RNAformer: A Simple Yet Effective Deep Learning Model for RNA Secondary Structure Prediction

Jörg K.H. Franke, Frederic Runge, Ryan Köksal, Rolf Backofen, Frank Hutter
Department of Computer Science
University of Freiburg
Freiburg, Germany
frankej@cs.uni-freiburg.de

Abstract

Traditional RNA secondary structure prediction methods, based on dynamic programming, often fall short in accuracy. Recent advances in deep learning have aimed to address this, but may not adequately learn the biophysical model of RNA folding. Many deep learning approaches are also too complex, incorporating multi-model systems, ensemble strategies, or requiring external data like multiple sequence alignments. In this study, we demonstrate that a single deep learning model, relying solely on RNA sequence input, can effectively learn a biophysical model and outperform existing deep learning methods in standard benchmarks, as well as achieve comparable results to methods that utilize multi-sequence alignments. We dub this model *RNAformer* and achieve these benefits by a two-dimensional latent space, axial attention, and recycling in the latent space. Further, we found that our model performance improves when we scale it up. We also demonstrate how to refine a pre-trained RNAformer with fine-tuning techniques, which are particularly efficient when applied to a limited amount of high-quality data. A further aspect of our work is addressing the challenges in dataset curation in deep learning, especially regarding data homology. We tackle this through an advanced data processing pipeline that allows for training and evaluation of our model across various levels of sequence similarity. Our models and datasets are openly accessible, offering a simplified yet effective tool for RNA secondary structure prediction.

1 Introduction

The functions of RNAs are largely determined by their structures [Vicens and Kieft, 2022] and therefore, the longstanding RNA secondary structure prediction problem is of paramount importance in computational biology [Bonnet et al., 2020]. Generally, RNA secondary structure prediction methods can be roughly categorized into *de novo* and *homology modeling* approaches. *De novo* methods predict the structure solely from the primary sequence, while homology modeling utilizes evolutionary information from a set of homologous sequences, typically as multiple sequence alignments (MSAs). While MSA-based methods often achieve higher accuracy, they face several practical limitations. The prediction of RNA structures for known families is itself of less practical relevance and can be accurately determined via comparative analysis [Szikszai et al., 2022]. Furthermore, obtaining MSAs is time-consuming and computationally intensive, with often only a few available homologs for a given RNA, which further limits the practical use of these approaches [Singh et al., 2021a]. Due to these limitations, *de novo* prediction methods are typically preferred [Szikszai et al., 2022].

Traditional methods for *de novo* prediction structure prediction employ dynamic programming (DP) based on thermodynamic nearest neighbor parameters derived from optical melting experi-

Preprint. Under review.

ments [Mathews et al., 1999] to find the lowest energy structure, typically tending to maximize the number of Watson-Crick base pair interactions [Vicens and Kieft, 2022]. More recently, deep-learning-based approaches have conquered the field with superior performance on benchmark datasets [Zhao et al., 2021]. Additionally, these methods are capable of predicting an arbitrary adjacency matrix, allowing them to consider any type of base interaction [Singh et al., 2019a], including non-canonical pairs, pseudoknots [Staple and Butcher, 2005], and base multiplets, rather than being limited to nested structures [Hofacker et al., 1994]. However, when using additional evolutionary information from a set of homologous sequences (MSA), the predictive performance could still be improved over *de novo* approaches [Singh et al., 2021a]. Other approaches to further improve the performance of *de novo* structure prediction leverage additional information from large-scale pre-trained models on RNA sequence data [Chen et al., 2022], the combination of multiple models into an ensemble [Singh et al., 2019a], or consist of complex systems that incorporate multiple algorithms Sato et al. [2021], Singh et al. [2021a]. This increased complexity negatively affects the runtime and usability, and limits the possibilities to adapt the model to new data. A lean and simple but efficient deep learning model could boost research in the field of RNA secondary structure prediction and serve as a starting point for future work.

However, especially in RNA secondary structure prediction, recent work has questioned whether the reported high accuracy of these methods reflects their ability to learn the underlying biophysical model of the folding process [Flamm et al., 2021], or are a result of similarities between the training and test datasets. One reason for this is that most RNA secondary structure training/test sets are biased toward certain large families, such as tRNAs. This leads to the need for improved data processing pipelines that better evaluate and reflect the model's ability to learn the underlying biophysical dynamics of the folding process. It is notably important to prevent homologies between the training and test datasets [Justyna et al., 2023], where homologies can be sequence-based or structure-sequence-based. Therefore, recently proposed methods applied different data processing pipelines to avoid such homologies, including filtering sequence similarities, BLAST search Altschul et al. [1997], or using covariance models to remove sequences based on sequence and structure similarity [Singh et al., 2021a]. For this reason, the specific training/test set combination poses a significant bias to the prediction quality, leading to a set of incomparable methods that all claim state-of-the-art results on different training/test set characteristics.

In this work, we tackle these challenges by introducing a lean deep learning architecture tailored to RNA secondary structure prediction. Our approach, the *RNAformer*, is capable of capturing long-range interactions while working on a 2D structure representation to leverage the advantages of deep learning methods. Our method does not require additional information like MSAs, embeddings, nor artificial improvements e.g. via ensembling techniques, but still outperforms previous *de novo* prediction methods in common RNA secondary structure prediction benchmarks while being on par with the current state-of-the-art homology modeling methods. We achieve this primarily through modeling the two-dimensional pairing matrix in the latent space, axial attention [Ho et al., 2019], latent space recycling [Jumper et al., 2021], and by applying fine-tuning methods on high-quality RNA samples. Using our advanced data processing pipelines, we can compare our model to existing state-of-the-art deep learning approaches on both intra- and inter-family secondary structure prediction tasks, addressing the problem of a lack of standardized benchmarks among approaches. Overall, our contributions can be summarized as follows:

- We propose a deep learning architecture termed *RNAformer* for RNA secondary structure prediction based on axial attention and latent space recycling (Section 3).
- We discuss the difficulties of dataset curation in RNA secondary structure prediction and train our methods at different levels of similarity (Section 4).
- We show that our method is capable of learning the underlying folding dynamics of an MFE model in an inter-family prediction setting.
- We achieve state-of-the-art performance on the commonly used benchmark datasets TS0, TS1-3, and TS hard, outperforming *de novo* prediction methods while being on par with the current best-performing homology modeling methods (Section 5).
- We provide an open source implementation of our model and publish all datasets[1].

[1]Code, Models, and Datasets: github.com/automl/RNAformer

2 Related Work

2.1 Non Deep Learning Base Methods

Early methods of computational biology sought to predict RNA structures using a dynamic programming (DP) approach based on thermodynamic nearest neighbor parameters to predict the single most likely secondary structure as the one that results in the minimum amount of free energy. In terms of traditional algorithms, *Vienna RNAfold* Hofacker et al. [1994] marked a breakthrough in computational methods for RNA secondary structure prediction. It uses dynamic programming to make predictions, based on optimizing for the Minimum Free Energy (MFE) model, implementing the partition function to compute base pair probabilities. This was based on the rationale that an RNA structure must be thermodynamically stable to perform its function effectively. It achieved relatively high accuracy and efficiency and remains the most widely used and cited approach. Huang et al. [2019] proposes *LinearFold*, a secondary structure prediction method based on 5' to 3' DP and beam search. It runs in linear ($\mathcal{O}(n)$) time, improving on existing DP-based approaches with cubic ($\mathcal{O}\left(n^3\right)$) runtime. It does this by scanning an RNA sequence in a left-to-right direction, rather than the traditional bottom-up fashion. This allows the use of the effective beam pruning heuristic, making it the first approach to achieve linear runtime and space.

