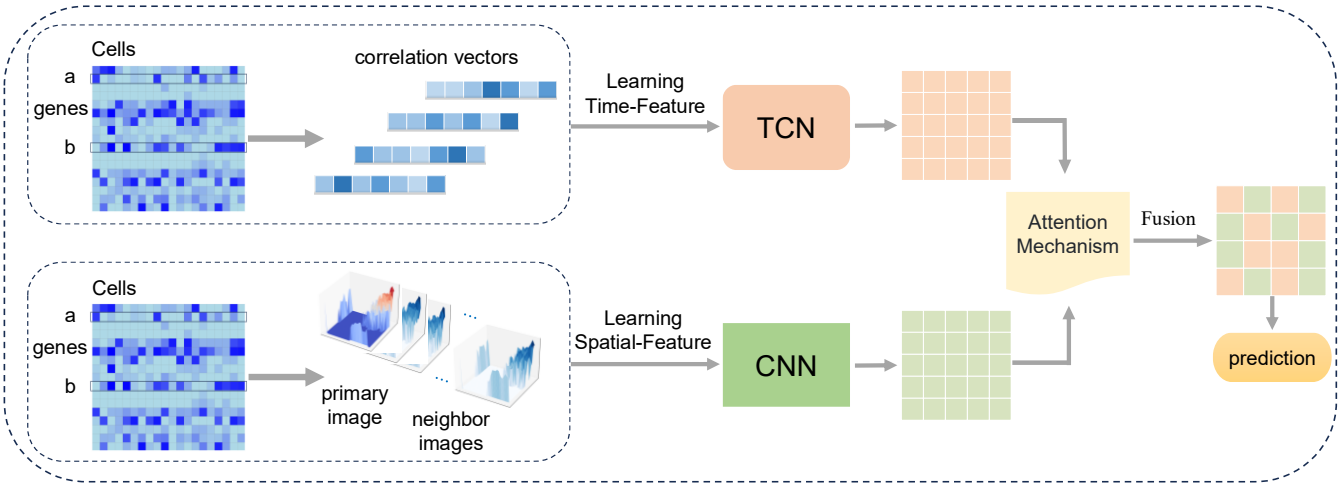

Fig. 1. TCGRN Framework. Initially, the original data is converted into correlation vectors and input into TCN to capture time features. Simultaneously, gene pairs are transformed into histogram images and combined with neighborhood images, which are input into CNN to capture spatial features. These features are subsequently integrated using the attention mechanism to infer the gene regulatory network.

$$\rho_{u,v} = Pearson\left(Z_{u,m}, Z_{v,m}\right) = \frac{\mathrm{cov}\left(Z_{u,m}, Z_{v,m}\right)}{\sigma\left(Z_{u,m}\right)\sigma\left(Z_{v.m}\right)} \quad (3)$$

$$\rho_m = \left(\rho_{m,0}, \rho_{m,1}, ..., \rho_{m,i-1}\right) \quad (4)$$

$$\rho_{m,n} = \left(\rho_0, \rho_1, ..., \rho_{j-1}\right) \quad (5)$$

where Z represents the gene expression vector arranged by pseudo-time, $Z_{u,m}$ denotes the data segment of TF u captured by the sliding window m, l_u expresses the interval of the sliding window corresponding to the TF, $Z_{v,n}$ represents the data segment of target gene v captured by sliding window n, and l_v represents the interval of the sliding window for the target gene. Furthermore, $m \in \{0,1,2,\cdots,i-1\}$, i is the number of related vectors, $n \in \{0,1,2,\cdots,j-1\}$, j is the length of each vector. For each intercepted data segment of TF and target gene, their Pearson correlation coefficients are calculated respectively to characterize the correlation between gene pairs.

Simultaneously, we convert gene pairs into histogram images, inspired by the DeepDRIM [28], considering potential neighboring pairs of genes. We generate 32x32 pixel images ($G_{u,v}$) for genes u and v, along with neighborhood images for genes u and v, which include: (1) $\{G_{u,p_1}, G_{u,p_2}, \cdots, G_{u,p_r}, G_{v,q_1}, G_{v,q_2}, \cdots, G_{v,q_r}\}$, where $(p_1, p_2, \cdots, p_r)$ and $(q_1, q_2, \cdots, q_r)$ are the top r genes with strong positive covariance with genes u and v, respectively (r defaults to 10); (2) two self-images $G_{u,u}$ and $G_{v,v}$. Specifically, we generate $2r$+2 neighborhood images of genes with strong positive covariance with the gene pair and input these images along with the main image into CNN to comprehensively extract spatial information from the data.

D. Time feature learning

To capture temporal dynamics in the data, we employ TCN, an architecture that combines parallel processing capabilities with dilated convolutional operations. TCN efficiently models temporal dependencies through its unique ability to process sequences in parallel while systematically expanding the receptive field using exponentially increasing dilation rates. The dilated convolution can be expressed as Equation (6).

$$y[t] = \sum_{i=0}^{k-1} w[i] \cdot x[t - d \cdot i] \quad (6)$$

Here, $y[t]$ represents the output at time step t, w is the convolution kernel, $w[i]$ denotes the i -th weight of the kernel, and $x[t - d \cdot i]$ states the value of the input sequence at time step $t - d \cdot i$. The size of the convolutional kernel k determines the number of input elements involved in each convolution operation, while the dilation rate d dictates the number of elements the kernel skips over in the input

sequence. Dilated convolution expands the receptive field by introducing gaps in the kernel, allowing for the capture of longer time dependencies. In this work, we use three different dilation rates to capture time dependencies at various scales.

