rpcFold: residual parallel convolutional neural network to decipher RNA folding from RNA sequence

Nandita Sharma[1,*] , Pralay Mitra[2]

[1]Centre for Computational and Data Sciences, Indian Institute of Technology Kharagpur, Kharagpur, West Bengal 721302, India

[2]Department of Computer Science and Engineering, Indian Institute of Technology Kharagpur, Kharagpur, West Bengal 721302, India

*Correspondence to
Nandita Sharma,
Centre for Computational and Data Sciences,
Indian Institute of Technology Kharagpur
West Bengal - 721302, India
Email: nandita.sharma9494@kgpian.iitkgp.ac.in
Phone: +91-3222-282344
ORCID ID: 0000-0003-2191-8462

Abstract

The complex and versatile structures of the RNA molecules allow them to exhibit a wide range of functionalities in a cell. Precise secondary structure information offers deeper insights into the functionality of many RNA molecules. Earlier approaches primarily relying on the concept of free-energy minimization proved inadequate, as RNA often adopts complex folds that do not adhere strictly to free energy rules, especially with increasing length. Subsequently, learning-based models were introduced, but they struggled with handling cross-family data and tended to avoid longer sequences to mitigate computational overhead and the risk of overfitting or underfitting due to limited data availability. Here, we propose a residual parallel convolutional neural network-based RNA secondary structure prediction model, called rpcFold, which can effectively accommodate RNAs of varying lengths by integrating the sliding window method. Further, we consider all the 16 possible base pairs, including canonical and noncanonical pairs, and compute the base-pairing possibility score for each nucleotide in the sequence using a locally weighted Gaussian function. This helps to gain intricate base pairing information and better understand the long- and short-range interactions. We pass these input features as an image-like representation to our model rpcFold. The rpcFold outperforms the existing works and shows a significant improvement of 15% in the F1-score over the previous best score on the benchmark cross-family dataset. We observe a 7% and 5% enhancement in F1-score and accuracy respectively over the last best score, for within family dataset. Our precise prediction of RNA structures will help in revealing the roles of functional RNAs and advancing the development of biomedical applications based on RNA.

Keywords: Deep Learning method, RNA secondary structure prediction, residual parallel convolutional neural network, sliding window method

Introduction

RNA molecules play crucial roles in diverse cellular processes such as genetic information transmission, transcriptional and post-transcriptional gene expression, and gene regulation[1]. Accurate structure information provides enhanced insights into the functionality of RNA. However, determining RNA structures is challenging because of their inherent flexibility and versatile structures[2][3]. Both secondary and tertiary structure of RNA holds the homology and evolutionary information, and contribute to functional analysis[4]. The tertiary structure of RNA is highly influenced by temperature and environmental factors, leading to poor stability[5]. Since RNA secondary structures are less complex and more stable than their corresponding tertiary structures, and contribute equally to functional predictions, researchers tend to focus more on RNA secondary structures. Also, the secondary structure is evolutionary more conserved than the primary sequence and is crucial in determining the function of many RNA molecules[6]. Experimentally, RNA structures are determined by techniques like X-ray crystallography, Nuclear magnetic resonance (NMR), cryo-electron microscopy, and chemical probing. These experimental techniques are expensive and have protracted experiment periods with low throughput[7]. To overcome this limitation, computational approaches are used as an alternative to predict the secondary structure of the RNA[8] [9].

Accurate computational prediction of RNA secondary structure is a long-standing problem in bioinformatics. To address this problem, many methods have been proposed, which are either based on (i) comparative sequence analysis[10] and evolutionary profile-based [4], [11]; or (ii) single sequence-based prediction [9][12][13]. The operational concept of the models in the first category closely resembles homology modeling, wherein the sequences exhibit a high degree of

similarity. Here the conserved regions of base pairs among homologous sequences are identified through the utilization of pairwise and multiple sequence alignment techniques. Here we can attain optimal outcomes if an adequate amount of homologous sequence data is available and those are aligned properly.

Some notable works like Pfold[14], RNAalifold[15], PETfold[16], CentroidAlifold[17], and ConsAlifold[10] utilized aligned RNA sequences for their predictions, whereas SPOT-RNA2[11] and GCNfold[4] used single sequences along with the evolutionary and structural information as features. Since the pool of available RNA sequences remains limited and new RNA families continue to be discovered, relying solely on alignment strategies may not yield satisfactory prediction accuracy. The second category, which is widely employed, involves predicting RNA secondary structures by folding a single sequence using a scoring function as a basis. In the second category, existing works like Mfold[18], CONTRAfold[19], RNAstructure[20], and mxfold[21] are based on this minimum free energy (MFE) concept efficiently implemented through dynamic programming. To reduce the time complexity of dynamic programming-based methods, Linearfold[22] and LinearPartition[23] have proposed an approximate MFE structure in linear time. For the short-range RNAs, the MFE state is applicable. However, the long-range RNAs do not have a stable MFE folding state rather they will be stable at a suboptimal energy state. Due to these limitations, this list of diverse methods reaches a plateau and cannot significantly improve the prediction accuracy after reaching the ceiling of about 80%.

People also tried to improve the accuracy of molecular structure prediction with the use of emerging learning techniques like machine learning and deep learning methods. Some notable

works in RNA structure prediction have already used existing deep-learning models to achieve better accuracy[8][24]. The learning-based models can overcome the shortcomings of thermodynamic-based models, as learning-based models do not require prior assumptions, consider both canonical and noncanonical interactions, can handle tertiary interactions and pseudoknots, and are able to predict previously unrecognized base pairs. Existing methods like SPOT-RNA[13], CDPfold[12], E2Efold[25], Ufold[9], and GCNfold[4] use different learning models like transfer learning, fully convolutional neural networks, and graph convolutional networks to efficiently train the structure prediction model. Though their model shows good prediction accuracy for within-family datasets but shows very poor performance against cross-family datasets. Only very few RNA sequences have already been discovered and the majority is yet to be discovered. The prediction model should show fair performance against unseen data. Another shortcoming of the existing models is that they only consider RNA sequences up to a certain length, in most cases, it is up to 600 long nucleotides. The longer RNA they simply omit or cut from any particular length and then fit into the model. Cutting RNA from any random length may lead to data loss as the length of RNA can significantly vary up to thousands of nucleotides. Further, due to the limited availability of data for RNA compared to proteins, existing large deep learning architectures may encounter challenges such as overfitting and vanishing gradient problems, leading to unsatisfactory outcomes.

To overcome these limitations, we propose a more generalized and simple deep learning model for predicting RNA secondary structure from a given RNA single sequence. First, we transform the RNA sequence into a two-dimensional (2D) image representation comprising of 16 channels, each corresponding to 16 distinct base-pairing possibilities, encompassing both canonical and

non-canonical base pairing, as well as long-range interactions. Also, we add base-paring possibility between each nucleotide using a defined Gaussian function based on the locally weighted pairing rules. Further, we efficiently utilized the concepts of residual networks and parallel convolutional blocks, and proposed our generalized residual parallel convolutional neural network-based model rpcFold to give accurate prediction results for both within-family and cross-family RNA datasets. Additionally, we incorporate the sliding window method to efficiently handle the long RNA sequences without escalating the computational burden. As a result, we achieved a remarkable 15% improvement over the previous best score on cross family dataset and a 5% improvement over the best score on within family dataset. Our rpcFold model is straightforward and well-suited for managing diverse datasets, resulting in shorter inference time and demonstrating enhanced model robustness. For the wider use, we have made our rpcFold available as a free web service (https://cosmos.iitkgp.ac.in/rpcFold/).

