Caveats to deep learning approaches to RNA secondary structure prediction

Christoph Flamm[1], Julia Wielach[1], Michael T. Wolfinger[1,2], Stefan Badelt[1], Ronny Lorenz[1], and Ivo L. Hofacker[1,2,*]

[1]Department of Theoretical Chemistry, University of Vienna, Währingerstraße 17, 1090 Vienna, Austria

[2]Research Group Bioinformatics and Computational Biology, Faculty of Computer Science, University of Vienna, Währingerstraße 29, 1090 Vienna, Austria

Abstract

Machine learning (ML) and in particular deep learning techniques have gained popularity for predicting structures from biopolymer sequences. An interesting case is the prediction of RNA secondary structures, where well established biophysics based methods exist. These methods even yield exact solutions under certain simplifying assumptions. Nevertheless, the accuracy of these classical methods is limited and has seen little improvement over the last decade. This makes it an attractive target for machine learning and consequently several deep learning models have been proposed in recent years. In this contribution we discuss limitations of current approaches, in particular due to biases in the training data. Furthermore, we propose to study capabilities and limitations of ML models by first applying them on synthetic data that can not only be generated in arbitrary amounts, but are also guaranteed to be free of biases. We apply this idea by testing several ML models of varying complexity. Finally, we show that the best models are capable of capturing many, but not all, properties of RNA secondary structures. Most severely, the number of predicted base pairs scales quadratically with sequence length, even though a secondary structure can only accommodate a linear number of pairs.

Keywords: RNA secondary structure, folding prediction, dataset biases, deep learning model, biophysical model

*To whom correspondence should be addressed: ivo@tbi.univie.ac.at

1 Introduction

Many RNAs rely on a well defined structure to exert their biological function. Moreover, many RNA functions can be understood without knowledge of the full tertiary structure, relying only on secondary structure, i.e. the pattern of Watson-Crick type base pairs formed when the RNA strand folds back onto itself. Prediction of RNA secondary structure from sequence is therefore a topic of long-standing interest for RNA biology and several computational approaches have been developed for this task. The most common approach is "energy directed" folding, where (in the simplest case) the structure of lowest free energy is predicted. The corresponding energy model is typically the Turner nearest-neighbor model [17], which compiles free energies of small structure motifs (loops) derived from UV melting experiments.

Under some simplifying assumptions, such as neglecting pseudo-knots and base triples, the optimal structure can be computed using efficient dynamic programming algorithms that solve the folding problem in $\mathcal{O}(n^3)$ time for a sequence of length n. While these algorithms yield an optimal solution given the model, the accuracy achieved on known secondary structures varies widely and averages about 67% in a benchmark accompanying the latest Turner parameter set [9]. While a variety of factors contribute to the inaccuracy of prediction, accuracy has hardly changed in comparison to the previous iteration of energy parameters [10], suggesting that it is the simplifying assumptions of the model, rather than measurement errors in the UV melting experiments, that limits prediction accuracy. It is therefore tempting to forego the simplifying assumptions necessary for dynamic programming and approach the problem using machine learning techniques. Inspired by the recent success of deep learning methods in protein structure prediction, several groups have proposed deep learning methods for the RNA secondary structure prediction problem [3, 16, 15, 7].

A major problem for all deep learning approaches is the limited availability of training data. Even before the recent machine learning boom, several works have attempted to replace or improve the Turner energy parameters by training on a set of known RNA secondary structures [6, 1, 19]. While these works demonstrated that learning energy parameters is feasible, they often reported overly optimistic accuracies. While it is common practice to ensure that test and training sets do not contain very similar sequences (e.g. with more than 80% identity), this is not sufficient to avoid overtraining. Ideally, test and training sets should be constructed from distinct RNA families. As shown in [13] setting up test/training sets that are *structurally* distinct leads to a significant drop in accuracy and largely eliminates any advantages of the trained over measured parameters.

Given the data hungry nature of deep networks, this becomes an even more pressing problem when deep learning is applied to structure prediction. The currently most used training set is the bpRNA set [5] which contains over 100000 distinct sequences. While the number of sequences in the set is sufficient to train sophisticated models, the structural diversity of the data set is limited. 55% of the sequences are ribosomal RNAs (rRNAs) from the Comparative RNA website[2]. The

next largest data source is the Rfam database [11], providing 43% of sequences. At first glance, this subset seems more diverse, since Rfam release 12.0, used for bpRNA, comprises 2450 RNA families. Again, however, rRNA and tRNAs make up over 90% of the sequences in Rfam 12.0. The dataset is therefore dominated by just four RNA families (three types of rRNA and tRNAs) and it seems highly unlikely that it can capture the full variety of the RNA structure space. This also reflected in the extremely uneven length distribution of sequences in bpRNA, see Fig. S1.

