


Structural bioinformatics

Deep learning models for RNA secondary structure prediction (probably) do not generalize across families

Marcell Szikszai ^{1,*}, Michael Wise^{1,2}, Amitava Datta¹, Max Ward^{1,3} and David H. Mathews⁴

¹Department of Computer Science & Software Engineering, The University of Western Australia, Perth, WA 6009, Australia, ²The Marshall Centre for Infectious Diseases Research and Training, The University of Western Australia, Perth, WA 6009, Australia, ³Department of Molecular and Cellular Biology, Harvard University, Cambridge, MA 02138, USA and ⁴Department of Biochemistry & Biophysics, Center for RNA Biology, and Department of Biostatistics & Computational Biology, University of Rochester, Rochester, NY 14642, USA

*To whom correspondence should be addressed.

Associate Editor: Yann Ponty

Received on March 28, 2022; revised on June 9, 2022; editorial decision on June 20, 2022; accepted on June 21, 2022

Abstract

Motivation: The secondary structure of RNA is of importance to its function. Over the last few years, several papers attempted to use machine learning to improve *de novo* RNA secondary structure prediction. Many of these papers report impressive results for intra-family predictions but seldom address the much more difficult (and practical) inter-family problem.

Results: We demonstrate that it is nearly trivial with convolutional neural networks to generate pseudo-free energy changes, modelled after structure mapping data that improve the accuracy of structure prediction for intra-family cases. We propose a more rigorous method for inter-family cross-validation that can be used to assess the performance of learning-based models. Using this method, we further demonstrate that intra-family performance is insufficient proof of generalization despite the widespread assumption in the literature and provide strong evidence that many existing learning-based models have not generalized inter-family.

Availability and implementation: Source code and data are available at <https://github.com/marcellsz/dl-rna>.

Contact: marcell.szikszai@research.uwa.edu.au

Supplementary information: [Supplementary data](#) are available at *Bioinformatics* online.

1 Introduction

Ribonucleic acid (RNA) molecules are extremely versatile polymers fulfilling numerous roles essential for life, including gene regulation and catalytic functions (Doudna and Cech, 2002; Serganov and Patel, 2007). Part of this versatility can be attributed to the structural diversity of RNA (Caprara and Nilsen, 2000). While chemically related to DNA, RNA often functions as a single strand. As a consequence, the molecules often fold back on themselves forming complex structures. It is well established that these folded configurations are of importance to the function of non-coding RNAs (ncRNAs) (Seetin and Mathews, 2012).

When discussing RNA, its structure is generally divided into a hierarchy of three levels. First, the foundation is the *primary structure*, which refers to the 1D sequence of the molecule. The sequences are made up of a succession of nucleobases, represented by four letters: adenine (A), cytosine (C), guanine (G) and uracil (U). Next, the *secondary structure*, which refers to the set of canonical base

pairings where bases are paired with one or zero other bases. For secondary structure, these pairs are formed by Watson–Crick base pairings (A–U, G–C) and by wobble G–U pairs. Finally—the last level generally considered—is the *tertiary structure* which refers to the 3D structure and the additional interactions that mediate the structure. However, since the secondary heavily informs the tertiary structure (Miao *et al.*, 2020; Shapiro *et al.*, 2007; Tinoco and Bustamante, 1999), the secondary structure is usually sufficient for developing some understanding of function.

Sequencing RNA molecules today is quick, inexpensive and accurate (Stark *et al.*, 2019); however, determining their structure is not. While high-resolution experimental techniques—such as nuclear magnetic resonance spectroscopy, X-ray crystallography and cryo-electron microscopy—exist, these methods tend to be expensive and time consuming. The contrast in the difficulty of determining sequence versus structure has created a sequence–structure gap, where there are vast amounts of sequenced RNA molecules without any known corresponding structure. In order to bridge this gap,

significant effort has gone into developing algorithms to predict RNA structures computationally (Hofacker, 2014; Miao *et al.*, 2020; Seetin and Mathews, 2012).

Broadly speaking, we can divide the secondary structure prediction methods into three categories: *homology modelling*, *comparative analysis* and *de novo* methods. Methods that start with nothing but the sequence, often termed *de novo*, have the advantage of being effective for single sequences without a need for homologous sequences. However, these *de novo* methods are not always accurate and have well-understood limitations (Ward *et al.*, 2017). These methods are generally implemented with dynamic programming-based tools and often make use of an underlying thermodynamic model to determine the minimum free energy (MFE) structures (Andronescu *et al.*, 2007, 2010; Lorenz *et al.*, 2011; Mathews *et al.*, 2016; Rivas, 2013). In contrast, homology modelling and comparative analysis are more accurate but require a set of homologous RNAs (and for homology modelling, their secondary structure). The sets of homologous RNA sequences are termed an RNA family (Kalvari *et al.*, 2021). These methods work by predicting a consensus structure that is conserved by evolution (Asai and Hamada, 2014; Havgaard and Gorodkin, 2014; Nawrocki and Eddy, 2013; Tan *et al.*, 2017).

In the last few years, a number of methodologies were published based on deep learning that report impressive results for RNA secondary structure prediction. However, many of these papers assess performance using *k*-fold cross-validation or simple train-test splits. We refer to these splits as *intra-family* (i.e. within-family), since there is no expectation that the families contained within the training and testing sets do not intersect. In contrast, we refer to splits where there is no such intersection as *inter-family* (i.e. between-family). Since the structure of RNA is highly conserved intra-family, performance derived from these metrics does not demonstrate generalization to novel RNAs (Rivas *et al.*, 2012). Homology modelling, and comparative analysis to an extent, is already well suited to the intra-family problem, and can not only determine the structure with high accuracy when used by domain experts, it can also provide other important insights about the molecule, such as its function (Nawrocki and Eddy, 2013). Because of this, the practical use cases of machine learning models with poor inter-family performance are limited.

2 Materials and methods

2.1 Demonstrative model

2.1.1 Basic concept

A common way to improve the performance of *de novo* tools is to utilize data from structure probing experiments. One example of such an approach is a technique by Deigan *et al.* (2009), that supplements dynamic programming-based methods via *selective 2'-hydroxyl acylation analysed by primer extension* (SHAPE) (Merino *et al.*, 2005; Wilkinson *et al.*, 2006), by which the experiment identifies nucleotides that are in more flexible regions of the secondary structure. SHAPE is an inexpensive probing experiment that scores the reactivity of each nucleotide in the RNA sequence. The reactivities found through SHAPE can be used to construct pseudo-free energy change terms for each nucleotide via the function,

$$\Delta G'(i) = m \log[\alpha(i) + 1] + b, \quad (1)$$

where $\alpha(i)$ is the SHAPE value for base i ; m and b are free parameters, and \log is the natural logarithm. Then, $\Delta G'$ is added as a free energy change term to each base pair stack involving nucleotide i in the *de novo* dynamic programming algorithm to improve predictive performance.

