cture(0,0)(-30,0)10 (-30,-5)(0,1)10 (-35,0)(1,0)30 (0,30)10 (-5,30)(1,0)10 (0,35)(0,-1)30 picturepicture(0,0)(30,0)10 (30,-5)(0,1)10 (35,0)(-1,0)30 (0,30

Published online xx xxxx xxxx

Nucleic Acids Research, xxxx, Vol. xx, No. xx **1–7**
doi:10.1093/nar/gkn000

RNA secondary structure prediction using deep learning with thermodynamic integrations

Kengo Sato [1]*, Manato Akiyama [1] and Yasubumi Sakakibara [1]

[1]Department of Biosciences and Informatics, Keio University, 3–14–1 Hiyoshi, Kohoku-ku, Yokohama 223–8522, Japan

ABSTRACT

RNA secondary structure prediction is one of the key technologies for unveiling the essential roles of functional non-coding RNAs. Although machine learning-based rich-parametrized models achieved extremely high performance in prediction accuracy, a possibility of the risk of overfitting for such a model has been reported. We propose a new algorithm for predicting RNA secondary structures using deep learning with thermodynamic integrations, which enable us robust predictions. Our algorithm computes folding scores for the nearest neighbor loops using a deep neural network. Similar to our previous work, the folding scores are integrated with the traditional thermodynamic parameters to enable robust predictions. Then, the Zuker-style dynamic programming is employed to find an optimal secondary structure that maximizes the folding score. We also propose a new regularization, called the thermodynamic regularization, to train our deep neural network with avoiding the overfit to training data. Our algorithm (MXfold2) archived the most robust and accurate predictions in the computational experiments assuming the discovery of new non-coding RNAs. The source code of MXfold2 and the datasets used in the experiments are available at https://github.com/keio-bioinformatics/mxfold2/.

INTRODUCTION

Recent studies have unveiled that functional non-coding RNAs (ncRNAs) play essential roles such as transcriptional regulation and guiding modification, resulting in various biological processes ranging from development and cell differentiation to the cause of diseases (1). Since it is well-known that functions of ncRNAs are deeply related to their structures rather than primary sequences, discovering the structures of ncRNAs leads to understanding the functions of ncRNAs. However, there are severe difficulties in experimental assays to determine RNA tertiary structures such as nuclear magnetic resonance (NMR) and X-ray crystal structure analysis due to high experimental costs and size limits of measurements on RNA. Therefore, instead of such experimental assays, we frequently perform computational prediction of RNA secondary structures, which is defined as a set of base-pairs consisting of hydrogen bonds between nucleotides.

The most popular approach for predicting RNA secondary structures is based on thermodynamic models such as Turner's nearest neighbor model (2, 3), in which a secondary structure is decomposed into several characteristic substructures, called nearest neighbor loops, such as hairpin loops, internal loops, bulge loops, base-pair stackings, multi-branch loops and external loops, as shown in Fig. 1. The free energy of each nearest neighbor loop can be calculated by summing free energy parameters that characterize the loop. The free energy parameters have been determined in advance by experimental methods such as the optical melting experiment (2). The free energy of an entire RNA secondary structure is calculated by summing up the free energy of the decomposed nearest neighbor loops. We can employ the dynamic programming technique to find an optimal secondary structure that minimizes the free energy for a given RNA sequence. A number of tools including Mfold/UNAfold (4, 5), RNAfold (6, 7) and RNAstructure (8, 9) have adopted this approach.

An alternative approach utilizes machine learning techniques, which train scoring parameters for decomposed substructures from reference structures, instead of the experimental techniques. This approach has successfully been adopted by CONTRAfold (10, 11), ContextFold (12) and so on, and thus has enabled us to predict more accurate RNA secondary structures. However, a rich parameterization could easily cause overfitting to the training data, resulting in disabling robust prediction for a wide range of RNA sequences (13).

