
CoDiffGRN: Rethinking Gene Regulatory Network Inference via the BEELINE-KGC Benchmark and Co-evolutionary Discrete Diffusion

Jiaze Song
Peking University
jzsong25@stu.pku.edu.cn

Runhao Zhao
National University of Defense Technology
runhaozhao@nudt.edu.cn

Minghao Xu
Peking University
minghao.xu@stu.pku.edu.cn

Bin Cui
Peking University
bin.cui@pku.edu.cn

Wentao Zhang*
Peking University
wentao.zhang@pku.edu.cn

Abstract

Inferring gene regulatory networks (GRNs) from single-cell transcriptomic data is crucial for biological discovery, yet existing approaches suffer from a fundamental misalignment with real-world needs. Researchers typically seek a small set of high-confidence regulatory interactions for experimental validation, often involving previously unseen genes. However, current benchmarks rely on transductive splits with global classification metrics, while prevailing models struggle to generalize under inductive settings. To bridge this gap, we reformulate GRN inference as an inductive, ranking-centric graph completion problem and introduce **BEELINE-KGC**, a new benchmark that incorporates an inductive gene-holdout split together with knowledge graph completion metrics to better evaluate top-ranked predictions. Building on this, we propose **CoDiffGRN**, the first co-evolutionary discrete diffusion framework that jointly models biologically coherent discretized gene expression states and regulatory interactions for robust inductive generalization and improved top-ranked regulatory discovery. We further introduce TF-ALL Subgraph Sampling (TASS) for scalable training. Extensive experiments on BEELINE-KGC show that CoDiffGRN establishes new state-of-the-art performance, significantly outperforming existing methods in novel regulatory discovery, and ablation studies further verify the effectiveness of our design.

1 Introduction

Gene regulatory networks (GRNs) dictate cellular identity and physiological function by encoding the complex interactions between transcription factors (TFs) and target genes [1–4]. While the advent of single-cell RNA sequencing (scRNA-seq) has fundamentally advanced GRN inference by resolving bulk-level signal averaging [5], the resulting data is notoriously heterogeneous and sparse [6, 7]. To tackle these challenges, recent progress has shifted from classical statistical methods to graph neural networks (GNNs), particularly graph autoencoder (GAE) architectures [8–13]. Despite these architectural advances, we argue that computational GRN inference suffers from a fundamental misalignment with practical biological discovery in two critical dimensions:

First, there is an evaluation misalignment. Standard benchmarks such as BEELINE [14] evaluate GRN inference under transductive settings, where all genes are observed during training. However,

*Corresponding author

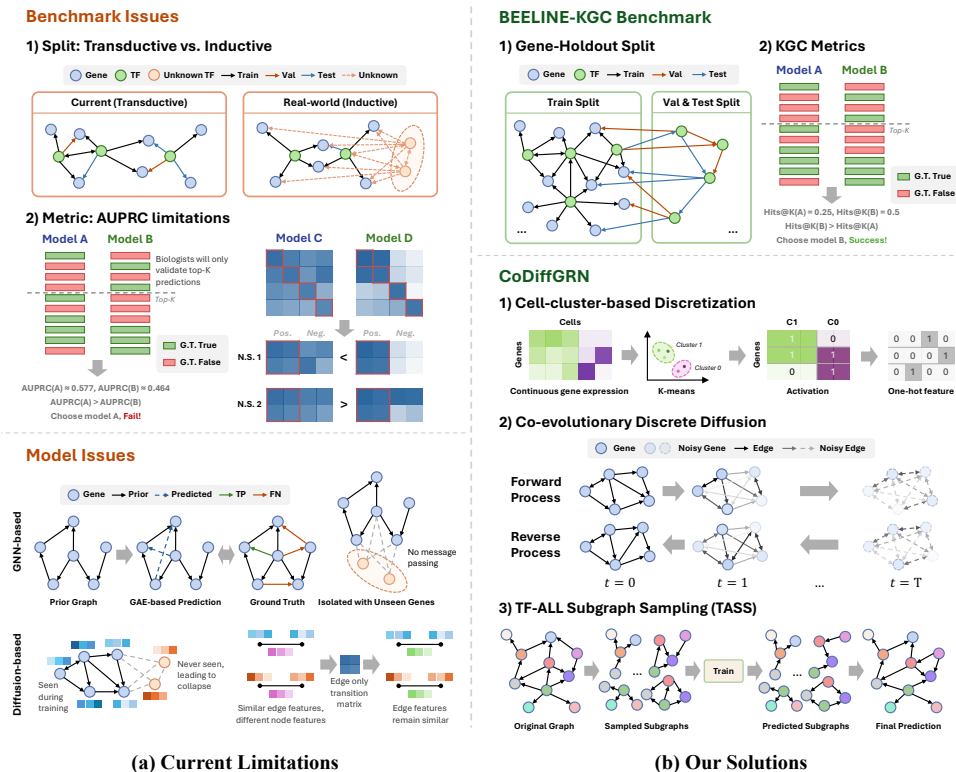


Figure 1: **Overview of current limitations and our proposed solutions.** (a) Current benchmarks mismatch with realistic demand of inductive generalization and top- K ranking quality, while existing methods degrade with unseen genes and fixed edge transitions. (b) Our BEELINE-KGC introduces an inductive, ranking-based evaluation protocol, and CoDiffGRN leverages cell-cluster discretization, co-evolutionary discrete diffusion with TASS. *Abbr.*, G.T.: Ground-truth, N.S.: Negative Sample.

Second, there exists a gene-level inductive generalization gap. Under realistic inductive settings, existing GAE-based methods rely heavily on predefined prior graphs. When holdout genes become disconnected during inference, they lose critical contextual signals and suffer severe performance degradation (Figure 1a). Recent diffusion-based methods partially alleviate this topological isolation, but still fail to generalize effectively to unseen genes, as the highly heterogeneous and sparse nature of scRNA-seq data prevents reliable feature correspondence and can lead to feature collapse.

To address these limitations, we introduce a new evaluation paradigm alongside a novel generative framework. First, we propose **BEELINE-KGC**, a systematic extension of BEELINE that enforces an *Inductive Gene-Holdout Split*. By recasting evaluation as a Knowledge Graph Completion (KGC) task [17], BEELINE-KGC uses Hits@K and Mean Reciprocal Rank (MRR) to directly measure the top- K ranking consistency of regulatory predictions. Under this rigorous protocol, existing state-of-the-art models exhibit severe generalization bottlenecks.

On the modeling side, we propose **CoDiffGRN**, the first gene-regulation co-evolutionary discrete diffusion model for GRN inference. Built upon the D3PM framework [18], CoDiffGRN departs from prior-dependent paradigms and is designed for robust inductive generalization. To mitigate feature collapse on unseen genes, we introduce a **Cell-cluster-based Discretization** strategy that transforms scRNA-seq profiles into biologically coherent discrete states, allowing unseen genes to inherit prior distributional structure during diffusion.

To capture the conditional nature of gene regulation, we introduce a joint node-edge diffusion process that directly models the co-evolution of discretized gene expression states and network topology at the level of diffusion dynamics. By devising a *gene-state-dependent transition matrix*, edge transitions are explicitly conditioned on the states of their endpoint genes, promoting the coherent neighborhoods required for top- K regulatory discovery. Furthermore, to enable scalable generative training under data scarcity, we introduce **TF-ALL Subgraph Sampling (TASS)**, a data augmentation strategy that preserves TF-target integrity while substantially improving scalability and generalization.

Comprehensive experimental results on BEELINE-KGC show that CoDiffGRN achieves state-of-the-art performance and consistently outperforms existing models. Extensive ablation studies examine the contributions of the proposed gene-regulation co-evolutionary discrete diffusion and TASS framework. These results collectively support the effectiveness of our approach in improving KGC-based ranking performance, enabling more accurate and reliable biological discovery.

2 Related Work

Evaluation protocols for GRN inference. BEELINE [14] established the standard for GRN inference, primarily utilizing transductive edge-centric splits and global metrics such as AUROC and AUPRC [19]. To improve discriminative sensitivity, negative sampling has evolved from random selection [20, 21] to heuristic strategies [10, 13]. However, these protocols exhibit a fundamental misalignment with practical biological discovery: transductive settings neglect the imperative for generalization to unseen genes, while global metrics fail to prioritize the top- K candidates required for experimental validation. In this work, we introduce **BEELINE-KGC** to bridge this gap by recasting evaluation as an inductive KGC task, utilizing ranking-based metrics (Hits@ K , MRR) to assess top- K regulatory predictions.

Gene regulatory network inference methods. Early GRN inference relied on statistical associations like GENIE3 [22] and GRNBoost2 [20], or feature-based deep learning models such as CNNE [21] and DeepSEM [23]. Recently, graph representation learning has become the dominant paradigm, with GAE-based frameworks like GENELink [8], GENELink+ [10] and GNNLink [9] utilizing prior regulatory graphs, while methods like GCLink [12] and GRNFormer [11] introduce contrastive learning and VGAE-based frameworks. More recently, generative models such as RegDiffusion [24], DigNet [25], and Planet [26] have applied continuous and discrete diffusion processes [18, 27] to model the evolution nature of gene regulations.

However, existing methods face critical limitations: traditional approaches struggle with noisy single-cell data, while graph-based models often hinge on prior-graph connectivity. Furthermore, current generative frameworks struggle with feature collapse in inductive scenarios and fail to capture the essential co-evolution between gene and regulation with severe data scarcity. In this work, we propose **CoDiffGRN**, which employs cell-cluster discretization and joint discrete diffusion to model regulatory dynamics, alongside TASS specifically designed to address the challenges of data scarcity.

3 Rethinking for Practical Discovery: The BEELINE-KGC Benchmark

We introduce BEELINE-KGC, a comprehensive benchmark designed to bridge the gap between computational GRN inference and realistic biological discovery. Built upon the BEELINE [14] (CC BY-NC 4.0) collection of experimental scRNA-seq datasets, BEELINE-KGC establishes a rigorous evaluation framework tailored to real-world laboratory constraints.

