-

sincFold: end-to-end learning of short- and long-range interactions for RNA folding

L.A. Bugnon[1]*, L. Di Persia[1], M. Gerard[1], J. Raad[1], S. Prochetto[1,2], E. Fenoy[1], U. Chorostecki[3], F. Ariel[2], G. Stegmayer[1] and D.H. Milone[1]

[1] Research Institute for Signals, Systems and Computational Intelligence sinc(i) (CONICET-UNL), Ciudad Universitaria, Santa Fe, Argentina.
[2] IAL, CONICET, Ciudad Universitaria UNL, (3000) Santa Fe, Argentina.
[3] Faculty of Medicine and Health Sciences, Universitat Internacional de Catalunya, 08195 Sant Cugat del Vallès, Barcelona, Spain.

*To whom correspondence should be addressed.

Abstract

Motivation: Coding and non-coding RNA molecules participate in many important biological processes. Non-coding RNAs fold into well-defined secondary structures to exert their functions. However, the computational prediction of the secondary structure from a raw RNA sequence is a long-standing unsolved problem, which has stagnated in the last decades. Traditional RNA secondary structure prediction algorithms have been mostly based on thermodynamic models and dynamic programming for free energy minimization. More recently deep learning methods have shown competitive performance compared with the classical ones, but still leaving a wide margin for improvement.
Results: In this work we present sincFold, an end-to-end deep learning approach that predicts the nucleotides contact matrix using only the RNA sequence as input. The model is based on hierarchical 1D-2D residual neural networks that can learn short- and long-range interaction patterns. We show that structures can be accurately predicted with minimal physical assumptions. Extensive experiments on well known benchmark datasets were conducted, comparing sincFold against classical methods and recent deep learning models. Results show that sincFold can outperform state-of-the-art methods on all datasets assessed.
Availability: The source code is available at https://github.com/sinc-lab/sincFold/ (v0.13) and a webdemo is provided at https://sinc.unl.edu.ar/web-demo/sincFold
Contact: lbugnon@sinc.unl.edu.ar

1 Introduction

Non-coding ribonucleic acid (ncRNA) molecules have emerged as crucial players in cellular processes, encompassing epigenetics, transcriptional and post-transcriptional regulation, chromosome replication, translation and protein activity and stability [20, 42]. Recent efforts have even explored the clinical potential of ncRNA in diagnostics, vaccines, and therapies [37]. This paradigm shift in our understanding of ncRNA, from being dismissed as "transcriptional noise" prior to the 1980s to being recognized as regulators of gene expression at multiple levels, has generated an explosion of research in this field over the past few decades [7].

The RNA itself consists of an ordered sequence of four basic nucleotides: adenine (A), cytosine (C), guanine (G), and uracil (U). Pairing these bases within an RNA molecule gives rise to its secondary structure, a crucial determinant of its functions and stability [29]. The secondary structure is characterized by hydrogen bonding interactions between complementary base pairs, which typically include the canonical Watson-Crick pairs A-U and C-G [36], along with the wobble pair G-U [34]. Basic stem-loop structures, formed by nested base pairs, are commonly observed. However, the secondary structure can also exhibit complex motifs arising from local bonding and long-range sequence interactions.

Despite the growing number of publicly accessible ncRNA sequences, a significant proportion of their true structures remains unknown [12]. Secondary structures can be inferred from tertiary structures using sophisticated experimental techniques such as X-ray crystallography or nuclear magnetic resonance [11, 16, 30], or they can be obtained using enzymatic probing methods such as nextPARS [8], or chemical probing, such as DMS-seq [39] or SHAPE-seq [9]. However, all these methods suffer from low throughput and high costs [23]. Consequently, due to its cost-effectiveness the computational prediction has gained substantial relevance in biological research and biotechnological applications.

Traditional computational methods for RNA secondary structure prediction employ a thermodynamic model of base-pair interactions optimized through dynamic programming to identify structures with minimal free energy [26, 33, 44]. Despite being proposed 20 years ago [19], these methods, such as RNAstructure [21], ProbKnot [3], RNAfold [17], LinearFold [14], LinearPartition [41], and Ipknot [25], continue to dominate the field. However, their average base-pair prediction performance remains at around $F_1 = 70\%$ [5].

To surpass this performance ceiling, machine learning (ML) techniques, and particularly deep learning (DL), have emerged as

promising alternatives [38]. DL techniques have been widely noticed for structure prediction problems in proteins with AlphaFold [15] and more recently several methods were presented for RNA secondary structure prediction [6, 10, 24, 27, 40]. Compared to classical approaches, DL methods rely less on assumptions about the thermodynamic mechanics of folding, instead adopting a data-driven approach. Consequently, they could be better suited to identify complex structures that defy modeling using traditional techniques. However, a recent systematic evaluation of techniques for comparatively assessing their performance on ncRNAs showed that DL has not yet clearly overperformed classical methods [5].

Currently, several DL approaches are available with different architectural designs, input representations, training data and optimization algorithms for parameter adjustments [43]. Among these proposals, SPOT-RNA [27] was the pioneering DL method based on ensembles of convolutional residual networks (CNNs) and bidirectional Long-short term memory neural networks (LSTM). SPOT-RNA2 [28] improved its predecessor by using predictions from thermodynamic models, evolution-derived sequence profiles and mutational coupling, however requiring multiple sequence alignments. Another hybrid approach was MXfold [1], combining support vector machines and thermodynamic models. Similarly, DMFold [35] and MXFold2 [24] integrated DL techniques with energy based methods. Another method based on both DL and dynamic programming was CDPfold [40], which iteratively computes a matrix representation of possible matchings between bases according to a physical model of base interactions, and then trains a convolutional network over this matrix to predict base pairing probabilities. Upon this, dynamic programming is applied to obtain the final RNA secondary structure.