Materials and methods

Dataset preparation

In our study we have used three well-known datasets, ArchiveII[26], RNAStralign[27], and bpRNA-1m[28]. ArchiveII contains 3975 RNA sequences from 10 different families where the length of RNA sequence varies from 28 nucleotides(nt) to 2968 nucleotides(nt)[26]. The RNAStralign contains 37,149 RNA sequences from 8 different families and the RNA sequence length ranges from 30nt to 1829nt[27]. The bpRNA-1m dataset, on the other hand, contains 102318 RNA sequences from 2588 different families, ranging between 1nt and 4381nt[28].

First, we remove all the duplicate instances and incomplete cases from our database. The data containing any nucleotide except standard adenine(A) / uracil(U) / guanine(G) / cytosine(C) is considered redundant and comes under incomplete cases. The concept of pseudoknot is mainly related to RNA tertiary structure and hence is omitted from the secondary structure calculation due to the lack of data availability. Further, we have excluded certain sequences if they do not form any structure (remain single-stranded), or if they are part of stems or loops, sections of an RNA structure, or peptide chains, particularly those sequences shorter than 20 nucleotides. As a result of applying all the aforementioned constraints, we will finally have 83173 RNA sequences, representing 58% of the initial dataset size, available for the training and testing of our model.

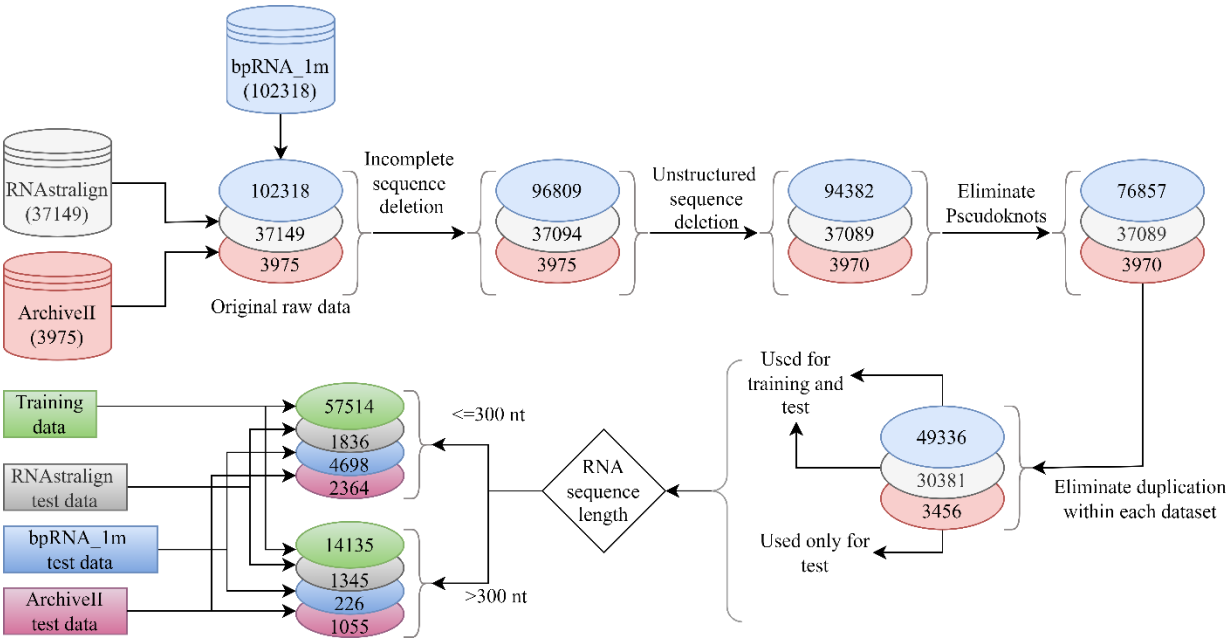

Figure 1: The data processing flow, where the number of RNAs after each step is labeled and color coded for easy visualization.

After the data processing, we randomly split the RNAStralign and bpRNA_1m datasets in a 9:1 ratio to construct training and test sets. Finally, our training dataset is constructed by the union of the RNAStralign and bpRNA_1m training data sets whereas, our model is tested separately with the bpRNA_1m test set and RNAStralign test set. Further to confirm the robustness of our model against unseen data, we test our model using the ArchiveII dataset which is never used for training.

We want to put forward a model that can manage RNA sequences of any length. However, our analysis shows that we cover more than 80% of our RNA dataset, which consists of sequences 300 nucleotides or shorter. The long RNA sequences (>300nt) are not very frequent in the dataset. Therefore, we separate out long-range and short-range interactions by dividing our dataset based on the length and train the model separately for those two subsets of data. The flow of data preprocessing is also discussed in Figure 1. As RNA shows very versatile structures in nature, it is important to separate the RNA sequences efficiently to extract the folding information using the learning model. So, the RNA sequences that lie under 300nt length are further separated into three clusters based on sequence length using the K-means clustering algorithm to efficiently capture the short-range interactions. The length-wise distribution of RNA sequences in each dataset is illustrated in Figure S2 and Figure S3 of the supplementary material.

Input feature selection

We aim to predict the base pairing probabilities from a given RNA sequence to get its whole secondary structure information. Suppose, $S = (s_1, s_2, s_3 \dots s_N)$, where N is the length of the input sequence and $s_i \in$ {'A', 'U', 'G', 'C'}. Our final output will be in the form of $T \in \{0,1\}^{N\times 3}$

where 3 represents one of the three labels ('.' / '(' / ')'). $T_{i,0} = 1$, denotes the output label 'dot' i.e. s_i is part of the loop region whereas $\{T_{i,1}\} = 1$ (or $\{T_{i,2}\} = 1$) represents the 'opening' (or 'closing') label, which implies s_i is part of a paired opening (or closing) stem region of the secondary structure. The details about stem and loop regions in the secondary structure of RNA are discussed in section S1 of our supplementary material.

Here, we have divided each of the RNA sequences with length >300nt into $((n-w)/d) + 1$ subsequences, where n is the sequence length, w is the window size, and d is the length of the slide. This process is described in step 1 of Figure 2.