E. Spatial Feature Learning

We deliberately extract spatial features from an image perspective, as image data possesses distinct spatial properties and structures that CNN can effectively exploit for efficient and accurate feature extraction. The filters in CNN share the same weights across all positions in the image, which reduces the number of model parameters, improves training efficiency, and enhances the model's translational invariance. By stacking convolutional and pooling layers, CNN can progressively extract features at low, mid, and high levels from images.

In our framework, the CNN architecture consists of two convolutional layers with 3×3 filters, generating feature maps with 32 channels, a max pooling layer with a 2×2 pool size, and a fully connected layer with 128 nodes, as illustrated in Fig. 2.

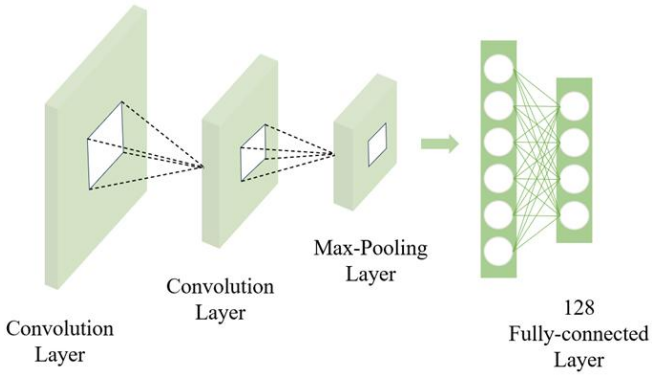

Fig. 2. Main Structure of the CNN

F. Feature Fusion

To effectively integrate spatiotemporal features for GRN inference, we employ an attention mechanism. Specifically, we use dot-product attention to compute the attention weights for the outputs of both the TCN and CNN, as shown in Equation (7). In this equation, a denotes the attention weight, $\boldsymbol{e}$ is the TCN output, $\boldsymbol{c}$ is the CNN output, and the "·" symbol denotes the dot product operation.

$$a = softmax\left(\frac{\boldsymbol{e}\cdot\boldsymbol{c}^{T}}{\|\boldsymbol{e}\|\|\boldsymbol{c}\|}\right) \tag{7}$$

The computed attention weights are then applied to $\boldsymbol{e}$ and $\boldsymbol{c}$ to generate their weighted feature representations, as demonstrated in Equations (8) and (9). Here, $\boldsymbol{h_1}$ stands for the weighted output of the TCN and $\boldsymbol{h_2}$ for that of the CNN, with "$\odot$" indicating element-wise multiplication.

$$\boldsymbol{h_1} = a \odot \boldsymbol{e} \tag{8}$$

$$\boldsymbol{h_2} = a \odot \boldsymbol{c} \tag{9}$$

These weighted features are subsequently concatenated to produce the final fused feature representation. Finally, a sigmoid activation function is applied, and the model is trained using binary cross-entropy loss as described in Equation (10).

$$BCE = -\frac{1}{N}\sum_{i=1}^{N} y_i \log(p(y_i)) + (1-y_i)\log(1-p(y_i)) \tag{10}$$

where N denotes the number of samples, y_i represents the label of sample i, and $p(y_i)$ represents the probability of sample i being predicted as a positive label by the sigmoid function.

G. Model Pseudocode

For a comprehensive overview of the TCGRN framework, Algorithm 1 formally presents the complete computational workflow, where $\boldsymbol{P_{u,v}}$ denotes the predicted probability of a regulatory interaction between gene u and gene v.

Algorithm 1: Overview of the TCGRN Algorithm

Algorithm 1: Gene Regulatory Network Inference Algorithm Based on Hybrid TCN-CNN Architecture.

Input: Correlation vector $\rho_{m,n}$, histogram images and neighborhood images of gene pairs $G_{u,v}$, $\{G_{u,p_1}, G_{u,p_2}, ..., G_{u,p_r}, G_{v,q_1}, G_{v,q_2}, ..., G_{v,q_r}\}$, $G_{u,u}$, $G_{v,v}$

Output: Prediction of gene regulatory relationships $\boldsymbol{P_{u,v}}$

1: begin
2: Initialize parameters
3: Extract time features $\boldsymbol{e}$ using TCN
4: Extract spatial features $\boldsymbol{c}$ using CNN
5: Obtain attention weights a for $\boldsymbol{e}$ and $\boldsymbol{c}$
6: Apply a to weight $\boldsymbol{e}$ and obtain weighted output $\boldsymbol{h_1}$
7: Apply a to weight $\boldsymbol{c}$ and obtain weighted output $\boldsymbol{h_2}$
8: Concatenate $\boldsymbol{h_1}$ and $\boldsymbol{h_2}$
9: Output gene regulatory relationship predictions $\boldsymbol{P_{u,v}}$ through a fully connected layer
10: Calculate loss using binary cross-entropy and update model parameters
11: end

III. Experimental and Results

To evaluate TCGRN's performance, we conduct comparative experiments with state-of-the-art unsupervised and supervised GRN inference methods across diverse mouse hematopoietic stem cell datasets. To further assess each component's contribution, we perform a series of ablation experiments to validate the effectiveness of the attention mechanism for feature integration and confirm the suitability of image-formatted data for CNN to extract spatial features. Additionally, we extend our evaluation to human datasets of different sizes, which underscores the generalizability of TCGRN. Finally, the datasets are partitioned into training, testing, and validation sets at multiple ratios to examine the impact of different data splitting strategies.