When both test and training set are derived from bpRNA, they will exhibit the same biases leading to unrealistically good benchmark results. The MXfold2 paper [15] addressed this problem by generating an additional data set, bpRNAnew, containing only sequences from Rfam families added after the 12.0 release. The bpRNAnew set was also used in the Ufold paper [7] to distinguish between within-family and cross-family performance. Arguably, within-family performance is largely irrelevant. Structure prediction for sequences belonging to a known family, should always proceed by identifying the RNA family and mapping the novel sequence to the consensus structure, e.g. using covariance models and the Infernal software [12]; this is in fact how most of the structures in the bpRNA set were generated. Only sequences that cannot be assigned to a known family should be subjected to structure prediction from sequence.

2 Training on artificial data

The fact that most known RNA structures are derived from a very small set of RNA families makes it hard to distinguish between shortcomings due to the biased training data and more fundamental problems in deep learning for RNA structures. To become independent of available structures we therefore propose to test deep learning methods on completely synthetic data sets generated by classical energy directed structure prediction methods. This allows to test the capabilities of neural network (NN) architectures to learn the essential characteristics of RNA secondary structure, and to explore their learning behavior without worrying that the network learns to exploit biases of the training set. The most promising architectures can, of course, be re-trained later with real world data.

In this contribution we use RNAfold from the ViennaRNA package [8] to fold random sequences allowing us to generate arbitrary large data sets and guarantee complete independence of all sequences in training and evaluation data sets. Most results shown below use a training set consisting of random sequences (equal A,U,C,G content) with a homogeneous length of 70 nt, but we also constructed further data sets with four different length distributions, to study scenarios where test and evaluation set follow different length distributions.

3 Predicting pairedness

In order to examine what can and cannot be easily predicted by deep learning approaches, we first consider a simplified problem. Rather than predicting base pairs, we restrict ourselves to predict whether a nucleotide is paired or unpaired, in other words if the nucleotide, in the context of RNA secondary structure, belongs to a helix or a loop region. Since this results in a much smaller structure space, one might expect the prediction problem to become easier to learn. This also corresponds to the traditional approach in protein secondary structure prediction, where each amino acid is predicted to be in one of three states (alpha helix, beta sheet, or coil) while ignoring which residues form hydrogen bonds to each other in a beta sheet. Note also that chemical probing of RNA structures [18] typically yields information on pairedness only.

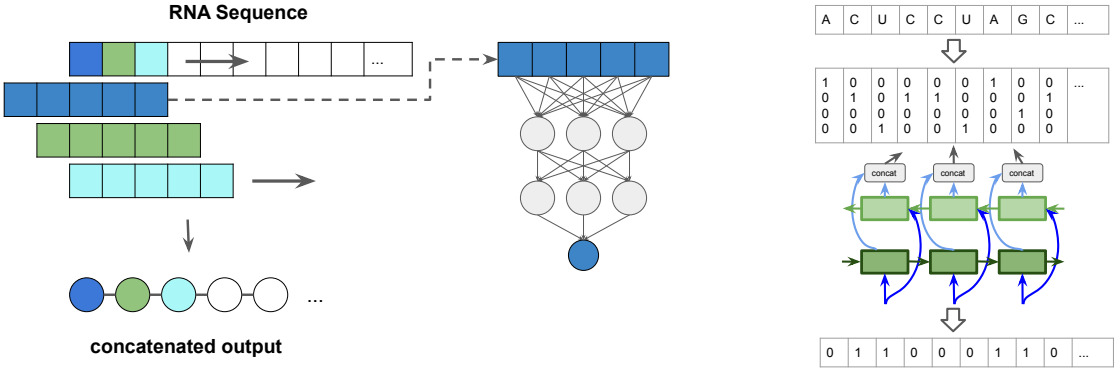


Figure 1: **Paired / unpaired prediction approach:** (left) sliding-window: A window, consisting of a central symbol and context in the form of a fixed number of leading and tailing symbols is slid along the sequence. The output sequence is a concatenation of the single predictions per window position. (right) schematic representation of the input / output encoding for the bidirectional long short term memory (BLSTM) neural network. The detailed network architectures can be seen in Fig. S2.

