Computational Understanding of non-coding RNA Pairwise Interactions

Marco Nicolini [1], Federico Stacchietti [1], Elena Casiraghi [1,2,3,4] and Giorgio Valentini,[1,2,*]

[1] *AnacletoLab, Dipartimento di Informatica, Università degli Studi di Milano, Italy*
[2] *ELLIS - European Lab for Learning and Intelligent Systems, Milan Unit*
[3] *Environmental Genomics and Systems Biology Division, Lawrence Berkeley National Laboratory, Berkeley, CA, United States*
[4] *Department of Computer Science, Aalto University, Espoo, Finland*

Correspondence*:
Giorgio Valentini
valentini@di.unimi.it

ABSTRACT

Non-coding RNAs (ncRNAs) govern a vast network of regulatory interactions within the cells, yet their pairwise relationships remain largely uncharted due to the complexity of RNA structure and the limits of current experimental methods. We present *CUPID* (Computational Understanding of Pairwise Interactions in ncRNA Data), a deep learning framework that predicts ncRNA-ncRNA interactions directly from primary sequence information. *CUPID* uses embeddings from a pre-trained RNA language model combined with a feed-forward classifier to identify patterns linked to molecular pairing. This approach avoids reliance on thermodynamic models or manual feature design and, unlike previously proposed models, is able to generalize across different types of ncRNAs, including long non-coding, circular, micro-, and small nuclear RNAs. By learning the hidden rules that govern RNA recognition, *CUPID* provides a scalable tool for exploring ncRNA interaction networks and advancing our understanding of RNA-based regulation.

Keywords: ncRNA–ncRNA interaction, deep learning, fine-tuning, artificial intelligence, machine learning, non-coding RNA, large language models

1 INTRODUCTION

Understanding RNA-RNA interactions is critical for deciphering the regulatory circuits that orchestrate gene expression, RNA processing, and signal transduction. Non-coding RNAs (ncRNAs), despite lacking protein-coding potential, play pivotal roles in these processes (Ali et al., 2021). However, experimental mapping of ncRNA interactions remains challenging due to the limitations of existing experimental and computational techniques (Lorenzi et al., 2021).

Methods such as Minimum Free Energy (MFE) calculations and accessibility-based models have been usually applied to predict RNA-RNA interactions. Tools like IntaRNA (Mann et al., 2017) estimate the interaction energy as $\Delta G_{\text{total}} = \Delta G_{\text{duplex}} + \Delta G_{\text{accessibility}}$, where the first term quantifies the energy released upon hybridization, and the second accounts for the cost of rendering binding regions accessible. Benchmark studies have demonstrated that accessibility-based algorithms can effectively differentiate native interactions from background noise (Umu and Gardner, 2017), yet these approaches rely on predefined parameters and simplified energy models. In parallel, experimental techniques such as RNA Antisense Purification (RAP-RNA) offer validation but remain limited by their high cost and labor intensity (Engreitz et al., 2014).

Advances in machine learning and graph-based modeling for biological data, including recent work on explainability and diffusion-based attention mechanisms, have motivated a surge of learning-driven approaches for predicting interactions across diverse molecular systems (Gliozzo et al., 2025; Cetin and Sefer, 2025; Sefer, 2025).

Machine learning methods, such as convolutional neural networks, deep forests and graph neural networks (Alipanahi et al., 2015; Tian et al., 2021; Wei et al., 2022) have been applied to RNA-protein interaction prediction, while graph-based approaches embed heterogeneous networks of ncRNAs and diseases using multigraph contrastive learning (Sun et al., 2025) or apply random-walk based graph representation learning techniques to predict non coding RNA interactions (Torgano et al., 2025).

While effective, these methods often rely on predefined feature extraction, graph structures, or supervised training, limiting their adaptability to novel ncRNA sequences.

In contrast, LLMs can directly learn from large corpora of proteins or RNA data (Valentini et al., 2023; Zhao et al., 2024; Shen et al., 2024; Nicolini et al., 2025a), capturing intricate interaction motifs beyond predefined energy models or graph-based constraints. Unlike thermodynamic models, which impose simplifying assumptions, LLMs infer interaction likelihoods from latent structural patterns, offering a flexible, data-driven approach. In particular, transformer-based foundation models can generate biologically meaningful representations directly from raw sequences, by exploiting large RNA sequence corpora (Sapoval et al., 2022; Chen et al., 2022; Yu et al., 2024). More in general several deep learning methods have been proposed to predict specific ncRNA interactions, using rna2vec pretraining and deep feature mining (Yu et al., 2022) or conditional random fields and graph convolutional networks (Wang et al., 2022), heterogeneous graph neural networks (Li et al., 2025) and convolutional neural networks combined with a Transformer Encoder (Yang et al., 2025) for the prediction of miRNA-lncRNA interactions.

We also recently proposed a deep neural network trained on embedded representations of a subset of ncRNAs obtained from the RNA-FM language model (Shen et al., 2024), achieving state-of-the-art results for predicting miRNA interactions with other ncRNA molecules (Nicolini et al., 2025b). However, our proposed model, like other models recently proposed in the literature (Li et al., 2025; Yang et al., 2025), is only able to predict specific ncRNA interactions (e.g., interactions with miRNAs). Furthermore, due to limitations on the maximum allowed sequence length of the underlying RNA-FM transformer, it can only process sequences shorter than approximately 1000 nucleotides, thus limiting the model's application to relatively long ncRNAs (e.g., lncRNAs).

To overcome these limitations, we propose a novel Transformed-based deep learning model, that, differently from previous models proposed in literature, is able to predict a large range of ncRNA interactions, including long non-coding RNA (lncRNA), circular RNA (circRNA), microRNA (miRNA), small nuclear RNA (snRNA), small nucleolar RNA (snoRNA), Small Cajal body-specific RNAs (scaRNAs), small cytoplasmic RNAs (scRNA) and other types of ncRNAs. Moreover, by adopting GenerRNA (Zhao et al., 2024) to encode RNA sequences, our model can process full-length ncRNA sequences (up to 4096 nucleotides) without truncation, thus significantly enlarging the set of ncRNAs that can be processed by the model.

We hypothesize that LLM-based contextual embeddings provide a rich representation for ncRNA interaction prediction, circumventing the limitations of manual feature engineering or predefined structural graphs. We reasoned that GenerRNA (Zhao et al., 2024), pretrained on a large corpus of ncRNA sequences using a masked language modeling objective, can capture long-range interactions of ncRNA molecules, thus facilitating downstream tasks such as ncRNA interaction prediction.

Our *CUPID* model (Computational Understanding of Pairwise Interactions in ncRNA Data), predicts ncRNA interactions using only sequence information. *CUPID* extracts embeddings from a pre-trained ncRNA language model and feeds a dense feed-forward neural network (FFNN) to automatically learn intricate sequence interaction features. This design circumvents the need for explicit thermodynamic parameterization and manually engineered features, offering a scalable and efficient alternative for uncovering novel regulatory interactions (Fabbri et al., 2019).

2 METHODS

2.1 Dataset

Our dataset comprises a subset of multispecies ncRNA interaction pairs from RNA-KG Cavalleri et al. (2024)[1].

The RNA-KG integrates physical and functional interactions between different types of ncRNAs, and their relationships with other biomolecules (genes and proteins) and chemicals, as well as with biomedical concepts coded in the Gene Ontology (Aleksander et al., 2023), the Human Phenotype Ontology (Gargano et al., 2023), Mondo (Vasilevsky et al., 2025) and other bio-medical ontologies related to the "RNA world".

In particular, we extracted RNA–RNA edges from RNA-KG by selecting only relations annotated as `interacts-with`. In RNA-KG, `interacts-with` denotes experimentally supported *physical* RNA–RNA interactions, and we therefore excluded other relation types encoding functional associations (e.g., regulatory links, co-expression, or disease associations). The `interacts-with` edges integrated in RNA-KG originate from multiple underlying curated interaction databases. Fig.1 presents an overview of the main RNA entities and their relationships available in the the RNA-KG. Readers may refer to the RNA-KG reference (Cavalleri et al., 2024) for the complete list of contributing sources and evidence provenance.

We filtered the dataset to retain only sequences that fit within the GenerRNA (Zhao et al., 2024)'s token limit (approximately 4096 nucleotides), since Byte Pair Encoding (BPE) compresses raw nucleotide sequences, allowing longer sequences to fit within the model's constraints. After applying this length filter, the dataset contains:

- 101088 interaction pairs (down from an initial 130310 pairs).
- 11212 unique sequences (selected from 19624 potential sequences) belonging to 9 different RNA molecule types: long non-coding RNA (lncRNA), circular RNA (circRNA), microRNA (miRNA), small nuclear RNA (snRNA), small nucleolar RNA (snoRNA), Small Cajal body-specific RNAs (scaRNAs), small cytoplasmic RNAs (scRNA), not (better) classified non coding RNA molecules (ncRNA) and pseudo RNA[2].