3.1 Formalization and Problem Setting

We formalize a gene regulatory network as a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where nodes $v \in \mathcal{V}$ represent genes and edges $(u, v) \in \mathcal{E}$ denote regulatory interactions. We partition $\mathcal{V}$ into TFs ($\mathcal{V}_{TF}$) and target

Table 1: Statistics of the BEELINE-KGC benchmark. For each cell type and reference network, we report the number of cells, the number of source TFs, and the edge counts under inductive splits. Values are shown as TFs+500 (TFs+1000). *Abbr.*, S. TFs: Source TFs.

Cell Types	#Cells	Specific					Non-Specific				
		#S. TFs	#Genes	#Train	#Valid	#Test	#S. TFs	#Genes	#Train	#Valid	#Test
hESC	758	34 (34)	815 (1260)	3535 (5995)	505 (544)	505 (545)	283 (292)	753 (1138)	2546 (3531)	447 (543)	448 (543)
hHEP	425	30 (31)	874 (1331)	7724 (12466)	1107 (1546)	1108 (1546)	322 (332)	825 (1217)	2993 (3490)	568 (930)	568 (931)
mDC	383	20 (21)	443 (684)	536 (903)	110 (145)	110 (145)	250 (254)	634 (969)	2115 (2704)	476 (607)	476 (607)
mESC	421	88 (89)	977 (1385)	22694 (33645)	3459 (4575)	3460 (4575)	516 (522)	890 (1214)	4677 (5738)	1108 (1146)	1108 (1146)
mHSC-E	1071	29 (33)	691 (1177)	9286 (17825)	1135 (2075)	1136 (2075)	144 (147)	442 (674)	1002 (1370)	211 (295)	212 (295)
mHSC-GM	889	22 (23)	618 (1089)	5796 (10901)	784 (1617)	784 (1617)	82 (88)	297 (526)	496 (925)	123 (216)	124 (217)
mHSC-L	847	16 (16)	525 (640)	3354 (3887)	522 (646)	522 (647)	35 (37)	164 (192)	175 (217)	52 (50)	52 (50)

Cell Types	#Cells	STRING					LOF/GOF				
		#S. TFs	#Genes	#Train	#Valid	#Test	#S. TFs	#Genes	#Train	#Valid	#Test
hESC	758	343 (351)	511 (695)	2862 (3433)	697 (858)	698 (858)	–	–	–	–	–
hHEP	425	409 (414)	646 (874)	4960 (6269)	1281 (1367)	1282 (1367)	–	–	–	–	–
mDC	383	264 (273)	479 (664)	3277 (4066)	769 (916)	769 (916)	–	–	–	–	–
mESC	421	495 (499)	638 (785)	5099 (5830)	1331 (1324)	1332 (1325)	34 (34)	775 (1099)	3322 (4851)	423 (445)	424 (446)
mHSC-E	1071	156 (161)	291 (413)	950 (1330)	210 (248)	211 (248)	–	–	–	–	–
mHSC-GM	889	92 (100)	201 (344)	505 (1108)	121 (101)	122 (102)	–	–	–	–	–
mHSC-L	847	39 (40)	70 (81)	89 (101)	24 (26)	24 (27)	–	–	–	–	–

genes ($\mathcal{V}_{tgt}$). Within the TFs, we further distinguish **source TFs** ($\mathcal{V}_{src} \subseteq \mathcal{V}_{TF}$), defined as regulators with at least one observed outgoing edge in the ground truth.

The corresponding scRNA-seq data is represented by an expression matrix $\mathbf{X} \in \mathbb{R}^{N \times F_X}$, where $\mathbf{X}_{i,j}$ denotes the expression level of gene $v_i \in \mathcal{V}$ in cell $j \in \{1, \dots, F_X\}$. Given the expression matrix $\mathbf{X}$ and a subset of observed interactions $\mathcal{E}_{obs}$, the primary objective of GRN inference is to recover the unobserved regulatory structure by predicting potential links in $\mathcal{E} \setminus \mathcal{E}_{obs}$.

3.2 A Unified Framework for Practical GRN Discovery

Standard GRN inference benchmarks adopt a transductive setting and rely on global metrics such as AUPRC. However, this paradigm fundamentally diverges from the practical demands of real biological discovery, which requires **inductive generalization** to novel regulators and **high top- K ranking quality** under limited experimental budgets.

Pillar 1: Inductive gene-holdout split. In realistic discovery scenarios, researchers predominantly focus on characterizing novel or unannotated TFs by identifying their downstream regulatory targets, necessitating an inductive setting where models must generalize to unseen regulators [15, 16, 28–33]. To simulate this, we extract a subset of $\mathcal{V}_{src}$ as holdout TFs ($\mathcal{V}_{hold}$), introducing an inductive protocol:

- (1) **Training:** The model observes only the subgraph induced by $\mathcal{V} \setminus \mathcal{V}_{hold}$. Expression profiles and all edges involving holdout TFs are withheld.
- (2) **Evaluation:** The model receives the expression profiles for the full gene set $\mathcal{V}$ and predicts regulations across the entire network. Performance is evaluated solely on edges involving at least one holdout TF.

Pillar 2: KGC reformulation and multi-scale ranking protocol. Practical biological discovery is governed by a **discrete experimental budget** K , where true interactions are useful only if ranked within the top- K and lower ranks are effectively equivalent to being undiscovered. However, global metrics like AUPRC are insensitive to top-ranked precision, and are further distorted by the severe class imbalance in GRNs leaving many candidate pairs unevaluated under standard negative sampling.

To resolve this, we reformulate GRN inference as a **Knowledge Graph Completion (KGC)** problem. Each regulation is modeled as a structural triple $(v_i, \text{regulates}, v_j)$, where v_i is the source TF (head) and v_j is the target gene (tail). The task of finding targets for a holdout TF naturally translates to *tail entity prediction*: given a query $(v_i, \text{regulates}, ?)$, the model must rank all candidate tails $t \in \mathcal{V}$ such that the ground-truth targets are placed as high as possible.

Aligned with this KGC formulation, we evaluate GRN inference as a resource-constrained ranking problem, adopting Hits@ K for top- K recovery under a fixed validation budget and Mean Reciprocal

Rank (MRR) for capturing overall ranking quality across heterogeneous budgets:

$$\text{Hits@}K = \frac{1}{|\mathcal{Q}|} \sum_{q \in \mathcal{Q}} \mathbf{1}[\text{rank}(t_q) \leq K], \quad \text{MRR} = \frac{1}{|\mathcal{Q}|} \sum_{q \in \mathcal{Q}} \frac{1}{\text{rank}(t_q)}, \quad (1)$$

where $\mathcal{Q}$ is the set of evaluation queries targeting the holdout TFs, and $\text{rank}(t_q)$ is the rank of the ground-truth tail. Following standard KGC practice, we employ the *filtered setting*, which excludes other known valid targets for the same query when calculating the rank. This reformulation guarantees that our benchmark directly quantifies a model’s practical capacity to accelerate discovery within realistic experimental budgets.

4 Modeling Gene-Regulation Co-evolution: The CoDiffGRN Framework

4.1 Preliminaries

Diffusion models consist of a forward noising process and a reverse denoising process [27]. The forward process gradually corrupts data $\mathbf{x}_0$ into noise $\mathbf{x}_T$ through a Markov chain $q(\mathbf{x}_{1:T} | \mathbf{x}_0) = \prod_{t=1}^T q(\mathbf{x}_t | \mathbf{x}_{t-1})$. The reverse Markov process is parameterized by neural networks and progressively denoises the latent states from $\mathbf{x}_T$ to $\mathbf{x}_0$. The reverse trajectory is modeled as

$$p_\theta(\mathbf{x}_{0:T}) = q(\mathbf{x}_T) \prod_{t=1}^T p_\theta(\mathbf{x}_{t-1} | \mathbf{x}_t). \quad (2)$$

Diffusion process. We base our formulation on discrete diffusion spaces. Following D3PM [18], we consider one-hot discrete random variables with K categories, where $\mathbf{x}_0, \mathbf{x}_1, \dots, \mathbf{x}_t \in \{0, 1\}^K$. The forward transition probabilities are encoded by matrices $Q_t \in \mathbb{R}^{K \times K}$, where $[Q_t]_{ij} = q(x_t = j | x_{t-1} = i)$. The transition distribution is defined as:

$$q(\mathbf{x}_t | \mathbf{x}_{t-1}) = \text{Cat}(\mathbf{x}_t; \mathbf{p} = \mathbf{x}_{t-1} Q_t), \quad (3)$$

where $\text{Cat}(\mathbf{x}; \mathbf{p})$ denotes a categorical distribution over the one-hot vector $\mathbf{x}$ with probabilities $\mathbf{p}$. The marginal at timestep t follows directly from the Markov property:

$$q(\mathbf{x}_t | \mathbf{x}_0) = \text{Cat}(\mathbf{x}_t; \mathbf{p} = \mathbf{x}_0 \bar{Q}_t), \quad \text{where} \quad \bar{Q}_t = Q_1 Q_2 \cdots Q_t. \quad (4)$$

Reverse process. The reverse process initializes $\mathbf{x}_T \sim \pi(\mathbf{x})$ and generates $\mathbf{x}_0$ by reversing the timesteps $t = T, T-1, \dots, 0$. Following prior work [18, 27], a neural network $\text{nn}_\theta(\mathbf{x}_t)$ predicts $\tilde{p}_\theta(\tilde{\mathbf{x}}_0 | \mathbf{x}_t)$. The reverse transition is then computed as:

$$p_\theta(\mathbf{x}_{t-1} | \mathbf{x}_t) \propto \sum_{\tilde{\mathbf{x}}_0} q(\mathbf{x}_{t-1}, \mathbf{x}_t | \tilde{\mathbf{x}}_0) \tilde{p}_\theta(\tilde{\mathbf{x}}_0 | \mathbf{x}_t). \quad (5)$$

By Bayes’ theorem and the Markov property, the joint posterior admits a tractable form:

$$q(\mathbf{x}_{t-1}, \mathbf{x}_t | \mathbf{x}_0) = \frac{q(\mathbf{x}_{t-1} | \mathbf{x}_t, \mathbf{x}_0) q(\mathbf{x}_{t-1} | \mathbf{x}_0)}{q(\mathbf{x}_t | \mathbf{x}_0)} = \text{Cat} \left(\mathbf{x}_{t-1}; \mathbf{p} = \frac{\mathbf{x}_t Q_t^\top \odot \mathbf{x}_0 \bar{Q}_{t-1}}{\mathbf{x}_0 \bar{Q}_t \mathbf{x}_t^\top} \right). \quad (6)$$

Choice of transition matrix and conditioning. Different designs of Q_t induce distinct inductive biases [18]. A simple yet effective parameterization is $Q_t = \alpha_t \mathbf{I} + (1 - \alpha_t) \mathbf{1m}^\top$, where $\mathbf{m}$ denotes the marginal distribution and α_t controls the noise scale. The cumulative transition admits a closed form $\bar{Q}_t = \bar{\alpha}_t \mathbf{I} + (1 - \bar{\alpha}_t) \mathbf{1m}^\top$, where $\bar{\alpha}_t = \prod_{\tau=1}^t \alpha_\tau$. The noise schedule is governed by a cosine function [34]. Furthermore, to enable conditional generation, the reverse process learns a conditional data distribution $p_\theta(\mathbf{x}_{t-1} | \mathbf{x}_t, \mathbf{c})$, where auxiliary condition $\mathbf{c}$ is injected into the denoiser to guide the reverse trajectory.