A. Model Training and Validation

Based on the standardized reference network, we designate documented TF target gene pairs as positive samples, while gene pairs lacking known regulatory interactions or annotations are classified as negative samples. As indicated by the network density in table 1, the GRN is notably sparse, with positive instances being substantially outnumbered by

negatives. To comprehensively assess the model's performance, we partition each dataset into training, validation, and test sets in a 3:1:1 ratio, ensuring that the positive-to-negative distribution remains consistent with that of the original dataset.

B. Evaluation Metrics

Given the inherent imbalance between positive and negative samples in sparse GRN, we evaluate model performance using both the area under the receiver operating characteristic curve (AUROC) and the area under the precision-recall curve (AUPRC).

AUROC evaluates a model's binary classification capability by measuring the true positive rate (TPR) against the false positive rate (FPR) across all classification thresholds. Equation (11) presents the formula for computing AUROC.

$$AUROC = \int_0^1 TPR(FPR)dFPR \tag{11}$$

AUPRC is used to evaluate model performance under imbalanced class conditions by computing the area under the precision-recall curve, which reflects the classifier's effectiveness in predicting positive samples.

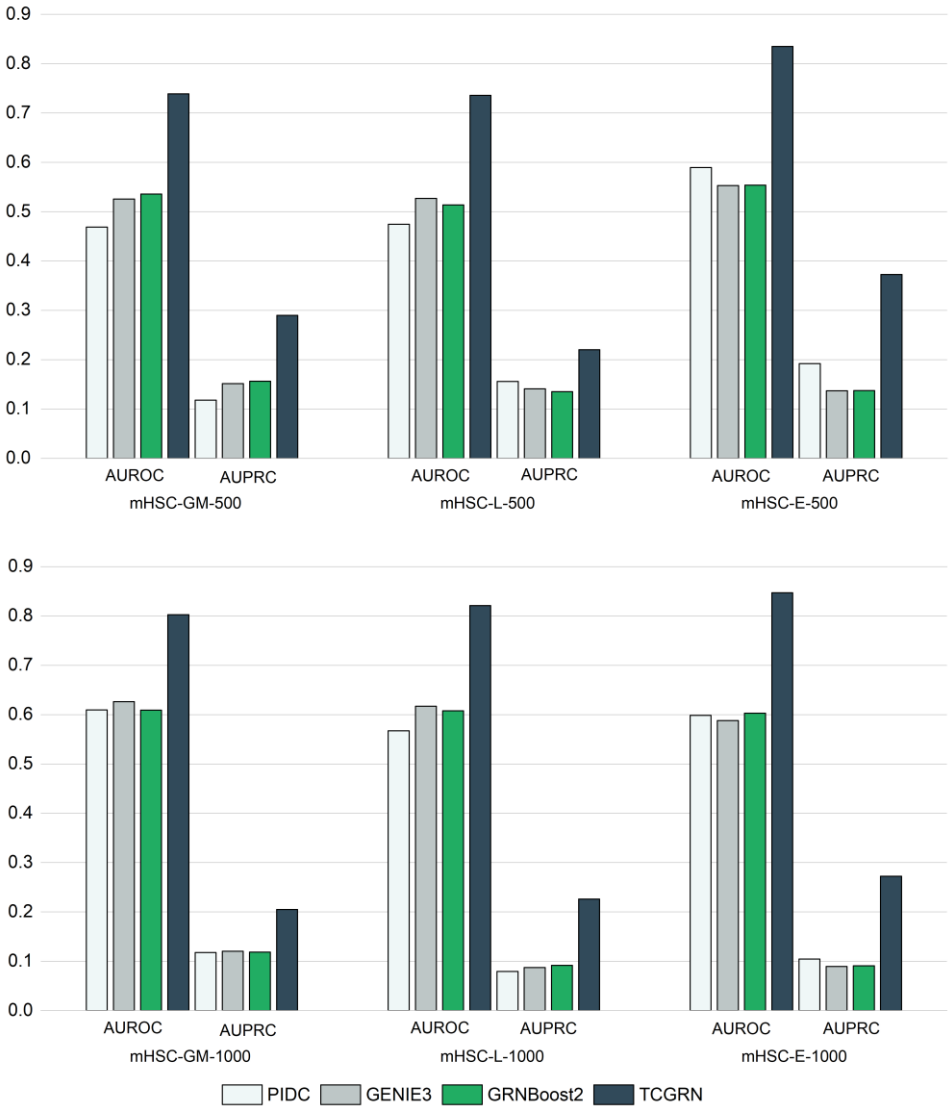

Fig. 3. A comparison of TCGRN and unsupervised methods across all datasets.

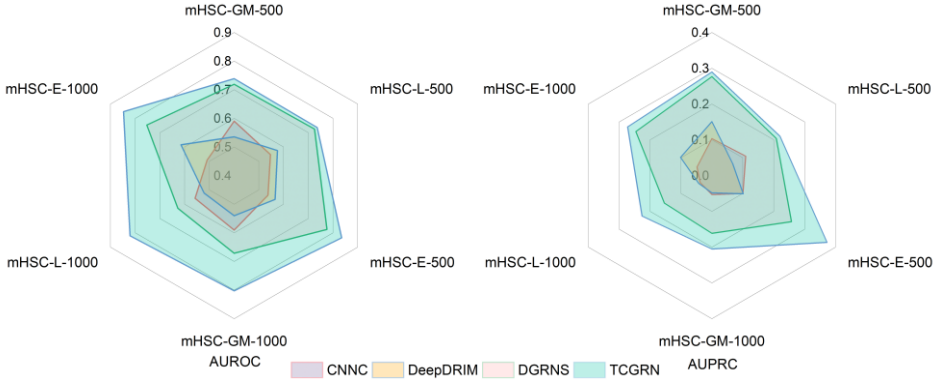

Fig. 4. A comparison of TCGRN and supervised methods across all datasets.