[1] Retrieval of interacting pairs and corresponding sequences was performed using the scripts available from the RNA-KG web site: `https://github.com/AnacletoLAB/RNA-KG`.

[2] In RNAinter, the term "pseudo" specifically denotes RNA sequences transcribed from pseudogenes. In this context, these are transcripts derived from genes that have lost their protein-coding capability due to accumulated mutations, yet they are still produced as RNA. Similar to other ncRNAs, such pseudogene RNAs can sometimes participate in regulatory networks by, for example, acting as miRNA decoys or sponges, despite not encoding functional proteins.

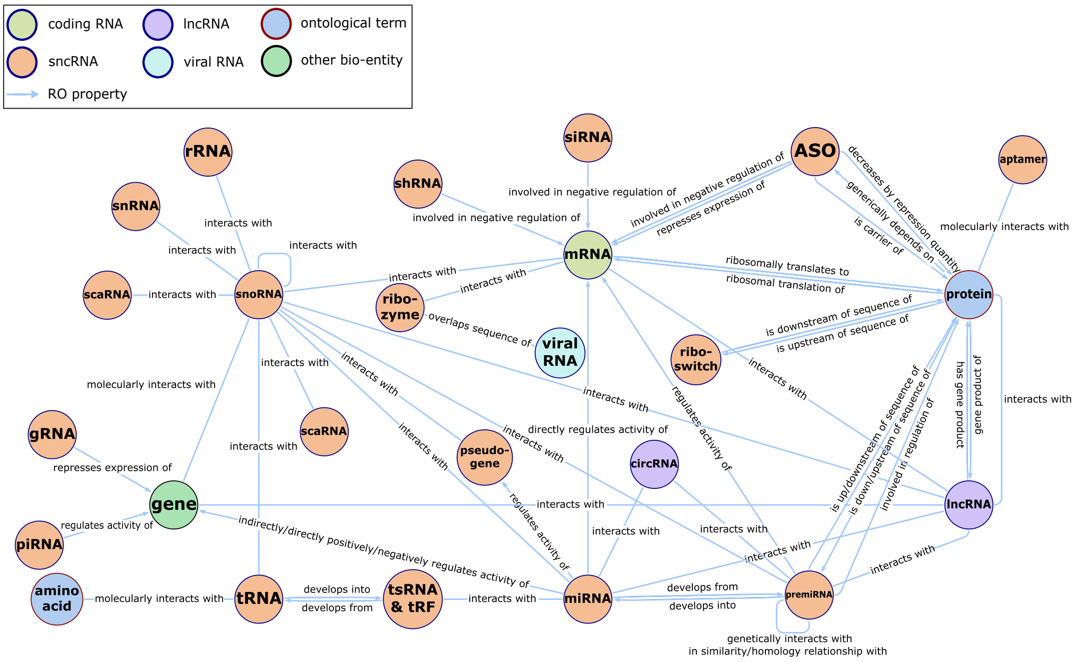


Figure 1. Simplified representation of the RNA-KG meta-graph, focused on ncRNAs and their interactions.

In the following, we denote the set of length-filtered molecules as

$$\mathcal{S} = \{s_i\}, \quad i = 1, \ldots, |\mathcal{S}|,$$

where the type of each molecule $s \in \mathcal{S}$ is given by $\phi(s)$, i.e. $\phi : \mathcal{S} \to \mathcal{T}$ represents a mapping of a ncRNA sequence $s \in \mathcal{S}$ to its ncRNA type $\mathcal{T}$, e.g. miRNA, lncRNA or any other ncRNA type.

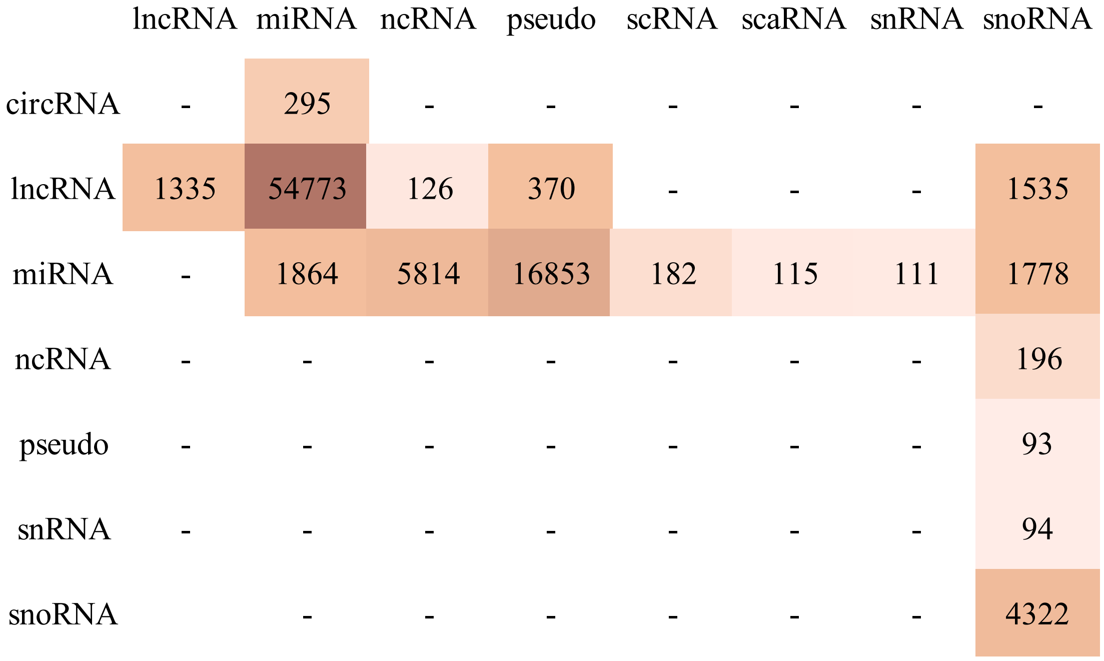
The identity of an interaction pair is solely determined by its constituent molecules, regardless of order; that is,

$$(s_i, s_j) = (s_j, s_i).$$

The type of an interaction (s_i, s_j) with $s_i \neq s_j$ and $s_i, s_j \in \mathcal{S}$ is determined by the types of the ncRNA s_i and s_j theirselves, regardless of their order:

$$(\phi(s_i), \phi(s_j)) = (\phi(s_j), \phi(s_i))$$

For instance, possible types of ncRNA interactions are miRNA-lncRNA or miRNA-circRNA. Assuming that interacting ncRNA pairs of different types exhibit distinct specificities that the model should learn, we reasoned that types with negligible sample sizes might introduce noise rather than valuable information. Therefore, the set of interaction pairs used in this work is obtained by further filtering the dataset of interacting pairs to remove interacting pair types represented by fewer than 100 samples, resulting in 10644 unique sequences composing 99841 interacting pairs. Fig 2 shows the distribution of the different types of ncRNA interactions.



	lncRNA	miRNA	ncRNA	pseudo	scRNA	scaRNA	snRNA	snoRNA
circRNA	-	295	-	-	-	-	-	-
lncRNA	1335	54773	126	370	-	-	-	1535
miRNA	-	1864	5814	16853	182	115	111	1778
ncRNA	-	-	-	-	-	-	-	196
pseudo	-	-	-	-	-	-	-	93
snRNA	-	-	-	-	-	-	-	94
snoRNA		-	-	-	-	-	-	4322

Figure 2. Distribution of ncRNA interactions pairs in the filtered interaction set. Rows: first (left) molecule type; Columns: right molecule type.

2.2 Data Augmentation

To address the issues due to the limited cardinality of the available training data, especially for specific types of ncRNA interactions (e.g., snRNA-miRNA or miRNA-circRNA), we employ a data augmentation strategy that effectively increases the dataset size by a factor of 4. For each original training instance represented as a pair of interacting ncRNA (s_i, s_j) we generate three additional augmented instances:

1. Molecule Order Reversal: Swap the order of the molecules: (s_j, s_i).
2. Sequence Flipping: Reverse the nucleotide order in both molecules (denoted by the superscript F): (s_i^F, s_j^F).
3. Combined Augmentation: Reverse both the molecule order and the nucleotide sequences: (s_j^F, s_i^F).

