LogicFusion: Differentiable Logical Rule Learning for Cancer Driver Gene Identification

Bang Chen[1] , **Lijun Guo**[1*] , **Wentao He**[1] , **Guang Cao**[2] and **Rong Zhang**[1]
[1]Ningbo University
[2]The University of Sydney
{2401100064, guolijun, hewentao}@nbu.edu.cn, gcao0005@uni.sydney.edu.au, zhangrong@nbu.edu.cn

Abstract

Identifying cancer driver genes (CDGs) requires integrating heterogeneous biological networks, each encoding distinct mechanistic insights into tumorigenesis. Existing multi-network methods typically fuse views at the feature level or enforce uniform representations, which obscures network-specific signals and limits interpretability. To address this, we propose LogicFusion, a novel differentiable framework that treats each biological network as an independent probabilistic evidence source and explicitly learns how to combine them using logical reasoning. Specifically, LogicFusion maps the output of each view into probabilistic space as independent evidence, and learns a set of logical rules in a differentiable manner. These rules adaptively select and fuse these view evidences through logical conjunction (AND) and disjunction (OR). The final prediction is then obtained via gene-specific weighting. LogicFusion enables both high-accuracy CDG identification and interpretable attribution of predictions to specific networks. Experiments on pan-cancer and cancer-specific datasets show that LogicFusion consistently outperforms state-of-the-art methods, while providing transparency in multi-view genomic reasoning through learned logical rules.

1 Introduction

Cancer is a highly complex genomic disease with pronounced heterogeneity, arising from the accumulation of diverse genomic alterations. Genes that play a causal role in cancer development are known as cancer driver genes (CDGs). [Hou *et al.*, 2024]. Accurate identification of CDGs is critical for understanding cancer mechanisms and facilitating precision therapies [Jang *et al.*, 2019]. With the rapid growth of large-scale genomic data and biological network resources, computational approaches that leverage molecular interaction information have advanced CDG identification with effectiveness and efficiency.

Recent studies have shown that integrating multiple biological networks is essential to improve the accuracy of CDG identification [Chen *et al.*, 2025]. By integrating protein–protein interaction networks from diverse sources [Han *et al.*, 2023], pathway co-occurrence and semantic similarity networks [Peng *et al.*, 2024], or gene co-expression and sequence similarity networks [Zhao *et al.*, 2022], the multi-network approaches aim to construct more comprehensive CDG identification models from varied perspectives [Chatzianastasis *et al.*, 2023]. However, current multi-network methods still face two critical limitations. First, most approaches perform integration at the feature level and rely on representation alignment or consistency constraints. These designs implicitly assume that gene relationships across biological networks can be homogenously aligned into a single latent space for integration [Han *et al.*, 2023]. This assumption may smooth out network-specific signals when conflicting evidence exists across networks, and it becomes difficult to clearly reveal the contribution mechanisms of each network to the final prediction. Second, biological network resources are rapidly expanding and exhibit substantial heterogeneity, incompleteness, and varying reliability. As a result, indiscriminately integrating all available views may introduce redundancy or noise, leading to suboptimal fusion results [Liu *et al.*, 2023]. To date, how to automatically identify the most informative subsets from an expanding collection of biological networks, and how to integrate heterogeneous information in a flexible and transparent way, remains a key open challenge in biomarker discovery and computational biology [Liang *et al.*, 2024].

In this paper, we propose LogicFusion, a differentiable rule learning framework for CDG identification that casts multi-network integration as a decision-level probabilistic logical reasoning problem. Unlike existing methods that rely on feature-level alignment and fusion, LogicFusion maps each biological network's output to a Beta distribution as an independent evidence perspective. Then, we introduce two closed-probability operations (conjunction and disjunction) to flexibly combine evidence from different networks into logical rules. Intuitively, the conjunction (AND) operator captures cases where multiple networks jointly support the same hypothesis, while the disjunction (OR) operator captures complementary evidence, allowing a hypothesis to be supported even when strong evidence is provided by only a

*Corresponding authors.

subset of networks. In addition, LogicFusion incorporates a differentiable rule generator that automatically constructs the most informative fusion rules for each gene in a data-driven manner. This design enables flexible, adaptive, and interpretable fusion of heterogeneous biological networks at the gene level. Our contributions are summarized as follows:

- We propose LogicFusion, a decision-level probabilistic logic fusion framework that moves beyond feature alignment fusion, allowing multi-network fusion while preserving network-specific heterogeneity.
- We introduce closed-form probabilistic logical operators (AND/OR) grounded in Beta distributions, enabling transparent and differentiable fusion of heterogeneous biological evidence through logical rules.
- We design a differentiable logical rule generator that automatically selects views and builds logical rules in a data-driven manner, avoiding combinatorial explosion while enabling gene-specific adaptive multi-view fusion.
- Extensive experiments on Pan-Cancer and cancer-specific datasets demonstrate that LogicFusion achieves consistent performance improvements over state-of-the-art methods and can provide interpretable gene-specific fusion patterns.

2 Related Work

2.1 Cancer Driver Gene Identification

The computational methods for CDG identification have evolved from traditional statistical analysis to intelligent computation based on complex networks. Early approaches primarily focused on identifying driver genes by assessing statistical significance based on mutation frequency [Dees *et al.*, 2012], but struggled to detect low-frequency driver genes. Subsequently, several studies employed machine learning algorithms to integrate multi-omics features within identification models [Davoli *et al.*, 2013; Tokheim *et al.*, 2016], yet these approaches relied on manual feature engineering and failed to incorporate correlation information between genes.

Graph neural networks (GNNs) have recently gained prominence in CDG identification due to their ability to jointly integrate both node attributes and network structure. Early studies such as EMOGI [Schulte-Sasse *et al.*, 2021] and MTGCN [Peng *et al.*, 2022] utilized graph convolutional architectures to integrate biological networks with multi-omics features, thereby capturing higher-order functional dependencies among genes. Subsequent works have sought to refine GNN-based CDG prediction through architectural enhancements and advanced learning paradigms. For instance, ECD-CDGI [Wang *et al.*, 2024] introduced an energy-constrained diffusion mechanism to suppress noisy propagation, while deepCDG [Wu *et al.*, 2025] improves gene representations by integrating multi-omics features through an attention-based GCN architecture. However, by relying on a single type of biological network, these approaches inherently fail to capture the multi-view and heterogeneous nature of cancer-related molecular mechanisms [Chatzianastasis *et al.*, 2023; Sun *et al.*, 2025].