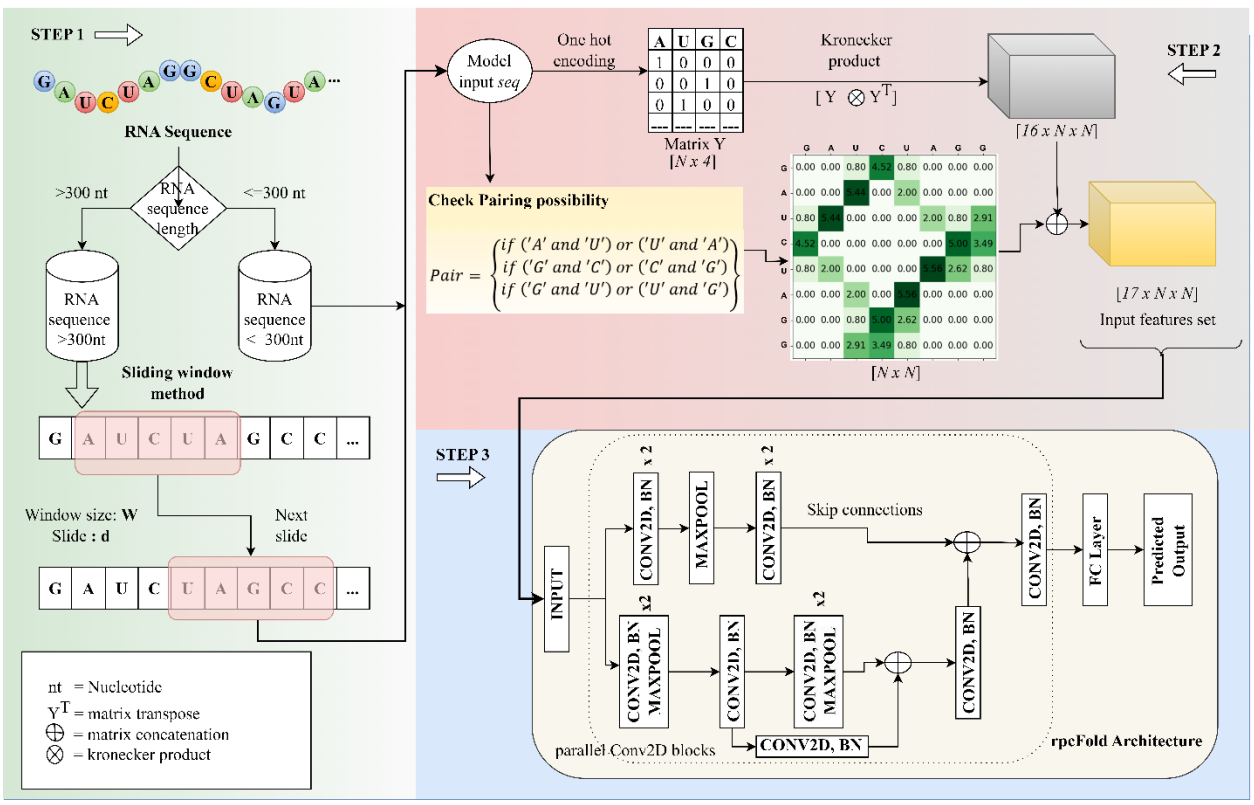

Figure 2: The overall architecture of our rpcFold model. Step 1(green background): The input sequence is separated based on the length. For each long RNA sequence, subsequences will be generated using the sliding window method, and each of the subsequences is then passed to step 2 for calculating the features set. The small sequences are directly passed to step 2 without subsequence generation. Step 2 (red background): Here we calculate the pairing possibility matrix based on three given rules. We also

calculate all 16 possible base pairs using the Kronecker product and use it as a feature. After concatenating both features, we have $[17 \times N \times N]$ an input feature set which is passed to our rpcFold model in Step 3. Step 3 (blue background): Our model as shown in step 3 consists of two parallel residual 2D convolutional layers followed by batch normalization and maxpool layers. The last layer contains the fully connected layer which transforms the learned features into the $[N \times 3]$ output format.

In step 2, we convert each of our small sequences, or the subsequences of long RNAs, into a $Y \in \{0,1\}^{N\times 4}$ binary matrix using one-hot encoding representation as shown in Eq1, where N is the sequence length, and 4 refers to the number of base types.

$$Y_i = \begin{cases} [1,0,0,0], & where\ s_i='A' \\ [0,1,0,0], & where\ s_i='U' \\ [0,0,1,0], & where\ s_i='G' \\ [0,0,0,1], & where\ s_i='C' \end{cases} \qquad \text{(Eq 1)}$$

Then this data is transformed into an image-like representation using the Kronecker product so that it can handle all the short-range as well as long-range base interactions. The Kronecker product on the matrix Y and its transpose Y^T gives us a $[4N \times 4N]$ matrix representation. To consider all the possible base pairs, the matrix is reshaped into a $[16 \times N \times N]$ matrix, which is treated as an $[N \times N]$ image with 16 channels.

To reduce the sparsity of our proposed matrix, we introduce the second feature where we consider the pairing possibilities between each base separately. Here instead of one-hot encoding, we used the concept from CDPFold[12] to encode the input sequence. The input RNA sequence of length N is converted into a matrix $W \in [N \times N]$, based on the algorithm as discussed in Algorithm1.

Algorithm1: The algorithm to compute the base pairing possibility matrix W for a given input RNA sequence

Input: RNA sequence of length N

Output: Pairing possibility matrix W of size $[N \times N]$

Steps:
begin
$W_{i,j} = W_{left} = W_{right} = 0$
$\boldsymbol{if}\ R_i\ and\ R_j\ are\ paired, \boldsymbol{then}$
$W_{i,j} = P(R_i, R_j)$
$\alpha = 0\ and\ \beta = 0$
$\boldsymbol{if}\ R_{i-\alpha}\ and\ R_{j+\alpha}\ are\ paired, \boldsymbol{then}$
$W_{left} = W_{left} + exp\left(-\frac{\alpha^2}{2\tau^2}\right) \times P\left(R_{i-\alpha}, R_{j+\alpha}\right)$
$\alpha = \alpha + 1$
$\boldsymbol{if}\ R_{i+\beta}\ and\ R_{j-\beta}\ are\ paired, \boldsymbol{then}$
$W_{right} = W_{right} + exp\left(-\frac{\beta^2}{2\tau^2}\right) \times P\left(R_{i+\beta}, R_{j-\beta}\right)$
$\beta = \beta + 1$

$W_{i,j} = W_{i,j} + W_{left} + W_{right}$
$\boldsymbol{return}\ W_{i,j}$
end

At first, the matrix W is initialized to zeros. Then for each base $R_i and\ R_j$, we calculate the base pairing possibility between them. Here $P(R_i, R_j)$ denotes the pairing weight based on the number of hydrogen bonds between those two bases. The calculation of pairing weights for two bases $R_i\ and\ R_j$ at any two positions $i\ and\ j$, is shown in Eq2.

$$P\left(R_i, R_j\right) = \begin{cases} 2, & if\ \left(\left(R_i='A'\ and\ R_j='U'\right) or\ \left(R_i='U'\ and\ R_j='A'\right)\right) \\ 3, & if\ \left(\left(R_i='G'\ and\ R_j='C'\right)\ or\ \left(R_i='C'\ and\ R_j='G'\right)\right) \\ 0.8, & if\ \left(\left(R_i='G'\ and\ R_j='U'\right)\ or\ \left(R_i='U'\ and\ R_j='G'\right)\right) \\ 0, & Otherwise \end{cases} \qquad \text{(Eq 2)}$$

For any two positions, i and j, on the RNA sequence, here we also consider whether these two bases form paired bases on the stem as that increases the stability of the structure. Therefore, we must check for the pairing of bases on both sides of i and j. The base pair in the middle of a stem

shows relatively more stability than the one at the end. Thus, to capture these biological insights, the concept of Gaussian function is used to address the local weight. Finally, we get the position-specific pairing possibility matrix W of size $[1 \times N \times N]$ as our second feature. Both the features are described in detail in section S2 of the supplementary material.

After getting the Kronecker product result $[16 \times N \times N]$ and the $[1 \times N \times N]$ base pair possibility matrix, these features are concatenated to get the final input feature vector of size $[17 \times N \times N]$, which is the final input for our deep learning model.