Thus, if the original dataset contains N instances, the augmented dataset becomes: $N_{\text{aug}} = 4N$ (Suppl. Fig. S1). This augmentation introduces invariance to both the order and orientation of sequences, thereby enabling the model to better capture the underlying biological patterns and improving its robustness against input variability.

In order to avoid leakage between training and test sets, data augmentation is performed after splitting the dataset.

2.3 Negative examples generation

In our dataset, only positive non-coding RNA-RNA interactions are explicitly provided, and they occur with varying frequencies.

Algorithm 1 Negative Sampling Algorithm

```
Require: Set of unique ncRNA sequences $\mathcal{S}$, positive augmented interaction set $\mathcal{P}$, and negative
    sampling parameter $n$
Ensure: Negative sample set $\mathcal{N}$
 1: Initialize $\mathcal{N} \leftarrow \emptyset$
 2: for each pair $(s, s') \in \mathcal{P}$ do
 3:     for $i = 1$ to $n$ do
 4:         Sample $s_{\text{neg}} \in \mathcal{S}$ such that $\phi(s_{\text{neg}}) = \phi(s')$
 5:         if $(s, s_{\text{neg}}) \notin \mathcal{P} \wedge (s, s_{\text{neg}}) \notin \mathcal{N}$ then
 6:             $\mathcal{N} \leftarrow \mathcal{N} \cup \{(s, s_{\text{neg}})\}$
 7:         end if
 8:     end for
 9: end for
10: return $\mathcal{N}$
```

To effectively train *CUPID*, we generated negative examples for each interaction pair type by matching the frequency distribution of the positive interactions. Specifically, negative examples were generated under the assumption that any pair of ncRNA sequences drawn from the set of unique sequences that is not observed as a positive interaction constitutes a possible negative instance.

Let $\mathcal{S} = \{s_1, s_2, \ldots, s_N\}$ be the set of unique ncRNA sequences present in the dataset. Denote by

$$\mathcal{P} = \{(s_i, s_j) \mid s_i, s_j \in \mathcal{S} \text{ interact}\}$$

the set of all positive ncRNA-ncRNA interactions. Then, the set of all possible ncRNA pairs is given by $\mathcal{S} \times \mathcal{S}$ (excluding self-interactions).

The set of *potential negatives* is defined as:

$$\mathcal{N}_{\text{potential}} = \{(s_i, s_j) \in \mathcal{S} \times \mathcal{S} \mid s_i \neq s_j\} \setminus \mathcal{P}.$$

Negative Sampling Procedure. To generate the negative samples for each interacting pair type, we corrupt its tuples. In other words, given a positive pair (s_i, s_j) with type $(\phi(s_i), \phi(s_j))$, we keep the first molecule s_i fixed and sample $s' \in \mathcal{S}$ such that:

$$s' \neq s_i, \quad \phi(s') = \phi(s_j), \quad (s_i, s') \notin \mathcal{P}$$

In this way we avoid generating negatives between ncRNA types that never interact (e.g. scaRNA and lncRNA).

Because we generate negatives for each positive pair (s_i, s_j) by corrupting the right molecule while keeping the same type pair $(\phi(s_i), \phi(s_j))$, the negative set preserves the interaction *type-pair* distribution of the positives in expectation (and approximately in practice, up to rejection of candidates already present as positives or previously sampled negatives).

For each positive edge, we selected n negative edges, in order to control the imbalance between positive and negative edges in the testing phase (we set $n = 20$ in our experiments).

Negative Sampling Algorithm. The negative sampling algorithm is detailed in Algorithm 1. In our implementation, we set $n = 20$. Note that, since the condition at line 5 of the algorithm cannot be always guaranteed, it is likely that the number of negatives $n \leq 20$. In our experiments we set $n = 20$.

2.4 Model Architecture

2.4.1 The overall *CUPID* Architecture

Our model follows a two-stage pipeline, as illustrated in Figure 3. It first extracts ncRNA sequence embeddings using a pre-trained ncRNA Language Model (GenerRNA Zhao et al. (2024)) and then processes these embeddings through a Feed-Forward Neural Network (FFNN) to predict interaction probabilities.

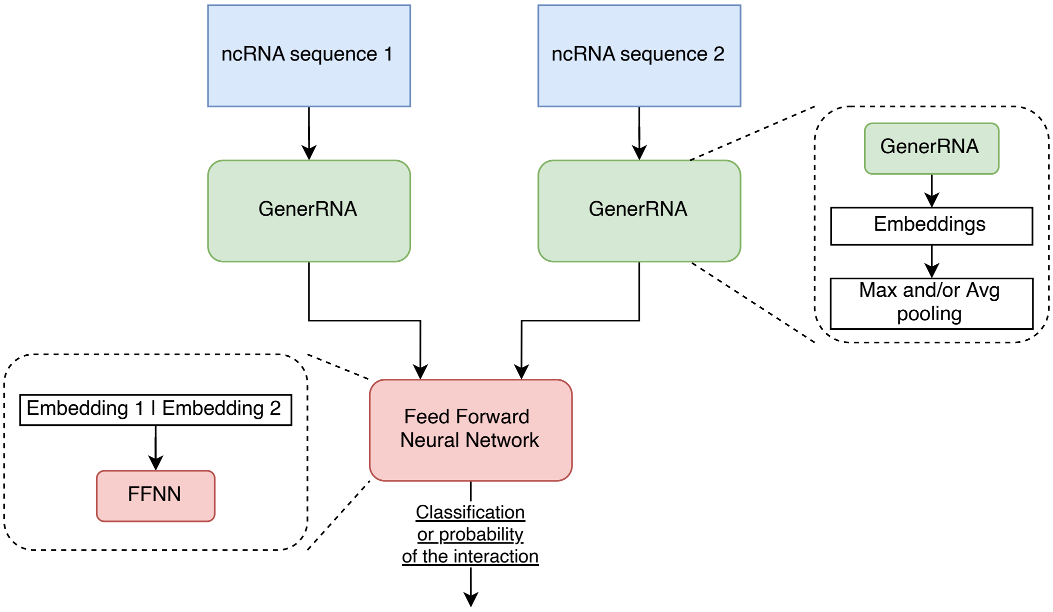

ncRNA sequence 1
ncRNA sequence 2
GenerRNA
GenerRNA
GenerRNA
Embeddings
Max and/or Avg pooling
Embedding 1 | Embedding 2
FFNN
Feed Forward Neural Network
Classification or probability of the interaction

Figure 3. High-level *CUPID* architecture schema.

The GenerRNA architecture mimics the GPT-2-medium model (Radford et al., 2019), and is composed of 24 stacked transformer-decoder layers, each incorporating a self-attention mechanism that models pairwise interactions among all positions in its input sequence. GenerRNA uses a context window of 1024 tokens, corresponding to input sequences with a length of approximately 4096 nucleotides coded through byte pair encoding Sennrich et al. (2016). Note that this maximum length permits the encoding of large RNA molecules. This decoder-only Transformer architecture operates in an autoregressive manner, predicting the subsequent token given the previous ones. Both the input and output of the model are represented as tokens, which are encoded and decoded by a trained tokenizer using byte pair encoding. A special token (EOS) is used to delimit sequences, indicating the start and end of each sequence.

Each transformer block is fed with a input of size $L \times H$, thus allowing to process RNA sequences having up to L tokens, each one represented through a H-dimensional real vector, with $L = H = 1024$, and outputs a latent representation with the same dimensionality for each input token. For each input sequence, the block employs a multi-head self-attention mechanism with 16 attention heads. This is followed by an "Add & Norm" sub-block, which applies residual addition and layer normalization. Subsequently, a feed-forward sub-layer expands the hidden states from 1024 to 4096 dimensions, applies a non-linear activation (ReLU), and then projects them back to 1024 dimensions. Another "Add & Norm" sub-block is applied after the feed-forward network, and finally, the block produces an output matrix $\mathbf{X} \in \mathbb{R}^{L \times H}$. A schematic diagram of this block is reported in the Fig 4.

