

RESEARCH

Open Access



AnomalGRN: deciphering single-cell gene regulation network with graph anomaly detection

Zhecheng Zhou¹, Jinhang Wei¹, Mingzhe Liu¹, Linlin Zhuo^{1*}, Xiangzheng Fu^{2*} and Quan Zou^{3*}

Abstract

Background Single-cell RNA sequencing (scRNA-seq) is now essential for cellular-level gene expression studies and deciphering complex gene regulatory mechanisms. Deep learning methods, when combined with scRNA-seq technology, transform gene regulation research into graph link prediction tasks. However, these methods struggle to mitigate the impact of noisy data in gene regulatory networks (GRNs) and address the significant imbalance between positive and negative links.

Results Consequently, we introduce the AnomalGRN model, focusing on heterogeneity and sparsification to elucidate complex regulatory mechanisms within GRNs. Initially, we consider gene pairs as nodes to construct new networks, thereby converting gene regulation prediction into a node prediction task. Considering the imbalance between positive and negative links in GRNs, we further adapt this issue into a graph anomaly detection (GAD) task, marking the first application of anomaly detection to GRN analysis. Introducing the cosine metric rule enables the AnomalGRN model to differentiate between homogeneity and heterogeneity among nodes in the reconstructed GRNs. The adoption of graph structure sparsification technology reduces noisy data impact and optimizes node representation.

Conclusions **Keywords** Gene regulation network (GRN), Link prediction, Graph anomaly detection, Heterogeneity and sparsification

Background

Gene regulation, crucial for controlling gene expression, dictates cellular and organismal responses to internal and external signals [1, 2]. Abnormal GRNs indicate dysregulation in gene–gene regulatory relationships, potentially triggering various pathogenic mechanisms [3]. Studying GRNs in depth can uncover disease-related key regulators and identify potential therapeutic targets [4]. This also offers researchers valuable insights for developing drug treatments. Focusing on specific regulatory pathways or gene targets enables the design of targeted drugs that minimize side effects and enhance therapeutic outcomes [5]. Consequently, a comprehensive understanding and accurate inference of GRNs are increasingly vital. Single-cell RNA sequencing, an advanced biotechnology,

*Correspondence:

Linlin Zhuo
20210339@wzjut.edu.cn
Xiangzheng Fu
fxz326@hnu.edu.cn
Quan Zou
zouquan@nclab.net

¹ School of Data Science and Artificial Intelligence, Wenzhou University of Technology, Wenzhou 325027, China

² College of Computer Science and Electronic Engineering, Hunan University, Changsha 410012, China

³ Institute of Fundamental and Frontier Sciences, University of Electronic Science and Technology of China, Chengdu 611730, China



© The Author(s) 2025. **Open Access** This article is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License, which permits any non-commercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if you modified the licensed material. You do not have permission under this licence to share adapted material derived from this article or parts of it. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by-nc-nd/4.0/>.

analyzes gene expression patterns at the single-cell level [6, 7]. Single-cell RNA sequencing precisely identifies and quantifies gene activities at the single-cell level, exposing functional and property differences among cells [8]. The rapid expansion of single-cell gene expression and regulatory data necessitates the development of efficient, accurate computational methods to explore the complex gene–gene interactions.

Machine learning and deep learning technologies have spurred the development of numerous computational models for GRN inference. Vân et al. introduced GENIE3, a model leveraging ensemble learning to break down GRN inference into several regression tasks, thereby inferring GRN through the integration of multiple regression models [9]. Shu et al. developed the GRN-Transformer model, employing a block cell approach to improve the predictive accuracy of single-cell GRNs [10]. They established a combined framework of pre-training and supervised learning, integrating single-cell RNA sequencing data with known GRN data from numerous cells to deduce GRNs for specific cell types.

The accuracy of GRN inference may be constrained by relying on a single data source. To address this limitation, Yuan et al. proposed the co-expression convolution neural network model (CNNC) to infer unknown GRNs [11]. CNNC converts expression data, which lacks local information, into an image-like format and applies a convolutional neural network (CNN) for feature extraction. Furthermore, CNNC is highly scalable, allowing the integration of diverse genomic data types to enhance its performance. Building on this, Chen et al. introduced DeepDRIM to integrate additional data sources [12]. DeepDRIM represents joint gene expression distributions of gene pairs as images, incorporating data on target genes, transcription factors, and potential neighbor correlations. Chen et al. employed multiple convolutional layers to integrate these images and compute gene–gene regulation scores.

GRNs depict the relationships between regulatory factors and target genes using graphs or networks. Kishan et al. introduced GNE, a scalable deep learning model based on network topology, known for its reliable predictive capabilities [13]. The GNE model integrates known gene–gene interaction and expression data, extracting node representations via message propagation and updates within the GRNs to predict unknown gene–gene interactions. Building on this, Chen et al. proposed GENELink, framing GRN inference as a link prediction task on a graph. GENELink employs a graph attention network (GAT) to infer potential interactions between transcription factors and target genes within GRNs [14]. GENELink first maps observed single-cell gene expression pairs to a low-dimensional space, then optimizes

this space to learn gene node representations for downstream predictions. Subsequently, Mao et al. developed the GNNLink model, incorporating an interactive graph encoder based on GNNs [15]. GNNLink then extracts node features and reconstructs GRNs through adjacency matrix completion.

Current computational models are capable of inferring GRNs, yet they encounter three significant challenges. Firstly, GRNs inherently contain structural noise, manifesting as “fake” links. Secondly, existing GNN-based methods lack an effective mechanism to differentiate between homogeneous and heterogeneous links, particularly during message propagation. Thirdly, the quantity of positive links is significantly lower than that of negative links. To address these challenges, we introduce the AnomalGRN model, designed from the perspectives of network heterogeneity and sparsity, to accurately infer GRNs. By creating new nodes from gene pairs, treating those with positive links as abnormal and those with negative links as normal, we transform the task of GRN inference into predicting abnormal nodes within the reconstructed GRN. Our contributions are summarized as follows:

1. This study innovates by transforming gene regulation prediction into node prediction tasks.
2. This study pioneers the application of graph node anomaly detection to GRN inference, targeting the significant imbalance between positive and negative links.
3. We employ a cosine similarity measure that accounts for message propagation among both homogenous and heterogeneous links.
4. We implement a sparsification strategy to eliminate noise and reduce the influence of “fake” links on model accuracy.

Results

Data preparation

This study utilized single-cell RNA sequencing (scRNA-seq) datasets for seven cell types from BEELINE [16]: (1) human embryonic stem cells (hESC), (2) human mature stem cells (hHEP), (3) mouse dendritic cells (mDC), (4) mouse embryonic stem cells (mESC), (5) erythroid mouse hematopoietic stem cells (mHSC-E), (6) granulocyte-monocyte lineage mouse hematopoietic stem cells (mHSC-GM), (7) mouse lymphoid hematopoietic stem cells (mHSC-L). These datasets are linked to three ground-truth networks (GTNs) derived from the STRING database, non-specific ChIP-seq, and cell-type-specific ChIP-seq. Additionally, loss-of-function/gain-of-function (Logf) GTNs are available for mESC.

The GTN from the STRING database primarily reflects extensive protein interactions. It is not specific to any cell type and focuses on general and global GRNs. Non-specific ChIP-seq data mainly comprise non-specific protein-DNA binding data resulting from technical reasons or experimental conditions. This type of GTN may contain more background noise and non-target signals, reflecting extensive protein-DNA relationships without cell type or state selectivity. Cell-type-specific ChIP-seq data are collected from specific cell types and provide accurate protein-DNA interaction information within a specific cell environment. This type of GTN is more accurate and relevant, revealing details of gene regulation in specific cell types, including transcription factor binding sites. Comparatively, cell-type-specific ChIP-seq data are of higher quality.

The volume of raw scRNA-seq data is substantial. To process this data, we utilized established preprocessing methods. Specifically, we removed genes expressed in fewer than 10% of cells and those with low expression levels from further analysis. We then calculated the variance and *p*-value for each gene’s expression, focusing particularly on genes with a Bonferroni-corrected *p*-value below 0.01. We normalized gene expression levels using logarithmic transformation. Following these steps, we constructed an $N \times M$ matrix, where N is the number of genes and M is the number of cells. Following Pratapa et al.’s method [17], we selected transcription factors (TFs) with high expression level and the top 500 and 1000 genes, thereby creating 14 distinct datasets, as shown in Table 1.

Experimental setup

To assess our model’s performance, we conducted comprehensive experiments on the aforementioned datasets. Furthermore, we benchmarked our model against several cutting-edge models, namely GNNLink [15], GENELink [14], GNE [13], CNNC [11], DeepDRIM [12], GRN-Transformer [10], and GENIE3 [9]. In our experimental setup, the division ratio for the training, validation, and test sets was uniformly set at 6:2:2, with the positive-to-negative sample ratio maintained at 1:1. The training was capped at 10,000 rounds, incorporating an early stopping mechanism whereby training ceases if no improvement is observed on the validation set for 500 consecutive rounds. The proposed model also involves the number of top- K neighbors (K), the hidden layer dimension, and the number of layers. The default values of these parameters are 10, 32, and 2, respectively. These parameters can be freely set. The primary evaluation metrics employed in the experiment were two well-known classification indicators: AUC and AUPR.