Our methodology looks to computationally mimic data from structure probing experiments, and ultimately construct pseudo-free energies that can be utilized by existing algorithms. This is similar to earlier work by Willmott *et al.* (2020) implementing *state inference*, whereby deep learning is used to estimate SHAPE-like scores.

We use this approach to construct a simple demonstrative model which shares many similarities with current learning-based efforts. We then show that our demonstrative model performs significantly

better than existing dynamic programming-based techniques for intra-family predictions. Finally, we show that our model performs poorly for inter-family predictions, demonstrating that intra-family performance does not necessarily generalize to inter-family cases.

Beyond our demonstrative model, we benchmarked or otherwise analysed several existing machine learning models for secondary structure prediction. See Section 2.4 for more details.

2.1.2 Network architecture

We implemented a convolutional neural network (CNN) for extracting per-nucleobase pairing probabilities from RNA sequences with the aim of constructing pseudo-free energies to improve secondary structure prediction performance. The extracted pairing probabilities are then converted to pseudo-free energies and fed to RNAstructure (Reuter and Mathews, 2010) version 6.3, which makes use of a conventional dynamic programming algorithm to find the MFE structures. A CNN was chosen for its ability to capture spatial information.

The architecture of the neural network is made up of two blocks, a convolution block and a fully connected block. The convolution block is comprised of two 1D convolution layers, with 256 filters each. The length of the kernels is 3, with strides of 1, and no dilation is applied. Each convolution layer is followed by a rectified linear unit (ReLU) activation layer. Spatial dropout (Tompson *et al.*, 2015) is applied between the convolution layers with a dropout rate of 0.25. The fully connected block is comprised of two fully connected layers, with 512 neurons each. Dropout is applied between the layers with a dropout rate of 0.25. The first activation is once again ReLU, and the final layer is followed by sigmoid activation,

$$\sigma(x) = \frac{1}{1 + e^{-x}}. \quad (2)$$

The network is trained with the Adam optimizer ($\gamma = 0.001$, $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 10^{-8}$) (Kingma and Ba, 2017) using binary cross-entropy loss and in mini-batches of 256. Early stopping is applied with a patience of 5.

2.1.3 Encoding

For our demonstrative model, the input nucleotide sequences are one-hot encoded as 2D matrices, where the nucleobases are represented by column vectors of size 4 and a sequence is the concatenation of these vectors. For example, a simple sequence UCG...AC is encoded as,

$$\mathbf{x} = \begin{bmatrix} \text{U} & \text{C} & \text{G} & \dots & \text{A} & \text{C} \\ 0 & 0 & 0 & \dots & 1 & 0 \\ 0 & 1 & 0 & \dots & 0 & 1 \\ 0 & 0 & 1 & \dots & 0 & 0 \\ 1 & 0 & 0 & \dots & 0 & 0 \end{bmatrix}. \quad (3)$$

The target structures' *shadows* are encoded as row vectors, classifying whether a particular base is paired or unpaired (without reference to the base-pairing partner). RNA secondary structures can be represented by dot-bracket notation, where unpaired nucleotides are represented by '.' characters, and paired nucleotides are represented by matches parentheses. Opening brackets indicate the 5'-nucleotide in a pair, and the matching closing brackets indicate the 3'-nucleotide in the pair.

These dot-bracket formatted structures can be easily converted to a structure's shadow for our demonstrative model. For example, the simple sequence-structure pair from Figure 1 is encoded as,

$$\mathbf{y} = [0 \ 0 \ 0 \ 1 \ 1 \ 1 \ 0 \ 0 \ 0 \ 1 \ 1 \ 1 \ 0]. \quad (4)$$

Both the sequences and structures are zero-padded at the 3' ends, to have shape (4×512) and (1×512) , respectively.

```
UCAUGCAGUCGCAC
...(((.....)))
```

Fig. 1. Example of a simple sequence-structure pair in dot-bracket format

2.1.4 Pseudo-free energy calculation

Since our neural network is designed to identify paired bases, $\hat{y}(i) \approx 1$ indicates a nucleotide is predicted as likely to be base paired, while $\hat{y}(i) \approx 0$ indicates a nucleotide is predicted as likely to be unpaired, where $\hat{y}(i)$ is the predicted shadow of the structure at base i . This differs from SHAPE, where values close to zero increase pairing likelihood, and values far from zero decrease it. Because of this, we apply the transformation,

$$\hat{\alpha}(i) = 1 - \hat{y}(i), \quad (5)$$

to our predicted values to produce SHAPE-like scores. No further normalization is performed. These scores can then be used via the pseudo-free energy equation by Deigan *et al.* (2009) (Equation 1) to improve the performance of the dynamic programming-based MFE folding algorithms. Note that this transformation is the equivalent of redefining the pseudo-free energy equation as,

$$\Delta G'(i) = m \log[2 - \hat{y}(i)] + b. \quad (6)$$

Also note that since our method makes use of RNAstructure's traditional dynamic programming algorithm, which cannot predict non-nested pairings, our method is unable to predict pseudoknots or multiplets. Such algorithms are widely used, and predicting pseudoknots is NP-Hard under most energy models (Lyngsø, 2004). However, the neural network itself is capable of identifying pseudo-knotted nucleotides as paired.

We performed grid search on the slope m , and intercept b of Equation 6 to refit the model for the values extracted by our neural network. The search was performed on the validation set, and an 11×11 grid of slope and intercept values were investigated via a reduced version of the parameter space used by Deigan *et al.* (2009). Because the training and validation sets contain RNA sequences from the same families, overfitting these pseudo-free energy parameters may also be a concern. Creating non-intersecting training and validation sets was found to be problematic due to the limited number of families available in the dataset. Therefore, to address the inter-family generalizability of the computationally found pseudo-free energies, we explored the behaviour of the thermodynamic nearest neighbour model with small, uniform pseudo-free energy changes applied. These small free energy nudges are completely general, so they eliminate any underlying bias, and allow us to investigate whether uniform changes to the model affect the performance of MFE folding differently across families. For each sequence in our data set, we performed folds using RNAstructure (Reuter and Mathews, 2010) with pseudo-free energy change terms $\Delta G'$ using parameters: $\alpha(i) = 0$, $m = 0$, and $b = \{-1.00, -0.98, \dots, 0.98, 1.00\}$, totalling to 101 folds per sequence. That is, we apply the pseudo-free energy change term from Equation 1 with $m = 0$, so our pseudo-free energy nudges are given by,

$$\Delta \Delta G' = b, \quad (7)$$

across the entire sequence. RNAstructure's default tenths precision was increased to hundredths for improved resolution.