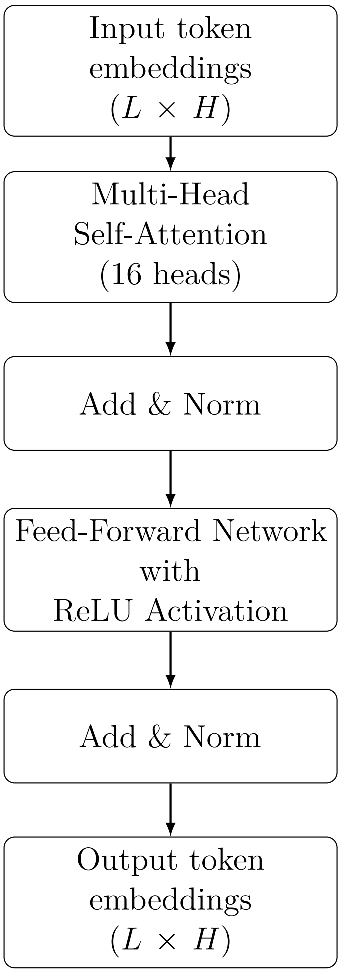

Input token embeddings ($L \times H$)
Multi-Head Self-Attention (16 heads)
Add & Norm
Feed-Forward Network with ReLU Activation
Add & Norm
Output token embeddings ($L \times H$)

Figure 4. High-level architecture of a GenerRNA block.

2.4.2 Pooling techniques

The i^{th} row of matrix $\mathbf{X}$ is a latent representation $\mathbf{x}_i \in \mathbb{R}^{1024}$ of the i^{th} token. To obtain a fixed-length embedding for the entire sequence, we tested two types of pooling over the sequence (i.e., across the H tokens), as well as their concatenation:

- Average (*Avg*) Pooling: obtained as the mean of the embeddings of all the tokens: $\mathbf{e}_{\text{avg}} = \frac{1}{L}\sum_{i=1}^{L} \mathbf{x}_i$.
- Maximum (*Max*) Pooling: Compute the element-wise maximum over all token embeddings: $\mathbf{e}_{\max} = \underset{i \in 1 \ldots L}{cmax}[x_{i1}, x_{i2}, \ldots x_{iH}]$, where *cmax* is the columnwise *max* operator, and x_{ij} are the elements of the $\mathbf{X}$ embedding matrix.
- Concatenation of [*Avg*, *Max*]: Combine both pooled representations into a single embedding vector: $\mathbf{e} = [\mathbf{e}_{\text{avg}}; \mathbf{e}_{\max}] \in \mathbb{R}^{2048}$.

These pooling strategies are schematically depicted in Fig. 5.

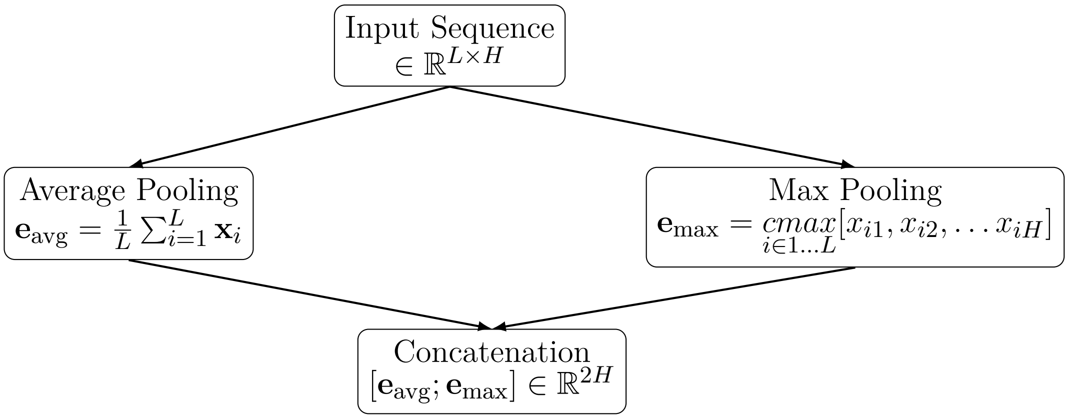

Input Sequence
$\in \mathbb{R}^{L \times H}$
Average Pooling
$\mathbf{e}_{\text{avg}} = \frac{1}{L}\sum_{i=1}^{L} \mathbf{x}_i$
Max Pooling
$\mathbf{e}_{\max} = \underset{i \in 1 \ldots L}{cmax}[x_{i1}, x_{i2}, \ldots x_{iH}]$
Concatenation
$[\mathbf{e}_{\text{avg}}; \mathbf{e}_{\max}] \in \mathbb{R}^{2H}$

Figure 5. Pooling Embedding Strategies.

The embedded representation of a candidate interacting ncRNA pair is composed by the concatenation of the embeddings of the two interacting molecules.

2.4.3 The classification unit

To predict the interaction we used the pooled embeddings of the RNA sequences as input to a dense Feed Forward Neural Network (FFNN), having the following architecture:

- **Input Layer Dimension:** 1024 for Avg and Max-pooling embedding strategies, 2048 when the embedding of the input molecule is obtained by concatenating the embeddings obtained by AVG and Max pooling,
- **Hidden Layers:** 4 hidden layers with 1024 neurons each and ReLU activation function,
- **Output Layer:** 1 neuron with sigmoid activation function.

To train the network we applied the following hyper-parameters:
Learning Rate: $\eta = 5 \times 10^{-4}$ with a linear warm-up phase of 4 epochs, followed by cosine decay.
Epochs: 50 epochs with early stopping (patience of 10 epochs). The model with the best validation loss is selected (e.g., if the lowest validation loss is observed at epoch 35, then early stopping is triggered at epoch 45, and the model from epoch 35 is used).
Batch Size: 512. *Dropout Rate:* 0.2. *Optimizer:* Adam. *Loss Function:* Binary Cross-Entropy.

Training and validation loss curves were monitored over epochs to assess model convergence and to avoid potential overfitting by early stopping.

2.4.4 Mini-batch balancing

Due to the imbalance in our dataset we adopted a training strategy designed to prevent the model from learning predominantly from the negatives. To address this, we constructed mini-batches that contain a controlled mix of positive and negative examples. Recall that our training set is composed of the set of positive interaction pairs, $\mathcal{P}$, $|\mathcal{P}| = N_+$, and the set of negative interaction pairs $\mathcal{N}$, with $|\mathcal{N}| = N_- = nN_+$, as detailed in Section 2.3. Each mini-batch B of size m is formed by randomly selecting m_p positive examples (using a uniform distribution with replacement) and m_n negative examples (using a uniform distribution without replacement). The ratio of negatives within each mini-batch is defined by

$$r = \frac{m_n}{m_n + m_p}, \quad \text{with} \quad m_n + m_p = m.$$

Here, r can vary between 0 and 1. A value of $r = 0.7$ implies that 70% of the mini-batch consists of negatives. The choice to sample positives with replacement is driven by their limited number, ensuring sufficient representation even in large batches, whereas sampling negatives without replacement allows for a broader coverage of these more abundant examples.

2.5 Experimental Evaluation

2.5.1 Data preparation and splitting.

In all our experiments the negative examples were sampled according to the relative frequency of the interacting pair types according to the procedure described in Section 2.3 using a negative:positive ratio equal to 20:1.

The dataset was partitioned into stratified training and test sets (train:test ratio = 90:10). The training set was further split into a stratified set for training (80% of interaction pairs) and the remaining 20% for validation. The validation set was used for early stopping and for tuning the classification threshold via maximization of the Matthews correlation coefficient (MCC Matthews (1975)) on the validation data.

2.5.2 Evaluation metrics.

To comprehensively assess model's performance, we computed a range of evaluation metrics, encompassing both threshold-dependent and threshold-independent measures. Specifically, we first evaluated standard binary classification metrics, including accuracy, balanced accuracy (to account for class imbalance), precision, recall, F1 score, AUROC (Area Under the Receiver Operating Characteristic Curve), and AUPRC (Area Under the Precision-Recall Curve). In addition to these overall metrics, we conducted a stratified analysis based on interacting pair types, computing the aforementioned measures separately for each pair type.

