
BAnG: Bidirectional Anchored Generation for Conditional RNA Design

Roman Klypa¹ Alberto Bietti² Sergei Grudin¹

Abstract

Designing RNA molecules that interact with specific proteins is a critical challenge in experimental and computational biology. Existing computational approaches require a substantial amount of experimentally determined RNA sequences for each specific protein or a detailed knowledge of RNA structure, restricting their utility in practice. To address this limitation, we develop RNA-BAnG, a deep learning-based model designed to generate RNA sequences for protein interactions without these requirements. Central to our approach is a novel generative method, Bidirectional Anchored Generation (BAnG), which leverages the observation that protein-binding RNA sequences often contain functional binding motifs embedded within broader sequence contexts. We first validate our method on generic synthetic tasks involving similar localized motifs to those appearing in RNAs, demonstrating its benefits over existing generative approaches. We then evaluate our model on biological sequences, showing its effectiveness for conditional RNA sequence design given a binding protein.

1. Introduction

Deep learning has significantly advanced bioinformatics and structural biology, particularly in predicting the structures, interactions, and functions of biomolecules (Callaway, 2024). It has also improved the efficiency of macromolecular design, facilitating applications in drug discovery and synthetic biology. In particular, significant progress has been made in protein sequence design. Some illustrative examples include ESM3 (Hayes et al., 2024) and Chroma (Ingraham et al., 2023). This remarkable progress has not only revolutionized protein design but also opened new opportu-

nities for addressing other complex biomolecular challenges. One such area is RNA generation, where similar principles of leveraging deep learning can be applied to advance our understanding and design of functional RNA sequences.

Among the many challenges in RNA design, generating RNA sequences capable of binding to specific proteins stands out as a critical task with significant implications for understanding RNA-protein interactions (Li et al., 2024; Fasogbon et al., 2024). These interactions, central to essential biological processes such as gene regulation, splicing, and translation (Hentze et al., 2018), highlight the need for precisely engineered RNA molecules. A notable example of such molecules are aptamers, short single-stranded RNA sequences that bind to specific proteins with high affinity and specificity. By acting as molecular inhibitors, probes, or delivery agents, aptamers offer versatile applications in therapeutics and diagnostics (Guo et al., 2010; Thavarajah et al., 2021). Traditionally, aptamers are identified through SELEX (Systematic Evolution of Ligands by Exponential Enrichment), a labor-intensive experimental process. Developing computational methods to design aptamers could significantly accelerate and simplify their discovery, expanding their potential in biomedical applications.

Several studies have explored RNA generation in this domain. More classical approaches exploited evolutionary signals and statistical models (Kim et al., 2007a;b; Aita & Husimi, 2010; Tseng et al., 2011; Zhang et al., 2023), molecular modeling (Torkamanian-Afshar et al., 2021), and Monte Carlo tree search (Lee et al., 2021; Wang et al., 2022; Shin et al., 2023; Obonyo et al., 2024). More recent works used conditional variation autoencoders (Chen et al., 2022; Iwano et al., 2022; Andress et al., 2023), long short-term memory models (Im et al., 2019; Park & Han, 2020), transformer-based architectures (Zhao et al., 2024; Zhang et al., 2024), and adversarial approach (Ozden et al., 2023). Most recent studies, e.g., AptaDiff (Wang et al., 2024), or RNAFLOW (Nori & Jin, 2024) also explored diffusion processes and flow matching. Almost all of the aforementioned approaches depend on a vast collection of nucleotide sequences known to interact with proteins to generate new ones, limiting their applicability to proteins for which extensive experimental data is available. To the best of our knowledge, RNAFLOW stands out as the only method that has not been trained on RNA affinity experimental data. However, it relies on RNA

¹ Univ. Grenoble Alpes, CNRS, Grenoble INP, LJK, 38000 Grenoble, France. ²Center for Computational Mathematics, Flatiron Institute, 162 5th Ave, New York, NY 10010, USA. Correspondence to: Roman Klypa <roman.klypa@univ-grenoble-alpes.fr>, Alberto Bietti <abietti@flatironinstitute.org>, Sergei Grudin <sergei.grudin@univ-grenoble-alpes.fr>.

structure prediction tools to guide RNA design. Since these tools often lack the accuracy needed for precise RNA structures (Rhiju et al., 2024), they ultimately reduce the model’s effectiveness in many practical scenarios.

In this work, we present a novel generative method combined with a deep-learning model that operates without relying on specific experimental data for the target protein and does not depend on RNA structural information. This approach enables a broader applicability and greater efficiency in RNA sequence generation than the ones mentioned above.

The motivation for the design of the proposed generative method stems from two key observations. First, the total length of the RNA sequence to be generated is often unknown. Therefore, the method adopts an autoregressive approach. Second, RNA sequences that interact with proteins typically contain functional binding motifs — specific regions that mediate interaction by forming molecular contacts with the protein. These binding motifs are embedded within larger sequence contexts, where the surrounding non-binding regions exert lesser influence on binding specificity. This makes it more effective to initiate sequence generation from the binding motif, rather than from the sequence’s ends, as is commonly done in current state-of-the-art NLP autoregressive models. These are the core ideas behind our method, Bidirectional ANchored Generation (BAnG).

The model, named RNA-BAnG, is based on a transformer architecture with geometric attention. The latter allows for the incorporation of protein structural information, which is crucial for predicting RNA-protein interactions. By utilizing AlphaFold2 (Jumper et al., 2021), a state-of-the-art protein structure prediction tool, we can obtain highly accurate structural data, even if the target protein is not solved experimentally. The resulting combination of the RNA-BAnG model and the generative method, schematically illustrated in Figure 1, produces RNA sequences that interact with a given protein, utilizing both its sequence and structural information. Our main contributions can be summarized as follows:

1. We propose a new bidirectional generation method, BAnG, along with a transformer-based architecture, RNA-BAnG, that are well-suited for RNA generation conditioned on a binding protein.
2. We thoroughly validate the effectiveness of our method on relevant synthetic tasks and compare it with other widely used sequence generation methods.
3. We evaluate our approach on experimental RNA-protein interaction data, showing promising results that outperform previous methods.

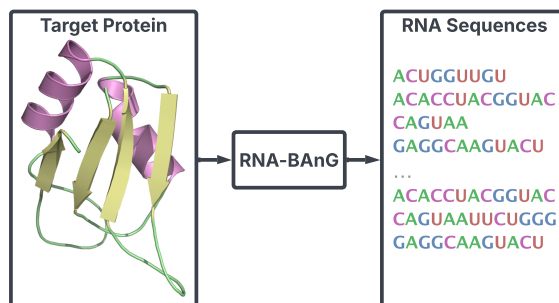


Figure 1. Schematic illustration of the RNA-BAnG generative process and its conditioning on the input protein 3D structure. The protein model was generated by AlphaFold2 and is colored by the secondary structure. RNA sequences are colored by nucleotides.

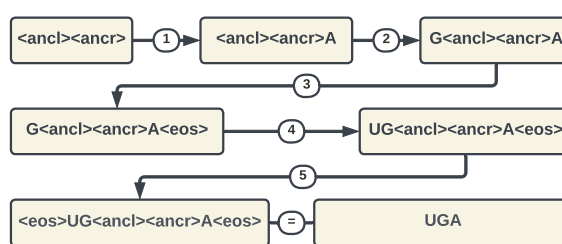


Figure 2. A step-by-step example process of RNA sequence 'UGA' generation.

2. Bidirectional anchored generation for RNA

In this section, we present our generative modeling approach, BAnG, and its application to conditional RNA sequence prediction through the RNA-BAnG model.

2.1. Description of the BAnG generative approach

In the BAnG framework, we leverage the following factorization of the joint distribution over a sequence $x = (x_{-n-1}, \dots, x_0, \dots, x_n)$:

$$P(x) = P(x_0) \times \underbrace{\prod_{i=0}^n P(x_{-i-1} | x_{-i \dots i})}_{\text{left tokens}} \underbrace{\prod_{i=1}^n P(x_i | x_{-i \dots i-1})}_{\text{right tokens}}, \quad (1)$$

where x_0 is the first generated token. Concretely, sequence generation begins with two special anchor tokens, <ancl> and <ancr>, representing the left and right boundaries of the sequence, respectively. Tokens are then generated one at a time, alternating between directions: we first sample a token on the right, then a token on the left, and repeat this

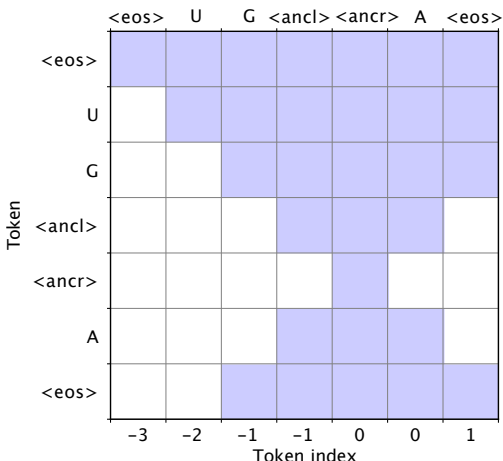


Figure 3. BAnG attention mask. Anchor tokens are indexed in a way to preserve relative distances in a sequence after anchors insertion.

process to progressively extend the sequence outward. At each step, the distribution of the next token is conditioned on already generated ones, following (1). If an end-of-sequence (<eos>) token is generated in either direction, no further tokens are produced along that axis. The generation process stops when <eos> tokens are produced for both boundaries or when a predefined maximum sequence length is reached (see Figure 2).

BAnG enables training a deep learning model to estimate the conditional distributions in (1) in order to perform bidirectional generation in the described manner, using a process analogous to autoregressive training. The key difference in the architecture compared to the standard autoregressive case is the replacement of the conventional lower triangular attention mask with a specifically designed bidirectional attention mask, shown in Figure 3. This custom mask ensures that any representation of a given token cannot depend on tokens beyond those appearing in the corresponding conditional in (1). This simple modification allows the model to learn the dependencies needed for our generation strategy, while ensuring efficient parallelization of the forward and backward passes across all tokens in a sequence during training. During model training and inference, the probabilities for the next token in each direction are derived from the embedding of the most recently generated token in the same direction, as shown schematically in Figure 4.