2.2 Datasets

Our experiments call for a large number of reliably known sequence-structure pairs, diverse in families. The data set used for this purpose is ArchivelI (Sloma and Mathews, 2016). This dataset contains 3974 RNAs, across tRNAs (Jühling *et al.*, 2009), signal recognition particle (SRP) RNAs (Rosenblad *et al.*, 2003), telomerase RNAs (Griffiths-Jones *et al.*, 2005), 5S rRNAs (Szymanski *et al.*, 2000), 16S rRNAs, 23S rRNAs (Andronescu *et al.*, 2008; Cannone *et al.*, 2002), tmRNAs (Zwieb *et al.*, 2003), Group I (Andronescu *et al.*, 2008; Cannone *et al.*, 2002) and II Introns (Michel *et al.*, 1989) and RNase P RNAs (Brown, 1998).

Most folding algorithms have polynomial time complexities $O(n^k)$ with $k \geq 3$ (Ward *et al.*, 2019), and the algorithm employed by RNAstructure is $O(n^3)$ (Mathews *et al.*, 2004). Similarly, many of the learning-based models also suffer from significantly slower training and predictions for longer sequences. Because of this, we filter out 109 sequences longer than 512 nt to limit the runtime of our

Table 1. Breakdown of RNA families in ArchivelI after filtering

Family	Mean length	N
5S rRNA	119	1283
SRP RNA	180	918
tRNA	77	557
tmRNA	366	462
RNase P RNA	332	454
Group I Intron	375	74
16S rRNA ^a	317	67
Telomerase RNA	438	35
23S rRNA ^a	326	15
Mean	281	
Total		3865

^a16S rRNA and 23S rRNA are split into independent folding domains (Mathews *et al.*, 1999).

experiments, reducing our data set to 3865 RNAs. See Table 1 for a count of RNAs in each family after filtering.

2.3 Train-test split

To assess intra-family performance, we perform k -fold cross-validation with $k = 5$ on our entire dataset. We note that despite its wide use in the literature, it is our opinion that this type of train-test split cannot be used to assess the generalization of machine learning models for RNA secondary structure prediction. We stress that this metric is used only to demonstrate the ease of achieving high accuracy for the intra-family case.

For benchmarking inter-family performance, we perform *family-fold cross-validation*, such that one family is left out for testing per cross-validation fold. The motivation behind this is to measure the models' performance on novel RNAs that do not belong to a known family. This eliminates most of the homology to the training set, providing a fair measure of performance against other *de novo* tools.

In both cases, for early stopping and grid search, we use a validation set which is a 10% randomly selected subset of the training set.

2.4 Existing models

We benchmarked or otherwise analysed several machine learning models for secondary structure prediction with a focus on investigating inter-family versus intra-family performance. The models considered were: DMfold (Wang *et al.*, 2019), RPRes (Wang *et al.*, 2021), CROSS (Delli Ponti *et al.*, 2017), E2Efold (Chen *et al.*, 2019), SPOT-RNA (Singh *et al.*, 2019), MXfold2 (Sato *et al.*, 2021) and Ufold (Fu *et al.*, 2021) (Table 2).

Where possible, we re-trained networks using family-fold cross-validation and benchmarked them for more generalized performance. In cases where we were unable to re-train the network, we provide evidence that the training/testing split does not appropriately consider RNA homology. All mentioned papers address intra-family performance, usually with simple k -fold cross-validation, and in many cases wrongly conflate it with inter-family performance. The inter-family case is seldom mentioned, except by Sato *et al.* (2021) and Fu *et al.* (2021).

2.5 Benchmarking

To assess performance, we followed prior practice of calculating sensitivity and PPV (Mathews, 2019). We calculated the F_1 score as the harmonic mean of positive predictive value (PPV) and sensitivity. Pairs were allowed to be displaced by one nucleotide position on either side so that for base pair $(i)-(j)$ both $(i \pm 1) - (j)$ and $(i) - (j \pm 1)$ are considered valid (Mathews, 2019). Additionally, we performed two-tailed paired t-tests for statistical testing (Mathews, 2019), considering $P \leq 0.05$ significant.

Table 2. Recent papers that used machine learning for RNA secondary structure prediction

Name	Authors	Year	Method	Intra-family	Inter-family	Re-trained
CROSS	Delli Ponti <i>et al.</i>	2017	ANN ^a	✓	✗	✗
DMfold	Wang <i>et al.</i>	2019	LSTM ^b	✓	✗	✗
SPOT-RNA	Singh <i>et al.</i>	2019	CNN ^c + BLSTM ^d	✓	✗	✗
E2Efold	Chen <i>et al.</i>	2019	CNN ^c + Transformer ^e	✓	✗	✗
RNA-state-inf	Willmott <i>et al.</i>	2020	BLSTM ^d	✓	✓	✗
RPreS	Wang <i>et al.</i>	2021	BLSTM ^d + ResNet ^f	✓	✗	✗
MXfold2	Sato <i>et al.</i>	2021	BLSTM ^d + ResNet ^c	✓	✓	✓
UFold	Fu <i>et al.</i>	2021	CNN ^c	✓	✓	✓

Note: Inter-family and intra-family columns indicate the splitting methodology used in the paper, while the re-trained column indicates whether we have successfully re-trained the model on our dataset. Attempts were made to re-train nearly every model, however, many do not publish training methodology or could not be re-trained for another reason. See Sections 3.2 and 4.2 and the [Supplementary Information](#) for a detailed discussion on this.

^aArtificial neural network (Rumelhart *et al.*, 1986).

^bLong short-term memory neural network (Hochreiter and Schmidhuber, 1997).

^cConvolutional neural network (LeCun *et al.*, 1999).

^dBidirectional long short-term memory neural network (Schuster and Paliwal, 1997).

^eAttention transformer (Vaswani *et al.*, 2017).

^fResidual neural network (He *et al.*, 2016).

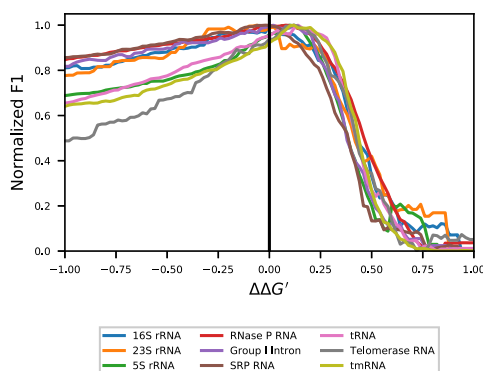


Fig. 2. Comparison of the effect of $\Delta\Delta G'$ nudges between families. The mean of all sequences in each family is calculated across the $\Delta\Delta G'$ values. F₁ scores have been normalized (min–max scaled) to account for the differences in underlying secondary structure prediction performance between families

Further, to assess the raw performance of our demonstrative model, we calculated the area under the curve of the receiver operating characteristic (AUC) for each RNA family. The receiver operating characteristic curve is constructed by plotting the sensitivity for predicting base pairing versus the false positive rate ($1 - \text{specificity}$) at different threshold values. This metric allows us to measure how well our model can capture the secondary structure's *shadow*. Specifically, the AUC can be interpreted as the probability that our model can correctly distinguish between a paired and an unpaired nucleotide, so we can gain a better understanding of how well our classifier performs prior to any pseudo-free energy calculations. It is worthwhile to note that while our method cannot predict pseudoknotted structures, the neural network is capable of identifying pseudoknotted nucleotides as paired. This is reflected in the AUC metric, but since the final structure cannot contain pseudoknots (Section 2.1.4), it is not reflected in the F₁ score.