Let $y_i \in \{0, 1\}$ be the ground-truth label and $\hat{p}_i \in [0, 1]$ the predicted probability for sample i. Given a decision threshold t, we define $\hat{y}_i = 1 \iff [\hat{p}_i \geq t]$ and the confusion matrix counts:

$$\text{TP} = \sum_i \mathbf{I}[y_i = 1 \wedge \hat{y}_i = 1],$$

$$\text{TN} = \sum_i \mathbf{I}[y_i = 0 \wedge \hat{y}_i = 0],$$

$$\text{FP} = \sum_i \mathbf{I}[y_i = 0 \wedge \hat{y}_i = 1],$$

$$\text{FN} = \sum_i \mathbf{I}[y_i = 1 \wedge \hat{y}_i = 0].$$

Threshold-dependent metrics are then computed as:

$$\begin{aligned}
&\text{Accuracy (rate of correctly predicted instances)} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}},\\
&\text{Recall (proportion of TP w.r.t. all positive samples)} = \frac{\text{TP}}{\text{TP} + \text{FN}},\\
&\text{Specificity (proportion of TN w.r.t. all negative samples)} = \frac{\text{TN}}{\text{TN} + \text{FP}},\\
&\text{Precision (proportion of TP among predicted positives)} = \frac{\text{TP}}{\text{TP} + \text{FP}},\\
&\text{F1 (harmonic mean of precision and recall)} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}},\\
&\text{BalancedAcc (accuracy balanced by class proportion)} = \frac{\text{Recall} + \text{Specificity}}{2}.
\end{aligned} \tag{1}$$

In our work the threshold t is chosen on the validation set by maximizing the MCC coefficient, which provides a balanced single-score summary that incorporates TP, TN, FP, and FN, and is therefore less sensitive than accuracy to class imbalance:

$$\text{MCC} = \frac{\text{TP} \cdot \text{TN} - \text{FP} \cdot \text{FN}}{\sqrt{(\text{TP} + \text{FP})(\text{TP} + \text{FN})(\text{TN} + \text{FP})(\text{TN} + \text{FN})}}.$$

Threshold-independent metrics summarize performance across all thresholds. The ROC curve plots $\text{TPR}(t) = \text{Recall}(t)$ versus $\text{FPR}(t) = \text{FP}(t)/(\text{FP}(t) + \text{TN}(t))$, and AUROC is the area under this curve. The precision–recall curve plots $\text{Precision}(t)$ versus $\text{Recall}(t)$, and AUPRC is its area; under strong class imbalance, AUPRC is often more informative than AUROC, with a random baseline equal to the positive prevalence $\pi = \frac{N_+}{N_+ + N_-}$.

2.5.3 Training hyper-parameters and baselines for comparison.

The hyper-parameters and configurations used for training the FFNN are reported in Section 2.4.3. Moreover, training and validation loss curves were monitored over epochs to assess model convergence and to avoid potential overfitting by early stopping.

Hyperparameter selection was performed in preliminary experiments on a reduced subset of the training/validation interaction pairs using a grid-search strategy. We varied the number of hidden layers in $\{2, 4, 6\}$, the dropout rate in $\{0.1, 0.2\}$, and the batch size in $\{16, 512, 1024\}$. For each configuration, models were trained using the same optimization settings described in Section 2.4.3, and the final model was selected as the configuration that maximized validation AUPRC. No hyperparameters were tuned on the test set.

Besides the random classifier, whose expected performance are AUROC = 0.5 and AUPRC = 0.047, we employed the IntaRNA method (Mann et al., 2017) as a baseline for comparison. IntaRNA estimates interaction energy. While interaction energy can be thresholded to obtain binary predictions, which enable the computation of accuracy, balanced accuracy, precision, recall, and F1 scores, we opted to limit the comparison to AUROC and AUPRC. These metrics provide a more robust and threshold-independent evaluation of predictive performance, ensuring a fair comparison across models.

3 RESULTS

We assessed the contribution of the data-augmentation strategy and the pooling operation used to obtain molecule-level embeddings. Table 1 summarizes AUROC and AUPRC results across all configurations, including a baseline random classifier, IntaRNA and *CUPID* models. For *CUPID* we compared results obtained with (Data-aug) and without (No-Data-aug) data augmentation, considering different pooling techniques, i.e. concatenation (concat), maximum (Max) and average (Avg) pooling.

Table 1. Comparison of AUROC and AUPRC across different experimental settings. Random baseline refers to the expected performance of the random classifiers. *CUPID* models are sorted in increasing order of both AUROC and AUPRC

Methods	**AUROC**	**AUPRC**
Random baseline	0.5	0.047
IntaRNA	0.544	0.055
CUPID :		
No-Data-aug	0.658	0.078
Data-aug-Max	0.810	0.147
Data-aug-concat	0.862	0.222
Data-aug-Avg	**0.919**	**0.364**

3.1 Random baselines

With a random classifier we can expect an AUROC = 0.5, while the estimated baseline AUPRC is:

$$\text{Baseline AUPRC} = \frac{N_+}{N_+ + N_-}$$

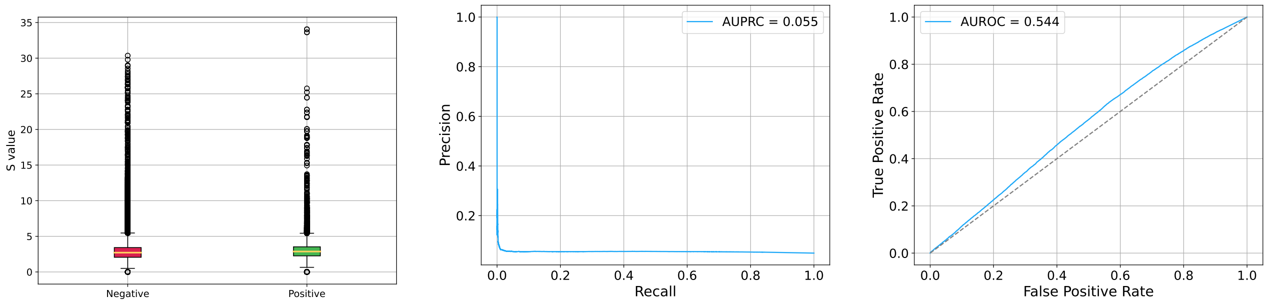
where N_+ is the number of positive samples, and N_- is the number of negative samples. Given the 1:20 ratio of positive to negative samples, the AUPRC baseline in the performed experiments is:

$$\text{Baseline AUPRC} = \frac{1}{1+20} \approx 0.0476.$$

Our top-performing model achieves an AUPRC of 0.364, corresponding to a *7.65-fold improvement* (0.364/0.0476). This margin quantifies the difficulty of the task: the extreme class imbalance renders precision–recall a stringent metric, and the observed gains indicate that the model extracts interaction-relevant information that is well above chance expectations.

3.2 IntaRNA results

Figure 6 reports IntaRNA performance on the augmented test set. In this setting, IntaRNA shows limited predictive power. Its scoring function relies on thermodynamic and accessibility components (e.g., hybridization energy and site accessibility), and in our experiments we used the default parameterization. Given the heterogeneity of ncRNA classes and sequence lengths in our benchmark, improved performance would likely require careful, class-specific calibration of both energy- and accessibility-related settings. Moreover, while IntaRNA is a general thermodynamics- and accessibility-based framework and is not inherently tied to a specific organism, it was originally introduced and most extensively evaluated in bacterial sRNA–mRNA interaction settings; consequently, when applied to heterogeneous ncRNA–ncRNA interactions (including long lncRNAs and diverse eukaryotic classes), its default parameterization may be suboptimal without additional tuning.



(a) (b) (c)

Figure 6. Results for IntaRNA results with augmented test set. (a) Distribution of predicted probabilities for negative and positive interactions; (b) AUPRC; (c) AUROC

3.3 *CUPID* results

Table 1 compares all *CUPID* configurations. We first evaluated a *CUPID* model without augmentation, using concatenation of average and max pooling. Fig. 7 shows the results obtained without data augmentation and with concatenated average-max pooling. The overall AUPRC results on the test set are relatively low (Fig. 7c), even if a certain learning is witnessed by the AUROC largely above 0.5 (Fig. 7f), and by the distribution of the predicted interaction probabilities for negative and positive examples (Fig. 7c), with probabilities for positives relatively higher with respect to negatives. Nevertheless, the relatively flat trend of the training loss reveals a certain difficulty of the model to learn the data. This is reflected also in the confusion matrix where most of negative examples (70%) are misclassified ((Fig. 7e) and in the degradation of the AUPRC performance between validation (Fig. 7a) and test (Fig. 7d) data. By looking at specific ncRNA interactions, for certain interaction types (e.g.snRNA-snoRNA) we obtained good results across the different metrics, but for several ncRNA interactions (e.g. miRNA-lncRNA, miRNA-miRNA, lncRNA-snoRNA) we achieved poor results, with AUPRC below 0.1 (Fig. 7g). Summarizing Fig. 7, shows that with this setting *CUPID* can provide a certain discrimination between positive and negative interactions (Fig. 7c), but its precision–recall and ROC curves indicate a limited separation between positive and negative examples (Fig. 7d,f).