Recent studies have explored multi-view graph learning. For example, MODIG [Zhao *et al.*, 2022] aggregated representations across multi-dimensional gene networks, while MPIT [Han *et al.*, 2023] integrated multiple PPI networks via feature alignment. More advanced methods like DIS-Fusion [Deng *et al.*, 2025] captured shared information between PPI networks and gene functional association through a cross-view decorrelation loss, and MNGCL [Peng *et al.*, 2024] employed contrastive learning to encourage consistency across network views. However, these methods mainly perform representation-level fusion. Aligning or forcing consistency across diverse networks often smooths out unique, network-specific signals. Furthermore, such implicit fusion also make it difficult to determine the specific contribution of each network view to the final prediction.

2.2 Probabilistic Logic

Probability distributions are often used to represent predictive outputs and latent quantities in learning models, providing a richer description than point estimates [Bishop and Quazaz, 1996]. Probabilistic logic formulations relax standard logical operators into differentiable forms, enabling neural networks to perform logical operations over continuous representations [Rocktäschel and Riedel, 2017]. Representative approaches include geometric query embedding methods (e.g., Query2Box[Ren *et al.*, 2020]) and probabilistic distribution–based models such as BetaE[Ren and Leskovec, 2020]. In particular, BetaE represented entities and queries using Beta distributions and defines neural logical operators in the embedding space, enabling compositional reasoning with conjunction (AND), disjunction (OR), and negation (NOT).

By embedding logical structure directly into learning objectives, these approaches provide an explicit, compositional mechanism for fusing multiple signals under semantic constraints, enabling structured and expressive reasoning over complex factor combinations that goes beyond what standard numerical aggregation can achieve[Maene and Tsamoura, 2025] . While such approaches have shown promise in domains requiring explicit symbolic reasoning, such as knowledge graph query answering[Ren and Leskovec, 2020] and recommender systems[Wu *et al.*, 2024], their potential for multi-view or multi-network fusion in biological contexts remains largely unexplored. To bridge this gap, we propose LogicFusion in this paper, a novel probabilistic logic-based framework for CDG identification that enables robust fusion of multi-view biological networks through decision-level logical reasoning.

3 Problem Statement

Let $\mathcal{G} = \{G_1, G_2, \ldots, G_N\}$ denote a set of N biological networks, where each graph $G_n = (V, E_n, X)$ shares the same node set $V = \{v_1, \ldots, v_{|V|}\}$ and feature matrix $X \in \mathbb{R}^{|V| \times m}$, but differs in its edge set $E_i \subseteq V \times V$, reflecting the heterogeneity across biological conditions. Each node represents a gene, and each row of X encodes an m-dimensional multi-omics feature vector $\mathbf{x}_i \in \mathbb{R}^m$.

Given the multi-network input $\mathcal{G}$, CDG identification is formulated as a node-level binary classification task. The goal is

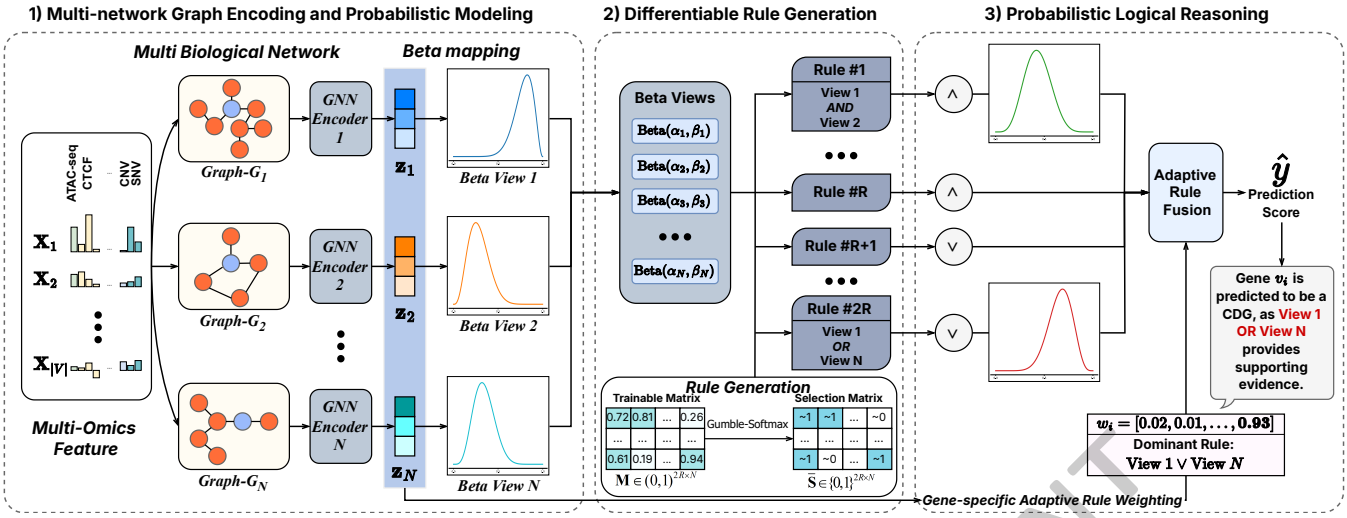


Figure 1: The framework of LogicFusion: The model extracts gene representations from each biological network and parameterizes them as beta distributions. A differentiable rule generator then combines views into distinct logical rules, which are fused via probabilistic logical operations AND (∧) and OR (∨). Finally, predictions are obtained in a gene-specific manner.

to learn a function $f : \mathcal{G} \to \{0,1\}^{|V|}$ that predicts whether each gene $v_i \in V$ is a CDG.