3 Results

3.1 Demonstrative model

3.1.1 Grid search

Our experiments show that the behaviour of the thermodynamic nearest neighbour model is different across families for constant pseudo-free energy nudges (Equation 7) between -1.0 and 1.0 kcal/mol (Fig. 2). Effectively, a uniform nudge with a negative value

increases the stabilities of canonical pairs and nudges of positive value decrease the stabilities of canonical pairs. Much of these differences could be explained by noise; however, the dramatic deviation as $\Delta\Delta G' \rightarrow -1$ suggests that even under completely generalized inputs, such as these $\Delta\Delta G'$ nudges, the predictive performance of families is not uniformly affected. As an example, for some families like telomerase RNAs, the degradation in the negative region is much more extreme than in others like RNase P RNA. The differences are less clear when looking at $\Delta\Delta G' \rightarrow +1$, but are still present, especially when contrasting certain pairs like 23S RNA and tmRNA.

As expected, no $\Delta\Delta G'$ significantly improves performance across all families simultaneously. However, at least one region where the nudges improve performance can be found for all families with the exception of 16S rRNA. The region $\Delta\Delta G' \in [0.04, 0.10]$ improves the F₁ score of 5S rRNA, RNase P RNA, tRNA, telomerase RNA and tmRNA—although not necessarily significantly. Regions with significant improvements are $[0.06, 0.14]$ for 5S rRNA, $[0.20, 0.26]$ for tRNA, and $[0.02, 0.22]$ for tmRNA. The performance of the remaining families: Group I, 16S RNA, 23S RNA and SRP RNA is degraded within $[0.04, 0.10]$ with significantly worse performance for 16S RNA and SRP RNA.

The grid search for the slope (m) and intercept (b) free parameters of Equation 6 revealed that while the optimal values differ from those found with real SHAPE experiments (Deigan *et al.*, 2009; Hajdin *et al.*, 2013), the region of well-performing parameters is also present for generated SHAPE-like values (Supplementary Figs S1 and S2). Our small pseudo-free energy nudges showed that even inherently general changes to the existing thermodynamic nearest neighbour model do not affect the families uniformly, so overfitting the slope and intercept values to our training set is likely. To minimize the possible impact of overfitting these parameters towards families present in the training set, we elected to use $m = 1.8$ kcal/mol and $b = -0.6$ kcal/mol, as found by Hajdin *et al.* (2013) for experimentally generated SHAPE data. We expect that these parameters themselves are general and any overfitting to families is a result of the underlying generated SHAPE-like values.

3.1.2 Intra-family versus inter-family performance

Our demonstrative deep learning model (Section 2.1) shows improvements in F₁ score across most families (except Group I Intron and 23S rRNA), when benchmarked using k -fold cross-validation, over RNAstructure's baseline scores (Table 3). This confirms that our simple model is able to trivially improve intra-family predictions over traditional dynamic programming MFE algorithms. Note that this is true even with our conservatively chosen pseudo-free energy-free parameters, and the relatively high AUC values across all families suggest that more aggressive optimization would

Table 3. Performance of the demonstrative model separated by RNA family

Family	N	Baseline	<i>k</i> -fold		Family-fold	
		F ₁	AUC	F ₁	AUC	F ₁
5S rRNA	1283	0.63	0.95	0.94	0.72	0.46
SRP RNA	918	0.64	0.88	0.81	0.73	0.50
tRNA	557	0.80	0.97	0.97	0.79	0.65
tmRNA	462	0.43	0.82	0.64	0.68	0.41
RNase P RNA	454	0.55	0.81	0.66	0.71	0.48
Group I Intron	74	0.53	0.73	0.53	0.72	0.49
16S rRNA	67	0.58	0.77	0.60	0.72	0.48
Telomerase RNA	35	0.50	0.76	0.61	0.68	0.45
23S rRNA	15	0.73	0.79	0.68	0.73	0.54
Total	3865					
Mean		0.60	0.83	0.72	0.72	0.50

Note: F₁ score refers to the performance of secondary structure prediction, and AUC refers to the performance of predicting the structures' shadow via deep learning. The baseline is RNAstructure for free energy minimization without the deep learning input. Both *k*-fold and family-fold models are included.

likely yield even better results. Most of this can be attributed to the similarity of structures within particular families.

While we see significant improvements (Supplementary Information) using *k*-fold cross-validation, this is not true for family-fold cross-validation. The F₁ score is degraded across all families when compared to the baseline RNAstructure predictions, and AUC values are also significantly worse (see Supplementary Information) for all families when compared to *k*-fold cross-validation. It seems reasonable to expect that this is simply explained by training for too many epochs since there is strong family overlap between the training and validation sets (so early stopping does not prevent overfitting). However, examining the per-epoch performance of any split (Fig. 3) suggests that this is not the case, and the model never shows signs of generalizing for inter-family cases.

The 36% difference (F₁ = 0.72 to F₁ = 0.50, Table 3) in performance between intra-family versus inter-family cases is strong evidence that *k*-fold cross-validation is insufficient for benchmarking deep learning methods for RNA secondary structure prediction.

3.2 Existing models

In order to evaluate how well existing models generalize, we attempted to re-train all of their networks using family-fold cross-validation to benchmark inter-family performance (Table 4). Unfortunately, many of these tools do not publish their source code, particularly for training. Further, we were unable to re-train a number of models with public source code due to bugs in the code, which in some cases prevented us from being able to run their tools at all. Please see Table 2, Section 4.2 and the Supplementary Information for a detailed discussion on this.

