

EGRL: Edge generation-guided relation-aware learning for RNA-protein interaction prediction

Danyu Li^a, Ling Zhou^a, Rubing Huang^{a,b,*}, Xian Zhong^c, Bin Zou^d, Kui Jiang^e

^a*School of Computer Science and Engineering, Macau University of Science and Technology, Macao SAR, 999078, China*

^b*Macau University of Science and Technology Zhuhai Research Institute, Zhuhai, Guangdong, 519099, China*

^c*Hubei Key Laboratory of Transportation Internet of Things, School of Computer Science and Artificial Intelligence, Wuhan University of Technology, Wuhan, Hubei, 430070, China*

^d*School of Food and Biological Engineering, Jiangsu University, Zhenjiang, Jiangsu, 212013, China*

^e*School of Computer Science and Technology, Harbin Institute of Technology, Harbin, Heilongjiang, 150001, China*

Abstract

RNA-Protein Interactions (RPIs) are critical for regulating cellular functions. While traditional wet-lab experiments for RPI detection are costly and time-consuming, *Deep Learning* (DL) methods provide an efficient computational alternative for *RPI Prediction* (RPIP). In particular, *Graph Neural Networks* (GNNs) are promising, as they naturally model RPI networks. However, existing GNN-based methods often rely on homogeneous graphs or predefined meta-paths, which limit their ability to handle data sparsity and to generalize to cold-start scenarios involving unknown molecules. To address these limitations, we propose Edge Generation-guided Relation-aware Learning (EGRL), a novel framework with several key components: implicit meta-path learning to capture relational semantics without handcrafted paths; a multi-relation-aware attention mechanism for adaptive fusion of interaction patterns; a graph generator that predicts potential (“soft”) edges to support cold-start nodes; and a multi-feature fusion predictor for final interaction scoring. EGRL is jointly trained with a primary task loss and an auxiliary generator loss. Comprehensive evaluations on four benchmark datasets demonstrate that EGRL achieves competitive overall performance. More importantly, it exhibits superior

*Corresponding author.

Email addresses: 3230002471@student.must.edu.mo (Danyu Li), lzhou@must.edu.mo (Ling Zhou), rbhuang@must.edu.mo (Rubing Huang), zhongx@whut.edu.cn (Xian Zhong), binzou2009@ujls.edu.cn (Bin Zou), jiangkui@hit.edu.cn (Kui Jiang)

generalization in cold-start settings, achieving an *Area Under the Receiver Operating Characteristic curve* (AUROC) of 0.867 and an *Area Under the Precision-Recall curve* (AUPR) of 0.861 on unknown molecules, corresponding to improvements of 8.6% in AUROC and 5.0% in AUPR over prior state-of-the-art methods. The code will be released soon.

Keywords: RNA-Protein Interaction Prediction, Graph Neural Networks, Implicit Meta-path Learning, Multi-relational Graph Modeling, Cold-start Generalization

1. Introduction

The *RNA-Protein Interactions* (RPIs) play a vital role in various cellular processes, including gene regulation [1] and protein synthesis [2], with significant implications for disease prediction [3], functional annotation of biomolecules [4], and drug design [5, 6]. Traditional biological experiments for RPI detection are often expensive and time-consuming [7], thereby motivating the development of computational approaches.

Early computational methods for *RPI Prediction* (RPIP) mainly relied on traditional machine learning models, such as support vector machines and random forests, using hand-crafted features extracted from sequence or structure information [8, 9]. Although these methods achieved some success, their performance was limited by the quality of manual feature engineering and their inability to capture complex nonlinear interaction patterns. Subsequently, *Deep Learning* (DL) has gained increasing attention in RPIP due to its ability to automatically learn complex features and interaction patterns from biological data [10]. DL techniques, especially *Convolutional Neural Networks* (CNNs) and *Recurrent Neural Networks* (RNNs), have been widely adopted for RPIP. CNNs extract local motif patterns from sequences, while RNNs and their variants model sequential dependencies. However, these methods typically treat each RNA-protein pair independently and ignore the global relational structure among multiple RNAs and proteins [11, 12].

In recent years, *Graph Neural Networks* (GNNs) have emerged as a powerful framework for modeling RPIs. GNNs naturally represent biological entities as nodes and their interactions as edges in a graph, enabling the modeling of relational structure

and dependencies within complex biological systems. By recursively aggregating information from neighboring nodes, GNNs learn enriched node representations that encode both intrinsic features and topological context. This capability makes them well-suited for analyzing diverse biological networks, including RNA-protein interaction graphs [13], protein-protein interaction graphs [14, 15], and RNA-RNA interaction graphs [16, 17].

However, conventional GNN-based methods still face notable limitations in RPIP. Many approaches rely on homogeneous graph constructions [18, 19], which oversimplify the inherent heterogeneity of biological systems, where nodes (*e.g.*, RNAs and proteins) and interactions (*e.g.*, binding and functional association) are of distinct types. Although some studies incorporate predefined meta-paths to model higher-order relations, such designs require substantial domain expertise and may fail to capture all relevant interaction patterns [20, 21]. These limitations restrict model expressiveness, hinder generalization under sparse data, and limit performance in predicting interactions involving uncharacterized molecules (*i.e.*, the cold-start problem).

To address these challenges, we propose Edge Generation-guided Relation-aware Learning (EGRL), a novel heterogeneous GNN framework that integrates implicit meta-path modeling, a graph generator for cold-start nodes, and multi-relational *Graph Attention Network* (GAT) layers with multi-feature fusion. The main contributions are summarized as follows: (1) *Implicit meta-path learning*: We propose an implicit meta-path learner that automatically captures the importance of different relation types without manual design, dynamically integrating relation-level semantics into node representations. (2) *Graph generator for cold-start nodes*: We introduce a jointly trained graph generator that predicts soft edges for unseen RNAs/proteins based on their sequence features, enabling effective cold-start interaction prediction. (3) *Multi-relational GAT with multi-feature fusion*: We design a multi-relational GAT that processes each edge type independently, along with a predictor that combines concatenation, element-wise product, and absolute difference of node embeddings for enhanced interaction modeling. (4) *Competitive performance and effective cold-start generalization*: Extensive experiments on four benchmark datasets demonstrate that EGRL achieves competitive performance against state-of-the-art methods and exhibits strong generalization under

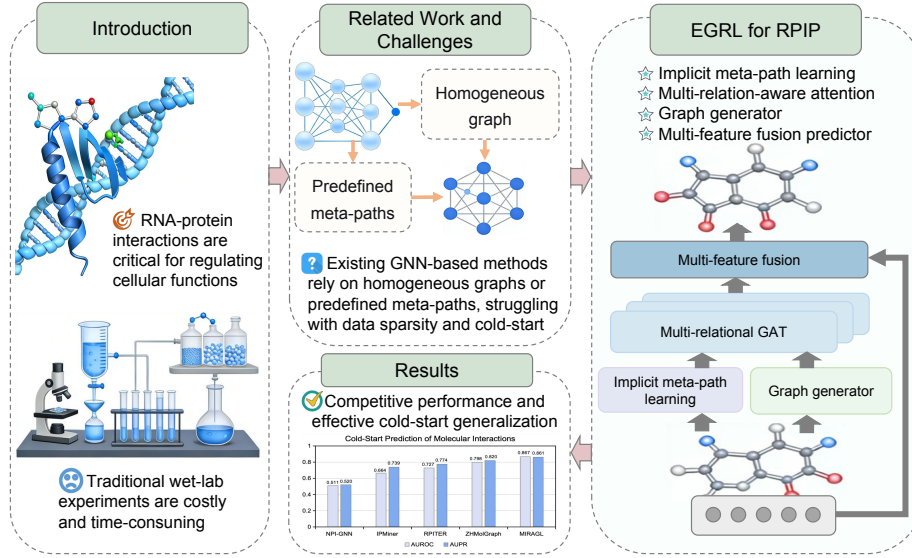


Figure 1: **Graphical abstract of EGRL.** Overview of the proposed framework and its key components for RNA-protein interaction prediction.

molecule hold-out and sequence-cluster-based cold-start settings.

The remainder of this paper is organized as follows. Section 2 reviews related work; Section 3 presents the research motivation; Section 4 describes the proposed EGRL framework; Section 5 details the experimental setup; Section 6 reports and analyzes the results; and Section 7 concludes the paper and discusses future directions.

Figure 1 presents the graphical overview. It highlights the importance and challenges of RPIP, emphasizing the limitations of traditional GNNs based on homogeneous graphs or predefined meta-paths under sparse and cold-start conditions. It further illustrates how EGRL addresses these challenges, achieving competitive overall performance while maintaining strong generalization in cold-start scenarios.

2. Related work

Traditional machine learning methods for RPIP typically rely on manually extracted features. For example, RPI-Pred [22] utilizes sequence and structural information with a support vector machine classifier. Despite their interpretability and low computational cost, these methods suffer from limited expressive power, as hand-crafted features may

fail to capture complex interaction patterns.

