TUM Data Innovation Lab

Munich Data Science Institute (MDSI)

Technical University of Munich

&

Helmholtz Munich, Computational RNA Biology Lab

Final report of project:

Predicting RNA-small molecule interactions with deep learning for RNA-targeted drug discovery

Authors	Maximilian Greißl, Jed Guzelkabaagac, Leonhard Kraft, Lars Pennig, Ziqing Zhao
Mentors	Giulia Cantini, Tobias Bernecker, Samuele Firmani
Project lead	Dr. Ricardo Acevedo Cabra (MDSI)
Supervisor	Prof. Dr. Massimo Fornasier (MDSI)

Apr 2025

Acknowledgements

We would like to thank our mentors for their invaluable guidance throughout this research project. Our thanks also go to the Munich Data Science Institute (MDSI) for handling all organizational aspects around the TUM Data Innovation Lab and thus making this project possible. This work was enabled by computational resources from the de.NBI Cloud, part of the German Network for Bioinformatics Infrastructure (de.NBI).

Abstract

RNA plays a crucial role in many biological processes, making it a compelling target for therapeutic intervention. While data-driven drug discovery has traditionally focused on proteins, recent advances highlight RNA as a viable and promising target for small-molecule therapeutics. However, due to RNA's structural complexity and the high cost of experimental validation, identifying selective and potent RNA-binding compounds remains a significant challenge. Deep learning approaches offer a promising avenue to accelerate drug discovery by predicting RNA-small molecule interactions directly from sequence data, bypassing the need for experimentally determined structural information. In this study, we systematically evaluate graph- and sequence-based deep learning models for predicting RNA-small molecule binding affinity, leveraging SMILES representations for molecules and nucleotide sequences for RNA. To address the inherent data scarcity, we include pre-trained molecular and RNA foundation models in our evaluations. Our results demonstrate that while sequence-based deep learning models can outperform prior work on specific RNA families, their effectiveness is constrained by the limited availability of high-quality training data. This limitation gives an advantage to alternative approaches that incorporate structural information or employ classical machine learning techniques that are less prone to overfitting. Our findings underscore the need for more extensive RNA-small molecule binding datasets and hybrid modeling strategies to enhance predictive performance, ultimately advancing RNA-targeted drug discovery.

Contents

Abstract ii

1 Introduction 1
1.1 Background . . . 1
1.1.1 RNA . . . 1
1.1.2 Small Molecules . . . 2
1.1.3 RNA-Ligand Binding . . . 2
1.2 Problem Definition and Project Scope . . . 3

2 Related Work 4
2.1 Early Screening Approaches . . . 4
2.2 Machine Learning Approaches . . . 4

3 Data 6
3.1 Data Acquisition . . . 6
3.2 Data Cleaning . . . 6
3.3 Statistical Data Exploration . . . 7
3.4 Data Splits . . . 9
3.5 Pocket-Interaction Dataset . . . 10

4 Methods 11
4.1 Molecule Encoders . . . 11
4.2 RNA Encoders . . . 13
4.3 Combination Layers . . . 14
4.4 Parameter-Efficient Finetuning via LoRA . . . 15
4.5 Pretraining with Pocket Information . . . 16
4.6 Inference for Classification Tasks . . . 16

5 Results 18
5.1 Reproducing RSAPred . . . 18
5.2 Regression Results with Deep Learning Methods . . . 19
5.2.1 Interpolation Split . . . 20
5.2.2 Extrapolation Split . . . 20
5.3 Results with Classification Inference . . . 21
5.4 LoRA-Finetuning . . . 23
5.5 Pocket Pretraining . . . 23

6 Conclusion 25

Appendix 30
A Use Cases . . . 30
B Performance Metrics . . . 31
C Supplementary Tables and Figures . . . 32

1 Introduction

RNA plays a crucial role in various biological processes by interacting with small molecules (ligands) that regulate gene expression, catalysis, and signal transduction [10, 13, 41]. In this work, we propose deep learning methods that leverage machine-readable representations of RNA and small molecules to predict RNA–ligand binding affinities. Figure 1 visualizes a ligand docked to an RNA molecule, illustrating how predictions are derived from RNA sequence and SMILES representations.

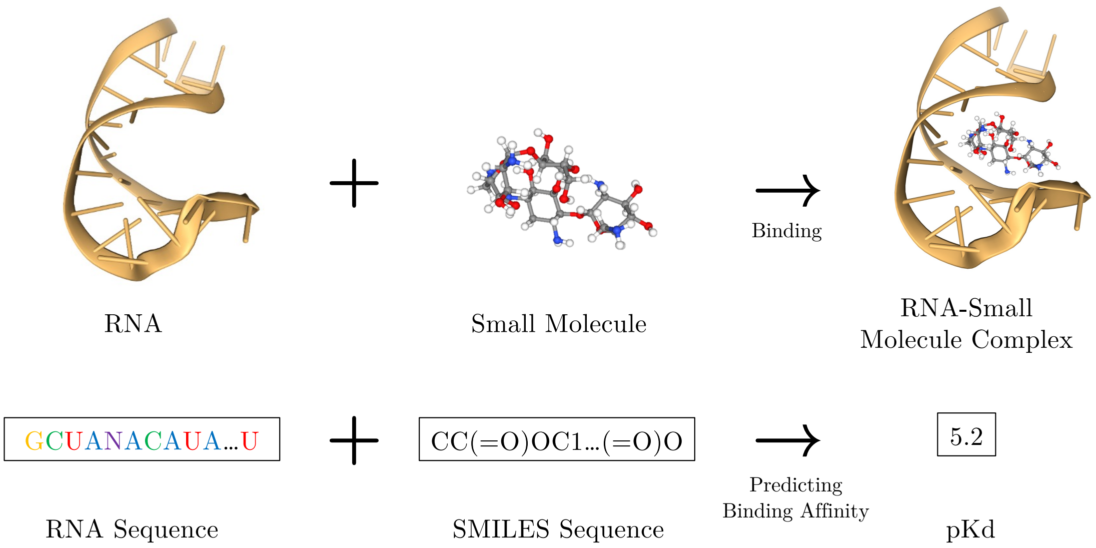


Figure 1: Illustration of RNA-small molecule binding.

1.1 Background

In this section, we introduce the key biological and chemical concepts that form the foundation of this research, namely RNA, small molecules, and RNA–ligand binding. To model these interactions computationally, we also describe how these entities are encoded into machine-readable formats.

1.1.1 RNA

Ribonucleic acid (RNA) is a fundamental biomolecule composed of four nucleotides: adenine (`A`), cytosine (`C`), guanine (`G`), and uracil (`U`), which together define its primary structure. It can fold into secondary structures via self-complementary sequences and these can further assemble into tertiary and quaternary conformations, thereby enabling diverse molecular functions [17, 21, 40]. RNAs can be classified into coding RNAs and non-coding RNAs, with coding RNAs serving as templates for protein synthesis, and non-coding RNAs fulfilling regulatory and structural roles [25].
Sequence-based formats are the backbone of computational RNA biology, representing RNA primary sequences as *strings of nucleotide symbols*. For example, the RNA sequence `AUGCGUA` corresponds to a single-stranded chain of seven nucleotides. However, these representations lack secondary or tertiary structural information.

1.1.2 Small Molecules

Small molecules (SMs) are organic compounds with low molecular weight, frequently used in drug development due to favorable absorption, distribution, and cost-effective synthesis [29].

The *Simplified Molecular-Input Line-Entry System* (SMILES) is a widely used notation for representing the chemical structure of molecules. In SMILES notation, all *atoms* are represented by their atomic symbols. Any unfulfilled valency of an atom is assumed to be hydrogen. Single, double, and triple *bonds* are represented by the symbols `-`, `=` and `#`, respectively. For simplicity, single bonds are often omitted. *Branches* in a molecule are represented using parentheses `()`, while *cyclic structures* are broken at an arbitrary point of the ring and denoted by assigning numbers to the atoms that close the ring, e.g., `C1` for a closing carbon atom of the first ring structure [48]. The molecule *aspirin* (see Figure 2) provides an excellent example, as it contains both branches and a cycle. The following string describes its chemical structure:

$$\texttt{CC(=O)O}\underbrace{\texttt{C1=CC=CC=C1}}_{\text{cyclic structure}}\underbrace{\texttt{C(=O)O}}_{\text{branch}}$$

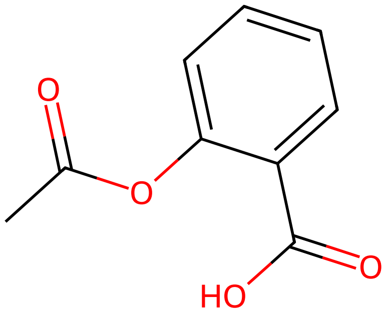


Figure 2: 2D Chemical structure of aspirin.

SMILES strings are widely used due to simplicity although they may be non-unique and do not encode 3D structural information.

1.1.3 RNA-Ligand Binding

RNA–ligand binding events occur through a variety of non-covalent interactions[1] – such as hydrogen bonding, electrostatic interactions, and van der Waals forces – that stabilize the RNA–ligand complex.

To quantify the affinity between ligand (L) and RNA (R), the *dissociation constant* K_d is commonly used. The binding process is represented by the equilibrium:

$$\mathrm{L} + \mathrm{R} \rightleftharpoons \mathrm{LR}.$$

[1] A non-covalent interaction differs from a covalent bond in that it does not involve the sharing of electrons.

And the corresponding K_d is defined as:

$$K_d = \frac{[\mathrm{L}][\mathrm{R}]}{[\mathrm{LR}]},$$

where [R], [L], and [LR] represent the molar concentrations of the RNA, ligand, and RNA–ligand complex, respectively.
Expressed in the unit of molarity (M), K_d is the ligand concentration [L] at which half of the RNAs are occupied. That is, at $[\mathrm{L}] = K_d$, the concentration of ligand-bound RNA [LR] equals the concentration of RNA with no ligand bound [R]. Lower K_d values (e.g., $K_d \approx 10^{-12}$ M) indicate tighter binding, whereas higher K_d values (e.g., $K_d \approx 10^{-3}$ M or higher) indicate weaker binding.
The binding affinity is often expressed in the form of the negative logarithm of K_d, represented as pK_d :

$$pK_d = -\log_{10}(K_d).$$

The negative logarithm compresses the wide range of K_d into a more manageable and interpretable scale.

1.2 Problem Definition and Project Scope

RNA-targeted therapies hold promise, yet identifying RNA-binding small molecules remains challenging due to limited structural data. Thus, computational methods that can predict binding affinities without relying solely on experimentally resolved RNA structures are needed.
This project proposes applying advanced deep-learning techniques to RNA and chemical compound representations to create a reliable model capable of predicting RNA-small molecule interactions based on sequence data. The model is intended to enhance virtual screening workflows and offer insights into compounds with strong RNA affinity.
The specific objectives are:

- **Develop abstract representations of RNA and small molecules** using RNA and chemical language models (LMs) and/or geometric deep learning approaches.
- **Combine these representations in a unified deep learning model** trained to predict RNA-small molecule binding affinities using publicly available datasets.
- **Analyse the model's outputs** and compare it to standard machine learning model baselines, such as RSAPred [24].

By addressing these objectives, the project aims to accelerate the discovery of effective RNA-targeting small molecules and contribute a robust computational tool to the field.

2 Related Work

As experimental screening of RNA-binding small molecules is costly, researchers increasingly turn to *in silico* methods to streamline discovery. In the following sections, we first examine the traditional techniques that laid the groundwork for RNA-focused drug discovery, highlighting their strengths and limitations. We then explore how modern ML approaches build upon and address these shortcomings, ultimately offering more scalable and predictive frameworks for identifying RNA-ligand pairings. We also review potential data augmentation strategies reported in the literature.

2.1 Early Screening Approaches

Early approaches, such as Inforna and RNALigands, first analyze RNA sequences to identify key secondary structure motifs [6, 42]. These methods then compare the identified motifs against a curated database of known RNA–small molecule interactions to generate a "molecular fingerprint" highlighting promising candidate binding sites and small molecule leads. The selected leads are subsequently validated through experiments to confirm their RNA-binding ability and biological activity. However, a major limitation of these approaches is their heavy reliance on in-house experimental databases; as a result, their performance often suffers when applied to new or unseen RNA queries, limiting their overall generalizability [10].
Alternatively, *rDock* and *RLDOCK* simulate RNA–ligand binding by positioning candidate molecules into the binding pockets of RNA [38, 43]. Although docking provides a more direct assessment of binding by estimating interaction energies, its utility is inherently constrained by the reliance on high-resolution three-dimensional RNA structures, which are available for only a limited subset of RNAs. Additionally, to remain feasible at scale, they often under-sample conformational states, ignore important interaction terms, and approximate potentials that are critical for chemical accuracy [5].

2.2 Machine Learning Approaches

Machine learning methods have become increasingly prominent in RNA–small molecule research. They can be grouped loosely into the two main categories of structure-based models and sequence-based models, which we detail in the following subsections.