Deep learning network architecture

Here we have proposed a residual parallel convolutional neural network based rpcFold model to predict the RNA secondary structure as shown in step 3 of Figure 2. The model consists of a sequence of convolutional layers followed by normalization layers to capture the intricate semantic representations of the input. The rpcFold model consists of two parallel 2D convolutional layers followed by batch normalization and maxpool layers. Here, we have added residual skip connections to achieve network resilience and stabilize the training by preventing the exploding gradient problem. Finally, we have a fully connected layer and a softmax activation function that transforms our learned high-level features into corresponding $[N \times 3]$ output formats suitable for classification.

Our rpcFold model can derive short-range interactions very efficiently and provide highly accurate and robust prediction results. After incorporating the sliding window concept along with our model, the ensemble model shows better performance for long RNA structure prediction as it

can capture the long-range interactions very accurately. Now, our model can handle RNA with any length and without unnecessary padding. It also reduces the overall time and space complexity from $O(N^2)$ to $O(W^2)$ where N is the length of the original RNA sequence and W is the length of window size with ($W<<N$).

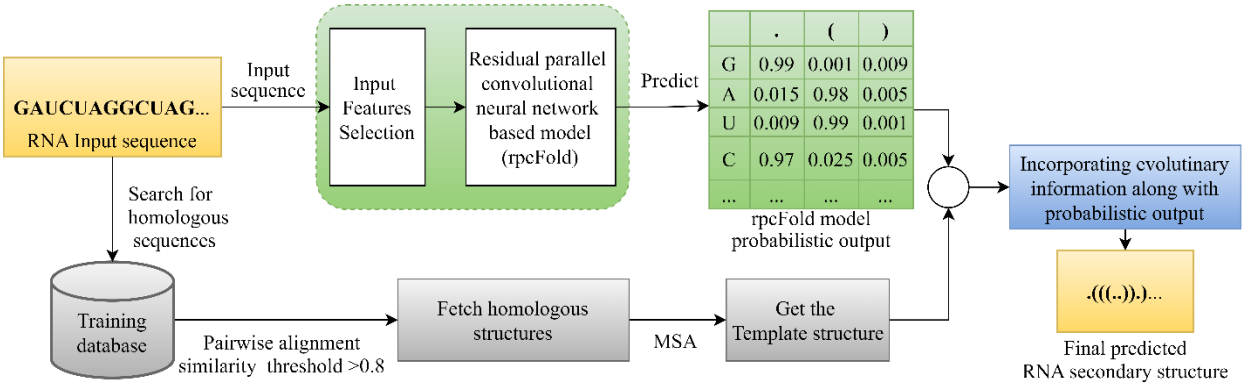

Figure 3: Incorporation of domain knowledge to have a meaningful outcome

Incorporating evolutionary information

Please note that only computational techniques suffer from biological insight. Therefore, we have incorporated biological information into the predicted result by applying the following constraints. To get a correct secondary structure we should check:

- Pairing/matching of all the opening '(' and closing ')' braces in the structure.
- Consider the precedence of opening and closing braces.
- The unpredicted structure information from the deep learning model for particular bases will also be handled carefully.

To fulfil the above constraints, we take the help of the evolutionary information. For each input sequence, we evaluate the corresponding homologous sequences present in our training dataset

and generate a template secondary structure. The overall flow is shown in Figure 3. If no homologous sequence is found, then for the unpredicted bases, we will take the output of the predecessor base as the output of the current base.

Results

Performance evaluation

Here, we consider the RNA secondary structure prediction as a classification problem[29], where a base can be unpaired or paired, and if paired then part of the opening or closing stem. We evaluated the performance of our RNA secondary structure prediction model through precision, recall, F1-score, and accuracy, which are defined as

$$precision = \frac{TP}{TP + FP} \quad \text{(Eq 3)}$$

$$recall = \frac{TP}{TP + FN} \quad \text{(Eq 4)}$$

$$F1\ score = \frac{(2 \times precision \times recall)}{precision + recall} \quad \text{(Eq 5)}$$

$$accuracy = \frac{(TP + TN)}{(TP + TN + FP + FN)} \quad \text{(Eq 6)}$$

When a base that is unpaired/left opening/right opening in the original structure is predicted truly as part of an unpaired region or paired stem region, then it is referred to as TP (i.e., true positive). If the base is not present as unpaired/left opening/right opening in the native structure but present in the predicted structure then it is considered as FP (false positive). The vice versa condition is known as FN (false negative) when a base is incorrectly predicted as not present in the predicted structure. When a base is not present as unpaired/left opening/right opening in the original structure and is also found as absent in the predicted structure is considered as TN (true

negative).F1-score is the harmonic mean of precision and recall and can handle the uneven class distribution. Accuracy is the ratio between correctly predicted by the total number of observations. It gives the overall performance of our model over all the classes.

Experimental results on different datasets

To benchmark the performance of our model, we have taken three widely used benchmark datasets where RNA length varies from 20nt to 4381nt, for the experimental studies. For small sequences, we used the sequence as it is for model training and testing. On the other hand, for sequences longer than 300nt, we have incorporated the sliding window-based model for training and testing. It helps in reducing the inference time, improves signal-to-noise ratio by avoiding unnecessary padding, and thus increases the robustness of the model. The selection of the window size and slide size has a significant impact on the prediction. If the window size is too small, the extracted features may be incomplete. Conversely, setting the window size too large can introduce redundant information into the feature matrix, resulting in longer model training times and potentially affecting the accuracy of the final prediction model. Supplementary Table S1 demonstrates the effects of modifying the hyperparameters like window size and slide value on both the training and testing phases. Further, we have also demonstrated the performance of rpcFold for short and long sequences separately in Supplementary Table S2.

Our proposed structure prediction model rpcFold is trained using the RNAStralign training set and bpRNA_1m training set and tested separately on the RNAStralign test set and bpRNA_1m test set. The RNAStralign dataset contains sequences from eight different families and contains mostly regulatory RNAs such as tRNAs and rRNAs. Whereas, the bpRNA_1m dataset contains

sequences from very diverse RNA families and shows versatile sequence lengths. Further, the ArchiveII dataset is used to check the performance of our model against cross-family data. Since this set is not revealed during the training, the profound accuracy in this test set ensures that our model has good generalizability. Table 1 shows the performance evaluation results of our model on the three benchmark test sets bpRNA_1m, RNAStralign, and ArchiveII.

Table 1: A comparison table of our method rpcFold with other existing methods. We have used accuracy (%) and F1-score measurement of each model on given three benchmark test datasets. A hyphen (‘-‘) in the entry indicates the non-availability of the metric score.