IPknot is a computational method based on maximizing the expected accuracy (MEA) of a predicted structure with pseudoknots by decomposing a pseudoknot structure into a set of pseudoknot-free sub-structures and approximates a base-pairing probability distance that considers pseudoknots [Sato et al., 2011]. It uses a heuristic method for refining base-pair probabilities to improve its prediction accuracy, as well as, integer programming with threshold cutoff to improve on the MEA approach. The updated version in 2022 employs the *LinearPartition* Zhang et al. [2020] model to automatically select the optimal threshold parameters based on the pseudo-expected accuracy. This aimed to solve previous scalability issues for longer sequences, allowing for linear time computation. *CONTRAfold* Do et al. [2006] is based on conditional log-linear models (CLLMs), a flexible class of probabilistic models that generalize upon Stochastic Context-Free Grammars (SCFGs) using discriminative training and feature-rich scoring. It aimed to close the gap between probabilistic and thermodynamic models. It achieved the highest single-sequence prediction accuracy at the time of publication, providing an effective alternative to the empirical measurement of thermodynamic parameters for RNA secondary structure prediction. *EternaFold* Wayment-Steele et al. [2022] is a multitask model trained on the varied data types in the EternaBench dataset, consisting of more than 20,000 synthetic RNA constructs designed on the RNA design platform Eterna Lee et al. [2014]. However, EternaFold was recently outperformed by RNA-FM and we excluded EternaFold from our evaluations.

PKiss Theis et al. [2010] introduces three heuristic strategies for folding RNA sequences into structures inclusive of kissing hairpin motifs. It is based primarily on the construction of kissing hairpin motifs from the overlay of two simple canonical pseudoknots to overcome the challenge that the overlay does not adhere to Bellman's Principle of Optimality. The three strategies employed consist of varying levels of computational complexity, with the simplest being found to yield the best performance. However, while some of these methods can predict structures with pseudoknots or noncanonical base pairs, none of them can predict both.

2.2 Deep Learning Base Methods

Singh et al. [2019b] introduces with *SPOT-RNA* the first algorithm which employs deep neural networks for end-to-end predictions of RNA secondary structures. It uses an ensemble of models with residual networks bidirectional LSTM Schuster and Paliwal [1997], and dilated convolution Yu and Koltun [2016] architectures. *SPOT-RNA* was trained on a large set of intra-family RNA data for *de novo* predictions and was further fine-tuned on a small set of experimentally derived RNA structures for predictions, including tertiary interactions. Its successor, *SPOT-RNA2* is a homology modeling method that incorporates evolutionary information through multi-sequence alignments (MSAs), sequence profiles, and features derived from direct coupling analysis for the prediction of RNA secondary structures Singh et al. [2021b]. It is based on an ensemble of models but solely uses dilated convolutions. Predictions are intra-family wise, independent of the curation of the dataset since homologies between the evolutionary information and the training or test sets were not explicitly

excluded during evaluations. *SPOT-RNA2* achieved state-of-the-art accuracy on their curated test sets based on data from the Protein Data Bank (PDB) wwp [2019].

UFold [Fu et al., 2022] employs a UNet [Ronneberger et al., 2015] architecture for *de novo* secondary structure prediction, additionally reporting results for predictions on data that contained tertiary interactions after fine-tuning the model. *UFold* treats an RNA sequence as an image of all possible base-pairing maps with an additional map for pair probabilities, represented as square matrices.

MXFold2 [Sato et al., 2021] seeks to learn the scoring function for a subsequent DP algorithm using a CNN/BiLSTM architecture. The network is trained to predict scores close to a set of thermodynamic parameters. In contrast to other described methods, *MXFold2* is restricted to predicting a limited set of base pairs due to its DP algorithm.

ProbTransformer Franke et al. [2022] is based on a Transformer architecture with the addition of a hierarchical latent distribution as a probabilistic enhancement for either an encoder- or decoder-based Transformer. It is the first model capable of sampling different structures of this latent distribution, shown by reconstructing structure ensembles of a distinct dataset with multiple structures for a given input sequence. This aimed to accommodate the ambiguities and stochastic nature of secondary structure data and in the folding process itself.

E2Efold [Chen et al., 2020] uses a Transformer encoder architecture for *de novo* prediction of RNA secondary structures. The algorithm was trained on a dataset of homologous RNAs and showed significantly reduced accuracy across several other works [Sato et al., 2021, Fu et al., 2022], which indicates strong overfitting. We use the same data as the respective work for evaluations and thus exclude *E2Efold* from our evaluations.

RNA-FM [Chen et al., 2022] uses sequence embeddings of an RNA foundation model that is trained on 23 million RNA sequences from 800000 species to perform intra-family predictions of RNA secondary structures in a downstream task. The foundation model consists of a 12-layer transformer architecture, while the downstream models use a ResNet32 architecture.

In contrast to these related works, our approach models the pairing matrix in the latent space using axial attention. This makes the model independent of the sequence length in contrast to all CNN-based approaches since their receptive field depends on the depth/sequence length ratio, while still allowing to predict pseudoknots and multiplets.

3 RNAformer

The *RNAformer* is inspired by the protein folding algorithm, AlphaFold [Jumper et al., 2021], which models a multi-sequence alignment (MSA) and a pair matrix in the latent space and processes it with the use of axial attention [Ho et al., 2019]. In contrast to AlphaFold and similar to Lin et al. [2023] in protein folding, we dispense the use of an MSA due to its well-known limitations [Singh et al., 2021a]. We further simplify this architecture to only use axial attention for modeling a latent representation for the pairing between all nucleotides of the input RNA sequence without the need for an additional structure module.

In the following we describe the RNAformer in detail; please find an overview of our architecture in Figure 1. We input a RNA sequence $X \in \{A, C, G, U, N\}^N$ of length N and embed it twice, one row- and one column-wise embedding to generate a 2D latent representation. The embeddings can be represented as:

$$E_{\text{row}} = \text{Embed}_{\text{row}}(X), \quad E_{\text{col}} = \text{Embed}_{\text{col}}(X), \tag{1}$$

where $E_{\text{row}} \in \mathbb{R}^{N \times d}$ and $E_{\text{col}} \in \mathbb{R}^{N \times d}$ are the row-wise and column-wise embeddings respectively, with d being the embedding dimension. The broadcasting and combination of these two matrices to form a 2D latent space can be represented as:

$$L^{(0)} = E_{\text{row}} \oplus E_{\text{col}}^T, \tag{2}$$

where $L^{(0)} \in \mathbb{R}^{N \times N \times 2d}$ is the 2D latent space, and $\oplus$ denotes the broadcasting and addition operation; i.e., $L^{(0)}[i, j] = E_{\text{row}}[i] + E_{\text{col}}^T[j]$. For the positional encoding, we make use of the rotary position embedding Su et al. [2024]. The resulting latent representation will be further processed by a stack of M RNAformer blocks

$$L^{(i)} = \text{RNAformerBlock}(L^{(i-1)}), \quad \text{for } i = 1, 2, \ldots, M. \tag{3}$$

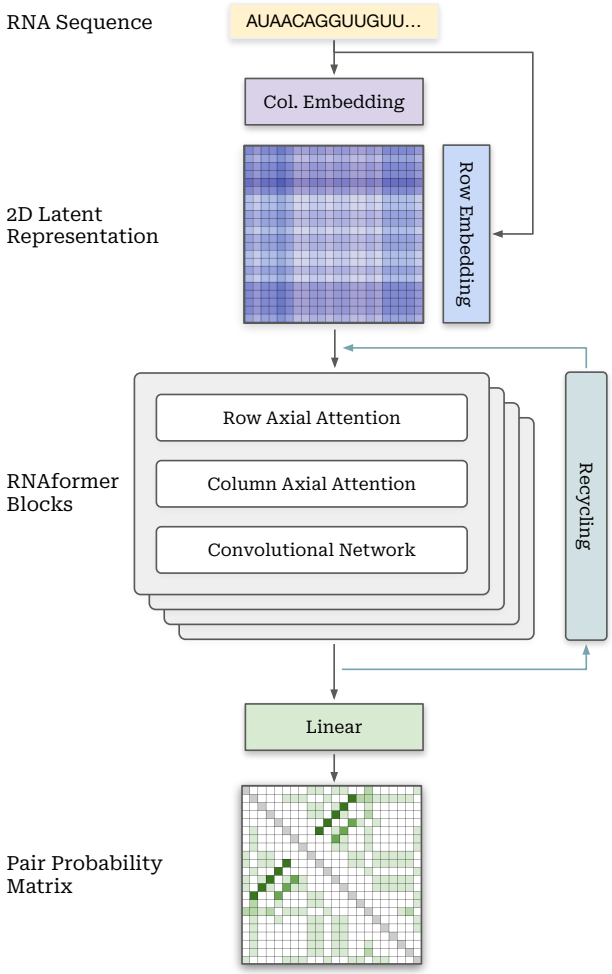

Figure 1: An overview of the *RNAformer* architecture.