All of these DL methods have been recently outperformed by Ufold [10]. Ufold approaches the secondary structure prediction problem using a well-known architecture from image segmentation, the U-Net encoder-decoder [22]. It uses a 2D feature map to encode the occurrences of one of the sixteen possible base pairs between nucleotides for each position in the map, including an additional channel with the matrix representation of possible matchings iteratively computed with the algorithm proposed in CDPfold. The predicted output is the contact score map between the bases of the input sequence, which goes through a post-processing step that involves solving a linear programming problem to obtain the optimum contact map. Interestingly, a very recent method, REDfold [6], reported to outperform UFold. This DL method also utilizes a U-Net encoder-decoder network to learn dependencies among the RNA sequence, together with symmetric skip connections to propagate activation information across layers and output post-processing with constrained optimization.

In this work we present sincFold, a novel end-to-end deep learning method for RNA secondary structure prediction. Our approach is based on ResNet bottlenecks to capture both short- and long-range dependencies in the RNA sequence. Unlike other DL models, we adopt a two-stage encoding process: initially, we model sequence encoding in 1D, enabling the learning of small context features and reducing computational costs; then a pairwise encoding in 2D is incorporated to capture distant relationships. Extensive experimental evaluations on two widely used ncRNA databases demonstrate that sincFold outperform classical methods and DL state-of-the-art techniques in terms of F_1 performance. We have made the source code for sincFold freely accessible, facilitating its adoption and further development in the research community[1]. Moreover, a web service to test the trained model is provided[2].

[1] Source code available at https://github.com/sinc-lab/sincFold/ (v0.13)

[2] Webserver available at https://sinc.unl.edu.ar/web-demo/sincFold

2 The sincFold model

In order to obtain a secondary structure prediction from a stand-alone RNA sequence, we propose sincFold. As shown in Figure 1, this novel deep learning architecture is composed of two stages: the first one learns local patterns in 1D encodings, while the second stage learns more distant interactions in 2D. The figure represents in detail the shapes and dimensions of the data along the pipeline (top) and the neural processing blocks (bottom).

The model takes as input a RNA sequence of length L encoded in one-hot (bottom-left) so that each nucleotide type of the sequence is represented with a vector of size 4. The encoded sequence goes through a 1D convolutional layer that performs a first automatic extraction of low-level features for each nucleotide. Then, identity blocks [13] are stacked in a 1D-ResNet. These blocks allow the model to propagate the signal and reduce vanishing gradient issues, while maintaining the same sequence length. Moreover, the identity blocks make the model capable of auto-defining the number of convolutional layers needed during training. Each block is composed of two batch normalization layers, ReLU activations and convolutional layers in 1D, with bottlenecks in the features (depicted with light green in the figure). Bottlenecks reduce the learnable parameters while helping to learn more relevant features.

Following the 1D bottlenecks, the $M \times L$ encoding, where M is the dimension of the feature vector of each nucleotide, is projected to an intermediate prediction ($L \times L$). First, using 1D-convolutional layers two compressed encodings of size $E \times L$ are obtained, where $2E \times L \ll L \times L$. Then, with a matrix transposition and the product between these representations, a first draft of the contact matrix is obtained. A binary mask for admissible canonical interactions is applied to the contact matrix. For example, this mask takes into account that A-A interaction is not possible, and also that two adjacent nucleotides cannot interact. Then, the matrix is forced to be symmetric by adding its transpose.

Once the information is represented in 2D, the new tensor $L \times L$ will go through a 2D-ResNet stage. Similarly to the 1D-ResNet stage, a 2D-convolutional layer is followed by 2D-ResNet blocks composed of batch normalization layers, ReLU activations and 2D convolutions. After several 2D-ResNet layers with bottlenecks the 2D pairwise encodings are flattened to a $L \times L$ output, where the valid nucleotides interactions are enforced again with a binary mask, and its transpose is added to force symmetry. This output matrix is the final 2D prediction of the secondary structure for the RNA sequence, and the entire model can be trained with a unified cost function. A simple post-processing retains only the maximum activation per nucleotide to ensure that each one interacts with at most one nucleotide.

To guide training, we propose a composed loss function

$$\mathcal{L} = \mathcal{L}_\alpha + \lambda_\beta \mathcal{L}_\beta + \lambda_1 \mathcal{L}_1 \tag{1}$$

where $\mathcal{L}_\alpha$ is the cross-entropy loss of the final prediction, $\mathcal{L}_\beta$ is the cross-entropy loss of the 1st 2D prediction prior to the 2D-ResNet block, and $\mathcal{L}_1$ is the L_1 norm of the prediction error, used to enforce the contact matrix sparseness. The weights λ_β and λ_1 of each of these terms are hyperparameters to be adjusted experimentally.

3 Data and performance measures

3.1 Data

In order to evaluate the performance of sincFold, we have run benchmarks with datasets widely used by the community.