Deep learning approaches have significantly advanced RPIP by automatically learning hierarchical representations. CNNs extract local motif patterns from RNA or protein sequences treated as one-dimensional signals. For instance, MCNN [23] predicts RNA-protein binding sites by integrating multiple CNNs, each processing RNA sequence segments of different lengths. RNNs, particularly long short-term memory (LSTM) models, capture long-range dependencies. Some methods further combine CNNs and RNNs to leverage both local and sequential patterns [24, 12]. In addition, RPI-CapsuleGAN [25], a generative adversarial network-based model, employs adversarial training to learn data distributions and enhance feature representations for RPIP.

In recent years, *Graph Neural Networks* (GNNs) have become a mainstream approach for RPIP and have achieved remarkable performance. The GNN paradigm aligns naturally with RNA-protein interaction modeling, where RNAs and proteins are represented as nodes and interactions as edges. Consequently, RPIP can be formulated as a link prediction problem. GNNs learn node representations by aggregating neighborhood information, enabling effective prediction of potential interactions. Several GNN-based methods have been proposed for RPIP. LPICGAE [26] incorporates graph autoencoders to infer potential interactions. LncPNet [27] designs a network embedding approach for lncRNA-protein interaction prediction. RPI-GGCN [28] integrates gated graph convolutional networks with a co-regularized variational autoencoder, leveraging Gated Recurrent Units and *Graph Convolutional Networks* (GCNs) to extract topological information. DeepPN [18] employs parallel CNN and GCN layers to capture hidden features of RNA sequences for RPIP. LPI-KCGCN [13] utilizes a two-layer GCN to learn latent representations of RNAs and proteins and subsequently predicts their interaction scores.

However, many GNN-based methods treat the RPI graph as homogeneous and primarily focus on enhanced node feature learning, which oversimplifies the intrinsic heterogeneity of biological systems. In practice, RPI graphs are heterogeneous, involving multiple types of nodes and interactions. To address this, heterogeneous graph learning methods have been explored [29]. For example, BiHo-GNN [30] integrates homogeneous and heterogeneous network features through bipartite graph embedding, coupled with a mutual optimization strategy. Meta-paths have also been widely adopted

to capture higher-order semantic relations in heterogeneous graphs, such as in RNA-disease association prediction [31]. An explicit multilevel meta-path aggregation graph embedding model has been proposed for miRNA-disease association prediction [32]. These studies suggest that meta-path-based feature learning is a viable strategy for modeling heterogeneous graph structures.

Graph Attention Networks (GATs) have been further introduced to enhance representation learning in heterogeneous interaction graphs. GATs adaptively aggregate information from neighboring nodes by learning attention coefficients, and have been shown to outperform conventional sequence-based models such as CNNs and bidirectional LSTM networks [33]. NABind [34] integrates sequence and structural descriptors with an attention mechanism to aggregate edge information. RPIembeddor [35] employs a GAT-based framework to incorporate structural and functional features for RPI classification. Unlike self-attention in Transformers, which relies on positional encoding, graph attention computes importance based on node features and graph connectivity, making it more suitable for graph-structured data. In RPIP, modeling the importance of interaction patterns between nodes is more meaningful than relying on absolute positional information. Therefore, GAT provides a natural and effective framework for RPI modeling.

Based on the above observations, we propose EGRL. To the best of our knowledge, this is the first study that integrates implicit meta-path learning, a graph generator for cold-start nodes, and a multi-relational GAT with multi-feature fusion for RPIP.

3. Motivation

RPIs constitute core regulatory mechanisms of cellular activities and are closely associated with the occurrence and progression of various diseases. Accurate RPI prediction is therefore of great significance for drug target discovery, disease diagnosis, and therapeutic development. In recent years, deep learning models have substantially improved the efficiency of RPIP; however, several key challenges remain:

- *Insufficient training data:* Although large volumes of RNA and protein sequences are available, experimentally verified interaction data remain relatively scarce,

limiting the generalization ability of supervised models.

- *Diverse interaction patterns*: RPIs involve multiple types (binding [36], regulation [37], catalysis [38], *etc.*), whereas most existing methods treat them as a single homogeneous relation, thereby losing important semantic distinctions.
- *Cold-start problem*: When a new RNA or protein appears without known interaction edges, conventional GNNs fail to produce meaningful embeddings, leading to degraded prediction performance.

To address these challenges, we propose EGRL, which integrates four key components: (1) an implicit meta-path learning module that automatically discovers and aggregates relation-level semantics from multiple edge types to enrich node representations, alleviating data sparsity without requiring additional labels; (2) a multi-relational GAT that computes separate attention weights for each edge type (*e.g.*, RNA-RNA similarity, protein-protein similarity, known RPIs, and generated soft edges), thereby preserving interaction heterogeneity; (3) a graph generator that predicts probabilistic soft edges between new and existing nodes using only sequence features, trained with a pseudo cold-start auxiliary loss to enable dynamic integration of unseen nodes during inference; and (4) an interaction predictor with multi-feature fusion, which captures both linear and multiplicative relationships for final interaction scoring. By integrating these components, EGRL achieves competitive performance on benchmark datasets and demonstrates strong generalization in cold-start scenarios.

4. Proposed method

We present the architecture of EGRL, as illustrated in Figure 2. The framework consists of four key components: (1) **Implicit meta-path learning**, which automatically learns the importance of different relational paths; (2) **Graph generator**, which dynamically synthesizes soft edges for cold-start nodes; (3) **Multi-relational GAT**, which propagates information across multiple edge types; and (4) **Interaction predictor** with multi-feature fusion, which combines node embeddings for final interaction scoring.

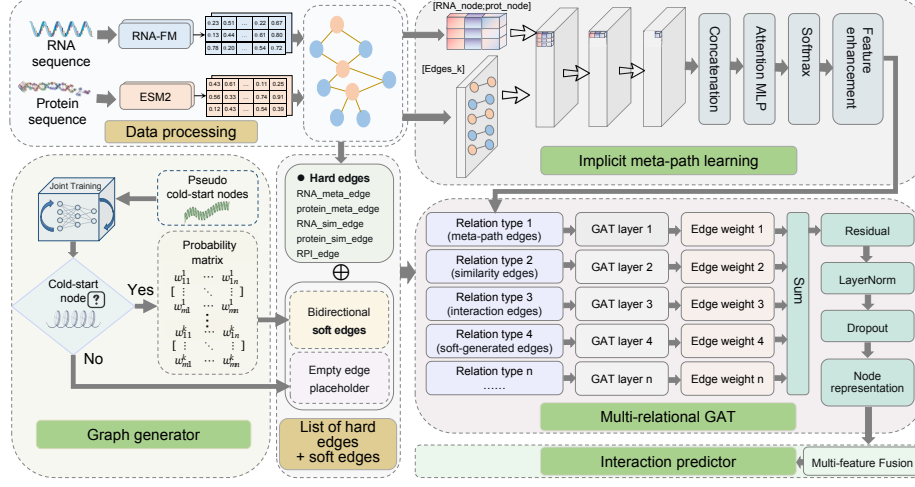


Figure 2: **Overall architecture of EGRL.** Illustration of the multi-relational graph modeling, implicit meta-path learning, and generator-based enhancement.

4.1. Implicit meta-path learning

Given a heterogeneous graph with node set $\mathcal{V} = \mathcal{R} \cup \mathcal{P}$ (RNAs and proteins) and multiple edge types $\mathcal{R} = \{r_1, r_2, \dots, r_K\}$, we aim to capture the semantic importance of each meta-path without manual design. Let $\mathbf{X} \in \mathbb{R}^{N \times d}$ denote the initial node feature matrix (after linear projection), where $N = |\mathcal{R}| + |\mathcal{P}|$ and d is the hidden dimension.

For each relation type r_k , we define the set of involved nodes as:

$$\hat{\mathcal{N}}_k = \{i \mid (i, j) \in \mathcal{E}_k \text{ or } (j, i) \in \mathcal{E}_k\}, \quad (1)$$

where $\mathcal{E}_k$ denotes the edge set of type k . A relation-specific linear transformation $\mathbf{W}_k \in \mathbb{R}^{d \times d}$ is applied, and the transformed features are averaged to obtain the relation representation:

$$\mathbf{m}_k = \frac{1}{|\hat{\mathcal{N}}_k|} \sum_{i \in \hat{\mathcal{N}}_k} \mathbf{W}_k \mathbf{X}[i, :]. \quad (2)$$

The importance of each relation is computed via an attention network operating on

concatenated representations:

$$\alpha = \text{Softmax}(\text{MLP}([\mathbf{m}_1 \parallel \cdots \parallel \mathbf{m}_K])), \quad (3)$$

where $\parallel$ denotes concatenation and the *Multilayer Perceptron* (MLP) consists of two linear layers with ReLU and dropout. The aggregated meta-path feature is:

$$\mathbf{h}_{\text{meta}} = \sum_{k=1}^K \alpha_k \mathbf{m}_k. \quad (4)$$

Finally, $\mathbf{h}_{\text{meta}}$ is broadcast to all nodes and added to the original features via a residual connection:

$$\mathbf{X}' = \mathbf{X} + \text{Dropout}(\mathbf{h}_{\text{meta}}). \quad (5)$$

This enriches node representations with relation-level semantics while preserving node-specific information.