We implemented three different types of predictors: (i) a simple feed forward network (FFN) that examines sequence windows and predicts the state of the central residue, (ii) a more complex 1D convolutional neural network (CNN), again working on sequence windows, and (iii) a bi-directional long short term memory (BLSTM) network, see Fig. 1. We tested several window sizes for the sliding window approaches (i and ii) and varied the number of layers and neurons in the BLSTM. The FFN architecture is inspired by classical protein secondary structure predictors, such as PHD [14].

The resulting performance when training on sequences of length 70 is shown in Table 1. While the BLSTM performed slightly better than the simpler sliding window approaches, none of the predictors achieve a satisfactory performance. This is most obvious when focusing on the Matthews Correlation Coefficient (MCC) [4]. For this task, an accuracy of 0.5 corresponds to pure chance and thus the networks did little more than learn that “A” nucleotides have a higher

Modeltype	Parameters	Epochs	Accuracy	F1	Loss	MCC
BLSTM	1 Layer, 40 Neurons	43	0.667	0.594	0.609	0.166
	1 Layer, 80 Neurons	27	0.664	0.589	0.612	0.168
	3 Layers, 40 Neurons	38	0.676	0.609	0.604	0.207
Sliding Window	Window 15	89	0.654	0.559	0.623	0.120
	Window 35	94	0.659	0.559	0.620	0.118
	Window 71	59	0.661	0.569	0.618	0.118
CNN Sliding Window	Window 15	67	0.660	0.588	0.616	0.156
	Window 35	65	0.666	0.586	0.609	0.166
	Window 71	30	0.668	0.580	0.608	0.170

Table 1: **Performance of the paired / unpaired prediction:** The performances on the validation set of 2000 sequences of length 70 for all models trained on 80000 sequences of length 70 for 100 epochs. After 100 epochs the best performing model is chosen based on maximum validation MCC. The epoch in which this performance is reached can also be seen in the table. The metrics used are accuracy, F1, loss and MCC. All values are rounded to three decimal places.

propensity to be unpaired than “G”s. Our results also indicate that the performance does not improve by increasing the number of neurons, or by using more training data (results not shown). The results were also consistent for different datasets and different training runs.

The poor performance suggests that the short cut simply doesn’t work. Pairedness cannot be predicted independently of the full secondary structure. Moreover, RNA secondary structure is apparently too non-local for sliding window approaches to succeed. This is also in contrast to the fact that RNA secondary structure formation is thought to be largely independent of tertiary structure.

4 Predicting base pair matrices

To account for the non-locality of secondary structure, recent deep learning approaches for RNA secondary structure have focused on predicting base pairing matrices. In the typical approach a sequence of length n is expanded to a $n \times n$ matrix, where each entry corresponds to a possible base pair. Convolutional networks (or variants thereof) are then used to predict an output matrix containing the predicted pairs, i.e. a 1 in row i and column j indicates that nucleotides i and j form a pair. Various postprocessing steps can be appended to derive a valid secondary structure from the pair matrix. Since we were interested in analysing the performance of the network, we avoided any sophisticated postprocessing and either directly analysed the output matrix (with values between 0 and 1), or obtained a single secondary structure by retaining only the highest entry per row, rounding to obtain values of 0 or 1, and removing pseudo-knots.

For our experiments we re-implemented the SPOT-RNA network [16], a deep network employing ResNets (residual networks), fully connected layers and 2D BLSTMs, see Fig. S3. We

implemented three variants, corresponding to Models 0, 1, and 3, in the SPOT-RNA paper, that differ in the size of the different blocks (only Model 3 contains the BLSTM part).

We first tested the simple scenario where all sequences in the training and evaluation sets have the same length of 70 nt. The three models achieved a performance in terms of MCC of 0.554 for model 0, 0.580 for model 1 and 0.640 for model 3. This is quite similar to the values reported for SPOT-RNA after initial training, though models 0 and 1 perform slightly worse in our case. Since model 3, the only one containing a BLSTM block, had the best overall performance, the results below are shown only for model 3.