Structure-Based Models. Structure-based approaches leverage three-dimensional structural information of RNA to identify putative ligand-binding pockets and assess RNA–ligand compatibility. Recent advances highlight both the potential and constraints of these methods. For instance, *RLaffinity* addresses binding affinity regression by combining contrastive self-supervised learning on unlabeled RNA–ligand complexes with a 3D convolutional neural network trained on a small, labeled dataset of 144 RNA–ligand pairs from PDBBind [2]. This voxelized atomic representation can capture detailed structural interactions. Still, the requirement for experimentally solved RNA coordinates and binding affinity measurements restricts the available training examples and potentially limits model generalizability. By contrast, *RNAmigos2* pursues a binary or ranking objective through synthetic or docking-derived data and a coarse-grained graph representation of

RNA, thereby achieving high-throughput screening but still depending on the presence of 3D structural information [8].
A key drawback of these structure-based methodologies is their heavy reliance on experimentally determined RNA structures. This limitation restricts the subset of RNAs that can be effectively targeted and underscores the practical challenges inherent in purely structure-driven approaches. Although RLaffinity and RNAmigos2 differ in their training objectives and data sources, both ultimately depend on structural data to learn expressive encodings of RNA–small molecule interactions.

Sequence-Based Models. By circumventing the need for explicit RNA 3D structures, sequence-based methods aim to directly infer binding propensities from RNA sequence information. In doing so, they mitigate the principal bottleneck of structure-based workflows, namely the scarcity of experimentally resolved RNA complexes and the computational overhead of accurate docking. Instead, these approaches harness sequence-level patterns, ranging from raw nucleotides to higher-order sequence features, to predict small molecule binding.
A purely sequence-based method, *RSAPred*, leverages multivariate linear regression to predict binding affinities between RNA sequences and small molecules [24]. The authors curated 1,524 experimentally validated RNA–ligand pairs from the R-SIM repository [23], using forward feature selection to identify informative sequence-based and molecular descriptors. While strong performance was reported in both cross-validation and external tests, our later investigation (see Section 5.1) revealed discrepancies between the paper's methodology and its publicly available code. More importantly, the authors' feature selection appeared to optimize on the entire dataset, leading to inflated results in cross-validation and leave-one-out trials.
Alternatively, *RNAsmol* applies data perturbation and augmentation to achieve robust binary classification of RNA-small molecule interactions [28]. The model encodes RNA sequences with multi-view CNNs and small molecules with a graph diffusion convolution network and then fuses these embeddings via bilinear attention. Although RNAsmol demonstrates strong results even under challenging *cold* conditions, where the test sets contain RNAs or ligands not seen in training, its classification focus contrasts with our aim of predicting continuous binding affinities. Nevertheless, certain aspects of RNAsmol such as its data augmentation strategies and feature extraction modules – may inform future regression-based frameworks.

Data Augmentation Strategies. Given the limited availability of experimentally validated RNA-ligand pairs, several studies have explored data augmentation techniques. *RNAsmol* [28] employs perturbation strategies – such as random RNA sequence shuffling, selecting small molecules with high fingerprint similarities, and network-based sampling – to generate additional samples for binding classification tasks. However, these approaches are primarily designed for binary classification rather than regression. Assigning appropriate negative affinity values to synthetic examples remains an unresolved challenge. Similarly, methods like *RLaffinity* [2] generate multiple RNA-ligand conformations from 3D structural databases, yet they require detailed structural information that is not universally available.

3 Data

This section presents the data sources, cleaning steps as well as data splits used.

3.1 Data Acquisition

To predict the binding affinity between RNA and small molecules based on sequence data, we use the dataset provided by Krishnan, Roy, and Gromiha [24]. The dataset has been composed from the R-SIM (**R**NA-**S**mall molecule **I**nteraction **M**iner) database. R-SIM comprises binding affinity information for 2,501 experimentally verified RNA-small molecule complexes [23]. The RNA-small molecule pairs are compiled from 216 articles published between 1977 and 2022.

To further refine the dataset, the authors removed (i) observations in which the RNA sequences did not belong to one of the six subtypes and (ii) entries with missing data. The final curated dataset comprises 1,439 RNA-small molecule pairs spanning these six subtypes [24].

Although relatively small, datasets of size ≈ 1,000 to 2,000 observations are state-of-the-art for data-driven RNA-ligand interaction prediction. For instance, Yazdani et al. [50] recently trained *classification* models on 2,373 RNA-binding molecules – the largest public, experimentally derived library of its kind (see Section 4.6). This reflects the scarcity of experimentally validated RNA-ligand binding data compared to protein-ligand resources.

3.2 Data Cleaning

The provided dataset was found to have several limitations that could hinder the generalization of a model trained on it and complicate further processing. We identified and addressed three main problems.

As described earlier, every row in our data describes one RNA-small molecule interaction. Each RNA is characterized by its sequence, name, and RNA ID; every small molecule is characterized by its SMILES string, name, and molecule ID. We noticed, however, that in 34 cases, the RNA IDs and molecule IDs, as well as names, were not unique for the same RNA sequence or molecule SMILES string. The same RNA sequence could be assigned up to three names and IDs (see Table 1). We resolved this issue by giving each RNA sequence and SMILES string a unique name and ID based on the first one appearing in the dataset. This reduced the number of unique RNAs in our dataset from 341 to 295 and the number of unique molecules from 792 to 759.

Furthermore, the same RNA sequence was assigned to two families across multiple samples in four cases. Because our data split relies on a stratified split between families, we decided to clearly assign each sequence to one family. As the family Aptamer was always contained as one of the possible families, we assigned all these samples to the Aptamer family.

Some RNA sequence - SMILES string combinations were non-unique and appeared multiple times in the database with slightly different associated pK_d values. Upon further investigation, we hypothesize that in some cases, this discrepancy may arise from different literature sources testing the same RNA-small molecule combinations under different experimental conditions. However, these conditions were unavailable in a standardized format and entirely absent for most. In one instance, only the differing RNA names

Feature Name	Description
Target_RNA_sequence	`GUGCAGGUAGUGAUAUGUGCAUCUACUGCAC`
SMILES	`NC(=[NH2+])C1=CC=C(NC2=CC=C(C=C2)C2=CC3=CC=C(C=C3N2)C(N)=[NH2+])C=C1`
Target_RNA_name	miR-17 Dicer site, miR-17 G-bulge
Target_RNA_ID	Target_316, Target_317
pKd	6.921, 4.824

Table 1: Example of a non-unique RNA target: miR-17. The same RNA sequence and SMILES string were associated with different target names, IDs, and experimental pK_d values in the original dataset.

(”single-stranded”, ”duplex”) suggested that the RNA was tested in distinct 3D configurations, leading to variations in the pK_d values. As a model would not be able to distinguish samples associated with different pK_d values but the same RNA sequence and SMILES string, we decided to randomly retain one combination and discard the rest. This process reduced the total number of samples from 1,439 to 1,362.

3.3 Statistical Data Exploration

In this subsection, we examine statistics of the cleaned dataset. The RNA targets were stratified into six subtypes: Aptamers, miRNAs, Repeats, Ribosomal RNAs, Riboswitches, and Viral RNAs.

Figure 3 illustrates the distribution of RNA families within the dataset. Most entries belong to the Aptamers family, which constitutes 37.8% of the data. Smaller contributions come from the Repeats, Riboswitch, and miRNA families with $\approx 10\%$ share.

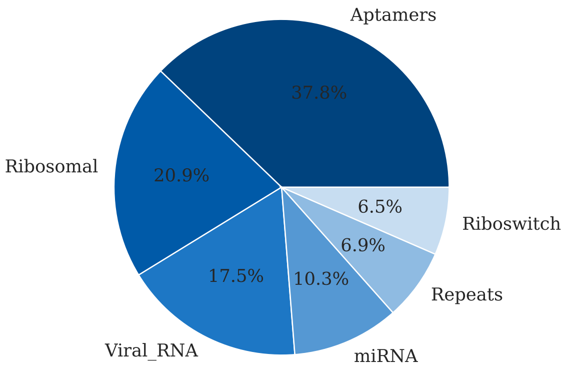


Figure 3: Distribution of RNA families.

The distribution of pK_d values across RNA families highlights variations in binding affinity (see Figure 4). The pK_d values range from -0.48 to 11.30 with a mean binding affinity of 5.39.

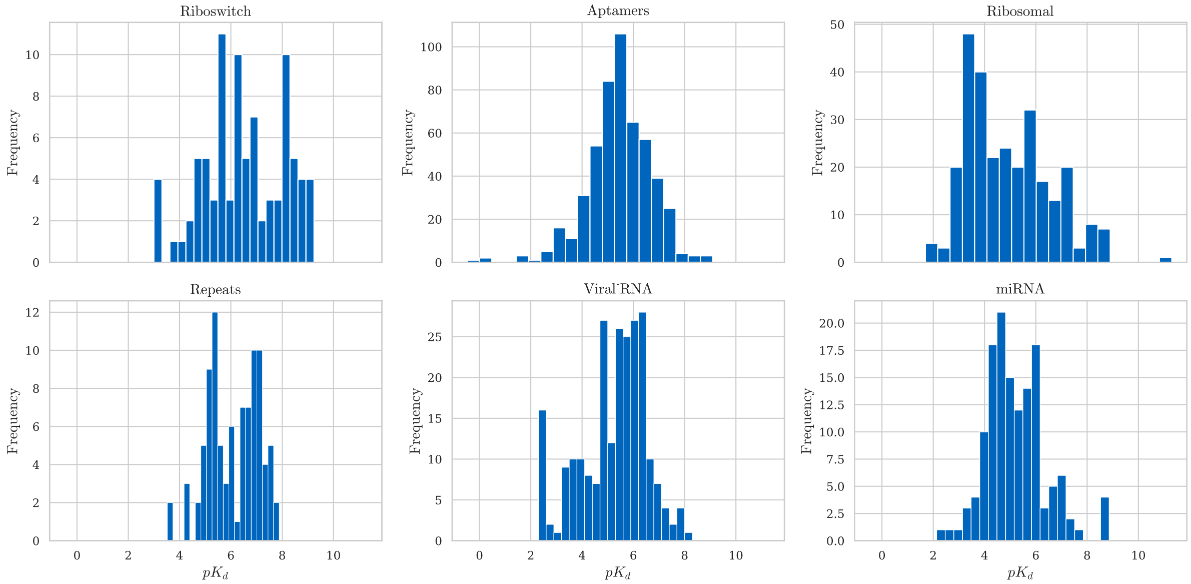


Figure 4: Distribution of pK_d per RNA family (adapted from [24]).

Drug-Likeness of Small Molecules in the Dataset. The sum of the atomic weights of all atoms in a molecule is called *molecular weight* and indicates the molecule's size and mass. The molecular weights of the SMs in the dataset mostly align with drug-like thresholds ($\leq$ 500 daltons), as seen in Figure 5. In particular, this distribution aligns with the well-established "Rule of 5" in drug discovery, which suggests that SMs within this range are more likely to exhibit favorable pharmacokinetics and bioavailability [27]. Overall, 53.04% of the SMs in the dataset fulfill the rule of thumb, i.e., no more than one violation of four criteria. Histograms and boxplots of the other three criteria can be found in the Appendix in Figures 14, 16 and 18 and Figures 13, 15, 17 and 19, respectively.

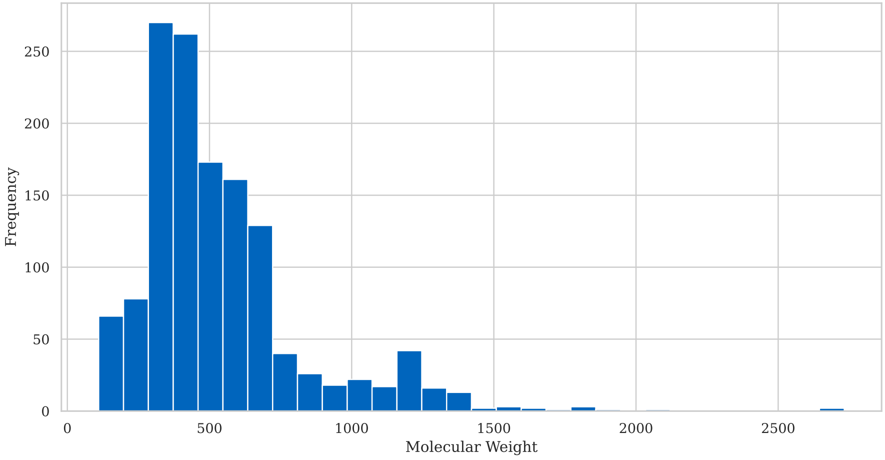


Figure 5: Distribution of molecular weights in daltons.

The dataset comprises 1,362 RNA-ligand complexes, of which 295 are unique RNAs and

759 are unique ligands. Most RNA sequences are tested with only a small number of SMs. In contrast, a few RNA sequences are tested with a large number of SMs, resulting in a significant overlap of candidates, as shown across different RNA families in Figure 6.