4 Methodology

The LogicFusion framework (Fig. 1) consists of three main components: 1) multi-network graph encoding and Probabilistic Modeling; 2) differentiable rule generation; and 3) probabilistic logical reasoning. Together, these components enable differentiable multi-network integration by learning gene-specific logical expressions for decision-level fusion.

4.1 Multi-network Graph Encoding and Probabilistic Modeling

Multi-network Graph Encoding

We employ GNN encoders to learn view-specific gene representations from N biological networks. To cope with the sparsity and noise commonly observed in biological networks, we adopt the ARMA graph convolutional operator [Bianchi *et al.*, 2021] to extract topology-aware and robust node representations for each network view.

Given a biological network of the n-th view $G_n = (V, E_n, \mathbf{X})$, ARMA convolution computes node embeddings by recursively filtering node features with skip connections. The node features are updated as:

$$\mathbf{X}^{(t+1)} = \sigma\left(\hat{\mathbf{L}}_n \mathbf{X}^{(t)} \mathbf{W} + \mathbf{X}^{(0)} \mathbf{W}_{skip}\right), \qquad (1)$$

with $\mathbf{X}^{(0)} = \mathbf{X}$, where $\sigma(\cdot)$ denotes a nonlinear activation function, $\mathbf{W}$ and $\mathbf{W}_{skip}$ are learnable weight matrices, and $\hat{\mathbf{L}}_n = \mathbf{D}_n^{-1/2} \mathbf{A}_n \mathbf{D}_n^{-1/2}$ is the normalized graph Laplacian constructed from the adjacency matrix $\mathbf{A}_n$ and degree matrix $\mathbf{D}_n$, both induced by the edge set E_n.

Applying the ARMA encoder to all N networks yields a set of view-specific node embeddings $\{\mathbf{z}_1, \ldots, \mathbf{z}_N\}$ for each gene, which serve as inputs for subsequent probabilistic modeling and rule-based reasoning.

View-wise Probabilistic Modeling

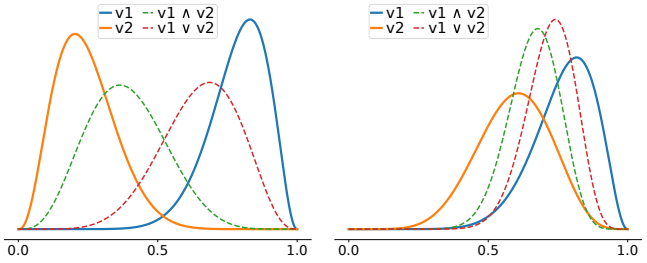
Given the view-specific node representation $\mathbf{z}_n$ from the n-th biological network, we associate each view with a Beta distribution. Following prior work such as BetaE [Ren and Leskovec, 2020], the Beta distribution serves as a bounded probabilistic embedding space for modeling and composing logical expressions. Formally, the Beta distribution is a continuous probability distribution defined on $[0, 1]$, making it suitable for modeling the probability if a gene is a cancer driver. Its probability density function $p(s \mid \alpha, \beta) = \frac{1}{B(\alpha,\beta)} s^{\alpha-1}(1-s)^{\beta-1}$, where $s \in [0, 1]$, $\alpha, \beta > 0$ are shape parameters, and $B(\alpha, \beta)$ denotes the beta function.

For each gene $v_i \in V$ and each view $n \in \{1, \ldots, N\}$, we employ a lightweight network consisting of a fully connected layer followed by a *Softplus* activation to produce the shape parameters $(\alpha_{i,n}, \beta_{i,n})$, defining a view-wise distribution $\mathrm{Beta}(\alpha_{i,n}, \beta_{i,n})$. By parameterizing a distinct Beta distribution for each network view, we capture view-specific heterogeneity and establish a foundation for rule generation and probabilistic logical operations.

4.2 Differentiable Rule Generation

After obtaining the probabilistic representations of each network view, we further construct a set of learnable logical rules based on these views and perform probabilistic logic fusion across multiple views through conjunction (AND) and disjunction (OR) operators. Each rule corresponds to a subset of selected views, capturing whether multiple views must jointly support a gene or whether alternative views can independently contribute. However, a key challenge is to select the optimal subset of views for each rule, which involves non-differentiable discrete decisions. To this end, we introduce a differentiable rule generator by adopting a Gumbel-Softmax relaxation [Jang *et al.*, 2017].

Specifically, we introduce a binary rule-selection matrix $\mathbf{S} \in \{0,1\}^{2R \times N}$, where $2R$ is the total number of rules (R



(a) conflicting views (b) consistent views.

Figure 2: Illustration of Beta-distributed views and the resulting distributions after applying probabilistic conjunction (AND, $\wedge$) and disjunction (OR, $\vee$).

conjunctive and R disjunctive). The entry $S_{ij} = 1$ indicates that the j-th view is selected into the i-th rule and $S_{ij} = 0$ otherwise. We parameterize this discrete selection matrix $\mathbf{S}$ with a trainable matrix $\mathbf{M} \in (0,1)^{2R\times N}$. For each M_{ij}, we apply Gumbel-Softmax to the pair $(M_{ij}, 1-M_{ij})$ and obtain a differentiable approximation:

$$[\overline{S}_{ij}, 1-\overline{S}_{ij}] = \mathrm{GumbelSoftmax}([M_{ij}, 1-M_{ij}]), \quad (2)$$

where $\overline{\mathbf{S}}$ denotes the relaxed rule-selection matrix used during training. This relaxation enables differentiable approximation of discrete logical rule selection and allows the rule structure to be learned end-to-end. The Gumbel-Softmax samples with class probabilities $\{\pi_i\}_{i=1}^k$ and Gumbel noise $\{g_i\}_{i=1}^k \sim \mathrm{Gumbel}(0,1)$ are computed as:

$$y_i = \frac{\exp((\log \pi_i + g_i)/\tau)}{\sum_{j=1}^k \exp((\log \pi_j + g_j)/\tau)}, \quad (3)$$

where $\tau > 0$ is a temperature parameter controlling the smoothness of the relaxation. As τ decreases, the samples approach one-hot vectors while remaining differentiable. This mechanism allows the model to automatically search the combinatorial space and identify informative view subsets for each logical rule.