We shall emphasize that the single pass training with the conditional factorization in (1) is only possible thanks to the introduction of two anchor tokens. Indeed, if only a single anchor token was used, it would be responsible for predicting the first tokens in both directions, right and left. To prevent information leakage, its attention would be re-

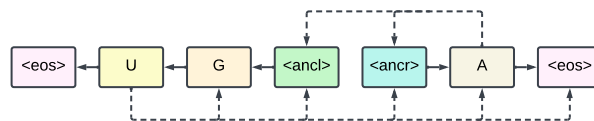


Figure 4. Schematic illustration of the masked attention mechanism and token probabilities derivation. Bold lines indicate the token embeddings from which the probabilities for each token are derived, while the dotted lines represent the tokens to which the given token’s attention is directed.

stricted to itself, which would make the prediction of the left token independent of the right one. This lack of conditioning could lead to the generation of incompatible token pairs.

2.2. Model

RNA-BAnG architecture consists of two main components: a protein module and a nucleotide module. The protein module derives a representation from the protein’s sequence and structure, while the nucleotide module generates a nucleotide sequence conditioned on this representation.

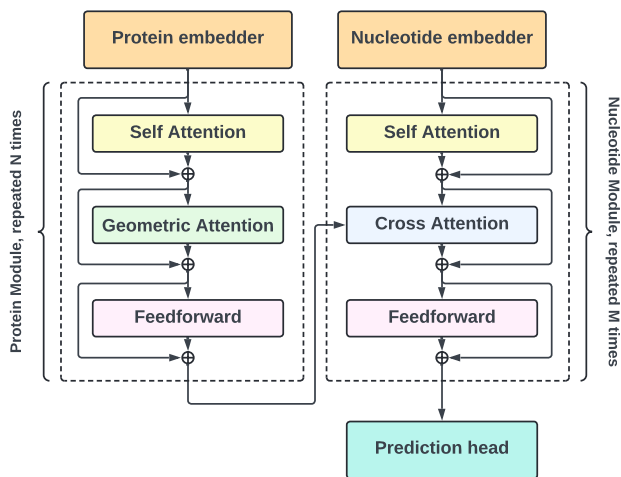


Figure 5. Schematic illustration of the RNA-BAnG architecture. The protein component is on the left, the nucleotide component is on the right. More details can be found in Appendix A.

Our model’s modules comprise several main blocks - **Embedder**, **Self Attention**, **Geometric Attention**, and **Cross Attention**, schematically illustrated in Figure 5. The **Embedder** block generates token embeddings from protein, RNA, and DNA sequences. The inclusion of RNA and DNA sequences in the training data serves to augment the dataset, as it can help the model learn the shared patterns.

The **Embedder** block processes amino acids with one-hot encoding residue types (20 canonical and a padding token), followed by a linear transformation and normalization of the resulting embeddings. For nucleic acids, it encodes sequence information in two steps. First, it one-hot encodes residue types, using tokens for the four standard RNA nucleotides, along with two anchor tokens, `<eos>` token, and a padding (`<pad>`) token. DNA residues are treated as RNA equivalents, with thymine (T) replaced by uracil (U). Second, the sequence type (DNA or RNA) is one-hot encoded, linearly transformed, and concatenated with the residue embeddings. This combined representation is then passed through a normalization layer.

Self-attention in the model was implemented in a classical way (Vaswani et al., 2023) using Rotary Position Embedding (RoPE) (Su et al., 2023). While RoPE has recently been predominantly used in autoregressive models, it was originally introduced for bidirectional transformers, making it particularly suitable for our case. Self-attention for protein sequences uses unmasked attention, while for nucleotide sequences, the BAnG mask is applied.

Cross-attention is implemented similarly to the self-attention mechanism, with one key difference: instead of using RoPE, sinusoidal positional embeddings (Vaswani et al., 2023) are applied to the nucleotide sequence. Specifically, we encoded the positions of nucleotide tokens relative to the anchor ones, where the anchor token `<anc1>` is assigned an index of zero. This choice was made because residues in different chains do not have relative sequential distances, but we still want to include nucleotide position information in the calculation of its attention to the protein.

Geometric Attention aims to incorporate protein structural information. To achieve this, we adopted the protein representation introduced in AlphaFold2 (Jumper et al., 2021). Specifically, for each protein residue, a rigid frame T is constructed based on the coordinates of its C , C_α , and N backbone atoms. Attention between these frames is then calculated using the geometric part from AlphaFold2’s Invariant Point Attention (IPA) mechanism (more details in Appendix A). The importance of this module is demonstrated by the fact that RNA-BAnG fails to converge in its absence.

We use GELU (Hendrycks & Gimpel, 2016) as the activation function in the Feed Forward layers. For normalization, we employed RMSNorm (Zhang & Sennrich, 2019). The number of attention heads is consistent across all attention blocks, and the dimension of each head is set independently of other model parameters. In both self-attention and cross-attention mechanisms, keys and queries are normalized prior to multiplication to prevent the attention-logit growth instability (Dehghani et al., 2023). The specific choice of hyperparameters is described in Appendix A.

3. Validating BAnG on a synthetic task

In this section, we evaluate the effectiveness of our BAnG strategy through a thorough analysis on a relevant synthetic task involving distinct subsequences.

Description of the task. To evaluate the BAnG method, we first consider a synthetic task that emulates a real-world conditional generation scenario. As mentioned above, RNA sequences often include a functional binding motif within a broader sequence context, where non-binding regions contribute minimally to binding specificity (Ray et al., 2013). Based on this observation, the task’s objective is to generate sequences that contain a predefined short subsequence, *synthetic motif*.

Synthetic data. The synthetic data consists of nucleotide sequences, each 50 residues long, with a synthetic motif placed at a random position. The remaining residues are uniformly distributed. We fixed two different random subsequences of length 6 as the synthetic motifs. The exact content of these motifs is not crucial to the task, as any possible subsequences of this length are equally likely to be uniformly sampled. The length of six was chosen because it closely matches the size of real binding motifs (Ray et al., 2013) and reduces the likelihood of random occurrences of such subsequences. As the anchor point for BAnG, we chose the center of the synthetic motif.

In the *SingleBind* training setup, sequences contained only the first synthetic motif, while in the *DoubleBind* setup, sequences could contain either the first or the second synthetic motif with equal probability. The objective of the *SingleBind* setup was to compare generative methods on a simpler task, while the goal of *DoubleBind* was to assess their performance under more realistic, thus more uncertain conditions.

Reference methods. We compare our method to existing generative approaches, including autoregressive generation, which is commonly used in natural language processing (Bengio et al., 2000), and iterative generation methods based on masked language modeling, as implemented in ESM3 (Hayes et al., 2024). Also, as the simplest baseline, we include in the comparison set of random sequences, where each token is sampled from a uniform distribution. For the autoregressive approach, we trained the model using a lower triangular attention mask. In the iterative approach, the model was trained with a demasking objective and a high masking rate of 50%, which, according to the ESM3 authors, has been shown to yield effective generation results. Additional model training details for each tested method can be found in Appendix B.1.

For the autoregressive approach, tokens are generated se-

Table 1. Comparison of generative methods on the SingleBind and DoubleBind tasks. Values represent the proportion of sequences that contain the correct synthetic motif.

GENERATIVE METHOD	SINGLEBIND \uparrow	DOUBLEBIND \uparrow
BANG	0.98	0.97
AUTOREGRESSIVE	0.94	0.53
IANG ENTROPY	0.91	0.54
IANG LOGIT MAX	0.89	0.54
ITERATIVE ENTROPY	0.06	0.04
ITERATIVE LOGIT MAX	0.04	0.05
RANDOM SEQUENCES	0.01	0.02

quentially, one at a time, starting from the start-of-sequence ($\langle\text{sos}\rangle$) token and continuing until the $\langle\text{eos}\rangle$ token is produced. In the iterative approach, however, all tokens but $\langle\text{sos}\rangle$ and $\langle\text{eos}\rangle$ are initially masked. Tokens are then unmasked one by one, with the next token to unmask chosen based on either the largest logit value (max logit decoding) or the smallest entropy (entropy decoding). Sampling details may be found in Appendix B.2.

We also introduced a modification of iterative methods, better suited for the task - Iterative Anchored Generation (IANG). This approach merges BAnG with iterative methods by incorporating an anchor token ($\langle\text{anc}\rangle$) placed in the middle of the synthetic motif. In this method, the anchor token remains unmasked during training. At the start of the inference, the anchor token is positioned at a random location within the sequence and remains unmasked throughout the process.

Evaluation results. With each tested approach we generated 1,000 sequences. Table 1 summarizes the performance of each method. Additional statistics on the frequency of each synthetic motif in the generated data for DoubleBind task are provided in Table 2. Examples of generated sequences can be found in Appendix B.3.

The tables show that BAnG outperforms other methods, with a particularly notable margin on the DoubleBind task. BAnG also generates fewer sequences containing mixed synthetic motifs. The very low performance of ESM3’s iterative methods is expected, as the lack of absolute positional dependencies in the data causes the model to assign nearly uniform probabilities across tokens. This reasoning is further supported by the significant performance improvement observed when positional information is introduced through the anchor token in IANG. Nonetheless, both methods suffer from a key limitation: a discrepancy between the demasking process during training and inference, which undermines their overall effectiveness.

Iterative methods may demonstrate improved performance

Table 2. Detailed statistics for the DoubleBind task: proportion of sequences containing either one of the synthetic motifs or both.

GENERATIVE METHOD	FIRST \uparrow	SECOND \uparrow	BOTH \downarrow
BANG	0.43	0.53	0.01
AUTOREGRESSIVE	0.17	0.25	0.11
IANG LOGIT MAX	0.09	0.44	0.01
IANG ENTROPY	0.10	0.44	0.01
ITERATIVE ENTROPY	0.03	0.02	0.01
ITERATIVE LOGIT MAX	0.01	0.04	0.01
RANDOM SEQUENCES	0.01	0.01	0

in *modality translation* scenarios, such as structure-to-sequence generation, which is a key focus of ESM3. However, they are unlikely to match the effectiveness of regressive approaches in purely generative tasks.