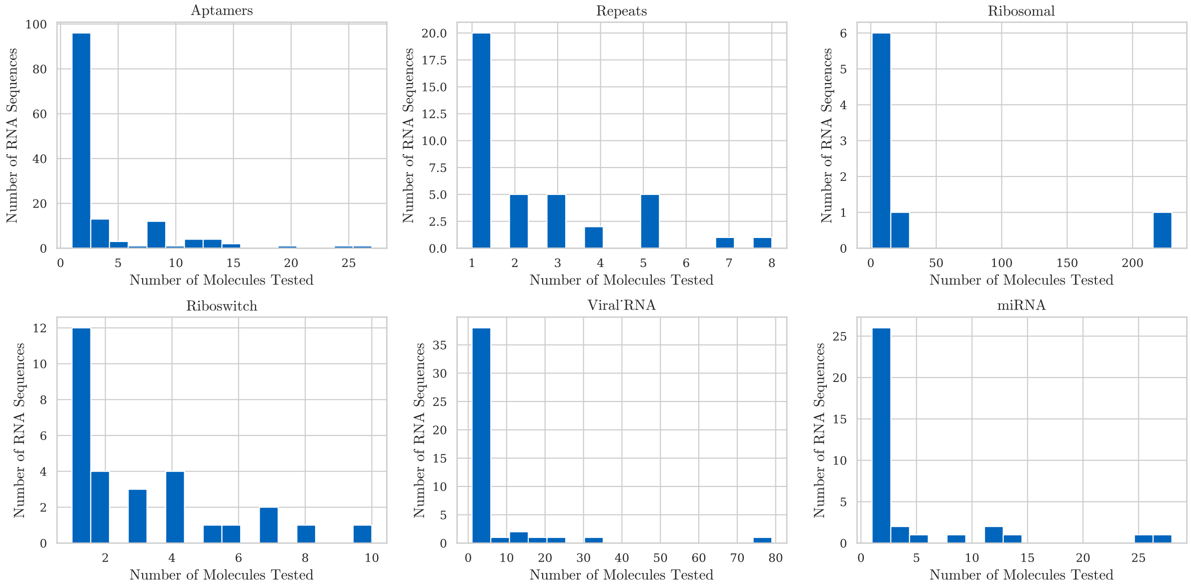


Figure 6: Candidate overlap per RNA family.

Overall, the dataset offers a valuable basis for developing predictive models for RNA-small molecule interactions. However, its limitations due to the small size are notable.

3.4 Data Splits

Although the dataset was published and utilized by Krishnan, Roy, and Gromiha [24], it does not include a predefined division into training and test sets. To enable a rigorous evaluation of our methods, we adopt three distinct approaches for splitting the dataset. In the first approach, we perform stratified 10-fold cross-validation, generating ten train-test splits where each split consists of 90% training samples and 10% test samples. We stratify the data based on RNA family and binned pK_d values to preserve the original data distribution across splits. Secondly, we perform a single test validation split, stratified based on pK_d and RNA family. However, we notice a big overlap in individual RNA sequences and molecules between the train and test sets. Thus, we call this split *interpolation split*. As assessing this split does not represent real-world inference, where usually neither RNA nor molecule were observed before, we develop a third approach designed to assess the generalization capabilities of our methods. Here, we again create a single train-test split, named *extrapolation split*, again allocating 90% of the samples to training and 10% to testing while ensuring stratification by RNA family. We enforce constraints such that no RNA sequence is shared between the training and test sets and minimize the overlap in small molecules. The resulting split statistics are presented in Table 2. We notice that this split still contains a similar pK_d distribution in both the train and test sets.

		RNA Family											
		Aptamers		**miRNA**		**Repeats**		**Ribosomal**		**Riboswitches**		**Viral RNA**	
Train	Samples	463	89.90%	126	90.00%	83	89.25%	249	87.37%	81	90.00%	211	88.28%
	RNAs	104	–	21	–	31	–	5	–	23	–	39	–
	SMs	175	–	86	–	37	–	241	–	57	–	164	–
	Samples w/ shared SMs	0	0.00%	0	0.00%	8	9.64%	1	0.40%	0	0.00%	0	0.00%
Test	Samples	52	10.10%	14	10.00%	10	10.75%	36	12.63%	9	10.00%	28	11.72%
	RNAs	35	–	14	–	8	–	3	–	6	–	6	–
	SMs	33	–	4	–	7	–	31	–	5	–	23	–
	Samples w/ shared SMs	0	0.00%	0	0.00%	4	40.00%	1	2.78%	0	0.00%	0	0.00%
	Shared unique SMs	0	0.00%	0	0.00%	1	2.33%	1	0.37%	0	0.00%	0	0.00%

Table 2: Data split statistics for the extrapolation split. No RNA is shared between the training and test split. The number of small molecules shared between the splits is minimal while ensuring RNA diversity in the test split.

3.5 Pocket-Interaction Dataset

Due to the limited size and lack of structural information in the R-SIM dataset, we constructed a pre-training dataset focused on binding pocket interactions.

Our dataset follows the methodology used in the RNA Geometric Library [30]. We downloaded 5,843 PDB entries containing at least one RNA and one ligand. Since some RNA molecules include polymeric ligands or linking monomers, classified as ligands but functionally belong to the RNA chain, we filtered out polymeric ligands to retain only non-polymeric small molecules, leaving us with 5,763 entries. The atomic structures of the remaining entries were parsed using the Biopython library [11].

To ensure relevance for small-molecule-based drug discovery, we applied filtering criteria inspired by the Hariboss dataset [35]. Specifically, we only considered ligands that (i) contain at least one carbon atom, (ii) have a molecular weight between 160 and 1000 daltons, and (iii) consist of atoms from the set {`C, H, N, O, Br, Cl, F, P, Si, B, S, Se`}. After filtering, 2,358 PDB entries with ligands of interest remain.

We identified RNA residues within a 6Å radius for each ligand atom and recorded their minimum atomic distances to the ligand. These distances were then transformed into interaction scores by inverting them with respect to the 6Å maximum distance, setting residues with no neighboring ligand atoms to negative infinity. The scores were normalized via a softmax function. This resulted in a dataset containing 5,026 RNA-small molecule interactions with 1,320 unique RNA sequences and 419 unique ligands. The SMILES string for each occurring small molecule was downloaded from the PDB using the molecule identifier. RNA sequences were obtained by replacing RNA linking molecules with their respective nucleotide representation or `N` if no corresponding representation was found and then iteratively transforming the RNA residue chain to a string representation, as we have observed that the listed RNA sequence often does not correspond to the recorded residue chain.

4 Methods

We adopt a sequence-driven approach for predicting RNA–small molecule interactions. To address the scarcity of datasets, we employ pre-trained foundation models, which have been trained in an unsupervised way on large scale data and output rich, transferable feature representations. These models are incorporated either in a *frozen state* – where their weights remain unchanged during backpropagation – or fine-tuned on our dataset. Fine-tuning can involve updating all or a subset of the model's weights, potentially with adapted learning rates, or by applying parameter-efficient techniques such as Low-Rank Adaptation (see Section 4.4).
As shown in Figure 7, our model separately encodes RNA sequence and molecule SMILES representation. The resulting embeddings are then concatenated and passed to the model head, which consists of a simple *multilayer perceptron* (MLP) that directly outputs the final regression prediction.

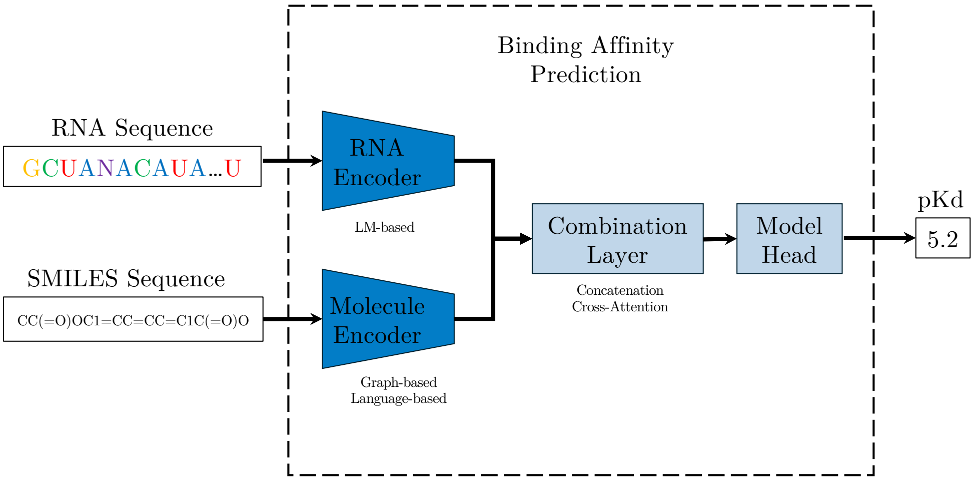


Figure 7: Schematic overview of our binding affinity prediction model. The model takes two input sequences: (i) an RNA sequence (top left) and (ii) a SMILES sequence (bottom left) representing a small molecule's chemical structure. Each input passes through a dedicated encoder: an LM-based RNA Encoder (top center) and either a graph-based or language-based Molecule Encoder (bottom center). Their outputs are then combined and fed into the Model Head, yielding a numeric pK_d prediction.

4.1 Molecule Encoders

We evaluate multiple *graph neural network* architectures to encode molecular structures represented as graphs. Our approaches include Graph Convolutional Networks (GCNs), Graph Isomorphism Networks (GINs), and Graph Diffusion Convolution Networks (GDCs), as well as a pre-trained model.

Graph Neural Networks (GNNs). We convert SMILES strings into *molecular graphs* $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where nodes in $\mathcal{V}$ represent atoms and edges in $\mathcal{E}$ represent bonds [4].
Conversion is performed using *RDKit*[2], which robustly parses SMILES into molecular

[2] RDKit: open-source cheminformatics software. https://www.rdkit.org

graphs. In Figure 8, we can see the molecular graph of aspirin with the atomic number as *node feature* and the bond type as *edge feature* (e.g., single bonds are represented as 1.0, aromatic bonds as 1.5 and double bonds as 2.0.).

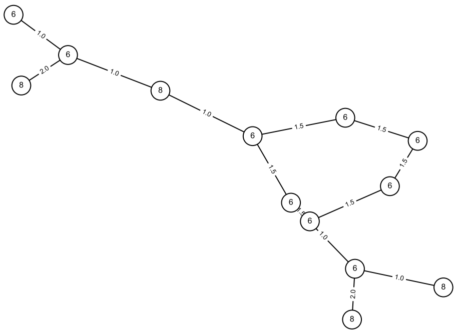


Figure 8: Molecular graph of aspirin.

Graph Convolutional Network (GCN). A GNN consists of learnable and differentiable functions that are invariant to graph permutations, in which GCN is a common choice that aggregates information from a node's neighbors [22]. For layer k, the node-wise aggregation is defined as:

$$h_v^{(k)} = \Theta^\top \sum_{u \in N(v) \cup \{v\}} \frac{1}{\sqrt{d_v d_u}} \cdot h_u^{(k-1)},$$

where d_v and d_u are the degrees of nodes v and u, respectively, and Θ are trainable weights. However, GCNs use a mean-based aggregation that is not injective, potentially causing different graphs to have identical embeddings.

Graph Isomorphism Network (GIN). To address this limitation, we also use the GIN [49], which uses a sum-based aggregation function defined as:

$$h_v^{(k)} = h_\Theta \left((1+\epsilon) \cdot h_v^{(k-1)} + \sum_{u \in N(v)} h_u^{(k-1)} \right).$$

Here, ϵ controls the node's importance relative to its neighbors, and h_Θ is typically an MLP. The sum aggregation in GIN is more powerful than the mean aggregation in GCNs, effectively allowing the network to distinguish between more similar graphs [15].

Graph Diffusion Convolution Network (GDC). Graph diffusion convolution networks extend traditional neighborhood-based aggregation by incorporating diffusion processes to capture local and global graph structures [3].
A diffusion kernel K is computed as a weighted sum over multiple diffusion steps:

$$K = \sum_{t=0}^{T} \alpha_t P^t,$$

where P is a transition matrix derived from the normalized adjacency matrix, α_t are the diffusion coefficients, and T is the maximum number of diffusion steps. The node update rule then becomes:

$$h_v^{(k)} = \Theta^\top \sum_{u \in \mathcal{V}} K_{vu} \, h_u^{(k-1)},$$

which aggregates features from nodes across varying distances.
This diffusion process enables the network to better capture the intricate connectivity patterns between atoms, thereby enhancing discrimination between molecular graphs of similar small molecules [28].