Hybrid methods that combine the thermodynamic and machine learning-based approaches to compensate for each other's shortcomings have been developed. SimFold (15, 16) estimated more accurate thermodynamic parameters from training data including RNA sequences and their known secondary structures as well as their free energy of known secondary structures. We previously proposed MXfold (17) that combines the thermodynamic energy parameters and rich-parameterized weight parameters. It learns more precise parameters for substructures observed in training data and avoids overfitting the rich-parameterized weight parameters to the training data by falling back to the thermodynamic parameters for unobserved substructures.

In recent years, deep learning has made remarkable progress in a wide range of fields, including bioinformatics

*To whom correspondence should be addressed. Tel: +81 45 566 1511; Fax: +81 45 566 1789; Email: satoken@bio.keio.ac.jp

© xxxx The Author(s)
This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (http://creativecommons.org/licenses/by-nc/2.0/uk/) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

cture(0,0)(-30,0)10 (-30,-5)(0,1)10 (-35,0)(1,0)30 (0,-30)10 (-5,-30)(1,0)10 (0,-35)(0,1)30 picturepicture(0,0)(30,0)10 (30,-5)(0,1)10 (35,0)(-1,0)30 (0,

cture(0,0)(-30,0)10 (-30,-5)(0,1)10 (-35,0)(1,0)30 (0,30)10 (-5,30)(1,0)10 (0,35)(0,-1)30 picturepicture(0,0)(30,0)10 (30,-5)(0,1)10 (35,0)(-1,0)30 (0,30

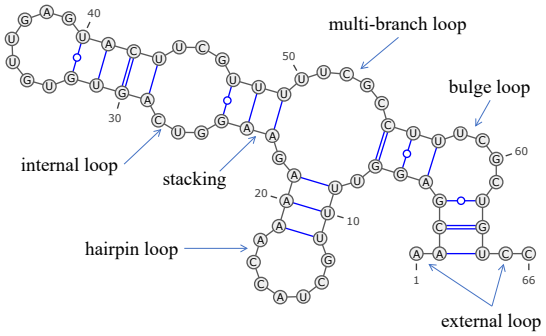

Figure 1. An RNA secondary structure can be decomposed into several types of the nearest neighbor loops including hairpin loops (e.g., bases 11–19), internal loops (e.g. bases 25–29 and 43–47), bulge loops (e.g. bases 4–5 and 57–62), base-pair stackings (e.g., bases 23–24 and 48–49), multi-branch loops (e.g. bases 7–9, 21–23 and 49–55) and external loops (e.g. bases 1–2 and 64–66). Drawing was made using VARNA (14).

and life science. SPOT-RNA (18) and E2Efold (19) have been developed using deep neural networks (DNNs) for the RNA secondary structure prediction. Both algorithms formulated the RNA secondary structure prediction as multiple binary classification problems that predict whether every pair of nucleotides forms a base-pair or not with a deep neural network trained from a large amount of training data. As with the other machine learning-based methods described above, concerns about overfitting due to the rich parameterization have still remained.

Inspired by MXfold and the DNN-based RNA secondary structure prediction methods, in this paper, we propose a new algorithm for predicting RNA secondary structures using deep learning with thermodynamic integrations, which enable us robust predictions. Our algorithm computes four types of folding scores for each pair of nucleotides using a deep neural network. The folding scores are used to calculate scores of the nearest neighbor loops. Similar to MXfold, we integrates the folding scores with the Turner's nearest neighbor free energy parameters. Then, our algorithm predicts an optimal secondary structure that maximizes the sum of the scores of the nearest neighbor loops by using the Zuker-style dynamic programming technique. The deep neural network is trained from training datasets so as to minimize the structured hinge loss function with the thermodynamic regularization that makes our model robust. The resulting method, called MXfold2, achieved the best prediction accuracy not only on the sequence-wise cross-validation, but also on the family-wise cross-validation, indicating the robustness of our algorithm.