4.2. Graph generator for cold-start nodes

To handle cold-start scenarios where new RNAs or proteins appear without known interactions, we introduce a graph generator that predicts probabilistic edges between new and existing nodes. The generator is a pairwise MLP that takes node features as input and outputs interaction probabilities.

Let $\mathbf{X}_{\text{new}} \in \mathbb{R}^{M \times d}$ denote the embeddings of M cold-start nodes (after the same feature projection and implicit meta-path enhancement as in Section 4.1), and let $\mathbf{X}_{\text{old}} \in \mathbb{R}^{N_{\text{old}} \times d}$ denote the embeddings of existing nodes. For each pair (i, j) , the interaction probability is computed as:

$$p_{ij} = \sigma(\text{MLP}_{\text{gen}}([\mathbf{X}_{\text{new}}[i, :] \parallel \mathbf{X}_{\text{old}}[j, :]])), \quad (6)$$

where $\parallel$ denotes concatenation and σ is the sigmoid function. MLP_{gen} consists of two hidden layers with ReLU activation followed by a linear output layer. The output is a probability matrix $\mathbf{P} \in [0, 1]^{M \times N_{\text{old}}}$.

To incorporate these predicted interactions into the graph, we construct bidirectional soft edges with weights equal to the predicted probabilities:

$$\mathbf{I}_{\text{edge_index}} = \begin{bmatrix} \mathbf{c} \otimes \mathbf{1}_{N_{\text{old}}} & \mathbf{1}_M \otimes \mathbf{t} \\ \mathbf{1}_M \otimes \mathbf{t} & \mathbf{c} \otimes \mathbf{1}_{N_{\text{old}}} \end{bmatrix}, \quad \mathbf{W}_{\text{edge_weight}} = \begin{bmatrix} \text{flatten}(\mathbf{P}) \\ \text{flatten}(\mathbf{P}) \end{bmatrix}, \quad (7)$$

where $\mathbf{c}$ and $\mathbf{t}$ denote the indices of cold-start and existing nodes, respectively, and $\otimes$ denotes a Kronecker product-like expansion that enumerates all pairwise connections. These soft edges are appended to the existing edge list and treated as an additional relation type in subsequent GAT layers.

Training Strategy for the Graph Generator: To ensure that the generator produces meaningful soft edges, we jointly train it with the main model using a self-supervised auxiliary loss. During each training epoch, a subset of RNA and protein nodes is randomly selected as pseudo cold-start nodes. For each such node, the generator predicts interaction probabilities with all candidate nodes, and a binary cross-entropy loss is computed against the ground-truth interaction matrix:

$$\mathcal{L}_{\text{gen}} = \frac{1}{M_{\text{rna}}} \sum_{i \in C_{\text{rna}}} \text{BCE}(\mathbf{P}_{i,\cdot}, \mathbf{Y}_{i,\cdot}) + \frac{1}{M_{\text{prot}}} \sum_{j \in C_{\text{prot}}} \text{BCE}(\mathbf{P}_{\cdot,j}, \mathbf{Y}_{\cdot,j}), \quad (8)$$

where C_{rna} and C_{prot} denote the pseudo cold-start RNA and protein sets, respectively, $\mathbf{P}$ is the predicted probability matrix, and $\mathbf{Y}$ is the corresponding submatrix of the training interaction matrix. This auxiliary loss is combined with the main prediction loss using a weighting factor λ_{gen} (see Eq. (14)) and optimized jointly.

4.3. Multi-relational GAT

The enhanced node features $\mathbf{X}'$ (after meta-path learning, Eq. (5)) are fed into a multi-relational GAT to perform relation-aware message passing. We consider multiple edge types, including original hard edges (RNA-RNA, protein-protein, similarity, and known RNA-protein interactions) as well as soft edges generated for cold-start nodes (during both training and inference).

For each relation type r , an independent GAT convolution is applied:

$$\mathbf{h}_r = \text{GATConv}_r(\mathbf{X}', \mathcal{E}_r, \text{edge_attr} = \mathbf{W}_r), \quad (9)$$

where $\mathcal{E}_r$ denotes the edge index of relation r , and $\mathbf{W}_r$ is the corresponding edge weight vector (for soft edges) or omitted for hard edges. Each GATConv employs h attention heads and outputs features of dimension d .

The outputs from all relations are averaged and combined with a residual connection, followed by layer normalization:

$$\mathbf{X}^{(l+1)} = \text{LayerNorm} \left(\mathbf{X}^{(l)} + \text{Dropout} \left(\frac{1}{|\mathcal{R}|} \sum_{r=1}^{|\mathcal{R}|} \mathbf{h}_r \right) \right), \quad (10)$$

where $\mathbf{X}^{(0)} = \mathbf{X}'$ and $\mathcal{R}$ denotes the set of all relation types (including soft edges). This process is repeated for L layers, enabling the model to capture both local and multi-hop dependencies.

4.4. Interaction predictor with multi-feature fusion

After the final GAT layer, we obtain embeddings for RNAs and proteins: $\mathbf{h}_{\mathcal{R}} \in \mathbb{R}^{|\mathcal{R}| \times d}$ and $\mathbf{h}_{\mathcal{P}} \in \mathbb{R}^{|\mathcal{P}| \times d}$. For a given RNA-protein pair (r, p) , we construct a feature vector:

$$\mathbf{f}_{\text{cross}} = [\mathbf{h}_{\mathcal{R}}[r] \parallel \mathbf{h}_{\mathcal{P}}[p] \parallel \mathbf{h}_{\mathcal{R}}[r] \odot \mathbf{h}_{\mathcal{P}}[p] \parallel \mathbf{h}_{\mathcal{R}}[r] - \mathbf{h}_{\mathcal{P}}[p]] \in \mathbb{R}^{4d}, \quad (11)$$

where $\odot$ denotes element-wise product. This formulation captures both linear and multiplicative relationships.

The feature vector is then passed through a two-layer predictor network with layer normalization and ReLU activation:

$$\hat{y} = \sigma(\mathbf{W}_2 \cdot \text{ReLU}(\text{LayerNorm}(\mathbf{W}_1 \mathbf{f}_{\text{cross}} + \mathbf{b}_1)) + \mathbf{b}_2), \quad (12)$$

where σ is the sigmoid function, $\mathbf{W}_1 \in \mathbb{R}^{d \times 4d}$ and $\mathbf{W}_2 \in \mathbb{R}^{1 \times d}$ are weight matrices, and $\mathbf{b}_1, \mathbf{b}_2$ are bias vectors. The output $\hat{y} \in [0, 1]$ denotes the predicted interaction probability.

4.5. Joint training objective

EGRL is trained by minimizing a weighted binary cross-entropy loss for the main prediction task, combined with a generator auxiliary loss. Let $y_i \in [0, 1]$ denote the ground-truth label for the i -th RNA-protein pair. The main loss is defined as:

$$\mathcal{L}_{\text{main}} = -\frac{1}{N} \sum_{i=1}^N \left[w_p \cdot y_i \log \hat{y}_i + (1 - y_i) \log (1 - \hat{y}_i) \right], \quad (13)$$

where w_p is the weight assigned to positive samples. In practice, $w_p = \beta \cdot \frac{\text{neg}}{\text{pos}}$, where β further emphasizes positive interactions.

The total loss is given by:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{main}} + \lambda_{\text{gen}} \mathcal{L}_{\text{gen}}, \quad (14)$$

where λ_{gen} is a hyperparameter controlling the contribution of the generator loss $\mathcal{L}_{\text{gen}}$ (see Eq. (8)). This joint optimization encourages the generator to learn meaningful soft edges from node features, enabling effective interaction prediction for unseen nodes during inference.

5. Experimental setup

We present the experimental setup, including research questions (RQs), datasets, comparison methods, and evaluation metrics.

5.1. Research questions

To evaluate the proposed approach, we investigate the following research questions (RQs):

1) RQ1: *How do the core architectural components of EGRL affect performance?*

To quantify the contribution of each module, we conduct an ablation study with five configurations:

- Full: All modules enabled.

- -A (w/o ImplicitMetaPath): Removes the implicit meta-path learner, directly feeding node features into the GAT.
- -B (w/o MultiRelationalGAT): Replaces the multi-relational GAT with a single-relation GAT that merges all edge types, disabling relation-aware attention.
- -C (w/o GraphGenerator): Disables the graph generator; no soft edges are introduced for cold-start nodes.
- -D (w/o Multi-featureFusion): Removes multi-feature fusion (Hadamard product and absolute difference), using only concatenated embeddings.

This design isolates the contribution of each component while controlling for model capacity.