4 Discussion

4.1 Demonstrative model

First, our results indicate that pseudo-free energy change terms affect RNA families differently. We propose that it is possible to overfit the estimation of these parameters to specific structures or families. For example, after adapting Deigan et al. (2009)'s equation (Equation 1) for alternate SHAPE-like values, refitting the parameters *m* and *b* requires careful consideration. This is especially true for any learning-based models attempting to improve RNA secondary structure prediction, since they require significant data to train already and may suffer from underlying overfitting issues. Our pseudo-free energy nudges have no inherent bias towards any

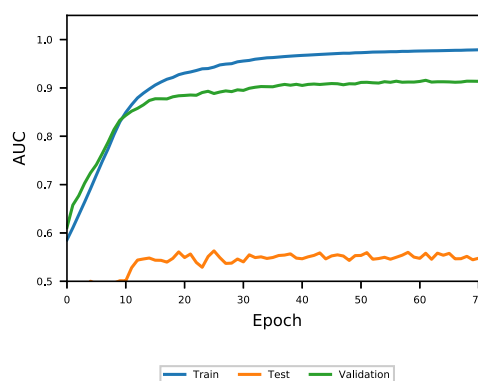


Fig. 3. Performance of family-fold testing on our demonstrative model. The training set is comprised of all families except 5S rRNA, the validation is a 10% split of the training set, while the testing set is 5S rRNAs. Note the consistently poor performance of the testing set throughout. (a) tRNA tdbR00000247. (b) tRNA tdbR00000372. (c) tRNA tdbR00000435

family, so it is possible that any model that is not completely general may suffer even more dramatically.

We were able to make use of RNAstructure's default parameters, found by jack-knife resampling (Hajdin et al., 2013) across several families, which has successfully eliminated issues with generalizability for real SHAPE experiments. These parameters were within the optimal region for our extracted SHAPE-like probing information, so the performance degradation is minimal. However, it is worth noting that intra-family performance can be further improved by optimizing *m* and *b* for the new distributions of α for base paired and unpaired nucleotides. Unfortunately, this can exacerbate issues with overfitting to the intra-family case even further.

In the case of the learned base-pairing probabilities, or more generally, in the case of all hyperparameter optimization tasks, creating unbiased training and validation sets is a challenge. After early stopping for our demonstrative model, we were able to look at the performance of the test set per epoch, and observe that we were not over-training. However, in a model that is able to generalize, our validation set would be insufficient for any sort of hyperparameter optimization. While the use of a training, validation and testing set is commonplace in machine learning tasks, for RNA homology, the overlap of the families within these sets is the most important consideration. Even considering sequence identity or similarity measures is not enough, as structure is so highly conserved amongst families. Ideally, in order to do hyperparameter optimization for learning-based models fairly, there can be no intersection between the families in the training, validation and testing sets. This can be difficult when the number of accurately known RNA structures in the dataset is fairly small and covers only a few families.

Second, our demonstrative model was able to achieve high AUC in the intra-family case; however, completely failed to generalize when it came to the inter-family case. This alone is evidence that it is insufficient to show good performance on intra-family predictions since it is, at the very least, possible to construct a model that does not work for practical applications and achieves high intra-family performance. Metrics like *k*-fold cross-validation as used for the demonstrative model, do not address RNA homology to any extent, since they do not address the intersection between families.

We suggest that benchmarking of learning-based methods for RNA secondary structure prediction be done by family-fold cross-validation in order to minimize the possibility of overfitting and accurately measure generalization. Previous work by Rivas et al. (2012), focusing on generative models instead of deep learning models, also supports our conclusions. For the purpose of fair benchmarking, a split is provided by this paper that attempts to minimize homology between training and testing sets as much as possible; however, it should be noted that the relatively small number of sequences in ArchiveII (Sloma and Mathews, 2016) means that we expect generalization to this dataset to be difficult. It should be

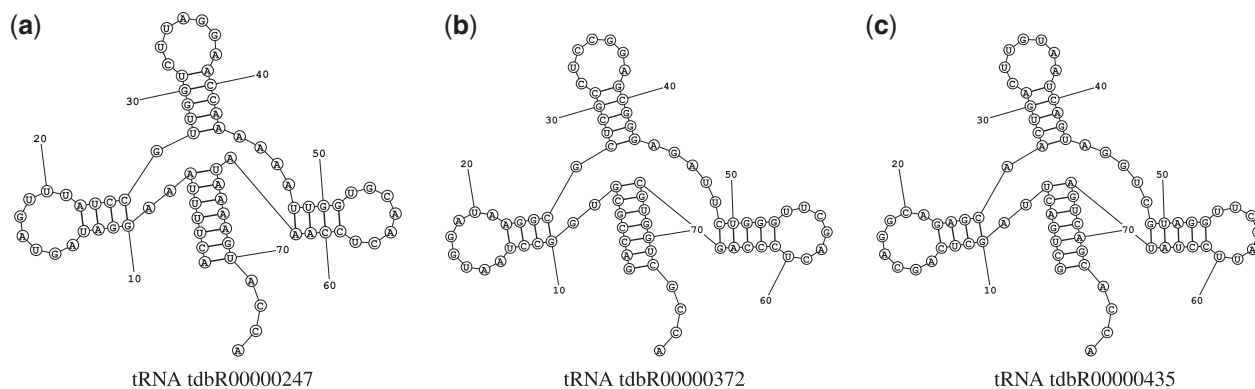


Fig. 4. Secondary structure of three tRNAs. Despite relatively low sequence identity (<60%), their secondary structures appear nearly identical. Many machine learning model benchmarks fail to separate these RNAs between the training and testing sets, causing significant overlap

Table 4. Performance of family-fold cross-validation on MXfold2 and Ufold

Family	RNAstructure	F ₁	
		MXfold2	Ufold
5S rRNA	0.63	0.54	0.53
SRP RNA	0.64	0.50	0.26
tRNA	0.80	0.64	0.26
tmRNA	0.43	0.46	0.40
RNase P RNA	0.55	0.51	0.41
Group I intron	0.53	0.45	0.45
16 S rRNA	0.58	0.55	0.41
Telomerase RNA	0.50	0.34	0.80
23S rRNA	0.73	0.64	0.45
Mean	0.60	0.51	0.44

noted that the split on families reduces the concerns on homology, but does not completely eliminate all concerns about generalization. tmRNA, for example, is tRNA-like and mRNA-like (Williams and Bartel, 1996). Therefore, the tRNA-like features could overtrain a model which cross-validation with tRNA would not reveal.

4.2 Existing models

For any machine learning model, an unbiased split of training and testing data is essential for benchmarking performance. In the case of biological data, this means considering the homology between these sets carefully in order to eliminate their overlap. Many current studies in RNA secondary structure prediction, especially those using learning-based models, do not appropriately address RNA homology. While it may be sufficient in many bioinformatics applications to consider sequence identity or sequence similarity, in the case of RNA, the structure is strongly conserved amongst families—often much more than sequence (Fig. 4). Because of this, it is possible (and highly probable) to create splits where despite considering sequence similarity, near-identical structures are present in both the training and testing data sets. Below is a breakdown of the training/testing split methodologies used by existing methods.

4.2.1 RNA-state-inference

While the authors of RNA-state-inference (Willmott *et al.*, 2020) do publish the entire source code on Github, we did not re-train their network due to their method's focus on single families.

The main results presented in the article are tested on a small set of 16 16S rRNAs used in SHAPE-directed experiments (Sükösd *et al.*, 2013), and trained on a large dataset of 17 032 16S rRNA sequences. Sequence similarity is addressed by removing training sequences with an over 10% match to any testing sequence, as well as training sequences that 'can be aligned such that they have

common nucleotides accounting for more than 80% of nucleotides of the shorter sequence' (Willmott *et al.*, 2020). This addresses sequence homology but does not address structure homology.