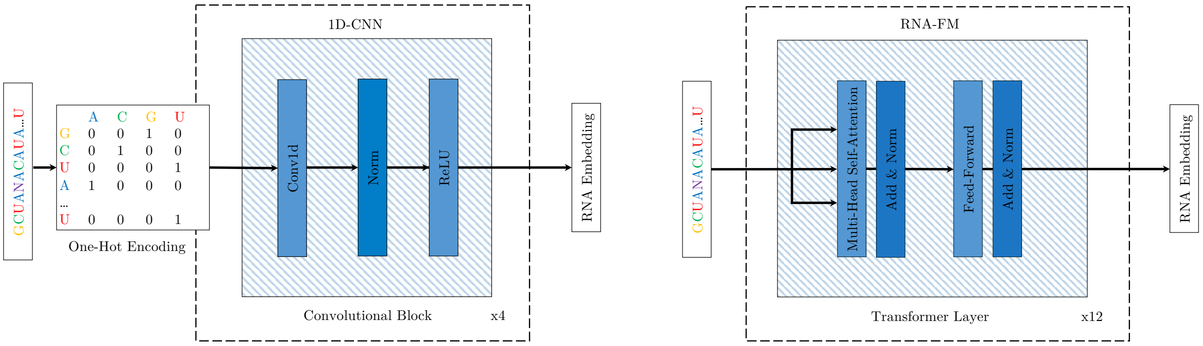
Molecular Contrastive Learning of Representations via Graph Neural Networks (MolCLR). Wang et al. [47] train GNNs in a self-supervised fashion using unlabeled graph representations of around 10 million unique molecules. They employ augmentations on the graph level to mask parts of the molecule representations. These augmentations allow the usage of different graph representations for the same molecule. A contrastive loss is used to learn strong feature representations for the molecules using the augmented representations. While the pre-trained MolCLR foundation model has not been trained with particular attention to small molecules, it has shown strong performance on general molecular learning benchmarks. Therefore, we use it as a molecule encoder for our experiments.

4.2 RNA Encoders

We decide to test two different approaches for encoding the RNA sequences, a one-dimensional *Convolutional Neural Network* (CNN) as well as a *pretrained Language Model*. The CNN is commonly used in literature as an easy-to-train baseline for RNA sequences.

1D-CNN as a Baseline. To train a CNN on RNA sequences, we first need to encode each nucleotide base. We employ *one-hot encoding* to represent each nucleotide – `A` (Adenine), `C` (Cytosine), `G` (Guanine) and `U` (Uracil) – as a distinct numerical vector. Our implementation also includes mapping for `T` (treated like `U`) and unknown nucleotides `X` and `Y`, which are all assigned a zero vector to handle missing or unrecognized bases. Once encoded, the RNA sequences are passed through 4 one-dimensional convolutional layers, allowing the model to extract meaningful sequence features while preserving the original sequence length (see Figure 9a). As a result, the CNN generates a higher-dimensional representation of each nucleotide, which can subsequently be combined with the molecular embeddings.

RNA Foundation Model (RNA-FM). Training a model entirely from scratch on a small dataset can lead to overfitting and poor performance [45]. Hence, leveraging a powerful, pre-trained encoder might improve the robustness of binding affinity predictions by providing more meaningful embeddings for the RNA sequences. RNA-FM [9] fulfills this role as a large-scale foundation model, pre-trained on over 23 million non-coding RNA sequences using a self-supervised masked token approach. Our dataset contains



(a) 1D-CNN Model.

(b) RNA-FM Model (adapted from [9]).

Figure 9: Schematic overview of model architectures for the RNA encoders.

both coding and non-coding RNAs as described by Krishnan, Roy, and Gromiha [23]. Therefore, we conject that a distribution mismatch exists between RNA-FM's pretraining dataset and our dataset. To address this, we not only utilize the frozen pre-trained embeddings but also finetune the model on our data, thus ensuring better alignment with our specific data distribution.

The RNA-FM model is an encoder-only Transformer based on BERT [12], consisting of 12 sequential Transformer blocks and approximately 100 million parameters (see Figure 9b). The model tokenizes and processes the RNA sequences at the nucleotide level, thus outputting a 640-dimensional embedding for each nucleotide in the sequence.

Since Transformer models are inherently permutation-invariant, they rely on positional embeddings to provide the attention mechanism with information about token positions within the sequence. RNA-FM employs learnable positional embeddings for sequences up to 1024 nucleotides in length.

However, 48 samples in our dataset exceed this length, making them incompatible with the model's encoding constraints. To address this, we truncate these sequences to their first 1024 nucleotides, discarding the remaining portion. We acknowledge that this truncation may, in some cases, impact the model's predictive accuracy – especially if, for example, molecular binding pockets lie beyond the retained segment.

Addressing this limitation would require retraining RNA-FM with longer positional encodings and extended sequence lengths, which was beyond the computational resources available for this project.

Chen et al. [9] show that the RNA-FM embeddings help the model to outperform other models on tasks such as 2D/3D structure prediction, evolution prediction, or protein-RNA binding preference modeling. Given these results, it is reasonable to assume that the embeddings produced by RNA-FM may also capture meaningful features relevant to RNA-small molecule interaction prediction.

4.3 Combination Layers

Molecule embedding and the RNA embedding are combined in the combination layer before being passed on to the final regression head, as depicted in Figure 7.

We decide to test two options for this combination layer, simple *concatenation* and *cross-attention*.

Concatenation Layer. The concatenation layer concatenates both embeddings. However, since their scales can differ significantly, we first normalize them. We opt for layer normalization, as it ensures a consistent scale for each embedding and has been found to improve training stability.

Cross-Attention Layer. During training, the cross-attention layer learns how to dynamically weight different parts of the RNA embedding based on the molecule embedding, providing a potentially richer interaction than mere concatenation. Corresponding to the size difference between molecules and RNA, we use each embedded RNA residue as a *key* (K) and *value* (V) and the global-pooled molecule embedding as a single *query* (Q) (see Figure 10). A final layer normalization step ensures stability. Due to data scarcity, we decide to only use a *single attention head* to not increase the parameter size further. We observe that some molecular information is lost, as it is only present in the query. To address this, we also test appending the molecular representation to the output embeddings. However, this approach does not lead to any improvement in performance.

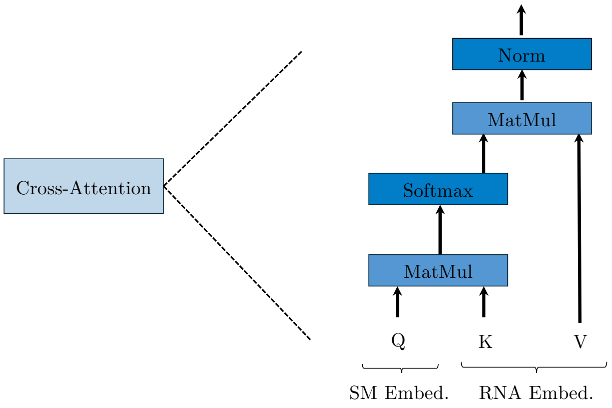


Figure 10: Single head of cross-attention (adapted from [46]).

4.4 Parameter-Efficient Finetuning via LoRA

Although large language and foundation models offer powerful feature representations, fully fine-tuning all parameters can lead to overfitting when data is limited while incurring high computational costs. To mitigate these issues, we employ *Low-Rank Adaptation* (LoRA) [18], a *parameter-efficient* fine-tuning paradigm that freezes most of the pre-trained weights and only updates a small number of parameters in each layer.

LoRA introduces a low-rank re-parameterization of a subset of the model weights, typically those associated with attention or feed-forward layers in Transformer-based architectures. Let $W_0 \in \mathbb{R}^{d \times k}$ be a pre-trained weight matrix, which we keep frozen, and $\Delta W \in \mathbb{R}^{d \times k}$ be its learnable adaptation. Instead of learning ΔW directly, LoRA factorizes it into two low-rank matrices $A \in \mathbb{R}^{r \times k}$ and $B \in \mathbb{R}^{d \times r}$, where $r \ll \min(d, k)$. Accordingly,

$$\Delta W = BA,$$

and the forward pass for a given input $x \in \mathbb{R}^k$ is modified to:

$$\begin{aligned} h &= W_0 x + \Delta W x \\ &= W_0 x + BAx, \end{aligned}$$

with only A and B being trainable. The rank r thus becomes a key hyperparameter balancing the expressivity-efficiency trade-off.

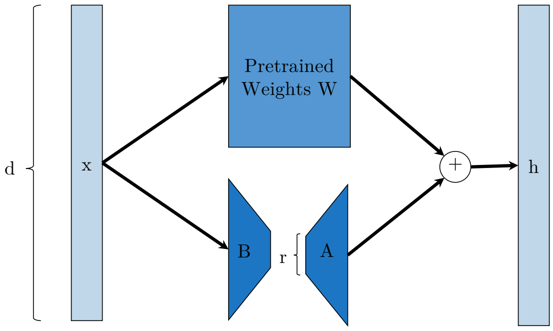


Figure 11: LoRA re-parametrization. We only train A and B (adapted from [18]).

By constraining the fine-tuning within a low-rank subspace, LoRA significantly reduces the number of learnable parameters, mitigating the risk of overfitting our small RNA-small molecule dataset. This also makes the approach more computationally tractable, as matrix multiplications with rank r and dimension k or d are cheaper than the full $\mathcal{O}(dk)$ operations required for fully trainable layers. From a practical standpoint, LoRA allows us to preserve the vast, generic knowledge embedded in the original parameters W_0 – which were learned from large-scale, self-supervised pre-training on diverse RNA sequences – while learning only task-specific patterns relevant to RNA-small molecule binding. Therefore, this method helps us adapt powerful encoders like RNA-FM to our regression task without excessive computational overhead or overfitting.

4.5 Pretraining with Pocket Information

Binding pockets of RNA-ligand complexes provide rich information on the binding behavior between a ligand and RNA. We try to incorporate this information into our model by pre-training on the pockets indicated in the dataset introduced in Section 3.5. The dataset provides binding weights for each nucleotide in the RNA sequence with respect to a given ligand. These weights are used as a supervision signal for pre-training the attention weights in the cross-attention layer (see Figure 10) combining molecule and RNA embeddings. Similar to [16] and because the target weights are often very sparse, we use a cross entropy loss to compare the attention weights to these target weights.

4.6 Inference for Classification Tasks

To complement our regression-based prediction of RNA-small molecule binding affinity, we extend our approach to include a classification framework for distinguishing binding from non-binding RNA-small molecule pairs. We use two classification strategies: one based on a fixed threshold and another based on pairwise ranking. We perform the classification test on ROBIN dataset [50], which will be discussed in Section 5.3.

Classification via Fixed Threshold. The continuous binding affinity predictions can be transformed into discrete classifications by adopting a fixed threshold. In this approach, a threshold value is predetermined – set to 4.0 in our case, reflecting the ligand concentration used in the small-molecule microarray experiments that underpinned our classification dataset [24, 50]. Predicted affinity values above this threshold are classified as *positive* (binding), while those below are deemed *negative* (non-binding).

Classification via Ranking. In this approach, the classification dataset is initially segmented into individual samples and then grouped by their corresponding RNA families. Each sample is assigned to the *positive* (binding) or *negative* (non-binding) class based on its experimental pK_d value. Within each RNA family, binding samples are randomly paired with non-binding samples. The model generates a pK_d prediction for each pair for both members. A pair is considered correctly classified if the predicted pK_d for the binding sample exceeds that for the non-binding sample.

5 Results

We first reproduce the baseline method RSAPred to establish a reliable reference against which to compare our deep learning methods. RSAPred selects features and trains a linear regression model individually per RNA family. Next, we evaluate the performance of our proposed deep learning models, which are trained jointly for all RNA families, across different data splits. Additionally, we assess the model's extrapolation capabilities in identifying binding pairs within a classification framework. Finally, we examine the impact of pretraining with pocket detection on downstream performance.

To evaluate the performance of our method and compare it with baseline approaches, we adopted three widely used metrics for binding affinity prediction from Krishnan, Roy, and Gromiha [24], including mean absolute error (MAE), Pearson correlation coefficient (PCC) and Spearman's correlation coefficient (SPCC) (see Appendix B).

MAE quantifies prediction error magnitude. PCC measures the linear correlation between predictions and ground truth, while SPCC evaluates monotonic relationships through rank correlation.

5.1 Reproducing RSAPred

Krishnan, Roy, and Gromiha [24] propose with RSAPred a linear regression approach for predicting pK_d binding affinities, akin to [7]. Their method relies exclusively on sequence-based features derived from RNA sequences and small molecules. Specifically, they extract 1,507 features and employ a forward feature selection algorithm, iteratively selecting features based on the Pearson correlation coefficient and mean absolute error (MAE). This is done for each RNA family separately.[3]

However, we encountered inconsistencies between the paper's description and the published code, preventing the reproduction of the selected features due to algorithmic discrepancies and implementation bugs. Consequently, we re-implemented the forward feature selection algorithm, using the published code as a baseline while ensuring alignment with the methodology described in the paper. This re-implementation was particularly necessary because the reported results in [24] were obtained using features optimized on the entire dataset. As a result, their 10-fold and leave-one-out cross-validation results are compromised, as feature selection was implicitly tailored to the test sets.

To enable a fair comparison between RSAPred and our methods, we introduced fixed training and test splits, ensuring that feature selection was performed exclusively on training sets, with evaluations strictly on test sets. We validated our re-implementation by comparing results obtained on the full dataset against those reported by RSAPred, as detailed in Table 14. The outcomes using our data splits, and re-implemented feature selection approach are presented in Table 3.

Even though feature selection was fitted to the complete dataset, RSAPred reports strong performance on external test sets (see Table 4). We were able to reproduce these results using the features reported in the paper, except for Viral and Ribosomal RNA, for which the available dataset differed from the reported dataset.