	Accuracy (%)			**F1-score**		
Method	**RNAStralign**	**bpRNA_1m**	**ArchiveII**	**RNAStralign**	**bpRNA_1m**	**ArchiveII**
Our model (rpcFold)	**92.60**	**87.25**	**86.12**	**0.89**	**0.808**	**0.791**
Ufold (2022) [9]	87.5	-		-	0.654	0.905
MXFold2 (2021)[30]	-	-	-	-	0.558	0.769
E2EFold (2020) [25]	78.8	-	-	0.821	0.189	0.554
SPOT-RNA (2019)[13]	31.4	-	-	-	0.619	0.711
LinearFold (2019) [22]	-	-	-	-	0.551	0.647
CDPFold (2019)[12]	-	-	-	0.614	-	0.545
RNAstructure (2013) [20]	38.5	-	-	-	0.532	0.628

Comparison with existing techniques

We have compared the performance of our only sequence feature-based model with the existing sequence feature-based benchmark methods. Table 1 presents a comparative analysis of our model's performance on three distinct test sets in comparison to seven existing works. Our model outperforms the others in terms of both accuracy and F1-score by at least 5% and 7% specifically on the RNAStralign test set. The ArchiveII dataset also shows a marginal improvement in the F1-score compared to all existing models except Ufold[9], where we are low in F1-score. For MXFold2[30], our F1-score is 22% better on bpRNA_1m, whereas on ArchiveII, we achieved a 2.2% improvement. Notably, our model rpcFold exhibits a significant enhancement in performance on the diverse bpRNA_1m test set, characterized by versatile sequences. The performance metrics of existing works are extracted from their respective published papers for fair comparison and are shown in detail in Table 1.

Additionally, we have compared the prediction accuracy of our rpcFold model with existing MXfold2[30] and RNAfold[36] web servers using three different length RNA sequences with RNA ID: AB264553.ct(Bacilli, 16sRNA) and RNA ID: AJ536100.ct (Actinobacteria, 16sRNA) from RNAstralign test set and RNA ID: bpRNA_CRW_24401 (Pseudomonas aeruginosa UCBPP-PA14, Bacteria, tRNA) from bpRNA_1m test set. The above three RNA sequences, taken for testing are all of different lengths, from different RNA families, and are associated with different cellular functions. Therefore, achieving profound prediction accuracy in all three test cases indicates that our model has strong generalizability for diverse sequences. From Figure 4, Figure 5, and Figure 6, we understand that the existing works are unable to predict the junctions(multiloop) in the structure properly for the long-range RNA sequences and hence the

prediction accuracy is hampered. Whereas our rpcFold method can efficiently predict all types of loops and stem structure for short as well as long sequences. For short sequences, the performance of our rpcFold and MXfold2 are the same and we achieve a 16% improvement in accuracy over RNAfold. For long sequences as shown in Figures 5 and 6, we achieve 10-19 % improvement in accuracy for an average length of 573nt sequence and 30% betterment in accuracy for a 1515nt long sequence. Thus, we can conclude that our method is capable of handling sequences of any length efficiently compared to the existing prediction works.

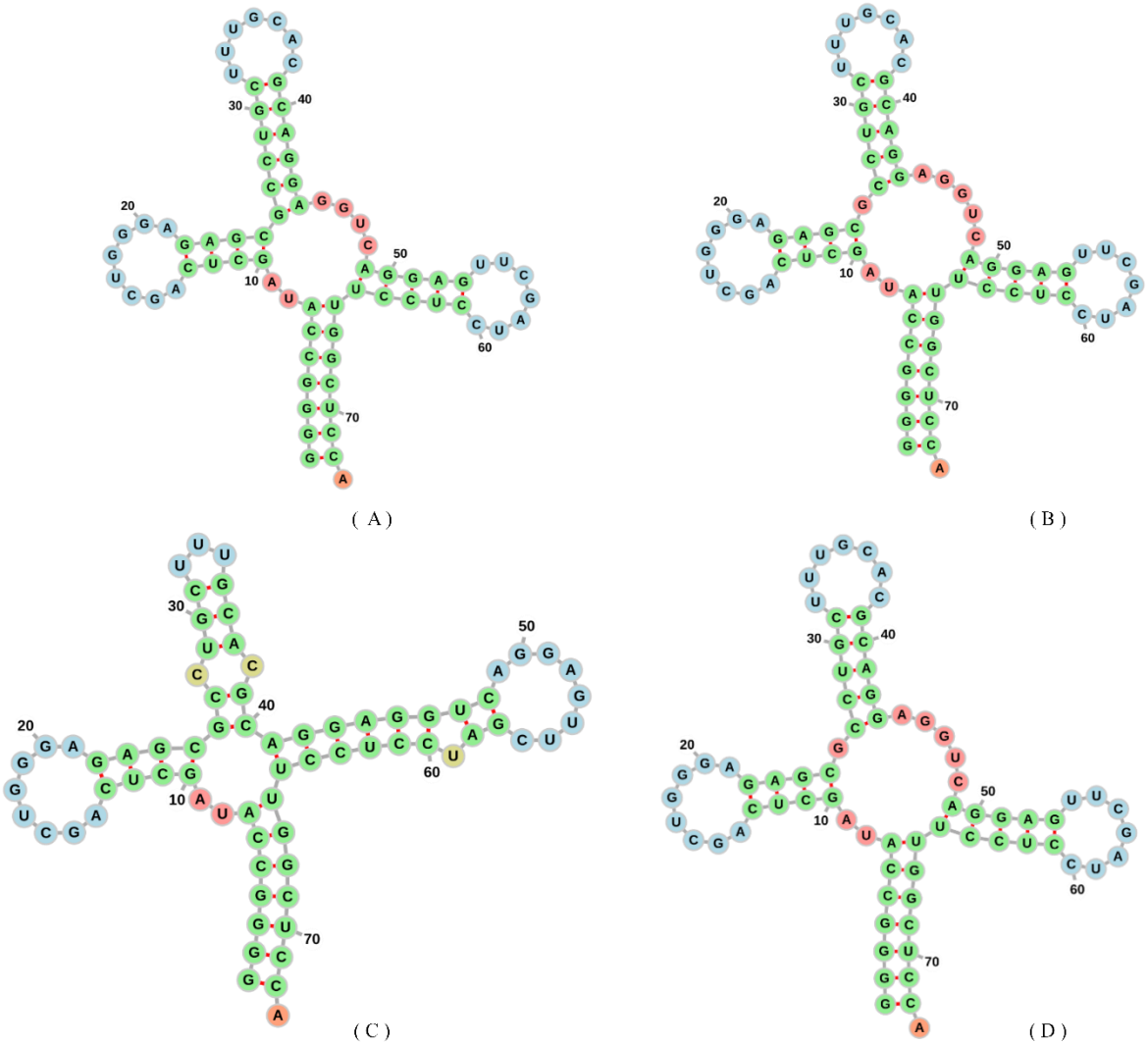

Figure 4: Prediction comparison of rpcFold, RNAfold, and MXFold2 with the native structure on RNA ID: bpRNA_CRW_24401.ct (Pseudomonas aeruginosa UCBPP-PA14, Bacteria, tRNA) taken from bpRNA_1m database where sequence length is 73nt. (A) native structure of the RNA, (B) structure predicted by our method rpcFold with 0.973 F1 score and 98.17% accuracy, (C) structure predicted by RNAfold[36] with 0.726 F1 score and 81.73% accuracy, and (D) structure predicted by MXold2 with 0.973 F1 score and 98.17% accuracy.