RNAstralign dataset [32]: contains 37,149 sequences from 8 large RNA families: 5S rRNAs, Group I Intron, tmRNA, tRNA, 16S rRNA,

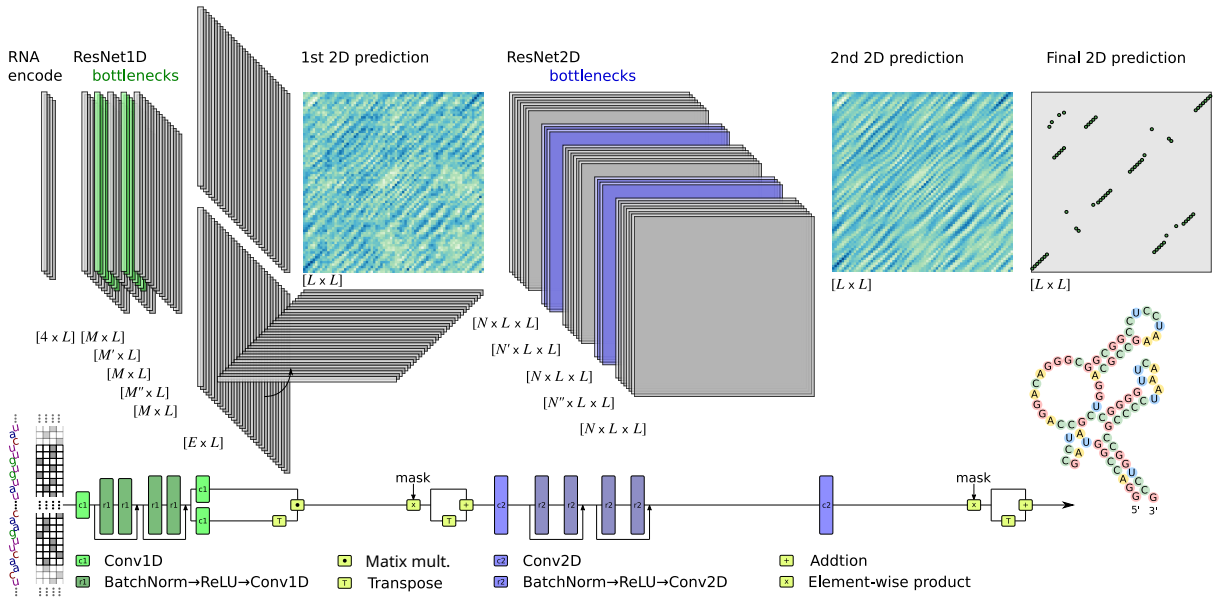

Fig. 1. The end-to-end architecture of sincFold. Top: data flow with its shapes and dimensionality in each point of the architecture, from the $[4 \times L]$ one-hot encoded RNA sequence at the input to the $[L \times L]$ connection matrix at the output. Bottom: neural processing blocks depicted as differentiable layers.

Signal Recognition Particle (SRP) RNA, RNase P RNA and Telomerase RNA. It is one of the most comprehensive RNA structure datasets available.

ArchiveII dataset [29]: the most widely used benchmark dataset for RNA folding methods, containing RNA structures from 9 RNA families: 5S rRNAs, SRP RNA, tRNA, tmRNA, RNase P RNA, Group I Intron, 16S rRNA, Telomerase RNA and 23S rRNA. The total number of sequences is 3,975.

Ablation dataset: in addition, we compiled a dataset of sequences derived from the URS server [2] to be used as a small independent dataset for model optimization. These sequences and secondary structures were extracted from the Protein Data Bank (PDB), consisting of 753 sequences ranging from 8 to 456 nucleotides.

As suggested in [31], sequences longer than 512 nucleotides were filtered to limit the runtime of experiments, leaving 22,611 sequences in the RNAstralign dataset and 3,864 sequences in the ArchiveII dataset. Group I intron RNAs were excluded from the RNAstralign dataset because it had sequences without a unique structure. To assess the performance, we perform k-fold cross-validation with $k = 5$. For the ArchiveII dataset, the original k-fold split provided by the authors was used [29]. For the other two datasets, sequences were randomly divided into 5 independent folds of approximately the same size, and each fold was in turn taken as the test data while the remaining folds were taken as the training data.

3.2 Performance measures

The focus of performance measures is on the predicted base pairs in comparison to a reference structure [18]. Pairs that are both in the prediction and in the reference structure are true positives (TP), while pairs predicted but not in the true structure are false positives (FP). Similarly, a pair in the reference structure that is not predicted is a false negative (FN), and a pair that is neither predicted nor in the true structure is a true negative (TN). To fully characterize the successes and failures of structure prediction, we use the F_1 score, defined as

$$F_1 = \frac{2TP}{2TP + FP + FN} \tag{2}$$

This measure was calculated with tolerance of 1 nucleotide, as in the standard scorer of the RNAstructure package [3].

4 Results

4.1 Ablation study and hyperparameters exploration

We conducted an ablation study to gain a deeper understanding on the contribution of each of the components of the sincFold architecture. We run several versions of the sincFold model: C1D) a baseline model with only 1D-convolutional networks; R1D) the same model replacing convolutions with 1D-residual blocks and bottlenecks; C1D+C2D) the model with the 2D-stage using only convolutional neural networks; C1D+R2D) replacing convolutions with the 2D-residual blocks in the 2D stage; and R1D+R2D) with residual blocks and bottlenecks in both stages. The F_1 scores for each ablated sincFold version, from a 5-fold cross-validation on the Ablation dataset, are shown in the boxplots of Figure 2.

It can be seen that changing the C1D to a R1D block slightly improves the median results, from a median $F_1 = 0.726$ to $F_1 = 0.743$. It can be observed that adding the 2D stage (C2D) to the output of the previous models increased their performance significantly, by 10% for each model. The F_1 raises up to 0.820 with C1D+C2D, and using a ResNet instead of a CNN in the 2D stage (C1D+R2D), performance further improves up to $F_1 = 0.830$. Finally, results are even further improved in the model with ResNet blocks in both stages (R1D+R2D), reaching $F_1 = 0.843$.