[3] The corresponding code is publicly available on GitHub: https://github.com/Sowmya-R-Krishnan/RSAPred

RNA Subtype	No. of Features in Final Model	Stratified 10-fold CV		Extrapolation Split	
		PCC ↑	MAE ↓	PCC ↑	MAE ↓
Aptamers	12	0.636	0.694	0.008	1.638
miRNAs	8	0.723	0.618	0.507	1.555
Repeats	13	0.570	0.669	0.476	1.265
Ribosomal RNAs	11	0.728	0.821	0.434	2.156
Riboswitches	13	0.656	1.103	0.251	1.443
Viral RNAs	8	0.645	0.808	0.245	1.649

Table 3: Performance of our re-implementation of the feature selection algorithm from RSAPred on the data splits proposed in Section 3.4. The Pearson correlation coefficient (PCC) and the mean absolute error (MAE) are reported for the test sets of each split. For the stratified 10-fold cross-validation (CV) the feature selection algorithm was run for each fold individually and the results on the test splits were aggregated by using the mean.

Dataset	Description	Type	Result	RNA Families
R-SIM [23]	Training data	Regression	Mean correlation 0.83 MAE 0.66	All RNA subtypes
External Blind Test Datasets	External test data	Classification & Regression	High correlation MAE < 1.0	Aptamers, miRNAs, Riboswitches, Viral RNAs
Subsets of External Blind Test Datasets:				
QSAR [7]	Model for specific RNA targets	Regression	–	Viral RNAs
ROBIN Repository [50]	Chemical library for RNA-ligand binding Classification	–	Aptamers, miRNAs, Riboswitches, Viral RNAs	
RNAmigos [34]	Ranking method for RNA-ligand binding	Classification	Higher SPCC	Viral RNAs
fingeRNAt-ML [44]	Classifying active & inactive compounds	Classification	Higher F1-score	Viral RNAs

Table 4: RSAPred [24] used a combination of datasets to report their performance. The training dataset was compiled from R-SIM and entails pK_d prediction as a regression task. This dataset was solely used for feature optimization and training. The test performance was evaluated on two external datasets. The test datasets entail samples from the Aptamer, miRNA, Riboswitch and Viral RNA families. For the Repeats and Ribosomal RNA families no test datasets were presented. Additional comparisons were made with the methods presented in [34] and [44].

5.2 Regression Results with Deep Learning Methods

Having determined the baseline performance of RSApred, we test our models on the three previously introduced splits and compare their performance. We use frozen pre-trained encoders, LoRA fine-tuning, and encoders trained from scratch.

All models were trained on a single Nvidia Tesla V100 GPU. Logging was performed using the *Weights and Biases* API. Configurations were managed using *Hydra.* All experiments were conducted with early stopping and a cosine learning rate scheduler. The maximum number of training epochs was set to 500. However, most models converged after around 100 epochs.

		RNA Family						
RNA Model	**Mol Model**	**All ↓**	**Aptamers ↓**	**miRNA ↓**	**Repeats ↓**	**Ribosomal ↓**	**Riboswitches ↓**	**Viral RNA ↓**
RSAPred	–	0.882	0.867	**0.434**	0.821	**0.706**	2.054	0.806
RNA-FM (frozen)	Graph Diffusion	0.905	0.893	0.595	0.884	1.098	0.899	0.899
RNA-FM (frozen)	Graph Diffusion Cross	0.994	0.790	0.945	1.120	1.189	1.088	0.883
RNA-FM (frozen)	MolCLR (frozen)	0.933	0.926	0.810	0.834	1.100	**0.834**	0.916
1D-CNN	Graph Diffusion Cross	0.805	0.724	0.481	**0.636**	1.191	1.112	0.666
RNA-FM (frozen)	GIN	**0.750**	**0.708**	0.715	0.939	0.949	1.320	**0.542**

Table 5: Model MAE test results for the interpolation split.

RNA Model	**Mol Model**	**PCC ↑**	**SPCC ↑**
RNA-FM (frozen)	Graph Diffusion	0.529	0.514
RNA-FM (frozen)	Graph Diffusion Cross	0.453	0.446
RNA-FM (frozen)	MolCLR (frozen)	0.521	0.504
1D-CNN	Graph Diffusion Cross	0.636	0.668
RNA-FM (frozen)	GIN	0.669	0.643

Table 6: Pearson and Spearman Correlation Coefficient test results for the interpolation split.

5.2.1 Interpolation Split

The results on the interpolation split shown in Table 5 clearly demonstrate that the deep learning-based models outperform RSApred in terms of MAE on the test set. RSApred performs especially worse in the Riboswitches class. Combining a GIN for molecule encoding with a frozen RNA-FM model yields especially strong performance.
We do not see a clear benefit of using a pre-trained model for molecule encoding, as the model using the MolCLR backbone results in a worse MAE than training a Graph Diffusion model from scratch. In comparison, we attribute the good performance of the GIN model to its rich molecule representations compared to the Graph Diffusion model. While Graph Diffusion smoothes the graph, GIN uses the exact molecule graph and distinguishes better between multiple graph structures.
Similarly, the models can be compared in terms of molecule and RNA embedding concatenation. The results of the Graph Diffusion and RNA-FM encoder combination indicate, that using concatenation is beneficial over cross attention, however the overall performance difference is only minor.

5.2.2 Extrapolation Split

		RNA Family						
RNA Model	**Mol Model**	**All ↓**	**Aptamers ↓**	**miRNA ↓**	**Repeats ↓**	**Ribosomal ↓**	**Riboswitches ↓**	**Viral RNA ↓**
RSAPred	–	1.721	1.638	1.555	1.265	2.156	1.443	1.649
RNA-FM (frozen)	Graph Diffusion	1.435	1.381	2.118	1.092	**1.557**	**0.713**	1.409
RNA-FM (frozen)	Graph Diffusion Cross	**1.394**	**1.340**	1.845	1.156	1.661	0.852	1.214
RNA-FM (frozen)	MolCLR (frozen)	1.406	1.465	1.727	1.159	1.659	0.823	1.119
1D-CNN	Graph Diffusion Cross	1.413	1.432	1.619	**0.809**	1.772	1.523	**1.084**
RNA-FM (frozen)	GIN	1.443	1.418	**1.473**	0.846	1.949	1.084	1.141
RNA-FM (LoRA)	Graph Diffusion	1.336	1.286	1.801	1.071	1.621	0.972	1.070
RNA-FM (LoRA, pret.)	Graph Diffusion Cross (pret.)	1.460	1.387	1.922	1.239	1.852	1.064	1.190
Mean Prediction		1.538	1.333	1.980	1.228	2.274	1.554	1.457

Table 7: Model MAE test results for the extrapolation split.

Overall, the models exhibit poorer generalization on the extrapolation test set than the interpolation test set. We attribute this to two key reasons. First, extrapolation is inher-

RNA Model	Mol Model	PCC ↑	SPCC ↑
RNA-FM (frozen)	Graph Diffusion	0.170	0.142
RNA-FM (frozen)	Graph Diffusion Cross	0.137	0.161
RNA-FM (frozen)	MolCLR (frozen)	0.211	0.210
1D-CNN	Graph Diffusion Cross	0.202	0.273
RNA-FM (frozen)	GIN	0.07	0.186

Table 8: Pearson and Spearman Correlation Coefficient test results for the extrapolation split.

ently more challenging, requiring the model to generalize beyond the RNA and molecule embeddings seen during training. Unlike interpolation, where the model can leverage known embeddings for familiar SMILES strings and RNA sequences and only has to predict the new combination, extrapolation demands the ability to make predictions for entirely new samples. Second, the extrapolation data split further reduces the number of unique RNA sequences available in the training set compared to interpolation. This decrease in diversity further limits the information available during training, making it even harder for the model to learn meaningful representations.

Notably, the magnitude of MAE itself is rather high across all models. The low PCC and SPCC shown in Table 6 indicate that the model struggles to capture the variance in the data distribution. This is further supported by the observation that a simple mean-prediction model, which always outputs the mean of the training distribution, also achieves a comparable MAE of 1.538 on the test set.

Still, several minor trends can be observed in the results. All deep learning models manage to outperform the RSApred baseline on the extrapolation test split. It is remarkable that RSApred performs poorer than a mean predictor in terms of MAE. Furthermore, models utilizing pre-trained RNA encoders perform slightly better on the extrapolation split than those trained from scratch. This suggests that pre-training provides greater benefits in the extrapolation setting, where the model must generalize to unseen RNA sequences, compared to interpolation, where it can rely more on learned embeddings from the training data.

5.3 Results with Classification Inference

We evaluate our models on the classification task on the ROBIN dataset [50], which comprises 2,373 RNA-small molecule pairs – spanning four RNA families: Aptamers (616 pairs), miRNA (232 pairs), Riboswitch (705 pairs), and Viral RNA (820 pairs).

For the fixed threshold approach, as described in Section 4.6, we computed standard classification metrics, including accuracy, area under receiver operating characteristic curve (AUROC) score, specificity, sensitivity/recall, precision, and F1-score (see Appendix B). Samples were classified as *positive* (binding) or *negative* (non-binding) based on a fixed threshold of 4.0 for the predicted pK_d values.

Tables 9 and 10 show that all models yield an overall accuracy of approximately 50% and an AUROC score near 0.5. These results suggest that the models cannot effectively discriminate between binding and non-binding samples based on their predicted pK_ds. Moreover, most models exhibited a recall of 1 and a specificity of 0, indicating a strong bias toward predicting the positive class, resulting in a high false-positive rate. The performance also varied across RNA families.

RNA Model	Mol Model	AUROC Score	Precision	Specificity	Sensitivity/Recall	F1 Score
RNA-FM (frozen)	MolCLR	0.500	0.541	0.000	1.000	0.702
RNA-FM (frozen)	Graph Diffusion	0.500	0.541	0.000	1.000	0.702
RNA-FM (frozen)	GIN	0.493	0.538	0.037	0.950	0.686

Table 9: Classification performance metrics evaluated at a fixed decision threshold of 4.0. Results are aggregated across RNA families.

		RNA Family				
RNA Model	**Mol Model**	**All**	**Aptamers**	**miRNA**	**Riboswitch**	**Viral RNA**
RNA-FM (frozen)	MolCLR	0.541	0.618	0.368	0.517	0.551
RNA-FM (frozen)	Graph Diffusion	0.541	0.617	0.368	0.517	0.551
RNA-FM (frozen)	GIN	0.530	0.618	0.368	0.517	0.520

Table 10: Classification accuracy at a fixed threshold of 4.0 per RNA family.

We suspect that the fixed threshold method struggles because the training data are predominantly binding (positive) samples, causing the models to output values clustered around the mean value of the training data distribution.

To further investigate the discriminative potential of our models, we also evaluated classification performance via ranking, as described in Section 4.6. We use ranking accuracy and pK_d difference as evaluation metrics. The ranking accuracy is computed as

$$\text{Accuracy} = \frac{\#\text{ correctly ranked sample pairs}}{\#\text{ total sample pairs}}.$$

And the pK_d differences are computed as:

$$\text{Difference} = \hat{y}_{\text{pos}} - \hat{y}_{\text{neg}}.$$

			RNA Family				
RNA Model	**Mol Model**	**Metric**	**All**	**Aptamers**	**miRNA**	**Riboswitch**	**Viral RNA**
RNA-FM (frozen)	MolCLR	Accuracy	0.598	0.740	0.491	0.684	0.449
		Difference	-0.018 (± 0.228)	-0.075 (± 0.094)	-0.116 (± 0.334)	-0.031 (± 0.156)	0.064 (± 0.283)
RNA-FM (frozen)	Graph Diffusion	Accuracy	0.517	0.547	0.409	0.606	0.450
		Difference	-0.019 (± 0.528)	-0.013 (± 0.320)	-0.021 (± 0.615)	0.104 (± 0.515)	-0.129 (± 0.607)
RNA-FM (frozen)	GIN	Accuracy	0.501	0.578	0.504	0.565	0.388
		Difference	-0.041 (± 0.438)	-0.094 (± 0.297)	-0.036 (± 0.372)	0.050 (± 0.379)	-0.222 (± 0.521)

Table 11: Evaluation on classification test with a ranking method. The column *All* indicates the overall metric computed over all pairs, while the remaining columns report per RNA family values. The accuracy is calculated as the fraction of correctly ranked pairs over the total number of pairs. The pK_d differences are computed as the positive sample's pK_d minus the negative sample's pK_d.

As shown in Table 11, the MolCLR model correctly ranks positive samples above negative ones in approximately 60% of cases, modestly surpassing random chance (50%). Other models performed near chance levels, suggesting their ability to distinguish between binding and non-binding samples is limited.

Overall, the fixed threshold and ranking approaches indicate that the current models perform only marginally above – or, in some cases, below – random chance when classifying RNA-small molecule interactions.

The variability in performance across RNA families suggests that certain families, such as Aptamers and Riboswitches, may present clearer signals for binding, whereas others are more challenging. These discrepancies could arise from differences in RNA sequence length, structure variability, or inherent noise in the binding sites.