4. Conditional RNA generation

In this section, we evaluate our RNA-BAnG modeling strategy for protein-conditioned generation of RNA sequences based on experimental biological data.

4.1. Data

We collected our protein-nucleotide interaction data from the Protein Data Bank (PDB) (Berman, 2000), utilizing information provided in the PPI3D database (Dapkūnas et al., 2024). However, we conducted independent postprocessing of the data, distinct from PPI3D, as they focus on structural RNA and DNA information, while we are concerned solely with their sequences. The postprocessing steps involved verifying chain interactions, discarding ambiguous protein structures, and excluding chains containing non-standard residues, as explained in more detail in Appendix C.1. During training and validation, each time a sample was encountered, its anchor point was randomly sampled from the interacting nucleotides. The anchor tokens were then inserted right after the selected nucleotide. Additionally, to diversify RNA sequence information, we collected non-coding sequences from RNACentral (release 24), a comprehensive database integrating RNA sequences from multiple expert sources (The RNACentral Consortium et al., 2019) (more details in Appendix C.2). Since we lack information about their interactions, the anchor points for solo RNA sequences were selected randomly.

4.2. RNA-BAnG training

We designed the training process to consist of two steps: the first for the model to learn general information about RNA sequences, and the second for the model to learn conditioning on proteins. In both steps, the training objective is the cross-entropy loss between the predicted and ground

truth nucleotide token probabilities (Bengio et al., 2000). As the first step we train RNA-BAnG nucleotide module without cross-attention block on standalone RNA sequence data from RNACentral. During training, the loss values for the four tokens closest to each anchor on either side are weighted at 0.01. This weighting scheme is applied because the model lacks the context needed to accurately predict these residues. Next, we train the full model, using weights from the previous step, on the combined protein-nucleotide sequence data, this time without applying any loss weighting. Additional training details can be found in Appendix D.

4.3. Evaluation protocol

To evaluate the performance of our model, we adopted the scoring approach proposed by the authors of GenerRNA (Zhao et al., 2024) and others (Im et al., 2019). This method leverages DeepCLIP (Grønning et al., 2020), a state-of-the-art predictive model that, after being trained on examples of interacting and non-interacting RNA sequences for a given protein, can assign binding probabilities to any RNA sequence with the same protein. These probabilities serve as a proxy for evaluating the quality of sequences generated by RNA-BAnG and other methods.

To obtain suitable training and testing data for DeepCLIP, we utilized RNACompete experiments (Ray et al., 2009), conducted by the authors of the RNA Compendium (Ray et al., 2013). Using this data, we trained DeepCLIP models for each sample and selected only those that met our performance criteria (see Appendix E.1). Separately, for each sample, we derived 1,000 sequences identified by RNACompete as highly likely to bind to the protein (positive examples) and another 1,000 sequences as highly unlikely to bind (negative examples), based on their experimental affinity scores (see Appendix E.1). We excluded these sequences from the DeepCLIP model training, ensuring that they served as an independent benchmark for testing the quality of both the DeepCLIP predictions and the RNA sequences generated by RNA-BAnG. Below, we refer to them as to the *positive* and *negative* experimental sets.

Due to the lack of experimentally solved protein structures for the test set, and aiming for a more robust method, we only used AlphaFold2 protein models for RNA-BAnG inference during evaluation. Concretely, for each RNA Compendium sample, we generated the corresponding three-dimensional protein model, and retained for further testing only those with a predicted local distance difference test (pLDDT) score greater than 70%. This threshold is well-accepted in the community and guarantees structural reliability of the produced protein structures, at least at the domain level. The final test set consisted of 71 samples, representing 67 unique protein sequences.

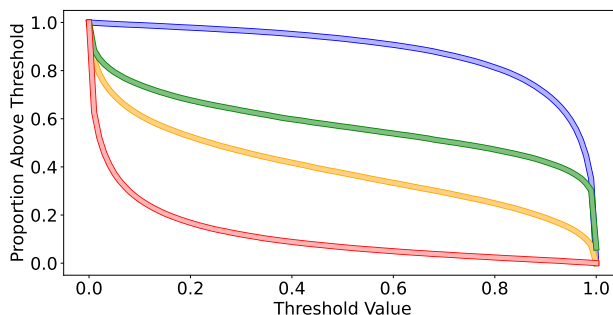


Figure 6. Proportion of sequences above the threshold: generated by RNA-BAnG (green) and randomly (yellow), the positive (blue) and negative (red) experimental sets. The values here represent the averages for the entire test set.

To match the amount of RNA sequences in experimental sets, we generated 1,000 of them with RNA-BAnG for each test sample. We also generated 1,000 random sequences to serve as the simplest baseline (see Appendix E.2).

4.4. Experiment results

To estimate RNA-BAnG’s performance, we use the proportion of sequences with DeepCLIP scores above a given threshold, rather than relying on the mean score for each test sample. This approach aligns better with practical applications, where the goal is often to maximize the number of sequences that meet a specific performance criterion, rather than optimizing for average performance across the entire set.

Statistics of generated sequences. Figure 6 presents the threshold-dependent performance curve, showing the relationship between sequences with DeepCLIP scores above a certain threshold and the threshold value. The area under the threshold-dependent performance curves in Figure 6 captures the method’s ability to generate affine sequences. A larger area indicates better performance, reflecting a higher proportion of sequences that exceed the threshold at different points. A high area value of 0.88 for the positive experimental set indicates that DeepCLIP is good at distinguishing high-affinity sequences from others. RNA-BAnG showed the value of 0.57, indicating moderate success in generating sequences that somewhat align with the positive set. The random set, with the area value of 0.39, suggests that our model is outperforming random generation. The negative set’s very low area value of 0.11 further underscores the DeepCLIP good performance, also suggesting that RNA-BAnG has some ability to avoid producing low-affinity outputs.

Averaging across samples conceals important details about

the statistics of the compared sequences. For a more detailed, sample-by-sample analysis, shown in Figure 7, we selected a threshold value of 0.75, as DeepCLIP scores around 0.5 can be ambiguous in interpretation. We consider sequences with DeepCLIP scores above this value as *high-affinity* RNA sequences (haRNA). As one can see from Figure 7, RNA-BAnG generates more high-affinity sequences than a random generator for 56 out of 71 test samples. For 33 of the test samples, more than half of the sequences generated by RNA-BAnG are the high-affinity ones. Additionally, our model generated less than 20% high-affinity sequences for only 14 out of 71 test samples. The high success rate of randomly generated sequences for some samples may be attributed to the low complexity of captured protein binding motifs (Ray et al., 2013). The examples of generated sequences for some representative samples and individual DeepCLIP score distributions can be found in Appendix F.

The performance of our model shows no significant correlation with the predicted quality of the protein 3D model (pLDDT) or with the sequence similarity of a protein target to the training data (Figure F.1 in Appendix F). This analysis suggests that the model does not only memorize the information, but exhibits a degree of generalization. We can also conclude that the performance accuracy of RNA-BAnG can not be attributed to protein structure prediction quality, if its pLDDT score is higher than 70%.

Novelty and diversity of generated sequences. For our model’s goal of accelerating experimental RNA design, diversity and novelty are essential metrics. By generating a wide range of diverse sequences, the model increases the chances of identifying optimal binders with varying affinities for the target protein, ensuring more effective candidates for experimental validation. Furthermore, the generation of novel sequences allows the discovery of unique RNA-protein interactions, which can lead to innovative therapeutic applications.

To quantify the diversity metric, we compute the ratio of the number of distinct clusters at a threshold of 0.9 sequence identity to the total number of generated sequences. A selected threshold of 0.9 is often considered as one of the lower limits used in RNA sequence clustering, especially when aiming to identify highly similar sequences or gene families (Edgar, 2018). The resulting average diversity across the test set is 0.93 ± 0.13 , indicating that the generated sequences are highly varied. The novelty is defined by the proportion of RNA sequences in the generated set that are not similar to the training data. We identify sequences for each test sample that have no similarity (see Appendix G for more details) to the training set, and the novelty is calculated as the ratio of these sequences to the total number of generated sequences. The resulting average novelty across the test set

Table 3. High-affinity RNA sequence proportion generated by GenerRNA and RNA-BAnG for each RNA Compendium sample. The protein sequences are identical for the same genes. Although underlined samples do not meet our test set selection criteria, they are included to enhance the diversity of the comparison. Sample IDs mapping to RNAcompete IDs mentioned in Appendix E.1.

SAMPLE ID	GENE	GENERRNA	RNA-BANG
<u>106</u>	SRSF1	0.42	0.49
107	SRSF1	0.59	0.74
108	SRSF1	0.56	0.77
109	SRSF1	0.62	0.85
<u>110</u>	SRSF1	0.43	0.66
<u>121</u>	ELAVL1	0.64	0.91

is 0.99 ± 0.01 , indicating that the generated sequences are highly novel and distinct from the training data, which is a positive outcome for our model.

Comparison with similar models. We compare our model, RNA-BAnG, with two other methods, GenerRNA (Zhao et al., 2024) and RNAFLOW (Nori & Jin, 2024), as they are the most relevant existing methods for generating RNA sequences for a diverse set of proteins. The first model, GenerRNA, leverages a substantial collection of RNA sequences known to interact with a specific protein during its fine-tuning process, allowing it to generate additional sequences with similar binding properties. Consequently, our comparison is limited to the proteins for which GenerRNA was fine-tuned by its authors. We used their published inference results, removing generated RNA larger than 50 nucleotides, leaving 921 and 909 sequences for the ELAVL1 and SRSF1 proteins, respectively. These proteins are already present in the RNA Compendium samples. As shown in Table 3, our model generates more high-affinity RNA sequences than GenerRNA. It is important to note that RNA-BAnG generates these sequences without relying on any additional information, whereas GenerRNA required extensive data mining from multiple experimental studies.