2) RQ2: *How do key hyperparameters influence the performance of EGRL?*

We evaluate the robustness of the model with respect to its key hyperparameters.

Experiment 1: Number of k -nearest neighbours (kNN). We vary $k \in \{3, 5, 10, 15, 20\}$ to examine how the number of neighbors in the similarity graph affects performance under the normal scenario (without generator-based soft edges). Larger k increases connectivity but may introduce noise.

Experiment 2: Generator-related hyperparameters. Under the cold-start scenario, all test nodes are treated as unseen, and the generator produces soft edges. During training, a pseudo cold-start strategy is applied. We vary:

- Generator loss weight $\lambda_{\text{gen}} \in \{0.02, 0.05, 0.1, 0.2\}$, balancing generator learning and main task performance.
- Pseudo cold-start node ratio $\in \{0.01, 0.05, 0.1, 0.15\}$, controlling the number of simulated unseen nodes.

3) RQ3: *How does EGRL compare with state-of-the-art methods?*

We compare EGRL with representative RPIP methods on benchmark datasets, including RLF-LPI [39], RPI-SAN [40], RPITER [41], IPMiner [42], and NPI-GNN [43] (see Section 5.3).

4) RQ4: *Can EGRL generalize to completely unseen RNA and protein families?*

Table 1: Statistics of the four benchmark RNA-protein interaction (RPI) datasets.

Dataset	# RNAs	# Proteins	# Interactions	# Non-interactions
RPI369 [45]	332	338	369	0
RPI1807 [22]	1,078	3,131	1,807	1,436
RPI2241 [45]	841	2,042	2,241	0
NPInter2 [46]	4,636	449	10,412	0

To assess generalization under cold-start conditions, we conduct two evaluations:

Experiment 1: Molecule hold-out. Selected molecules are entirely removed from training, and their interactions are predicted at test time using generator-produced soft edges.

Experiment 2: Sequence-cluster-based partitioning. To prevent data leakage, RNA and protein sequences are clustered using CD-HIT [44] with identity thresholds of 80% (RNA) and 40% (protein). Entire clusters are held out as test data, ensuring no high-similarity overlap with training samples. This setting evaluates generalization to novel sequence families under a strict cold-start scenario.

5.2. Datasets

We utilize four benchmark datasets from prior studies [41], as summarized in Table 1, to evaluate our method. The datasets RPI369, RPI1807, and RPI2241 are derived from RNA-protein complexes collected from the RNA-Protein Interaction Database (PRIDB) [47] and the Protein Data Bank (PDB) [48]. These datasets are constructed based on the minimum atomic distance criterion, where an RNA-protein pair is considered interacting if the distance between any RNA atom and protein atom falls below a predefined threshold. RPI369 is a subset of RPI2241 [45], excluding interactions involving ribosomal proteins or ribosomal RNA. The RPI1807 dataset is obtained by parsing data from the Nucleic Acid Database (NDB) [49] and PRIDB, which provide RNA-protein complex structures and atomic interaction interfaces.

In contrast, the NPInter2 dataset consists of experimentally validated ncRNA-protein interactions, particularly involving lncRNAs, collected from the NPInter2 database [46], rather than being defined by structural distance criteria.

Since RPI369, RPI2241, and NPInter2 do not provide negative samples, we follow [41] to generate an equal number of non-interacting pairs by randomly combining

RNAs and proteins from positive samples. To reduce false negatives, pairs are discarded if they are highly similar to known interactions; specifically, a generated pair (R_1, P_1) is removed if there exists (R_2, P_2) such that the sequence identity between R_1 and R_2 exceeds 80% and that between P_1 and P_2 exceeds 40%.

5.3. Previous RPIP methods under comparison

We evaluate EGRL on the four datasets by comparing it with representative methods in RPIP. The following approaches are considered:

- **RLF-LPI** [39]: An ensemble framework that integrates a long short-term memory autoencoder with an attention mechanism to extract latent representations, combined with a fuzzy decision-making strategy to model lncRNA-protein relationships.
- **RPI-SAN** [40]: A sequence-based method that combines deep learning techniques with a random forest classifier, leveraging RNA sequences and protein evolutionary information for prediction.
- **RPITER** [41]: A multi-level deep learning framework that integrates convolutional neural networks and stacked autoencoders, exploiting both sequence and structural information of RNA and proteins.
- **IPMiner** [42]: A stacked autoencoder-based approach that learns latent interaction patterns from RNA and protein sequences, followed by a random forest classifier within an ensemble framework.
- **NPI-GNN** [43]: An end-to-end graph neural network-based method for RPIP, which models interactions using both network structure and sequence information.

5.4. Evaluation metrics

We adopt the commonly used five-fold cross-validation to evaluate model performance. Multiple metrics are employed, including *Accuracy* (ACC), *Precision* (PREC), *Recall* (REC), *F1-score* (F1), and *Matthews Correlation Coefficient* (MCC). In addition,

threshold-independent metrics, namely the *Area Under the Receiver Operating Characteristic Curve* (AUROC) and the *Area Under the Precision-Recall Curve* (AUPR), are also reported.

The metrics are defined as follows:

$$\text{ACC} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}}, \quad (15)$$

$$\text{PREC} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \quad (16)$$

$$\text{REC} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \quad (17)$$

$$\text{MCC} = \frac{\text{TP} \times \text{TN} - \text{FP} \times \text{FN}}{\sqrt{(\text{TP} + \text{FP})(\text{TP} + \text{FN})(\text{TN} + \text{FP})(\text{TN} + \text{FN})}}, \quad (18)$$

$$\text{F1} = 2 \cdot \frac{\text{PREC} \times \text{REC}}{\text{PREC} + \text{REC}}, \quad (19)$$

$$\text{AUROC} = \int_0^1 \text{TPR}(\text{FPR}) d(\text{FPR}), \quad (20)$$

$$\text{AUPR} = \int_0^1 \text{PREC}(\text{REC}) d(\text{REC}), \quad (21)$$

where TP, FP, TN, and FN denote the numbers of true positives, false positives, true negatives, and false negatives, respectively. TPR denotes the true positive rate (recall), and FPR is defined as:

$$\text{FPR} = \frac{\text{FP}}{\text{FP} + \text{TN}}. \quad (22)$$

5.5. Hardware and software environment

EGRL is implemented in Python using PyTorch 2.6.0+cu118 and PyTorch Geometric 2.6.1. The input features for RNA and protein nodes are obtained from pre-trained language models: RNA-FM provides 256-dimensional embeddings for RNAs, and ESM2-t33-650M produces 1280-dimensional embeddings for proteins. These features are projected into a unified 128-dimensional hidden space via linear layers.

The hidden dimension of both the meta-path processor and the relation-aware network is set to 128. The model is trained using the AdamW optimizer with an initial learning rate of 1×10^{-4} and a cosine annealing scheduler. Training is performed for 300 epochs with a batch size of 256. A weighted binary cross-entropy loss is employed,

where the positive sample weight is dynamically set to $10 \cdot \frac{\text{neg}}{\text{pos}}$ to address class imbalance. Gradient clipping is applied with a maximum norm of 1.0, and dropout is set to 0.2. All experiments are conducted on a single NVIDIA V100 (32GB) GPU using five-fold cross-validation.

6. Experimental results

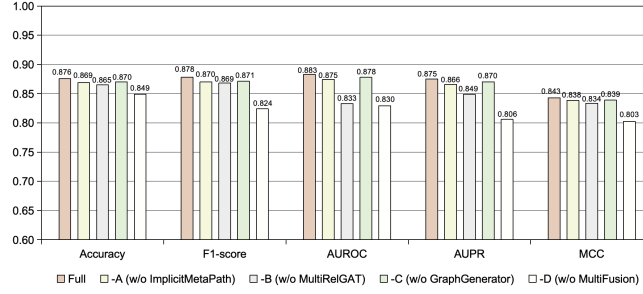
In this section, we mainly provide the experimental results to answer the above RQs.

6.1. Answer to RQ1: Ablation study

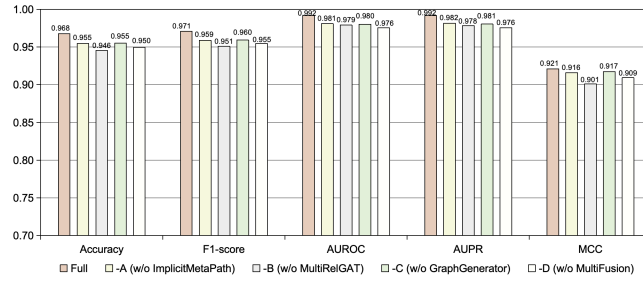
The ablation results on the four datasets are shown in Figure 3. Overall, the core components of EGRL contribute to the final performance to different extents. Removing the implicit meta-path module (-A) decreases the Accuracy on RPI2241 from 0.925 to 0.889. For the graph generator (-C), its removal leads to only slight performance degradation, e.g., on RPI369 (Accuracy: 0.876 $\rightarrow$ 0.870, F1: 0.878 $\rightarrow$ 0.871) and RPI1807 (Accuracy: 0.968 $\rightarrow$ 0.955), suggesting limited benefit under the five-fold cross-validation setting, which does not fully reflect strict cold-start scenarios.