Each block consists of row-wise and column-wise axial attention, followed by a 'transition' convolutional layer which serves a similar role as the point-wise feed-forward layer in the vanilla transformer architecture. We found the convolutional structure to perform better in our architecture, which we attribute to the fact that, while the axial attention layers can capture long-range information across the entire input structure, the convolutional network can help to model local structures like stem-loops.

We apply residual connections, pre-layer norm, and dropout to all three layers; the RNAformer block can then be represented as:

$$\begin{aligned} L^{(i)\prime} &= L^{(i)} + \text{AxialAttentionLayer}_{\text{row}}(L^{(i)}) \\ L^{(i)\prime\prime} &= L^{(i)\prime} + \text{AxialAttentionLayer}_{\text{col}}(L^{(i)\prime}) \\ L^{(i+1)} &= L^{(i)\prime\prime} + \text{TransitionConvLayer}(L^{(i)\prime\prime}). \end{aligned} \tag{4}$$

Here, the *TransitionConvLayer* consists of two convolutional layers with a SiLU activation function [Elfwing et al., 2018] in the middle, and an *AxialAttentionLayer* consists of a linear layer to create the query, key, and value for the *AxialAttention* and a linear layer to project its output. An axial attention, introduced by Ho et al. [2019], applies attention mechanisms over each axis independently, enabling efficient processing of higher-dimensional data. In more detail, the axial attention mechanism can be mathematically represented with indices for rows i and columns j for each 2-dimensional input to the attention mechanism Vaswani et al. [2017]: query $Q \in \mathbb{R}^{N \times N \times d}$, key $K \in \mathbb{R}^{N \times N \times d}$, and value $V \in \mathbb{R}^{N \times N \times d}$ for a sequence length of N and a latent dimension of d. We compute for

each column $j = 1, \cdots, N$

$$\text{AxialAttention}_{\text{row}}(Q, K, V, j) = \text{softmax}\left(\frac{Q_{:,j,:}K_{:,j,:}^T}{\sqrt{d}}\right)V_{:,j,:}$$

and for each row $i = 1, \cdots, N$

$$\text{AxialAttention}_{\text{col}}(Q, K, V, i) = \text{softmax}\left(\frac{Q_{i,:,:}K_{i,:,:}^T}{\sqrt{d}}\right)V_{i,:,:}.$$

Our model achieves a complete receptive field by applying attention consecutively along each axis, in contrast to convolutional networks (CNN) that expand this field over multiple layers. So in a CNN the number of layers to achieve a full receptive field depends on the input length. This could be harmful for a high variance in the input sequence length. Our approach may be better suited for secondary structure prediction since each layer accesses the whole sequence and can iteratively refine the structure prediction. To generate a prediction, we apply a linear layer after the RNAformer blocks and output the binary pairing probability matrix $P \in \mathbb{R}^{N \times N}$ of the secondary structure directly:

$$\text{P} = \text{sigmoid}(\text{Linear}(L^{(M)}))$$

We note that directly outputting the secondary structure's binary pairing probability matrix $P \in \mathbb{R}^{N \times N}$ is advantageous in many ways: if we were to instead output dot-bracket notation, like the ProbTransformer [Franke et al., 2022], this would make it unpractical to predict multiples, hard to predict pseudoknots and also require postprocessing to create a pair matrix.

To artificially increase the model depth, we apply recycling in the latent space, similar to AlphaFold, allowing the model to reprocess and correct predictions internally. Therefore, we apply the RNAformer blocks multiple times by normalizing and adding the block output to the embedded input and then infer the RNAformer blocks again. During training, gradients are only computed for the last recycling iteration. We additionally found that we can achieve a similar increase in performance by recycling solely during the fine-tuning stage, which is more efficient than applying recycling in the pre-training stage.

4 Data Pipelines

In recent years, several deep learning models have been proposed for RNA secondary structure prediction, with each of them claiming state-of-the-art performance on various datasets. However, we find that these reported results are often misleading due to the different data processing pipelines used to derive the training data. One way to ensure a fair comparison would be to retrain the models on the same training datasets; this is computationally infeasible and often even impossible due to the undisclosed training pipelines used. An alternative strategy is to define different levels of sequence similarity between the training set and the test and validation sets to then compare only against methods that use the same setting. We used this latter strategy and observed three different levels of similarity used in recent publications to define the training datasets. Next, we explain the initial data pool, the validation and test sets, as well as the data processing pipelines in further detail. All data processing pipelines used were provided by RnaBench [Runge et al., 2024].

4.1 Training Data Pool

We use the same training data pool for all experiments and apply different similarity settings to derive training subsets that allow the comparison against different methods trained on data of the same similarity setting. For the initial data pool, we collect data from the following public sources: the bpRNA-1m meta-database [Danaee et al., 2018], the ArchiveII [Sloma and Mathews, 2016] and RNAStrAlign [Tan et al., 2017] dataset provided by [Chen et al., 2020], all data from RNA-Strand [Andronescu et al., 2008], as well as all RNA-containing data from the Protein Data Bank (PDB) [wwp, 2019], downloaded in September 2023. Secondary structures for PDB samples were derived from the 3D structure information using DSSR [Lu et al., 2015]. For annotation of pseudoknots, we use bpRNA [Danaee et al., 2018] while ignoring base multiplets.

We use the test sets provided by Singh et al. [2019a] and Singh et al. [2021a]: TS0 derived from the bpRNA-1m meta-database, TS1 derived from high-resolution structures in the PDB, TS2 derived

from NMR structures, TS3 also derived from PDB, and TS-hard, a subset of TS1 and TS3. For validation, we use the two sets VL0 and VL1 [Singh et al., 2019a, 2021a]. Supplementary Material B provides an overview of our datasets.

4.2 Reducing Data Homology

To obtain datasets at different levels of similarity to the test data, we apply either sequence similarity, sequence similarity with a subsequent BLAST search, or all three similarity pipelines to both the samples from the training pool and the validation data. This allows us to compare between methods in the same settings originally used.

Sequence Similarity. UFold [Fu et al., 2022], SPOT-RNA [Singh et al., 2019a], MXFold2 [Sato et al., 2021], RNA-FM [Chen et al., 2022], and the ProbTransformer [Franke et al., 2022] report results on the testset TS0, using a similarity pipeline that considers sequence similarity between the training and test data. To achieve this, we remove sequence similarity between the training, validation and test sets, we follow the literature and apply CD-Hit [Fu et al., 2012] with a similarity cutoff of 80% between all sets.