The second baseline model is RNAFLOW, which uses RNA structure prediction tools. Due to its much lower speed, approximately 50 times slower than RNA-BAnG, we only generated RNA sequences for proteins that had zero sequence similarity with our training set, resulting in 100 sequences of length 50 for each protein. Unfortunately, RNAFLOW generated sequences with an unusually high frequencies of G and C nucleotides — 62% and 32%, respectively — regardless of protein structure or sequence. As a result, it produced high-affinity sequences for only a couple of proteins, and none for the others. Consequently, RNA-BAnG outperformed RNAFLOW on most of the test samples (Figure 8).

The low performance of RNAFLOW can be explained by

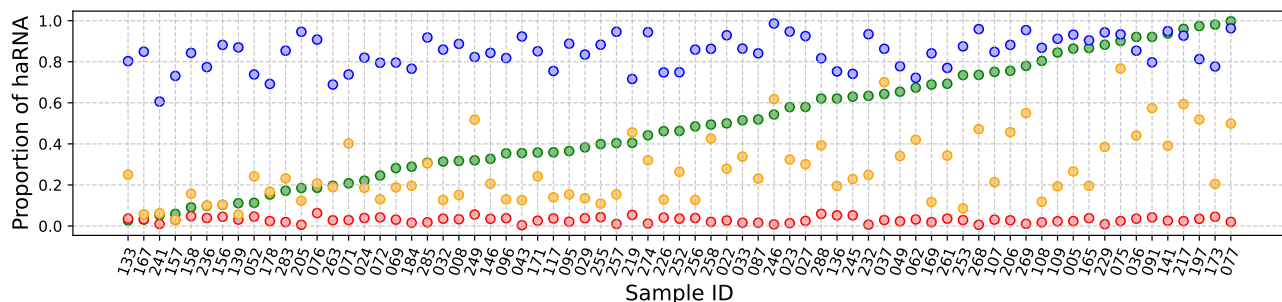


Figure 7. Proportions of high-affinity RNA sequences for each test sample, generated by RNA-BAnG (green) and randomly (yellow). These are compared with the proportions in the positive (blue) and negative (red) experimental sets. Test samples are ordered by RNA-BAnG performance. Sample IDs mapping to RNAComplete IDs mentioned in Appendix E.1.

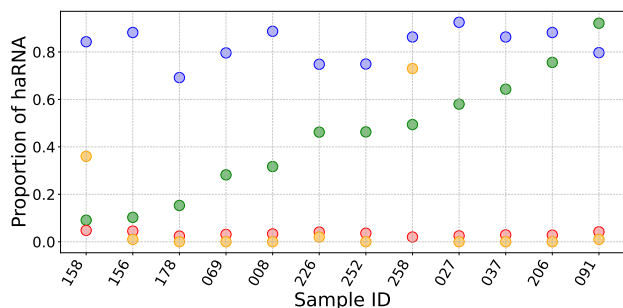


Figure 8. Proportion of high affinity sequences generated by RNA-BAnG (green) and RNAFLOW (yellow). Samples are sorted by the performance of RNA-BAnG. Sample IDs mapping to RNAComplete IDs mentioned in Appendix E.1.

a discrepancy between its training process and the inference we conducted. Concretely, RNAFLOW was trained on protein sequences and structures truncated to 50 residues, ensuring the inclusion of the protein binding site. However, we tested it on proteins of varying lengths, ranging from 177 to 589 amino acids, without highlighting their binding sites through truncation (since this information is unavailable). Additionally, the proteins we tested had zero sequence similarity to those in RNAFLOW’s training set (PDB), making it more challenging to predict their structure in complex with RNA. These limitations significantly restrict the applicability of RNAFLOW to RNA sequence design, especially when compared to RNA-BAnG.

5. Conclusion

This study introduces a novel deep-learning model, RNA-BAnG, and a sequence generation method, BAnG, for designing RNA sequences that bind to a given protein. Unlike previous approaches, our model demonstrates remarkable

flexibility by eliminating the need for extensive structural or interaction data. Although our model relies on protein structure, AlphaFold predictions make sequence information sufficient. This innovation significantly broadens the applicability of our method, making it a versatile tool for RNA-protein interaction studies.

Our method is based on the observation that RNA sequences contain functional binding motifs, while the surrounding sequence context is less critical for interaction. When evaluated on a synthetic task against other sequence generation approaches, BAnG demonstrated superior performance. Importantly, the method’s design makes it applicable beyond RNA-protein interactions, extending to any scenario where the focus is on optimizing functional subsequences within a larger sequence. According to the state-of-the-art DeepCLIP scoring, RNA-BAnG outperforms existing methods, generating a higher proportion of sequences with strong predicted binding affinity. The generated sequences exhibit both diversity and novelty, expanding the range of potential RNA candidates for further experimental validation.

Future work could focus on integrating experimental feedback to further refine the model, optimizing its architecture for enhanced performance, and improving its usability for broader applications. Additionally, experimental validation of the generated sequences would provide further insights into the practical applicability of our method. In summary, our work represents a significant step forward in RNA sequence generation, offering a powerful and flexible tool for researchers in the field of RNA-protein interactions.

Software and Data. The code and the model, along with the model weights, will be available upon publication.

Acknowledgments

This work was granted access to the HPC resources of IDRIS under the allocation 2024-AD011015647 made by GENCI. A substantial part of the computations presented in this paper was performed using the GRICAD infrastructure (<https://gricad.univ-grenoble-alpes.fr>), which is supported by Grenoble research communities.

References

- Aita, T. and Husimi, Y. Biomolecular information gained through in vitro evolution. *Biophysical reviews*, 2:1–11, 2010. Publisher: Springer.
- Altschul, S. F., Gish, W., Miller, W., Myers, E. W., and Lipman, D. J. Basic local alignment search tool. *Journal of Molecular Biology*, 215(3):403–410, October 1990. ISSN 00222836. doi: 10.1016/S0022-2836(05)80360-2. URL <https://linkinghub.elsevier.com/retrieve/pii/S0022283605803602>.
- Andress, C., Kappel, K., Villena, M. E., Cuperlovic-Culf, M., Yan, H., and Li, Y. DAPTEV: Deep aptamer evolutionary modelling for COVID-19 drug design. *PLOS Computational Biology*, 19(7):e1010774, July 2023. ISSN 1553-7358. doi: 10.1371/journal.pcbi.1010774. URL <https://dx.plos.org/10.1371/journal.pcbi.1010774>.
- Bengio, Y., Ducharme, R., and Vincent, P. A Neural Probabilistic Language Model. In Leen, T., Dietterich, T., and Tresp, V. (eds.), *Advances in Neural Information Processing Systems*, volume 13. MIT Press, 2000. URL https://proceedings.neurips.cc/paper_files/paper/2000/file/728f206c2a01bf572b5940d7d9a8fa4c-Paper.pdf.
- Berman, H. M. The Protein Data Bank. *Nucleic Acids Research*, 28(1):235–242, January 2000. ISSN 13624962. doi: 10.1093/nar/28.1.235. URL <https://academic.oup.com/nar/article-lookup/doi/10.1093/nar/28.1.235>.
- Callaway, E. Chemistry Nobel goes to developers of AlphaFold AI that predicts protein structures. *Nature*, 634(8034):525–526, October 2024. ISSN 0028-0836, 1476-4687. doi: 10.1038/d41586-024-03214-7. URL <https://www.nature.com/articles/d41586-024-03214-7>.
- Chen, J. C., Chen, J. P., Shen, M. W., Wornow, M., Bae, M., Yeh, W.-H., Hsu, A., and Liu, D. R. Generating experimentally unrelated target molecule-binding highly functionalized nucleic-acid polymers using machine learning. *Nature Communications*, 13(1):4541, August 2022. ISSN 2041-1723. doi: 10.1038/s41467-022-31955-4. URL <https://www.nature.com/articles/s41467-022-31955-4>.
- Dapkūnas, J., Timinskas, A., Olechnovič, K., Tomkuvienė, M., and Venclovas, C. PPI3D: a web server for searching, analyzing and modeling protein–protein, protein–peptide and protein–nucleic acid interactions. *Nucleic Acids Research*, 52(W1):W264–W271, July 2024. ISSN 0305-1048, 1362-4962. doi: 10.1093/nar/gkae278. URL <https://academic.oup.com/nar/article/52/W1/W264/7645776>.
- Dehghani, M., Djolonga, J., Mustafa, B., Padlewski, P., Heek, J., Gilmer, J., Steiner, A., Caron, M., Geirhos, R., Alabdulmohsin, I., Jenatton, R., Beyer, L., Tschannen, M., Arnab, A., Wang, X., Riquelme, C., Minderer, M., Puigcerver, J., Evci, U., Kumar, M., Steenkiste, S. v., Elsayed, G. F., Mahendran, A., Yu, F., Oliver, A., Huot, F., Bastings, J., Collier, M. P., Gritsenko, A., Birodkar, V., Vasconcelos, C., Tay, Y., Mensink, T., Kolesnikov, A., Pavetić, F., Tran, D., Kipf, T., Lučić, M., Zhai, X., Keysers, D., Harmsen, J., and Houlsby, N. Scaling Vision Transformers to 22 Billion Parameters, February 2023. URL <http://arxiv.org/abs/2302.05442>. arXiv:2302.05442 [cs].
- Edgar, R. C. Updating the 97% identity threshold for 16S ribosomal RNA OTUs. *Bioinformatics*, 34(14):2371–2375, July 2018. ISSN 1367-4803, 1367-4811. doi: 10.1093/bioinformatics/bty113. URL <https://academic.oup.com/bioinformatics/article/34/14/2371/4913809>.
- Fasogbon, I. V., Ondari, E. N., Tusubira, D., Rangasamy, L., Venkatesan, J., Musyoka, A. M., and Aja, P. M. Recent focus in non-SELEX-computational approach for de novo aptamer design: A mini review. *Analytical Biochemistry*, pp. 115756, 2024. Publisher: Elsevier.
- Grønning, A., Doktor, T. K., Larsen, S., Petersen, U., Holm, L. L., Bruun, G., Hansen, M. B., Hartung, A.-M., Baumbach, J., and Andresen, B. S. DeepCLIP: predicting the effect of mutations on protein–RNA binding with deep learning. *Nucleic Acids Research*, pp. gkaa530, June 2020. ISSN 0305-1048, 1362-4962. doi: 10.1093/nar/gkaa530. URL <https://academic.oup.com/nar/advance-article/doi/10.1093/nar/gkaa530/5859960>.
- Guo, P., Coban, O., Snead, N. M., Trebley, J., Hoepflich, S., Guo, S., and Shu, Y. Engineering RNA for Targeted siRNA Delivery and Medical Application. *Advanced Drug Delivery Reviews*, 62(6):650–666, April 2010. ISSN 0169409X. doi: 10.1016/j.addr.2010.03.008. URL <https://linkinghub.elsevier.com/retrieve/pii/S0169409X10000773>.