AUC represents the area under the ROC curve, plotted with the *true positive rate (Recall)* on the *y*-axis and the false positive rate on the *x*-axis. AUPR represents the area under the PR curve, plotted with *Precision* on the *y*-axis and *Recall* on the *x*-axis. AUC evaluates the ability to distinguish between positive and negative samples and is a comprehensive performance metric. AUPR focuses on distinguishing positive samples, particularly when the sample sizes are unbalanced, better reflecting the ability to recognize positive samples. The calculations for *true positive rate (Recall)*, *false positive rate*, and *Precision* are as follows:

$$\text{Truepositiverate(Recall)} = \frac{\text{TP}}{\text{TP} + \text{FN}} \text{Falsepositiverate} = \frac{\text{FP}}{\text{TN} + \text{FP}} \text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{1}$$

Table 1 Detailed statistics of the datasets

		STRING				Non-Specific				Specific				Lofgof			
		500		1000		500		1000		500		1000		500		1000	
	Cells	Node	Edge	Node	Edge	Node	Edge	Node	Edge	Node	Edge	Node	Edge	Node	Edge	Node	Edge
hESC	758	910	4257	1410	5149	910	3441	1410	4617	910	4545	1410	7084				
hHEP	425	948	7523	1448	9003	948	4129	1448	5351	948	9939	1448	15,558				
mDC	383	821	4815	1321	5898	821	3067	1321	3918	821	756	1321	1193				
mESC	421	1120	7762	1620	8479	1120	6893	1620	8030	1120	29,613	1620	42,795	1120	4169	1620	5742
mHSC-E	1071	704	1371	1204	1826	704	1425	1204	1960	704	11,557	1204	21,975				
mHSC-GM	889	632	748	1132	1311	632	743	1132	1358	632	7364	1132	14,135				
mHSC-L	847	560	137	692	154	560	279	692	317	560	4398	692	5180				

Where TP (TN) represents the number of correctly predicted positive (negative) samples, and FP (FN) represents the number of incorrectly predicted positive (negative) samples.

Performance on benchmark datasets

We conducted experiments across all benchmark datasets to compare model performances, with results presented in Fig. 1. TF+500 (1000) indicates that the top 500 (1000) genes based on gene expression levels are retained after screening scRNA-seq data. The screened genes are integrated with the GTN in the database to construct the datasets. The proposed GNN-based models, GNNLink and GENELink, significantly outperform other predictive models, demonstrating their superior capability in elucidating topological structures compared to conventional deep learning technologies. It is noteworthy that the proposed model surpasses the performance of all existing models. Specifically, on all TF+500 datasets, the proposed model's AUC and AUPR scores are respectively 14% and 57% higher than those of the

suboptimal GNNLink model. For all TF+1000 datasets, the AUC and AUPR values of the proposed model exceed those of the suboptimal GNNLink model by 10% and 58%, respectively.

AUC – denotes the difference between the model's average AUC on the TF+1000 datasets and its average AUC on the TF+500 datasets, while AUPR- denotes the difference between the model's average AUPR on the TF+1000 datasets and its average AUPR on the TF+500 datasets.

We infer that the high performance of all models on the Specific datasets may be due to its higher quality. All models perform poorly on the Non-specific datasets, likely due to more unpredictable noise. Additionally, STRING provides general data with unclear external factors, possibly resulting in lower data quality and, consequently, lower model performance. Notably, the proposed model demonstrates reliable performance across all datasets. Overall, the proposed model employs graph anomaly detection technology, demonstrating the potential for identifying noisy data.

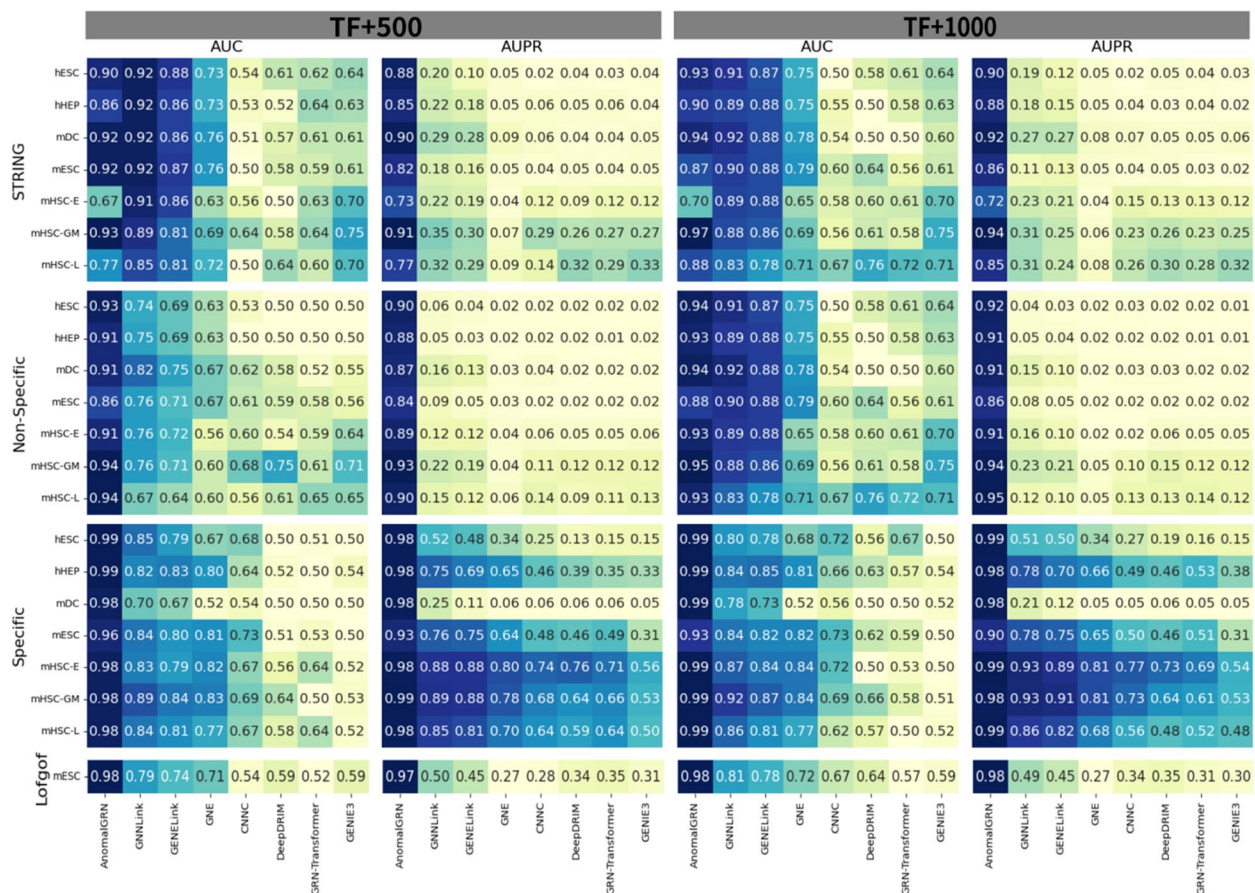


Fig. 1 Performance comparison of all models on all benchmark datasets. The result data for these comparison methods are cited from GNNLink [15] without re-running the models

Specifically, we attribute this to several factors. First, the proposed model calculates node similarity using cosine, revealing correlation, irrelevance, and heterogeneity between nodes. Second, the model sets a fixed threshold to filter out potential noise links. Third, the model uses top-k variants to identify the most reliable links. Fourth, during message propagation, the model uses cosine instead of a fixed “1” in the numerator of Eq. 12 to better distinguish heterogeneity from homogeneity.

We calculated the mean and standard deviation of the results for each model on all TF + 1000 datasets, the TF + 500 datasets, and the difference between the results on the two datasets, detailed in Table 2. The AUPR performance of all models on the TF + 1000 and TF + 500 datasets shows slight differences. However, the AUC performance on the TF + 1000 dataset is slightly better than on the TF + 500 dataset, suggesting that increased training data correlates with enhanced model performance.

Recently, Yuan et al. proposed the scMAGTGRN model [18], which utilizes multi-view technology to capture local features and high-order neighbor information, while employing an attention mechanism to adjust feature weights. Wang et al. introduced the Transformer-based scGREAT model [19] for GRN inference and

demonstrated its effectiveness in identifying unknown gene–gene relationships. Table 3 presents the AUC and AUPR metrics for AnomalGRN alongside the comparison models, scMGATGRN and scGREAT. The proposed AnomalGRN model outperforms scGREAT and scMAGTGRN significantly. These results further validate the effectiveness of the proposed AnomalGRN model in GRN inference.

Performance evaluation across positive–negative ratios

In GRNs, known links are significantly outnumbered by unknown links. The substantial imbalance between positive and negative samples presents a significant challenge for accurate GRN inference by the model. We conducted experiments to assess each model’s predictive performance across various positive-to-negative ratios on the mESC-500 dataset in the LOF/GOF network. Initially, we established positive-to-negative ratios of 1:5, 1:6, and 1:7, and assessed the performance of the proposed model alongside GNNLink, GENELink, GNE, CNNC, DeepDRIM, GRN-Transformer, and GENIE3. For more pronounced imbalances, we adjusted the positive-to-negative ratios to 1:10, 1:20, 1:30, 1:40, and 1:50, and benchmarked against GNN models like GCN, GAT, and GraphSAGE.