Finally, the paper does address poor inter-family generalization by also testing on 5S and 23S rRNAs. These test sets show weaker results (with an average accuracy of 0.514 for 5S rRNA and 0.611 for 23S rRNA) when compared to testing on 16S rRNA (with an average accuracy of 0.839) (Willmott *et al.*, 2020), supporting our conclusions.

4.2.2 CROSS and RPRes

Both Computational Recognition of Secondary Structure (CROSS) (Delli Ponti *et al.*, 2017), and RPRes (Wang *et al.*, 2021) are methods that attempt to recreate SHAPE experiments *in silico*, sharing many similarities with our demonstrative model. Unfortunately, no source code is provided for CROSS, and as such, we were unable to re-train their model on our dataset. While the authors of RPRes do publish the source code on Github, we were unable to re-train their network. See the [Supplementary Information](#) for more details.

Neither paper sufficiently addresses concerns regarding poor inter-family generalization. Both models are evaluated by training on one dataset at a time (PARS yeast, PARS human, HIV SHAPE, icSHAPE and high-quality nuclear magnetic resonance spectroscopy/X-ray crystallography structures) and testing on all others one by one. With this methodology, there is no guarantee, or indeed expectation, that the secondary structures in the datasets do not overlap. According to Delli Ponti *et al.* (2017) '[n]egligible overlap exists between training and testing sets' (Delli Ponti *et al.*, 2017) with Jaccard indices < 0.002 between each pair of datasets, where $Jaccard(S_1, S_2) = (S_1 \cap S_2) / (S_1 \cup S_2)$ for sequences S_1 and S_2 . This addresses sequence similarity but does not comprehensively address inter-family cases.

4.2.3 DMfold

While the authors of DMfold (Wang *et al.*, 2019) do publish the entire source code on Github, we were unable to re-train their network. See the [Supplementary Information](#) for more details.

The train/test split methodology used by DMfold produces sets which heavily overlap families. After using their packaged tools for generating the splits, we found that all testing families were covered in the training set, without any consideration to RNA homology whatsoever. In this case, the training set contained 2111 RNAs, with 957 5S rRNAs, 437 tRNAs, 377 RNase P RNAs and 340 tmRNAs, while the testing set contained 234 RNAs, with 102 5S rRNA, 49 tRNAs, 45 RNase P RNAs and 38 tmRNAs. This set contains many identical, or nearly identical structures between the training and testing sets, with a mean minimum tree edit distance of 14.16, compared to the 134.99 of our family-fold cross-validation splits. See the [Supplementary Information](#) for more details.

4.2.4 E2Efold

While the authors of E2Efold (Chen et al., 2019) do publish the entire source code on Github, we did not re-train their network due to high memory requirements. However, other recent publications have already pointed out E2Efold's poor inter-family performance, reporting F_1 scores as low as $F_1 = 0.036$ (Sato et al., 2021; Fu et al., 2021) on the bpRNA-new dataset.

The original E2Efold benchmarks use stratified sampling, generating train/test splits which heavily overlap families. The training set, based on RNAStralign (Tan et al., 2017), contained 24 895 RNAs, with 9325 16 s rRNAs, 7687 5S rRNAs, 5412 tRNAs, 1243 Group I Introns, 379 SRP RNAs, 431 tmRNAs, 360 RNase P RNAs and 28 telomerase RNAs. The first testing set, based on RNAStralign once again, contained 2825 RNAs, with 1150 16 s rRNAs, 879 5S rRNAs, 504 tRNAs, 136 Group I Introns, 53 SRP RNAs, 61 tmRNAs, 37 RNase P RNAs and 5 telomerase RNAs. The second training set, based on ArchiveII (Sloma and Mathews, 2016), explicitly only contained families that overlap with the RNAStralign dataset.

4.2.5 MXfold2 and UFold

Our re-training of MXfold2 (Sato et al., 2021) and UFold (Fu et al., 2021) with family-fold cross-validation indicates that the models do not generalize well to inter-family performance. However, as previously pointed out, it could be argued that our tests are particularly hard due to the small number of families in our dataset.

Sato et al. (2021) did address inter-family performance using their own bpRNA-new dataset for which they reported positive results. To address inter-family performance, the model is trained on bpRNA-1m, a dataset derived from Rfam 12.2 (Danaee et al., 2018). The model is then tested on bpRNA-new, which is derived from a newer version of Rfam (14.2) (Griffiths-Jones et al., 2005). Newly discovered and novel RNA families are extracted from Rfam 14.2 making up the bpRNA-new testing set.

Since this testing set does not share any families with the training set, we expect that good performance on this split provides reasonable evidence for generalization. It should be noted however, that these results are still less robust than our proposed family-fold cross-validation, since secondary structures amongst families are often similar in Rfam, particularly within 'clans', which are 'group[s] of families that either share a common ancestor but are too divergent to be reasonably aligned or a group of families that could be aligned, but have distinct functions' (<https://docs.rfam.org/en/latest/glossary.html#clan>). For example, the SRP clan is divided into nine separate families. There are clear homologs within this clan, such as, for example, Fungi_SRP and Metazoa_SRP. Because of this, to firmly address inter-family generalization, Rfam families should also be split by clan. Additionally, reporting of the performance should always be broken down by family to provide context about generalization.

UFold (Fu et al., 2021) applies the same testing methodology as MXfold2, and reports similarly positive results, although according to self-reported metrics (Fu et al., 2021) on the bpRNA-new dataset, both tools are outperformed by Eternafold (Wayment-Steele et al., 2021), a multitask-learning-based method that uses a crowdsourced RNA design dataset (Lee et al., 2014) to train a model, and Contrafold (Do et al., 2006), a statistical learning method that uses conditional log-linear models.

4.2.6 SPOT-RNA

SPOT-RNA (Singh et al., 2019) does not provide the source code for training, or the ability to re-train the model. As such, we were unable to evaluate it on our dataset.

Singh et al. (2019) initially pre-trained the model on bpRNA-1m (Danaee et al., 2018), and then applied transfer learning to train, validate and test on a small set of 217 high-resolution structures. Both the pre-training and training sets are separated from the testing set by filtering based on 80% sequence identity, and BLAST-N (Altschul et al., 1997) is used to address homology with an e-value cut-off of 10. While better than relying solely on a sequence-identity

cutoff, BLAST-N itself is a sequence similarity-based metric and does not address the secondary structure of the RNAs, meaning that this split cannot be considered inter-family. Indeed, the self-reported improvement of 20% ($F_1 = 0.49$ to $F_1 = 0.58$) over the dynamic programming-based RNAfold (Lorenz et al., 2011), becomes a 3% ($F_1 = 0.62$ to $F_1 = 0.60$) (Sato et al., 2021) deterioration when benchmarked on MXfold2's bpRNA-new dataset (Sato et al., 2021) of novel families.