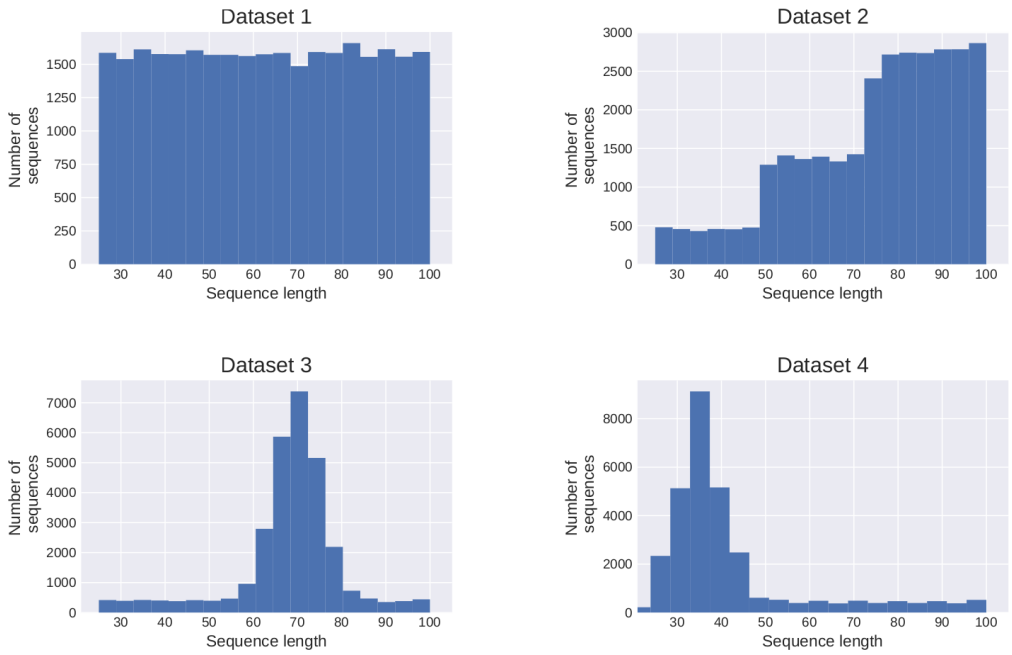


Figure 2: Length distribution of the four synthetic datasets used for prediction of base pair matrices.

The bpRNA data set shows a very uneven distribution of sequence lengths, with most sequences in the range of 70-120 nt, the length of tRNAs and 5S rRNAs (see Fig S1). We therefore explored scenaria where the length distribution of sequences in test and training set differs, by generating 4 synthetic data sets with sequences of 25–100 nt, but markedly different length distributions, see Fig. 2. In each case, the training set consisted of 30000 and the validation set 5000 different random sequences.

We then trained and evaluated our networks on all 16 combinations of training and evaluation sets. Results for Model 3 are shown in Table 2. Even though the datasets were restricted to a rather small range of lengths, from 25 to 100 bases, notable differences are already observable. In general, performance on validation set 4 is best, simply because it contains mostly very short sequences whose structures are easier to predict. Conversely, networks trained on set 4 perform poorly on longer sequences. In addition, we usually observe better performance when training

Training set	Validation set				Performance
	1	2	3	4	(training set)
1	0.64	0.59	0.61	0.71	0.72
2	0.61	0.58	0.59	0.68	0.66
3	0.64	0.60	0.62	0.70	0.71
4	0.63	0.57	0.59	0.75	0.87

Table 2: The performances of all combinations of training and validation data sets for the four distributions shown in Figure 2. The diagonal in red shows the performance, when training and validation dataset have the same distribution.

and evaluation set follow the same length distribution, as seen in the diagonal entries of the table. This happens even though all sets are perfectly independent.

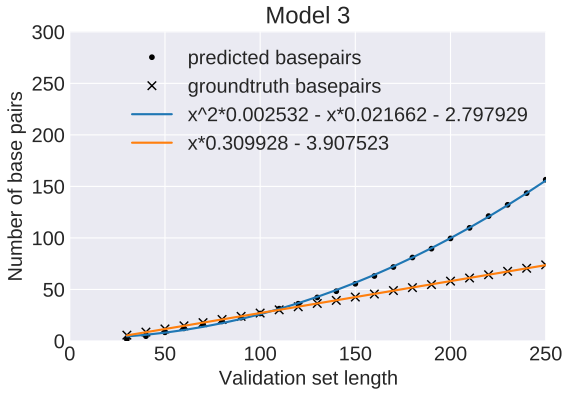


Figure 3: **Predicted number of base pairs:** Average number of base pairs predicted by model 3 (bullets) and in the ground truth data set (crosses) for 2000 sequence per length bin (30-250). The blue and orange curves are least-square regression fits of the data points. The ML-model predicts a wrong quadratic growth (blue curve) for the number of base pairs in contrast to a correct linear growth (orange line).