4.2 Joint Discrete Diffusion on Gene Regulatory Network

As illustrated in Figure 2, we propose a joint discrete diffusion framework for GRNs. In contrast to prior works that decouple gene features from edge diffusion [24, 25], our framework unifies discretized gene expression and regulatory interactions into a single co-evolutionary state space.

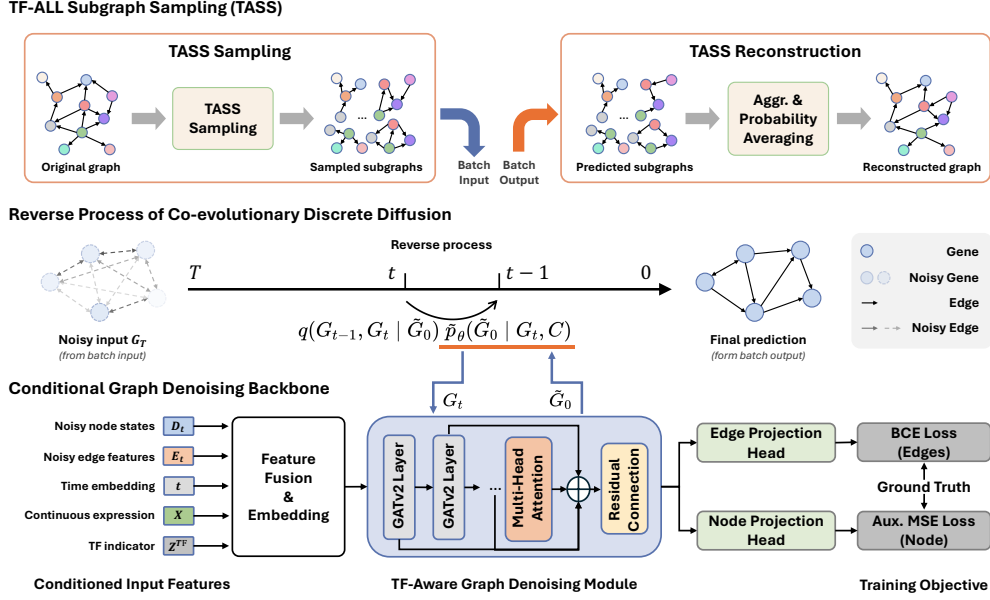


Figure 2: **Overview of the CoDiffGRN framework.** The framework combines TF-ALL Subgraph Sampling (TASS) with a co-evolutionary discrete diffusion model. TASS enables efficient training of a multi-conditioned TF-Aware Graph Denoising Module. At inference, predictions from multiple subgraphs are aggregated via the reverse diffusion process. *Abbr.*, Aggr.: Aggregation.

Cell-cluster discretization. To mitigate feature collapse on unseen genes, we transform continuous scRNA-seq profiles into biologically coherent discrete states. Prior studies show that discretized gene activation patterns capture stable functional structures across diverse biological settings [35, 36]. We cluster cells into k groups and binarize gene expression within each cluster via thresholding, producing a k -bit activation pattern per gene. Each binary pattern is mapped to a categorical variable and represented as a one-hot vector in $\mathbb{R}^{F_D}$, where $F_D = 2^k$.

Joint modeling. We follow a joint diffusion formulation [37] to unify gene expression and regulatory edges. Edges are represented as $\mathbf{E} \in \mathbb{R}^{N \times N \times 2}$ (indicating absence/presence), and node features as $\mathbf{D} \in \mathbb{R}^{N \times F_D}$. The global graph token for each node i is constructed via concatenation: $\mathbf{G}[i] = [\mathbf{d}_i \parallel \mathbf{e}_{i1} \parallel \dots \parallel \mathbf{e}_{iN}]$, where $\mathbf{d}_i \in \mathbb{R}^{F_D}$ denotes the discretized node features, and $\mathbf{e}_{ij} \in \mathbb{R}^2$ is the one-hot encoding of the edge from node i to node j .

Design of the transition matrix. We formulate a joint transition matrix $Q_G \in \mathbb{R}^{F_G \times F_G}$, constructed from four component transition probability matrices: Q_D (gene $\rightarrow$ gene), Q_E (regulation $\rightarrow$ regulation), Q_{ED} (regulation $\rightarrow$ gene), and Q_{DE} (gene $\rightarrow$ regulation):

$$Q_G = \begin{bmatrix} Q_D & \mathbf{1}_N^\top \otimes Q_{DE} \\ \mathbf{1}_N \otimes Q_{ED} & \mathbf{1}_{N \times N} \otimes Q_E \end{bmatrix}, \quad (7)$$

where $\otimes$ denotes the Kronecker product, and $\mathbf{1}_N$, $\mathbf{1}_{N \times N}$ are all-ones vectors and matrices of appropriate dimensions. Following Vignac et al. [38], Q_D and Q_E are derived from the marginal distributions $\mathbf{m}_D$ and $\mathbf{m}_E$ computed from the training data. For the cross-dependencies, $\mathbf{m}_{ED}$ and $\mathbf{m}_{DE}$ capture the conditional co-occurrence probabilities between gene expression types and regulation types. This explicitly ensures that edge transitions are biologically conditioned on the states of their endpoint genes.

Forcing true D technique. During training, the reverse process approximates $p_\theta(G_{t-1} | G_t) \propto \sum_{\tilde{G}_0} q(G_{t-1}, G_t | \tilde{G}_0) \tilde{p}_\theta(\tilde{G}_0 | G_t)$. To mitigate error accumulation and sharpen the learning signal for unknown edge structures, we apply a teacher-forcing-style technique that replaces the predicted node component $\tilde{\mathbf{D}}_0$ within the generated $\tilde{G}_0$ with the ground-truth $\mathbf{D}_0$.

4.3 Conditional Graph Denoising Architecture

Conditioning mechanism. We incorporate multiple sources of conditioning information into the denoiser to guide the reverse process. Combining sinusoidal timestep embeddings $\mathbf{t}_i$, continuous gene expression features $\mathbf{x}_i$, noisy discrete node states $\mathbf{d}_{t,i}$, flattened incoming/outgoing edge features $\mathbf{e}_{t,i}$, and a boolean TF indicator $\mathbf{z}_i^{\text{TF}}$, the conditioned input feature for node i is constructed as:

$$\mathbf{h}_i^{(0)} = [\mathbf{t}_i \parallel \mathbf{x}_i \parallel \mathbf{d}_{t,i} \parallel \mathbf{e}_{t,i} \parallel \mathbf{z}_i^{\text{TF}}]. \quad (8)$$

Denoiser backbone. We parameterize the reverse process using a TF-aware graph denoiser built upon GATv2 [39]. Each layer updates node representations dynamically:

$$\mathbf{h}_i^{(l+1)} = \sigma \left(\sum_{j \in \mathcal{N}(i)} \alpha_{ij}^{(l)} \mathbf{W}^{(l)} \mathbf{h}_j^{(l)} \right), \quad (9)$$

where the attention coefficients $\alpha_{ij}^{(l)}$ are computed via:

$$\alpha_{ij}^{(l)} = \text{softmax}_j \left(\mathbf{a}^\top \text{LeakyReLU}(\mathbf{W}^{(l)} [\mathbf{h}_i^{(l)} \parallel \mathbf{h}_j^{(l)}]) \right). \quad (10)$$

The node representations are then projected into TF-specific ($\mathbf{h}_i^{\text{tf}}$) and target-specific ($\mathbf{h}_j^{\text{tg}}$) embedding spaces. A score head yields a scalar compatibility score $s_{ij} = f_{\text{score}}([\mathbf{h}_i^{\text{tf}} \parallel \mathbf{h}_j^{\text{tg}}])$. Binary edge logits are formed as $[-s_{ij}, s_{ij}]$ and added residually to the noisy edge input $\mathbf{E}_t$. Node logits are obtained via a linear projection of averaged embeddings with a residual connection to $\mathbf{D}_t$.

Training objective. The denoiser is trained using a weighted combination of discrete edge reconstruction loss ($\mathcal{L}_{\text{edge}}$) and a node-level auxiliary regression objective ($\mathcal{L}_{\text{node}}$). Let $\hat{e}_{i,j}$ denote the predicted probability of edge existence and $e_{i,j} \in \{0, 1\}$ the ground-truth. The total loss is defined as:

$$\mathcal{L} = - \sum_{i,j} (e_{i,j} \log \hat{e}_{i,j} + (1 - e_{i,j}) \log(1 - \hat{e}_{i,j})) + \lambda \frac{1}{N} \sum_{i=1}^N \left\| \hat{\mathbf{x}}_{0,i} - \mathbf{x}_i \right\|_2^2, \quad (11)$$

where the auxiliary MSE loss preserves continuous biological signals within the discretized space, and $\lambda \ll 1$ controls its strength.