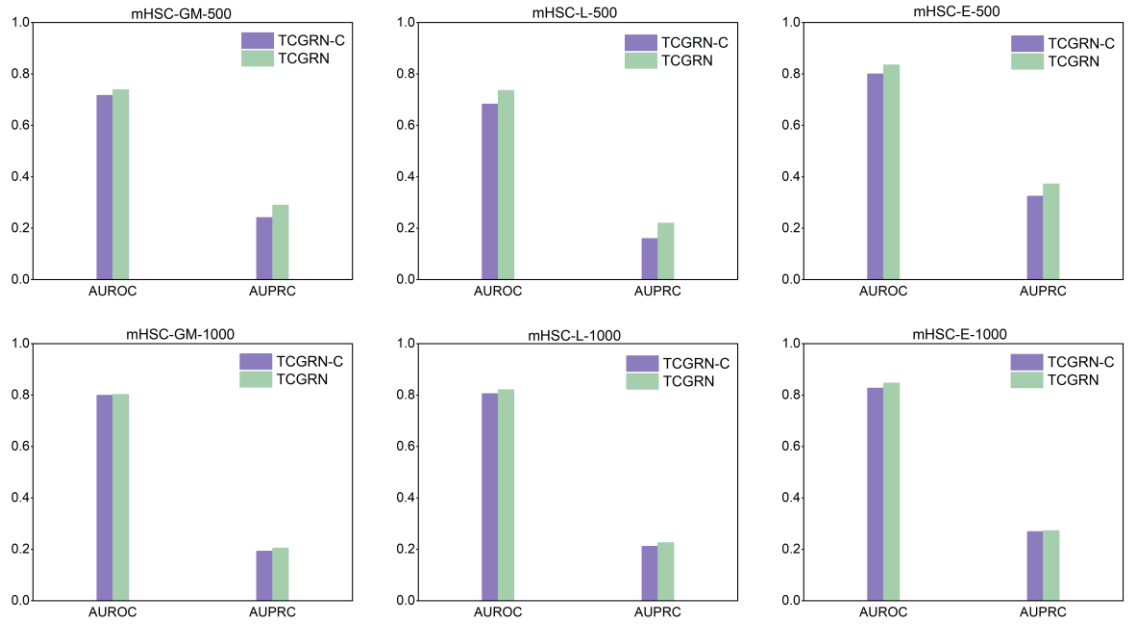

Fig. 5. Comparison of TCGRN-C and TCGRN methods.

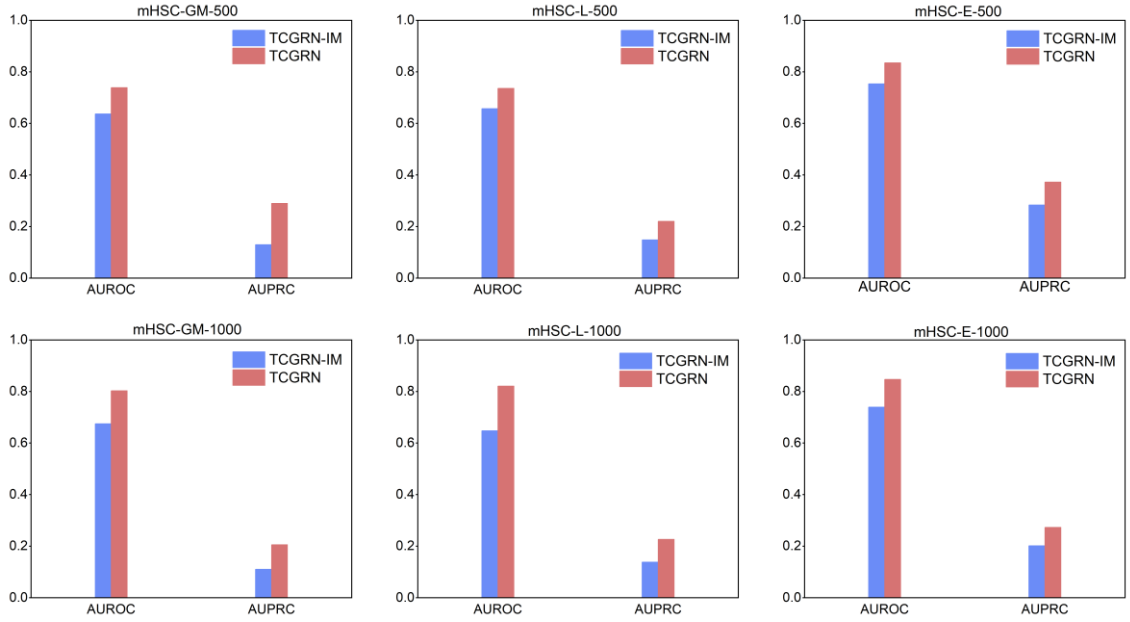

Fig. 6. Comparison of TCGRN-IM and TCGRN methods.

C. Comparison with Baseline Methods

To assess the effectiveness of TCGRN in GRN inference, we employ three unsupervised learning algorithms and three supervised learning algorithms as benchmark methods.