To further analyze how predictions change with sequence length we generated a series of evaluation sets, varying sequence length from 30 to 250 nt. The number of base pairs is expected to grow linearly with sequence length, since a structure of length n must form less than $n/2$ pairs. The ground truth provided by RNAfold perfectly follows the expected behavior. However, for all three networks the number of base pairs, as measured by the number of entries in the output matrix > 0.5, grows quadratically. This happens, because the output matrix has n^2 entries and asymptotically, the networks predict a constant fraction of all possible base pairs.

This failure of the network models to reproduce the correct asymptotic behavior exemplifies that it is much easier to learn local properties than global ones. We therefore compared the statistics for several additional structural properties between NN predicted structures and the RNAfold ground truth.

As can be seen in Table 3, the network almost perfectly recapitulates the relative frequency

Frequency of bases in context external loop (EL), bulge loop (BL), hairpin loop (HL), internal loop (IL), multi loop (ML), base pairs (bps)							
model / length	paired	EL	BL	HL	IL	ML	
VRNA / 70	0.508	0.176	0.033	0.156	0.114	0.014	
NN / 70	0.445	0.222	0.027	0.161	0.127	0.019	
VRNA / 100	0.541	0.123	0.031	0.143	0.126	0.035	
NN / 100	0.433	0.185	0.030	0.146	0.152	0.053	
Average number of structural element							
model / length	helix	EL	BL	HL	IL	ML	
VRNA / 70	4.825	0.992	1.112	1.754	1.841	0.118	
NN / 70	4.354	0.993	0.840	1.730	1.686	0.098	
VRNA / 100	7.132	0.991	1.586	2.314	2.889	0.343	
NN / 100	6.146	0.991	1.080	2.135	2.632	0.299	
Relative frequency of base pair types)							
model / length	GC	CG	AU	UA	GU	UG	NC
VRNA / 70	0.257	0.262	0.169	0.170	0.071	0.071	0.00
NN / 70	0.258	0.260	0.170	0.172	0.070	0.070	$9.63 \cdot 10^{-5}$
VRNA / 100	0.262	0.255	0.173	0.170	0.068	0.071	0.00
NN / 100	0.257	0.252	0.177	0.175	0.068	0.070	$2.30 \cdot 10^{-5}$

Table 3: **Predicted structural features** for RNAfold (VRNA) and Model 3 (NN) trained on sequences of length 70. The test sets consisted of 2000 sequences each of lengths 70 and 100.

of GC vs AU vs GC pairs and essentially never predicts non-canonical pairs. Frequency and length of hairpin and interior loops are learned quite well. The largest discrepancy is observed for multi-loops, where the network predicts more nucleotides in multi-loops even though there it predicts fewer such loops. Indeed the median length of multi-loops at sequence length 100 is 9 for RNAfold and 16 for model 3. , do we observe some deviations, where the network predicts fewer but on average longer loops. Mulit-loops are, of course, harder to learn since they are rarer than the other types and also less local.

5 Conclusion

The performance of deep networks is strongly dependent on quantity and quality of the available training data. This makes it hard to study the capabilities and shortcomings of the networks independently of the available data. This problem can be avoided if there is a way to generate synthetic training data that are statistically sufficiently similar to real data. For RNA secondary structure prediction, algorithms that compute the minimum free energy structure via dynamic programming can provide such a data source.

While recent RNA secondary structure data sets provide a large number of training sequences, this comes at the expense of making the data set extremely unbalanced, with more than 95% of sequences deriving from ribosomal RNAs or tRNAs.

Our experiments show that networks are sensitive to biases in training sets, in that, for

example, performance suffers when training set and evaluation set follow a different length distribution. In general, networks trained on synthetic data can reproduce many local features of RNA structures, such as the prevalence of different types of base pairs and loops, almost perfectly. At the same time, the networks struggle to correctly reproduce global properties and scaling behavior, as exemplified by the fact that for all networks the number of predicted base pairs scales quadratically with sequence length. While this behavior can easily be addressed during postprocessing, it is not clear whether that would correct or merely hide the underlying problem.

Author Contributions

ILH and CF concepted and supervised the study. JW implemented the machine learning models and the code to generate training and test data sets. JW further trained and evaluated the machine learning models and made predictions using the Google Colab environment. SB helped with the statistical analysis of predictions made with the trained machine-learning models. MTW and RL were involved in planing and supervision of the machine learning experiments. ILH and CF wrote the first draft of the manuscript. All authors contributed actively to rewriting the draft into the final manuscript. The final manuscript has been read and approved by all authors.

Funding

This work was supported in part by grants from the Austrian Science Foundation (FWF), grant numbers F 80 to ILH and I 4520 to RL.