BLAST Search. In addition to removing similar sequences via CD-Hit, Singh et al. [2019a] (SPOT-RNA) applied a BLAST-search [Altschul et al., 1997] at a high e-value of 10 to further remove training and validation samples that are hit by BLAST for any of the test samples. For a fair comparison with SPOT-RNA on TS1 and TS2, we apply the same two pipelines.

Covariance Models. While both the previously described pipelines only consider similarity on a sequence level, Singh et al. [2021a] recently proposed the use of covariance models to split the data in a family-based manner, including structure information. Inspired by this approach, we use BLASTN [Altschul et al., 1997] to search for homologs for each sample of the test set TS-hard using NCBI's nt database as reference. We create sequence- and structure-aware alignments using LocARNA-P [Will et al., 2012]. Note again that at this point, our pipeline differs from the approach of Singh et al. [2021a] that uses SPOT-RNA for the prediction of the consensus structure of the alignment which appears sub-optimal. For each of the resulting alignments, we build a covariance model using Infernal [Nawrocki and Eddy, 2013] and remove training and validation samples with a hit to the covariance model at an e-value of 0.1.

5 Experiments

We evaluate the RNAformer in three settings: (1) The learning of a simplified biophysical model as proposed by [Flamm et al., 2021], (2) intra-family predictions including pseudoknots and non-canonical base pairs, and (3) predictions on experimental data from the PDB, by training an RNAformer model on low-quality data and finetuning on high-quality PDB data. We test our finetuning in the intra- and inter-family prediction setting.

In line with the current literature, we report the F1 score for all results. Following [Mathews, 2019], we also report the F1-shifted, considering a predicted pair to be correct even if it is displaced by one position on one side to account for the dynamic nature of RNA.

Training Details. During the training of each experiment, we minimize the mean binary cross-entropy loss between the prediction P and the true adjacency matrices of the secondary structure. Since the adjacency matrices are heavily unbalanced, we mask 60% (during fine-tuning 80%) of the unpaired entries in the adjacency matrix before calculating the cross-entropy loss. We find that this masking helps to stabilize the training while not harming the training progress significantly. We further use a cosine learning rate schedule with warm-up and AdamW Loshchilov and Hutter [2019] for all experiments. We train RNAformer models with 6 blocks and latent dimensions of 64, 128, and 256, resulting in a total model size of about $2M$, $8M$, and $32M$ parameters. Depending on the model size, we train the RNAformer on 1 A10 or 8 A100 GPUs. Due to the two-dimensional latent space and a maximum sequence length of 500, we fit only one sample in the large RNAformer configuration on one A100 (40GB) though we use FlashAttention for a memory-efficient implementation of the axial attention Dao et al. [2022]. Therefore, we make use of gradient accumulation to achieve larger batch sizes. For a list of hyperparameters of the different experiments, we refer the reader to Supplementary Material A.

Table 1: We train different sizes of our model on the Rfam dataset on three different random seeds and report the mean performance.

Model	Rfam TS	
	F1 Score	Solved
RNAformer $32M+$	**0.967**	**83.5**%
RNAformer $32M$	0.948	68.1%
RNAformer $8M$	0.919	49.7%
RNAformer $2M$	0.846	22.9%
RNAfold Lorenz et al. [2011]	1.0	100%

5.1 Learning a Biophysical model

Setup. Flamm et al. [2021] recently proposed to evaluate deep learning-based models regarding their capabilities of learning a simplified biophysical model derived from predictions of a model based on thermodynamic parameters. In this experiment, we assess whether the *RNAformer* can replicate such a biophysical model in an inter-family prediction setting before we apply it to more advanced settings and real-world data. In particular, we evaluate to which degree the *RNAformer* can learn to mimic the predictions of *RNAfold* [Lorenz et al., 2011]. To test the scalability of our model, we trained multiple RNAformer models with $2M$, $8M$, and $32M$ parameters. We train the largest model with and without recycling to see the benefit of it.

Data. We derive a training dataset from families of the Rfam database version 14.9 [Kalvari et al., 2020] by selecting all families with a covariance model with maximum *CLEN* of ≤ 500 and sample a large set of sequences for each family from the covariance models using Infernal. We then build an initial dataset with two-thirds of the sequences from families with *CLEN* ≤ 200 and one-third of sequences from the families with *CLEN* > 200 to increase the family diversity. We randomly select 25 and 30 families from this set for validation and testing, respectively, and leave all samples from other families for training. All sequences are folded using *RNAfold*. We apply a length cutoff at 200 nucleotides since we expect *RNAfold* predictions to be more reliable for sequences below this threshold, since it reduces computational costs, and because all datasets of experimentally derived RNA structures from the literature consistently report a maximum sequence length below 200 nucleotides. We split these datasets into training, validation, and test families detailed in Supplementary Material B.

Results. We trained the RNAformer three times with random initialization on the dataset described previously. As shown in Table 1, we increasingly approach RNAfold's results with increasing model size. Our largest models achieve a mean F1 score on the test set of $94.8(\pm 0.026)$ without recycling and 96.7 ± 0.017 with recycling, respectively. The best model predicts 84% of the structures correctly (see Table 1). This result indicates that the *RNAformer* can learn the underlying biophysical model of the RNAfold modeling the folding process. A more comprehensive evaluation, also for all following experiments and including the performance on different base-pairs, is shown in Supplementary Material C.

5.2 Learning Structure Predictions for bpRNA

Setup. While the learning of a biophysical model is limited to canonical base pair interactions of nested RNA structures from predictions of *RNAfold*, our second experiment involves non-canonical interactions and pseudoknotted structures. We compare our model against others that report results using the same level of similarity: RNA-FM [Chen et al., 2022], SPOT-RNA [Singh et al., 2019a], MXFold2 [Sato et al., 2021], UFold [Fu et al., 2022], and ProbTransformer [Franke et al., 2022]. To also assess the influence of recycling in this case, we evaluate our model twice, once with recycling and once without.

Data. To be comparable to previous work, we train the *RNAformer* in an intra-family setting using a training set where we only applied the sequence similarity pipeline described in Section 4. The test

Table 2: The mean performance of three runs with different random seeds of the RNAformer with and without recycling () in comparison to the best competitors on the TS0 benchmark dataset. We evaluated all competitors based on their open-sourced models.

Model	TS0	
	F1 Score	F1-Shifted
RNAformer $32M$	**0.725**	**0.775**
RNAformer $32M$	**0.714**	**0.751**
RNAformer $8M$	**0.702**	**0.744**
RNAformer $2M$	**0.669**	0.713
RNA-FM Chen et al. [2022]	0.667	0.713
UFold Fu et al. [2022]	0.630	0.687
ProbTransformer Franke et al. [2022]	0.625	0.674
SPOT-RNA Singh et al. [2019a]	0.586	0.624
MXFold2 Sato et al. [2021]	0.550	0.596

set TS0 lacks base interactions with more than one pairing partner, so we exclude all samples that contain base multiplets from the training data.

Results. The comparison of intra-family predictions on the TS0 dataset is shown in Table 2. We observe that our model clearly outperforms all other methods, achieving an F1 score of 0.73 (with a low standard deviation of 0.002 across the three random seeds), solving 17% of the task. Remarkably, the RNAformer variant without recycling still achieves the second-best results (F1: 0.71; 14% solved), while the next best competitor is RNA-FM (F1: 0.67; 10% solved), followed by the ProbTransformer (F1: 0.63; 11% solved), UFold (F1: 0.63; 4% solved) and SPOT-RNA (F1: 0.59; 0.5% solved).

We conclude that recycling appears generally beneficial and that our model is capable of learning more complex structures, including non-canonical interactions and pseudoknots.

Furthermore, we note that RNA-FM leverages large-scale pre-training on vast amounts of sequence data while SPOT-RNA uses an ensemble of five models.