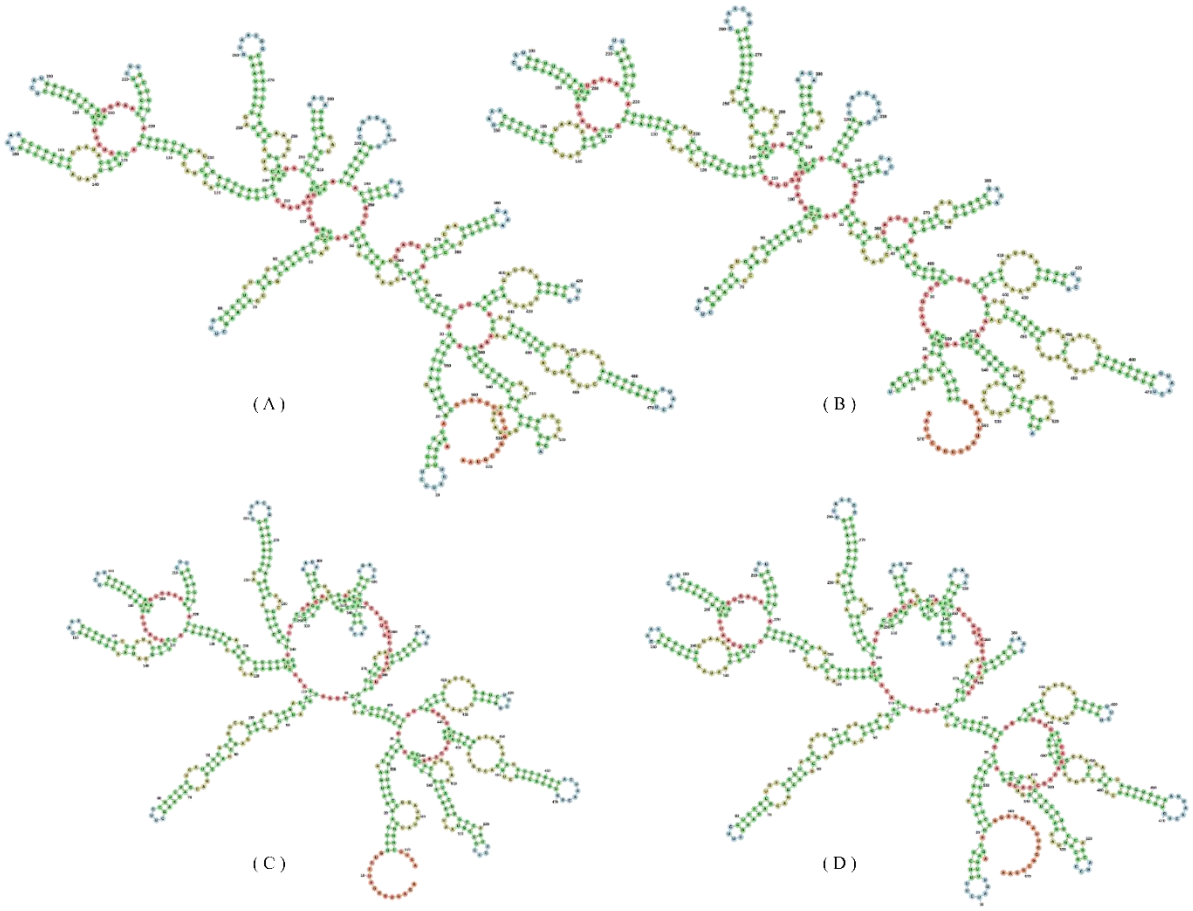

Figure 5: Prediction comparison of rpcFold, RNAfold and MXFold2 with native structure on RNA ID: AB264553.ct, (Bacilli, 16sRNA) from RNAstralign database where sequence length 573nt. (A) Native structure, (B) predicted structure by rpcFold with 0.972 F1-score and 98.13% accuracy, (C) predicted

structure by RNAfold with 0.727 F1-score and 81.8% accuracy, (D) predicted structure by MXFold2 with 0.855 F1-score and 90.3% accuracy.

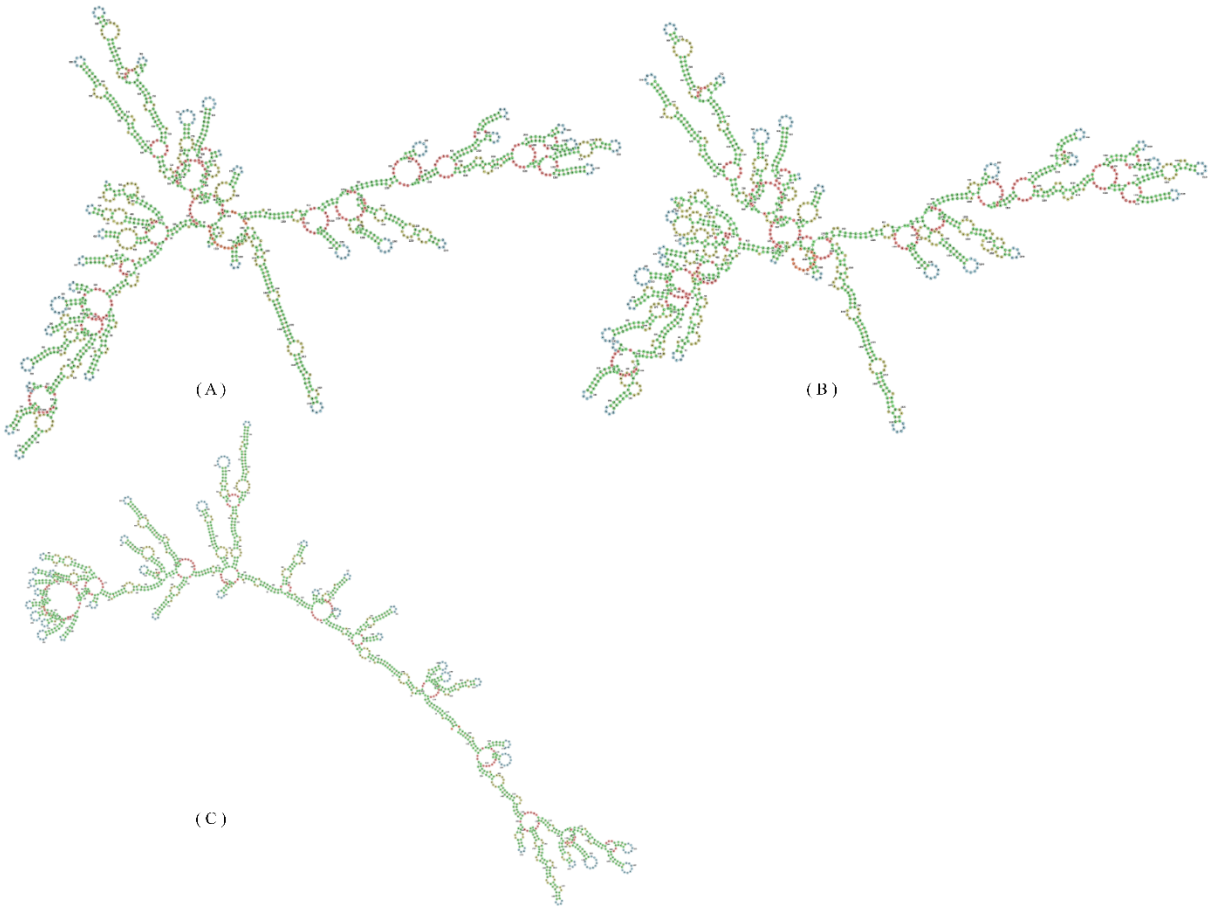

Figure 6: Prediction comparison of rpcFold, RNAfold and MXFold2 with native structure on RNA ID: AJ536100.ct, (Actinobacteria, 16sRNA) from RNAstralign database where sequence length 1515nt. (A) Native structure, (B) predicted structure by rpcFold with 0.966 F1-score and 97.75% accuracy, (C) predicted structure by RNAfold with 0.511 F1-score and 67% accuracy. MXfold2 webserver cannot handle sequences larger than 1000nt long.