In contrast, removing the multi-feature fusion module (-D) results in substantial degradation across all metrics. On RPI369, Accuracy drops from 0.876 to 0.849 and AUPR from 0.875 to 0.806; on RPI2241, Accuracy decreases from 0.925 to 0.859 and MCC from 0.888 to 0.820. This indicates that simple concatenation is insufficient to capture effective interactions, while the Hadamard product and absolute difference play a key role in modeling feature interactions. The multi-relational GAT module (-B) is also critical: removing it reduces AUROC on RPI369 from 0.883 to 0.833 and Accuracy on RPI1807 from 0.968 to 0.946, highlighting the importance of relation-aware modeling in heterogeneous graphs.

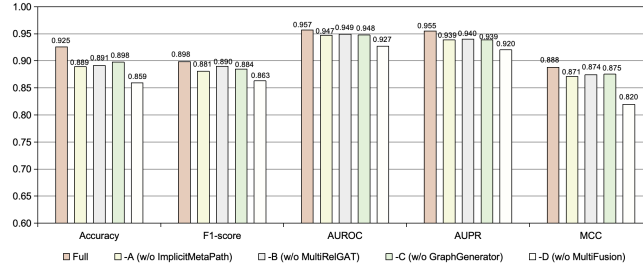
Notably, all ablated variants contain fewer parameters than the full model. Overall, the complete EGRL consistently achieves the best performance, indicating that its effectiveness arises from the coordinated design of multiple modules. The multi-feature fusion enhances discriminative capability, while the multi-relational GAT captures structural information. The implicit meta-path module and graph generator further improve generalization from semantic and data augmentation perspectives, respectively.



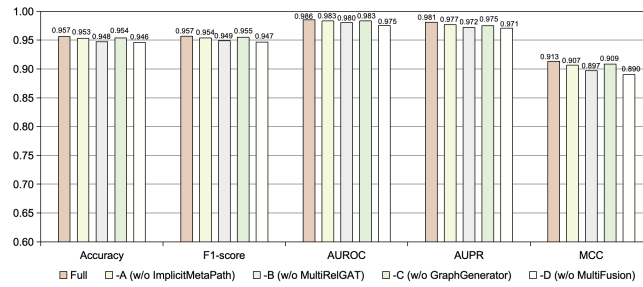
(a) On RPI369 dataset.



(b) On RPI1807 dataset.



(c) On RPI2241 dataset.



(d) On NPInter2 dataset.

Figure 3: **Ablation study on four datasets.** Performance comparison of different EGRL variants under component removal.

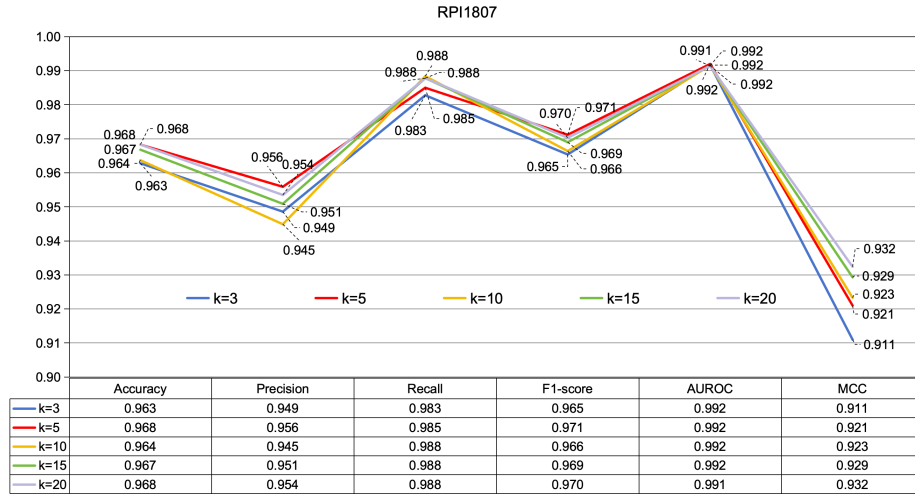


Figure 4: **Effect of kNN neighbors.** Impact of the number of neighbors k on model performance under the standard evaluation setting.

Summary of answers to RQ1: All ablated variants underperform the full model, with varying degrees of degradation, confirming that EGRL benefits from the synergistic design of its components rather than parameter scaling alone.

6.2. Answer to RQ2: Impacts of hyperparameter tuning

Following Section 5.1, we conduct systematic analyses on three key hyperparameters: the number of kNN neighbors (k), the generator loss weight (λ_{gen}), and the pseudo cold-start node ratio (*cold-ratio*).

We first examine the effect of k on model performance. As shown in Figure 4, performance on the RPI1807 dataset remains stable across different values of k . While AUROC stays consistently high as k increases from 3 to 20, Recall improves from 0.983 to 0.988, and both F1-score and MCC show a gradual increase (with MCC peaking at 0.932 for $k = 20$). Precision slightly decreases at $k = 10$ (0.945), indicating that enlarging the neighborhood captures more positives but may introduce noise, reflecting a precision-recall trade-off. Overall, $k = 5$ provides a balanced configuration in terms of Accuracy (0.968) and Precision (0.956), whereas $k = 20$ favors higher Recall and MCC. These results indicate that the model is robust to the choice of neighborhood size under

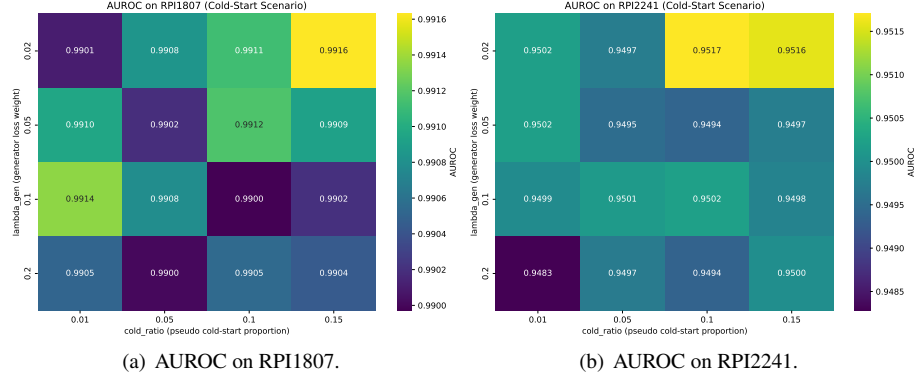


Figure 5: **Effect of generator hyperparameters under cold-start scenarios.** Sensitivity of AUROC to λ_{gen} and cold-start ratio.

the standard evaluation setting.

In cold-start scenarios, where test nodes are absent from the training graph, the soft-edge completion mechanism of the graph generator becomes critical. We therefore further evaluate the impact of λ_{gen} and cold_ratio . As shown in Figure 5, on both RPI1807 and RPI2241, varying λ_{gen} within $[0.02, 0.05, 0.1, 0.2]$ and cold_ratio within $[0.01, 0.05, 0.1, 0.15]$ leads to only minor AUROC fluctuations (within 0.0016 and 0.0022, respectively). Although the absolute AUROC on RPI2241 (approximately 0.95) is lower than that on RPI1807 (approximately 0.99), both datasets consistently show low sensitivity to these hyperparameters. This indicates that the generator module operates stably across a wide configuration range and effectively supports cold-start prediction.

In summary, EGRL demonstrates strong robustness across both normal and cold-start scenarios. Its performance remains stable with respect to k , adapting well to different local graph structures, and shows minimal sensitivity to generator-related hyperparameters (λ_{gen} and cold_ratio). These observations suggest that the model maintains reliable performance over a broad hyperparameter space, reducing the need for extensive tuning.

Table 2: **Performance comparison on four datasets under five-fold cross-validation.** The best and second-best results are highlighted in lightgreen and lightred , respectively.