After the ablation study we conclude that ResNet blocks effectively improve the generalization capability, in comparison to simple convolutional layers. Figure 3 presents a detailed analysis of the true positive rate of predictions in this dataset along the interconnection distance. Interestingly, when the 2D stage (green) is added to the 1D stage (blue) the model performance improves for all connections, but especially for distances longer than 200 nt. This shows that in fact the sincFold 2D stage improves the learning of long range dependencies. Moreover, a very interesting property of ResNet blocks is that when there are many blocks available, the model is capable of automatically selecting how

[3] http://rna.urmc.rochester.edu/Releases/

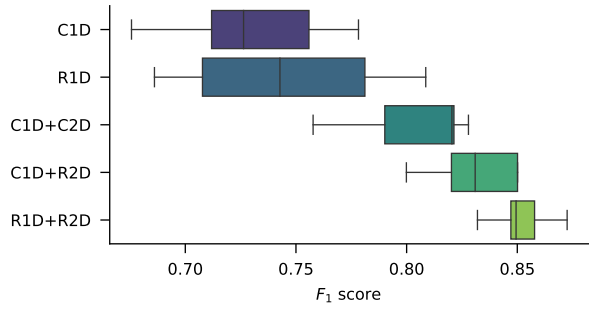

Fig. 2. Ablation study on each of the components of the sincFold architecture. Each box has the F_1 scores from a 5-fold cross-validation on the Ablation dataset.

many of them are really necessary, skipping the non-necessary blocks during training. This reduces the learnable parameters while helping to learn more relevant features.

Using the best performing sincFold architecture (R1D+R2D), we performed a hyperparameter space search in the Ablation dataset, exploring: batch size, learning rate, the use of learning rate schedule, weights for the loss components λ_β, λ_1, architecture of the 1D-ResNet stage (kernel size and dilation, number of filters and number of layers) and the 2D-ResNet stage (kernel size, number of filters, bottleneck size and number of layers). Parameters were explored randomly [4], and the best configuration was selected for the next experiments (Sup. Mat 1).

4.2 sincFold can outperform other classical and DL methods on standard datasets

Figure 4 shows the comparative results among classical folding methods (RNAfold, RNAstructure, ProbKnot, IPKnot, LinearPartition-V and LinearFold-V), hybrid methods (LinearPartition-C and LinearFold-C), DL based methods (MXfold2, UFold and REDfold) and the proposed sincFold, in terms of F_1 for 5-fold cross-validation on RNAstralign and ArchiveII datasets. All DL methods were trained and evaluated from scratch with the same dataset partitions on cross-validation. For the RNAstralign dataset (Fig. 4 left) it can be seen that all classical methods have a performance between 0.65 and 0.71 of F_1. Methods with learnable parameters like LinearPartition-C and LinearFold-C can improve performance up to $F_1 = 0.73$. The DL methods, instead, have significantly better performance, except MXFold2 which is closer to classical methods (median $F_1 = 0.74$). UFold reaches a median $F_1 = 0.971$ and REDfold arrives at median $F_1 = 0.980$. The proposed method, sincFold, achieves $F_1 = 0.985$. The variance of our method is very small, and the box is not overlapped with the performance of the other DL methods. For the ArchiveII dataset (Fig. 4 right), it can be seen that all classical methods and MXfold2 have a median performance below $F_1 = 0.65$. In this case UFold has a median $F_1 = 0.876$ and REDfold arrives at $F_1 = 0.848$. SincFold achieves the highest median $F_1 = 0.911$, with statistical significant difference in comparison to all other methods. Thus, it can be stated that in these datasets our proposal achieves a significantly better performance than state-of-the-art DL methods.

These results clearly show how cutting edge DL models can be more effective than classical ones for RNA secondary structure predictions in benchmark datasets. Among them, the sincFold has the specific capability of better capturing interactions in a wide range of distances between elements of the sequence. Moreover, thanks to the ResNet blocks and the bottlenecks layers, sincFold has proven to be better suited to identify complex structures that defy traditional modeling, without significantly increasing the effective number of trainable parameters (Sup. Mat 3). Therefore, a larger generalization capability is provided, useful for example in the case of RNA families with particular structures or for longer sequences (these aspects will be analyzed in detail in the next subsection). In summary, results show that our method has significantly outperformed state-of-the-art methods for RNA secondary structure prediction, in both the most cited and used benchmark datasets for this task. The proposed sincFold is an end-to-end model that can accurately predict the secondary structure from a RNA sequence without requiring multi-sequence alignments, or any other pre-processing of the input sequences.

4.3 sincFold has the best performance analyzed across RNA families and sequence lengths

To have more insights on the sincFold performance, in this section we report comparative results of our model regarding specific aspects: RNA families, test-train similarity and sequence length. For a detailed description of ArchiveII sequences see Sup. Mat 2. Figure 5 shows the detailed performance of the compared methods for each RNA family. Only the best classical methods according to the previous comparisons are shown here (RNAfold and LinearPartition-C), and the best DL-based methods (REDfold and UFold). These per-family results are ordered from the least populated families (left) to the most populated families (right). Light blue bars indicate the relative frequency of each family in the dataset. It can be seen that for the RNA family with the lowest number of samples (23S rRNA, on the left), the classical models can predict structures with relatively high performance (F_1 between 0.70 and 0.80), while DL models have lower performance (F_1 between 0.40 and 0.65). This family has only 15 sequences, that is, just 10 samples to train in each fold, thus DL models cannot be adequately trained to generalize in test sequences. In the case of Telomerase RNA family, which has twice the number of examples than the smallest one, classical methods have a performance between $F_1 = 0.50$ and $F_1 = 0.60$, DL methods have also a similar moderate performance, while our sincFold model achieves the highest $F_1 = 0.70$. In the case of 16S rRNA family, all methods are approximately equally performant with F_1 between 0.50 and 0.70. For Group I Intron family, despite having only 50 sequences in a training set, sincFold is beginning to clearly differentiate from its DL competitors, and it is even superior to classical methods. It can be seen for tmRNA the largest difference between DL and classical methods, showing that indeed their structures are hard to fold, possibly explained by the presence of pseudoknots. Notably, although there are still relatively few samples to train RNase P RNA and tmRNA families (around 340 sequences per fold), sincFold is clearly superior to the other DL-based methods. For the last 3 families, the most populated ones, all DL-based methods are superior to classical methods. In fact, as more examples are available, the within-family performance of these models increases as can be expected.