Discussion

The starting approaches in the RNA structure prediction domain are mainly based on thermodynamics-based properties. Afterward, researchers incorporated thermodynamic features along with learning techniques for better learning of their model. However, most of the thermodynamic-based models use prior assumptions and they check the stability of individual stems and loops to calculate the stability of the whole secondary structure. Altogether, we avoided any prior assumptions and used only sequence features for more generic and wide applicability of our rpcFold model which relies only on RNA sequence information for predicting the corresponding secondary structure.

Further, CDPFold[12] and ATTFold[31] have trained their model merely on the RNAStralign dataset, which has data from simply 8 different families, and tested on a very similar ArchiveII dataset. As an outcome, their model became biased, and though they received very high training and test accuracy but showed very poor performance when tested on diverse RNA datasets like bpRNA_1m where sequences are from more than 2400 different families. So, we aim to propose a generalized model, which can provide satisfactory results both within and across family datasets.

Initially, we examined several established neural architectures such as ResNet32[32], VGG-16[33], and MobilenetV2[34], with parameter counts of 23M, 37.9M, and 2.7M respectively. Deep neural networks with hundreds of hidden layers excel in managing vast datasets comprising millions of data points. Most of these well-known deep neural network architectures are trained on millions of data and then extract the features from datasets. For RNA secondary structure prediction, the amount of available data is limited. So, these existing deep learning

models do not give us satisfactory results and here the validation accuracies we receive from these models are less than 0.70. Supplementary Table S3 shows the number of training parameters used in these existing models along with the parameters of rpcFold. Further, in traditional machine learning methods like random forests and decision trees, a large number of features are typically required for effective input processing. However, when dealing with sequences, feature availability is limited, which leads to a preference for deep-learning networks.

Here, we proposed our rpcFold model which used the advantage of both skip connections along with parallel convolutional layers. The skip connection helps in more efficient and stable learning by vanishing the gradient problem during training whereas the parallel convolutional layers have the potential for capturing diverse features simultaneously, leading to better performance on sequence feature extraction tasks. We also introduce the sliding window method which gives us the flexibility to handle RNA sequences of any length without illogically breaking them in-between or without adding unnecessary padding which increases the computational cost. The sliding concept also helps the model capture local patterns and dependencies and facilitates the extraction of relevant features from input sequences.

To enhance accessibility, rpcFold is available as a free web service that is accessible at https://cosmos.iitkgp.ac.in/rpcFold/. Here in our rpcFold webserver, user can directly upload their RNA sequence in fasta file format or can write the RNA sequence along with its name in the text box. Using the help of a pre-trained rpcFold model, our server predicts the RNA secondary structure from the given input sequence. The server provides the output structure in dot-bracket notation in the display box along with the download output file option. For

visualizing the predicted structures, the server offers an interface connection to the Forna tool[35]. The steps of running the rpcFold server are given in detail in Supplementary section S5.

Conclusion

We proposed a novel algorithm for the secondary structure prediction of noncoding RNAs. As the length of the noncoding RNAs varies from a few nucleotides to thousands of nucleotides, here we integrate the concept of residual parallel convolutional neural network, which can efficiently capture the short-range as well as the long-range interactions between the bases. Further sliding window concept is also incorporated to handle long sequences in the dataset, to avoid sequence breaking or unnecessary padding as done in existing works. We have created our input feature set using only sequence information from the existing benchmark dataset. Further, we have converted the raw sequence into an image-like representation where all the possible base pairing possibilities are considered. Another feature matrix considering the base pairing possibilities of each nucleotide, whether present in stem or loop region is generated based on a defined Gaussian local weighted function. We concatenate both the feature matrix and pass it to our rpcFold model for model training. We evaluate our rpcFold model using benchmark RNA datasets from both within and across families, showing that rpcFold surpasses previous methods notably on across-family datasets, achieving a 15% improvement from the previous best score. It also shows a 5-40% improvement in within family dataset over existing models. In the future, we will add environmental features like salt concentration, temperature constant, and pseudoknot concepts for predicting the tertiary structure of noncoding RNAs as well. Our research yields promising results in the study of RNA secondary structure, offering valuable insights and serving

as an excellent resource for fellow researchers tackling structural prediction challenges and can easily incorporate in their respective research works.

Availability and Implementation

Our method rpcFold is implemented as a web service (https://cosmos.iitkgp.ac.in/rpcFold/), where there is no login requirement and it is free for all.

Acknowledgment

We do hereby acknowledge the support of Yogesh in web service and the use of Supercomputer facility Paramshakti of the Indian Institute of Technology Kharagpur.

Conflicts of Interest: None declared

Authors Contribution

N.S.: Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Software, Validation, Visualization, Writing – original draft, Writing – review and editing

P.M.: Conceptualization, Formal analysis, Investigation, Methodology, Resources, Supervision, Writing – original draft, Writing – review and editing

References

[1] L. A. Bugnon *et al.*, "Secondary structure prediction of long noncoding RNA: review and experimental comparison of existing approaches," *Briefings in Bioinformatics*, vol. 23, no. 4. Oxford University Press, Jul. 01, 2022. doi: 10.1093/bib/bbac205.

[2] R. C. Spitale and D. Incarnato, “Probing the dynamic RNA structurome and its functions,” *Nature Reviews Genetics*, vol. 24, no. 3. Nature Research, pp. 178–196, Mar. 01, 2023. doi: 10.1038/s41576-022-00546-w.

[3] D. Elliott and Michael Ladomery, *Molecular Biology of RNA*. Oxford University Press, 2017.

[4] E. Yang *et al.*, “GCNfold: A novel lightweight model with valid extractors for RNA secondary structure prediction,” *Comput Biol Med*, vol. 164, Sep. 2023, doi: 10.1016/j.compbiomed.2023.107246.

[5] Y. Chen and G. Varani, “ RNA Structure ,” in *eLS*, Wiley, 2010. doi: 10.1002/9780470015902.a0001339.pub2.

[6] J. Fallmann, S. Will, J. Engelhardt, B. Grüning, R. Backofen, and P. F. Stadler, “Recent advances in RNA folding,” *Journal of Biotechnology*, vol. 261. Elsevier B.V., pp. 97–104, Nov. 10, 2017. doi: 10.1016/j.jbiotec.2017.07.007.

[7] M. A. G. Matarrese, A. Loppini, M. Nicoletti, S. Filippi, and L. Chiodo, “Assessment of tools for RNA secondary structure prediction and extraction: a final-user perspective,” *J Biomol Struct Dyn*, vol. 41, no. 14, pp. 6917–6936, 2023, doi: 10.1080/07391102.2022.2116110.

[8] K. Sato and M. Hamada, “Recent trends in RNA informatics: a review of machine learning and deep learning for RNA secondary structure prediction and RNA drug discovery,” in *Briefings in Bioinformatics*, Oxford University Press, Jul. 2023. doi: 10.1093/bib/bbad186.