Dataset	Method	ACC	PREC	REC	MCC	AUROC
RPI369	RLF-LPI [39]	0.844	0.819	0.882	0.610	0.905
	RPI-SAN [40]	-	-	-	-	-
	RPITER [41]	0.728	0.701	0.797	0.461	0.821
	IPMiner [42]	0.752	0.713	0.735	0.507	0.773
	NPI-GNN [43]	0.602	0.600	0.615	0.212	-
	EGRL (Ours)	0.876	0.878	0.875	0.843	0.883
RPI1807	RLF-LPI [39]	0.973	0.968	0.984	0.927	0.987
	RPI-SAN [40]	0.961	0.914	0.936	0.924	0.999
	RPITER [41]	0.968	0.959	0.986	0.936	0.990
	IPMiner [42]	0.986	0.978	0.982	0.972	0.998
	NPI-GNN [43]	-	-	-	-	-
	EGRL (Ours)	0.968	0.962	0.993	0.921	0.992
RPI2241	RLF-LPI [39]	0.842	0.874	0.800	0.549	0.924
	RPI-SAN [40]	0.907	0.840	0.861	0.822	0.962
	RPITER [41]	0.890	0.871	0.917	0.781	0.957
	IPMiner [42]	0.824	0.836	0.833	0.650	0.906
	NPI-GNN [43]	0.626	0.672	0.498	0.270	-
	EGRL (Ours)	0.925	0.877	0.933	0.888	0.957
NPIInter2	RLF-LPI [39]	-	-	-	-	-
	RPI-SAN [40]	0.986	-	-	-	-
	RPITER [41]	0.955	0.939	0.973	0.910	0.985
	IPMiner [42]	0.952	0.945	0.946	0.904	0.995
	NPI-GNN [43]	0.933	0.915	0.956	0.868	0.970
	EGRL (Ours)	0.957	0.945	0.977	0.913	0.986

Summary of answers to RQ2: EGRL exhibits strong robustness to variations in the number of neighbors, the generator loss weight, and the pseudo cold-start node ratio, maintaining stable AUROC performance across settings.

6.3. Answer to RQ3: EGRL vs. State-of-the-art methods

We evaluate EGRL using Accuracy, Precision, Recall, MCC, and AUROC, with detailed results reported in Table 2. Based on comparisons with representative methods, the main observations are as follows.

EGRL achieves competitive performance across all four benchmark datasets. On RPI369, it attains the highest Accuracy, Precision, and MCC, indicating a clear advan-

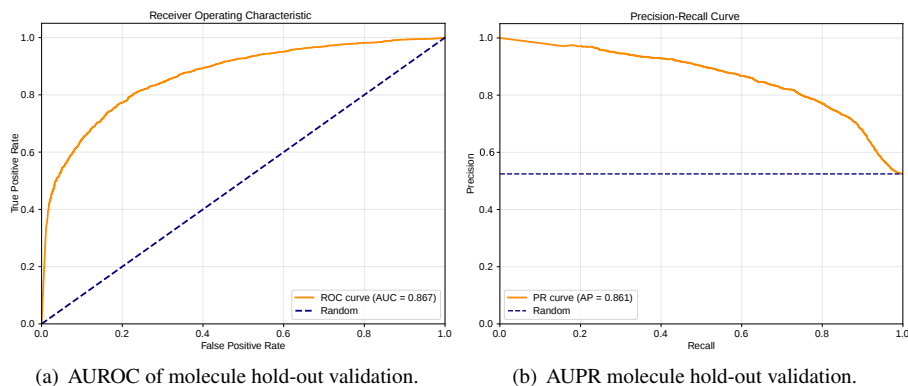


Figure 6: **Molecule hold-out cold-start evaluation.** AUROC and AUPR performance on NPInter2 under unseen node settings.

tage and strong capability in learning from limited data. On RPI1807, EGRL achieves the highest Recall (0.993), while other metrics remain close to the best-performing methods; for instance, its Precision (0.962) and AUROC (0.992) are slightly below those of IPMiner [42] (0.978) and RPI-SAN [40] (0.999), respectively, yet remain well balanced overall. On RPI2241, EGRL obtains the highest Accuracy (0.925) and MCC (0.888), with consistently strong performance across other metrics, demonstrating robustness on larger datasets. On NPInter2, EGRL achieves the highest Recall (0.977) and maintains competitive AUROC (0.986) and MCC (0.913), further validating its effectiveness across diverse data distributions.

Summary of answers to RQ3: EGRL consistently delivers competitive performance across datasets of different scales and characteristics, demonstrating robust and well-balanced predictive capability.

6.4. Answer to RQ4: Generalization capability validation

Figure 6 presents the molecule hold-out cold-start results on the NPInter2 dataset. Evaluation on the unseen test set yields an AUROC of 0.867 and an AUPR of 0.861. Although these values are lower than those obtained under five-fold cross-validation (see Table 2), EGRL achieves state-of-the-art performance under the same evaluation

Table 3: **Cold-start performance for RNA-protein interaction prediction on unseen nodes.** The best and second-best results are highlighted in lightgreen and lightred, respectively.

Method	AUROC	AUPR
NPI-GNN [43]	0.511	0.520
IPMiner [42]	0.664	0.739
RPITER [41]	0.727	0.774
ZHMolGraph	0.798	0.820
EGRL(ours)	0.867	0.861

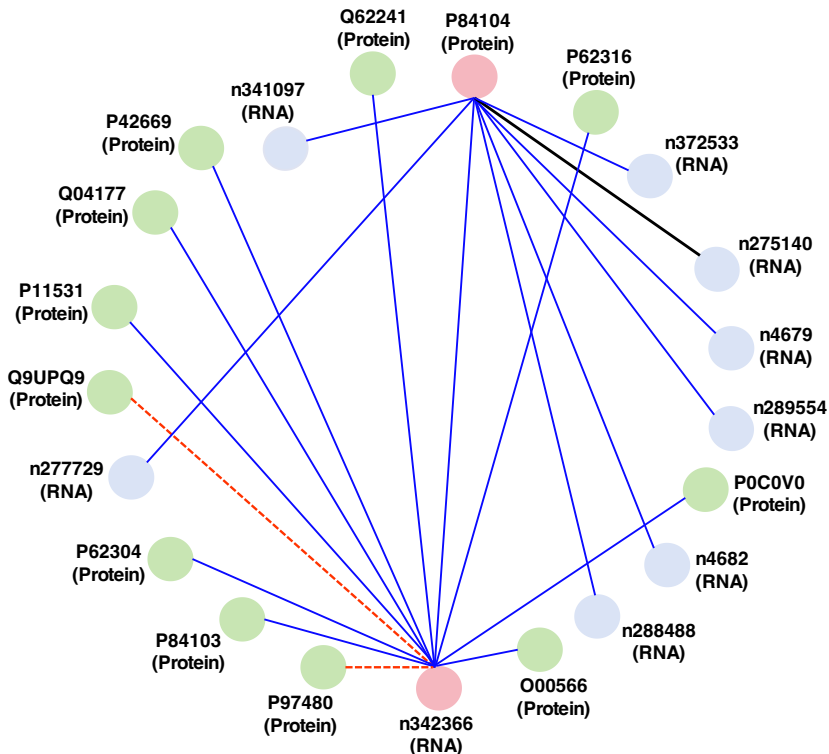
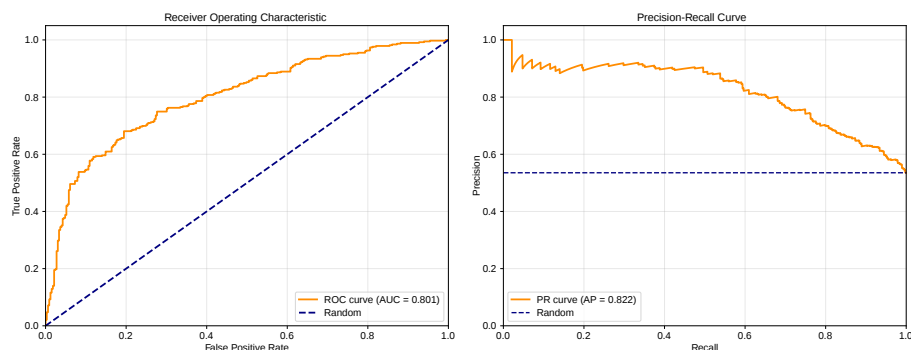


Figure 7: **Molecule hold-out interaction prediction.** Visualization of predicted interactions for removed molecules.

protocol (see Table 3, with comparative data from [50]). In particular, compared with the previous state-of-the-art method ZHMolGraph [50] (AUROC: 0.798, AUPR: 0.820), EGRL improves AUROC by approximately 8.6% and AUPR by 5.0%.

Figure 7 further illustrates the molecule hold-out results. After removing the protein (P84104) and the RNA (n342366) from NPIInter2, these nodes are treated as cold-start targets. For protein P84104, only one interacting RNA is missed, while for RNA



(a) AUROC of sequence similarity clustering validation. (b) AUPR of sequence similarity clustering validation.

Figure 8: Sequence-cluster-based cold-start evaluation. AUROC and AUPR performance under cluster-level data partitioning.

n342366, all interacting partners are correctly predicted. Some newly predicted interactions (red dashed lines) may correspond to false positives or potentially undiscovered interactions, indicating both prediction uncertainty and exploratory capability.

Figure 8 reports the results under the more stringent sequence-cluster-based setting. Evaluation on independent clusters yields an AUROC of 0.801 and an AUPR of 0.822. Compared with the molecule hold-out setting (AUROC: 0.867, AUPR: 0.861), both metrics decrease, likely because this strategy removes clusters with high sequence similarity to the training data, thereby reducing potential information leakage and increasing task difficulty.

As shown in Figure 9, although false negatives (black lines) and false positives (red lines) exist, the majority of interactions are correctly predicted (blue lines). This indicates that EGRL maintains stable predictive capability even under more challenging conditions.