The results also imply that the model architectures and feature representations derived from SMILES and RNA sequences might not fully capture the interactions required for classification in this context.

5.4 LoRA-Finetuning

To adapt the pre-trained RNA model better to our data domain, we decided to perform finetuning using LoRA [18]. However, due to the increased memory requirements of finetuning the model, we decide to use gradient accumulation to achieve an effective batch size comparable to the remaining experiments.

LoRA reduces the number of trainable parameters of RNA-FM to around 1% to 3% of its original parameter size. This reduction is comparable to the range reported by Hu et al. [18]. For instance, the model using a GIN as molecule encoder and a LoRA finetuned RNA-FM contains $101,671,051$ total parameters, of which only $1,499,265$ are trainable. The two main factors driving this reduction in parameter size are the size of the latent space r, corresponding to the rank of the LoRA matrix AB, and the number of frozen layers of the model that are not updated.

We find that the combination of a fully LoRA finetuned RNA-FM and a Graph Diffusion model performs best on the extrapolation test set as shown in Table 7, however the improvement is minor regarding the additional training efforts and parameters. Arguably, considering that the models trained from scratch do not manage to generalize well to the extrapolation set due to a lack of data, the LoRA finetuning also does not manage to fully capture the distribution shift from only non-coding RNAs to coding and non-coding samples. Still, as seen in Figures 20a and 20b, the model finetuned with LoRA slightly improves the clustering of RNA embed- dings, particularly for the Ribosomal family.

Additionally, we conducted an ablation study to investigate the influence of the number of frozen RNA-FM layers and the size of the latent space, shown in Tables 15 and 16. Our results indicate that the best configuration is achieved with $r = 32$ and no frozen layers. Furthermore, we observe that applying LoRA to query, value, and key projections results in lower validation and training loss than restricting LoRA to only the query and value projections. However, this configuration leads to a higher test loss. This suggests that the additional parameters improve performance on the training and validation data but increase the overfitting on our training data. Since our test data comes from the extrapolation datasplit, this overfitting reduced the performance on the test dataset.

5.5 Pocket Pretraining

We find that the attention weights learned in the cross-attention layer of our models are naturally sparse, meaning that most of the binding affinity prediction is based on the interaction between the molecule and only a few nucleotides of the RNA, as seen in Figure 21. Motivated by this observation, we investigate pretraining the attention mechanism on our self-curated dataset described in Section 3.5. We observe that while the model can overfit to a small dataset, learning RNA pockets across all training samples requires greater model capacity than predicting binding affinity alone. To address this, we unfreeze the last three RNA-FM layers and fine-tune them using LoRA. Still, predicting pocketes purely based on RNA sequence inputs remains to be challenging task for the model, as it struggles to generalize well to the validation data. This limitation can possibly also be attributed to the limited size of our pretraining dataset. Still, we explored leveraging this pretrained model for the downstream task of binding affinity prediction. However, our results on the extrapolation split reached a comparable performance as reported in Table 7 and thus the additional pretraining does not show a clear benefit.

6 Conclusion

Predicting RNA–small molecule interactions solely from sequence data can offer valuable insights for drug discovery, eliminating the need for costly and experimentally derived structural information. In this report, we studied how deep learning techniques can be leveraged to predict the binding affinity between RNA and molecules and assess the performance both across interpolation and extrapolation data.

To this end, we investigate architectures that consist of separate molecule and RNA encoders along with a dedicated combination layer to model the interaction. We mainly study language-model based RNA encoders and only graph based molecule encoders.

Our results reveal that the models are able to perform well on interpolation data, which mostly contains RNA sequences and molecules already encountered in training but presented in new combinations. Furthermore, our findings demonstrate that the models struggle to generalize to extrapolation data, where the test set contains entirely novel RNA sequences and only a limited number of molecules previously seen during training. Still, the proposed deep learning models outperform the baseline method, which is based on a linear regression, across both datasplits. Additionally, we investigate the extrapolation performance on a previously unseen classification dataset. We show that our model is only able to make predictions for binding interactions and does not generalize to non-binding samples.

To adress the data scarcity present for the given task, we curate a separate dataset, which can be used by pretraining on the related task of binding site detection. Furthermore, we test utilizing and finetuning larger pretrained models for both RNA and molecule encoders. However, both approaches did not show larger performance improvements in terms of generalization capabilities to the extrapolation data.

Overall, we see the main limitation for the task in both availability and quality of the data. In addition to the need for larger datasets, our data exploration suggests that incorporating more detailed experimental conditions could be crucial for improving predictive performance. Consequently, we conclude that, in the absence of larger datasets, future research in this field could benefit from hybrid approaches that integrate simulation techniques with deep learning models to enhance predictive accuracy and generalization.

Bibliography

[1] Walid Ahmad et al. *ChemBERTa-2: Towards Chemical Foundation Models.* Sept. 2022. DOI: `10.48550/arXiv.2209.01712`.

[2] Sun et al. *Contrastive pre-training and 3D convolution neural network for RNA and small molecule binding affinity prediction.*

[3] James Atwood and Don Towsley. "Diffusion-convolutional neural networks". In: *Advances in neural information processing systems* 29 (2016).

[4] Kenneth Atz, Francesca Grisoni, and Gisbert Schneider. "Geometric deep learning on molecular representations". In: *Nature Machine Intelligence* 3.12 (Dec. 2021). DOI: `10.1038/s42256-021-00418-8`.

[5] BJ Bender, S Gahbauer, A Luttens, et al. "A practical guide to large-scale docking". In: *Nature Protocols* 16.10 (2021). DOI: `10.1038/s41596-021-00597-z`.

[6] Anke Busch and Rolf Backofen. "INFO-RNA—a fast approach to inverse RNA folding". In: *Bioinformatics* 22.15 (2006). DOI: `10.1093/bioinformatics/btl194`.

[7] Zhengguo Cai et al. "Quantitative Structure–Activity Relationship (QSAR) Study Predicts Small-Molecule Binding to RNA Structure". In: *Journal of Medicinal Chemistry* 65.10 (May 2022). DOI: `10.1021/acs.jmedchem.2c00254`.

[8] Juan G. Carvajal-Patiño et al. *RNAmigos2: Fast and accurate structure-based RNA virtual screening with semi-supervised graph learning and large-scale docking data.* June 2024. DOI: `10.1101/2023.11.23.568394`.

[9] Jiayang Chen et al. *Interpretable RNA Foundation Model from Unannotated Data for Highly Accurate RNA Structure and Function Predictions.* Aug. 2022. DOI: `10.48550/arXiv.2204.00300`.

[10] Jessica L Childs-Disney et al. "Targeting RNA structures with small molecules". In: *Nature Reviews Drug Discovery* 21.10 (2022).

[11] Peter JA Cock et al. "Biopython: freely available Python tools for computational molecular biology and bioinformatics". In: *Bioinformatics* 25.11 (2009).

[12] Jacob Devlin et al. *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding.* 2019.

[13] Matthew D Disney. "Targeting RNA with small molecules to capture opportunities at the intersection of chemistry, biology, and medicine". In: *Journal of the American Chemical Society* 141.17 (2019).

[14] Puck van Gerwen et al. *"3DReact": Geometric deep learning for chemical reactions.* July 2024.

[15] Justin Gilmer et al. "Neural message passing for quantum chemistry". In: *International conference on machine learning.* PMLR. 2017.

[16] Mogan Gim et al. "ArkDTA: attention regularization guided by non-covalent interactions for explainable drug–target binding affinity prediction". In: *Bioinformatics* 39.Supplement$_1$ (June 2023). DOI: `10.1093/bioinformatics/btad207`.

[17] KA Hartman and GJ Thomas Jr. “Secondary structure of ribosomal RNA”. In: *Science* 170.3959 (1970).

[18] Edward J. Hu et al. *LoRA: Low-Rank Adaptation of Large Language Models.* Oct. 2021. DOI: 10.48550/arXiv.2106.09685.

[19] Wengong Jin, Regina Barzilay, and Tommi Jaakkola. *Hierarchical Generation of Molecular Graphs using Structural Motifs.* 2020. DOI: 10.48550/ARXIV.2002.03230.

[20] John Jumper et al. “Highly accurate protein structure prediction with AlphaFold”. In: *Nature* 596.7873 (Aug. 2021). DOI: 10.1038/s41586-021-03819-2.

[21] Sung-Hou Kim et al. “Three-dimensional structure of yeast phenylalanine transfer RNA: folding of the polynucleotide chain”. In: *Science* 179.4070 (1973).

[22] Thomas N. Kipf and Max Welling. *Semi-Supervised Classification with Graph Convolutional Networks.* 2017.

[23] Sowmya R. Krishnan, Arijit Roy, and M. Michael Gromiha. “R-SIM: A Database of Binding Affinities for RNA-small Molecule Interactions”. In: *Journal of Molecular Biology* 435.14 (July 2023). DOI: 10.1016/j.jmb.2022.167914.

[24] Sowmya R. Krishnan, Arijit Roy, and M. Michael Gromiha. “Reliable method for predicting the binding affinity of RNA-small molecule interactions using machine learning”. In: *Briefings in Bioinformatics* 25.2 (Jan. 2024). DOI: 10.1093/bib/bbae002.

[25] Jiao Li and Chun Liu. “Coding or Noncoding, the Converging Concepts of RNAs”. In: *Frontiers in Genetics* 10 (May 2019). DOI: 10.3389/fgene.2019.00496.

[26] Kunpeng Li et al. *Tell Me Where to Look: Guided Attention Inference Network.* 2018.

[27] Christopher A Lipinski et al. “Experimental and computational approaches to estimate solubility and permeability in drug discovery and development settings”. In: *Advanced Drug Delivery Reviews* 46.1-3 (Mar. 2001). DOI: 10.1016/S0169-409X(00)00129-0.

[28] Hongli Ma et al. *RNA-ligand interaction scoring via data perturbation and augmentation modeling.* June 2024. DOI: 10.1101/2024.06.26.600802.

[29] Yu-Shui Ma et al. “Paving the way for small-molecule drug discovery”. In: *American Journal of Translational Research* 13.3 (2021).

[30] Vincent Mallet et al. “RNAglib: a python package for RNA 2.5 D graphs”. In: *Bioinformatics* 38.5 (2022).

[31] Frank Yiyang Mao et al. “Comparison of Three Computational Tools for the Prediction of RNA Tertiary Structures”. In: *Non-Coding RNA* 10.6 (2024).

[32] S. David Morley and Mohammad Afshar. “Validation of an empirical RNA-ligand scoring function for fast flexible docking using RiboDock®”. In: *Journal of Computer-Aided Molecular Design* 18.3 (Mar. 2004). DOI: 10.1023/B:JCAM.0000035199.48747.1e.

[33] Medard Edmund Mswahili and Young-Seob Jeong. "Transformer-based models for chemical SMILES representation: A comprehensive literature review". In: *Heliyon* 10.20 (Oct. 2024). DOI: 10.1016/j.heliyon.2024.e39038.

[34] Carlos Oliver et al. "Augmented base pairing networks encode RNA-small molecule binding preferences". In: *Nucleic Acids Research* 48.14 (Aug. 2020). DOI: 10.1093/nar/gkaa583.

[35] Francesco P Panei et al. "HARIBOSS: a curated database of RNA-small molecules structures to aid rational drug design". In: *Bioinformatics* 38.17 (2022).

[36] Rafael Josip Penić et al. *RiNALMo: General-Purpose RNA Language Models Can Generalize Well on Structure Prediction Tasks*. Nov. 2024. DOI: 10.48550/arXiv.2403.00043.

[37] Zhong-Hao Ren et al. "DeepMPF: deep learning framework for predicting drug–target interactions based on multi-modal representation with meta-path semantic analysis". In: *Journal of Translational Medicine* 21.1 (Jan. 2023). DOI: 10.1186/s12967-023-03876-3.

[38] Sergio Ruiz-Carmona et al. "rDock: A Fast, Versatile and Open Source Program for Docking Ligands to Proteins and Nucleic Acids". In: *PLoS Computational Biology* 10.4 (Apr. 2014). Ed. by Andreas Prlic. DOI: 10.1371/journal.pcbi.1003571.

[39] Shaghayegh Sadeghi et al. *Can Large Language Models Understand Molecules?* May 2024. DOI: 10.48550/arXiv.2402.00024.

[40] Wolfram Saenger. *Principles of nucleic acid structure*. Springer Science & Business Media, 2013.

[41] Ge Shan et al. "A small molecule enhances RNA interference and promotes microRNA processing". In: *Nature biotechnology* 26.8 (2008).

[42] Saisai Sun, Jianyi Yang, and Zhaolei Zhang. "RNALigands: a database and web server for RNA - ligand interactions". In: *RNA* 28 (Nov. 2021). DOI: 10.1261/rna.078889.121.

[43] Li-Zhen Sun et al. "RLDOCK: A New Method for Predicting RNA–Ligand Interactions". In: *Journal of Chemical Theory and Computation* 16.11 (2020). DOI: 10.1021/acs.jctc.0c00798.

[44] Natalia A Szulc et al. "Structural interaction fingerprints and machine learning for predicting and explaining binding of small molecule ligands to RNA". In: *Briefings in Bioinformatics* 24.4 (2023).