[9] L. Fu, Y. Cao, J. Wu, Q. Peng, Q. Nie, and X. Xie, “UFold: fast and accurate RNA secondary structure prediction with deep learning,” *Nucleic Acids Res*, vol. 50, 2021, doi: 10.1093/nar/gkab1074.

[10] M. Tagashira and K. Asai, “ConsAlifold: considering RNA structural alignments improves prediction accuracy of RNA consensus secondary structures,” *Bioinformatics*, vol. 38, no. 3, pp. 710–719, Feb. 2022, doi: 10.1093/bioinformatics/btab738.

[11] J. Singh, K. Paliwal, T. Zhang, J. Singh, T. Litfin, and Y. Zhou, “Improved RNA secondary structure and tertiary base-pairing prediction using evolutionary profile, mutational coupling and two-dimensional transfer learning,” *Bioinformatics*, vol. 37, no. 17, pp. 2589–2600, Sep. 2021, doi: 10.1093/bioinformatics/btab165.

[12] H. Zhang *et al.*, “A new method of RNA secondary structure prediction based on convolutional neural network and dynamic programming,” *Front Genet*, vol. 10, no. MAY, 2019, doi: 10.3389/fgene.2019.00467.

[13] J. Singh, J. Hanson, K. Paliwal, and Y. Zhou, “RNA secondary structure prediction using an ensemble of two-dimensional deep neural networks and transfer learning,” *Nat Commun*, vol. 10, no. 1, Dec. 2019, doi: 10.1038/s41467-019-13395-9.

[14] B. Knudsen and J. Hein, “Pfold: RNA secondary structure prediction using stochastic context-free grammars,” *Nucleic Acids Res*, vol. 31, no. 13, pp. 3423–3428, Jul. 2003, doi: 10.1093/nar/gkg614.

[15] S. H. Bernhart, I. L. Hofacker, S. Will, A. R. Gruber, and P. F. Stadler, “RNAalifold: Improved consensus structure prediction for RNA alignments,” *BMC Bioinformatics*, vol. 9, Nov. 2008, doi: 10.1186/1471-2105-9-474.

[16] S. E. Seemann, P. Menzel, R. Backofen, and J. Gorodkin, “The PETfold and PETcofold web servers for intra- and intermolecular structures of multiple RNA sequences,” *Nucleic Acids Res*, vol. 39, no. SUPPL. 2, Jul. 2011, doi: 10.1093/nar/gkr248.

[17] M. Hamada, K. Sato, and K. Asai, “Improving the accuracy of predicting secondary structure for aligned RNA sequences,” *Nucleic Acids Res*, vol. 39, no. 2, pp. 393–402, Jan. 2011, doi: 10.1093/nar/gkq792.

[18] M. Zuker, “Mfold web server for nucleic acid folding and hybridization prediction,” *Nucleic Acids Res*, vol. 31, no. 13, 2003, doi: 10.1093/nar/gkg595.

[19] C. B. Do, D. A. Woods, and S. Batzoglou, “CONTRAfold: RNA secondary structure prediction without physics-based models,” in *Bioinformatics*, Oxford University Press, Jul. 2006. doi: 10.1093/bioinformatics/btl246.

[20] J. S. Reuter and D. H. Mathews, “RNAstructure: software for RNA secondary structure prediction and analysis,” 2010. [Online]. Available: http://www.biomedcentral.com/1471-2105/11/129

[21] M. Akiyama, K. Sato, and Y. Sakakibara, “A max-margin training of RNA secondary structure prediction integrated with the thermodynamic model,” in *Journal of Bioinformatics and Computational Biology*, 2018. doi: 10.1142/S0219720018400255.

[22] L. Huang *et al.*, “LinearFold: Linear-time approximate RNA folding by 5’-to-3’ dynamic programming and beam search,” in *Bioinformatics*, 2019. doi: 10.1093/bioinformatics/btz375.

[23] H. Zhang, L. Zhang, D. H. Mathews, and L. Huang, “LinearPartition: Linear-time approximation of RNA folding partition function and base-pairing probabilities,” *Bioinformatics*, vol. 36, pp. I258–I267, 2020, doi: 10.1093/BIOINFORMATICS/BTAA460.

[24] Q. Zhao, Z. Zhao, X. Fan, Z. Yuan, Q. Mao, and Y. Yao, “Review of machine learning methods for RNA secondary structure prediction,” *PLoS Computational Biology*, vol. 17, no. 8. Public Library of Science, Aug. 01, 2021. doi: 10.1371/journal.pcbi.1009291.

[25] X. Chen, Y. Li, R. Umarov, X. Gao, and L. Song, “RNA SECONDARY STRUCTURE PREDICTION BY LEARNING UNROLLED ALGORITHMS,” in *8th International Conference on Learning Representations, ICLR 2020*, 2020.

[26] M. F. Sloma and D. H. Mathews, “Exact calculation of loop formation probability identifies folding motifs in RNA secondary structures,” *RNA*, vol. 22, no. 12, pp. 1808–1818, Dec. 2016, doi: 10.1261/rna.053694.115.

[27] Z. Tan, Y. Fu, G. Sharma, and D. H. Mathews, “TurboFold II: RNA structural alignment and secondary structure prediction informed by multiple homologs,” *Nucleic Acids Res*, vol. 45, no. 20, pp. 11570–11581, Nov. 2017, doi: 10.1093/nar/gkx815.

[28] P. Danaee, M. Rouches, M. Wiley, D. Deng, L. Huang, and D. Hendrix, “BPRNA: Large-scale automated annotation and analysis of RNA secondary structure,” *Nucleic Acids Res*, vol. 46, no. 11, pp. 5381–5394, Jun. 2018, doi: 10.1093/nar/gky285.

[29] D. H. Mathews, “How to benchmark RNA secondary structure prediction accuracy,” *Methods*, vol. 162–163. Academic Press Inc., pp. 60–67, Jun. 01, 2019. doi: 10.1016/j.ymeth.2019.04.003.

[30] K. Sato, M. Akiyama, and Y. Sakakibara, “RNA secondary structure prediction using deep learning with thermodynamic integration,” *Nat Commun*, vol. 12, no. 1, Dec. 2021, doi: 10.1038/s41467-021-21194-4.

[31] Y. Wang *et al.*, “ATTfold: RNA Secondary Structure Prediction With Pseudoknots Based on Attention Mechanism,” *Front Genet*, vol. 11, Dec. 2020, doi: 10.3389/fgene.2020.612086.

[32] K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition.” [Online]. Available: http://image-net.org/challenges/LSVRC/2015/

[33] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *arXiv preprint arXiv:1409.1556.*, Sep. 2014.

[34] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, “MobileNetV2: Inverted Residuals and Linear Bottlenecks.”

[35] P. Kerpedjiev, S. Hammer, and I. L. Hofacker, “Forna (force-directed RNA): Simple and effective online RNA secondary structure diagrams,” *Bioinformatics*, vol. 31, no. 20, pp. 3377–3379, Mar. 2015, doi: 10.1093/bioinformatics/btv372.

[36] A. R. Gruber, R. Lorenz, S. H. Bernhart, R. Neuböck, and I. L. Hofacker, “The Vienna RNA websuite.,” *Nucleic Acids Res*, vol. 36, no. Web Server issue, 2008, doi: 10.1093/nar/gkn188.