4.3 Probabilistic Logical Reasoning

Based on the differentiable rule selection learned in the previous section, we perform probabilistic logical reasoning directly in the Beta embedding space to fuse multiple views according to the generated logical rules. Inspired by BetaE [Ren and Leskovec, 2020], we leverage the closure properties of Beta distributions to define logical operators in a bounded probabilistic space.

We utilize closed-form conjunction (AND) and disjunction (OR) operators to combine evidence. As illustrated in Fig. 2(a), under conflicting views, conjunction produces a conservative belief biased toward the lower-support view, whereas disjunction favors the higher-support view. Comparing Fig. 2(a) and (b) further shows that the proposed operators preserve uncertainty under disagreement while yielding more concentrated beliefs when views are consistent, enabling confident fusion under agreement.

Conjunction Operator (AND)

The conjunction operator models cases where all views jointly support the prediction (e.g., "View A AND View B both indicate this is a driver). Given a set of selected views $\mathcal{V} = \{v_1, \ldots, v_m\}$, we model each logical atom v_i as a Beta distribution $v_i = (\alpha_i, \beta_i)$ and aggregate conjunction belief using a weighted product:

$$\mu_{\mathrm{and}} = \exp\left(\sum_{i=1}^m w_i \log \mu_i\right) = \prod_{i=1}^m \mu_i^{w_i}, \quad (4)$$

where $\mu_i = \alpha_i/(\alpha_i+\beta_i)$ is the expectation of the i-th view. The learnable importance weights w_i are obtained via an attention mechanism conditioned on the Beta parameters:

$$w_i = \frac{\exp(\tau\, \phi([\alpha_i \| \beta_i]))}{\sum_{j=1}^m \exp(\tau\, \phi([\alpha_j \| \beta_j]))}, \quad (5)$$

where $\phi(\cdot)$ denotes a learnable linear mapping with softplus activation and τ is a temperature parameter.

To map the aggregated belief back to the Beta space, we perform moment matching with a variance that reflects both individual uncertainty and cross-view disagreement. The baseline variance is approximated as:

$$\sigma^2_{\mathrm{and}} = \sum_{i=1}^m w_i^2 \sigma_i^2, \quad (6)$$

where $\sigma_i^2 = \alpha_i\beta_i/[(\alpha_i+\beta_i)^2(\alpha_i+\beta_i+1)]$. We further calibrate the variance using the between-view variance $V_{\mathrm{between}} = \sum_i w_i(\mu_i - \bar{\mu})^2$, where $\bar{\mu} = \sum_i w_i\mu_i$, yielding:

$$\tilde{\sigma}^2_{\mathrm{and}} = \sigma^2_{\mathrm{and}} + \lambda \frac{V_{\mathrm{between}}}{V_{\mathrm{between}} + \sum_i w_i\sigma_i^2 + \epsilon} V_{\mathrm{between}}. \quad (7)$$

Finally, the output Beta parameters are obtained by moment matching:

$$\kappa = \frac{\mu_{\mathrm{and}}(1-\mu_{\mathrm{and}})}{\tilde{\sigma}^2_{\mathrm{and}}} - 1, \quad \alpha = \mu_{\mathrm{and}}\kappa,\ \beta = (1-\mu_{\mathrm{and}})\kappa. \quad (8)$$

Disjunction Operator (OR)

The disjunction operator models scenarios where support from any single view may be sufficient. Given the same set of selected views $\mathcal{V} = \{v_i = (\alpha_i, \beta_i)\}_{i=1}^m$, we aggregate belief using a weighted noisy-OR formulation:

$$\mu_{\mathrm{or}} = 1 - \prod_{i=1}^m (1-\mu_i)^{w_i}, \quad (9)$$

Similar to the conjunction operator, we calculate the aggregated variance and conflict-aware calibration to map the result back to Beta parameters $(\alpha_{or}, \beta_{or})$ using moment matching. This ensures the output remains closure in the bounded Beta embedding space.

4.4 Rule Aggregation and Model Training

The differentiable rule selection and probabilistic logical operators produce a set of rule-level outputs, where each rule yields a Beta distribution representing its belief about whether

a gene is a cancer driver. For the r-th rule, we obtain a prediction score from the expectation of its Beta distribution:

$$\hat{y}_r = \frac{\alpha_r}{\alpha_r + \beta_r}. \tag{10}$$

To capture gene-specific reliance on different rules, we introduce learnable aggregation weights. Given a gene representation $\mathbf{h}$ obtained by concatenating the view-specific embeddings $\{\mathbf{z}^{(1)}, \ldots, \mathbf{z}^{(N)}\}$, a lightweight neural network produces normalized rule weights:

$$\mathbf{w} = [w_1, \ldots, w_{2R}] = \mathrm{softmax}(\mathrm{MLP}(\mathbf{h})), \tag{11}$$

where w_r denotes the aggregation weight of the r-th rule. The final prediction is computed as a weighted sum of rule-level scores:

$$\hat{y} = \sum_{r=1}^{2R} w_r \hat{y}_r. \tag{12}$$

The model is trained end-to-end using a binary classification objective. We apply a binary cross-entropy loss on the final prediction:

$$\mathcal{L}_{\mathrm{bce}} = -y \log \hat{y} - (1-y)\log(1-\hat{y}), \tag{13}$$

which directly supervises both the differentiable rule generation process and the final rule aggregation.