4.4 TF-ALL Subgraph Sampling (TASS)

Diffusion models typically require large training datasets, but GRN inference is inherently data-scarce since each dataset constitutes a single large graph. To address this, we propose TF-ALL Subgraph Sampling (TASS), which constructs a training set by sampling subgraphs from the input graph (Figure 1(c)). The model is trained on these subgraphs and, at inference time, repeatedly samples subgraphs, predicts edges, and aggregates results to reconstruct the full graph. This expands effective supervision by exposing diverse gene-topology patterns, improving generalization under limited data.

Sampling. As shown in Figure 2, we adopt node-based sampling due to the lack of reliable edge priors, following uniform node sampling [40]. Exploiting the bipartite-like structure of GRNs (TFs $\rightarrow$ targets), each subgraph of size k is formed by sampling a fraction t (TF-to-ALL ratio) of nodes from the TF set, and the remainder from all genes, with resampling for TF duplicates. Let $p = k/n$. Following theoretical bounds for graph recovery [40], we set the number of subgraphs to $m = \lceil p^{-2} \log n \log(1/\delta) \rceil$. In GRNs, this bipartite sampling strategy inherently yields higher edge coverage due to the sparse, hierarchical nature of regulatory interactions.

Reconstruction. At inference, we apply the same sampling strategy, generate predictions across subgraphs, and map them back to the global edge indices. Outputs are converted to probabilities via softmax, and overlapping predictions are averaged to form the final consensus. Uncovered edges are assigned a default baseline prior.

Table 2: Benchmark results on BEELINE-KGC (**Specific** network). We report *mean* over runs. The best, second-best, and third-best are denoted by **bold**, underline, and *italic*, respectively. Δ indicates change relative to the baseline: **green** up, **red** down. Leading zeros are omitted. *Abbr.*, Trad.: Traditional methods, DL.: Feature-based deep learning models, GNN.: GNN-based models, Diff.: Diffusion-based models.

Model	hESC			hHEP			mDC			mESC			mHSC-E			mHSC-GM			mHSC-L				
	H@10	H@50	MRR	H@10	H@50	MRR	H@10	H@50	MRR	H@10	H@50	MRR	H@10	H@50	MRR	H@10	H@50	MRR	H@10	H@50	MRR		
Setting: TFs+500																							
DL	Trad.	GENIE3	.000	.002	.003	.001	.004	.007	.000	.009	.003	.004	.013	.007	.001	.005	.004	.003	.006	.005	.004	.004	.005
		GRNBoost2	.000	.002	.003	.000	.004	.006	.000	.009	.003	.001	.013	.006	.000	.005	.004	.003	.006	.006	.000	.004	.004
DL	CNNC	.006	.006	.005	.009	.009	.013	.009	.009	.004	.026	.036	.019	.000	.005	.004	.000	.000	.005	.000	.000	.004	
		DeepSEM	.000	.004	.003	.001	.002	.007	.000	.000	.003	.005	.017	.007	.000	.002	.004	.000	.003	.005	.000	.000	.004
GNN	GNNLink	.026	.110	.015	.059	.200	.045	.012	.037	.004	.037	.157	.020	.049	.141	.032	.044	.119	.037	.055	.152	.024	
		GENELink+	.052	.218	.030	.141	.385	.100	.025	.077	.009	.082	.343	.050	.124	.355	.073	.115	.290	.072	.122	.304	.067
		GRNFormer	.036	.150	.020	.093	.248	.072	.017	.047	.006	.061	.250	.030	.076	.189	.052	.069	.172	.044	.077	.216	.038
		GCLink	.033	.131	.016	.079	.224	.060	.014	.040	.005	.046	.202	.024	.066	.180	.043	.057	.148	.040	.064	.170	.030
Diff	RegDiffusion	.033	.151	.021	.077	.230	.058	.023	.071	.009	.051	.195	.033	.133	.362	.074	.094	.249	.065	.066	.162	.036	
		DigNet	.063	.242	.029	.138	.355	.097	.019	.055	.006	.051	.236	.035	.168	.361	.088	.079	.216	.056	.139	.314	.085
		CoDiffGRN	.079	.264	.034	.173	.499	.126	.028	.122	.014	.123	.412	.060	.175	.407	.099	.167	.463	.108	.162	.361	.111
		$\Delta(\%)$	+24.9%	+9.1%	+13.0%	+22.1%	+29.6%	+25.1%	+11.3%	+57.1%	+52.7%	+49.3%	+20.1%	+20.2%	+4.1%	+12.6%	+11.8%	+44.7%	+59.5%	+48.9%	+16.6%	+14.8%	+30.4%
Setting: TFs+1000																							
DL	Trad.	GENIE3	.000	.000	.002	.001	.003	.005	.000	.007	.002	.001	.004	.004	.001	.018	.001	.643	.017	.002	.006	.007	
		GRNBoost2	.000	.000	.002	.000	.003	.004	.000	.007	.002	.001	.004	.004	.000	.280	.017	.000	.643	.017	.000	.006	.006
DL	CNNC	.005	.015	.004	.006	.008	.008	.000	.000	.002	.011	.022	.009	.017	.295	.030	.010	.652	.024	.000	.003	.006	
		DeepSEM	.000	.002	.002	.001	.003	.005	.000	.000	.002	.003	.008	.004	.001	.287	.018	.003	.647	.018	.000	.006	.006
GNN	GNNLink	.010	.037	.006	.029	.142	.016	.007	.028	.006	.038	.081	.029	.050	.160	.020	.112	.301	.052	.086	.303	.035	
		GENELink+	.019	.070	.010	.067	.340	.038	.014	.061	.009	.086	.211	.058	.073	.301	.046	.249	.648	.101	.166	.531	.073
		GRNFormer	.015	.051	.007	.045	.216	.024	.008	.034	.007	.058	.114	.040	.058	.196	.031	.144	.367	.066	.105	.420	.047
		GCLink	.012	.043	.006	.035	.163	.021	.007	.031	.006	.044	.093	.035	.055	.186	.025	.129	.333	.062	.098	.361	.038
Diff	RegDiffusion	.022	.089	.014	.068	.347	.043	.009	.046	.007	.083	.191	.055	.060	.230	.041	.272	.602	.106	.100	.306	.038	
		DigNet	.021	.071	.011	.067	.335	.044	.013	.065	.009	.117	.266	.077	.063	.249	.039	.217	.609	.089	.135	.441	.055
		CoDiffGRN	.027	.108	.015	.079	.398	.053	.016	.069	.010	.139	.299	.093	.091	.429	.051	.334	.711	.140	.276	.587	.117
		$\Delta(\%)$	+23.5%	+22.3%	+12.3%	+15.3%	+14.7%	+20.2%	+13.9%	+5.8%	+14.1%	+19.4%	+12.1%	+21.1%	+25.4%	+42.5%	+11.7%	+20.7%	+9.0%	+31.4%	+66.1%	+10.5%	+61.6%

5 Experiments

5.1 Experimental Setup

Evaluation protocols. We primarily evaluate performance using Hits@K and MRR on Inductive Gene-Holdout Split introduced in Section 3. To reflect practical experimental settings, we report Hits@10 and Hits@50, corresponding to low- and high-throughput validation regimes.

Baseline models. We compare against a comprehensive set of baselines, including two traditional methods (GENIE3 [22], GRNBoost2 [20]), two feature-based deep learning models (CNNC [21], DeepSEM [23]), four graph-based models (GNNLink [9], GENELink+ [10], GRNFormer [11], GCLink [12]), and two diffusion-based methods (RegDiffusion [24], DigNet [25]).

Model setups. For fair comparison, all baselines are implemented following their original configurations. Our model adopts the TF-Aware Graph Denoiser, built upon a multi-layer GATv2 architecture with three attention heads per layer and progressively reduced hidden dimensions (128, 64, 64, 32), producing a 16-dimensional output embedding. We use a LeakyReLU with negative slope 0.2 and concatenate multi-head outputs. Score head f_{score} is implemented as a two-layer MLP.

Training setups. All baselines are trained according to their original protocols. Our model is optimized using AdamW with a learning rate of 1×10^{-4} , weight decay 1×10^{-3} , batch size 64, and trained for 500 epochs. We use a discrete state space with $k = 4$ clusters (yielding 16 classes) and 500 diffusion steps. Subgraphs are sampled with 100 nodes each. All experiments are conducted on a local server with 100 CPU cores and 4 NVIDIA RTX 4090 GPUs (24 GB), and are run over three random seeds (0, 1, and 2), with both mean and standard deviation reported.

5.2 Results

We report the performance of representative models on the Specific network under the proposed BEELINE-KGC in Table 2. Based on these results, we highlight the following key findings:

- **The proposed BEELINE-KGC benchmark is substantially more challenging than existing protocols.** Traditional and feature-based methods largely collapse under inductive gene holdout, yielding near-zero Hits@10 and MRR. Even several GNN-based models degrade substantially, indicating that BEELINE-KGC reveals inductive generalization failures obscured by transductive benchmarks and global metrics.

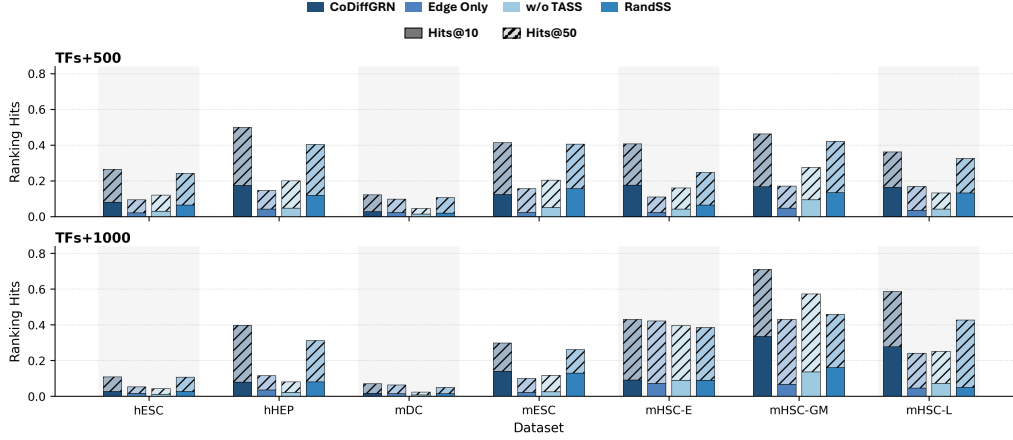


Figure 3: Ablation study results on BEELINE-KGC (**Specific** network). We report *mean* performance of Hits@10 and Hits@50 over runs, under the TFs+500 (top) and TFs+1000 (bottom) settings. *Abbr.*, w/o: without.