Among various unsupervised algorithms, we select PIDC [12], GENIE3 [11], and GRNBoost2 [22] as comparison models. PIDC is based on multivariate information theory and partial information decomposition (PID), and it identifies regulatory interactions among genes by analyzing the statistical dependencies among gene triplets in single-cell gene expression data. GENIE3 is a random forest-based ensemble method that infers regulatory relationships through feature importance scores derived from regression trees. GRNBoost2 is a regression-based inference method that predicts the expression profile of each gene by training tree-based regression models, generating partial GRN. These resulting partial GRNs are subsequently ranked and integrated based on their importance, ultimately yielding the complete GRN.

The comparison results between TCGRN and unsupervised methods on six mouse hematopoietic stem cell datasets are shown in Fig. 3. As illustrated in the figure, TCGRN achieves the best performance across all six datasets in terms of both AUROC and AUPRC metrics. On the mESC-E-500 dataset, the AUROC score of TCGRN is approximately 24% higher than that of the second-best unsupervised method. Due to the class imbalance in the datasets, all methods exhibit relatively low AUPRC values. However, on the mESC-E-500 dataset, the AUPRC of TCGRN still exceeds that of the second-best unsupervised method by around 18%. These experimental results demonstrate that TCGRN effectively learns data features, thereby enhancing the accuracy of GRN inference.

Among various supervised GRN inference methods, we select CNNC [23], DeepDRIM [28], and DGRNS [29] as comparative models. CNNC transforms each gene pair into a

histogram and employs CNN to analyze expression-level relationships between genes. DeepDRIM extends the conventional histogram representation of gene pairs by incorporating neighborhood images for each pair, thereby mitigating false positives induced by transitive gene interactions. DGRNS adopts a hybrid learning framework that encodes the original data and integrates recurrent neural networks (RNNs) with CNN to determine whether gene pairs exhibit regulatory relationships.

Fig. 4 presents the comparative results between TCGRN and supervised methods on six mouse hematopoietic stem cell datasets. The experimental results demonstrate that TCGRN achieves performance improvements in both AUROC and AUPRC metrics across all datasets when compared to other models. On the mHSC-GM-1000 dataset, TCGRN achieves an AUROC approximately 13% higher than the second-best method, DGRNS. On the mHSC-E-500 dataset, TCGRN's AUPRC is about 11.5% higher than that of DGRNS. These results validate the effectiveness of TCGRN in GRN inference.

D. The role of the attention mechanism

To verify the effectiveness of the attention mechanism in integrating time and spatial features to enhance model performance, we design a set of comparative experiments. Specifically, we replace the attention mechanism in the feature integration module of TCGRN with simple concatenation and compare this method with TCGRN on mouse hematopoietic stem cell datasets of varying lineage sizes, using AUROC and AUPRC as evaluation metrics.

The experimental results are presented in Fig. 5, where TCGRN-C denotes the model in which the attention mechanism is replaced by simple concatenation. As shown in the figure, TCGRN achieves higher AUROC and AUPRC values than TCGRN-C on all experimental datasets. For example, on the mHSC-L-500 dataset, the AUROC of TCGRN is approximately 5% higher than that of TCGRN-C, and its AUPRC is improved by about 6%. These results demonstrate that the attention mechanism can better capture the complex relationships between temporal and spatial features, thereby enhancing the model's inference performance.

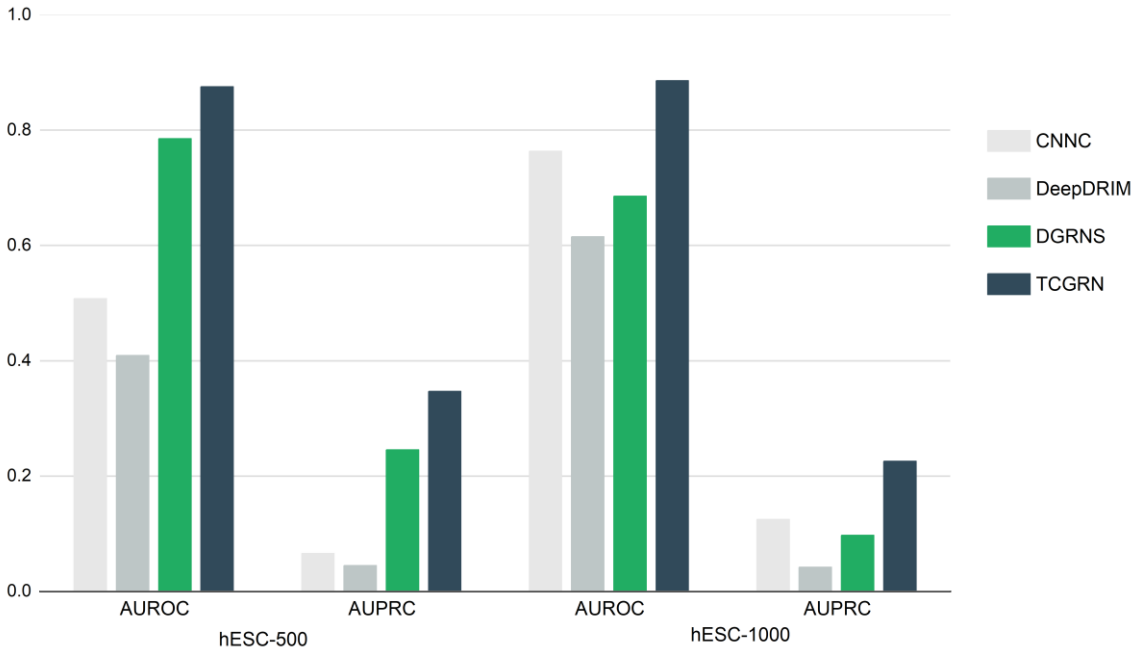

Fig. 7. Experiments on hESC datasets of varying sizes.