Table 2 Results of all models on all TF + 1000 and all TF + 500 datasets, respectively

Models						
Dataset/metric	TF + 500		TF + 1000			
	AUC	AUPR	AUC	AUPR	AUC –	AUPR –
AnomalGRN	0.919 ± 0.075	0.903 ± 0.070	0.934 ± 0.063	0.921 ± 0.921	0.015	0.018
GNNLink	0.825 ± 0.074	0.365 ± 0.277	0.871 ± 0.040	0.360 ± 0.360	0.046	−0.005
GENELink	0.779 ± 0.071	0.329 ± 0.283	0.843 ± 0.045	0.325 ± 0.325	0.064	−0.004
GNE	0.696 ± 0.084	0.224 ± 0.278	0.738 ± 0.073	0.222 ± 0.222	0.042	−0.002
CNNC	0.593 ± 0.071	0.215 ± 0.229	0.608 ± 0.070	0.221 ± 0.221	0.015	0.006
DeepDRIM	0.567 ± 0.060	0.207 ± 0.224	0.594 ± 0.075	0.212 ± 0.212	0.027	0.005
GRN-Transformer	0.574 ± 0.057	0.210 ± 0.224	0.583 ± 0.060	0.207 ± 0.207	0.010	−0.002
GENIE3	0.589 ± 0.078	0.183 ± 0.175	0.612 ± 0.081	0.178 ± 0.178	0.023	−0.005

Table 3 AUC and AUPR scores of models on cell-type-specific networks (TF + 500)

AUC/datasets	hESC	hHEP	mDC	mESC	mHSC-E	mHSC-GM	mHSC-L
scMGATGRN	0.87	0.89	0.76	0.92	0.90	0.93	0.86
scGREAT	0.89	0.91	0.81	0.94	0.93	0.93	0.88
AnomalGRN	0.99	0.99	0.98	0.96	0.98	0.98	0.98
AUPR/datasets	hESC	hHEP	mDC	mESC	mHSC-E	mHSC-GM	mHSC-L
scMGATGRN	0.59	0.80	0.12	0.81	0.91	0.93	0.86
scGREAT	0.89	0.91	0.84	0.95	0.94	0.94	0.89
AnomalGRN	0.98	0.98	0.98	0.93	0.98	0.99	0.98

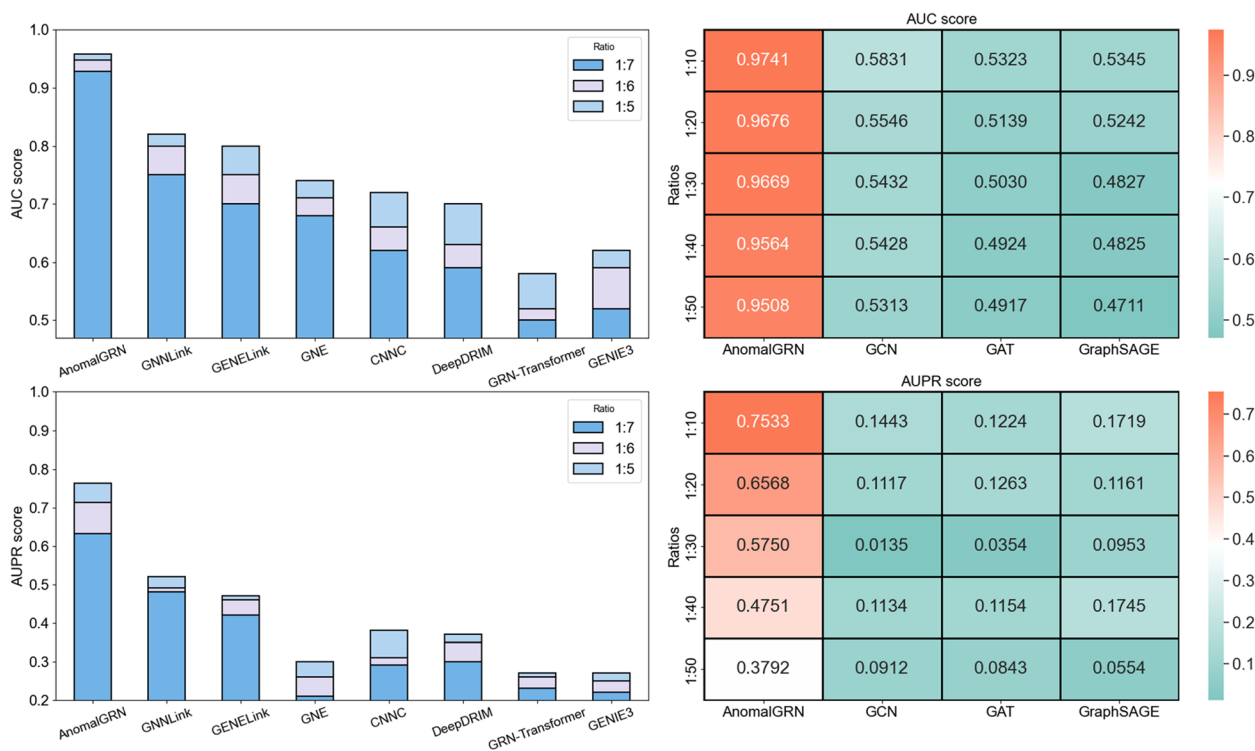


Fig. 2 Results of the models under multiple positive–negative ratios on the Lofgof_mESC_500 dataset. The performance of the proposed model was evaluated against the current cutting-edge model at positive-to-negative ratios of 1:5, 1:6, and 1:7. At positive-to-negative ratios of 1:10, 1:20, 1:30, 1:40, and 1:50, the proposed model’s performance was benchmarked against basic GNN models including GCN, GAT, and GraphSAGE. The result data for these comparison methods are cited from GNNLink [15] without re-running the models

Figure 2 illustrates that with widening positive-to-negative ratios, all models exhibit a trend of declining performance. This demonstrates that the imbalance between positive and negative samples is a critical factor limiting model performance. Nevertheless, the proposed model sustains high and stable performance at negative to positive ratios of 1:10, 1:20, 1:30, 1:40, and 1:50. This indicates that the model is capable of efficiently identifying potential positive links within highly imbalanced samples. Additionally, this validates the efficacy of using anomaly detection for GRN inference. This stability may stem from the model’s ability to filter

“fake” links and retain potential links via network sparsification, and its capacity to differentiate between homogeneous and heterogeneous links using cosine similarity during message propagation. In summary, the model is less susceptible to interference from “fake” links, effectively retains potential links, and thus accurately identifies positive links in imbalanced datasets.

Ablation experiment

The proposed AnomalGRN model primarily includes two core components: “Weight Reconstruction” and “Network Sparsification.” We conducted several experiments

Table 4 Results of ablation experiment

Datasets/metrics	Lofgof_mESC				Specific_mESC			
	500		1000		500		1000	
	AUC	AUPR	AUC	AUPR	AUC	AUPR	AUC	AUPR
w/o All	0.8987	0.8761	0.9135	0.8973	0.8814	0.8532	0.8837	0.8591
w/o WR	0.9308	0.9015	0.9326	0.9076	0.9083	0.8705	0.9005	0.8689
w/o GNS	0.9208	0.8941	0.9426	0.9118	0.9193	0.8868	0.9155	0.8852
w/o Cosmic	0.9313	0.9035	0.9335	0.9017	0.9025	0.8691	0.8939	0.8649
AnomalGRN	0.9608	0.9372	0.9826	0.9779	0.9575	0.9263	0.9279	0.8966

on the Lofgof_mESC and Specific_mESC datasets, using AUC and AUPR as evaluation metrics to verify the importance of these core components. In Table 4, “w/o WR” refers to the model without the “Weight Reconstruction” component, where the weights between two nodes are not computed using cosine similarity and take values of either 1 or 0 directly. In the reconstructed GRN, edges are defined according to the steps outlined in the “Task Transformation” section. “w/o GNS” refers to the model excluding the “Network Sparsification” component, where the adjacency matrix remains unchanged after the default number of rounds. “w/o Cosmic” denotes that in GCN message propagation, S_{ij} in Eq. 12 is set to 1. The “w/o All” experiment involves removing the “Weight Reconstruction,” “Network Sparsification,” and “Cosmic” message propagation components from the proposed model, relying solely on traditional GCN to perform the GRN inference task. However, this experiment was conducted on the reconstructed GRN, indicating that it underwent a task transformation process, as detailed in the “Task transformation” section. The results indicate that removing the “Weight Reconstruction,” “Network Sparsification,” and “Cosmic” message propagation components leads to the poorest model performance. Furthermore, the removal of any single component or mechanism results in a significant decline in model performance. These findings demonstrate that the components and mechanisms positively contribute to the model’s performance, validating their rationality and effectiveness. The aforementioned process occurs during the execution of the GCN.

Additionally, when all components or mechanisms are removed, the primary difference between the proposed model and other models lies in the “task transformation” process, as observed in the “w/o All” condition. During the “task transformation” process, the original GRN was reconstructed as the initial input graph. Specifically, gene–gene pairs are formed into new nodes, and the initial input graph is generated based on the “Weight Reconstruction” and “Network Sparsification” components. However, during the construction of the initial graph, the newly created nodes have no neighbor. This indicates that, as described in Eq. 7, neighbor screening is performed solely based on a threshold. This process generates only the initial graph and does not directly influence the GCN execution process. Therefore, we hypothesize that the “task transformation” process is a critical factor in enhancing model performance. Reconstructing the GRN (as an initial input graph) using the “Weight Reconstruction” and “Network Sparsification” components likely eliminates the impact of “fake” links and identifies potential neighbors, contributing to improved model performance.

Parameter experiment

The proposed model includes several adjustable hyperparameters: the training data ratio, the number of neighbors (K) in the KNN variant, the number of GNN layers, and the output dimension size. After edge weights are constructed, the KNN variant primarily identifies potential neighbors, which are critical for GNN message propagation within the reconstructed GRN. The number of layers and output features in the GNN model significantly influence feature extraction efficiency, consequently impacting future prediction tasks. The cell-type-specific ChIP-seq data are of comparatively higher quality. To minimize external interference, we conducted a series of experiments using the cell-type-specific ChIP-seq ground truth dataset to analyze how parameter variations influence performance.

Specifically, the K value was set to 5, 10, 15, and 20; the GNN output dimension was configured as 8, 16, 32, and 64; and the number of GNN layers was set to 1, 2, 3, and 4. The results are presented in Fig. 3B, C, and D. Meanwhile, other parameters were kept constant to adhere to the univariate principle. Experimental results indicate that the proposed model’s performance remains stable with minor fluctuations as the K value varies. Similarly, the model’s performance is largely unaffected by variations in the number of GNN layers or output feature size. These findings suggest that the model exhibits stable performance and strong adaptability to parameter variations. The proposed model can be rapidly deployed and applied to new GRN data with minimal parameter adjustments, thereby enhancing research efficiency.

Given the sparsity of known GRNs, the available training data is inherently limited. A series of experiments were conducted to evaluate the model’s predictive performance using 10% to 50% of the available training data. For comparison, the state-of-the-art GNNLink model was evaluated on the TF+500 dataset, as illustrated in Fig. 3A. The experimental results demonstrate that the performance of both models improves progressively as the training data increases. The experimental results demonstrate that the performance of both models improves progressively as the training data increases. Additionally, the proposed model exhibits stable performance with minimal fluctuations.