- **GNN-based models show limited inductive generalization.** While GENELink+ remains competitive, most GNN methods suffer from low performance. Their heavy reliance on predefined prior graphs leads to disconnected components for unseen genes, causing error accumulation during message passing and widening the gap against diffusion-based approaches.
- **Diffusion-based models offer partial robustness yet remain insufficient.** According to Table 2, diffusion-based methods generally outperform GNN baselines and match GENELink+ in some cases, but still fall behind CoDiffGRN. This suggests that existing diffusion models alleviate reliance on explicit graph structure, yet remain vulnerable to feature collapse on unseen genes. Moreover, their single-graph optimization paradigm appears suboptimal for diffusion.
- **CoDiffGRN achieves consistent SOTA performance.** CoDiffGRN ranks first across all cell types and metrics in both TFs+500 and TFs+1000 settings, with relative gains up to +184.8% (and +24.5% on average) over the strongest baselines on the most competitive Specific network. These results validate the effectiveness of our co-evolutionary discrete diffusion framework with gene-state-dependent transitions and the TASS strategy for addressing inductive GRN inference and top- K ranking quality.

5.3 Ablation Study

To verify the effectiveness of our designs, we conduct extensive ablation experiments on BEELINE-KGC, focusing on the joint diffusion mechanism and the TASS augmentation strategy. The variants are: (i) **Edge Only**: a variant that only using edge diffusion; (ii) **w/o TASS**: the model trained on the original graph without TASS; and (iii) **RandSS**: replacing TASS with uniform random subgraph sampling.

Effect of joint modeling. As shown in Figure 3, removing joint modeling (Edge Only) consistently degrades performance across datasets and metrics. This indicates that decoupling gene expression states from regulatory edges limits the model’s ability to capture conditional regulatory interactions and reduces top- K ranking quality, especially when generalizing to unseen genes in inductive settings.

Effect of subgraph sampling. Removing TASS leads to a substantial performance drop across all experimental settings. In contrast, uniform random sampling (RandSS) partially recovers the performance and achieves superior results in a few isolated cases, but remains consistently less effective overall. These results highlight the importance of informative subgraph construction and demonstrate that the task-aware sampling strategy of TASS provides more reliable improvements than generic random sampling under inductive and data-scarce GRN inference settings.

6 Conclusion

In this work, we address the gap between computational GRN inference and real-world biological discovery. We introduce BEELINE-KGC, an application-aligned framework that enforces an inductive, KGC-based evaluation protocol to reflect the top- K constraints of wet-lab validation. To overcome the exposed gene-level generalization gap, we propose CoDiffGRN, the first joint node-edge discrete diffusion model for GRNs. By capturing the co-evolution of gene expression and network topology, and further enhanced by our TASS strategy, CoDiffGRN achieves robust inductive generalization. Together, our benchmark and model establish a unified paradigm for reliable and budget-aware biological discovery in systems biology.

Limitations and Future Work. Despite its performance, our approach has several limitations that suggest directions for future research. The current evaluation is restricted to established scRNA-seq systems, which may not fully represent the complexity of rare tissues or cross-species regulatory dynamics. We plan to extend our benchmark to include more diverse biological contexts and validate the model’s transferability across different species. Moreover, the joint diffusion process and TASS strategy introduce non-trivial computational overhead during training and inference. Future work will focus on developing optimized sampling kernels and parallelized subgraph processing to enhance scalability.

References

- [1] Yu-Chieh Wang, Suzanne E Peterson, and Jeanne F Loring. Protein post-translational modifications and regulation of pluripotency in human stem cells. *Cell research*, 24(2):143–160, 2014.
- [2] Constantinos Chronis, Petko Fiziev, Bernadett Papp, Stefan Butz, Giancarlo Bonora, Shan Sabri, Jason Ernst, and Kathrin Plath. Cooperative binding of transcription factors orchestrates reprogramming. *Cell*, 168(3):442–459, 2017.
- [3] Michael Hecker, Sandro Lambeck, Susanne Toepfer, Eugene Van Someren, and Reinhard Guthke. Gene regulatory network inference: data integration in dynamic models—a review. *Biosystems*, 96(1):86–103, 2009.
- [4] Patricia J Wittkopp and Gizem Kalay. Cis-regulatory elements: molecular mechanisms and evolutionary processes underlying divergence. *Nature Reviews Genetics*, 13(1):59–69, 2012.
- [5] Aleksandra A Kolodziejczyk, Jong Kyoung Kim, Valentine Svensson, John C Marioni, and Sarah A Teichmann. The technology and biology of single-cell rna sequencing. *Molecular cell*, 58(4):610–620, 2015.
- [6] Allon Wagner, Aviv Regev, and Nir Yosef. Revealing the vectors of cellular identity with single-cell genomics. *Nature biotechnology*, 34(11):1145–1160, 2016.
- [7] Peter V Kharchenko, Lev Silberstein, and David T Scadden. Bayesian approach to single-cell differential expression analysis. *Nature methods*, 11(7):740–742, 2014.
- [8] Guangyi Chen and Zhi-Ping Liu. Graph attention network for link prediction of gene regulations from single-cell rna-sequencing data. *Bioinformatics*, 38(19):4522–4529, 2022.
- [9] Guo Mao, Zhengbin Pang, Ke Zuo, Qinglin Wang, Xiangdong Pei, Xinhai Chen, and Jie Liu. Predicting gene regulatory links from single-cell rna-seq data using graph neural networks. *Briefings in Bioinformatics*, 24(6):bbad414, 2023.
- [10] Wei Zhang, Bowen Shao, Wenrui Li, Wenbo Guo, Jiaxin Lyu, Guangyi Chen, Chuanyuan Wang, and Zhi-Ping Liu. Inferring cell-type-specific gene regulatory network from cellular transcriptomics data with genelink+. *Briefings in Bioinformatics*, 26(4):bbaf359, 2025.
- [11] Akshata Hegde and Jianlin Cheng. Grnformer: accurate gene regulatory network inference using graph transformer. *Bioinformatics*, 42(4):btac144, 2026.
- [12] Weiming Yu, Zerun Lin, Miaofang Lan, and Le Ou-Yang. Gclink: a graph contrastive link prediction framework for gene regulatory network inference. *Bioinformatics*, 41(3):btaf074, 2025.
- [13] Jiacheng Wang, Yaojia Chen, and Quan Zou. Inferring gene regulatory network from single-cell transcriptomes with graph autoencoder model. *PLoS Genetics*, 19(9):e1010942, 2023.
- [14] Aditya Pratapa, Amogh P Jalihal, Jeffrey N Law, Aditya Bharadwaj, and T M Murali. Benchmarking algorithms for gene regulatory network inference from single-cell transcriptomic data. *Nature methods*, 17(2):147–154, 2020.
- [15] Jingwen W Ding, Chang N Kim, Megan S Ostrowski, Yashodara Abeykoon, Bryan J Pavlovic, Jenelle L Wallace, Nathan K Schaefer, Tomasz J Nowakowski, and Alex A Pollen. Dissecting gene regulatory networks governing human cortical cell fate. *Nature*, pages 1–11, 2026.
- [16] Rachel B Gilmore, Yaling Liu, Christopher E Stoddard, Michael S Chung, Gordon G Carmichael, and Justin Cotney. Identifying key underlying regulatory networks and predicting targets of orphan c/d box snord116 snornas in prader–willi syndrome. *Nucleic Acids Research*, 52(22):13757–13774, 2024.
- [17] Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. Translating embeddings for modeling multi-relational data. *Advances in neural information processing systems*, 26, 2013.