In addition, the same binary cross-entropy loss is applied to the prediction produced by each logical rule. This auxiliary supervision encourages every learned rule to be individually predictive and helps stabilize end-to-end training. The overall training objective is defined as:

$$\mathcal{L} = \mathcal{L}_{\mathrm{bce}}^{\mathrm{final}} + \sum_{r=1}^{2R} \mathcal{L}_{\mathrm{bce}}^{(r)} + \lambda_1 \mathcal{L}_{\mathrm{ent}} + \lambda_2 \mathcal{L}_{\Theta}, \tag{14}$$

where $\mathcal{L}_{\mathrm{bce}}^{\mathrm{final}}$ denotes the binary cross-entropy loss applied to the final aggregated prediction, and $\mathcal{L}_{\mathrm{bce}}^{(r)}$ is the same loss applied to the prediction produced by the r-th logical rule. $\mathcal{L}_{\mathrm{ent}}$ is an entropy-based constraint on the rule aggregation weights, defined as:

$$\mathcal{L}_{\mathrm{ent}} = -\mathbb{E}_i \left[\sum_{r=1}^{2R} w_{i,r} \log w_{i,r} \right], \tag{15}$$

where $\mathbb{E}_i[\cdot]$ denotes the average over genes in a training batch, and $w_{i,r}$ denotes the aggregation weight of the r-th rule for the i-th gene. $\mathcal{L}_{\Theta} = \sum_j \|\Theta_j\|_F^2$ is a standard parameter regularization term to prevent overfitting. The coefficients λ_1 and λ_2 balance the contributions of different loss components.

5 Experiments

In this section, we conduct experiments to answer the following questions: **Q1**: How does LogicFusion perform compared with other baselines? **Q2**: How does LogicFusion perform relative to other multi-network methods under larger view settings? **Q3**: How do the key components of LogicFusion contribute to its performance? **Q4**: How are logical rules learned and utilized by LogicFusion during training? **Q5**: How sensitive is LogicFusion to the number of candidate logical rules?

5.1 Experimental Setup

Dataset Preprocessing

We evaluate our method on pan-cancer dataset and two cancer-specific datasets, all of which contain multi-omics gene features and multiple biological networks. The pan-cancer dataset follows the construction of EMOGI [Schulte-Sasse *et al.*, 2021], while the MCF7 (breast cancer) and A549 (lung cancer) datasets are adopted from MPIT [Han *et al.*, 2023]. Gene labels are collected following prior studies. As a result, the Pan-Cancer, MCF7, and A549 datasets contain 870/2187, 379/1581, and 425/2557 positive/negative genes, respectively.

For each dataset, we construct ten biological networks to provide complementary relational information. These include CPDB [Kamburov and Herwig, 2022], STRING [Szklarczyk *et al.*, 2023], six networks from NDEx [Pratt *et al.*, 2015] (PCNet, iRefIndex, Multinet, HumanNet, ProteomeHD, and BioGRID), as well as two functional networks based on Gene Ontology semantic similarity and pathway co-occurrence [Peng *et al.*, 2024]. Together, these networks capture diverse interaction, functional, and pathway-level relationships. Genes are randomly split into training, validation, and test sets with a ratio of 6:2:2 for all datasets.

Baseline Methods

We compare the proposed method with representative baselines covering three categories: (i) classical graph neural networks, including GCN [Kipf and Welling, 2017] and GTN [Yun *et al.*, 2019]; (ii) single-network CDG identification methods, such as EMOGI [Schulte-Sasse *et al.*, 2021], MTGCN [Peng *et al.*, 2022], ECD-CDGI [Wang *et al.*, 2024] and deepCDG [Wu *et al.*, 2025]; and (iii) multi-network integration methods, including MODIG [Zhao *et al.*, 2022], MNGCL [Peng *et al.*, 2024], and MPIT [Han *et al.*, 2023]. All baselines are evaluated under the settings reported in their original implementations.

Implementation Details

All models are trained for 300 epochs using the Adam optimizer with a batch size of 128 and a learning rate of 1×10^{-3}. Early stopping is applied based on validation performance. For our method, the number of learned logical rules is set to R=6. The hyperparameters are set to, $\lambda_1 = \lambda_2 = 1 \times 10^{-5}$. GCN and GTN are implemented using the torch_geometric library. For the remaining baseline methods, we follow the parameter settings reported in their original implementations. All experiments are repeated with three random seeds, and the average results are reported.

5.2 Overall Performance(Q1, Q2)

Table 1 demonstrates the performance of various methods across three datasets. As we can see, LogicFusion achieves better performance compared to other models, and its advantage becomes more pronounced on datasets with a larger number of biological network views.

For single-network methods, approaches that optimize the model architecture (such as MTGCN, ECD-CDGI, and deepCDG) generally outperform direct application of GNN methods. This indicates that targeted architectural modifications have a positive effect on CDG identification. When

	Methods	Pan Cancer			Breast Cancer			Lung Cancer		
		F1	AUC	AUPRC	F1	AUC	AUPRC	F1	AUC	AUPRC
Single	GCN [Kipf and Welling, 2017]	0.7061	0.9094	0.8140	0.7768	0.9436	0.8494	0.6725	0.9163	0.7468
	GTN [Yun *et al.*, 2019]	0.7354	0.9162	0.8412	0.7907	0.9627	0.8888	0.7090	0.9381	0.7784
	EMOGI [Schulte-Sasse *et al.*, 2021]	0.7187	0.9091	0.8200	0.8230	0.9672	0.8798	0.7399	0.9347	0.7296
	MTGCN [Peng *et al.*, 2022]	0.7297	0.9221	0.8576	0.7897	0.9590	0.8828	0.7143	0.9349	0.7981
	ECD-CDGI [Wang *et al.*, 2024]	0.7357	0.9234	0.8584	0.8547	0.9786	0.9406	0.7718	0.9541	0.8552
	deepCDG [Wu *et al.*, 2025]	0.7463	0.9255	0.8584	0.8661	0.9734	0.9359	0.7953	0.9672	0.8859
5graph	MODIG [Zhao *et al.*, 2022]	0.7481	0.9204	0.8545	0.7636	0.9435	0.8237	0.7412	0.9441	0.7872
	MPIT [Han *et al.*, 2023]	0.7584	0.9134	0.8390	0.8681	0.9720	0.9442	0.7448	0.9419	0.8205
	MNGCL [Peng *et al.*, 2024]	0.7544	0.9189	0.8611	0.8381	0.9658	0.9059	0.7503	0.9446	0.8536
	LogicFusion(Ours)	0.7565	0.9250	0.8511	0.8812	0.9839	0.9600	0.8204	0.9586	0.8914
10graph	MODIG [Zhao *et al.*, 2022]	0.7496	0.9151	0.8373	0.7557	0.9516	0.8218	0.7582	0.9591	0.8302
	MPIT [Han *et al.*, 2023]	0.7566	0.9108	0.8380	0.8497	0.9774	0.9356	0.7927	0.9655	0.8789
	MNGCL [Peng *et al.*, 2024]	0.7315	0.9031	0.8244	0.8230	0.9692	0.9182	0.7851	0.9698	0.8980
	LogicFusion(Ours)	**0.7682**	**0.9349**	**0.8692**	**0.8894**	**0.9844**	**0.9615**	**0.8571**	**0.9772**	**0.9292**