Acknowledgments

This is a short text to acknowledge the contributions of specific colleagues, institutions, or agencies that aided the efforts of the authors.

Supplemental Data

Additional figures with details on the neural network architectures can be found in the supplement.

Data Availability Statement

Python notebooks implementing our models and data files are available at `https://github.com/ViennaRNA/RNAdeep`

References

[1] Andronescu, M., Condon, A., Hoos, H. H., Mathews, D. H., and Murphy, K. P. (2010). Computational approaches for RNA energy parameter estimation. *RNA* 16, 2304–18. doi:10.1261/rna.1950510

[2] Cannone, J. J., Subramanian, S., Schnare, M. N., Collett, J. R., D'Souza, L. M., Du, Y., 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. doi:10.1186/1471-2105-3-2

[3] Chen, X., Li, Y., Umarov, R., Gao, X., and Song, L. (2019). RNA secondary structure prediction by learning unrolled algorithms. In *International Conference on Learning Representations*

[4] Chicco, D. and Jurman, G. (2020). The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. *BMC Genomics* 21. doi:10.1186/s12864-019-6413-7

[5] Danaee, P., Rouches, M., Wiley, M., Deng, D., Huang, L., and Hendrix, D. (2018). bpRNA: large-scale automated annotation and analysis of RNA secondary structure. *Nucleic Acids Res* 46, 5381–5394. doi:10.1093/nar/gky285

[6] Do, C. B., Woods, D. A., and Batzoglou, S. (2006). Contrafold: RNA secondary structure prediction without physics-based models. *Bioinformatics* 22, e90–8. doi: 10.1093/bioinformatics/btl246

[7] Fu, L., Cao, Y., Wu, J., Peng, Q., Nie, Q., and Xie, X. (2021). Ufold: fast and accurate RNA secondary structure prediction with deep learning. *Nucleic Acids Res* doi:10.1093/nar/gkab1074

[8] Lorenz, R., Bernhart, S. H., Höner zu Siederdissen, C., Tafer, H., Flamm, C., Stadler, P. F., et al. (2011). ViennaRNA Package 2.0. *Algo Mol Biol* 6, 26. doi:10.1186/1748-7188-6-26

[9] Mathews, D. H., Disney, M. D., Childs, J. L., Schroeder, S. J., Zuker, M., and Turner, D. H. (2004). Incorporating chemical modification constraints into a dynamic programming algorithm for prediction of RNA secondary structure. *Proc Natl Acad Sci U S A* 101, 7287–92. doi:10.1073/pnas.0401799101

[10] Mathews, D. H., Sabina, J., Zuker, M., and Turner, D. H. (1999). Expanded sequence dependence of thermodynamic parameters improves prediction of RNA secondary structure. *J Mol Biol* 288, 911–40. doi:10.1006/jmbi.1999.2700

[11] Nawrocki, E. P., Burge, S. W., Bateman, A., Daub, J., Eberhardt, R. Y., Eddy, S. R., et al. (2015). Rfam 12.0: updates to the RNA families database. *Nucleic Acids Res* 43, D130–7. doi:10.1093/nar/gku1063

[12] Nawrocki, E. P. and Eddy, S. R. (2013). Infernal 1.1: 100-fold faster RNA homology searches. *Bioinformatics* 29, 2933–5. doi:10.1093/bioinformatics/btt509

[13] Rivas, E. (2013). The four ingredients of single-sequence RNA secondary structure prediction. a unifying perspective. *RNA Biology* 10, 1185–1196. doi:10.4161/rna.24971

[14] Rost, B. and Sander, C. (1993). Prediction of protein secondary structure at better than 70% accuracy. *J Mol Biol* 232, 584–99. doi:10.1006/jmbi.1993.1413

[15] Sato, K., Akiyama, M., and Sakakibara, Y. (2021). RNA secondary structure prediction using deep learning with thermodynamic integration. *Nat Commun* 12, 941. doi:10.1038/s41467-021-21194-4

[16] Singh, J., Hanson, J., Paliwal, K., and Zhou, Y. (2019). RNA secondary structure prediction using an ensemble of two-dimensional deep neural networks and transfer learning. *Nat Commun* 10, 5407. doi:10.1038/s41467-019-13395-9

[17] Turner, D. H. and Mathews, D. H. (2010). Nndb: the nearest neighbor parameter database for predicting stability of nucleic acid secondary structure. *Nucleic Acids Res* 38, D280–2. doi:10.1093/nar/gkp892