Summary of answers to RQ4: Both validation settings demonstrate that EGRL achieves strong generalization in cold-start scenarios, maintaining reliable predictions even when specific molecules or entire sequence clusters are excluded from training.

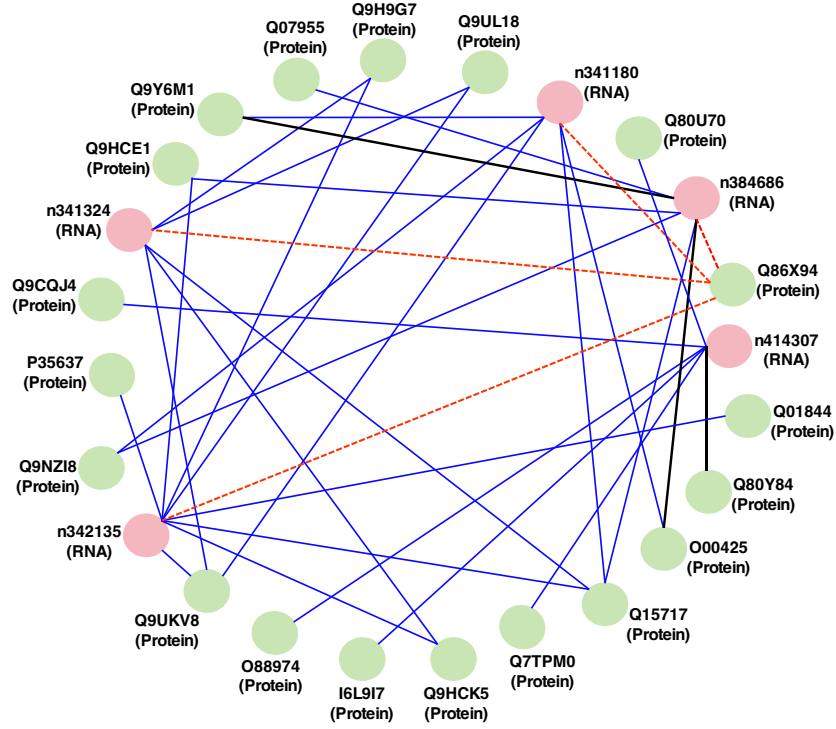


Figure 9: **Sequence-cluster-based interaction prediction.** Visualization of predicted interactions under cluster-level cold-start setting.

7. Conclusions and future work

Graph neural networks (GNNs) have emerged as a promising paradigm for RNA-protein interaction (RPI) prediction. Given the multi-level and multi-type regulatory mechanisms underlying RPIs, constructing heterogeneous RNA-protein graphs that capture both node-level and path-level relational information is essential. Such modeling provides a solid foundation for advancing AI-driven biomedical research and drug discovery.

In this work, we propose EGRL for accurate RPI prediction. EGRL performs relation-aware encoding over multi-relational graphs. Its key strengths are threefold: (1) it automatically captures interaction patterns via implicit meta-path exploration; (2) it models multiple subgraphs (RNA-RNA, protein-protein, similarity, RPI, and generator-induced soft edges) using a multi-relational GAT and integrates their representations;

and (3) it incorporates a graph generator trained jointly with the backbone to enhance generalization to unseen nodes (cold-start). Extensive experiments based on five-fold cross-validation demonstrate that EGRL achieves competitive performance across four benchmark datasets. Moreover, it shows robust generalization under both molecule hold-out and sequence-cluster-based settings, indicating its ability to capture underlying network connectivity and support the discovery of potential novel interactions.

For future work, we plan to explore multimodal data integration by incorporating additional biological information, such as RNA secondary structure, protein tertiary structure, gene expression profiles, and epigenetic signals, together with sequence features. This is expected to provide more comprehensive representations and further improve prediction accuracy and reliability. We also aim to develop more transferable frameworks for RPI prediction across species and tissue types, thereby extending the applicability of artificial intelligence in bioinformatics.

CRedit authorship contribution statement

Danyu Li: Conceptualization, Data curation, Formal analysis, Methodology, Project administration, Visualization, Writing - original draft; **Ling Zhou:** Conceptualization, Formal analysis, Methodology, Validation, Writing - original draft, Writing - review & editing; **Rubing Huang:** Funding acquisition, Methodology, Supervision, Validation, Writing - original draft, Writing - review & editing; **Xian Zhong:** Formal analysis, Methodology, Validation, Visualization, Writing - review & editing; **Bin Zou:** Conceptualization, Data curation, Validation, Writing - review & editing; **Kui Jiang:** Conceptualization, Formal analysis, Methodology, Writing - review & editing.

Declaration of Generative AI and AI-assisted technologies in the writing process

The authors used a generative AI tool solely for language editing, including grammar, formatting, and spelling checks. The authors take full responsibility for the content of the manuscript.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported in part by the Science and Technology Development Fund of Macau, Macao SAR, under Grants 0069/2025/RIB2 and 0021/2023/RIA1, the National Natural Science Foundation of China under Grant 62271361, the Hubei Provincial Key Research and Development Program under Grant 2024BAB039, and the Fundamental Research Funds for the Central Universities under Grant 104972026KFYzxk0029.

Data availability

Data will be made available on request.

References

- [1] P. Avila-Lopez, S. M. Lauberth, Exploring new roles for RNA-binding proteins in epigenetic and gene regulation, *Current Opinion in Genetics & Development* 84 (2024) 102136.
- [2] E. Wanowska, A. McFeely, J. Sztuba-Solinska, The role of epitranscriptomic modifications in the regulation of RNA–protein interactions, *BioChem* 2 (4) (2022) 241–259.
- [3] Y. Zhang, J. Wu, Q. Zhao, M. Dong, J. Deng, X. Gao, K. Hu, D. Xiong, Improving mutation pathogenicity prediction of metal-binding sites in proteins with a panoramic attention mechanism, *Pattern Recognit.* 169 (2026) 111935.
- [4] C. Xu, Y. Cao, J. Bao, Building RNA-protein germ granules: insights from the multifaceted functions of DEAD-box helicase vasa/ddx4 in germline development, *Cellular and Molecular Life Sciences* 79 (2022) 1–15.

- [5] J. B. Bertoldo, S. Müller, S. Hüttelmaier, RNA-binding proteins in cancer drug discovery, *Drug Discovery Today* 28 (6) (2023) 103580.
- [6] Y. Xie, W. Wang, J. Xu, L. Li, S. Wu, H. Wei, ncRD-LG: integrating molecular language models and subgraph learning for drug–ncRNA target prediction, *Pattern Recognit.* 179 (2026) 113556.
- [7] T. S. Stenum, A. D. Kumar, F. A. Sandbaumhüter, J. Kjellin, J. Jerlström-Hultqvist, P. E. Andrén, S. Koskiniemi, E. T. Jansson, E. Holmqvist, RNA interactome capture in escherichia coli globally identifies RNA-binding proteins, *Nucleic Acids Research* 51 (9) (2023) 4572–4587.
- [8] L. Y. Han, C. Z. Cai, S. L. Lo, M. C. M. Chung, Y. Z. Chen, Prediction of RNA-binding proteins from primary sequence using support vector machines, *RNA* 10 (3) (2004) 355–368.
- [9] Z.-P. Liu, L.-Y. Wu, Y. Wang, X.-S. Zhang, L. Chen, Prediction of protein–RNA binding sites using random forest with combined features, *Bioinformatics* 26 (13) (2010) 1616–1622.
- [10] D. Li, R. Huang, C. Cui, D. Towey, L. Zhou, J. Tian, B. Zou, Ribonucleic acid–protein interaction prediction based on deep learning: a comprehensive survey, *Appl. Soft Comput.* (2025) 113795.
- [11] S. Patiyal, A. Dhall, K. Bajaj, H. Sahu, G. P. Raghava, Prediction of RNA-interacting residues in a protein using CNN and evolutionary profile, *Briefings in Bioinformatics* 24 (1) (2023) bbac538.
- [12] X. Pan, P. Rijnbeek, J. Yan, H.-B. Shen, Prediction of RNA–protein binding preferences using deep convolutional and recurrent neural networks, *BMC Genomics* 19 (1) (2018) 511.
- [13] C. Shen, D. Mao, J. Tang, Z. Liao, S. Chen, Prediction of lncRNA–protein interactions based on kernel combinations and graph convolutional networks, *IEEE J. Biomed. Health Inform.* 28 (4) (2023) 1937–1948.