Table 1: Performance comparison on three datasets (F1, AUC, AUPRC). Results for multi-network methods are reported using the first five networks (5graph) and all ten networks (10graph). Bold and underlined numbers denote the best and best baseline results, respectively.

leveraging multiple biological networks, most multi-network approaches achieve better performance than single-network models by utilizing complementary information across different views. However, this improvement does not necessarily increase linearly with the number of networks. Notably, several existing multi-network methods failed to achieve sustained improvement when increasing the number of network views from 5 to 10. For instance, MODIG exhibited performance degradation across pan-cancer (F1, AUC), breast cancer (F1, AUPRC), and all lung cancer metrics. Similarly, MNGCL and MPIT failed to improve on some metrics as the number of views increased. These findings suggest that, as the number of views increases, feature-level fusion with representation alignment may tend to smooth out view-specific signals, while indiscriminately integrating all available networks further amplifies noise and redundancy, ultimately leading to performance degradation.

In contrast, LogicFusion consistently benefits from an increasing number of available views. Across ten biological networks, LogicFusion improved performance on all three datasets, with particularly notable gains on the lung cancer dataset. This advantage stems from the fact that LogicFusion avoids feature-level alignment and retains view-specific signals by modeling each network as an independent probabilistic evidence source. Moreover, through differentiable logical rule learning, LogicFusion can automatically identify and filter informative subsets of views from the data for flexible fusion. Consequently, as the scale of network resources expands, LogicFusion maintains stable and efficient multi-view fusion capabilities.

5.3 Ablation Study(Q3)

We conducted three ablation studies to further evaluate the contributions of the differentiable rule selector and probabilistic logical operators. We designed three variants (w/o select, w/o disjunction, and w/o conjunction) for comparison with LogicFusion. When removing the differentiable rule selector, we assumed all views participated in logical fusion.

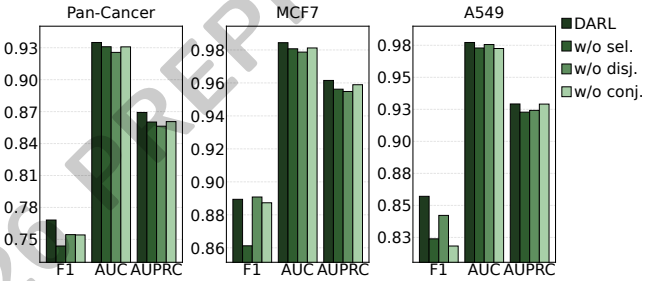


Figure 3: Ablation study of LogicFusion on three datasets.

When removing a logical operator, all rules used only another logical operator. As shown in Figure 3, removing the rule selector resulted in a significant decline in model performance, indicating that the adaptive subset of views learned from data is crucial for effective multi-view fusion. Disabling either logical operator also led to performance degradation, suggesting that relying on a single, static fusion operator struggles to cover diverse samples and likely compromises model flexibility. Overall, these results confirm that the combined use of differentiable rule selection and AND/OR operators is essential for achieving robust performance.

5.4 Interpretability of Learned Logical Rules(Q4)

To determine whether the model was genuinely learning to construct flexible logical rules, we analyzed its training process on the A549 dataset. Fig. 5(a) shows the temporal evolution of the average weight across all genes for the 12 rules. During the initial training phase, the weights of individual logical rules fluctuated significantly, reflecting the model's exploration process. After a certain number of exploration iterations, the weights of each rule rapidly stabilized. Ultimately, most genes depended on a dominant rule—in this case, Rule 10 exhibited the highest weight for the majority of genes. Simultaneously, a few rules (such as Rule 2 and Rule 6) retained significant weights for certain genes.

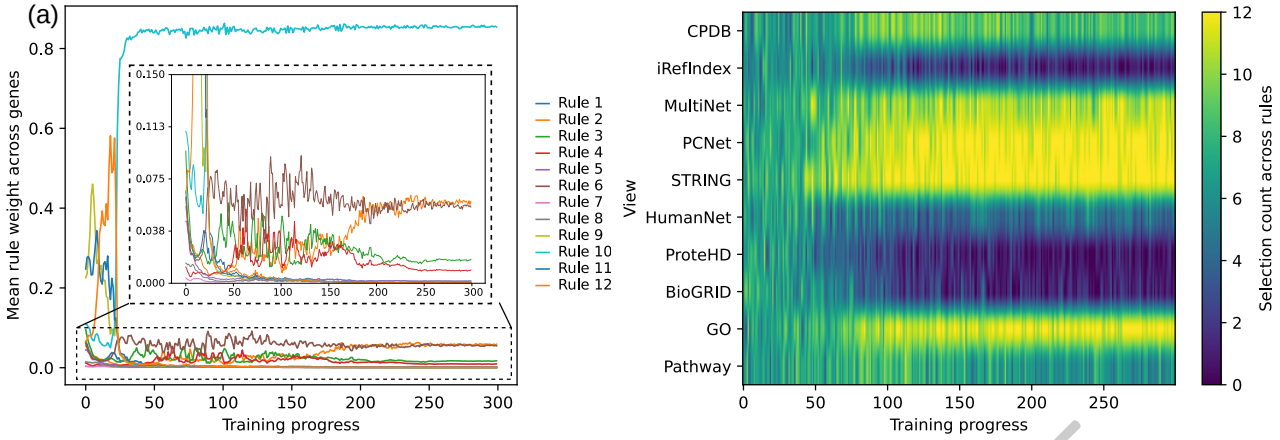


Figure 4: Training dynamics of logical rule weights and view selection. (a) Mean rule weight across genes during training. (b) View selection frequencies across rules over training.