The performance for each method according to test-train structural distance is shown in Figure 6. The distance between two structures was computed using RNAdistance from the ViennaRNA package [17]. For each test sequence, the test-train distance is defined as the minimum structural distance between this sample and all the cases in its corresponding training fold. The x-axis has ranges of structural distances, from large distances between test-train sequences (left) down to low distances between test and train sequences (right). Light blue bars indicate the proportion of test sequences in each bin of structural distances. It can be clearly seen that as the test-train distance diminishes, all methods (including the classical, non-learnable ones) improve performance. This also makes evident that structures on the left side are really harder to predict, even for classical models that are not trained and thus they are agnostic to the test-train structural distances. For the 23 structures in the first 2 bins of distances all methods have median $F_1 < 0.50$. It can be seen that for distances between 0.40 and 0.25, both classical and DL methods have again a low performance $F_1 \in (0.40, 0.60)$. In the middle cases, from 0.25 to 0.20

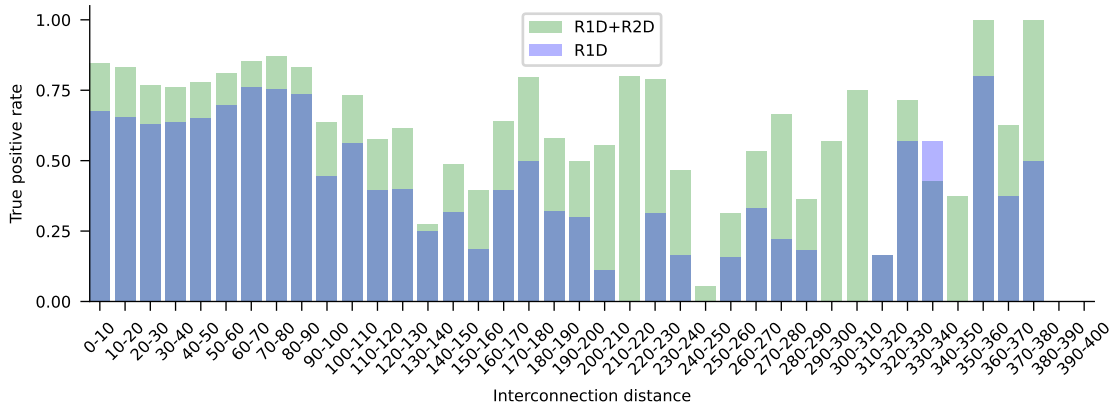

Fig. 3. True positive rate for each range of interaction distance, comparing the model with only the first stage (R1D) and both stages (R1D+R2D).

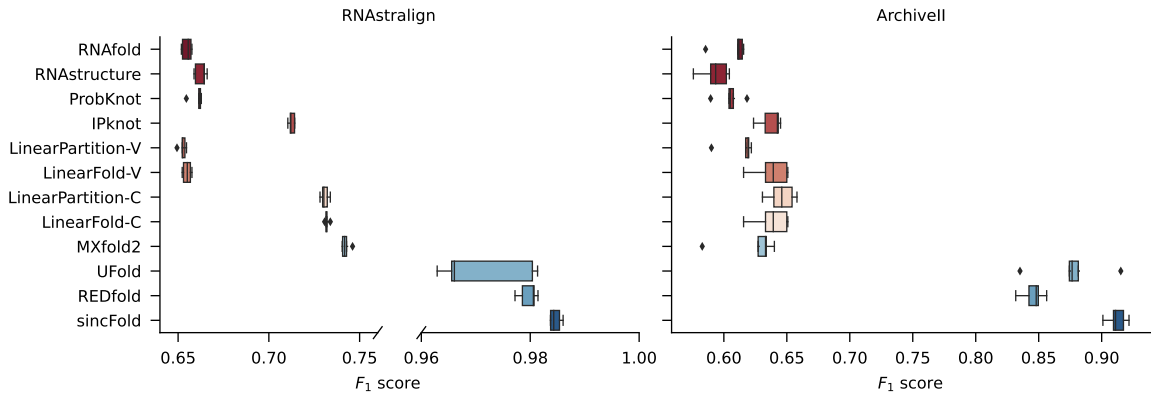

Fig. 4. Comparative results among classical folding methods, DL-based folding methods and sincFold, for a 5-fold cross-validation on the RNAstralign dataset (left) and the ArchiveII dataset (right). Horizontal scale was adjusted to improve visualization.