5.3 Learning Structure Predictions from Experimental Data

The current gold standard secondary structure data is experimental data that is obtained from 3D structure information provided by the PDB. The main difference to other data sources is that many samples contain base multiplets. However, one problem with experimental data from the PDB is that there is relatively little diversity with many samples belonging to few types of RNAs. Therefore, removing homologies between the training and the test data is even more important when dealing with experimental data to avoid overfitting. For our experiments, we therefore apply additional pre-processing pipelines that consider sequence and structure similarity.

However, the amount of high-quality data is limited and thus some of the recent methods approached the problem with finetuning [Singh et al., 2019a, Fu et al., 2022]. We adopt the finetuning strategy and pre-train a single model for all following experiments, using the strictest data pipeline, including covariance models to save computational costs. We finetune two models: One model that is finetuned using the similarity pipeline of SPOT-RNA [Singh et al., 2019a] to evaluate on TS1 and TS2, and one model that is finetuned with the similarity pipeline of SPOT-RNA2 [Singh et al., 2021a] to evaluate on the test sets TS1, TS2, TS3, and TS-Hard. We use a low-rank adaptation, instead of full-parameter finetuning, to reduce the trainable parameter count and decrease the memory consumption Hu et al. [2022].

5.3.1 Intra-Family Predictions on TS1 and TS2

Singh et al. [2019a] apply a sequence similarity cutoff of 80% between training and test samples, followed by a BLAST-search to further remove homologous sequences from the training set. This data pipeline is the strictest similarity pipeline applied so far for *de novo* prediction methods. However, RNA homologies are typically sequence- and structure-based, and thus the predictions can still be considered intra-family because the data pipeline only considers sequence similarity measures. We

Table 3: The mean performance of three fine-tunings with different random seeds of the RNAformer in comparison to the best competitors on the TS1 and TS2 benchmarks. We evaluated all competitors based on their open-sourced models.

Model	TS1		TS2	
	F1	F1-Shifted	F1	F1-Shifted
RNAformer $32M$	**0.743**	**0.776**	**0.805**	**0.836**
SPOT-RNA Singh et al. [2019a]	0.714	0.734	0.800	0.833
ContraFold Do et al. [2006]	0.625	0.651	0.761	0.793
ipknot Sato et al. [2011]	0.604	0.621	0.733	0.753
RNAFold Lorenz et al. [2011]	0.593	0.618	0.775	0.799
LinearFold-V Huang et al. [2019]	0.592	0.619	0.771	0.799
PKiss Janssen and Giegerich [2015]	0.535	0.555	0.767	0.789

compare the performance of the RNAformer and SPOT-RNA on the testsets TS1 and TS2, provided by Singh et al. [2019a] for SPOT-RNA [Singh et al., 2019a].

Results. In addition to SPOT-RNA, we report results for the following non-deep learning methods: ContraFold, ipKnot, RNAfold, LinearFold, and PKiss. The comparison of all methods is on the testsets TS1 and TS2 and shown in Table 3. For the RNAformer, we evaluate three different random seeds to assess the stability across different training runs.

The RNAformer achieves state-of-the-art performance, outperforming all other methods on TS1 with an average F1 score of 0.743 ($\pm$0.002), followed by SPOT-RNA with an F1 score of 0.714. The next best non-deep learning method, ContraFold, is far behind, achieving an F1 score of 0.625.

For the testset TS2, the results show less variance. However, the RNAformer still achieves the best result with an average F1 score of 0.805 ($\pm$ 0.007), again followed by SPOT-RNA. For the non-deep learning-based methods, RNAfold as well as LinearFold both show better performance than ContraFold, achieving F1 scores of 0.775, 0.771, and 0.761, respectively.

5.3.2 Comparison to Homology Modeling Methods

Table 4: The mean performance of three fine-tunings with different random seeds of the RNAformer in comparison to two models requiring MSA on the TS1-3 and TS-hard benchmarks. We evaluated SPOT-RNA2 based on their open-sourced model.

Model	Requires	TS1		TS2		TS3		TS-Hard	
	MSA	F1	F1-Shift	F1	F1-Shift	F1	F1-Shift	F1	F1-Shift
RNAformer $32M$		**0.739**	**0.771**	**0.802**	**0.837**	0.702	0.720	0.662	0.684
SPOT-RNA2	✗	0.737	0.769	0.790	0.836	**0.743**	**0.768**	**0.666**	**0.700**
CentroidAlifold*	✗	0.688	–	0.733	–	0.667	–	0.653	–

* Results obtained from SPOT-RNA2 [Singh et al., 2021a].

We also evaluate RNAformer in the same setting as the current state-of-the-art MSA-based method, SPOT-RNA2, using the testsets TS1, TS2, TS3, and TS-hard. To do so, and similar to SPOT-RNA2, we use all three pipelines to generate the training data. The resulting finetuning set contains 28% fewer training samples than the fine-tuning set from the previous experiment where we do not apply a covariance model to remove similarity and due to the additional similarity with TS3 and TS-hard. The resulting training set can be considered inter-family-based with respect to TS-hard due to the inclusion of sequence and structure-aware alignments and the resulting covariance models.

Results. We summarize the comparison against homology modeling methods in Table 4. We report the mean performance across three random seeds. The standard deviation lies between 0.002 and 0.007. As one can expect, the performance of the RNAformer on the test sets TS1 and TS2 slightly decreases in comparison to the previous experiment. This is likely a result of training samples that have been removed due to similarity to TS3 or TS-hard.

However, we observe that the RNAformer is on par with the current state-of-the-art homology modeling method SPOT-RNA2 while not utilizing MSA at all. The RNAformer achieves the best results on the test sets TS1 and TS2, while SPOT-RNA2 achieves better performance on the sets TS3 and TS-hard (F1 scores: 0.743 compared to 0.702 ± 0.007; 0.666 compared to 0.662 ± 0.004). Remarkably, the next best non-deep learning competitor, CentroidAlifold is clearly outperformed by both deep learning methods (results obtained from the SPOT-RNA2 publication [Singh et al., 2021a]).

6 Discussion

The RNAformer model, presented in our study, marks a notable advancement in the field of RNA secondary structure prediction. By demonstrating that a relatively simple deep learning architecture can reach state-of-the-art performance, the RNAformer challenges the prevailing notion in the field that complexity, through ensemble methods or the integration of multiple sequence alignments (MSAs), is necessary for high accuracy.

The scalability of the RNAformer with increasing model size is a promising feature. An increasing number of parameters in our model improves the performance rather than leads to overfitting the data. This could indicate that the model learns the underlying mechanics instead of just memorizing the data. However, with the current model size and with a reasonable sequence length of 500, we are already limited to training one sample per A100 GPU. Further scaling thus either requires larger GPUs or model-distributed training. Nevertheless, despite the computational demands of the modeling of the pairing matrix directly in the latent space and the axial attention based architecture, the inference per sequence requires less than a second on a modern GPU.

Another feature of our model is the effectiveness of fine-tuning on high-quality data, which compensates for the limitations of training on larger, but lower-quality datasets. This approach could have important implications for how training datasets are compiled and used, especially in the context of limited availability of high-quality RNA secondary structure data. Also, fine-tuning allows adapting an existing pre-trained model to new data without the need for a more expensive pertaining.

Furthermore, our advanced data processing pipelines emphasize the importance of careful dataset curation and managing data homology in RNA secondary structure prediction. While our approach helps to minimize biases, it also highlights the challenges in preparing datasets that are both comprehensive and unbiased, an area that requires ongoing attention.

Our approach to dataset curation and homology management could raise important considerations for data usage in the field. By highlighting the need for careful dataset preparation and the challenges of avoiding biases, our study contributes to setting standards for dataset creation and usage, which is crucial for the reliability and reproducibility of scientific research.

Future work involving our method could extend to the field of 3D RNA structure prediction. By adapting our 2D latent space approach, there is potential to enhance the accuracy and efficiency of 3D RNA structural modeling, a frontier that remains challenging.