Noise-induced attack

To investigate the model’s resilience against noise attacks, we conducted experiments on the reconstructed GRN utilizing Random attack [20] and DICE attack [21] strategies. In a Random attack, artificial noise, including “fake” links, is introduced into the reconstructed GRN. Random

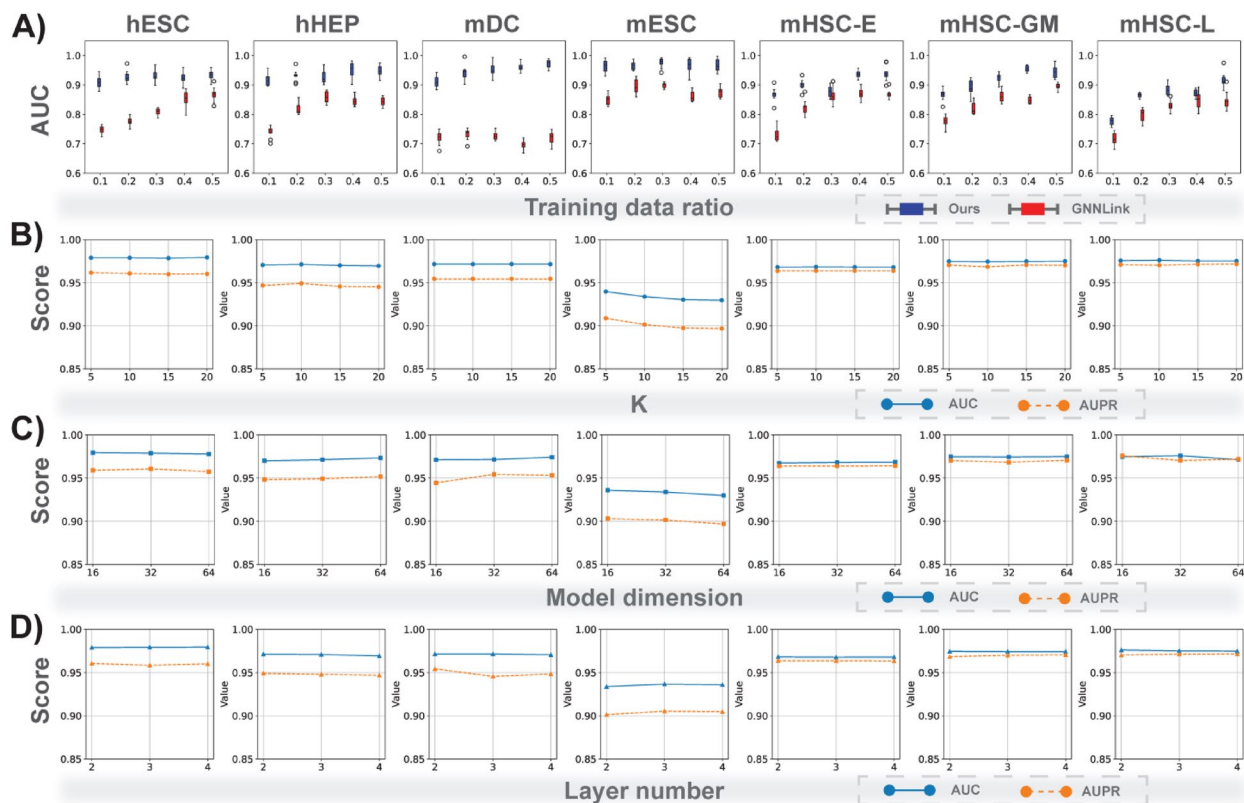


Fig. 3 Results of parameter experiments. **A** The performance of the proposed model is compared with that of the cutting-edge GNNLink model across various training data proportions. And the performance of the proposed model with varying K values (**B**), GNN output dimensions (**C**), and layer number (**D**), respectively

edges ranging from 0 to 100% are incorporated into the test set. During a DICE attack, existing links in the reconstructed GRN are eliminated, followed by the addition of an equivalent number of “fake” links. The attack intensity is similarly adjusted from 0 to 100%. These experiments primarily utilized mESC datasets. The comparative analysis involved models such as GCN [22], GAT [23], and GraphSAGE [24], with AUC and AUPR serving as the evaluation metrics.

Figures 4 A and B showcase the performance impact of Random and DICE attacks on each model, respectively. Figure 4 A reveals a notable decrease in performance for standard GNN models like GCN, GAT, and GraphSAGE as the proportion of “fake” links increases. This decline may stem from the injected noise disrupting the aggregation process within GNNs. In contrast, the proposed model remains largely unaffected. This resilience could be attributed to the proposed model’s effectiveness in filtering “fake” links.

Figure 4 B illustrates a marked decline in the performance of GCN, GAT, and GraphSAGE models as the attack intensity increases. This phenomenon is intuitively understandable. Removing real links and introducing

“fake” links effectively doubles the impact of a Random Attack. Nevertheless, the proposed model maintains stable performance with minimal fluctuation. This indicates that the proposed model is capable of effectively filtering out “fake” links while also revealing potential real links. Consequently, the proposed model can adeptly handle Random and DICE attacks, showing a higher tolerance for the inevitable noises encountered during the data collection process.

Case study

This study involved the selection of six genes across seven cell types for analysis. The TCF4 gene, essential for stem cell pluripotency and self-renewal, was selected from human embryonic stem cells. Dysfunctional or abnormally expressed TCF4 gene can lead to diseases such as schizophrenia, skin conditions, and certain cancers [25]. Additionally, the ARID2 gene was chosen from human mature stem cells. The ARID2 gene, along with other SWI/SNF complex members, significantly influences mature stem cell renewal and differentiation [26]. Selected from mouse embryonic stem cells, the EHMT2 gene is part of the histone methyltransferase

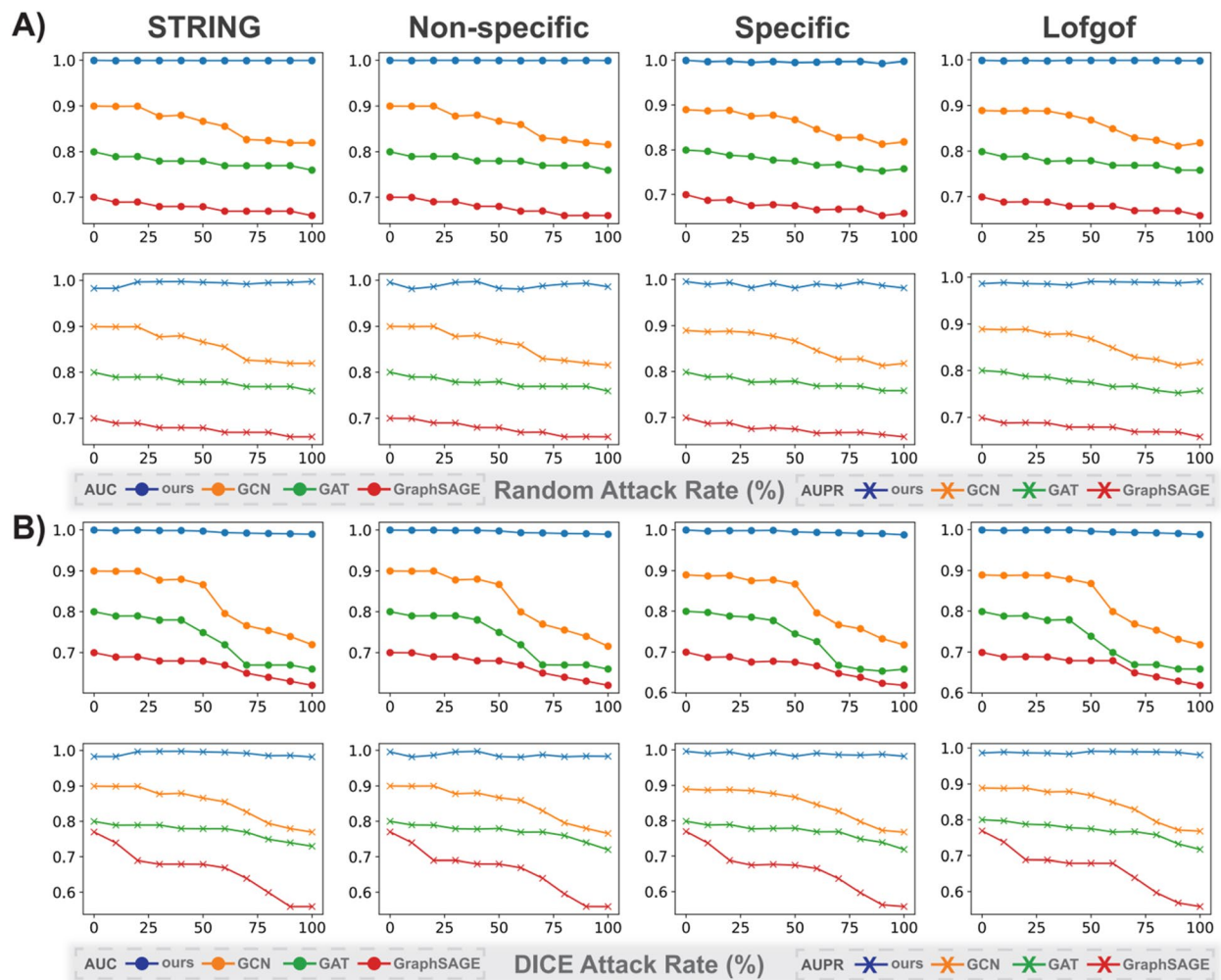


Fig. 4 Comparison of model performance under noise attacks. **A** Random attack. **B** DICE attack

family. These enzymes modify chromatin structure and gene programming by adding methylation marks to specific lysine residues on histone proteins [27]. From mouse dendritic cells, the MAFF gene was selected; its protein modulates specific gene expressions [28]. The RPA2 gene, selected from granulocyte-monocyte lineage mouse hematopoietic stem cells, potentially ensures cell maturation and function by maintaining DNA replication and repair accuracy [29]. The MEF2C gene, selected from mouse hematopoietic stem cells of both erythroid and lymphoid lineages. The MEF2C gene regulates the generation, maturation, and proliferation of erythroid lineage mouse hematopoietic stem cells, stabilizing and optimizing hematopoietic system function [30]. In lymphoid lineage mouse hematopoietic stem cells, the MEF2C gene influences immune system development and function [31].