- [18] Jacob Austin, Daniel D Johnson, Jonathan Ho, Daniel Tarlow, and Rianne Van Den Berg. Structured denoising diffusion models in discrete state-spaces. *Advances in neural information processing systems*, 34:17981–17993, 2021.
- [19] Itunuoluwa Isewon, Mariam Kanonte, Emmanuel Alagbe, and Jelili Oyelade. Benchmarking frameworks for gene regulatory network inference methods: A review. *NIPES JSTR SPECIAL ISSUE*, 7(1):2224–2228, 2025.
- [20] Thomas Moerman, Sara Aibar Santos, Carmen Bravo González-Blas, Jaak Simm, Yves Moreau, Jan Aerts, and Stein Aerts. Grnboost2 and arboreto: efficient and scalable inference of gene regulatory networks. *Bioinformatics*, 35(12):2159–2161, 2019.
- [21] Ye Yuan and Ziv Bar-Joseph. Deep learning for inferring gene relationships from single-cell expression data. *Proceedings of the National Academy of Sciences*, 116(52):27151–27158, 2019.
- [22] Vân Anh Huynh-Thu, Alexandre Irrthum, Louis Wehenkel, and Pierre Geurts. Inferring regulatory networks from expression data using tree-based methods. *PloS one*, 5(9):e12776, 2010.
- [23] Hantao Shu, Jingtian Zhou, Qiuyu Lian, Han Li, Dan Zhao, Jianyang Zeng, and Jianzhu Ma. Modeling gene regulatory networks using neural network architectures. *Nature Computational Science*, 1(7):491–501, 2021.
- [24] Hao Zhu and Donna Slonim. From noise to knowledge: diffusion probabilistic model-based neural inference of gene regulatory networks. *Journal of Computational Biology*, 31(11):1087–1103, 2024.
- [25] Chuanyuan Wang and Zhi-Ping Liu. Diffusion-based generation of gene regulatory networks from scrna-seq data with dignet. *Genome Research*, 35(2):340–354, 2025.
- [26] Shiyu Xu, Na Yu, Daoliang Zhang, and Chuanyuan Wang. Attention-guided probabilistic diffusion model for generating cell-type-specific gene regulatory networks from gene expression profiles. *Genes*, 16(11):1255, 2025.
- [27] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.
- [28] Shruthi Subramanian, Julie AI Thoms, Yizhou Huang, Paola Cornejo-Páramo, Forrest C Koch, Sebastien Jacquelin, Sylvie Shen, Emma Song, Swapna Joshi, Chris Brownlee, et al. Genome-wide transcription factor-binding maps reveal cell-specific changes in the regulatory architecture of human hspcs. *Blood*, 142(17):1448–1462, 2023.
- [29] Caroline RM Wiggers, Burak Yüzügüldü, Nathaniel G Tadros, Tayla B Heavican-Foral, Eugene Y Cho, Zachary C Eisenbies, Merve Ozdemir, Steffen B Kulp, Yun-Cheol Chae, Alejandro Gutierrez, et al. Genome-wide crispr screen identifies irf1 and tfap4 as transcriptional regulators of galectin-9 in t cell acute lymphoblastic leukemia. *Science advances*, 11(12):eads8351, 2025.
- [30] Qiong Zhang, Wei Liu, Hong-Mei Zhang, Gui-Yan Xie, Ya-Ru Miao, Mengxuan Xia, and An-Yuan Guo. htftarget: a comprehensive database for regulations of human transcription factors and their targets. *Genomics, proteomics & bioinformatics*, 18(2):120–128, 2020.
- [31] Junlin Zhang, Bao Liu, Yan Song, Yang Chen, Jiao Fu, Jianquan Liu, Tao Ma, Zhenxiang Xi, and Huanhuan Liu. Genome-wide (chip-seq) identification of target genes regulated by wrky33 during submergence stress in arabidopsis. *BMC genomic data*, 22(1):16, 2021.
- [32] Wei Niu, Zhi John Lu, Mei Zhong, Mihail Sarov, John I Murray, Cathleen M Brdlik, Judith Janette, Chao Chen, Pedro Alves, Elicia Preston, et al. Diverse transcription factor binding features revealed by genome-wide chip-seq in c. elegans. *Genome research*, 21(2):245–254, 2011.
- [33] Emmanouil Bastakis, Jennifer Gerke, Seyma Özkan, Rebekka Harting, Tanja Lienard, Christoph Sasse, Emmanouil S Xylakis, Merle Aden, Anja Strohdiek, Gabriele Heinrich, et al. Molecular circuit between aspergillus nidulans transcription factors msna and velb to coordinate fungal stress and developmental responses. *PLoS Genetics*, 21(7):e1011578, 2025.

- [34] Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *International conference on machine learning*, pages 8162–8171. PMLR, 2021.
- [35] Dominic Grün, Anna Lyubimova, Lennart Kester, Kay Wiebrands, Onur Basak, Nobuo Sasaki, Hans Clevers, and Alexander Van Oudenaarden. Single-cell messenger rna sequencing reveals rare intestinal cell types. *Nature*, 525(7568):251–255, 2015.
- [36] Gerard A Bouland, Ahmed Mahfouz, and Marcel JT Reinders. Differential analysis of binarized single-cell rna sequencing data captures biological variation. *NAR genomics and bioinformatics*, 3(4):lqab118, 2021.
- [37] Gang Liu, Jiaxin Xu, Tengfei Luo, and Meng Jiang. Graph diffusion transformers for multi-conditional molecular generation. *Advances in Neural Information Processing Systems*, 37: 8065–8092, 2024.
- [38] Clement Vignac, Igor Krawczuk, Antoine Siraudin, Bohan Wang, Volkan Cevher, and Pascal Frossard. Digress: Discrete denoising diffusion for graph generation. *arXiv preprint arXiv:2209.14734*, 2022.
- [39] Shaked Brody, Uri Alon, and Eran Yahav. How attentive are graph attention networks? *arXiv preprint arXiv:2105.14491*, 2021.
- [40] Stratis Limnios, Praveen Selvaraj, Mihai Cucuringu, Carsten Maple, Gesine Reinert, and Andrew Elliott. Sagess: Sampling graph denoising diffusion model for scalable graph generation. *arXiv preprint arXiv:2306.16827*, 2023.
- [41] Etienne Becht, Leland McInnes, John Healy, Charles-Antoine Dutertre, Immanuel WH Kwok, Lai Guan Ng, Florent Gehroux, and Evan W Newell. Dimensionality reduction for visualizing single-cell data using umap. *Nature biotechnology*, 37(1):38–44, 2019.
- [42] Meenakshi S Kagda, Bonita Lam, Casey Litton, Corinn Small, Cricket A Sloan, Emma Spragins, Forrest Tanaka, Ian Whaling, Idan Gabdank, Ingrid Youngworth, et al. Data navigation on the encode portal. *Nature Communications*, 16(1):9592, 2025.
- [43] Zhaonan Zou, Tazro Ohta, and Shinya Oki. Chip-atlas 3.0: a data-mining suite to explore chromosome architecture together with large-scale regulome data. *Nucleic Acids Research*, 52 (W1):W45–W53, 2024.
- [44] Huilei Xu, Caroline Baroukh, Ruth Dannenfelser, Edward Y Chen, Christopher M Tan, Yan Kou, Yujin E Kim, Ihor R Lemischka, and Avi Ma’ayan. Escape: database for integrating high-content published data collected from human and mouse embryonic stem cells. *Database*, 2013:bat045, 2013.
- [45] Luz Garcia-Alonso, Christian H Holland, Mahmoud M Ibrahim, Denes Turei, and Julio Saez-Rodriguez. Benchmark and integration of resources for the estimation of human transcription factor activities. *Genome research*, 29(8):1363–1375, 2019.
- [46] Zhi-Ping Liu, Canglin Wu, Hongyu Miao, and Hulin Wu. Regnetwork: an integrated database of transcriptional and post-transcriptional regulatory networks in human and mouse. *Database*, 2015:bav095, 2015.
- [47] Heonjong Han, Jae-Won Cho, Sangyoung Lee, Ayoung Yun, Hyojin Kim, Dasom Bae, Sunmo Yang, Chan Yeong Kim, Muyoung Lee, Eunbeen Kim, et al. Trtrust v2: an expanded reference database of human and mouse transcriptional regulatory interactions. *Nucleic acids research*, 46(D1):D380–D386, 2018.
- [48] Damian Szklarczyk, Rebecca Kirsch, Mikaela Koutrouli, Katerina Nastou, Farrokh Mehryary, Radja Hachilif, Annika L Gable, Tao Fang, Nadezhda T Doncheva, Sampo Pyysalo, et al. The string database in 2023: protein–protein association networks and functional enrichment analyses for any sequenced genome of interest. *Nucleic acids research*, 51(D1):D638–D646, 2023.

A Results on Previous Transductive Setting

Table 3: Results on transductive setting (**Specific** network). Caption settings are the same as Table 2.

Model	hESC			hHEP			mDC			mESC			mHSC-E			mHSC-GM			mHSC-L		
	H@10	H@50	MRR	H@10	H@50	MRR	H@10	H@50	MRR	H@10	H@50	MRR	H@10	H@50	MRR	H@10	H@50	MRR	H@10	H@50	MRR
Setting: TFs+500																					
TRAD. GENIE3	.012	.012	.008	.023	.023	.020	.009	.018	.005	.034	.036	.024	.008	.008	.010	.010	.010	.014	.013	.013	.014
GRNBoost2	.006	.006	.007	.001	.001	.007	.000	.000	.003	.005	.005	.010	.002	.002	.005	.000	.000	.005	.002	.002	.006
DL. CNNC	.028	.140	.016	.047	.198	.028	.011	.109	.013	.058	.237	.033	.021	.103	.016	.032	.149	.018	.016	.095	.015
DeepSEM	.024	.071	.012	.054	.059	.030	.045	.073	.022	.094	.149	.042	.078	.086	.049	.074	.075	.065	.067	.036	
GNN. GNNLink	.015	.051	.013	.001	.006	.007	.027	.130	.015	.000	.020	.006	.000	.002	.003	.010	.022	.006	.017	.025	.015
GENELink+	.040	.169	.022	.137	.377	.057	.016	.071	.010	.103	.358	.053	.329	.598	.167	.048	.213	.029	.130	.330	.057
GRNFormer	.039	.187	.026	.106	.323	.049	.033	.125	.015	.118	.382	.058	.034	.119	.020	.040	.209	.024	.035	.137	.024
GCLink	.098	.296	.048	.146	.394	.072	.049	.141	.031	.188	.461	.092	—	—	—	—	—	.058	.213	.032	
DIFF. RegDiffusion	.031	.148	.017	.035	.140	.019	.022	.060	.018	.053	.176	.025	.040	.208	.024	.087	.311	.044	.081	.257	.036
DigNet	.043	.168	.023	.055	.198	.029	.049	.120	.019	.064	.281	.031	.049	.184	.025	.132	.434	.066	.048	.235	.028
CoDiffGRN	.102	.318	.056	.247	.539	.128	.049	.179	.033	.259	.524	.137	.470	.769	.259	.435	.755	.227	.349	.669	.189
$\Delta(\%)$	+4.1%	+7.3%	+15.9%	+68.7%	+36.9%	+76.3%	+0.0%	+26.9%	+4.8%	+37.6%	+13.9%	+48.7%	+42.8%	+28.7%	+54.9%	+228.7%	+74.0%	+244.1%	+167.6%	+102.7%	+231.2%
Setting: TFs+1000																					
TRAD. GENIE3	.004	.004	.004	.010	.010	.010	.014	.014	.007	.021	.022	.015	.003	.283	.020	.006	.646	.022	.011	.011	.016
GRNBoost2	.000	.000	.002	.001	.001	.006	.000	.000	.002	.005	.005	.008	.001	.281	.017	.002	.643	.018	.000	.000	.006
DL. CNNC	.014	.096	.013	.032	.151	.018	.004	.042	.006	.043	.162	.023	.153	.520	.066	.129	.464	.056	.022	.142	.017
DeepSEM	.024	.046	.008	.044	.049	.023	.014	.035	.008	.064	.112	.032	.046	.322	.043	.043	.668	.045	.043	.043	.037
GNN. GNNLink	.004	.021	.005	.000	.005	.005	.014	.046	.007	.000	.000	.003	.000	.516	.019	.005	.340	.016	.026	.059	.020
GENELink+	.041	.144	.022	.085	.295	.042	.025	.099	.010	.130	.356	.066	.445	.698	.251	.390	.670	.225	.084	.307	.036
GRNFormer	.029	.112	.019	.062	.254	.035	.021	.053	.010	.062	.268	.033	.442	.660	.258	.396	.676	.207	.039	.134	.024
GCLink	.042	.203	.024	.091	.274	.055	.042	.141	.021	.126	.351	.064	—	—	—	—	—	.103	.319	.044	
DIFF. RegDiffusion	.022	.080	.013	.024	.089	.013	.021	.057	.012	.034	.119	.017	.064	.354	.036	.090	.334	.041	.154	.429	.082
DigNet	.027	.107	.013	.026	.116	.016	.042	.064	.008	.042	.191	.023	.063	.346	.038	.055	.288	.034	.121	.407	.060
CoDiffGRN	.079	.269	.035	.203	.439	.115	.053	.159	.025	.245	.502	.128	.464	.708	.267	.410	.702	.235	.311	.609	.144
$\Delta(\%)$	+88.6%	+32.7%	+47.6%	+122.8%	+49.1%	+110.3%	+25.0%	+12.5%	+20.3%	+88.9%	+41.0%	+94.1%	+3.3%	+1.5%	+3.5%	+3.5%	+3.8%	+4.4%	+101.1%	+42.0%	+76.2%