- Hayes, T., Rao, R., Akin, H., Sofroniew, N. J., Oktay, D., Lin, Z., Verkuil, R., Tran, V. Q., Deaton, J., Wiggert, M., Badkundri, R., Shafkat, I., Gong, J., Derry, A., Molina, R. S., Thomas, N., Khan, Y., Mishra, C., Kim, C., Bartie, L. J., Nemeth, M., Hsu, P. D., Sercu, T., Candido, S., and Rives, A. Simulating 500 million years of evolution with a language model, July 2024. URL <http://biorxiv.org/lookup/doi/10.1101/2024.07.01.600583>.
- Hendrycks, D. and Gimpel, K. Gaussian Error Linear Units (GELUs), 2016. URL <https://arxiv.org/abs/1606.08415>. Version Number: 5.
- Hentze, M. W., Castello, A., Schwarzl, T., and Preiss, T. A brave new world of RNA-binding proteins. *Nature Reviews Molecular Cell Biology*, 19(5):327–341, May 2018. ISSN 1471-0072, 1471-0080. doi: 10.1038/nrm.2017.130. URL <https://www.nature.com/articles/nrm.2017.130>.
- Im, J., Park, B., and Han, K. A generative model for constructing nucleic acid sequences binding to a protein. *BMC genomics*, 20(Suppl 13):967, 2019. Publisher: Springer.
- Ingraham, J. B., Baranov, M., Costello, Z., Barber, K. W., Wang, W., Ismail, A., Frappier, V., Lord, D. M., Ng-Thow-Hing, C., Van Vlack, E. R., Tie, S., Xue, V., Cowles, S. C., Leung, A., Rodrigues, J. V., Morales-Perez, C. L., Ayoub, A. M., Green, R., Puentes, K., Oplinger, F., Panwar, N. V., Obermeyer, F., Root, A. R., Beam, A. L., Poelwijk, F. J., and Grigoryan, G. Illuminating protein space with a programmable generative model. *Nature*, 623(7989):1070–1078, November 2023. ISSN 0028-0836, 1476-4687. doi: 10.1038/s41586-023-06728-8. URL <https://www.nature.com/articles/s41586-023-06728-8>.
- Iwano, N., Adachi, T., Aoki, K., Nakamura, Y., and Hamada, M. Generative aptamer discovery using RaptGen. *Nature Computational Science*, 2(6):378–386, June 2022. ISSN 2662-8457. doi: 10.1038/s43588-022-00249-6. URL <https://www.nature.com/articles/s43588-022-00249-6>.
- Jumper, J., Evans, R., Pritzel, A., Green, T., Figurnoy, M., Ronneberger, O., Tunyasuvunakool, K., Bates, R., Žídek, A., Potapenko, A., Bridgland, A., Meyer, C., Kohli, S. A. A., Ballard, A. J., Cowie, A., Romera-Paredes, B., Nikolov, S., Jain, R., Adler, J., Back, T., Petersen, S., Reiman, D., Clancy, E., Zielinski, M., Steinegger, M., Pacholska, M., Berghammer, T., Bodenstein, S., Silver, D., Vinyals, O., Senior, A. W., Kavukcuoglu, K., Kohli, P., and Hassabis, D. Highly accurate protein structure prediction with AlphaFold. *Nature*, 596(7873):583–589, August 2021. ISSN 0028-0836, 1476-4687. doi: 10.1038/s41586-021-03819-2. URL <https://www.nature.com/articles/s41586-021-03819-2>.
- Kim, N., Gan, H. H., and Schlick, T. A computational proposal for designing structured RNA pools for in vitro selection of RNAs. *RNA*, 13(4):478–492, 2007a. Publisher: Cold Spring Harbor Lab.
- Kim, N., Shin, J. S., Elmetwaly, S., Gan, H. H., and Schlick, T. RagPools: RNA-As-Graph-Pools: a web server for assisting the design of structured RNA pools for in vitro selection. *Bioinformatics*, 23(21):2959–2960, 2007b. Publisher: Oxford University Press.
- Kingma, D. P. and Ba, J. Adam: A Method for Stochastic Optimization, January 2017. URL <http://arxiv.org/abs/1412.6980>. arXiv:1412.6980 [cs].
- Lee, G., Jang, G. H., Kang, H. Y., and Song, G. Predicting aptamer sequences that interact with target proteins using an aptamer-protein interaction classifier and a Monte Carlo tree search approach. *PLoS one*, 16(6):e0253760, 2021. Publisher: Public Library of Science San Francisco, CA USA.
- Li, D., Huang, R., Cui, C., Towey, D., Zhou, L., Tian, J., and Zou, B. RNA-Protein Interaction Prediction Based on Deep Learning: A Comprehensive Survey. *arXiv preprint arXiv:2410.00077*, 2024.
- Li, W. and Godzik, A. Cd-hit: a fast program for clustering and comparing large sets of protein or nucleotide sequences. *Bioinformatics*, 22(13):1658–1659, July 2006. ISSN 1367-4811, 1367-4803. doi: 10.1093/bioinformatics/btl158. URL <https://academic.oup.com/bioinformatics/article/22/13/1658/194225>.
- Nori, D. and Jin, W. RNAFlow: RNA Structure & Sequence Design via Inverse Folding-Based Flow Matching, June 2024. URL <http://arxiv.org/abs/2405.18768>. arXiv:2405.18768 [q-bio].
- Obonyo, S., Jouandeau, N., and Owuor, D. RNA Generative Modeling With Tree Search. In *2024 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB)*, pp. 1–9. IEEE, 2024.
- Olechnovič, K. and Venclovas, C. Voronota: A fast and reliable tool for computing the vertices of the Voronoi diagram of atomic balls. *Journal of Computational Chemistry*, 35(8):672–681, March 2014. ISSN 0192-8651, 1096-987X. doi: 10.1002/jcc.23538. URL <https://onlinelibrary.wiley.com/doi/10.1002/jcc.23538>.

- Olechnovič, K. and Venclovas, C. VoroContacts: a tool for the analysis of interatomic contacts in macromolecular structures. *Bioinformatics*, 37(24):4873–4875, December 2021. ISSN 1367-4803, 1367-4811. doi: 10.1093/bioinformatics/btab448. URL <https://academic.oup.com/bioinformatics/article/37/24/4873/6300513>.
- Ozden, F., Barazandeh, S., Akboga, D., Tabrizi, S. S., Seker, U. O. S., and Cicek, A. E. RNAGEN: A generative adversarial network-based model to generate synthetic RNA sequences to target proteins, July 2023. URL <http://biorxiv.org/lookup/doi/10.1101/2023.07.11.548246>.
- Park, B. and Han, K. Discovering protein-binding RNA motifs with a generative model of RNA sequences. *Computational Biology and Chemistry*, 84:107171, February 2020. ISSN 14769271. doi: 10.1016/j.compbiolchem.2019.107171. URL <https://linkinghub.elsevier.com/retrieve/pii/S1476927119305365>.
- Ray, D., Kazan, H., Chan, E. T., Castillo, L. P., Chaudhry, S., Talukder, S., Blencowe, B. J., Morris, Q., and Hughes, T. R. Rapid and systematic analysis of the RNA recognition specificities of RNA-binding proteins. *Nature Biotechnology*, 27(7):667–670, July 2009. ISSN 1087-0156, 1546-1696. doi: 10.1038/nbt.1550. URL <https://www.nature.com/articles/nbt.1550>.
- Ray, D., Kazan, H., Cook, K. B., Weirauch, M. T., Najafabadi, H. S., Li, X., Gueroussov, S., Albu, M., Zheng, H., Yang, A., Na, H., Irimia, M., Matzat, L. H., Dale, R. K., Smith, S. A., Yarosh, C. A., Kelly, S. M., Nabet, B., Mecnas, D., Li, W., Laishram, R. S., Qiao, M., Lipshitz, H. D., Piano, F., Corbett, A. H., Carstens, R. P., Frey, B. J., Anderson, R. A., Lynch, K. W., Penalva, L. O. F., Lei, E. P., Fraser, A. G., Blencowe, B. J., Morris, Q. D., and Hughes, T. R. A compendium of RNA-binding motifs for decoding gene regulation. *Nature*, 499(7457):172–177, July 2013. ISSN 0028-0836, 1476-4687. doi: 10.1038/nature12311. URL <https://www.nature.com/articles/nature12311>.
- Reddi, S. J., Kale, S., and Kumar, S. On the Convergence of Adam and Beyond, April 2019. URL <http://arxiv.org/abs/1904.09237>. arXiv:1904.09237 [cs].
- Rhiju, D., Shujun, H., Alissa, H., and Rachael, K. Nucleic Acid Assessment CASP16, December 2024. URL https://predictioncenter.org/casp16/doc/presentations/Day-3/Day3-01-Kretsche_CASP16_NA_Assessment_PuntaCana_RCK_v1.pptx.
- Shin, I., Kang, K., Kim, J., Sel, S., Choi, J., Lee, J.-W., Kang, H. Y., and Song, G. AptaTrans: a deep neural network for predicting aptamer-protein interaction using pretrained encoders. *BMC bioinformatics*, 24(1):447, 2023. Publisher: Springer.
- Su, J., Lu, Y., Pan, S., Murtadha, A., Wen, B., and Liu, Y. RoFormer: Enhanced Transformer with Rotary Position Embedding, November 2023. URL <http://arxiv.org/abs/2104.09864>. arXiv:2104.09864 [cs].
- Thavarajah, W., Hertz, L. M., Bushhouse, D. Z., Archuleta, C. M., and Lucks, J. B. RNA Engineering for Public Health: Innovations in RNA-Based Diagnostics and Therapeutics. *Annual Review of Chemical and Biomolecular Engineering*, 12(1):263–286, June 2021. ISSN 1947-5438, 1947-5446. doi: 10.1146/annurev-chembioeng-101420-014055. URL <https://www.annualreviews.org/doi/10.1146/annurev-chembioeng-101420-014055>.
- The RNAcentral Consortium, Sweeney, B. A., Petrov, A. I., Burkov, B., Finn, R. D., Bateman, A., Szymanski, M., Karlowski, W. M., Gorodkin, J., Seemann, S. E., Cannon, J. J., Gutell, R. R., Fey, P., Basu, S., Kay, S., Cochrane, G., Billis, K., Emmert, D., Marygold, S. J., Huntley, R. P., Lovering, R. C., Frankish, A., Chan, P. P., Lowe, T. M., Bruford, E., Seal, R., Vandesompele, J., Volders, P.-J., Paraskevopoulou, M., Ma, L., Zhang, Z., Griffiths-Jones, S., Bujnicki, J. M., Boccaletto, P., Blake, J. A., Bult, C. J., Chen, R., Zhao, Y., Wood, V., Rutherford, K., Rivas, E., Cole, J., Lauderkind, S. J. F., Shimoyama, M., Gillespie, M. E., Orlic-Milacic, M., Kalvari, I., Nawrocki, E., Engel, S. R., Cherry, J. M., Team, S., Berardini, T. Z., Hatzigeorgiou, A., Karagkouni, D., Howe, K., Davis, P., Dinger, M., He, S., Yoshihama, M., Kenmochi, N., Stadler, P. F., and Williams, K. P. RNAcentral: a hub of information for non-coding RNA sequences. *Nucleic Acids Research*, 47(D1):D221–D229, January 2019. ISSN 0305-1048, 1362-4962. doi: 10.1093/nar/gky1034. URL <https://academic.oup.com/nar/article/47/D1/D221/5160993>.
- Torkamanian-Afshar, M., Nematzadeh, S., Tabar zad, M., Najafi, A., Lanjanian, H., and Masoudi-Nejad, A. In silico design of novel aptamers utilizing a hybrid method of machine learning and genetic algorithm. *Molecular diversity*, 25:1395–1407, 2021. Publisher: Springer.
- Tseng, C.-Y., Ashrafuzzaman, M., Mane, J. Y., Kaptj, J., Mercer, J. R., and Tuszynski, J. A. Entropic Fragment-Based Approach to Aptamer Design. *Chemical Biology & Drug Design*, 78(1):1–13, 2011. Publisher: Wiley Online Library.
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. Attention Is All You Need, August 2023. URL <http://arxiv.org/abs/1706.03762>. arXiv:1706.03762 [cs].