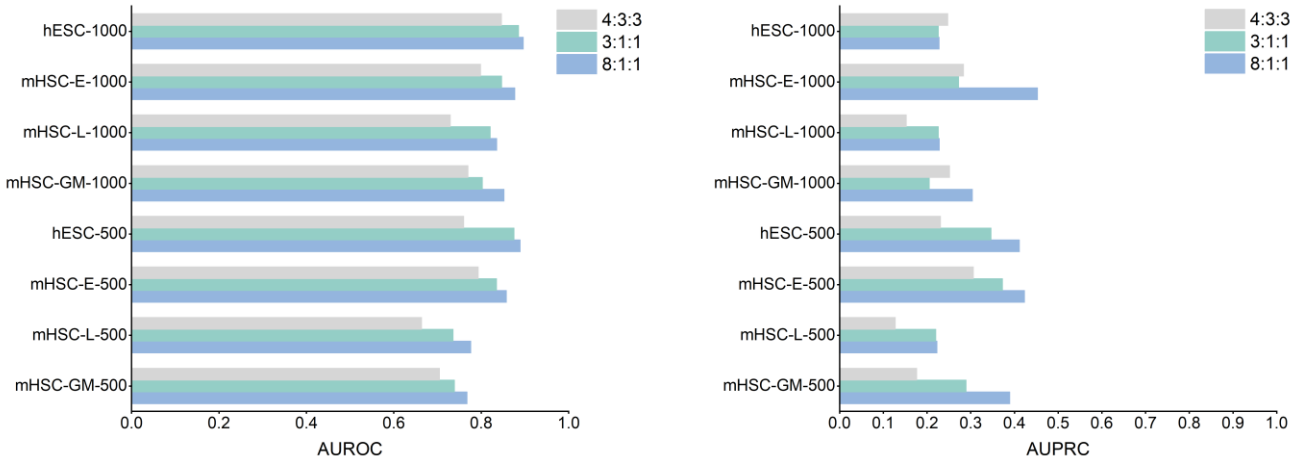

Fig. 8. Experiments on TCGRN with various data split ratios.

E. Effectiveness of image-format data input

To validate the impact of image-formatted data on spatial feature learning and overall model performance, we conducted ablation experiments to analyze the role of histogram-image representations in feature extraction. Specifically, while keeping all other operations unchanged, we replace the CNN input in TCGRN with the corresponding vectors instead of image data, and compare this method with TCGRN across mouse hematopoietic stem cell datasets of varying lineage sizes.

The experimental results are shown in Fig. 6, where TCGRN-IM denotes the model variant in which the CNN input is changed from images to vectors. As shown in the figure, across all datasets, the AUROC and AUPRC values of TCGRN-IM are significantly lower than those of the TCGRN model. For instance, on the mHSC-GM-1000 dataset, TCGRN's AUROC is approximately 12% higher than that of TCGRN-IM; on the mHSC-GM-500 dataset, TCGRN's AUPRC is about 16% higher than that of TCGRN-IM. The experimental results demonstrate that transforming the raw data into histogram images and inputting them into the CNN allows for more effective utilization of spatial structural information, enabling the model to learn more complex and globally correlated features.

F. Experiments on human datasets

Many deep learning models may achieve exceptional performance on specific types of data but perform less effectively on others. With this in mind, we evaluate TCGRN on hESC datasets of varying sizes. As illustrated in Fig. 7, TCGRN achieves consistent superior performance on multiple hESC datasets of varying sizes, outperforming all benchmark models in both AUROC and AUPRC metrics. On the hESC-500 dataset, TCGRN achieves an AUROC approximately 9% higher and an AUPRC about 10% higher than the second-best model, DGRNS. These results demonstrate TCGRN's generalization capability for handling complex and diverse data.

G. Experiments with varying split ratios

To investigate the impact of training, validation, and test set ratios on model performance, we conduct experiments on mHSC datasets of varying sizes as well as on hESC datasets. In these experiments, each dataset was partitioned into training, validation, and test sets according to the ratios 8:1:1, 3:1:1, and 4:3:3. The experimental results, shown in Fig. 8, reveal that when the training set proportion is high (8:1:1), the model achieves the highest AUROC values. This indicates that a larger training set facilitates the learning of more robust and comprehensive features, thereby enhancing overall inference capability. Conversely, when the training set proportion is low (4:3:3), the model is unable to obtain sufficient training data, resulting in poor AUROC performance across all datasets. When the ratio is 3:1:1, the model achieves second-best AUROC results on all datasets. On the hESC-1000 dataset, the model achieves the highest AUPRC under the 4:3:3 split, possibly due to the class imbalance in the data. Similarly, on the mHSC-GM-1000 and mHSC-E-1000 datasets, the model's AUPRC under the 4:3:3 split is higher than that under the 3:1:1 split, likely for the same reason.