Introducing data augmentation consistently improves performance (Table 1). Fig. 8 shows the results obtained with data augmentation and average pooling. The AUPRC is more than 4 times larger than without data augmentation (Fig. 8d and Table 1). Enlarging the size of training data by data augmentation allows the model to better learn the training data, as witnessed by the training loss that continues to decrease across epochs (Fig. 8b). This results in a clear separation between the scores predicted for positive and negative examples – note that the probabilities predicted for negatives are compressed toward zero while for most positives are largely above 0.7 (even if with several outliers for both positive and negative examples, Fig. 8b). The confusion matrix also confirms that the model with augmented data can better predict negative examples (Fig. 8e); AUPRC (Fig. 8d) significantly improves, and AUROC is larger than 0.9 (Fig. 8f). Analyzing results for each specific ncRNA interaction, we can observe a significant improvement across all the considered metrics, with AUROC in most cases larger than 0.9, except for circRNA-miRNA, miRNA-scRNA, miRNA-snRNA and miRNA-scaRNA (even if for these two last ncRNA interactions values are close to 0.9 (Fig. 8g).

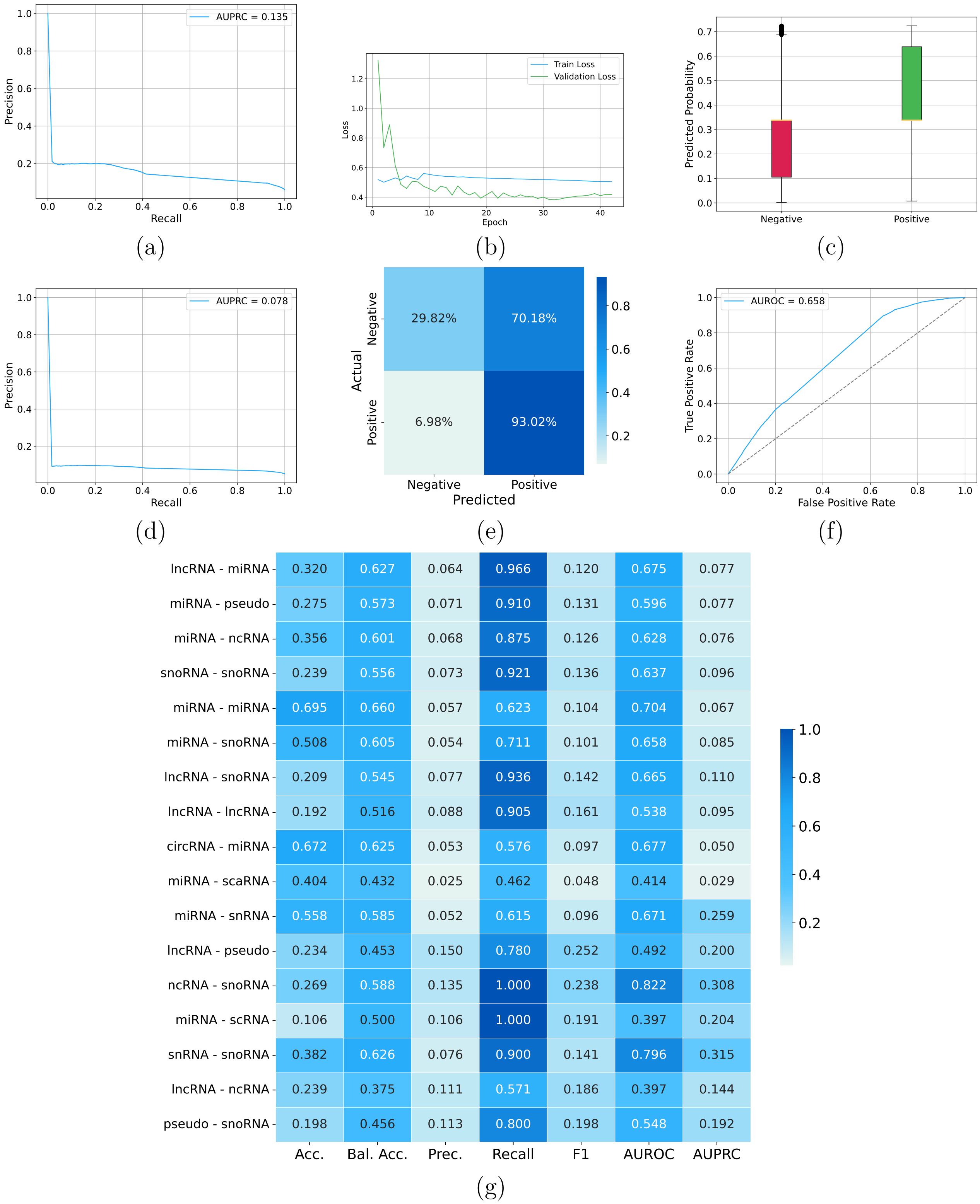
These results confirm that data augmentation is crucial to improve results for two main reasons: at first the model has training data enough to better generalize; second, improves generalization leveraging molecule order and orientation, two symmetries that are not guaranteed to be learned from limited training data. Augmentation effectively enforces these invariances, reducing overfitting to sequence presentation and mitigating the scarcity of positive examples.

Pooling strategy has a direct impact on the stability of the molecule-level embedding. Average pooling—yielding a smoothed representation over the full sequence—achieves the highest AUROC and AUPRC (Fig. 8) compared to max pooling (Suppl. Fig. S2) and concatenation pooling (Suppl. Fig. S3). This indicates that interaction-relevant information is not confined to a small set of token embeddings but arises from distributed features along the sequence. Max pooling, in contrast, appears sensitive to local outliers and overly compresses positional variability, while concatenation does not provide additional benefits once augmentation is introduced. The results suggest that, for ncRNA interaction prediction, the aggregate signal across nucleotides is more informative than isolated high-activation sites.

4 DISCUSSION

The results shown in this work demonstrate that RNA sequence-only inference can recover interaction signals across diverse ncRNA classes. The best-performing configuration reaches AUROC values above 0.9 on the test set, despite operating without structural, evolutionary, or thermodynamic information. This suggests that pretrained RNA language models encode latent features associated with intermolecular recognition. These features may reflect statistical regularities of pairing propensities and local compositional biases captured during pretraining, even in the absence of explicit structural supervision.

From a methodological standpoint, two contributions appear essential. First, the augmentation scheme addresses symmetries inherent to the problem. Because interacting RNAs can be presented in either order, and because sequence orientation can vary, enforcing invariance to these transformations is critical for robust generalization. Data augmentation also increases the number of examples available for training, thus improving the generalization performance of the model. Second, average pooling provides stable embeddings for ncRNA sequences. For molecules such as lncRNAs——whose



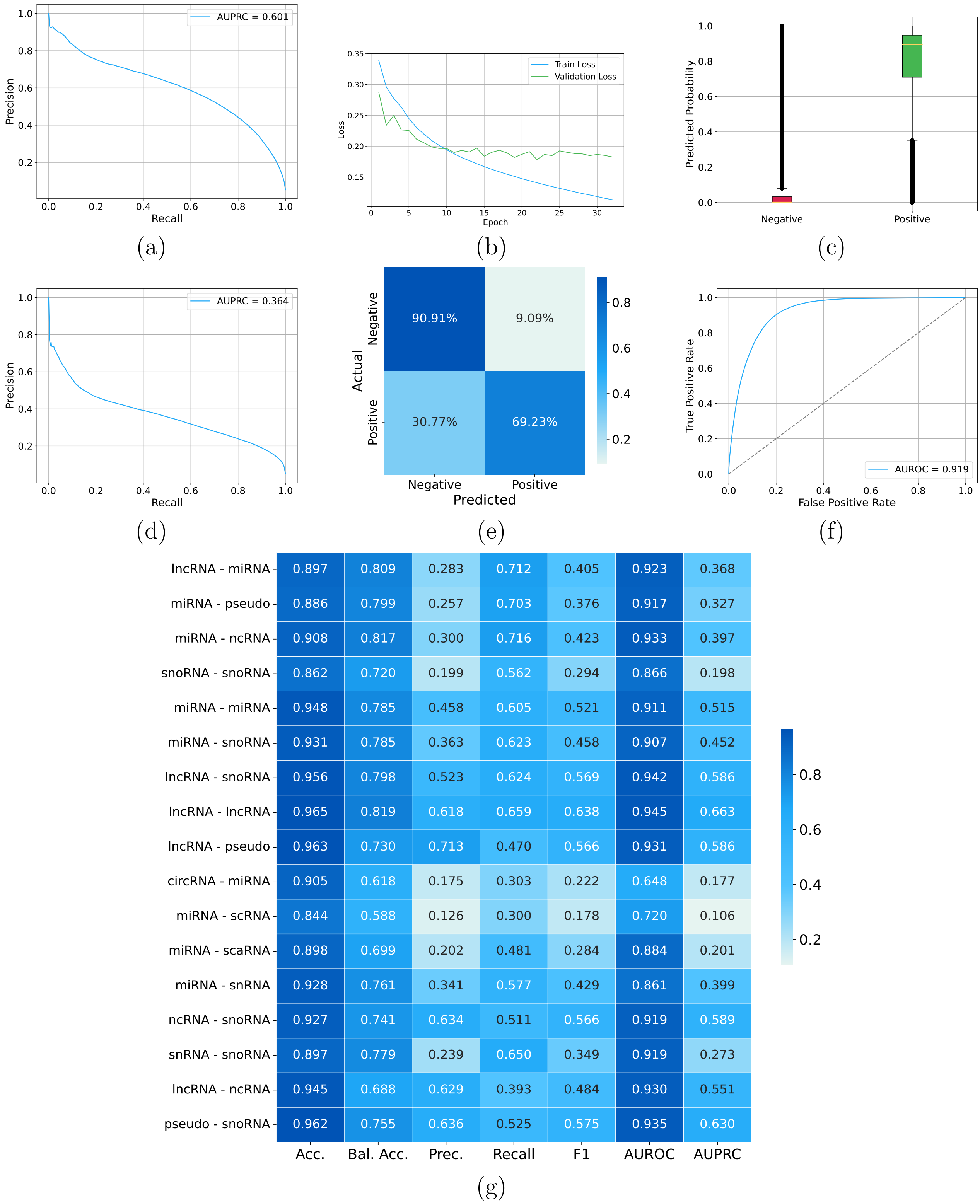
(a) (b) (c)