5 Conclusion

Our results show that a basic CNN can be used to construct pseudo-free energies to improve secondary structure prediction for intra-family cases. We proposed the more rigorous testing methodology of family-fold cross-validation, which along with our model was used to demonstrate that intra-family performance does not guarantee generalization to inter-family cases. We argued that k -fold cross-validation is an unsuitable method for benchmarking deep learning RNA secondary structure prediction models. Finally, we used these findings as evidence that many recent publications wrongly conflate intra-family with inter-family results, and that this results in inflated self-reported accuracy.

Future work in this area will have start by addressing the more general problem of predicting pairedness on artificial data, which removes any biases present due to data availability. Current research is handicapped by the limited number of RNA families and the high-resolution structures available. A recent pre-print by Flamm et al. (2021) showed that approximating thermodynamics-based folding algorithms is not trivial with deep learning, even when arbitrary amounts of data are available.

Additionally, the curation of larger datasets of reliably known sequence-structure pairs will be important for future tools. While there are larger datasets than ArchiveII (Sloma and Mathews, 2016), such as those by Leontis and Zirbel (2012) and Becquey et al. (2021), these still require significant work before they are suitable for deep learning to improve RNA secondary structure prediction. In particular, the splitting of these datasets for intra-family performance measurement requires careful consideration.

Finally, we found that $\Delta\Delta G'$ nudges could improve the structure prediction performance, although no value generalized across families. This suggests that there are limitations in the thermodynamic nearest neighbour model that manifest differently across families. Elucidating the structure-specific limitations might lead to improvements in the parametrization.

Funding

This work was supported by an Australian Government Research Training Program (RTP) Scholarship (to Marcell Szikszai), and by a U.S. National Institutes of Health grant [R01GM076485 to David H. Mathews].

Conflict of interest: none declared.

References

- Altschul, S.F. et al. (1997) Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res.*, 25, 3389–3402.
- Andronescu, M. et al. (2007) Efficient parameter estimation for RNA secondary structure prediction. *Bioinformatics (Oxford, England)*, 23, i19–28.
- Andronescu, M. et al. (2008) RNA STRAND: the RNA secondary structure and statistical analysis database. *BMC Bioinformatics*, 9, 340.
- Andronescu, M. et al. (2010) Computational approaches for RNA energy parameter estimation. *RNA*, 16, 2304–2318.
- Asai, K. and Hamada, M. (2014) RNA structural alignments, part II: non-Sankoff approaches for structural alignments. *Methods Mol. Biol. (Clifton, NJ)*, 1097, 291–301.
- Becquey, L. et al. (2021) RNA-Net: an automatically built dual-source dataset integrating homologous sequences and RNA structures. *Bioinformatics*, 37, 1218–1224.