IV. Discussion

To address the sparsity issue in single-cell sequencing data, TCGRN integrates multiple forms of data information. Specifically, it constructs correlation vector matrices and feeds them into TCN to capture time features. Simultaneously, it transforms the joint expression of transcription factors and genes into histogram images, including neighborhood images of gene pairs, which are then input into CNN to extract spatial features. Additionally, TCGRN leverages the attention mechanism to effectively integrate the time and spatial features captured by TCN and CNN. We conduct a comprehensive analysis of TCGRN on multiple single-cell transcriptome datasets, evaluating its performance using the standard metrics AUROC and AUPRC. The experimental results demonstrate that TCGRN consistently excels in both AUROC and AUPRC across all datasets, significantly enhancing the accuracy of GRN inference. This provides a solid theoretical foundation and technical support for subsequent biological research and applications.

In the future, we plan to enhance and extend TCGRN by incorporating multimodal data sources, for example, integrating single-cell Assay for Transposase-Accessible Chromatin using sequencing (scATAC-seq) and scRNA-seq, to construct more comprehensive GRN. Moreover, exploring more interpretable and biologically meaningful approaches to understanding and applying the inferred GRN will be an important direction for future research.

References

[1] D. Marbach *et al.*, "Wisdom of crowds for robust gene network inference," *Nature methods,* vol. 9, no. 8, pp. 796-804, 2012, doi: 10.1038/nmeth.2016.

[2] A. Pataskar and V. K. Tiwari, "Computational challenges in modeling gene regulatory events," *Transcription,* vol. 7, no. 5, pp. 188-195, 2016, doi: 10.1080/21541264.2016.1204491.

[3] C. A. Jackson, D. M. Castro, G.-A. Saldi, R. Bonneau, and D. Gresham, "Gene regulatory network reconstruction using single-cell RNA sequencing of barcoded genotypes in diverse environments," *elife,* vol. 9, p. e51254, 2020, doi: 10.7554/eLife.51254.

[4] Z. Razaghi-Moghadam and Z. Nikoloski, "Supervised learning of gene-regulatory networks based on graph distance profiles of transcriptomics data," *NPJ systems biology and applications,* vol. 6, no. 1, p. 21, 2020, doi: 10.1038/s41540-020-0140-1.

[5] R. Qi, A. Ma, Q. Ma, and Q. Zou, "Clustering and classification methods for single-cell RNA-sequencing data," *Briefings in bioinformatics,* vol. 21, no. 4, pp. 1196-1208, 2020, doi: 10.1093/bib/bbz062.

[6] Z. Wang, H. Ding, and Q. Zou, "Identifying cell types to interpret scRNA-seq data: how, why and more possibilities," *Briefings in functional genomics,* vol. 19, no. 4, pp. 286-291, 2020, doi: 10.1093/bfgp/elaa003.

[7] Z. Zhang *et al.*, "webSCST: an interactive web application for single-cell RNA-sequencing data and spatial transcriptomic data integration," *Bioinformatics,* vol. 38, no. 13, pp. 3488-3489, 2022, doi: 10.1093/bioinformatics/btac350.

[8] P. Badia-i-Mompel *et al.*, "Gene regulatory network inference in the era of single-cell multi-omics," *Nature Reviews Genetics,* vol. 24, no. 11, pp. 739-754, 2023, doi: 10.1038/s41576-023-00618-5.

[9] D. M. Alawad, A. Katebi, M. W. U. Kabir, and M. T. Hoque, "AGRN: accurate gene regulatory network inference using ensemble machine learning methods," *Bioinformatics Advances,* vol. 3, no. 1, p. vbad032, 2023, doi: 10.1093/bioadv/vbad032.

[10] L. Li, L. Sun, G. Chen, C.-W. Wong, W.-K. Ching, and Z.-P. Liu, "LogBTF: gene regulatory network inference using Boolean threshold network model from single-cell gene expression data," *Bioinformatics,* vol. 39, no. 5, p. btad256, 2023, doi: 10.1093/bioinformatics/btad256.

[11] V. A. Huynh-Thu, A. Irrthum, L. Wehenkel, and P. Geurts, "Inferring regulatory networks from expression data using tree-based methods,"

PloS one, vol. 5, no. 9, p. e12776, 2010, doi: 10.1371/journal.pone.0012776.

[12] T. E. Chan, M. P. Stumpf, and A. C. Babtie, "Gene regulatory network inference from single-cell data using multivariate information measures," *Cell systems,* vol. 5, no. 3, pp. 251-267. e3, 2017, doi: 10.1016/j.cels.2017.08.014.

[13] H. Matsumoto *et al.*, "SCODE: an efficient regulatory network inference algorithm from single-cell RNA-Seq during differentiation," *Bioinformatics,* vol. 33, no. 15, pp. 2314-2321, 2017, doi: 10.1093/bioinformatics/btx194.

[14] N. Papili Gao, S. M. Ud-Dean, O. Gandrillon, and R. Gunawan, "SINCERITIES: inferring gene regulatory networks from time-stamped single cell transcriptional expression profiles," *Bioinformatics,* vol. 34, no. 2, pp. 258-266, 2018, doi: 10.1093/bioinformatics/btx575.

[15] K. Kamimoto, C. M. Hoffmann, and S. A. Morris, "CellOracle: Dissecting cell identity via network inference and in silico gene perturbation," *BioRxiv,* p. 2020.02. 17.947416, 2020, doi: 10.1101/2020.02.17.947416.

[16] A.-C. Haury, F. Mordelet, P. Vera-Licona, and J.-P. Vert, "TIGRESS: trustful inference of gene regulation using stability selection," *BMC systems biology,* vol. 6, pp. 1-17, 2012, doi: 10.1186/1752-0509-6-145.