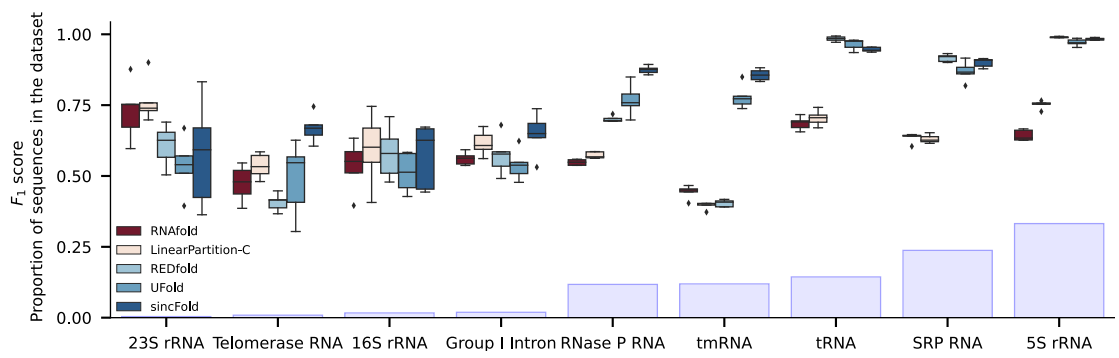

Fig. 5. Per-family performance of compared methods, from less populated families (left) to more populated families (right). Light blue bars indicate the relative frequency of each family in the dataset.

distance, DL methods are slightly better than classical ones. Finally, for the lowest test-train structural distances (< 0.15), DL methods are clearly better for RNA secondary structure prediction, being sincFold the best method in all cases. These trends can be explained by two facts. First, the abundance of structure samples benefits DL models more than the classical ones because the former have more cases to learn from. Besides, the benefits for the classical methods are indirect since they do not learn, but were developed looking at the most abundant or popular structures that therefore better fit the thermodynamic models. Secondly, based on the advantage of structure abundance for data-driven approaches, sincFold is the one that best takes advantage of the ability to adapt and to learn new structures, regardless of how much is known about the thermodynamics of the molecules. Thanks to its capability of learning by examples, structures can be effectively predicted and with very high performance ($F1 > 0.90$), while classical modeling can only guarantee at most a $F_1 = 0.60$.

Figure 7 shows the detailed performance for each method according to the lengths of the test sequences, from shorter (left) to longer sequences (right). Light blue bars indicate the proportion of each bin of lengths in the dataset. Here it can be seen that for shorter sequences, all methods have average performance above $F_1 = 0.70$, being particularly very good at this task all DL methods with an impressive $F_1 > 0.95$. As sequence length is increased, classical methods lower their performance up to a median $F_1 < 0.60$. However, DL methods are less affected by the increase in sequence length, maintaining $F_1 > 0.75$ in most cases and being sincFold always the best for long sequences, achieving $F_1 = 0.86$ for sequences between 300 and 400 nucleotides. At the extreme, for the sequences longer than 400 nucleotides, while UFold and

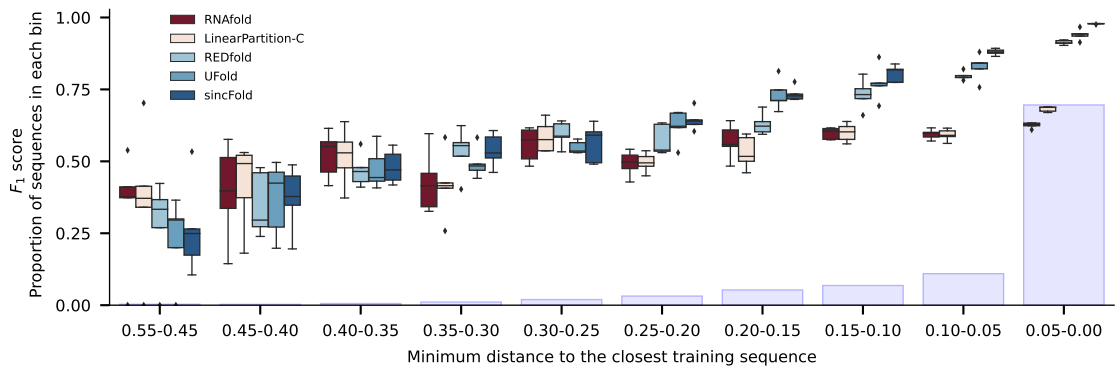

Fig. 6. Performance according to test-train structural distance, from large distances (left) to low distances (right). Bars indicate the proportion of each bin in the dataset.

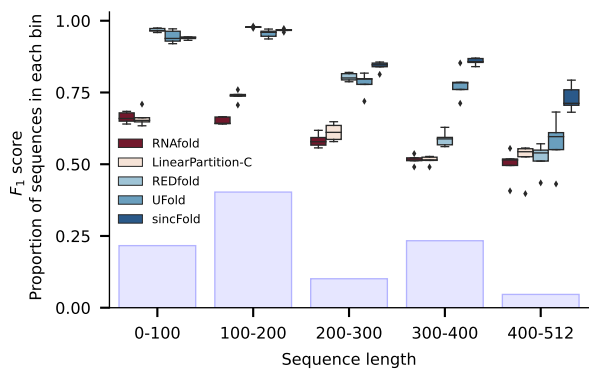

Fig. 7. Performance for each method according to the mean lengths of the sequences, from shorter (left) to longer (right) sequences.

REDfold lower performance up to similar figures than classical methods, sincFold is always higher than them and even despite the few examples available to learn from, achieving a median $F_1 = 0.73$, which is superior to the average performance of classical method in the shortest sequences. SincFold, thanks to its capability for capturing different range of interactions with the combination of 1D and 2D stages, is significantly better than all other methods for the secondary structure prediction of longer ncRNA sequences.

5 Conclusions

In this work, we presented sincFold, an end-to-end deep learning model for RNA secondary structure prediction. We show that local and distant relationships can be encoded effectively using a hierarchical 1D-2D ResNet architecture, improving the state-of-the-art in RNA secondary structure prediction. The sincFold model is based on ResNet blocks, bottlenecks layers and a 1D-to-2D projection. It has proven to be better suited to identify structures that might defy traditional modeling. That is, thanks to its capability of learning by examples, in many cases where classical methods have a performance around 60%, sincFold can achieve 90% of $F1$. The per-family performance analysis has shown that even when there is a small number of examples to learn from (around hundreds of sequences), sincFold can learn the structures with high performance. As more examples are available, the within-family generalization capability of the model is significantly better than other methods. Results also show that sincFold, thanks to its capability for capturing a wide range distances in interactions, is significantly better than all other methods for the secondary structure prediction in longer ncRNA sequences. In summary, the comprehensive experiments performed in this study demonstrate that sincFold achieves the best performance compared to well established folding algorithms and state-of-the-art deep-learning based methods, even for ncRNAs longer than 200 nucleotides.