7 Conclusion

We introduce RNAformer, a novel deep learning architecture for RNA secondary structure prediction. We utilize axial attention and latent space recycling, showing that our model's performance scales with model size. With a carefully designed data processing pipeline and fine-tuning strategies, we achieved state-of-the-art accuracy, outperforming existing *de novo* methods and performing on-par with existing homology-based (MSA-based) structure prediction methods. We address recent concerns on deep learning based methods by replicating a biophysical RNA folding algorithm, which demonstrates that our model is generally capable of learning a biophysical folding process. The accuracy of our method could potentially be enhanced by increasing the number of parameters, utilizing more data, or by employing an ensemble. However, because our model is open source, we can encourage the community to elaborate on these opportunities jointly.

Acknowledgments

This research was funded by the German Research Foundation (DFG) under SFB 1597 (SmallData), grant no. 499552394, and through grant no. 417962828. The authors gratefully acknowledge the Gauss Centre for Supercomputing e.V. (www.gauss-centre.eu) for funding this project by providing computing time on the GCS JUWELS Cluster at Jülich Supercomputing Centre (JSC). The authors acknowledge support by the state of Baden-Württemberg through bwHPC and the German Research Foundation (DFG) through grant no INST 39/963-1 FUGG. Finally, we acknowledge funding by the European Union (via ERC Consolidator Grant DeepLearning 2.0, grant no. 101045765). Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Research Council. Neither the European Union nor the granting authority can be held responsible for them.

Funded by the European Union

References

Quentin Vicens and Jeffrey S Kieft. Thoughts on how to think (and talk) about RNA structure. *Proceedings of the National Academy of Sciences*, 119(17):e2112677119, 2022.

Édouard Bonnet, Paweł Rzażewski, and Florian Sikora. Designing RNA secondary structures is hard. *Journal of Computational Biology*, 27(3):302–316, 2020.

Marcell Szikszai, Michael Wise, Amitava Datta, Max Ward, and David H Mathews. Deep learning models for RNA secondary structure prediction (probably) do not generalize across families. *Bioinformatics*, 38(16):3892–3899, 2022.

Jaswinder Singh, Kuldip Paliwal, Tongchuan Zhang, Jaspreet Singh, Thomas Litfin, and Yaoqi Zhou. Improved RNA secondary structure and tertiary base-pairing prediction using evolutionary profile, mutational coupling and two-dimensional transfer learning. *Bioinformatics*, 37, 2021a.

DH Mathews, J Sabina, M Zuker, and DH Turner. Expanded sequence dependence of thermodynamic parameters improves prediction of RNA secondary structure. *J Mol Biol*, 288(5):911–40, 1999.

Qi Zhao, Zheng Zhao, Xiaoya Fan, Zhengwei Yuan, Qian Mao, and Yudong Yao. Review of machine learning methods for rna secondary structure prediction. *PLoS computational biology*, 17(8): e1009291, 2021.

Jaswinder Singh, Jack Hanson, Kuldip Paliwal, and Yaoqi Zhou. RNA secondary structure prediction using an ensemble of two-dimensional deep neural networks and transfer learning. *Nature communications*, 10(1):1–13, 2019a.

David W Staple and Samuel E Butcher. Pseudoknots: RNA structures with diverse functions. *PLoS biology*, 3(6):e213, 2005.

Ivo Hofacker, Walter Fontana, Peter Stadler, Sebastian Bonhoeffer, Manfred Tacker, and Peter Schuster. Fast Folding and Comparison of RNA Secondary Structures. *Monatshefte fuer Chemie/Chemical Monthly*, 125:167–188, 02 1994.

Jiayang Chen, Zhihang Hu, Siqi Sun, Qingxiong Tan, Yixuan Wang, Qinze Yu, Licheng Zong, Liang Hong, Jin Xiao, Irwin King, et al. Interpretable rna foundation model from unannotated data for highly accurate rna structure and function predictions. *arXiv preprint arXiv:2204.00300*, 2022.

Kengo Sato, Manato Akiyama, and Yasubumi Sakakibara. RNA secondary structure prediction using deep learning with thermodynamic integration. *Nature communications*, 12(1):1–9, 2021.

Christoph Flamm, Julia Wielach, Michael T Wolfinger, Stefan Badelt, Ronny Lorenz, and Ivo L Hofacker. Caveats to deep learning approaches to RNA secondary structure prediction. *Biorxiv*, pages 2021–12, 2021.

Marek Justyna, Maciej Antczak, and Marta Szachniuk. Machine learning for RNA 2d structure prediction benchmarked on experimental data. *Briefings in Bioinformatics*, 24(3):bbad153, 2023.

Stephen F Altschul, Thomas L Madden, Alejandro A Schäffer, Jinghui Zhang, Zheng Zhang, Webb Miller, and David J Lipman. Gapped blast and psi-blast: a new generation of protein database search programs. *Nucleic acids research*, 25(17):3389–3402, 1997.

Jonathan Ho, Nal Kalchbrenner, Dirk Weissenborn, and Tim Salimans. Axial attention in multidimensional transformers. *arXiv preprint arXiv:1912.12180*, 2019.

John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, et al. Highly accurate protein structure prediction with alphafold. *Nature*, 596(7873):583–589, 2021.

Liang Huang, He Zhang, Dezhong Deng, Kai Zhao, Kaibo Liu, David A Hendrix, and David H Mathews. Linearfold: linear-time approximate RNA folding by 5'-to-3'dynamic programming and beam search. *Bioinformatics*, 35(14):i295–i304, 2019.

Kengo Sato, Yuki Kato, Michiaki Hamada, Tatsuya Akutsu, and Kiyoshi Asai. Ipknot: fast and accurate prediction of RNA secondary structures with pseudoknots using integer programming. *Bioinformatics*, 27(13):i85–i93, 2011.

H. Zhang et al. Linearpartition: linear-time approximation of RNA folding partition function and base-pairing probabilities. *Bioinformatics*, 36:258–267, 2020.

Chuong B Do, Daniel A Woods, and Serafim Batzoglou. CONTRAfold: RNA secondary structure prediction without physics-based models. *Bioinformatics*, 22(14):e90–e98, 2006.

Hannah K Wayment-Steele, Wipapat Kladwang, Alexandra I Strom, Jeehyung Lee, Adrien Treuille, Alex Becka, Eterna Participants, and Rhiju Das. RNA secondary structure packages evaluated and improved by high-throughput experiments. *Nature Methods*, 19(10):1234–1242, 2022.

J. Lee et al. RNA design rules from a massive open laboratory. *Proceedings of the National Academy of Sciences*, 111(6):2122–2127, 2014.

Corinna Theis, Stefan Janssen, and Robert Giegerich. Prediction of RNA secondary structure including kissing hairpin motifs. In *Algorithms in Bioinformatics: 10th International Workshop, WABI 2010, Liverpool, UK, September 6-8, 2010. Proceedings 10*, pages 52–64. Springer, 2010.

J. Singh et al. RNA secondary structure prediction using an ensemble of two-dimensional deep neural networks and transfer learning. *Nat Commun*, 10(5407), 2019b.

Mike Schuster and Kuldip K Paliwal. Bidirectional recurrent neural networks. *IEEE transactions on Signal Processing*, 45(11):2673–2681, 1997.

F. Yu and V. Koltun. Multi-scale context aggregation by dilated convolutions. *arXiv:1511.07122*, 2016.

J. Singh et al. Improved RNA secondary structure and tertiary base-pairing prediction using evolutionary profile, mutational coupling and two-dimensional transfer learning. *Bioinformatics*, 37(17): 2589–2600, 2021b.

Protein data bank: the single global archive for 3d macromolecular structure data. *Nucleic acids research*, 47(D1):D520–D528, 2019.