Table 4: Results of AUPRC on transductive setting (**Specific** network).

Model	TFs+500							TFs+1000						
	hESC	hHEP	mDC	mESC	mHSC-E	mHSC-GM	mHSC-L	hESC	hHEP	mDC	mESC	mHSC-E	mHSC-GM	mHSC-L
TRAD. GENIE3	.156	.395	.055	.314	.566	.532	.525	.158	.384	.056	.312	.547	.532	.484
GRNBoost2	.154	.387	.069	.327	.578	.522	.597	.150	.380	.059	.324	.547	.527	.481
DL. CNNC	.255	.469	.066	.484	.746	.682	.648	.271	.499	.058	.597	.774	.737	.565
DeepSEM	.196	.461	.054	.314	.567	.520	.537	.192	.419	.056	.320	.565	.543	.523
GNN. GNNLink	.521	.751	.251	.766	.881	.898	.855	.515	.787	.216	.785	.930	.931	.864
GENELink+	.519	.813	.153	.796	.943	.937	.870	.521	.817	.122	.776	.954	.951	.853
GRNFormer	.503	.713	.121	.802	.874	.869	.764	.522	.744	.125	.797	.917	.915	.743
GCLink	.465	.707	.207	.739	.856	.849	.814	.476	.740	.159	.722	.898	.870	.840
DIFF. RegDiffusion	.447	.808	.117	.761	.932	.925	.863	.483	.814	.132	.784	.944	.944	.859
DigNet	.495	.799	.112	.775	.717	.703	.871	.527	.813	.139	.798	.625	.612	.869
CoDiffGRN	.582	.817	.155	.836	.923	.909	.840	.598	.822	.154	.856	.931	.926	.839
$\Delta(\%)$	+11.8%	+0.5%	-38.4%	+4.3%	-2.0%	-3.0%	-3.6%	+13.3%	+0.5%	-28.4%	+7.2%	-2.4%	-2.6%	-3.5%

B Cell-cluster Discretization

This section provides additional details and supporting evidence for the proposed Cell-cluster-based Discretization strategy. We first present the algorithmic procedure, followed by visualizations that demonstrate the biological coherence of the induced discrete gene state space.

B.1 Algorithm

Algorithm 1 Cell-cluster Discretization

Require: Gene expression matrix $\mathbf{X} \in \mathbb{R}^{N \times F_X}$, number of clusters k

Ensure: Discrete one-hot representations $\mathbf{D} \in \mathbb{R}^{N \times F_D}$, where $F_D = 2^k$

- 1: Construct $\mathbf{X}_{\text{vis}}$ using only visible data
 - 2: Apply k -means on cell-wise expression profiles of $\mathbf{X}_{\text{vis}}$ to obtain clusters $\{\mathcal{C}_1, \dots, \mathcal{C}_k\}$
 - 3: For each cluster $\mathcal{C}_j$, compute $\tau_j = \frac{1}{|\mathcal{C}_j| \cdot N} \sum_{c \in \mathcal{C}_j} \sum_{g=1}^N \mathbf{X}_{g,c}$
 - 4: **for** each gene g **do**
 - 5: **for** each cluster j **do**
 - 6: Compute mean expression $\mu_{g,j} = \frac{1}{|\mathcal{C}_j|} \sum_{c \in \mathcal{C}_j} \mathbf{X}_{g,c}$
 - 7: Set binary activation $b_{g,j} = \mathbb{I}(\mu_{g,j} \geq \tau_j)$
 - 8: **end for**
 - 9: Form binary vector $\mathbf{b}_g \in \{0, 1\}^k$
 - 10: Map $\mathbf{b}_g$ to category index $c_g \in \{0, \dots, F_D - 1\}$
 - 11: Convert c_g to one-hot vector $\mathbf{d} \in \mathbb{R}^{F_D}$
 - 12: **end for**
 - 13: **return** $\mathbf{D} = \{\mathbf{d}\}_{g=1}^N$
-

B.2 Design Motivation and Biological Evidence

We provide the design motivation and biological evidence supporting the proposed Cell-cluster-based Discretization. The goal of this design is to construct a biologically coherent discrete state space that enables stable inductive diffusion over unseen genes.

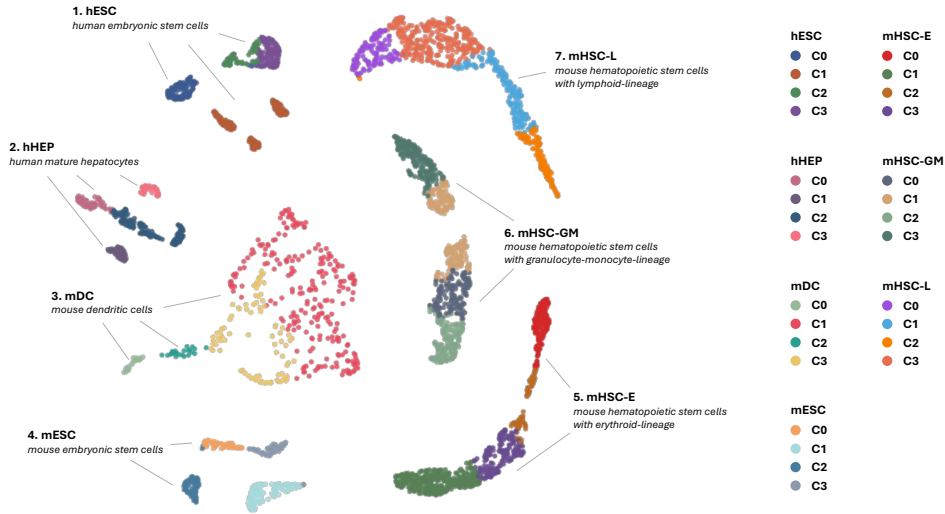


Figure 4: **Cell-state structure induced by clustering.** Visualization of cell clusters obtained from scRNA-seq profiles using UMAP [41]. Each cluster corresponds to a distinct cellular state, supporting the use of cell-level structure as the basis for discretization.

Motivation. A natural alternative is to directly cluster gene expression profiles using standard unsupervised methods (e.g., k-means). However, such approaches may produce clusters dominated by statistical artifacts such as sparsity patterns or zero inflation in scRNA-seq data, rather than biologically meaningful structure. In contrast, scRNA-seq measurements are inherently organized by cellular contexts (e.g., conditions or states), suggesting that discretization should be grounded in cell-level structure to preserve functional coherence.

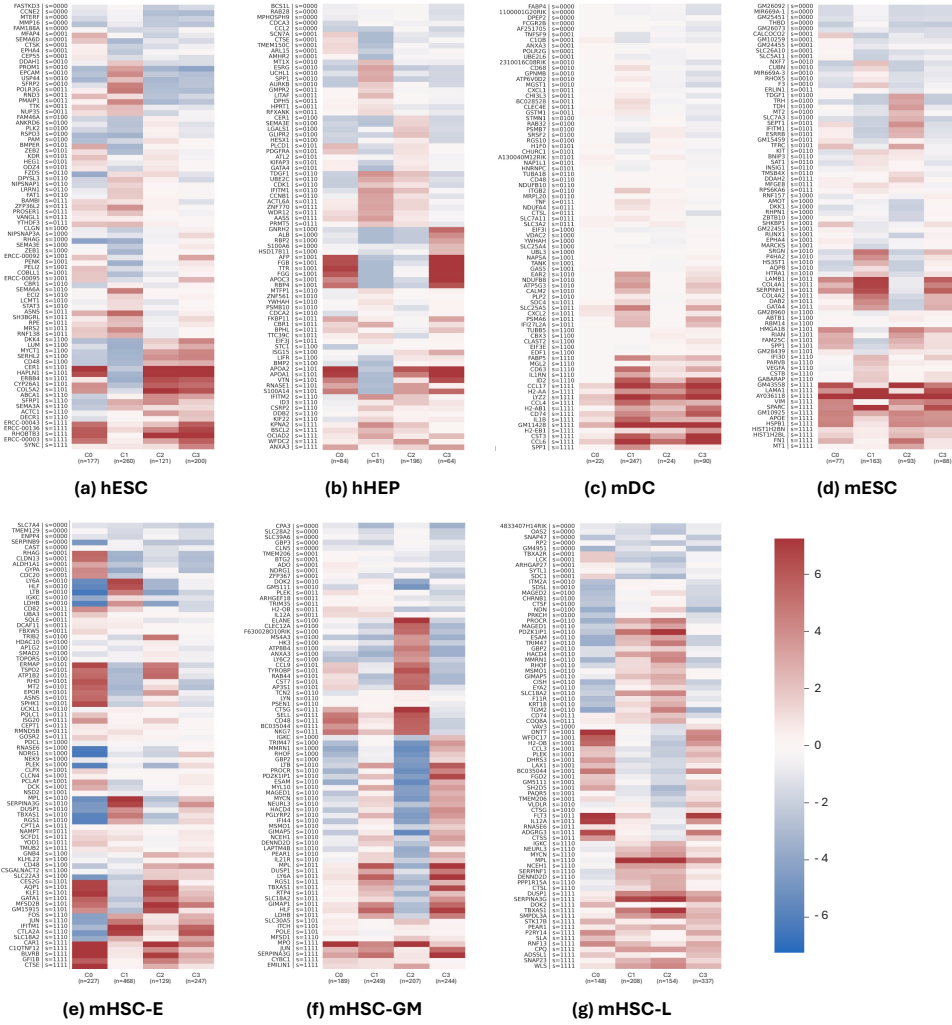


Figure 5: **Gene activation consistency across cell clusters.** Representative genes show stable and coherent activation patterns within each cell cluster, indicating that discretization preserves biologically meaningful gene activity structure.