- Brown, J.W. (1998) The ribonuclease P database. *Nucleic Acids Res.*, **26**, 351–352.
- Cannone, J.J. *et al.* (2002) The comparative RNA web (CRW) site: an online database of comparative sequence and structure information for ribosomal, intron, and other RNAs. *BMC Bioinformatics*, **3**, 2.
- Caprara, M.G. and Nilsen, T.W. (2000) RNA: versatility in form and function. *Nat. Struct. Biol.*, **7**, 831–833.
- Chen, X. *et al.* (2019). RNA secondary structure prediction by learning unrolled algorithms. In: International Conference on Learning Representations. <https://github.com/ml4bio/e2efold#references>.
- Danae, P. *et al.* (2018) bpRNA: large-scale automated annotation and analysis of RNA secondary structure. *Nucleic Acids Res.*, **46**, 5381–5394.
- Deigan, K.E. *et al.* (2009) Accurate SHAPE-directed RNA structure determination. *Proc. Natl. Acad. Sci. USA*, **106**, 97–102.
- Delli Ponti, R. *et al.* (2017) A high-throughput approach to profile RNA structure. *Nucleic Acids Res.*, **45**, e35.
- Do, C.B. *et al.* (2006) CONTRAfold: RNA secondary structure prediction without physics-based models. *Bioinformatics*, **22**, e90–e98.
- Doudna, J.A. and Cech, T.R. (2002) The chemical repertoire of natural ribozymes. *Nature*, **418**, 222–228.
- Flamm, C. *et al.* (2021). Caveats to deep learning approaches to RNA secondary structure prediction. *bioRxiv*, <https://doi.org/10.1101/2021.12.14.472648>.
- Fu, L. *et al.* (2021) UFold: fast and accurate RNA secondary structure prediction with deep learning. *Nucleic Acids Res.*, **50**, e14.
- Griffiths-Jones, S. *et al.* (2005) Rfam: annotating non-coding RNAs in complete genomes. *Nucleic Acids Res.*, **33**, D121–D124.
- Hajdin, C.E. *et al.* (2013) Accurate SHAPE-directed RNA secondary structure modeling, including pseudoknots. *Proc. Natl. Acad. Sci. USA*, **110**, 5498–5503.
- Havgaard, J.H. and Gorodkin, J. (2014) RNA structural alignments, part I: Sankoff-based approaches for structural alignments. *Methods Mol. Biol. (Clifton, NJ)*, **1097**, 275–290.
- He, K. *et al.* (2016). Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778.
- Hochreiter, S. and Schmidhuber, J. (1997) Long short-term memory. *Neural Comput.*, **9**, 1735–1780.
- Hofacker, I.L. (2014) Energy-directed RNA structure prediction. *Methods in Molecular Biology (Clifton, NJ)*, **1097**, 71–84.
- Jühling, F. *et al.* (2009) tRNAdb 2009: compilation of tRNA sequences and tRNA genes. *Nucleic Acids Res.*, **37**, D159–162.
- Kalvari, I. *et al.* (2021) Rfam 14: expanded coverage of metagenomic, viral and microRNA families. *Nucleic Acids Res.*, **49**, D192–D200.
- Kingma, D.P. and Ba, J. (2017) Adam: a method for stochastic optimization. *arXiv, 1412.6980 [cs.LG]*.
- LeCun, Y. *et al.* (1999). Object recognition with gradient-based learning. In: Forsyth, D. A., *et al.* (eds) *Shape, Contour and Grouping in Computer Vision, Lecture Notes in Computer Science*. Springer, Berlin, Heidelberg, pp. 319–345.
- Lee, J. *et al.*; EteRNA Participants. (2014) RNA design rules from a massive open laboratory. *Proc. Natl. Acad. Sci. USA*, **111**, 2122–2127.
- Leontis, N.B. and Zirbel, C.L. (2012). Nonredundant 3D structure datasets for RNA knowledge extraction and benchmarking. In: Leontis, N. and Westhof, E. (eds) *RNA 3D Structure Analysis and Prediction, Nucleic Acids and Molecular Biology*. Springer, Berlin, Heidelberg, pp. 281–298.
- Lorenz, R. *et al.* (2011) ViennaRNA package 2.0. *Algorithms Mol. Biol.*, **6**, 26.
- Lyngsø, R.B. (2004). Complexity of pseudoknot prediction in simple models. In: Díaz, J. *et al.*, (eds) *Automata, Languages and Programming, Lecture Notes in Computer Science*. Springer, Berlin, Heidelberg, pp. 919–931.
- Mathews, D.H. (2019) How to benchmark RNA secondary structure prediction accuracy. *Methods (San Diego, CA)*, **162–163**, 60–67.
- Mathews, D.H. *et al.* (1999) Expanded sequence dependence of thermodynamic parameters improves prediction of RNA secondary structure. *J. Mol. Biol.*, **288**, 911–940.
- Mathews, D.H. *et al.* (2004) Incorporating chemical modification constraints into a dynamic programming algorithm for prediction of RNA secondary structure. *Proc. Natl. Acad. Sci. USA*, **101**, 7287–7292.
- Mathews, D.H. *et al.* (2016) RNA secondary structure prediction. *Curr. Protoc. Nucleic Acid Chem.*, **67**, 11.2.1–11.2.19.
- Merino, E.J. *et al.* (2005) RNA structure analysis at single nucleotide resolution by selective 2'-hydroxyl acylation and primer extension (SHAPE). *J. Am. Chem. Soc.*, **127**, 4223–4231.
- Miao, Z. *et al.* (2020) RNA-Puzzles round IV: 3D structure predictions of four ribozymes and two aptamers. *RNA*, **26**, 982–995.
- Michel, F. *et al.* (1989) Comparative and functional anatomy of group II catalytic introns – a review. *Gene*, **82**, 5–30.
- Nawrocki, E.P. and Eddy, S.R. (2013) Infernal 1.1: 100-fold faster RNA homology searches. *Bioinformatics*, **29**, 2933–2935.
- Reuter, J.S. and Mathews, D.H. (2010) RNAstructure: software for RNA secondary structure prediction and analysis. *BMC Bioinformatics*, **11**, 129.
- Rivas, E. (2013) The four ingredients of single-sequence RNA secondary structure prediction. A unifying perspective. *RNA Biol.*, **10**, 1185–1196.
- Rivas, E. *et al.* (2012) A range of complex probabilistic models for RNA secondary structure prediction that includes the nearest-neighbor model and more. *RNA (New York, NY)*, **18**, 193–212.
- Rosenblad, M.A. *et al.* (2003) SRPDB: signal recognition particle database. *Nucleic Acids Res.*, **31**, 363–364.
- Rumelhart, D.E. *et al.* (1986) Learning representations by back-propagating errors. *Nature*, **323**, 533–536.
- Sato, K. *et al.* (2021) RNA secondary structure prediction using deep learning with thermodynamic integration. *Nat. Commun.*, **12**, 941.
- Schuster, M. and Paliwal, K. (1997) Bidirectional recurrent neural networks. *IEEE Trans. Signal Process.*, **45**, 2673–2681.
- Seetin, M.G. and Mathews, D.H. (2012) RNA structure prediction: an overview of methods. *Methods Mol. Biol. (Clifton, NJ)*, **905**, 99–122.
- Serganov, A. and Patel, D.J. (2007) Ribozymes, riboswitches and beyond: Regulation of gene expression without proteins. *Nat. Rev. Genet.*, **8**, 776–790.
- Shapiro, B.A. *et al.* (2007) Bridging the gap in RNA structure prediction. *Curr. Opin. Struct. Biol.*, **17**, 157–165.
- Singh, J. *et al.* (2019) RNA secondary structure prediction using an ensemble of two-dimensional deep neural networks and transfer learning. *Nat. Commun.*, **10**, 5407.
- Sloma, M.F. and Mathews, D.H. (2016) Exact calculation of loop formation probability identifies folding motifs in RNA secondary structures. *RNA (New York, NY)*, **22**, 1808–1818.
- Stark, R. *et al.* (2019) RNA sequencing: the teenage years. *Nat. Rev. Genet.*, **20**, 631–656.
- Sükkösd, Z. *et al.* (2013) Evaluating the accuracy of SHAPE-directed RNA secondary structure predictions. *Nucleic Acids Res.*, **41**, 2807–2816.
- Szymanski, M. *et al.* (2000) 5S ribosomal RNA database Y2K. *Nucleic Acids Res.*, **28**, 166–167.
- Tan, Z. *et al.* (2017) TurboFold II: RNA structural alignment and secondary structure prediction informed by multiple homologs. *Nucleic Acids Res.*, **45**, 11570–11581.
- Tinoco, I. and Bustamante, C. (1999) How RNA folds. *J. Mol. Biol.*, **293**, 271–281.
- Tompson, J. *et al.* (2015) Efficient object localization using convolutional networks. *arXiv, 1411.4280 [cs.CV]*.
- Vaswani, A. *et al.* (2017). Attention is all you need. *arXiv, 1706.03762 [cs.CL]*.
- Wang, L. *et al.* (2019) DMfold: a novel method to predict RNA secondary structure with pseudoknots based on deep learning and improved base pair maximization principle. *Front. Genet.*, **10**, 143.
- Wang, L. *et al.* (2021) A novel end-to-end method to predict RNA secondary structure profile based on bidirectional LSTM and residual neural network. *BMC Bioinformatics*, **22**, 169.
- Ward, M. *et al.* (2017) Advanced multi-loop algorithms for RNA secondary structure prediction reveal that the simplest model is best. *Nucleic Acids Res.*, **45**, 8541–8550.
- Ward, M. *et al.* (2019) Determining parameters for non-linear models of multi-loop free energy change. *Bioinformatics (Oxford, England)*, **35**, 4298–4306.
- Waynt-Steele, H.K. *et al.* (2021). RNA secondary structure packages evaluated and improved by high-throughput experiments. *bioRxiv*, <https://doi.org/10.1101/2020.05.29.124511>.
- Wilkinson, K.A. *et al.* (2006) Selective 2'-hydroxyl acylation analyzed by primer extension (SHAPE): quantitative RNA structure analysis at single nucleotide resolution. *Nat. Protoc.*, **1**, 1610–1616.
- Williams, K.P. and Bartel, D.P. (1996) Phylogenetic analysis of tmRNA secondary structure. *RNA*, **2**, 1306–1310.
- Willmott, D. *et al.* (2020) Improving RNA secondary structure prediction via state inference with deep recurrent neural networks. *Comput. Math. Biophys.*, **8**, 36–50.
- Zwieb, C. *et al.* (2003) tmRDB (tmRNA database). *Nucleic Acids Res.*, **31**, 446–447.