In the training phase, we remove all associations involving the selected genes from the datasets. During the inference phase, our attention is on the selected genes, using the trained model to rebuild the GRN around them. Using the TCF4 gene in human embryonic stem cells as an example, the model accurately predicted both true positive and true negative rates at 100%. Notably, as demonstrated in Fig. 5A, all genes the model associates with TCF4 have been confirmed in the hESC dataset. Furthermore, Figs. 5B~G demonstrate that the model performs similarly in predicting ARID2, EHMT2, MEF2C, RPA2, and MAFF genes. Thus, we can conclude that the AnomalGRN model is indeed capable of inferring unknown GRNs. It is anticipated to aid in uncovering complex gene regulatory mechanisms and disease causation in the future.

Additionally, we used the LIANA toolkit [32] to load tissue sample data from patients with myocardial

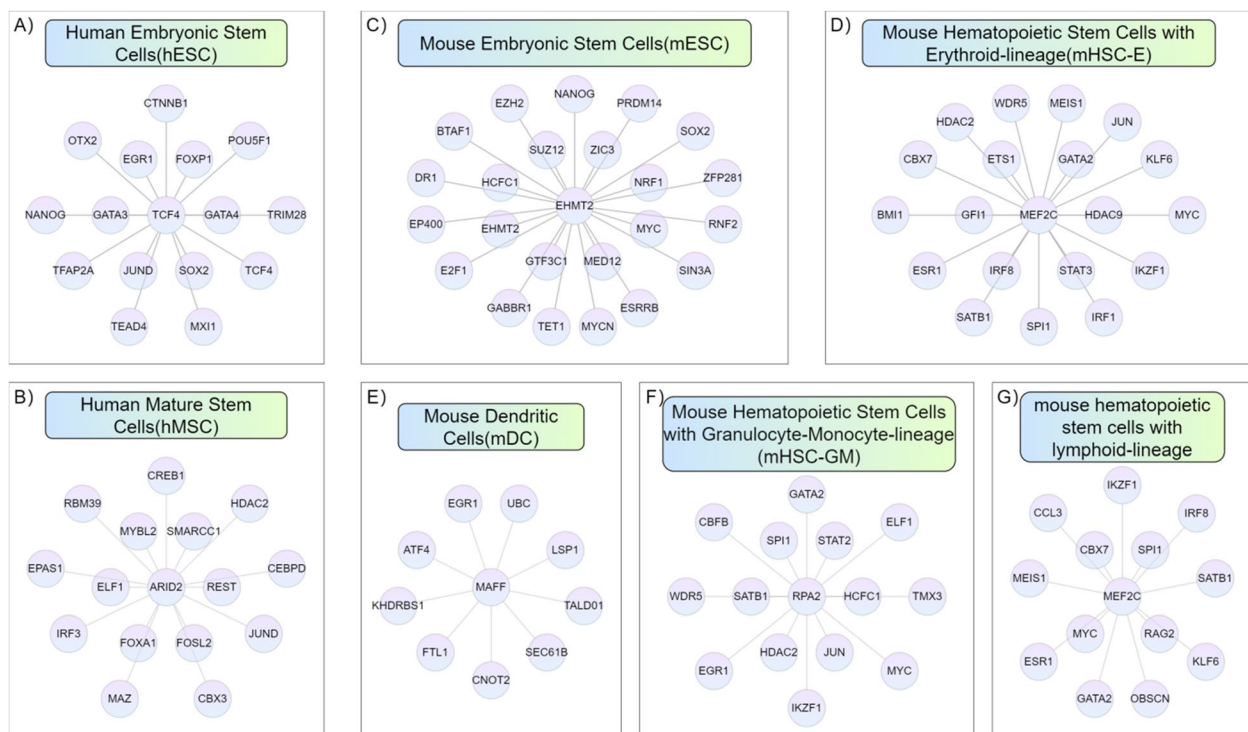


Fig. 5 The trained AnomalGRN model predicts GRNs centered on genes **A** TCF4 in human embryonic stem cells, **B** ARID2 in human mature stem cells, **C** EMT2 in mouse embryonic stem cells, **D** MEF2C in lineage mouse hematopoietic stem cells, **E** MAFF in mouse dendritic cells, **F** RPA2 in granulocyte-monocyte lineage mouse hematopoietic stem cells, and **G** MEF2C in lymphoid lineage mouse hematopoietic stem cells, respectively

infarction [33]. This sample is concentrated in the ischemic area of cardiac tissue and provides only gene expression data, without a corresponding GRN. Then, we obtained known GTNs from previous work [34], which collected many such GTNs. We integrated gene expression data and known GTNs, using gene expression data as node features and gene–gene pairs in known GTNs as links to reconstruct the GRN of the ischemic area in myocardial infarction patients. The integrated dataset was then divided into training, validation, and test sets in a 6:2:2 ratio. After training, the proposed model achieved an AUC of 72.69% and an AUPR of 75.05%. This indicates that the proposed model can potentially predict unknown gene–gene regulatory relationships based on feature data such as gene expression and known GTNs.

Additionally, we discussed applying this method to predict GRN in mouse kidney slices. Initially, we downloaded gene expression data of mouse kidney slices, generated using Visium technology, comprising 1438 cells and 32,285 genes. Based on previous work [35, 36], we acquired the known transcription factor (TF) and gene–target association information, referred to as the GRN. We integrated the gene expression data of mouse kidney slices with the known GRN, using genes as nodes and

their expression data as node features. Links were constructed based on the association information of known transcription factors (TFs) and target genes, while irrelevant genes in the GRN were discarded. This process allowed us to reconstruct the GRN of mouse kidney slices. The integrated dataset was then split into training, validation, and test sets in a 6:2:2 ratio. Following training, the proposed model achieved an AUC of 75.13% and an AUPR of 73.37%. These results also demonstrate the proposed model's scalability across diverse biological contexts and larger datasets.

Discussion

The integration of single-cell RNA sequencing (scRNA-seq) with deep learning has revolutionized GRN inference, enabling the exploration of gene regulation at an unprecedented resolution. This study proposes an innovative approach that frames GRN inference as a graph anomaly detection problem, effectively addressing several inherent challenges in the field. One of the significant challenges is the inevitable noise in scRNA-seq data, which can result in incorrect predictions of gene interactions. By sparsifying the GRN and filtering out false links, the impact of noise is reduced, thereby enhancing the

reliability of predictions. Another notable contribution of this study is its approach to addressing the imbalance between positive and negative links in GRNs. Typically, the number of positive links in a network is significantly lower than that of negative links, posing a challenge to model training. By classifying positive links as anomaly nodes and negative links as normal nodes, this study redefines GRN inference as a graph anomaly detection problem. This approach offers a novel solution to the imbalance problem and enhances the accuracy of GRN predictions.

A series of experimental results validate the rationality of the model and its key components. However, several important considerations remain. The reconstruction of new nodes and edges in GRNs necessitates further exploration. Moreover, the quality of data utilized for GRN inference is critical to the accuracy of the results. Integrating multimodal data, including gene epigenetic data, transcription factor binding sites, and protein–protein interactions, can substantially improve the model's robustness. Future work should focus on integrating advanced multimodal fusion techniques with large language models to enhance the accuracy and comprehensiveness of GRN inference. By incorporating diverse biological data sources, the proposed model can uncover intricate and subtle gene regulatory mechanisms, with potential applications in elucidating cellular processes, disease mechanisms, and therapeutic development.

Conclusions

Single-cell RNA sequencing (scRNA-seq) technology has significantly advanced our in-depth understanding and exploration of gene regulatory mechanisms at the single-cell level. Recent studies integrating scRNA-seq with deep learning technology have succeeded in predicting GRNs. However, these studies face two principal limitations. Firstly, noise is inevitable during the data collection process. Secondly, it is recognized that within GRNs, positive links are significantly outnumbered by negative links. To overcome the first limitation, we sparsify the GRN and filter out a portion of the “fake” links to mitigate the impact of noisy data. To tackle the second limitation, we categorize positive links as abnormal nodes and negative links as normal nodes. Subsequently, we reconstruct the GRN and convert the task of GRN inference into one of anomaly detection. Simultaneously, we employ cosine similarity to assess the similarity between nodes, differentiate between homogeneous and heterogeneous links, and mitigate the effects of the imbalance between positive and negative samples. Adopting this novel perspective, we conducted an in-depth study on the inference problem of GRN, substantially enhancing prediction accuracy. This marks the first application of anomaly

detection to GRN inference, achieving considerable success. This further validates the feasibility of employing anomaly detection in unbalanced datasets like single-cell GRN, thereby encouraging broader applications.

The goal of GRN inference is to identify potential gene–gene relationships from numerous gene–gene pairs, thereby uncovering gene regulatory mechanisms. In graph anomaly detection, abnormal nodes are typically much fewer than normal nodes. This study reframes GRN inference as a graph anomaly detection task, which appears to be both reasonable and innovative. Nevertheless, several critical aspects warrant careful consideration. First, developing effective methods for reconstructing new nodes and edges in the reconstructed GRN. Second, integrating higher-quality data. In future work, we aim to employ advanced multimodal fusion and large language model technologies to integrate multimodal data, such as gene epigenetic data, transcription factor binding site information, protein–protein interactions, and known GRNs. This is expected to address the identified challenges, enhance the proposed model, and more effectively uncover underlying gene regulatory mechanisms.

Methods

Model architecture

Previous research typically treats GRN inference as a graph link prediction task. In a novel approach, we reconceptualize GRN inference as a task of classifying nodes within the reconstructed GRNs. Specifically, interacting gene pairs are defined as abnormal nodes, while non-interacting pairs are considered normal nodes. Given the significantly lower number of positive links compared to negative links in GRNs, GRN inference can effectively be transformed into an anomaly detection task. We then introduce the AnomalGRN model, which is developed from perspectives of heterogeneity and sparsity to uncover complex regulatory mechanisms within GRNs.