(d) (e) (f)

	Acc.	Bal. Acc.	Prec.	Recall	F1	AUROC	AUPRC
lncRNA - miRNA	0.320	0.627	0.064	0.966	0.120	0.675	0.077
miRNA - pseudo	0.275	0.573	0.071	0.910	0.131	0.596	0.077
miRNA - ncRNA	0.356	0.601	0.068	0.875	0.126	0.628	0.076
snoRNA - snoRNA	0.239	0.556	0.073	0.921	0.136	0.637	0.096
miRNA - miRNA	0.695	0.660	0.057	0.623	0.104	0.704	0.067
miRNA - snoRNA	0.508	0.605	0.054	0.711	0.101	0.658	0.085
lncRNA - snoRNA	0.209	0.545	0.077	0.936	0.142	0.665	0.110
lncRNA - lncRNA	0.192	0.516	0.088	0.905	0.161	0.538	0.095
circRNA - miRNA	0.672	0.625	0.053	0.576	0.097	0.677	0.050
miRNA - scaRNA	0.404	0.432	0.025	0.462	0.048	0.414	0.029
miRNA - snRNA	0.558	0.585	0.052	0.615	0.096	0.671	0.259
lncRNA - pseudo	0.234	0.453	0.150	0.780	0.252	0.492	0.200
ncRNA - snoRNA	0.269	0.588	0.135	1.000	0.238	0.822	0.308
miRNA - scRNA	0.106	0.500	0.106	1.000	0.191	0.397	0.204
snRNA - snoRNA	0.382	0.626	0.076	0.900	0.141	0.796	0.315
lncRNA - ncRNA	0.239	0.375	0.111	0.571	0.186	0.397	0.144
pseudo - snoRNA	0.198	0.456	0.113	0.800	0.198	0.548	0.192

(g)

Figure 7. *CUPID* results with concatenated pooling, and without using augmented data. (a) Overall precision recall curve on the validation set including all the type of ncRNA interactions; (b) Training and validation loss across epochs; (c) Distribution of the *CUPID* predicted probabilities on negative and positive examples on the test set; (d) Overall precision recall curve on the test set including all the type of ncRNA interactions; (e) Confusion matrix on the test set; (f) ROC curve on the test set including all the type of ncRNA interactions; (g) *CUPID* results on the test set across different types on ncRNA interactions (rows) for different types of metrics (columns).



(a) (b) (c)

(d) (e) (f)

(g)

Figure 8. *CUPID* results with average pooling and using augmented data. (a) Overall precision recall curve on the validation set including all the type of ncRNA interactions; (b) Training and validation loss across epochs; (c) Distribution of the *CUPID* predicted probabilities on negative and positive examples on the test set; (d) Overall precision recall curve on the test set including all the type of ncRNA interactions; (e) Confusion matrix on the test set; (f) ROC curve on the test set including all the type of ncRNA interactions; (g) *CUPID* results on the test set across different types on ncRNA interactions (rows) for different types of metrics (columns).

functional elements are dispersed and whose lengths vary over orders of magnitude——summarizing the full sequence avoids overemphasis on isolated positions and instead captures global contextual tendencies. Moreover, to our knowledge, *CUPID* is the first model able to predict a large set of ncRNA interactions, and in principle can be applied to predict any ncRNA interaction.

The limitations observed for IntaRNA highlight the difference between energy-based and representation-based approaches. Thermodynamic models rely on explicit structural motifs and accessibility assumptions, which may not generalize to long, structured, or poorly conserved ncRNAs. In contrast, *CUPID* does not attempt to reconstruct secondary structure but leverages contextual sequence statistics learned from large corpora. These complementary perspectives suggest potential synergies: coupling language-model embeddings with coarse structural predictions could refine the discrimination between spurious and functionally relevant pairing events.

Despite these promising results, we note that the resources used to train *CUPID* are limited in size and exhibits a strong imbalance across interaction types. Although our type-constrained negative sampling preserves the empirical distribution of interaction types, rare types remain challenging; they can yield higher-variance estimates and may prevent the model from learning robust type-specific patterns. Accordingly, we emphasize AUPRC in our per-type analyses, as it is generally more informative than AUROC under severe class imbalance. Future work will benefit from larger and more balanced interaction resources, and could further improve stability on underrepresented classes via targeted strategies such as class-aware reweighting, resampling, or cost-sensitive objectives.

As larger ncRNA catalogs become available through resources such as RNAcentral Sweeney et al. (2020), and as experimental protocols expand the coverage of ncRNA–ncRNA interactions, the training regime of models like *CUPID* can be scaled accordingly. Future developments may integrate longer receptive fields, explicit cross-attention between molecules, or joint fine-tuning on experimentally resolved interactomes. These extensions could help reveal constraints underlying ncRNA recognition and improve the resolution of regulatory maps in eukaryotic transcriptomes.

In addition, while our study focuses on a resource-efficient paradigm that leverages pretrained RNA language models with a lightweight interaction-specific prediction head, it would be interesting to complement our analysis with baselines that train a long-context Transformer from scratch. We did not include such a baseline here because, under the current supervision regime (approximately 10^5 interaction pairs after filtering), end-to-end training from random initialization may be difficult to optimize and may not yield generalizable representations. As larger and more diverse labeled interaction resources become available, systematic comparisons between pretrained and from-scratch Transformer encoders will become increasingly informative.

A similar consideration holds when considering studies substituting RNA-LM models with several Transformer-based nucleotide language models. While these models could, in principle, be considered as alternative backbones for RNA sequence embeddings (e.g., models pretrained predominantly on DNA such as Nucleotide Transformer, which has been reported to transfer RNA-related signals (Dalla-Torre et al., 2025)), we selected GenerRNA because it is pretrained specifically on RNA sequences, provides a long-context representation and it is expected to better capture RNA-class-specific features. We therefore expect RNA-specialized pretraining to yield representations that are more directly tailored to RNA sequence regularities than more generic DNA-pretrained alternatives, even when the latter can capture some RNA features. In this work, we focused on characterizing the proposed interaction-prediction pipeline using a single RNA-specialized backbone, including ablations on augmentation and pooling. As larger and more diverse interaction resources become available, it will be important to benchmark GenerRNA in a zero-shot setting against more general nucleotide Transformers, and to evaluate both backbones also after task-specific fine-tuning.

In summary, the results show that *CUPID* provides a scalable sequence-based framework for ncRNA–ncRNA interaction prediction, achieving AUROC larger than 0.9 for several types on ncRNA interactions. Its performance, robustness to class heterogeneity, and limited dependence on domain-specific priors make it suitable for large-scale in silico screening and for guiding targeted experimental profiling of ncRNA regulatory networks.

CONFLICT OF INTEREST STATEMENT

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

AUTHOR CONTRIBUTIONS

M.N. E.C. and G.V. designed the overall work; M.N. and F.S. developed and implemented the AI and computational methods; M.N. conceived and executed the experiments; M.N. F.S. E.C. and G.V. analyzed the results; G.V. M.N. and E.C. drafted and wrote the paper; G.V. and E.C. supervised the overall work; all the authors revised and approved the final manuscript.

FUNDING

This work was supported by National Center for Gene Therapy and Drugs Based on RNA Technology—MUR (Project no. CN_00000041) funded by NextGeneration EU program and by the National Plan for NRRP Complementary Investments (PNC) in the call for the funding of research initiatives for technologies and innovative trajectories in the health—project n. PNC0000003—AdvaNced Technologies for Human-centrEd Medicine (project acronym: ANTHEM).