Gene	Dominant logical rules	Rule Weight	Pred.
IGF1	PCNet ∧ STRING ∧ GO	0.92	0.94
SAAL1	CPDB ∨ MultiNet ∨ PCNet ∨ STRING ∨ GO	0.98	0.08

Table 2: Example learned logical rules for representative genes.

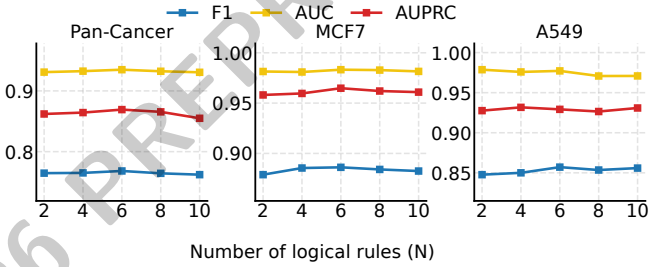


Figure 5: Hyper-parameter Sensitivity

This phenomenon demonstrates that the model can indeed autonomously learn to construct logical rules from data, capable of capturing both global patterns across most genes and accommodating inter-gene variability.

Fig. 4(b) further illustrates how the 12 rules select the 10 views during training. During the exploratory phase early in training, views were selected randomly. As training iterations increased, the frequency of view selection by the rules gradually diverged. Views such as PCNet, STRING, and GO were frequently chosen, while the usage frequency of views like iRefIndex and ProtHD decreased. This phenomenon further validates that the model can automatically screen out better views to form logical rules.

Furthermore, different rules represent distinct fusion patterns. Rule 10 in Fig. 4(a), which carries the highest weight, employs an OR structure. This structure improves robustness by allowing alternative sources of evidence. In contrast, the diversified rules 2 and 6 employ an AND structure, requiring multiple views to provide evidence simultaneously. Table 2 presents two gene examples from the A549 dataset, illustrating how different types of logical rules contribute to correct predictions. For example, IGF1 is predicted as a driver gene based on consistent evidence jointly supported by the PCNet, STRING, and GO networks.

5.5 Hyper-parameter Sensitivity(Q5)

We study the sensitivity of LogicFusion to the number of candidate logical rules R. Fig. 5 shows that LogicFusion achieves stable performance across a broad range of R values on all three datasets. Performance improves as R increases from small values and consistently peaks at $R = 6$, while further increasing R leads to marginal gains.

This trend reflects a balance between expressive capacity and optimization stability. Too few rules limit the ability to capture diverse view-combination patterns, whereas too many rules introduce redundancy and complicate rule selection and aggregation. In our setting, $R = 6$ provides sufficient expressive coverage for multi-view logical composition without unnecessary redundancy.

6 Conclusion

We proposed LogicFusion, a differentiable rule learning framework that formulates multi-network integration for CDG identification as an explicit, decision-level rule learning problem. By modeling each biological network as an independent probabilistic view and automatically learning logical rules composed of conjunction and disjunction operators, LogicFusion enables flexible and structured fusion of multiple networks in a closed Beta embedding space. Experiments on pan-cancer and cancer-specific datasets demonstrate that LogicFusion consistently outperforms state-of-the-art methods and exhibits robust performance across different settings. In addition, the learned rules provide a transparent view of how multiple networks are combined for individual predictions. Future work will explore more effective utilization of uncertainty information and extend the proposed framework to other multi-view learning scenarios.

Acknowledgments

This work was supported by the Guangxi Key Technologies R&D Program (No. FN2504240023), the Medical and Health Science Program of Zhejiang Province (No. WKJ-ZJ-26021), and Zhejiang Provincial Natural Science Foundation of China (No. LQN26F010018).

References

[Bianchi *et al.*, 2021] Filippo Maria Bianchi, Daniele Grattarola, Lorenzo Livi, and Cesare Alippi. Graph neural networks with convolutional arma filters. *IEEE transactions on pattern analysis and machine intelligence*, 44(7):3496–3507, 2021.

[Bishop and Quazaz, 1996] Christopher Bishop and Cazhaow Quazaz. Regression with input-dependent noise: A bayesian treatment. *Advances in neural information processing systems*, 9, 1996.

[Chatzianastasis *et al.*, 2023] Michail Chatzianastasis, Michalis Vazirgiannis, and Zijun Zhang. Explainable multilayer graph neural network for cancer gene prediction. *Bioinformatics*, 39(11):btad643, 2023.

[Chen *et al.*, 2025] Bang Chen, Lijun Guo, Houli Fan, Wentao He, and Rong Zhang. Soft-evidence fused graph neural network for cancer driver gene identification across multiview biological graphs. In *2025 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 827–832. IEEE, 2025.

[Davoli *et al.*, 2013] Teresa Davoli, Andrew Wei Xu, Kristen E Mengwasser, Laura M Sack, John C Yoon, Peter J Park, and Stephen J Elledge. Cumulative haploinsufficiency and triplosensitivity drive aneuploidy patterns and shape the cancer genome. *Cell*, 155(4):948–962, 2013.

[Dees *et al.*, 2012] Nathan D Dees, Qunyuan Zhang, Cyriac Kandoth, Michael C Wendl, William Schierding, Daniel C Koboldt, Thomas B Mooney, Matthew B Callaway, David Dooling, Elaine R Mardis, et al. Music: identifying mutational significance in cancer genomes. *Genome research*, 22(8):1589–1598, 2012.

[Deng *et al.*, 2025] Chao Deng, Hongdong Li, and Jianxin Wang. Improving cancer gene prediction by enhancing common information between the ppi network and gene functional association. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 137–145, 2025.