Figure 6 illustrates the AnomalGRN model, which comprises multiple modules. Module A constructs the initial input graph (reconstructed GRN) by converting gene–gene pairs into new nodes and splicing the scRNA-seq data of gene–gene pairs to create feature vectors for these nodes. The initial input graph is then constructed through the following two steps. First, Module B uses a deep neural network (DNN) to linearly map the node feature vectors and calculates the cosine similarity between the new nodes, as detailed in the “[Weight reconstruction](#)” section. Second, based on the cosine similarity between nodes, some neighbors are identified according to a pre-defined threshold. Additionally, an improved top-*K* algorithm is employed to determine additional neighbors, as explained in the “[Network sparsification](#)” section. It

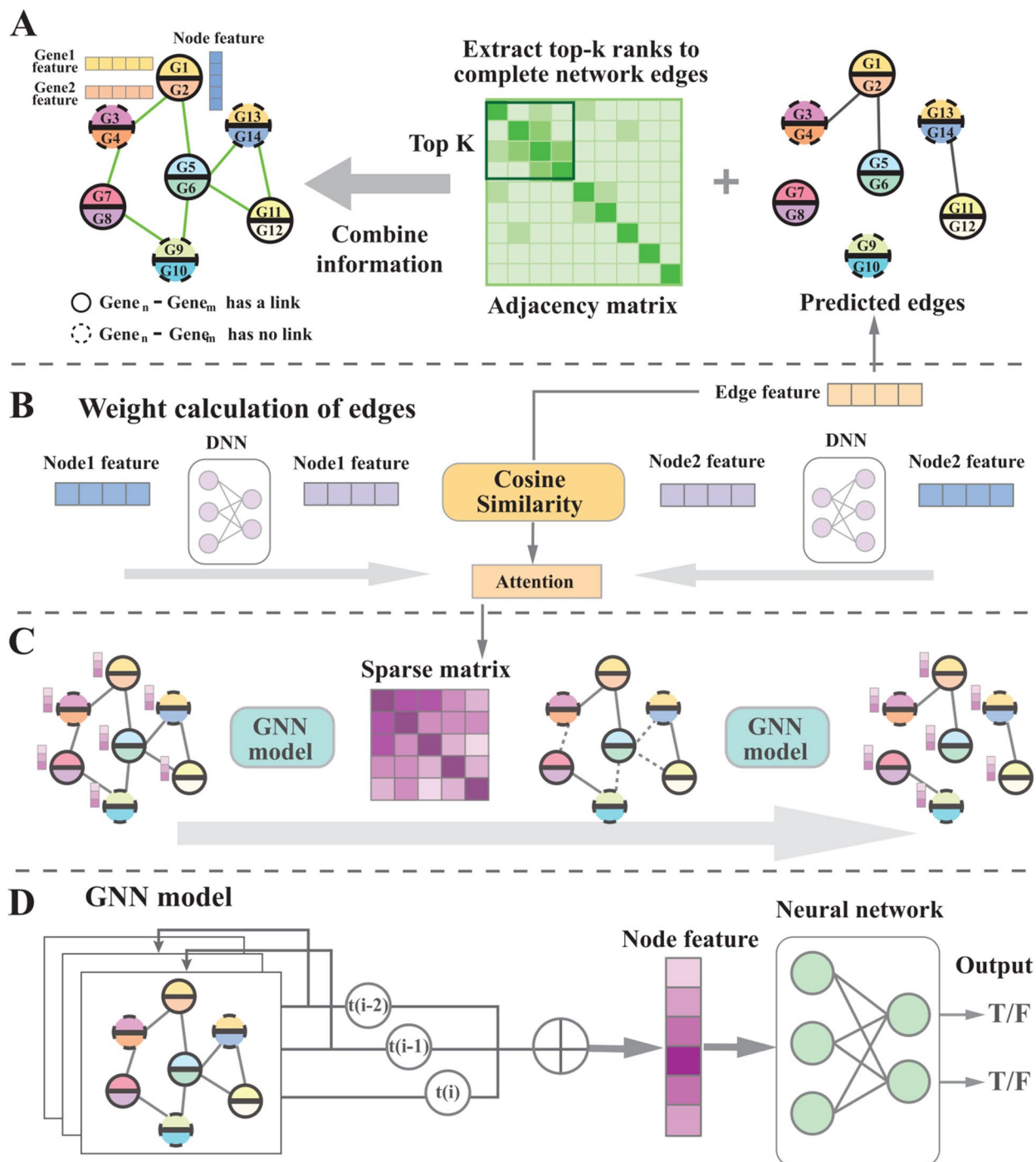


Fig. 6 The proposed AnomalGRN architecture includes four main modules: **A** task transformation to construct a reconstructed GRN, **B** weight reconstruction, **C** an overview of the entire process, and **D** integration of outputs from multi-layer GNNs to extract node features and make predictions

is noteworthy that a sparse matrix is also constructed between GNNs following these steps. Module C details the model's entire execution process. The reconstructed GRN generated by Module A serves as input to the GNN

model, then Eqs. 7 and 9 are applied to produce a sparse matrix. The outputs of multi-GNN layers, after multiple iterations, are integrated to derive node features, which are then used for label prediction through a neural

network. Module D provides an abstract representation of Module C, where $t(i)$ denotes the output of the GCN encoder at the i th layer, $t(i-1)$ is the input to the i th layer, and “ \oplus ” signifies the concatenation operation. Subsequently, we provide a detailed explanation of these key modules and technologies.

Task transformation

GRNs typically feature far fewer positive links than negative ones. This scenario aligns with anomaly detection tasks. Generally, normal nodes significantly outnumber abnormal nodes. Initially, we innovatively aggregate node pairs into singular nodes. Gene–gene pairs with links are designated as abnormal nodes, while those without links are considered normal nodes. Subsequently, we reconstruct the GRN using the B module, transforming the GRN inference problem into a graph anomaly detection task. Naturally, the reconstructed GRN often exhibits “fake” links, where normal nodes connect to abnormal ones. Links between two new nodes of the same type are defined as homogeneous links, whereas links between nodes of different types are defined as heterogeneous links.

Clearly, to accurately and effectively complete the anomaly detection task, two prerequisites are essential. First, the GNN model must simultaneously consider message propagation modes for both homogeneous and heterogeneous links. Second, efforts should be made to minimize the model’s susceptibility to “fake” links. To address the first condition, we plan to employ cosine similarity for calculating edge weights, distinguishing between heterogeneous and homogeneous link message propagation. Regarding the second condition, we propose utilizing graph sparsification technology to filter out “unimportant” links, thereby mitigating the effect of “fake” links on model performance.

The original GRN is represented as $G=(V, E, X)$, where V denotes the set of genes, E the set of links between genes, and X the scRNA sequence and other gene attribute data. The reconstructed GRN is denoted as $G_r=(V_r, E_r, X_r, S)$, where V_r is the set of nodes reconstructed from all gene–gene pairs, X_r the spliced feature vectors of these pairs, and E_r the set of reconstructed links. Edge weights $S \in [-1, 1]$, calculated using cosine similarity, represent the weights of the reconstructed links. From a statistical viewpoint, the GRN inference (graph anomaly detection) problem can be formalized as:

$$P(Y|G_r) = P(Y|G_r) \approx \sum_{g \in GS} P(Y|g)P(g|G_r) \quad (2)$$

where Y denotes labels, where normal nodes (unconnected gene–gene pairs) are assigned 0, and abnormal nodes (connected gene–gene pairs) are assigned 1. G_S denotes a series of sparse graph sets derived from graph G_r . Traversing G_S is challenging and time-intensive; thus,

based on previous work [37, 38], Eq. 2 can be approximated as follows:

$$\sum_{g \in GS} P(Y|g)P(g|G_r) \approx \sum_{g \in GS} K_\phi(Y|g, S)K_\chi(g|G_r) \quad (3)$$

where K_ϕ and K_χ are the estimators for $P(Y|g)$ and $P(g|G_r)$, parameterized by ϕ and χ , respectively. K_χ takes G_r as input, generates a series of subgraphs (G_S) centered around each node as described in Eqs. 7 and 9, and ultimately forms a sparse adjacency matrix A . The K_ϕ function employs the GCN model to output node labels based on the input adjacency matrix A , which comprises all subgraphs (G_S). To ensure the differentiability of Eqs. 2 and 3, we generate a differentiable sample Δg using the parameterization technique [39]:

$$\sum_{g \in GS} K_\phi(Y|g)K_\chi(g|G_r) \propto \sum_{\Delta g \sim K_\chi(g|G_r)} K_\phi(Y|\Delta g) \quad (4)$$

where $\Delta g \sim K_\chi(g|G_r)$ represents $\Delta g = (V, \Delta A, X)$, where Δg is a sparse graph sampled from $K_\chi(g|G_r)$.

Weight reconstruction

In the reconstructed GRNs, many abnormal links are heterogeneous. Thus, adequately addressing heterogeneous links can enhance anomaly detection task performance. Similar to previous work [38], we employ cosine similarity to reconstruct the adjacency weight matrix S , enabling consideration of both homogeneous and heterogeneous links in GNN message propagation. In the reconstructed GRN, for nodes i and j with initial features z_i and z_j , respectively, we input these features into a two-layer multilayer perceptron (MLP) and calculate the cosine similarity between the node outputs as follows:

$$S_{ij} = \frac{z_i^0 \cdot z_j^0}{|z_i^0| \cdot |z_j^0|} \in [-1, 1] \quad (5)$$

where $z_i^0 = \text{MLP}(z_i)$ denotes node i ’s representation as output by the MLP, with z_i originating from X_2 . S_{ij} quantifies the cosine similarity between nodes i and j . The relationship between nodes i and j is categorized based on three criteria: homogeneity, heterogeneity, and neutrality:

$$\begin{cases} S_{ij} \rightarrow 1, & \text{if } i \text{ to } j \text{ tend to be homogeneous.} \\ S_{ij} \rightarrow 0, & \text{if } i \text{ to } j \text{ tend to be worthless.} \\ S_{ij} \rightarrow -1, & \text{if } i \text{ to } j \text{ tend to be heterogeneous.} \end{cases} \quad (6)$$

The determination of weight S is independent of the edges in the reconstructed GRN. Unlike the graph attention mechanism, this approach uses cosine similarity to calculate node weights, ranging from -1 to 1 , diverging from the non-negative weights used in graph attention. Consequently, GNN models can account for both

homogeneous and heterogeneous links in the information dissemination process. Furthermore, as per Eq. 7, the calculation is symmetrical.