Wang, Y., Mistry, B. A., and Chou, T. Discrete stochastic models of SELEX: Aptamer capture probabilities and protocol optimization. *The Journal of Chemical Physics*, 156(24), 2022. Publisher: AIP Publishing.

Wang, Z., Liu, Z., Zhang, W., Li, Y., Feng, Y., Lv, S., Diao, H., Luo, Z., Yan, P., He, M., and others. AptaDiff: de novo design and optimization of aptamers based on diffusion models. *Briefings in Bioinformatics*, 25(6): bbae517, 2024. Publisher: Oxford University Press.

Zhang, B. and Sennrich, R. Root Mean Square Layer Normalization, October 2019. URL <http://arxiv.org/abs/1910.07467>. arXiv:1910.07467 [cs].

Zhang, Y., Jiang, Y., Kuster, D., Ye, Q., Huang, W., Fürbacher, S., Zhang, J., Tang, Z., Ibberson, D., Wild, K., and others. Single-step discovery of high-affinity RNA ligands by UltraSelex. 2023.

Zhang, Z., Chao, L., Jin, R., Zhang, Y., Zhou, G., Yang, Y., Yang, Y., Huang, K., Yang, Q., Xu, Z., and others. RNA-Genesis: Foundation Model for Enhanced RNA Sequence Generation and Structural Insights. *bioRxiv*, pp. 2024–12, 2024. Publisher: Cold Spring Harbor Laboratory.

Zhao, Y., Oono, K., Takizawa, H., and Kotera, M. GenerRNA: A generative pre-trained language model for de novo RNA design. *PLOS ONE*, 19(10):e0310814, October 2024. ISSN 1932-6203. doi: 10.1371/journal.pone.0310814. URL <https://dx.plos.org/10.1371/journal.pone.0310814>.

Appendix

A. RNA-BAnG architecture

RNA-BAnG architecture is schematically presented in Figure 5 of the main text and is composed of protein and nucleotide modules, detailed in Figure A.1 and Figure A.2. The latent dimension of the model is $c_s = 128$, all heads dimensions are set to $c_h = 64$. Feedforward blocks have a scaling factor $n = 2$. We set the number of protein and nucleotide modules to 10 each. While reducing this number led to poorer performance, increasing it did not yield any noticeable improvements. We adjusted the rest of the hyperparameters by selecting the smallest model size combination that ensured stable convergence. Resulting model contains 14,5 million parameters.

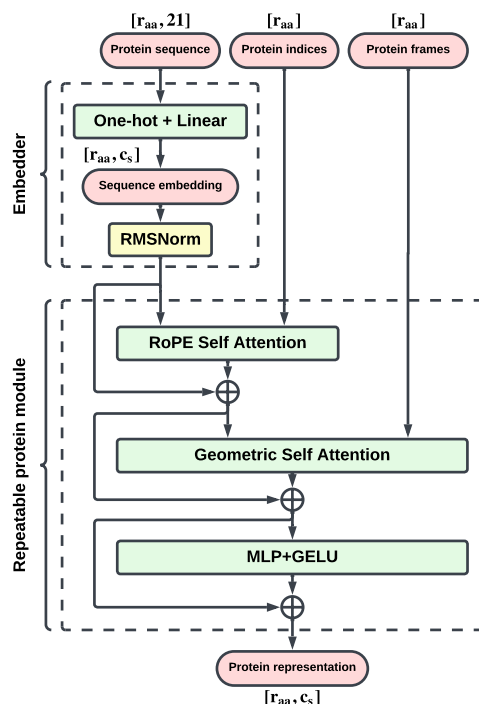


Figure A.1. Schematic illustration of the protein module in the RNA-BAnG architecture. Here, r_{aa} is the number of protein residues, \otimes stands for concatenation.

Algorithm A.1 describes the **Geometric Attention** block. It takes frames T_i and single representation s_i of every protein residue in the chain as inputs. The number of attention heads is $h = 12$, the number of query points is $N_{\text{query points}} = 4$, the number of value points is $N_{\text{value points}} = 8$. The weight per head $\gamma^h \in \mathbb{R}$ is the softplus of a learnable scalar. ω is a weighting factor. We adjusted the block’s hyperparameters to align with the AlphaFold2 IPA choices.

B. Synthetic task

B.1. Toy model

To avoid direct memorization on the synthetic task, we opted for a compact model, same for each tested method. Its final configuration is a two-block transformer (with a hidden dimension of 64) with the RoPE positional attention and two attention heads, resulting in 17k parameters. We trained the model for each method for 80k steps with a batch size of 8, using ADAM optimizer (Kingma & Ba, 2017) with the learning rate of 0.0001. During the inference, the length was fixed to 50 tokens for iterative methods.

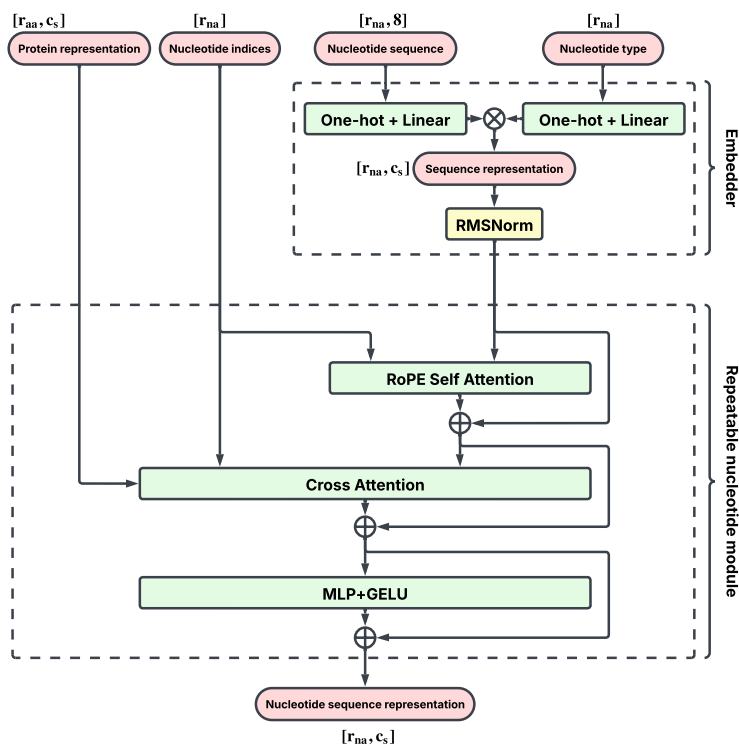


Figure A.2. Schematic illustration of the nucleotide module in the RNA-BAnG architecture. Here, r_{na} is the number of nucleotides, r_{aa} is the number of protein residues, \otimes stands for concatenation.

B.2. Synthetic sampling

During the inference, the sequence length was fixed to 50 tokens for iterative methods. For all the methods tested on the synthetic task, we sample tokens from their distributions using the top-k strategy with $k = 4$.

B.3. Generated synthetic sequences

Table B.1 lists examples of sequences generated for the DoubleBind task with each tested approach.

Table B.1. Example of generated sequences for the synthetic task. Expected synthetic motifs are underlined.