[Han *et al.*, 2023] Zebei Han, Gufeng Yu, and Yang Yang. Enhancing cancer gene prediction through aligned fusion of multiple ppi networks using graph transformer models. In *2023 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 542–547. IEEE, 2023.

[Hou *et al.*, 2024] Wenju Hou, Yan Wang, Ziqi Zhao, Yizhi Cong, Wei Pang, and Yuan Tian. Hierarchical graph neural network with subgraph perturbations for key gene cluster discovery in cancer staging. *Complex & Intelligent Systems*, 10(1):111–128, 2024.

[Jang *et al.*, 2017] Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. In *International Conference on Learning Representations*, 2017.

[Jang *et al.*, 2019] Hyo Sik Jang, Nakul M Shah, Alan Y Du, Zea Z Dailey, Erica C Pehrsson, Paula M Godoy, David Zhang, Daofeng Li, Xiaoyun Xing, Sungsu Kim, et al. Transposable elements drive widespread expression of oncogenes in human cancers. *Nature genetics*, 51(4):611–617, 2019.

[Kamburov and Herwig, 2022] Atanas Kamburov and Ralf Herwig. Consensuspathdb 2022: molecular interactions update as a resource for network biology. *Nucleic acids research*, 50(D1):D587–D595, 2022.

[Kipf and Welling, 2017] Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations*, 2017.

[Liang *et al.*, 2024] Jing Liang, Zhuo Hu, Ying Bi, Han Cheng, Kun jie Yu, Cai-Tong Yue, Xianfang Wang, and Wei-Feng Guo. A survey on evolutionary computation for identifying biomarkers of complex disease. *IEEE Transactions on Evolutionary Computation*, 2024.

[Liu *et al.*, 2023] Wei Liu, Yufei Chen, Xiaodong Yue, Changqing Zhang, and Shaorong Xie. Safe multi-view deep classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 8870–8878, 2023.

[Maene and Tsamoura, 2025] Jaron Maene and Efthymia Tsamoura. Embeddings as probabilistic equivalence in logic programs. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

[Peng *et al.*, 2022] Wei Peng, Qi Tang, Wei Dai, and Tielin Chen. Improving cancer driver gene identification using multi-task learning on graph convolutional network. *Briefings in bioinformatics*, 23(1):bbab432, 2022.

[Peng *et al.*, 2024] Wei Peng, Zhengnan Zhou, Wei Dai, Ning Yu, and Jianxin Wang. Multi-network graph contrastive learning for cancer driver gene identification. *IEEE Transactions on Network Science and Engineering*, 2024.

[Pratt *et al.*, 2015] Dexter Pratt, Jing Chen, David Welker, Ricardo Rivas, Rudolf Pillich, Vladimir Rynkov, Keiichiro Ono, Carol Miello, Lyndon Hicks, Sandor Szalma, et al. Ndex, the network data exchange. *Cell systems*, 1(4):302–305, 2015.

[Ren and Leskovec, 2020] Hongyu Ren and Jure Leskovec. Beta embeddings for multi-hop logical reasoning in knowledge graphs. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 19716–19726. Curran Associates, Inc., 2020.

[Ren *et al.*, 2020] Hongyu Ren, Weihua Hu, and Jure Leskovec. Query2box: Reasoning over knowledge graphs in vector space using box embeddings. In *International Conference on Learning Representations*, 2020.

[Rocktäschel and Riedel, 2017] Tim Rocktäschel and Sebastian Riedel. End-to-end differentiable proving. *Advances in neural information processing systems*, 30, 2017.

[Schulte-Sasse *et al.*, 2021] Roman Schulte-Sasse, Stefan Budach, Denes Hnisz, and Annalisa Marsico. Integration of multiomics data with graph convolutional networks to identify new cancer genes and their associated molecular mechanisms. *Nature Machine Intelligence*, 3(6):513–526, 2021.

[Sun *et al.*, 2025] Weicheng Sun, Ping Zhang, Jinsheng Xu, Weihan Zhang, Yongbin Zeng, and Li Li. Bridging attention fusion-based multi-view graph neural networks for spatial gene expression prediction. *Pattern Recognition*, page 112207, 2025.

[Szklarczyk *et al.*, 2023] 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.

[Tokheim *et al.*, 2016] Collin J Tokheim, Nickolas Papadopoulos, Kenneth W Kinzler, Bert Vogelstein, and Rachel Karchin. Evaluating the evaluation of cancer driver genes. *Proceedings of the National Academy of Sciences*, 113(50):14330–14335, 2016.

[Wang *et al.*, 2024] Tao Wang, Linlin Zhuo, Yifan Chen, Xiangzheng Fu, Xiangxiang Zeng, and Quan Zou. Ecd-cdgi: An efficient energy-constrained diffusion model for cancer driver gene identification. *PLOS Computational Biology*, 20(8):e1012400, 2024.

[Wu *et al.*, 2024] Maonian Wu, Bang Chen, Shaojun Zhu, Bo Zheng, Wei Peng, and Mingyi Zhang. Neuro-symbolic recommendation model based on logic query. *Knowledge-Based Systems*, 284:111311, 2024.

[Wu *et al.*, 2025] Yingzhuo Wu, Jialuo Xu, Junming Li, Jia Gu, Xuequn Shang, and Xingyi Li. Deep graph convolutional network-based multi-omics integration for cancer driver gene identification. *Briefings in Bioinformatics*, 26(4):bbaf364, 2025.

[Yun *et al.*, 2019] Seongjun Yun, Minbyul Jeong, Raehyun Kim, Jaewoo Kang, and Hyunwoo J Kim. Graph transformer networks. *Advances in neural information processing systems*, 32, 2019.

[Zhao *et al.*, 2022] Wenyi Zhao, Xun Gu, Shuqing Chen, Jian Wu, and Zhan Zhou. Modig: integrating multi-omics and multi-dimensional gene network for cancer driver gene identification based on graph attention network model. *Bioinformatics*, 38(21):4901–4907, 2022.