[18] Weeks, K. M. (2010). Advances in RNA structure analysis by chemical probing. *Curr Opin Struct Biol* 20, 295–304. doi:10.1016/j.sbi.2010.04.001

[19] Zakov, S., Goldberg, Y., Elhadad, M., and Ziv-Ukelson, M. (2011). Rich parameterization improves RNA structure prediction. *J Comput Biol* 18, 1525–42. doi:10.1089/cmb.2011.0184

Caveats to deep learning approaches to RNA secondary structure prediction - Supplementary Material -

Christoph Flamm[1], Julia Wielach[1], Michael T. Wolfinger[1,2], Stefan Badelt[1], Ronny Lorenz[1], and Ivo L. Hofacker[1,2,*]

[1]Department of Theoretical Chemistry, University of Vienna, Währingerstraße 17, 1090 Vienna, Austria

[2]Research Group Bioinformatics and Computational Biology, Faculty of Computer Science, University of Vienna, Währingerstraße 29, 1090 Vienna, Austria

1 Supplementary Data

The ML-modles and the trainingsdata can be found on github `https://github.com/ViennaRNA/RNAdeep`

2 Supplementary Tables and Figures

2.1 Figures

References

[1] Singh, J., Hanson, J., Paliwal, K., and Zhou, Y. (2019). RNA secondary structure prediction using an ensemble of two-dimensional deep neural networks and transfer learning. *Nat Commun* 10, 5407. doi:10.1038/s41467-019-13395-9

*To whom correspondence should be addressed: `ivo@tbi.univie.ac.at`

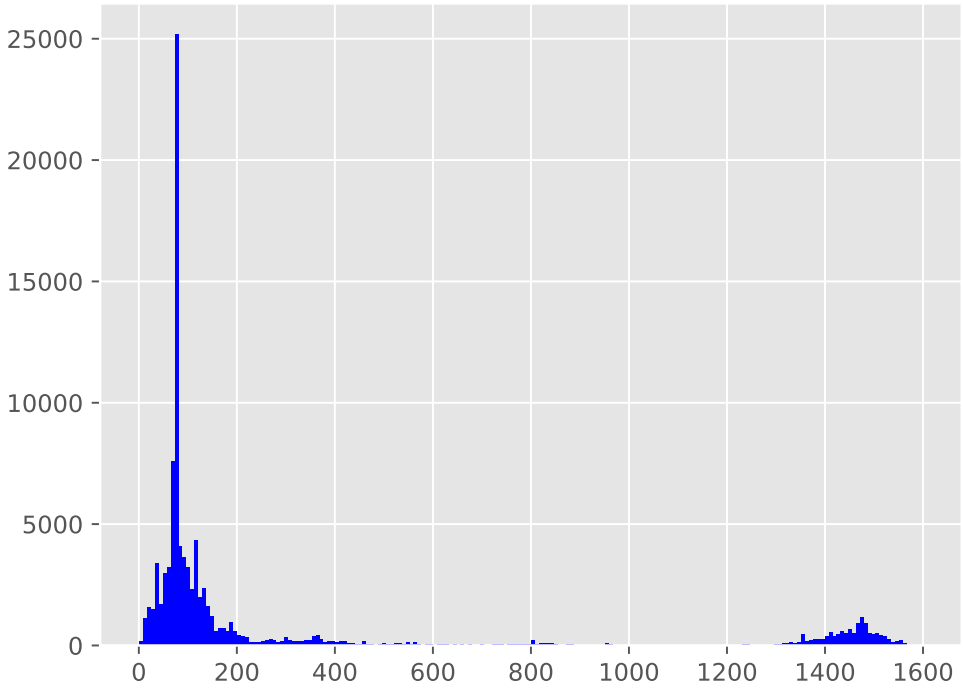


Figure S1: Length distribution of the sequences in the bpRNA-1m dataset Version 1 `http://bprna.cgrb.oregonstate.edu/`. The highest peak correspond to tRNAs of a length of about 75 nucleotides (nts). For the plot the dataset was truncated at length 1600 nts (removing 736 sequences longer than 1600 nts).