GENERATIVE METHOD	SEQUENCES
BANG	UCCCGGCUUGUCCGAUCGGAAC <u>UGACUC</u> GCACCUGGUUCCGACUUAUUAU UAUCGGUACGCUACAGGCGUCU <u>CAAUUG</u> AGAGCGGCUGGCAUGUAUUGCU
AUTOREGRESSIVE	GGAC <u>CAAUUG</u> UUGAU <u>UGACUC</u> GACUAAUUGCUCCACC CGUAACGAUUGC
IANG ENTROPY	CCU <u>CAAUUG</u> GGGAGCCGGCUGCCGUGCUGCGAGUGACAUUUGAACGUGAU CUCGCCCGGGGACCAUUGCACAAAU <u>UGACUC</u> UAGCGCAGAAUGGUAC
ITERATIVE ENTROPY	AUAGAAUUUACCCACCUGAUGAUGCCCCACUUAGCGGAUAUCUGCUUUCG AAAAUAUUCGUGGUUUUACUCCACCACUCCUAAACGCGAACUGAACCUAC

Algorithm A.1 Geometric Attention

input s_i, T_i
 $q_i^{hp}, k_i^{hp} \leftarrow \text{LinearNoBias}(s_i)$ $p \in [1, N_{\text{query points}}]$
 $v_i^{hp} \leftarrow \text{LinearNoBias}(s_i)$ $p \in [1, N_{\text{point values}}]$
 $w \leftarrow \sqrt{\frac{2}{9N_{\text{query points}}}}$
 $a_{ij}^h \leftarrow \text{softmax}\left(-\frac{\gamma^h w}{2} \sum_p \|T_i \circ q_i^{hp} - T_j \circ k_j^{hp}\|^2\right)$
 $o_i^{hp} \leftarrow T_i^{-1} \circ \sum_j a_{ij}^h (T_j \circ v_j^{hp})$
output $\text{Linear}\left(\text{concat}_{h,p}(o_i^{hp}, \|o_i^{hp}\|)\right)$

C. Data processing

C.1. Protein coupled nucleotide sequences

We defined two protein-RNA or protein-DNA chains as interacting if at least one interaction occurred between their residues. We defined an interaction between two residues if they share at least one atom-atom contact, as calculated using the VoroContacts software (Olechnovič & Venclovas, 2021). The VoroContacts method identifies contacts based on the Voronoi tessellation of atomic balls, constrained within the solvent-accessible surface (Olechnovič & Venclovas, 2014).

We excluded all samples where protein atom coordinates are ambiguous (had alternative locations), where the protein chain contains non-canonical residues (though we substitute 'SCE' with 'CYS' and 'MSE' with 'MET' beforehand), or where the nucleotide chain contains non-standard residues or a mixture of standard RNA and DNA residues. We also excluded samples with the nucleotide sequence length of less than 10 residues. To avoid potential computational resource problems, we included in the training only samples with a protein length of fewer than 500 residues.

During training, we split the data by protein sequence homology. We measured it using the clustering provided by PPI3D (Dapkūnas et al., 2024), where proteins within the same cluster share less than 40% sequence similarity with those in other clusters. We allocated samples from 95% of randomly selected clusters to the train set and the rest we used for validation. This approach allows tracking the model generalization across the protein space. To enhance our potential test set, we removed from the training data clusters containing proteins from the PDB samples 5ITH (chain A), 7CRE (chain A), 6QW6 (chain R), 8OPS (chain B), and 1CVJ (chain A).

Some proteins and nucleotide sequences are overrepresented in the data, which may lead to training imbalance. To address this, we introduced an additional level of sample clustering. First, we clustered nucleotide sequences (DNAs and RNAs separately) based on sequence similarity, grouping those with 90% identity using CD-HIT-EST (Li & Godzik, 2006). Then, we clustered the samples based on the combination of protein and nucleotide sequences clusters. At each epoch during training, we selected eight random samples (the mean cluster population) to represent each of the latter clusters, repeating the samples when necessary.

The resulting dataset consisted of 123,043 samples, distributed across 3,580 protein sequence clusters, 2,807 nucleotide sequence clusters (915 RNA and 1,892 DNA), and 12,667 combined clusters. The protein sequences in the dataset have a mean length of 155 residues with a standard deviation of 90. For nucleotide sequences, RNA lengths average 1,834 nucleotides ($\pm 1,564$), while DNA lengths average 76 nucleotides (± 78).

To prevent potential computational resource issues and to focus the model on the binding motifs, we truncated nucleotide sequences exceeding 300 residues during training and validation. This truncation limited sequences to 300 residues centered around the anchor point, which was selected as described prior to the truncation.

C.2. Standalone RNA sequences

From RNACentral database we selected sequences containing only standard residues and with lengths between 10 and 500 nucleotides. The selection was then deduplicated using CD-HIT-EST with a 90% similarity threshold, resulting in a final set of 3 million sequences.

Table F.1. Example of sequences generated by RNA-BAnG. Samples are ordered by decreasing proportion of high-affinity sequences. First two generated nucleotides are in bold and marked by red color.

RNACOMPETE ID	SEQUENCES
RNCMPT00077	AUUUUUAAAUAUU UU UAAAAAAAAA UAUU A UUUAUUUAAAA GUAUGUAUUUAU UU
RNCMPT00173	GUGA AC GUAAAACUUUUAAACUAAAAUCCUCA AUUGAAA G CUUUUAUGCCUUUACAAUAAA CGACUCAAAAGACAAUCUAAUACU CA AAAAACGGAUUAAACUAAAAAUA
RNCMPT00167	CUUGUCU GA CACG GCCCCUUGACCUUGAGUCCCAUGU GG CAGAGCAGUACAGGCUGAGUCGCU UG AG GUACACCA
RNCMPT00133	GCGCAGUGCCCAUAGACUCUGCAU AA UGGGACUCCAAGGAGCCGUCGGUU CCUGCGAACUUAUCAUUUCUAUAG UG AUGCAAUAUGUACUAAUUUUUA CGGAACGGAUUAUUUGUUUUAA AU AAUUAUGAAAAGUAUUUUUAUUAUA

D. Training details

As explained in the main text, we conducted the RNA-BAnG training in two steps. In the first step, we used a batch size of 64 and trained for 255k steps. In the second step, we used a batch size of 8 and trained for 216k steps. We stopped training when the validation loss decline became negligible. In both training steps, we used the ASMGraD (Reddi et al., 2019) variation of the ADAM optimizer (Kingma & Ba, 2017) with default parameters and the learning rate of 0.0001. Learning rate was warmed up linearly for the first 1k steps and then decayed exponentially with $\gamma = 0.99$ and a period of 1k steps. Complete training took 4 days on a single MI120 AMD GPU.

E. Evaluation details

E.1. Test data

The RNA Compendium study provides 244 samples, each comprising a protein sequence paired with approximately 200,000 RNA sequences and their corresponding experimental binding scores. These samples are designated by the authors as RNCMPT00XXX, where XXX represents a numerical identifier. For simplicity, throughout this paper, we refer to these samples solely by their numerical identifiers XXX.

We processed the data as follows. For each sample, RNA sequences were ranked by their binding scores. The top 2,000 sequences were labeled as the positive (interacting) class, while the bottom 2,000 were labeled as the negative (non-interacting) class. To eliminate sequence redundancy and prevent data leakage, duplicate RNA sequences were removed using CD-HIT-EST with a sequence identity threshold of 90%. These sequences were then randomly split into training and testing sets, so that each test set would have a 1,000 sequences. We then trained individual DeepCLIP models for each sample, following the protocol outlined by the authors. Only models achieving an area under the receiver operating characteristic curve (AUROC) of 0.95 or higher on the corresponding test set were selected for further analysis.

E.2. RNA-BAnG sampling

For RNA-BAnG, we sample tokens from their distributions using the top-k strategy with $k = 4$. Maximum sequence length is set to 50 nucleotides. Average sampling time is 20 minutes for 1,000 RNA sequences. To sample random sequences, we first randomly select a sequence length between 40 and 50, then uniformly sample that many nucleotides. This length choice ensures that the mean sequence length of random samples matches that of RNA-BAnG-generated sequences (45.6 nucleotides).

F. Additional results

Table F.1 lists randomly selected examples of generated RNA-BAnG sequences for the two best and two worst performance test samples.

Figure F.1 shows the proportion of high-affinity generated sequences as a function of AlphaFold pLDDT scores and protein sequence similarity with the train set. Appendix G details protein sequence similarity calculations. Figure F.2 shows kernel density plots calculated over the whole test set of 71 samples. We assessed four different DeepCLIP score threshold values (0.65, 0.75, 0.85, 0.95, from left to right in the Figure). We can conclude that the value of 0.95 is too stringent, as the mean of the positive experimental sets approaches the value of 0.5. Subjectively, the most visually appealing threshold value is 0.75. Nonetheless, in the main text (Figure 6) we continuously assess all the values by plotting the mean proportions above a certain threshold value. Distributions of DeepCLIP scores for each of 71 test samples are depicted in Figure F.3 and Figure F.4.

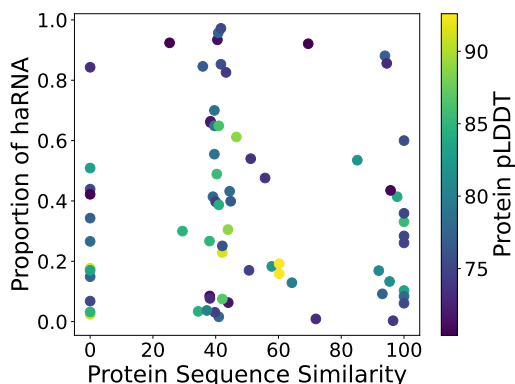


Figure F.1. Proportion of generated high-affinity sequences as a function of protein sequence similarity to the training data and the AlphaFold pLDDT scores of the predicted protein structures.