In the proposed model, this component is primarily used for initial graph construction, the “Network Sparsification” component, and the message propagation mechanism. First, this component calculates the similarity between nodes and uses the top- K algorithm to find the K nearest neighbors of each node, constructing the initial graph. During training, the adjacency matrix is updated every fixed number of rounds (default is 250 rounds). Second, in the “Network Sparsification” component, between the layers of the GNN encoder, this component calculates node similarity, selects neighbors by setting a threshold, and uses the top- K algorithm to find the K nearest neighbors, finally merging the final neighbors. Third, in the message propagation mechanism, as shown in Eq. 12, the GCN message propagation rule changes the numerator from “1” to the cosine similarity between nodes. This weakens and distinguishes some irrelevant links.

Network sparsification

Pervasive “fake” links or neighbors in the reconstructed GRNs significantly impair model performance. Furthermore, the incompleteness of data arises from limitations in collection technology and conditions. Consequently, we apply sparsification to the reconstructed GRN. Our goals are twofold: firstly, to diminish the impact of “fake” links, and secondly, to discern potentially significant node neighbors.

As outlined in Sect. 2.2, the absolute value of S_{ij} indicates the influence of node i on node j . Therefore, setting a threshold α enables the filtering of non-essential neighbor nodes:

$$N1(i) = \{j \in V | j \in N(i), |S_{ij}| > \alpha\} \quad (7)$$

where $N(i)$ denotes the set of neighbors associated with node i . Consequently, certain nodes with high similarity but not classified as neighbors at this stage are ignored. Therefore, we identify potential neighbors utilizing the weight matrix S and an enhanced KNN algorithm. We employ the Gumbel-top- K method, an advanced variant of the KNN algorithm, to calculate the probability value of node pairs and select the K most promising neighbors [39, 40]. Adhering to the principles of underlying data distribution [39], the probability value for any node j not in $N(i)$ is calculated as follows:

$$P_{ij} = \frac{|S_{ij}|}{\sum_{c \notin N(i)} |S_{ic}|} \quad (8)$$

Subsequently, logarithmic transformation of the probability value serves as the criterion for selecting the top- K neighbors:

$$\begin{aligned} \delta_{ij} &= \log(p_{ij}) - \log(-\log(\varepsilon)) \\ N2(i) &= \{j \in V | j \notin N(i), \delta_{ij} \in \text{top} - K(\delta * i)\} \end{aligned} \quad (9)$$

The formal parameter ε is sampled independently from a uniform distribution (0,1) for each node. The function **top- K** () identifies the set of the top K largest δ values. Our study iteratively applied this operation across the entire reconstructed GRNs.

Neighbor sets N_1 and N_2 , derived from Eqs. 7 and 9, are merged to create the final neighbor set $N' = N_1 \cup N_2$. For node i , if j is in $N'(i)$, the corresponding element A_{ij}' in the sparse adjacency matrix A' is assigned a value of 1. Based on this, the edge set E_2 is derived. At this point, $G' = (V_p, E_2, X_p, S)$ signifies the completion of the network's sparsification. This is an iterative and reusable process.

Numerically, S_{ij} spans a broad range of values. This implies a connection between any two nodes, necessitating processing of the reconstructed GRN to preserve sparsity. Methods like minimizing the 1 norm [41] or the nuclear norm [42] are used to optimize the weight matrix S , though they may have limitations [20, 43]. Drawing inspiration from Gumbel-Softmax, we achieve sparse regularization by constraining the absolute value of S to approach 1 or 0:

$$Lr = - \sum_{A_{ij}=1} (\omega_{ij} \log \omega_{ij} + (1 - \omega_{ij}) \log (1 - \omega_{ij})) / \tau \quad (10)$$

where $\omega_{ij} = \max(\alpha, \min(|S_{ij}|, 1 - \alpha))$ and it requires a smaller α , rendering a larger value meaningless. Furthermore, in $\max(\alpha, \min(|S_{ij}|, 1 - \alpha))$, if S_{ij} is significantly small and below α , it is ignored and replaced by α . This also mirrors the treatment of fake links below α in Eq. 7. The parameter $\tau > 0$ regulates the anticipated sparsity level of S , where smaller τ results in more S values being optimized to 0, thus yielding a sparser graph Δg .

Message propagation

Following the aforementioned procedures, we obtain the sparse graph $G' = (V_p, E_2, X_p, S)$. Subsequently, we derive node representations using the GNN model and perform node classification to fulfill the anomaly detection task, thereby aiming to infer the GRN. Any GNN model may be selected to facilitate this process. In our study, we utilized the GCN [22] model as an exemplar to demonstrate information propagation in the reconstructed GRN. The node representation of node i at the l th layer can be denoted as follows:

$$z_i^h = \sigma \left(\left(z_i^{h-1} + \sum_{j \in N'(i)} \gamma_{ij}^h z_i^{h-1} \right) W^h \right) \quad (11)$$

where j is a member of $N'(i)$, denoting the adjacency of node i as derived from matrix A' . The trainable weight matrix is represented by W^h , and the activation function by σ . The coefficient γ_{ij}^h dictates the contribution value of neighboring nodes to i .

The value of γ_{ij}^h warrants particular attention. As detailed in Sects. 2.2 and 2.3, reconstructed GRNs inherently contain heterogeneous links. Homogeneous and heterogeneous nodes frequently exhibit distinct attributes. Concurrently, S_{ij} reveals the relationship between nodes i and j , categorizing it as homogeneous, heterogeneous, or unrelated. In GCN's message propagation, both homogeneity and heterogeneity are incorporated:

$$\gamma_{ij}^h = \frac{S_{ij}}{\sqrt{(d_i + 1)(d_j + 1)}} \quad (12)$$

where d_i and d_j denote the degrees of nodes i and j within the sparse network G' , respectively. Diverging from standard GCN models, S_{ij} replaces “1,” and both positive and negative values are introduced to account for the influence of homogeneous and heterogeneous neighbors concurrently.

Model optimization

The model's optimization objectives primarily encompass the refinement of the weight matrix S and minimizing classification loss. For any given node i , upon obtaining its final representation \hat{z}_i , the data undergoes transformation and prediction via a linear layer:

$$\hat{y}_i = \sigma(\hat{z}_i W + b) \quad (13)$$

where W denotes a trainable weight matrix, while b signifies the bias. \hat{y}_i indicates the predicted category of the node, being either 1 for a normal node or 0 for an abnormal node. Subsequently, the binary cross-entropy function is employed to compute the classification loss:

$$L_c = \sum_{i \in V_1} y_i \log \hat{y}_i + (1 - y_i) \log (1 - \hat{y}_i) \quad (14)$$

where y_i denotes the actual label of node i . Thus, by setting the weight coefficient λ and incorporating both optimization and classification loss, including that of the weight matrix S , the ultimate optimization objective is established as follows:

$$L = L_c + \lambda L_r \quad (15)$$

Acknowledgements

We would like to thank the National Natural Science Foundation of China for their support of this work (No. 62450002, 62302339, 62372158). We also appreciate the contributions of all authors involved in this study, whose collaboration made this research possible.

Authors' contribution

All authors read and approved the final manuscript. Zhecheng Zhou and Jinhang Wei contributed to the experimental design and initial draft preparation. Mingzhe Liu and Linlin Zhuo provided experimental guidance, while Linlin Zhuo additionally conducted data analysis. Xiangzheng Fu and Quan Zou oversaw manuscript revision and review.

Funding

This work was supported by the National Natural Science Foundation of China (No. 62131004 to QZ, No. 62302339 to LZ, No. 62372158 to XF).

Data availability

All data generated or analysed during this study are included in this published article, its supplementary information files and publicly available repositories. Our code is publicly available in the GitHub repository: <https://github.com/ZZCrazy00/AnomalGRN>, and is also publicly available in the Zenodo repository: <https://zenodo.org/records/14917279>. Additionally, the data is publicly available on Zenodo: <https://zenodo.org/records/12176604>.

Declarations

Ethics approval and consent to participate

This study involves computational experiments that are non-invasive and do not directly intervene with any human or animal subjects. Therefore, ethical approval from an institutional review board is not required. The consent of participants and the protection of personal information are not applicable, as the experimental data are sourced from publicly available datasets.

Consent for publication

All authors have provided their consent for publication of this study. There are no identifiable individuals or personal data included in this manuscript, ensuring compliance with publication ethics.

Competing interests

The authors declare no competing interests.