Laiyi Fu, Yingxin Cao, Jie Wu, Qinke Peng, Qing Nie, and Xiaohui Xie. Ufold: fast and accurate RNA secondary structure prediction with deep learning. *Nucleic acids research*, 50(3):e14–e14, 2022.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

Jörg Franke, Frederic Runge, and Frank Hutter. Probabilistic transformer: Modelling ambiguities and distributions for RNA folding and molecule design. *Advances in Neural Information Processing Systems*, 35:26856–26873, 2022.

Xinshi Chen, Yu Li, Ramzan Umarov, Xin Gao, and Le Song. RNA secondary structure prediction by learning unrolled algorithms. In *International Conference on Learning Representations*, 2020.

Zeming Lin, Halil Akin, Roshan Rao, Brian Hie, Zhongkai Zhu, Wenting Lu, Nikita Smetanin, Robert Verkuil, Ori Kabeli, Yaniv Shmueli, Allan dos Santos Costa, Maryam Fazel-Zarandi, Tom Sercu, Salvatore Candido, and Alexander Rives. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*, 379(6637):1123–1130, 2023.

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024.

Stefan Elfwing, Eiji Uchibe, and Kenji Doya. Sigmoid-weighted linear units for neural network function approximation in reinforcement learning. *Neural networks*, 107:3–11, 2018. Special issue on deep reinforcement learning.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

Frederic Runge, Karim Farid, Jorg KH Franke, and Frank Hutter. RnaBench: a comprehensive library for in silico RNA modelling. *bioRxiv*, pages 2024–01, 2024.

Padideh Danaee, Mason Rouches, Michelle Wiley, Dezhong Deng, Liang Huang, and David Hendrix. bpRNA: large-scale automated annotation and analysis of RNA secondary structure. *Nucleic acids research*, 46(11):5381–5394, 2018.

Michael F Sloma and David H Mathews. Exact calculation of loop formation probability identifies folding motifs in RNA secondary structures. *RNA*, 22(12):1808–1818, 2016.

Zhen Tan, Yinghan Fu, Gaurav Sharma, and David H Mathews. Turbofold ii: RNA structural alignment and secondary structure prediction informed by multiple homologs. *Nucleic acids research*, 45(20):11570–11581, 2017.

Mirela Andronescu, Vera Bereg, Holger H Hoos, and Anne Condon. RNA STRAND: the RNA secondary structure and statistical analysis database. *BMC bioinformatics*, 9:1–10, 2008.

Xiang-Jun Lu, Harmen J Bussemaker, and Wilma K Olson. Dssr: an integrated software tool for dissecting the spatial structure of RNA. *Nucleic acids research*, 43(21):e142–e142, 2015.

Limin Fu, Beifang Niu, Zhengwei Zhu, Sitao Wu, and Weizhong Li. Cd-hit: accelerated for clustering the next-generation sequencing data. *Bioinformatics*, 28(23):3150–3152, 2012.

Sebastian Will, Tejal Joshi, Ivo L Hofacker, Peter F Stadler, and Rolf Backofen. LocARNA-P: accurate boundary prediction and improved detection of structural RNAs. *RNA*, 18(5):900–914, 2012.

Eric P Nawrocki and Sean R Eddy. Infernal 1.1: 100-fold faster RNA homology searches. *Bioinformatics*, 29(22):2933–2935, 2013.

David H Mathews. How to benchmark RNA secondary structure prediction accuracy. *Methods*, 162: 60–67, 2019.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

Tri Dao, Daniel Y. Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. FlashAttention: Fast and memory-efficient exact attention with IO-awareness. In *Advances in Neural Information Processing Systems*, 2022.

Ronny Lorenz, Stephan H. Bernhart, Christian Höner zu Siederdissen, Hakim Tafer, Christoph Flamm, Peter F. Stadler, and Ivo L. Hofacker. ViennaRNA package 2.0. *Algorithms for Molecular Biology*, 6(1):26, Nov 2011. ISSN 1748-7188.

Ioanna Kalvari, Eric P Nawrocki, Nancy Ontiveros-Palacios, Joanna Argasinska, Kevin Lamkiewicz, Manja Marz, Sam Griffiths-Jones, Claire Toffano-Nioche, Daniel Gautheret, Zasha Weinberg, Elena Rivas, Sean R Eddy, Robert D Finn, Alex Bateman, and Anton I Petrov. Rfam 14: expanded coverage of metagenomic, viral and microRNA families. *Nucleic Acids Research*, 49(D1):D192–D200, 11 2020. ISSN 0305-1048.

Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022.

Stefan Janssen and Robert Giegerich. The RNA shapes studio. *Bioinformatics*, 31(3):423–425, 2015.

Supplementary Material

A A. Training Hyperparameters

Table 1: The hyperparameters of the *RNAformer* training.

Experiment Group	Parameter	BioPhys. Model	bpRNA	Exp. Data
		Value		
Training	GradientClipVal		1.0	
	MaxSteps	100k	50k	20k
	RandomIgnoreMat		0.4	
	Batch Size	8		
Optimizer	Optimizer		AdamW	
	LearningRate		0.001	
	WeightDecay		0.1	
	Betas		0.9 / 0.98	
	NumWarmupSteps		2000	
	LR Schedule		cosine annealing	
	LR DecayFactor		0.01	
Model	MaxLen	200	500	500
	Recycling		6	
	ResiDropout		0.4	
	EmbedDropout		0.4	
	InitializerRange		0.02	

Table 2: The hyperparameters of the *RNAformer* finetuning.

Category	Parameter	Value
LoRA	r	32
	Alpha	63
	Dropout	0.1
Finetune Layer	attn_pair_row.Wqkv	
	attn_pair_row.out_proj	
	attn_pair_col.Wqkv	
	attn_pair_col.out_proj	
	pair_transition.conv1	
	pair_transition.conv2	
Training	Epochs	4
	Batch Size	32
	RandomIgnoreMat	0.2
	Recycling	6
Optimizer	LearningRate	0.001
	WeightDecay	0.1
	GradientClipVal	0.1
	NumWarmupEpochs	2
	LR Schedule	cosine annealing
	LR DecayFactor	0.1

B B. Dataset

Table 3: Overview of datasets used in biophysical model experiment. This dataset is generated by inferring RNAfold.

		Length							
Dataset	# Samples	Min	Max	Mean	Median	# Families	Pseudoknots	Non-Canonical Base Pairs	Multiplets
Rfam-Train	410408	22	200	95.2	85.0	3796	0 (0.00%)	0 (0.00%)	0 (0.00%)
Rfam-Valid	2727	34	160	80.2	78.0	25	0 (0.00%)	0 (0.00%)	0 (0.00%)
Rfam-Test	3344	37	182	79.4	74.0	30	0 (0.00%)	0 (0.00%)	0 (0.00%)

Table 4: Overview of datasets used in bpRNA experiment. We filtered the training set with the use of sequence similarity against the test and validation set.

		Length						
Dataset	# Samples	Min	Max	Mean	Median	Pseudoknots	Non-Canonical Base Pairs	Multiplets
TR0	38184	13	500	128.0	99.0	6123 (16.04%)	23696 (62.06%)	2612 (6.84%)
VL0	1184	33	497	128.1	107.0	82 (6.93%)	870 (73.48%)	0 (0.00%)
TS0	1305	22	499	136.1	109.0	129 (9.89%)	947 (72.57%)	0 (0.00%)

Table 5: Overview of datasets used in intra-family predictions experiment. We filtered the training set with the use of sequence similarity and BLAST against the test and validation set. We only used the test, validation and fine-tuning set for the intra-family predictions experiment, the model was pretrained with the dataset from the homology modeling experiment.