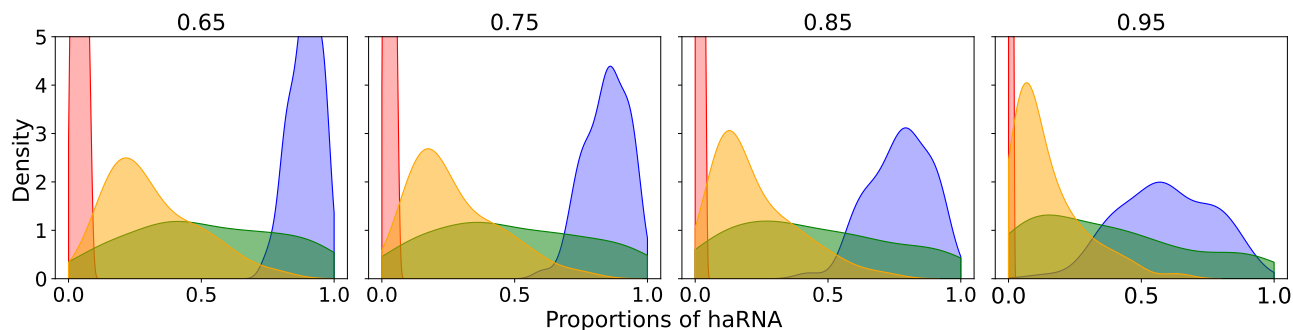


Figure F.2. Density plots of the proportion of high-affinity sequences for several threshold values (0.65, 0.75, 0.85, and 0.95, listed on top) across the whole test set. RNA-BAnG is shown in green, random sequences in yellow, positive experimental sequences in blue and negative experimental sequences in red.

G. Tools parameters

Clustering of nucleotide sequences was always performed using CD-HIT-EST (Li & Godzik, 2006) with a sequence identity threshold of 90%. We used CD-HIT-EST with these parameters: `-c 0.9 -n 9 -d 0 -T 10 -U 10 -l 9`. For protein sequence identity to the train set calculations we used blastp (Altschul et al., 1990) with default parameters. For the nucleotide sequences similarity to the train set calculations we used blastp (Altschul et al., 1990) with default parameters.

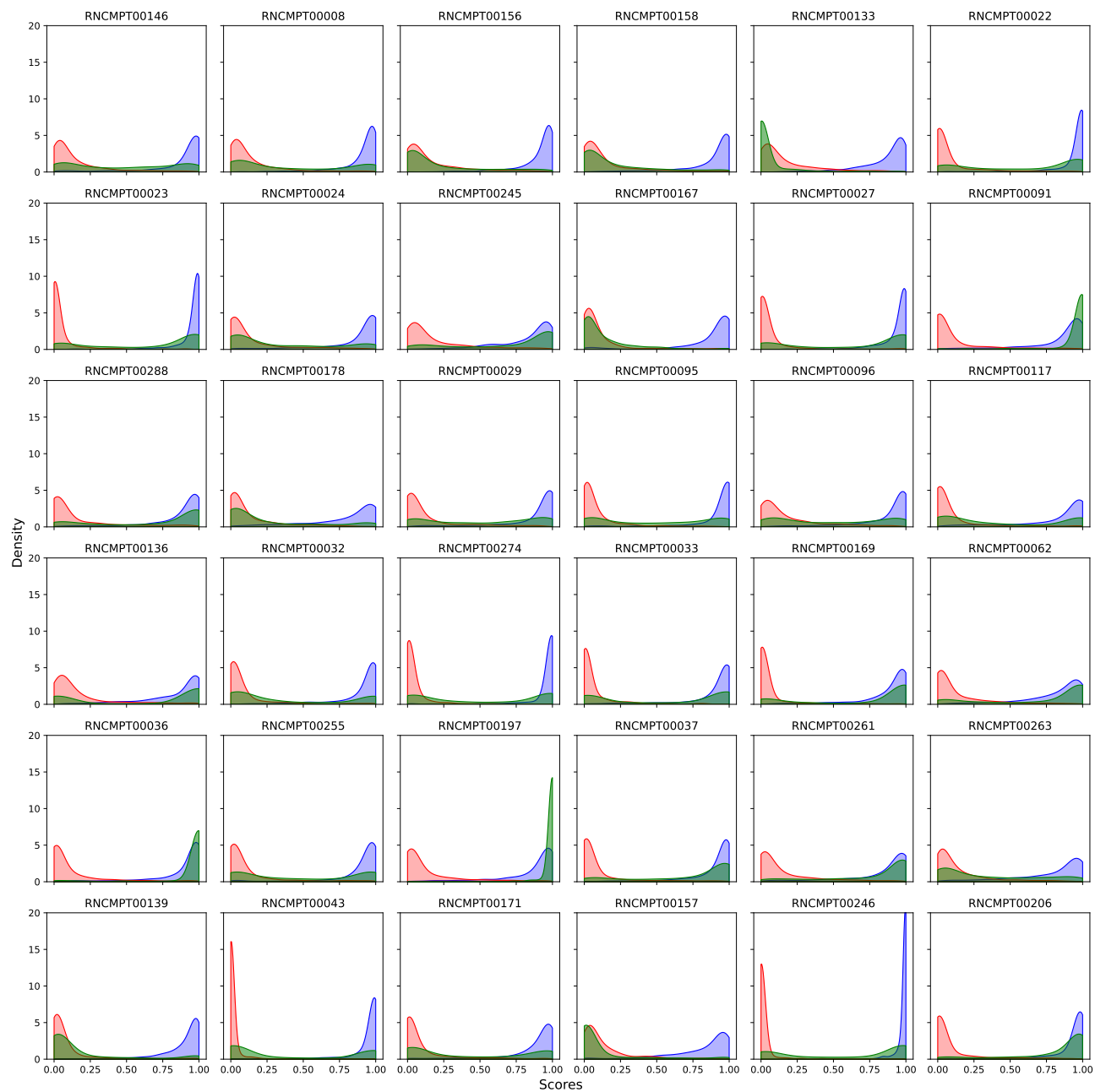


Figure F.3. Distribution of DeepCLIP scores for the first part of the test samples (RNAcompete sample IDs on top). Scores of RNA-BAnG are in green. Scores of positive and negative experimental sequences are in blue and red, respectively.

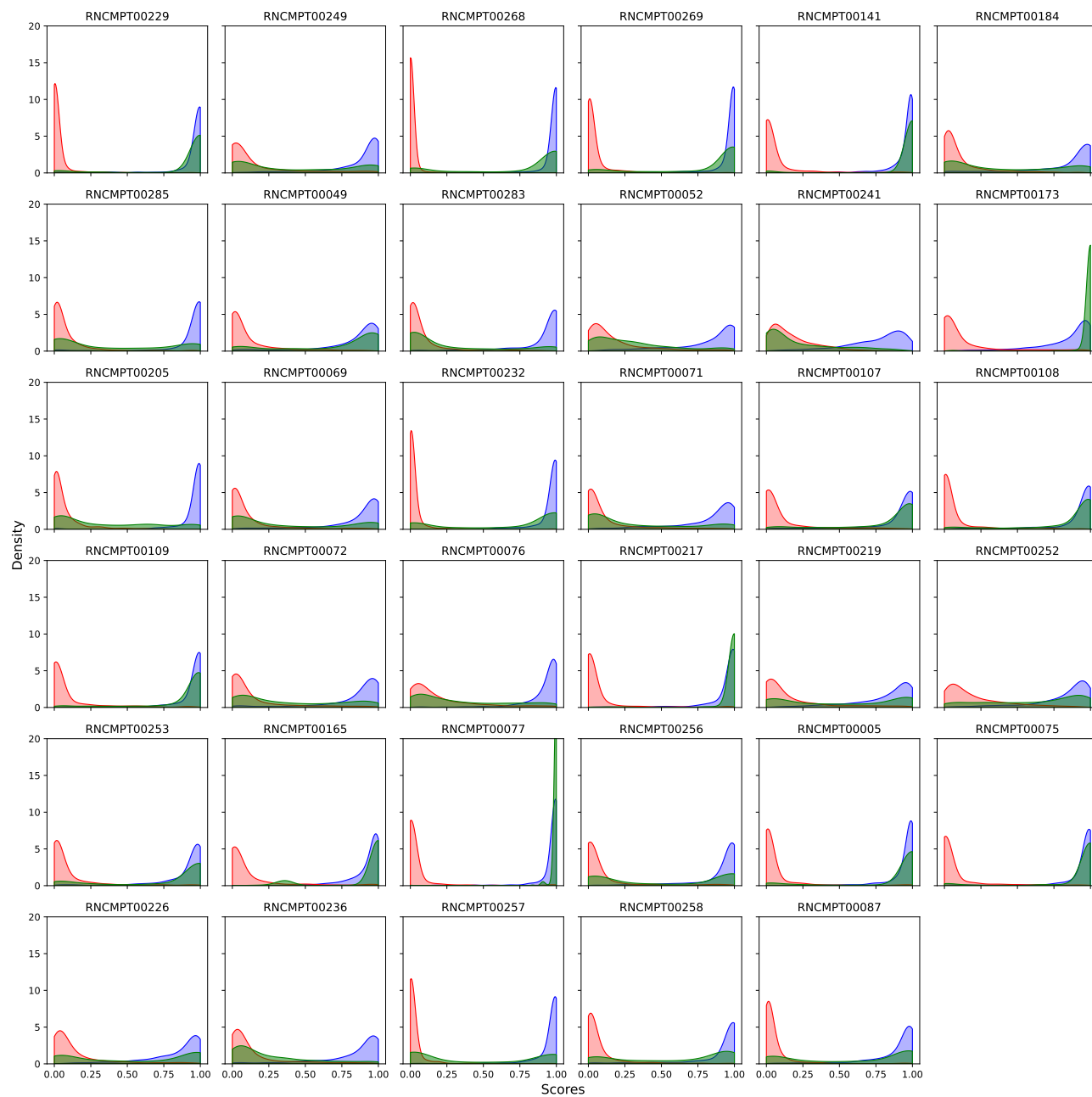


Figure F.4. Distribution of DeepCLIP scores for the second part of test samples (RNAcomplete sample IDs on top). Scores of RNA-BAnG are in green. Scores of positive and negative experimental sequences are in blue and red, respectively.