Funding

We would like to thank the AWS Cloud Credit for Research, the Ministerio de Producción, Ciencia y Tecnología, Santa Fe (PEICID-2022-075) and the Agencia Nacional de Promoción de la Investigación, el Desarrollo Tecnológico y la Innovación (Ciencia y tecnología contra el hambre) for their support for this work. This work was supported by ANPCyT (PICT 2018 3384) and UNL (CAI+D 2020 115).
Conflict of Interest: none to be declared.

References

[1]Akiyama, M., Sato, K., and Sakakibara, Y. (2018). A max-margin training of RNA secondary structure prediction integrated with the thermodynamic model. *Journal of Bioinformatics and Computational Biology*, **16**(06), 1840025.

[2]Baulin, E., Yacovlev, V., Khachko, D., Spirin, S., and Roytberg, M. (2016). URS DataBase: universe of RNA structures and their motifs. *Database*, **2016**, baw085.

[3]Bellaousov, S. and Mathews, D. H. (2010). ProbKnot: Fast prediction of RNA secondary structure including pseudoknots. *RNA*, **16**(10), 1870–1880.

[4]Bergstra, J. and Bengio, Y. (2012). Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, **13**(10), 281–305.

[5]Bugnon, L. A., Edera, A. A., Prochetto, S., Gerard, M., Raad, J., Fenoy, E., Rubiolo, M., Chorostecki, U., Gabaldón, T., Ariel, F., Persia, L. E. D., Milone, D. H., and Stegmayer, G. (2022). Secondary structure prediction of long noncoding RNA: review and experimental comparison of existing approaches. *Briefings in Bioinformatics*, **23**(4).

[6]Chen, C.-C. and Chan, Y.-M. (2023). REDfold: accurate RNA secondary structure prediction using residual encoder-decoder network. *BMC Bioinformatics*, **24**(1).

[7]Chen, X. and Huang, L. (2022). Computational model for ncRNA research. *Briefings in Bioinformatics*, **23**(6).

[8]Chorostecki, U., Willis, J. R., Saus, E., and Gabaldon, T. (2021). *Profiling of RNA Structure at Single-Nucleotide Resolution Using nextPARS*, pages 51–62. Springer US, New York, NY.

[9]D, L., KE, W., AH, S., and JB, L. (2014). Shape-seq 2.0: systematic optimization and extension of high-throughput chemical probing of rna secondary structure with next generation sequencing. *Nucleic Acids*, **42**(21).

[10]Fu, L., Cao, Y., Wu, J., Peng, Q., Nie, Q., and Xie, X. (2022). UFold: Fast and accurate RNA secondary structure prediction with deep learning. *Nucleic Acids Research*, **50**(3), e14.

[11]Fürtig, B., Richter, C., Wöhnert, J., and Schwalbe, H. (2003). NMR spectroscopy of RNA. *ChemBioChem*, **4**(10), 936–962.

[12]Gao, W., Yang, A., and Rivas, E. (2023). Thirteen dubious ways to detect conserved structural rnas. *IUBMB Life*, **75**(6), 471–492.

[13]He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[14]Huang, L., Zhang, H., Deng, D., Zhao, K., Liu, K., Hendrix, D. A., and Mathews, D. H. (2019). LinearFold: linear-time approximate RNA folding by 5'-to-3' dynamic programming and beam search. *Bioinformatics*, **35**(14), i295–i304.

[15]Jumper, J., Evans, R., Pritzel, A., Green, T., Figurnov, M., Ronneberger, O., Tunyasuvunakool, K., Bates, R., Žídek, A., Potapenko, A., Bridgland, A., Meyer, C., Kohl, S. A. A., Ballard, A. J., Cowie, A., Romera-Paredes, B., Nikolov, S., Jain, R., Adler, J., Back, T., Petersen, S., Reiman, D., Clancy, E., Zielinski, M., Steinegger, M., Pacholska, M., Berghammer, T., Bodenstein, S., Silver, D., Vinyals, O., Senior, A. W., Kavukcuoglu, K., Kohli, P., and Hassabis, D. (2021). Highly accurate protein structure prediction with AlphaFold. *Nature*, **596**(7873), 583–589.

[16]Keel, A. Y., Rambo, R. P., Batey, R. T., and Kieft, J. S. (2007). A general strategy to solve the phase problem in RNA crystallography. *Structure*, **15**(7), 761–772.

[17]Lorenz, R., Bernhart, S. H., zu Siederdissen, C. H., Tafer, H., Flamm, C., Stadler, P. F., and Hofacker, I. L. (2011). ViennaRNA package 2.0. *Algorithms for Molecular Biology*, **6**(1), 1–10.

[18]Mathews, D. H. (2019). How to benchmark RNA secondary structure prediction accuracy. *Methods*, **162-163**, 60–67.

[19]Mathews, D. H., Sabina, J., Zuker, M., and Turner, D. H. (1999). Expanded sequence dependence of thermodynamic parameters improves prediction of RNA secondary structure. *Journal of Molecular Biology*, **288**(5), 911–940.