Design Principle. Our method explicitly leverages this structure by first clustering cells into k biologically meaningful groups, and then constructing gene-level discrete representations based on expression consistency across these cell states. This ensures that discretization reflects stable functional activity rather than superficial similarity in gene expression vectors.

Figure Organization and Evidence. We use UMAP [41] to visualize the resulting cell clusters obtained from scRNA-seq data in Figure 4. The clusters correspond to distinct cellular states, indicating that the clustering captures meaningful biological organization rather than arbitrary partitions.

Figure 5 presents representative genes across different cell clusters. We observe that genes exhibit consistent activation patterns within clusters, suggesting that discretization captures stable gene-level

functional behavior conditioned on cellular context. This is further proved by Figure 6, which visualizes the induced gene representation manifold in the discrete state space using UMAP [41]. The resulting structure exhibits smooth organization and local coherence, suggesting that the discretization preserves biologically meaningful continuity while providing a structured state space suitable for diffusion modeling.

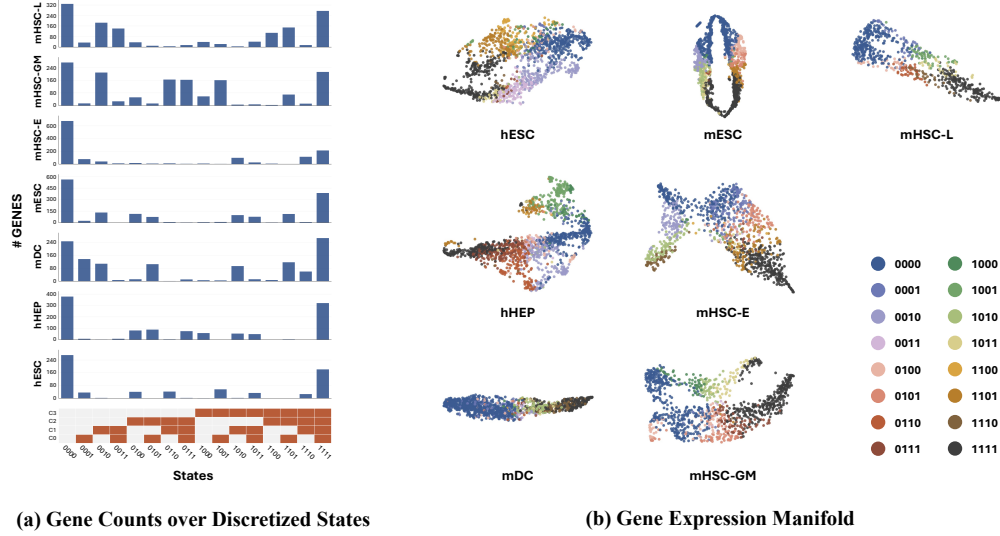


Figure 6: **Discrete gene manifold structure.** The induced gene representation space exhibits smooth and coherent structure, indicating that the discretization preserves biologically meaningful organization while enabling stable diffusion over discrete states.

Together, these results support the role of Cell-cluster-based Discretization as a biologically grounded representation scheme that enables stable inductive generalization under sparse and heterogeneous scRNA-seq data.

C TF-ALL Subgraph Sampling (TASS)

In this section, we present the complete pipeline of TF-ALL Subgraph Sampling (TASS), which consists of a stochastic subgraph sampling strategy followed by a consensus-based reconstruction procedure for global edge inference.

C.1 Sampling

Algorithm 2 TF-ALL Subgraph Sampling (TASS)

Require: Graph $G = (V, E)$, TF set V_{TF} , gene set V_{G} , subgraph size k
Ensure: Subgraph collection $\mathcal{S} = \{G_1, \dots, G_m\}$

- 1: Compute $n \leftarrow |V|$, $p \leftarrow k/n$, TF-ALL ratio $t \leftarrow |V_{\text{TF}}|/|V_{\text{G}}|$
- 2: Calculate theoretical bound $m_{\text{base}} \leftarrow \lceil p^{-2} \log n \log(1/\delta) \rceil$
- 3: Set $m = 100 \cdot \lceil m_{\text{base}}/100 \rceil$ ▷ Round up to nearest multiple of 100
- 4: Initialize $\mathcal{S} \leftarrow \emptyset$
- 5: **for** $i = 1$ to m **do**
- 6: Sample TF nodes: $V_{\text{TF}}^{(i)} \sim \text{Uniform}(V_{\text{TF}}, \lfloor |V_{\text{TF}}| \cdot t \rfloor)$
- 7: Sample gene nodes: $V_{\text{G}}^{(i)} \sim \text{Uniform}(V_{\text{G}}, k - |V_{\text{TF}}^{(i)}|)$
- 8: Duplicates $V_{\text{dup}} \leftarrow V_{\text{TF}}^{(i)} \cap V_{\text{G}}^{(i)}$, $n_{\text{dup}} \leftarrow |V_{\text{dup}}|$
- 9: **while** $n_{\text{dup}} \neq 0$ **do** ▷ Resolve duplicates
- 10: Resample a new gene $g \sim \text{Uniform}(V_{\text{G}})$
- 11: **if** $g \notin V_{\text{TF}}^{(i)} \cup V_{\text{G}}^{(i)}$ **then**
- 12: Add g into the $V_{\text{G}}^{(i)}$, $n_{\text{dup}} \leftarrow n_{\text{dup}} - 1$
- 13: **end if**
- 14: **end while**
- 15: Construct induced subgraph: $G_i = G[V_{\text{TF}}^{(i)} \cup V_{\text{G}}^{(i)}]$
- 16: $\mathcal{S} \leftarrow \mathcal{S} \cup \{G_i\}$
- 17: **end for**
- 18: **return** $\mathcal{S}$

C.2 Reconstruction

Algorithm 3 TF-ALL Subgraph Sampling (TASS): Reconstruction

Require: Global graph $G = (V, E)$, sampled node sets $\mathcal{V} = \{V_i\}_{i=1}^m$, trained model Φ , placeholder λ
Ensure: Global edge score matrix $\mathbf{S} \in \mathbb{R}^{|V| \times |V|}$

- 1: Initialize accumulation matrix $\mathbf{A} \leftarrow \mathbf{0}$, count matrix $\mathbf{C} \leftarrow \mathbf{0}$
- 2: **for** each sampled node set $V_i \in \mathcal{V}$ **do**
- 3: Construct induced subgraph $G_i = G[V_i]$
- 4: Predict raw edge logits: $\mathbf{E}_i = \Phi(G_i)$
- 5: Normalize edge scores: $\mathbf{P}_i = \text{Softmax}(\mathbf{E}_i)$ ▷ within-subgraph normalization
- 6: Map local edges to global index space: $(U, V) \leftarrow \pi(V_i \times V_i)$
- 7: Accumulate predictions: $\mathbf{A}_{U,V} \leftarrow \mathbf{A}_{U,V} + \mathbf{P}_i$, $\mathbf{C}_{U,V} \leftarrow \mathbf{C}_{U,V} + 1$
- 8: **end for**
- 9: **Consensus aggregation:**
- 10: **for** each potential edge $(u, v) \in V \times V$ **do**
- 11: **if** $\mathbf{C}_{u,v} > 0$ **then**
- 12: $\mathbf{S}_{u,v} = \mathbf{A}_{u,v} / \mathbf{C}_{u,v}$ ▷ mean across subgraphs
- 13: **else**
- 14: $\mathbf{S}_{u,v} = \lambda$ ▷ unobserved edge prior
- 15: **end if**
- 16: **end for**
- 17: **return** $\mathbf{S}$

D Dataset Information of BEELINE-KGC

The original BEELINE benchmark is constructed from five experimental single-cell RNA-seq datasets covering seven cell types (hESC, hHEP, mDC, mESC, mHSC-E, mHSC-GM, mHSC-L) across human and mouse systems [14]. These datasets span diverse biological contexts, including differentiation and developmental processes, and provide measurements that enable pseudotime inference.

Each dataset is paired with up to four types of ground-truth networks:: cell-type-specific (Specific) network derived from ChIP-seq experiments and curated databases such as ENCODE [42], ChIP-Atlas [43], and ESCAPE [44]; non-specific (Non-Specific) network aggregated from large-scale resources such as DoRothEA [45], RegNetwork [46], and TRRUST [47]; functional interaction networks from STRING [48], capturing broader biological associations that may include indirect regulatory relationships; loss-/gain-of-function (LOF/GOF) evidence from ESCAPE [44] (available only for mESC). For each dataset, two gene subsets are constructed by selecting the top 500 or 1,000 most variably expressed genes with transcription factors (*Abbr.*: TFs+500 and TFs+1000, respectively), culminating in total 44 distinct evaluation tasks.

Among these settings, future work may place particular emphasis on the Specific networks, as their cell-type-matched regulatory evidence provides the closest alignment between the expression measurements and the evaluation targets, making them a more representative testbed for context-dependent GRN discovery. The other network types remain valuable as complementary settings for assessing robustness to broader, heterogeneous, or functional interaction evidence.