[45] Vladimir N Vapnik and A Ya Chervonenkis. "On the uniform convergence of relative frequencies of events to their probabilities". In: *Measures of complexity: festschrift for alexey chervonenkis*. Springer, 2015.

[46] Ashish Vaswani et al. "Attention Is All You Need". In: *CoRR* abs/1706.03762 (2017).

[47] Yuyang Wang et al. *Molecular Contrastive Learning of Representations via Graph Neural Networks*. Jan. 2022. DOI: 10.48550/arXiv.2102.10056.

[48] David Weininger. “SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules”. In: *Journal of Chemical Information and Computer Sciences* 28.1 (Feb. 1988). DOI: 10.1021/ci00057a005.

[49] Keyulu Xu et al. *How Powerful are Graph Neural Networks?* 2019.

[50] Kamyar Yazdani et al. “Machine Learning Informs RNA-Binding Chemical Space”. In: *Angewandte Chemie International Edition* 62.11 (Mar. 2023). DOI: 10.1002/anie.202211358.

[51] Yuanzhe Zhou, Yangwei Jiang, and Shi-Jie Chen. “RNA-ligand molecular docking: advances and challenges”. In: *Wiley interdisciplinary reviews. Computational molecular science* 12.3 (Aug. 2021). DOI: 10.1002/wcms.1571.

Appendix

A Use Cases

By leveraging only sequence-level information, researchers can investigate potential RNA-small molecule interactions early in the experimental pipeline, even in the absence of structural data.

A primary application of our approach is the ranking of small molecules based on their predicted binding affinity to a target RNA, facilitating the identification of promising drug candidates. This prioritization accelerates drug discovery by reducing both time and resource expenditures for experiments on less likely candidates.

Moreover, by eliminating the reliance on costly structural determination techniques and minimizing the number of required experimental validations, our method lowers the barriers to computational drug discovery for RNA therapeutics. This democratization further enables laboratories with limited funding to participate in RNA-targeted drug research.

Reduced costs could also make personalized therapeutics – designed for rare diseases or tailored to individual genetic profiles – more feasible, potentially increasing accessibility for broader patient populations. Additionally, the COVID-19 pandemic has highlighted the urgency of rapid drug development.

Approaches that streamline therapeutic candidate identification can be critical for future pandemic preparedness, reinforcing the importance of efficient, accessible computational methodologies in drug discovery.

B Performance Metrics

Regression Metrics

Let $y = \{y_i\}_{i=1}^n$ be the ground truth and $\hat{y} = \{\hat{y}_i\}_{i=1}^n$ the predictions. Then,

$$\begin{aligned}
\mathrm{MAE}(y, \hat{y}) &= \frac{1}{n}\sum_{i=1}^{n} |y_i - \hat{y}_i|;\\
\mathrm{PCC}(y, \hat{y}) &= \frac{\sum_{i=1}^{n}(y_i - \overline{y})(\hat{y}_i - \overline{\hat{y}})}{\sqrt{\sum_{i=1}^{n}(y_i - \overline{y})^2}\sqrt{\sum_{i=1}^{n}(\hat{y}_i - \overline{\hat{y}})^2}};\\
\mathrm{SPCC}(y, \hat{y}) &= \frac{\sum_{i=1}^{n}(R_i - \overline{R})(S_i - \overline{S})}{\sqrt{\sum_{i=1}^{n}(R_i - \overline{R})^2}\sqrt{\sum_{i=1}^{n}(S_i - \overline{S})^2}};
\end{aligned}$$

where n is the number of samples; $\bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i$ and $\bar{\hat{y}} = \frac{1}{n}\sum_{i=1}^{n} \hat{y}_i$ denote their means, respectively; $R_i = \mathrm{rank}(y_i)$ and $S_i = \mathrm{rank}(\hat{y}_i)$ are the ranks of y_i and $\hat{y}_i$; and $\bar{R}$ and $\bar{S}$ denote their means.

Classification Metrics

$$\begin{aligned}
\text{Accuracy} &= \frac{TP + TN}{TP + TN + FP + FN};\\
\text{Precision} &= \frac{TP}{TP + FP};\\
\text{Recall/Sensitivity} &= \frac{TP}{TP + FN};\\
\text{Specificity} &= \frac{TN}{TN + FP};\\
\text{F1 Score} &= \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}};
\end{aligned}$$

where TP, TN, FP, and FN denote the number of true-positives, true-negatives, false-positives and false-negatives, respectively.

C Supplementary Tables and Figures

Feature Name	Description
Entry_ID	ID assigned to the interacting RNA-molecule pair
SMILES	Sequence representation of the molecule (atoms and bonds)
Target_RNA_sequence	Sequence representation of the target RNA
Molecule_name	Name of the molecule
Molecule_ID	ID assigned to the molecule
Target_RNA_name	Name of the target RNA
Target_RNA_ID	ID assigned to the target RNA
pKd	$-\log_{10}$ of the dissociation constant

Table 12: List of features included in the dataset provided by RSAPred [24].

RNA Subtype	No. of Features in Final Model	Training		Stratified 10-fold CV		Jack-Knife	
		PCC ↑	MAE ↓	PCC ↑	MAE ↓	PCC ↑	MAE ↓
Aptamers	12	0.730	0.619	0.718	0.607	0.703	0.631
Aptamers (reproduced)	12	0.696	0.623	0.687	0.642	0.677	0.639
miRNAs	8	0.897	0.408	0.881	0.434	0.882	0.434
miRNA (reproduced)	8	0.897	0.408	0.882	0.435	0.882	0.434
Repeats	13	0.921	0.315	0.894	0.367	0.889	0.372
Repeats (reproduced)	13	0.921	0.315	0.901	0.383	0.889	0.372
Ribosomal RNAs	11	0.831	0.663	0.836	0.687	0.810	0.698
Ribosomal RNAs (reproduced)	11	0.831	0.661	0.814	0.694	0.809	0.696
Riboswitches	13	0.923	0.466	0.909	0.537	0.896	0.541
Riboswitch (reproduced)	12	0.878	0.579	0.843	0.664	0.835	0.667
Viral RNAs	8	0.796	0.591	0.784	0.607	0.779	0.616
Viral RNAs (reproduced)	8	0.811	0.585	0.802	0.607	0.789	0.610

Table 13: Comparison of the performance reported in RSAPred [24] and the reproduced performance using their reported features selected on the whole dataset.

RNA Subtype	No. of Features in Final Model	Training		Stratified 10-fold CV		Jack-Knife	
		PCC ↑	MAE ↓	PCC ↑	MAE ↓	PCC ↑	MAE ↓
Aptamers	12	0.730	0.619	0.718	0.607	0.703	0.631
Aptamers (ours)	12	0.744	0.602	0.726	0.624	0.721	0.621
miRNAs	8	0.897	0.408	0.881	0.434	0.882	0.434
miRNAs (ours)	11	0.902	0.390	0.886	0.424	0.879	0.431
Repeats	13	0.921	0.315	0.894	0.367	0.889	0.372
Repeats (ours)	17	0.934	0.292	0.911	0.374	0.888	0.371
Ribosomal RNAs	11	0.831	0.663	0.836	0.687	0.810	0.698
Ribosomal RNAs (ours)	11	0.832	0.653	0.818	0.682	0.813	0.686
Riboswitches	13	0.923	0.466	0.909	0.537	0.896	0.541
Riboswitches (ours)	16	0.928	0.452	0.902	0.542	0.886	0.556
Viral RNAs	8	0.796	0.591	0.784	0.607	0.779	0.616
Viral RNAs (ours)	9	0.815	0.585	0.804	0.602	0.798	0.608

Table 14: Performance comparison of the features reported in RSAPred [24] and the features selected through our implementation of the feature selection algorithm.

RNA Model	**Mol Model**	**No. Frozen Layers**	**MAE ↓**
RNA-FM (LoRA)	Graph Diffusion	0	1.336
		4	1.349
		6	1.356
		8	1.349
		10	1.337
		12	1.394

Table 15: Ablation study of number of frozen layers not finetuned with LoRA.

RNA Model	**Mol Model**	r	**MAE ↓**
RNA-FM (LoRA)	Graph Diffusion	16	1.339
		32	1.336
		64	1.391

Table 16: Ablation study of latent LoRA size.

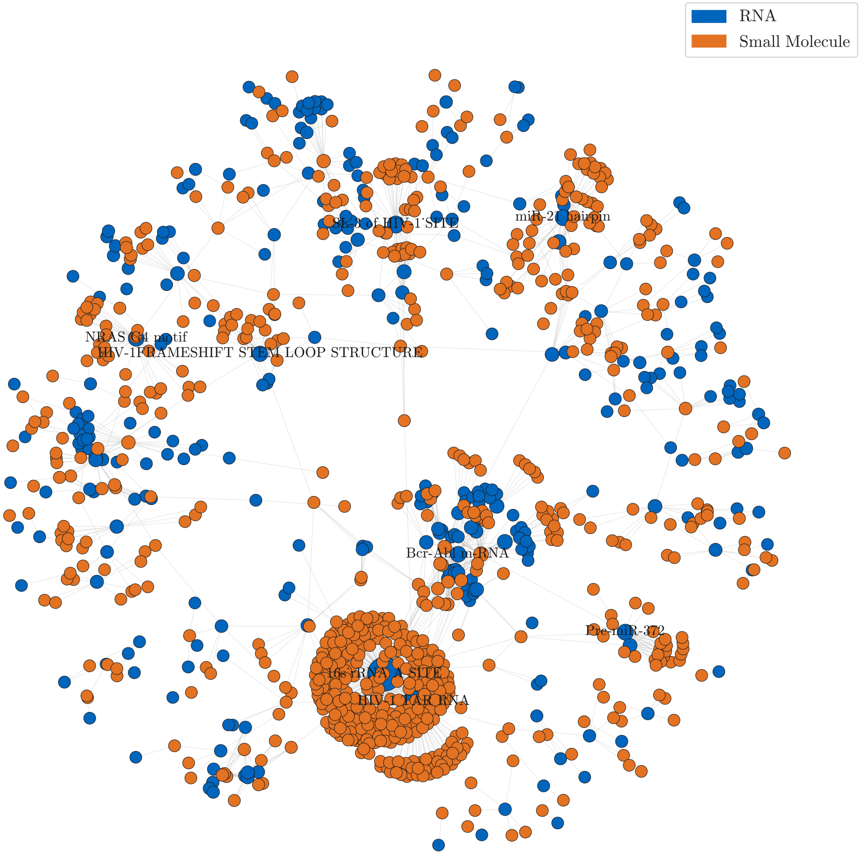


Figure 12: RNA-small molecule interaction network. Edges represent interacting pairs of RNAs (blue nodes) and small molecules (orange nodes).

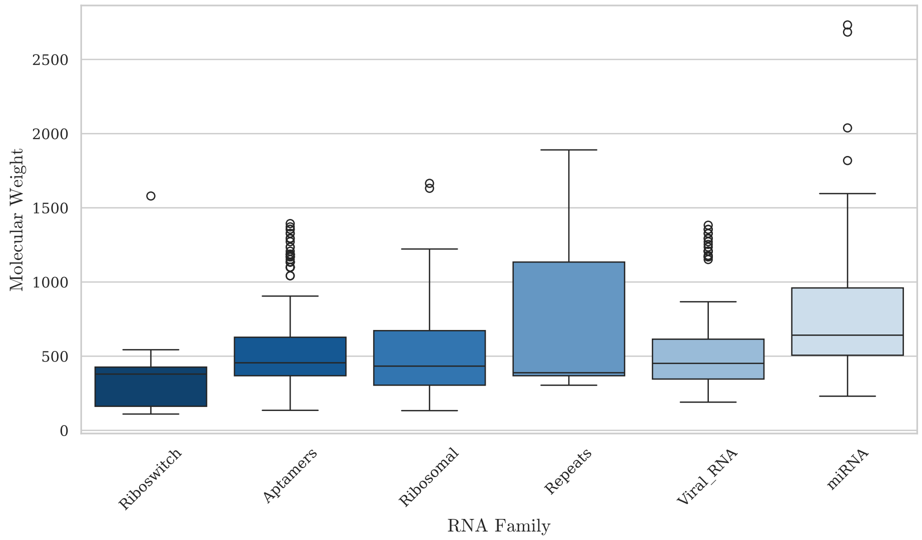


Figure 13: Boxplot: distribution of molecular weights in Daltons per RNA family.