[20]Mattick, J. S., Amaral, P. P., Carninci, P., Carpenter, S., Chang, H. Y., Chen, L.-L., Chen, R., Dean, C., Dinger, M. E., Fitzgerald, K. A., Gingeras, T. R., Guttman, M., Hirose, T., Huarte, M., Johnson, R., Kanduri, C., Kapranov, P., Lawrence, J. B., Lee, J. T., Mendell, J. T., Mercer, T. R., Moore, K. J., Nakagawa, S., Rinn, J. L., Spector, D. L., Ulitsky, I., Wan, Y., Wilusz, J. E., and Wu, M. (2023). Long non-coding RNAs: definitions, functions, challenges and recommendations. *Nature Reviews Molecular Cell Biology*, **24**(6), 430–447.

[21]Reuter, J. S. and Mathews, D. H. (2010). RNAstructure: software for RNA secondary structure prediction and analysis. *BMC Bioinformatics*, **11**(1), 1–10.

[22]Ronneberger, O., Fischer, P., and Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In *Lecture Notes in Computer Science*, pages 234–241. Springer International Publishing.

[23]Ross, C. J. and Ulitsky, I. (2022). Discovering functional motifs in long noncoding RNAs. *WIREs RNA*.

[24]Sato, K., Akiyama, M., and Sakakibara, Y. (2021). RNA secondary structure prediction using deep learning with thermodynamic integration. *Nature Communications*, **12**(1), 1–10.

[25]Sato, K., Akiyama, M., and Sakakibara, Y. (2022). RNA secondary structure prediction using deep learning with thermodynamic integration. *Nature Communications*, **12**(1), 1–10.

[26]Schroeder, S. J. and Turner, D. H. (2009). Optical melting measurements of nucleic acid thermodynamics. In *Methods in Enzymology*, pages 371–387. Elsevier.

[27]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. *Nature Communications*, **10**(1).

[28]Singh, J., Paliwal, K., Zhang, T., Singh, J., Litfin, T., and Zhou, Y. (2021). Improved RNA secondary structure and tertiary base-pairing prediction using evolutionary profile, mutational coupling and two-dimensional transfer learning. *Bioinformatics*, **37**(17), 2589–2600.

[29]Sloma, M. F. and Mathews, D. H. (2016). Exact calculation of loop formation probability identifies folding motifs in RNA secondary structures. *RNA*, **22**(12), 1808–1818.

[30]Spokoini-Stern, R., Stamov, D., Jessel, H., Aharoni, L., Haschke, H., Giron, J., Unger, R., Segal, E., Abu-Horowitz, A., and Bachelet, I. (2020). Visualizing the structure and motion of the long noncoding rna hotair. *RNA*, **26**(5), 629–636.

[31]Szikszai, M., Wise, M., Datta, A., Ward, M., and Mathews, D. H. (2022). Deep learning models for RNA secondary structure prediction (probably) do not generalize across families. *Bioinformatics*, **38**(16), 3892–3899.

[32]Tan, Z., Fu, Y., Sharma, G., and Mathews, D. H. (2017). TurboFold II: RNA structural alignment and secondary structure prediction informed by multiple homologs. *Nucleic Acids Research*, **45**(20), 11570–11581.

[33]Turner, D. H. and Mathews, D. H. (2009). NNDB: the nearest neighbor parameter database for predicting stability of nucleic acid secondary structure. *Nucleic Acids Research*, **38**(suppl_1), D280–D282.

[34]Varani, G. and McClain, W. H. (2000). The g·u wobble base pair. *EMBO reports*, **1**(1), 18–23.

[35]Wang, L., Liu, Y., Zhong, X., Liu, H., Lu, C., Li, C., and Zhang, H. (2019). DMfold: A novel method to predict RNA secondary structure with pseudoknots based on deep learning and improved base pair maximization principle. *Frontiers in Genetics*, **10**, 1–10.

[36]WATSON, J. D. and CRICK, F. H. C. (1953). Molecular structure of nucleic acids: A structure for deoxyribose nucleic acid. *Nature*, **171**(4356), 737–738.

[37]Winkle, M., El-Daly, S. M., Fabbri, M., and Calin, G. A. (2021). Noncoding RNA therapeutics — challenges and potential solutions. *Nature Reviews Drug Discovery*, **20**(8), 629–651.

[38]Wu, K. E., Zou, J. Y., and Chang, H. (2023). Machine learning modeling of RNA structures: methods, challenges and future perspectives. *Briefings in Bioinformatics*.

[39]Y, D., Y, T., CK, K., Y, Z., PC, B., and SM, A. (2014). In vivo genome-wide profiling of rna secondary structure reveals novel regulatory features. *Nature*, **505**, 696–700.

[40]Zhang, H., Zhang, C., Li, Z., Li, C., Wei, X., Zhang, B., and Liu, Y. (2019a). A new method of RNA secondary structure prediction based on convolutional neural network and dynamic programming. *Frontiers in Genetics*, **10**, 1–10.

[41]Zhang, H., Zhang, L., Mathews, D. H., and Huang, L. (2020). LinearPartition: linear-time approximation of RNA folding partition function and base-pairing probabilities. *Bioinformatics*, **36**(Supplement_1), i258–i267.

[42]Zhang, P., Wu, W., Chen, Q., and Chen, M. (2019b). Non-coding RNAs and their integrated networks. *Journal of Integrative Bioinformatics*, **16**(3).

[43]Zhao, Q., Zhao, Z., Fan, X., Yuan, Z., Mao, Q., and Yao, Y. (2021). Review of machine learning methods for RNA secondary structure prediction. *PLOS Computational Biology*, **17**(8), e1009291.

[44]Zuker, M. and Stiegler, P. (1981). Optimal computer folding of large RNA sequences using thermodynamics and auxiliary information. *Nucleic Acids Research*, **9**(1), 133–148.