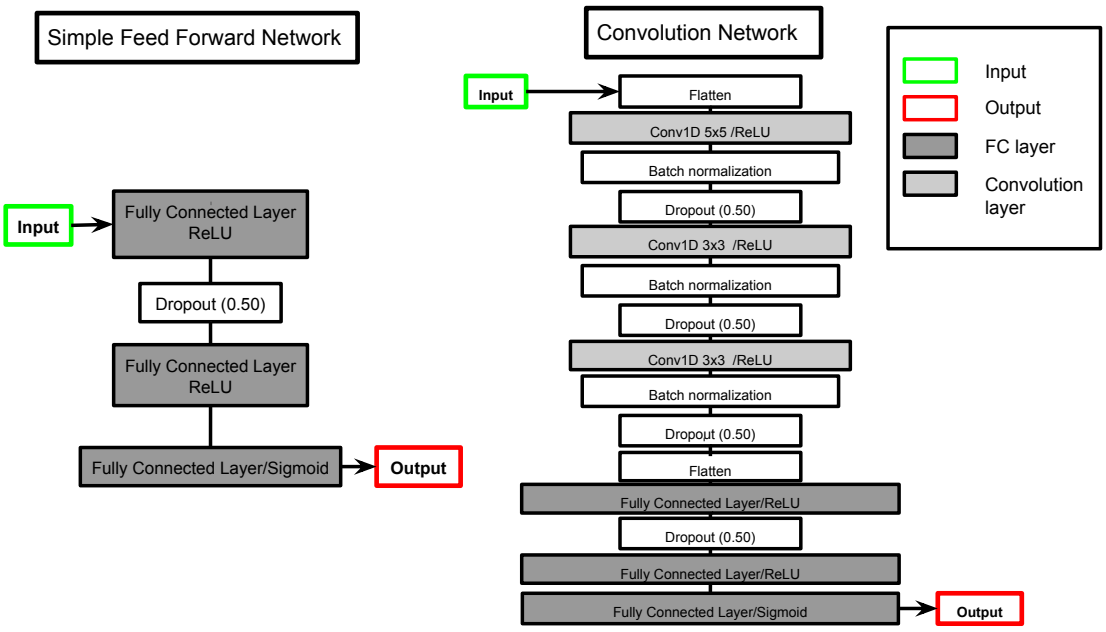
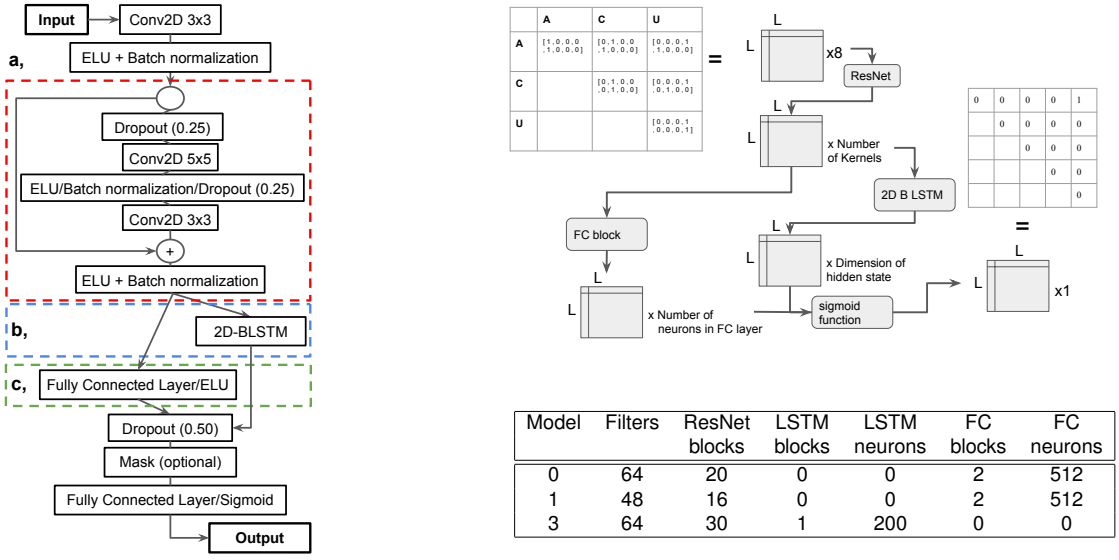


Figure S2: Neural network architectures for the sliding window approach. (left) feed forward neural (FFN) network (right) 1D convolutional neural network (CNN)



Model	Filters	ResNet blocks	LSTM blocks	LSTM neurons	FC blocks	FC neurons
0	64	20	0	0	2	512
1	48	16	0	0	2	512
3	64	30	1	200	0	0

Figure S3: Detailed view of the reimiplemented network architectures of the SPOT-RNA network [1] refered to as model 0-3. The repeat numbers for the ResNet (in red), the LSTM (in blue) and the fully connected layers (in green) are listed in table on the bottom right. How the data is fed into and retreved from the models is shown in the schematic figure on the top right.