Computational resources at CINECA for this work have been funded by UNITECH INDACO, which is an HPC project at the state University of Milan.

DATA AVAILABILITY STATEMENT

The data, the *CUPID* code, and the scripts to reproduce the experiments and tutorials are available from GitHub: `https://github.com/AnacletoLAB/ncRNA-CUPID`.

REFERENCES

Aleksander, S., Balhoff, J., Carbon, S., et al. (2023). The gene ontology knowledgebase in 2023. *Genetics* 224, iyad031. doi:10.1093/genetics/iyad031

Ali, S., Peffers, M., Ormseth, M., et al. (2021). The non-coding RNA interactome in joint health and disease. *Nat Rev Rheumatol* 17, 692–705. doi:10.1038/s41584-021-00687-y

Alipanahi, B., Delong, A., Weirauch, M. T., and Frey, B. J. (2015). Predicting the sequence specificities of dna-and rna-binding proteins by deep learning. *Nature Biotechnology* 33, 831–838

Cavalleri, E., Cabri, A., Soto-Gomez, M., Bonfitto, S., Perlasca, P., Gliozzo, J., et al. (2024). An ontology-based knowledge graph for representing interactions involving rna molecules. *Scientific Data* 11, 906

Cetin, S. and Sefer, E. (2025). A graphlet-based explanation generator for graph neural networks over biological datasets. *Current Bioinformatics* 20, 840–851

Chen, J., Hu, Z., Sun, S., Tan, Q., Wang, Y., Yu, Q., et al. (2022). Interpretable rna foundation model from unannotated data for highly accurate rna structure and function predictions. *arXiv preprint arXiv:2204.00300*

Dalla-Torre, H., Gonzalez, L., Mendoza-Revilla, J., et al. (2025). Nucleotide Transformer: building and evaluating robust foundation models for human genomics. *Nat Methods* 22, 287–297. doi:10.1038/s41592-024-02523-z

Engreitz, J. M., Sirokman, K., McDonel, P., Shishkin, A. A., Surka, C., Russell, P., et al. (2014). Rna-rna interactions enable specific targeting of noncoding rnas to nascent pre-mrnas and chromatin sites. *Cell* 159, 188–199

Fabbri, M., Girnita, L., Varani, G., and Calin, G. A. (2019). Decrypting noncoding rna interactions, structures, and functional networks. *Genome research* 29, 1377–1388

Gargano, M., Matentzoglu, N., Coleman, B., et al. (2023). The Human Phenotype Ontology in 2024: phenotypes around the world. *Nucleic Acids Research* 52, D1333–D1346. doi:10.1093/nar/gkad1005

Gliozzo, J., Soto Gomez, M. A., Bonometti, A., et al. (2025). miss-SNF: a multimodal patient similarity network integration approach to handle completely missing data sources. *Bioinformatics* 41, btaf150. doi:10.1093/bioinformatics/btaf150

Li, Z., Li, K., Lian, X., and Li, J. (2025). Lncrna-mirna interaction prediction based on multi-source heterogeneous graph neural network and multi-level attention mechanism. *International Journal of Biological Macromolecules* 319, 145614. doi:https://doi.org/10.1016/j.ijbiomac.2025.145614

Lorenzi, L., Chiu, H.-S., Cobos, A., et al. (2021). The rna atlas expands the catalog of human non-coding rnas. *Nature Biotechnology* 39, 1453–1465. doi:https://doi.org/10.1038/s41587-021-00936-1

Mann, M., Wright, P. R., and Backofen, R. (2017). Intarna 2.0: enhanced and customizable prediction of rna–rna interactions. *Nucleic acids research* 45, W435–W439

Matthews, B. W. (1975). Comparison of the predicted and observed secondary structure of t4 phage lysozyme. *Biochimica et Biophysica Acta (BBA)-Protein Structure* 405, 442–451

Nicolini, M., Saitto, E., Jimenez-Franco, R., et al. (2025a). Fine-tuning of conditional Transformers improves in silico enzyme prediction and generation. *Computational and Structural Biotechnology Journal* 27, 1318–1334. doi:10.1016/j.csbj.2025.03.037

Nicolini, M., Stacchietti, F., Cano, C., Casiraghi, E., and G, V. (2025b). A transformer-based model to predict micro rna interactions. In *18th International Work-Conference on Artificial Neural Networks, IWANN 2025*. vol. 16008 of *Lecture Notes in Computer Science*. doi:10.1007/978-3-032-02725-2_8

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I., et al. (2019). Language models are unsupervised multitask learners. *OpenAI blog* 1, 9

Sapoval, N., Aghazadeh, A., Nute, M., et al. (2022). Current progress and open challenges for applying deep learning across the biosciences. *Nat Commun* 13. doi:http://doi.org/10.1038/s41467-022-29268-7

Sefer, E. (2025). Drgat: Predicting drug responses via diffusion-based graph attention network. *Journal of Computational Biology* 32, 330–350

Sennrich, R., Haddow, B., and Birch, A. (2016). Neural Machine Translation of Rare Words with Subword Units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, eds. K. Erk and N. A. Smith (Berlin, Germany: Association for Computational Linguistics), 1715–1725. doi:10.18653/v1/P16-1162

Shen, T., Hu, Z., Sun, S., Liu, D., Wong, F., Wang, J., et al. (2024). Accurate RNA 3D structure prediction using a language model-based deep learning approach. *Nature Methods* , 1–12

Sun, S.-L., Jiang, Y.-Y., Yang, J.-P., Xiu, Y.-H., Bilal, A., and Long, H.-X. (2025). Predicting noncoding rna and disease associations using multigraph contrastive learning. *Scientific Reports* 15, 230

Sweeney, B., Petrov, A., Ribas, C., et al. (2020). RNAcentral 2021: secondary structure integration, improved sequence search and new member databases. *Nucleic Acids Research* 49, D212–D220. doi:http://dx.doi.org/10.1093/nar/gkaa921

Tian, X., Shen, L., Wang, Z., Zhou, L., and Peng, L. (2021). A novel lncrna–protein interaction prediction method based on deep forest with cascade forest structure. *Scientific reports* 11, 18881

Torgano, F., Soto-Gomez, M., Zignani, M., Gliozzo, J., Cavalleri, E., Mesiti, M., et al. (2025). Rna knowledge-graph analysis through homogeneous embedding methods. *Bioinformatics Advances* 5, vbaf109. doi:10.1093/bioadv/vbaf109

Umu, S. U. and Gardner, P. P. (2017). A comprehensive benchmark of rna–rna interaction prediction tools for all domains of life. *Bioinformatics* 33, 988–996

Valentini, G., Malchiodi, D., Gliozzo, J., Mesiti, M., Soto-Gomez, M., Cabri, A., et al. (2023). The promises of large language models for protein design and modeling. *Frontiers in Bioinformatics* 3

Vasilevsky, N., Toro, S., Matentzoglu, N., et al. (2025). Mondo: Integrating Disease Terminology Across Communities. *Genetics* , iyaf215doi:10.1093/genetics/iyaf215

Wang, W., Zhang, L., Sun, J., Zhao, Q., and Shuai, J. (2022). Predicting the potential human lncrna–mirna interactions based on graph convolution network with conditional random field. *Briefings in Bioinformatics* 23, bbac463. doi:10.1093/bib/bbac463

Wei, J., Chen, S., Zong, L., Gao, X., and Li, Y. (2022). Protein–rna interaction prediction with deep learning: structure matters. *Briefings in Bioinformatics* 23, bbab540. doi:10.1093/bib/bbab540

Yang, T., He, Y., and Wang, Y. (2025). Introducing tec-lncmir for prediction of lncrna-mirna interactions through deep learning of rna sequences. *Briefings in Bioinformatics* 26, bbaf046. doi:10.1093/bib/bbaf046

Yu, H., Yang, H., Sun, W., et al. (2024). An interpretable rna foundation model for exploring functional rna motifs in plants. *Nat Mach Intell* 6, 1616–1625. doi:10.1038/s42256-024-00946-z

Yu, X., Jiang, L., Jin, S., Zeng, X., and Liu, X. (2022). premli: a pre-trained method to uncover microrna–lncrna potential interactions. *Briefings in Bioinformatics* 23, bbab470. doi:10.1093/bib/bbab470

Zhao, Y., Oono, K., Takizawa, H., and Kotera, M. (2024). GenerRNA: A generative pre-trained language model for de novo RNA design. *PLoS One* 19, e0310814