[17] M. Sanchez-Castillo, D. Blanco, I. M. Tienda-Luna, M. Carrion, and Y. Huang, "A Bayesian framework for the inference of gene regulatory networks from time and pseudo-time series data," *Bioinformatics,* vol. 34, no. 6, pp. 964-970, 2018, doi: 10.1093/bioinformatics/btx605.

[18] J. J. Faith *et al.*, "Large-scale mapping and validation of Escherichia coli transcriptional regulation from a compendium of expression profiles," *PLoS biology,* vol. 5, no. 1, p. e8, 2007, doi: 10.1371/journal.pbio.0050008.

[19] C. F. Aliferis, A. Statnikov, I. Tsamardinos, S. Mani, and X. D. Koutsoukos, "Local causal and Markov blanket induction for causal discovery and feature selection for classification part I: algorithms and empirical evaluation," *Journal of Machine Learning Research,* vol. 11, no. 1, 2010.

[20] A. J. Butte and I. S. Kohane, "Mutual information relevance networks: functional genomic clustering using pairwise entropy measurements," *Biocomputing,* pp. 418-429, 2000, doi: 10.1142/9789814447331_0040.

[21] X. Zhang *et al.*, "NARROMI: a noise and redundancy reduction technique improves accuracy of gene regulatory network inference," *Bioinformatics,* vol. 29, no. 1, pp. 106-113, 2013, doi: 10.1093/bioinformatics/bts619.

[22] T. Moerman *et al.*, "GRNBoost2 and Arboreto: efficient and scalable inference of gene regulatory networks," *Bioinformatics,* vol. 35, no. 12, pp. 2159-2161, 2019, doi: 10.1093/bioinformatics/bty916.

[23] Y. Yuan and Z. Bar-Joseph, "Deep learning for inferring gene relationships from single-cell expression data," *Proceedings of the National Academy of Sciences,* vol. 116, no. 52, pp. 27151-27158, 2019, doi: 10.1073/pnas.1911536116.

[24] J. Wang, A. Ma, Q. Ma, D. Xu, and T. Joshi, "Inductive inference of gene regulatory network using supervised and semi-supervised graph neural networks," *Computational and structural biotechnology journal,* vol. 18, pp. 3335-3343, 2020, doi: 10.1016/j.csbj.2020.10.022.

[25] C. Angermueller, T. Pärnamaa, L. Parts, and O. Stegle, "Deep learning for computational biology," *Molecular systems biology,* vol. 12, no. 7, p. 878, 2016, doi: 10.15252/msb.20156651.

[26] G. Eraslan, Ž. Avsec, J. Gagneur, and F. J. Theis, "Deep learning: new computational modelling techniques for genomics," *Nature Reviews Genetics,* vol. 20, no. 7, pp. 389-403, 2019, doi: 10.1038/s41576-019-0122-6.

[27] S. Jin, X. Zeng, F. Xia, W. Huang, and X. Liu, "Application of deep learning methods in biological networks," *Briefings in bioinformatics,* vol. 22, no. 2, pp. 1902-1917, 2021, doi: 10.1093/bib/bbaa043.

[28] J. Chen *et al.*, "DeepDRIM: a deep neural network to reconstruct cell-type-specific gene regulatory network using single-cell RNA-seq data," *Briefings in bioinformatics,* vol. 22, no. 6, p. bbab325, 2021, doi: 10.1093/bib/bbab325.

[29] M. Zhao, W. He, J. Tang, Q. Zou, and F. Guo, "A hybrid deep learning framework for gene regulatory network inference from single-cell transcriptomic data," *Briefings in bioinformatics,* vol. 23, no. 2, p. bbab568, 2022, doi: 10.1093/bib/bbab568.

[30] A. Pratapa, A. P. Jalihal, J. N. Law, A. Bharadwaj, and T. Murali, "Benchmarking algorithms for gene regulatory network inference from single-cell transcriptomic data," *Nature methods,* vol. 17, no. 2, pp. 147-154, 2020, doi: 10.1038/s41592-019-0690-6.

[31] S. Nestorowa *et al.*, "A single-cell resolution map of mouse hematopoietic stem and progenitor cell differentiation," *Blood, The Journal of the American Society of Hematology,* vol. 128, no. 8, pp. e20-e31, 2016, doi: 10.1182/blood-2016-05-716480.

[32] L.-F. Chu *et al.*, "Single-cell RNA-seq reveals novel regulators of human embryonic stem cell differentiation to definitive endoderm," *Genome biology,* vol. 17, pp. 1-20, 2016, doi: 10.1186/s13059-016-1033-x.

[33] H. Han *et al.*, "TRRUST: a reference database of human transcriptional regulatory interactions," *Scientific reports,* vol. 5, no. 1, p. 11432, 2015, doi: 10.1038/srep11432.

[34] Z.-P. Liu, C. Wu, H. Miao, and H. Wu, "RegNetwork: an integrated database of transcriptional and post-transcriptional regulatory networks in human and mouse," *Database,* vol. 2015, p. bav095, 2015, doi: 10.1093/database/bav095.

[35] L. Garcia-Alonso, C. H. Holland, M. M. Ibrahim, D. Turei, and J. Saez-Rodriguez, "Benchmark and integration of resources for the estimation of human transcription factor activities," *Genome research,* vol. 29, no. 8, pp. 1363-1375, 2019, doi: 10.1101/gr.240663.118.

[36] K. Street *et al.*, "Slingshot: cell lineage and pseudotime inference for single-cell transcriptomics," *BMC genomics,* vol. 19, pp. 1-16, 2018, doi: 10.1186/s12864-018-4772-0.