Dataset	# Samples	Min	Max	Mean	Pseudoknots	Non-Canonical Base Pairs	Multiplets
Train	64535	13	500	130.5	9886 (15.32%)	36217 (56.12%)	3881 (6.01%)
Valid	1325	33	497	130.6	133 (10.04%)	964 (72.75%)	27 (2.04%)
Fine-tune	4824	19	500	104.6	3483 (72.20%)	4406 (91.33%)	3881 (80.45%)
TS1	67	33	189	74.8	56 (83.58%)	62 (92.54%)	53 (79.10%)
TS2	39	33	155	51.3	26 (66.67%)	38 (97.44%)	29 (74.36%)

Table 6: Overview of datasets used in homology modeling experiment. We filtered the training set with the use of sequence similarity, BLAST, and covariance models against the test and validation set. This training set is used for the intra-family predictions experiment and the homology modeling experiment.

		Length						
Dataset	# Samples	Min	Max	Mean	Median	Pseudoknots	Non-Canonical Base Pairs	Multiplets
Train	44091	13	500	141.3	110.0	7602 (17.24%)	26931 (61.08%)	2547 (5.78%)
Valid	1112	33	497	130.3	109.0	1029 (92.54%)	790 (71.04%)	15 (1.35%)
Fine-tune	3481	19	500	109.2	73.0	2171 (62.37%)	3066 (88.08%)	2547 (73.17%)
TS1	67	33	189	74.8	70.0	56 (83.58%)	62 (92.54%)	53 (79.10%)
TS2	39	33	155	51.3	42.0	26 (66.67%)	38 (97.44%)	29 (74.36%)
TS3	19	38	167	78.6	69.0	18 (94.74%)	18 (94.74%)	18 (94.74%)
TS-hard	28	34	189	65.6	50.5	20 (71.43%)	24 (85.71%)	21 (75.00%)

C C. Experiments

Table 7: Results of biophysical model experiment

	All base-pairs			Watson-Crick base-pairs			Wobble base-pairs			Non-canonical base-pairs			Canonical base-pairs			Pseudoknot base-pairs			Multiplet base-pairs		
	F1	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall
RNAformer $32M+$	0.967	0.962	0.973	0.905	0.848	0.974	0.197	0.114	0.898	-	-	-	0.967	0.962	0.973	-	-	-	-	-	-
RNAformer $32M$	0.948	0.941	0.956	0.887	0.831	0.957	0.192	0.110	0.879	-	-	-	0.948	0.941	0.956	-	-	-	-	-	-
RNAformer $8M$	0.919	0.900	0.944	0.861	0.796	0.946	0.182	0.105	0.861	-	-	-	0.919	0.900	0.944	-	-	-	-	-	-
RNAformer $2M$	0.846	0.822	0.878	0.794	0.729	0.882	0.162	0.093	0.783	-	-	-	0.846	0.822	0.878	-	-	-	-	-	-

Table 8: Results of bpRNA experiment

	All base-pairs			Watson-Crick base-pairs			Wobble base-pairs			Non-canonical base-pairs			Canonical base-pairs			Pseudoknot base-pairs			Multiplet base-pairs		
	F1	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall
RNAformer $32M+$	0.725	0.765	0.707	0.689	0.657	0.748	0.124	0.073	0.554	0.057	0.034	0.296	0.724	0.730	0.738	0.019	0.012	0.065	-	-	-
RNAformer $32M$	0.714	0.794	0.677	0.684	0.685	0.718	0.125	0.075	0.530	0.054	0.033	0.273	0.716	0.761	0.707	0.020	0.013	0.062	-	-	-
RNAformer $8M$	0.702	0.752	0.681	0.672	0.651	0.724	0.118	0.070	0.522	0.051	0.031	0.263	0.703	0.721	0.712	0.019	0.012	0.063	-	-	-
RNA-FM	0.667	0.664	0.695	0.644	0.586	0.750	0.107	0.062	0.544	0.029	0.017	0.188	0.676	0.648	0.738	0.014	0.009	0.047	-	-	-
UFold	0.630	0.611	0.674	0.621	0.552	0.745	0.103	0.059	0.543	0.000	0.000	0.001	0.653	0.611	0.732	0.008	0.005	0.035	-	-	-
ProbTransformer	0.625	0.665	0.612	0.598	0.577	0.653	0.108	0.063	0.484	0.042	0.025	0.214	0.628	0.640	0.643	0.012	0.008	0.044	-	-	-
SPOT-RNA	0.586	0.544	0.673	0.569	0.487	0.731	0.079	0.045	0.471	0.024	0.013	0.200	0.592	0.531	0.711	0.009	0.006	0.026	-	-	-
MXFold2	0.550	0.517	0.626	0.544	0.471	0.694	0.083	0.046	0.476	0.000	0.000	0.000	0.569	0.517	0.678	0.005	0.003	0.018	-	-	-

Table 9: Results of intra-family prediction experiment

	All base-pairs			Watson-Crick base-pairs			Wobble base-pairs			Non-canonical base-pairs			Canonical base-pairs			Pseudoknot base-pairs			Multiplet base-pairs		
	F1	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall
RNAformer	0.766	0.815	0.741	0.750	0.684	0.851	0.118	0.067	0.589	0.103	0.064	0.339	0.782	0.751	0.834	0.104	0.075	0.244	0.094	0.064	0.266
SPOT-RNA2	0.756	0.850	0.695	0.769	0.734	0.821	0.109	0.063	0.509	0.082	0.052	0.239	0.791	0.797	0.798	0.077	0.059	0.156	0.075	0.054	0.195
SPOT-RNA	0.745	0.864	0.676	0.772	0.758	0.811	0.110	0.063	0.484	0.067	0.043	0.203	0.793	0.821	0.786	0.066	0.049	0.128	0.080	0.059	0.190
ContraFold	0.675	0.809	0.594	0.742	0.751	0.751	0.098	0.057	0.389	0.000	0.000	0.000	0.756	0.809	0.722	0.055	0.054	0.088	0.061	0.048	0.135
RNAFold	0.660	0.792	0.577	0.724	0.735	0.729	0.097	0.056	0.381	0.000	0.000	0.000	0.738	0.792	0.701	0.041	0.040	0.065	0.052	0.041	0.122
LinearFold-V	0.658	0.790	0.575	0.722	0.734	0.726	0.097	0.057	0.386	0.000	0.000	0.000	0.736	0.790	0.699	0.041	0.040	0.065	0.052	0.041	0.122
ipknot	0.652	0.818	0.559	0.723	0.763	0.705	0.094	0.055	0.350	0.000	0.000	0.000	0.732	0.818	0.678	0.047	0.042	0.080	0.053	0.045	0.113
LinearFold-C	0.639	0.813	0.545	0.703	0.752	0.682	0.103	0.061	0.377	0.000	0.000	0.000	0.719	0.813	0.661	0.049	0.053	0.067	0.057	0.048	0.116
PKiss	0.620	0.730	0.552	0.682	0.678	0.699	0.087	0.051	0.334	0.001	0.000	0.001	0.693	0.729	0.671	0.070	0.057	0.112	0.051	0.039	0.114

Table 10: Results of homology modeling experiment

	All base-pairs			Watson-Crick base-pairs			Wobble base-pairs			Non-canonical base-pairs			Canonical base-pairs			Pseudoknot base-pairs			Multiplet base-pairs		
	F1	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall
SPOT-RNA2	0.738	0.833	0.679	0.757	0.723	0.811	0.104	0.060	0.490	0.077	0.050	0.221	0.777	0.783	0.786	0.076	0.059	0.153	0.080	0.058	0.207
RNAformer	0.736	0.785	0.711	0.728	0.663	0.829	0.110	0.063	0.549	0.094	0.059	0.290	0.757	0.726	0.810	0.095	0.068	0.219	0.093	0.064	0.257