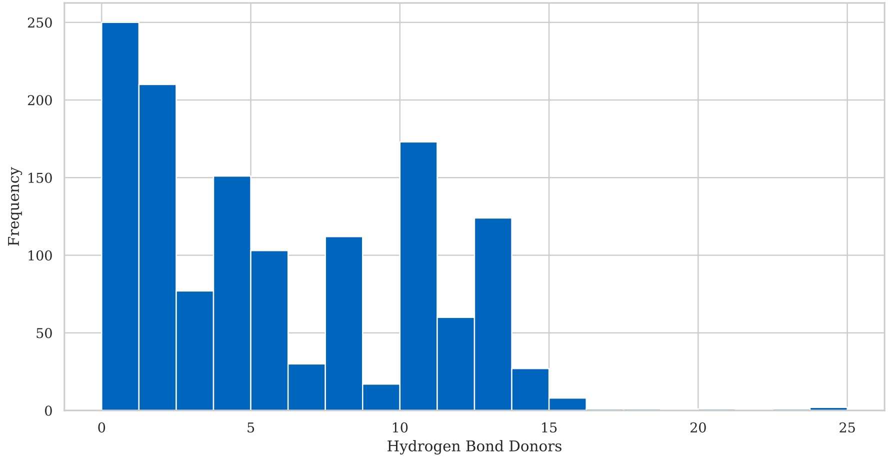


Figure 14: Distribution of the number of hydrogen bond donors (nitrogen–hydrogen, and oxygen-hydrogen bonds).

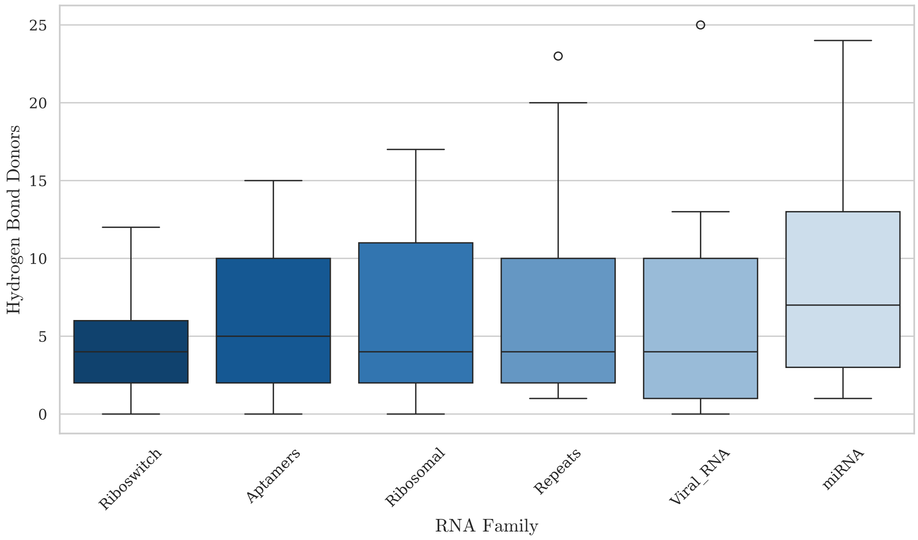


Figure 15: Boxplot: distribution of the number of hydrogen bond donors (nitrogen–hydrogen and oxygen–hydrogen bonds) per RNA family.

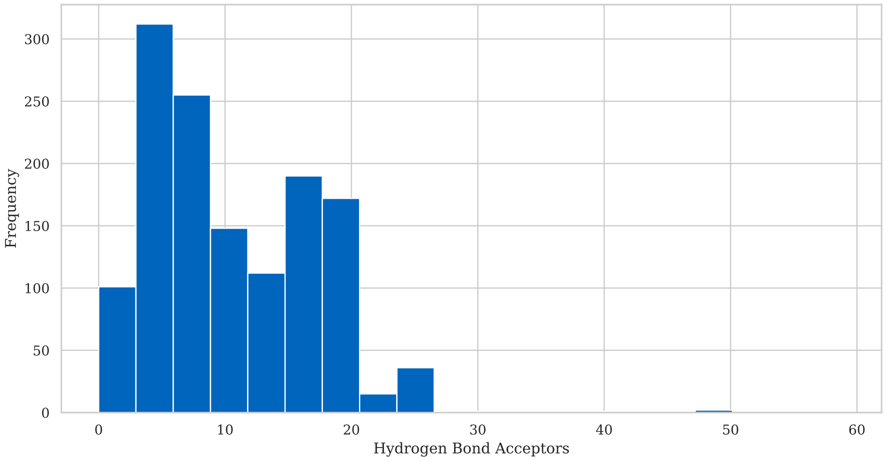


Figure 16: Distribution of the number of hydrogen bond acceptors (nitrogen or oxygen atoms).

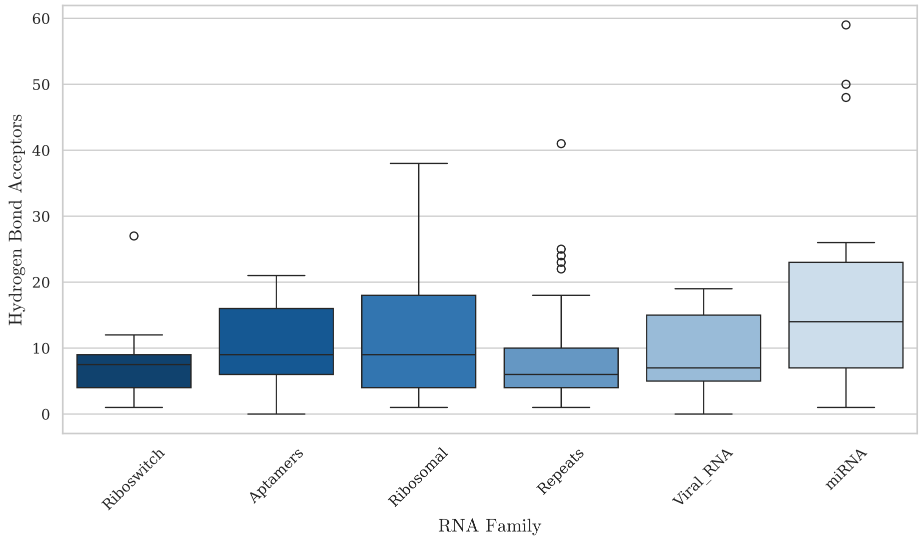


Figure 17: Boxplot: distribution of the number of hydrogen bond acceptors (nitrogen or oxygen atoms) per RNA family.

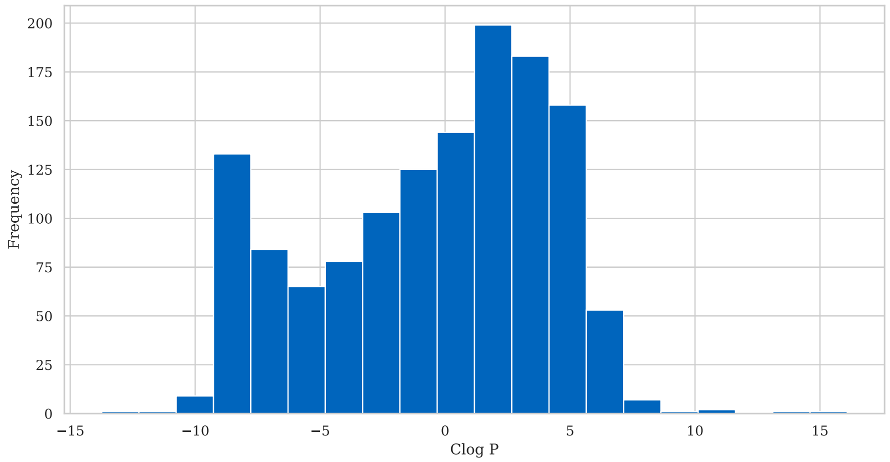


Figure 18: Distribution of the octanol-water partition coefficient (Clog P).

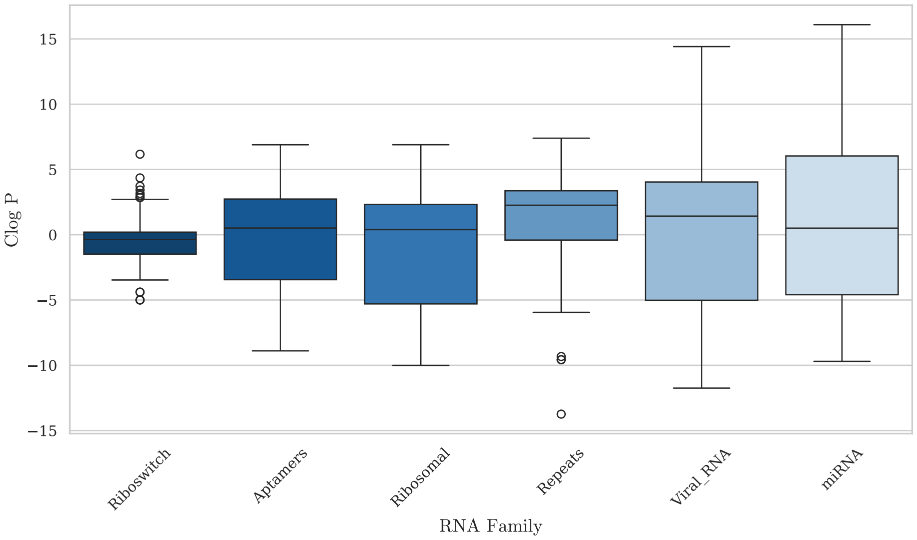
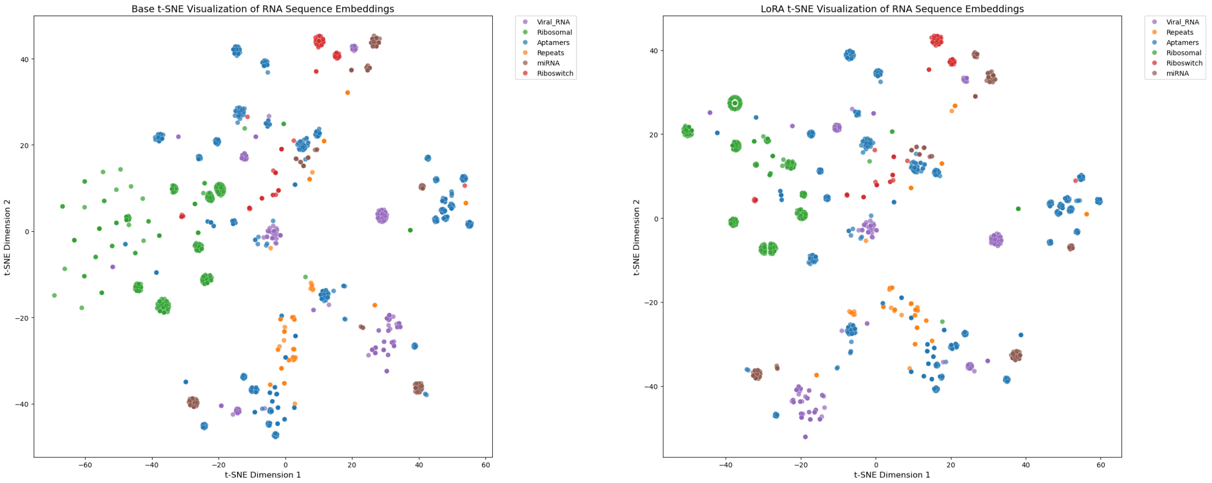


Figure 19: Boxplot: distribution of the octanol-water partition coefficient (Clog P) per RNA family.



(a) Pre-trained RNA-FM.

(b) After finetuning with LoRA.

Figure 20: RNA embeddings visualization with t-SNE.

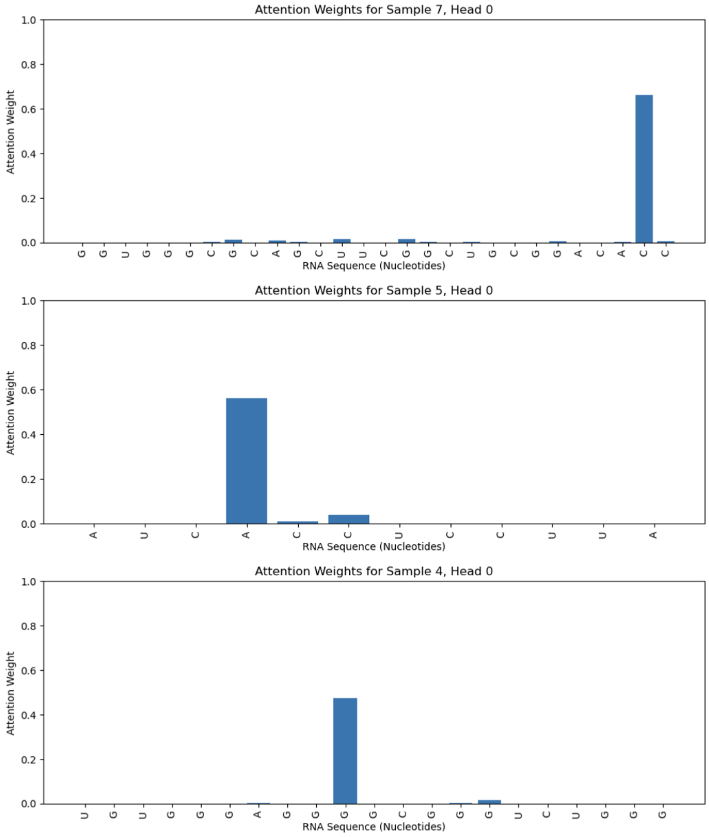


Figure 21: Attention weights as learned by combining Graph Diffusion molecule embeddings with RNA-FM embeddings. The learned attention weights are sparse.