Received: 3 November 2024 Accepted: 25 February 2025

Published online: 11 March 2025

References

- De Nadal E, Ammerer G, Posas F. Controlling gene expression in response to stress. *Nat Rev Genetics*. 2011;12(12):833–845–833–45.
- López-Maury L, Marguerat S, Bähler J. Tuning gene expression to changing environments: from rapid responses to evolutionary adaptation. *Nature Reviews Genetics*. 2008;9(8):583–593–583–93.
- Parikshak NN, Gandal MJ, Geschwind DH. Systems biology and gene networks in neurodevelopmental and neurodegenerative disorders. *Nature Reviews Genetics*. 2015;16(8):441–458–441–58.
- Potkin SG, Maciardi F, Guffanti G, Fallon JH, Wang Q, Turner JA, Lakatos A, Miles MF, Lander A, Vawter MP, et al. Identifying gene regulatory networks in schizophrenia. *Neuroimage*. 2010;53(3):839–847–839–47.
- Moding EJ, Kastan MB, Kirsch DG. Strategies for optimizing the response of cancer and normal tissues to radiation. *Nature reviews Drug discovery*. 2013;12(7):526–542–526–42.
- Buettner F, Natarajan KN, Casale FP, Proserpio V, Scialdone A, Theis FJ, Teichmann SA, Marioni JC, Stegle O. Computational analysis of cell-to-cell heterogeneity in single-cell RNA-sequencing data reveals hidden subpopulations of cells. *Nature biotechnology*. 2015;33(2):155–160–155–60.
- Ding J, Adiconis X, Simmons SK, Kowalczyk MS, Hession CC, Marjanovic ND, Hughes TK, Wadsworth MH, Burks T, Nguyen LT, et al. Systematic

- comparison of single-cell and single-nucleus RNA-sequencing methods. *Nature biotechnology*. 2020;38(6):737–746–737–46.
8. Chiu IM, Barrett LB, Williams EK, Strohlic DE, Lee S, Weyer AD, Lou S, Bryman GS, Roberson DP, Ghasemlou N, et al. Transcriptional profiling at whole population and single cell levels reveals somatosensory neuron molecular diversity. *Elife*. 2014;3:e04660–e04660.
9. Huynh-Thu VA, Irrthum A, Wehenkel L, Geurts P. Inferring regulatory networks from expression data using tree-based methods. *PLoS One*. 2010;5(9):e12776–e12776.
10. Shu H, Ding F, Zhou J, Xue Y, Zhao D, Zeng J, Ma J. Boosting single-cell gene regulatory network reconstruction via bulk-cell transcriptomic data. *Briefings in Bioinformatics*. 2022;23(5):bbac389–bbac389.
11. Yuan Y, Bar-Joseph Z. Deep learning for inferring gene relationships from single-cell expression data. *Proceedings of the National Academy of Sciences*. 2019;116(52):27151–27158–27151–8.
12. Chen J, Cheong C, Lan L, Zhou X, Liu J, Lyu A, Cheung WK, Zhang L. DeepDRIM: a deep neural network to reconstruct cell-type-specific gene regulatory network using single-cell RNA-seq data. *Briefings in Bioinformatics*. 2021;22(6):bbab325–bbab325.
13. Kc K, Li R, Cui F, Yu Q, Haake AR. GNE: a deep learning framework for gene network inference by aggregating biological information. *BMC systems biology*. 2019;13(2):1–14–11–4.
14. Chen G, Liu ZP. Graph attention network for link prediction of gene regulations from single-cell RNA-sequencing data. *Bioinformatics*. 2022;38(19):4522–4529–4522–9.
15. Mao G, Pang Z, Zuo K, Wang Q, Pei X, Chen X, Liu J. Predicting gene regulatory links from single-cell RNA-seq data using graph neural networks. *Briefings in Bioinformatics*. 2023;24(6):bbad414–bbad414.
16. Pratapa A, Jaliha AP, Law JN, Bharadwaj A, Murali T. Benchmarking algorithms for gene regulatory network inference from single-cell transcriptomic data. *Nat Methods*. 2020;17(2):147–54.
17. Pratapa A, Jaliha AP, Law JN, Bharadwaj A, Murali TM. Benchmarking algorithms for gene regulatory network inference from single-cell transcriptomic data. *Nature methods*. 2020;17(2):147–154–147–54.
18. Yuan L, Zhao L, Jiang Y, Shen Z, Zhang Q, Zhang M, Zheng C-H, Huang D-S. scMGATGRN: a multiview graph attention network-based method for inferring gene regulatory networks from single-cell transcriptomic data. *Briefings in Bioinformatics*. 2024;25(6):bbae526.
19. Wang Y, Chen X, Zheng Z, Huang L, Xie W, Wang F, Zhang Z, Wong K-C. scGREAT: Transformer-based deep-language model for gene regulatory network inference from single-cell transcriptomics. *Science*. 2024;27:27(4).
20. Jin W, Ma Y, Liu X, Tang X, Wang S, Tang J. Graph structure learning for robust graph neural networks. In: *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*. 2020.
21. Zügner D, Borchert O, Akbarnejad A, Günnemann S. Adversarial attacks on graph neural networks: perturbations and their patterns. *ACM Transactions on Knowledge Discovery from Data (TKDD)*. 2020;14(5):1–31–31–31.
22. Kipf TN, Welling M. Semi-supervised classification with graph convolutional networks. 5th International Conference on Learning Representations, {ICLR} 2017, Toulon, France, April 24–26, 2017, Conference Track Proceedings. 2017. <https://openreview.net/forum?id=SJU4ayYgl>.
23. Velickovic P, Cucurull G, Casanova A, Romero A, Liò P, Bengio Y. Graph attention networks. 2018.
24. Hamilton WL, Ying Z, Leskovec J. Inductive representation learning on large graphs. 2017.
25. Forrest MP, Hill MJ, Quantock AJ, Martin-Rendon E, Blake DJ. The emerging roles of TCF4 in disease and development. *Trends in molecular medicine*. 2014;20(6):322–331–322–31.
26. Li M, Zhao H, Zhang X, Wood LD, Anders RA, Choti MA, Pawlik TM, Daniel HD, Kannangai R, Offerhaus GJA, et al. Inactivating mutations of the chromatin remodeling gene ARID2 in hepatocellular carcinoma. *Nature genetics*. 2011;43(9):828–829–828–9.
27. Auclair G, Borgel J, Sanz LA, Vallet J, Guibert S, Dumas M, Cavelier P, Girardot M, Forné T, Feil R, et al. EHM2 directs DNA methylation for efficient gene silencing in mouse embryos. *Genome research*. 2016;26(2):192–202–192–202.
28. Moon EJ, Mello SS, Li CG, Chi JT, Thakkar K, Kirkland JG, Lagory EL, Lee JJ, Diep AN, Miao Y, et al. The HIF target MAFF promotes tumor invasion and metastasis through IL11 and STAT3 signaling. *Nature communications*. 2021;12(1):4308–4308.
29. Lee DH, Pan Y, Kanner S, Sung P, Borowiec JA, Chowdhury D. A PP4 phosphatase complex dephosphorylates RPA2 to facilitate DNA repair via homologous recombination. *Nature structural and molecular biology*. 2010;17(3):365–372–365–72.
30. Arnold MA, Kim Y, Czubyrt MP, Phan D, McAnally J, Qi X, Shelton JM, Richardson JA, Bassel-Duby R, Olson EN. MEF2C transcription factor controls chondrocyte hypertrophy and bone development. *Developmental cell*. 2007;12(3):377–389–377–89.
31. Lin Q, Schwarz J, Bucana C, Olson EN. Control of mouse cardiac morphogenesis and myogenesis by transcription factor MEF2C. *Science*. 1997;276(5317):1404–1407–1404–7.
32. Dimitrov D, Schäfer PSL, Farr E, Rodriguez Mier P, Lobentanz S, Dugourd A, Tanevski J, Ramirez Flores RO, Saez-Rodriguez J. LIANA+: an all-in-one cell-cell communication framework. *BioRxiv*. 2023. 2023.2008.2019.553863. <https://doi.org/10.1101/2023.08.19.553863>.
33. Kuppe C, Ramirez Flores RO, Li Z, Hayat S, Levinson RT, Liao X, Hannani MT, Tanevski J, Wünnemann F, Nagai JS. Spatial multi-omic map of human myocardial infarction. *Nature*. 2022;608(7924):766–77.
34. Huang X, Song C, Zhang G, Li Y, Zhao Y, Zhang Q, Zhang Y, Fan S, Zhao J, Xie L. scGRN: a comprehensive single-cell gene regulatory network platform of human and mouse. *Nucleic Acids Res*. 2024;52(D1):D293–303.
35. Ding Q, Yang W, Xue G, Liu H, Cai Y, Que J, Jin X, Luo M, Pang F, Yang Y. Dimension reduction, cell clustering, and cell-cell communication inference for single-cell transcriptomics with DcComm. *Genome Biol*. 2024;25(1):241.
36. McCalla SG, Fotuhi Siahpirani A, Li J, Pyne S, Stone M, Periyasamy V, Shin J, Roy S. Identifying strengths and weaknesses of methods for computational network inference from single-cell RNA-seq data. *G3: Genes, Genomes, Genetics*. 2023;13(3):jkad004.
37. Zheng C, Zong B, Cheng W, Song D, Ni J, Yu W, Chen H, Wang W. Robust graph representation learning via neural sparsification. In: *International conference on machine learning*. 2020.
38. Gong Z, Wang G, Sun Y, Liu Q, Ning Y, Xiong H, Peng J. Beyond homophily: robust graph anomaly detection via neural sparsification. In: *Proceedings of the thirty-second international joint conference on artificial intelligence*. 2023.
39. Kool W, van Hoof H, Welling M. Estimating gradients for discrete random variables by sampling without replacement. 8th International Conference on Learning Representations, {ICLR} 2020, Addis Ababa, Ethiopia, April 26–30, 2020. <https://openreview.net/forum?id=rklEj2EFvB>.
40. Kazi A, Cosmo L, Ahmadi S-A, Navab N, Bronstein MM. Differentiable graph module (dgm) for graph convolutional networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2022;45(2):1606–1617–1606–17.
41. Chen Y, Wu L, Zaki M. Iterative deep graph learning for graph neural networks: better and robust node embeddings. *Advances in neural information processing systems*. 2020;33:19314–19326–19314–26.
42. Koltchinskii V, Lounici K, Tsybakov AB. Nuclear-norm penalization and optimal rates for noisy low-rank matrix completion. 2011.
43. Zhu Y, Xu W, Zhang J, Du Y, Zhang J, Liu Q, Yang C, Wu S. A survey on graph structure learning: progress and opportunities. *arXiv preprint arXiv:2103.03036*, 2021. <https://doi.org/10.48550/arXiv.2103.03036>.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.