- [14] J. Yang, Y. Li, G. Wang, Z. Chen, D. Wu, End-to-end knowledge graph fused graph neural network for protein–protein interaction prediction, *IEEE/ACM Trans. Comput. Biol. Bioinform.* (2024).
- [15] X. Bi, W. Ma, H. Jiang, W. Lu, J. Nie, Z. Wei, S. Zhang, Protein interaction pattern recognition via heterogeneous semantic mining and hierarchical graph representation, *Pattern Recognit.* 172 (2026) 112563.
- [16] W. Yin, S. Wang, Y. Zhang, S. Qiao, S. Wang, F. Khan, R. Altuki, Z. Lyu, Autonomously adjusting multi-relational hypergraph structures for predicting circRNA–miRNA associations, *IEEE J. Biomed. Health Inform.* (2025).
- [17] M. Wei, L. Wang, X. Su, B. Zhao, Z. You, Multi-hop graph structural modeling for cancer-related circRNA–miRNA interaction prediction, *Pattern Recognit.* 170 (2026) 112078.
- [18] J. Zhang, B. Liu, Z. Wang, K. Lehnert, M. Gahegan, DeepPN: a parallel neural network combining CNN and GCN for predicting RNA–protein binding sites, *BMC Bioinform.* 23 (1) (2022) 257.
- [19] L. Zhuo, B. Song, Y. Liu, Z. Li, X. Fu, Predicting ncRNA–protein interactions via dual graph convolution and pairwise learning, *Brief. Bioinform.* 23 (6) (2022) bbac339.
- [20] S. Liang, S. Liu, J. Song, Q. Lin, S. Zhao, S. Li, J. Li, S. Liang, J. Wang, HM-CDA: a heterogeneous graph neural network with metapath for circRNA–disease association prediction, *BMC Bioinform.* 24 (1) (2023) 335.
- [21] A. Liu, B. Zhang, P. Zeng, Metapath-based multiple instance learning with heterogeneous graph neural networks for lncRNA–disease association prediction, in: *Proc. IEEE Smart World Congr.*, 2024, pp. 2471–2476.
- [22] V. Suresh, L. Liu, D. Adjero, X. Zhou, RPI-Pred: predicting ncRNA–protein interaction using sequence and structural information, *Nucleic Acids Res.* 43 (3) (2015) 1370–1379.

- [23] Z. Pan, S. Zhou, H. Zou, C. Liu, M. Zang, T. Liu, Q. Wang, MCNN: multiple convolutional neural networks for RNA–protein binding site prediction, *IEEE/ACM Trans. Comput. Biol. Bioinform.* 20 (2) (2022) 1180–1187.
- [24] J. Zhang, Q. Chen, B. Liu, DeepDRBP-2L: identifying DNA- and RNA-binding proteins via CNN and LSTM, *IEEE/ACM Trans. Comput. Biol. Bioinform.* 18 (4) (2019) 1451–1463.
- [25] Y. Wang, X. Wang, C. Chen, H. Gao, A. Salhi, X. Gao, B. Yu, RPI-CapsuleGAN: predicting RNA–protein interactions via interpretable adversarial capsule networks, *Pattern Recognit.* 141 (2023) 109626.
- [26] J. Zhao, J. Sun, S. C. Shuai, Q. Zhao, J. Shuai, Predicting potential interactions between lncRNAs and proteins via combined graph auto-encoder methods, *Brief. Bioinform.* 24 (1) (2023) bbac527.
- [27] G. Zhao, P. Li, X. Qiao, X. Han, Z.-P. Liu, Predicting lncRNA–protein interactions by heterogeneous network embedding, *Front. Genet.* 12 (2022) 814073.
- [28] Y. Wang, P. Ding, C. Wang, S. He, X. Gao, B. Yu, RPI-GGCN: prediction of RNA–protein interactions via gated graph convolution and co-regularized variational autoencoders, *IEEE Trans. Neural Netw. Learn. Syst.* (2024).
- [29] D. Li, J. Xie, J. Chen, Y. Ma, Heterogeneous graph representation with LSTM for predicting RNA–protein interactions, in: *Proc. Int. Conf. Electron. Commun. Netw. Comput. Technol.*, 2024, pp. 323–326.
- [30] Y. Ma, H. Zhang, C. Jin, C. Kang, Predicting lncRNA–protein interactions with bipartite graph embedding and deep graph neural networks, *Front. Genet.* 14 (2023) 1136672.
- [31] L. Deng, J. Yang, H. Liu, Predicting circRNA–disease associations using metapath-based representation learning on heterogeneous networks, in: *Proc. IEEE Int. Conf. Bioinform. Biomed.*, 2020, pp. 5–10.

- [32] W. Cao, Y. Chen, J.-Y. Yang, F.-Y. Xue, Z.-H. Yu, J. Feng, Z.-J. Wu, J. Gong, X.-H. Niu, Metapath-aggregated multilevel graph embedding for miRNA–disease association prediction, in: *Proc. IEEE Int. Conf. Bioinform. Biomed.*, 2023, pp. 468–473.
- [33] X. Wang, M. Zhang, C. Long, L. Yao, M. Zhu, Self-attention based neural network for predicting RNA–protein binding sites, *IEEE/ACM Trans. Comput. Biol. Bioinform.* 20 (2) (2022) 1469–1479.
- [34] Z. Jiang, Y.-Y. Shen, R. Liu, Structure-based prediction of nucleic acid binding residues via deep learning and template-based integration, *PLOS Comput. Biol.* 19 (9) (2023) e1011428.
- [35] D. Matus, F. Runge, J. K. H. Franke, L. Gerne, M. Uhl, F. Hutter, R. Backofen, RNA–protein interaction classification via sequence embeddings, *bioRxiv* (2024).
- [36] Y. Xiao, Y.-M. Chen, Z. Zou, C. Ye, X. Dou, J. Wu, C. Liu, S. Liu, H. Yan, P. Wang, Profiling RNA-binding protein binding sites via in situ reverse transcription sequencing, *Nat. Methods* 21 (2) (2024) 247–258.
- [37] H. Nakanishi, T. Yoshii, S. Kawasaki, K. Hayashi, K. Tsutsui, C. Oki, S. Tsukiji, H. Saito, Light-controllable RNA–protein devices for translational regulation in mammalian cells, *Cell Chem. Biol.* 28 (5) (2021) 662–674.
- [38] J. Strecker, F. E. Demircioglu, D. Li, G. Faure, M. E. Wilkinson, J. S. Gootenberg, O. O. Abudayyeh, H. Nishimasu, R. K. Macrae, F. Zhang, RNA-activated protein cleavage with a CRISPR-associated endopeptidase, *Science* 378 (6622) (2022) 874–881.
- [39] J. Song, S. Tian, L. Yu, Q. Yang, Q. Dai, Y. Wang, W. Wu, X. Duan, RLF-LPI: an ensemble learning framework for predicting lncRNA–protein interaction, *Math. Biosci. Eng.* 19 (2022) 4749–4764.
- [40] H.-C. Yi, Z.-H. You, D.-S. Huang, X. Li, T.-H. Jiang, L.-P. Li, A deep learning framework for accurate prediction of ncRNA–protein interactions using evolutionary information, *Mol. Ther. Nucleic Acids* 11 (2018) 337–344.

- [41] C. Peng, S. Han, H. Zhang, Y. Li, RPITER: a hierarchical deep learning framework for ncRNA–protein interaction prediction, *Int. J. Mol. Sci.* 20 (5) (2019) 1070.
- [42] X. Pan, Y.-X. Fan, J. Yan, H.-B. Shen, IPMiner: hidden ncRNA-protein interaction sequential pattern mining with stacked autoencoder, *BMC Genomics* 17 (2016) 1–14.
- [43] Z.-A. Shen, T. Luo, Y.-K. Zhou, H. Yu, P.-F. Du, NPI-GNN: predicting ncRNA–protein interactions with graph neural networks, *Brief. Bioinform.* 22 (5) (2021) bbab051.
- [44] W. Li, A. Godzik, CD-HIT: a fast program for clustering and comparing large protein or nucleotide sequence sets, *Bioinformatics* 22 (13) (2006) 1658–1659.
- [45] U. K. Muppirala, V. G. Honavar, D. Dobbs, Predicting RNA-protein interactions using only sequence information, *BMC Bioinform.* 12 (2011) 1–11.
- [46] J. Yuan, W. Wu, C. Xie, G. Zhao, Y. Zhao, R. Chen, NPInter v2.0: an updated database of ncRNA interactions, *Nucleic Acids Res.* 42 (D1) (2014) D104–D108.
- [47] B. A. Lewis, R. R. Walia, M. Terribilini, J. Ferguson, C. Zheng, V. Honavar, D. Dobbs, PRIDB: a protein–RNA interface database, *Nucleic Acids Res.* 39 (Suppl. 1) (2010) D277–D282.
- [48] H. M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T. N. Bhat, H. Weissig, I. N. Shindyalov, P. E. Bourne, The protein data bank, *Nucleic Acids Res.* 28 (1) (2000) 235–242.
- [49] H. M. Berman, J. D. Westbrook, Z. Feng, L. Iype, B. Schneider, C. Zardecki, The nucleic acid database, *Struct. Bioinform.* 44 (2003) 199–216.
- [50] H. Liu, Y. Jian, C. Zeng, Y. Zhao, RNA-protein interaction prediction using network-guided deep learning, *Communications Biology* 8 (1) (2025) 247.