MATERIALS AND METHODS

Our algorithm consists of the following three components: (1) a deep neural network that computes folding scores of nearest neighbor loops from a given sequence, (2) dynamic programming that calculates a secondary structure that maximizes the sum of loop scores, and (3) structured SVM that trains neural network parameters using these two components.

Computing folding scores with a deep neural network

Figure 2 shows the overview of our deep neural network that calculates four types of folding scores for each pair of nucleotides in a given sequence: helix stacking, helix open, helix close and unpaired region.

The input of the network is an RNA sequence of length L. Each base (`A`, `C`, `G` or `U`) is encoded into d-dimensional embeddings using a trainable embedding function.

The $L \times d$-dimentional sequence embedding is entered into 1D convolution blocks (Block 1) repeated N_1 times. Each 1D convolution block consists of two convolutional layers, each with a kernel size of 5 and 3, respectively, and a depth of d. The layer normalization (20) and the continuously differentiable exponential linear units (CELU) activation function (21) are employed. A dropout rate of 50 % is used to avoid overfitting (22). To efficiently train deep neural networks, the residual connection (23) that skips each layer is employed.

The next block (Block 2) contains N_2 layers of the bidirectional long short-term memory (LSTM) networks (24). The number of hidden units of each direction of the LSTM network is $d/2$. Each LSTM layer is followed by the layer normalization and CELU activation function. A dropout rate of 50 % is also used in LSTM layers.

The $L \times d$ matrix obtained so far is divided into two matrices of size $L \times d/2$, one of two matrices expands in the column direction and the other in the row direction, and then the two matrices are concatenated into $L \times L \times d$ matrix. This matrix is entered into 2D convolution blocks (Block 3) repeated N_3 times. Each 2D convolution block consists of two convolutional layers, each with a kernel size of 5×5 and 3×3, respectively, and a depth of d. The layer normalization, CELU and a dropout rate of 50 % are also utilized.

Finally, four types of folding scores for each pair of i-th and j-th nucleotides are calculated by 3-layer multilayer perceptron with h hidden units.

We tuned the hyperparameters $d = 64$, $N_1 = 4$, $N_2 = 2$, $N_3 = 4$ and $h = 32$.

Predicting RNA secondary structures with dynamic programming

Similar to traditional folding methods such as RNAfold and CONTRAfold, our algorithm adopts the nearest neighbor model (3, 25) as the RNA secondary structure model without pseudoknots. Alternative to the thermodynamic parameters, our algorithm calculates the free energy of the nearest neighbor loops using four types of folding scores computed by the deep neural network described in the previous section. The four types of the folding scores include helix stacking scores, helix opening scores, helix closing scores, and unpaired region scores (Fig 2). For example, the folding score of the internal loop consisting of bases 25–29 and 43–47 in Fig. 1 is calculated by summing the helix closing score for (25, 47), the helix opening score for (28, 44), unpaired region scores for (26, 28) and (44, 46). Similarly, the folding score of the base-pair stacking consisting of 23–24 and 48–49 in Fig 1 is calculated from the helix stacking score for (23, 49).

Similar to MXfold, we integrate our model with the thermodynamic model by adding the negative of the Turner free energy (3) to the folding scores calculated by the deep neural network for each nearest neighbor loop. We define our

cture(0,0)(-30,0)10 (-30,-5)(0,1)10 (-35,0)(1,0)30 (0,-30)10 (-5,-30)(1,0)10 (0,-35)(0,1)30 picturepicture(0,0)(30,0)10 (30,-5)(0,1)10 (35,0)(-1,0)30 (0,

cture(0,0)(-30,0)10 (-30,-5)(0,1)10 (-35,0)(1,0)30 (0,30)10 (-5,30)(1,0)10 (0,35)(0,-1)30 picturepicture(0,0)(30,0)10 (30,-5)(0,1)10 (35,0)(-1,0)30 (0,30

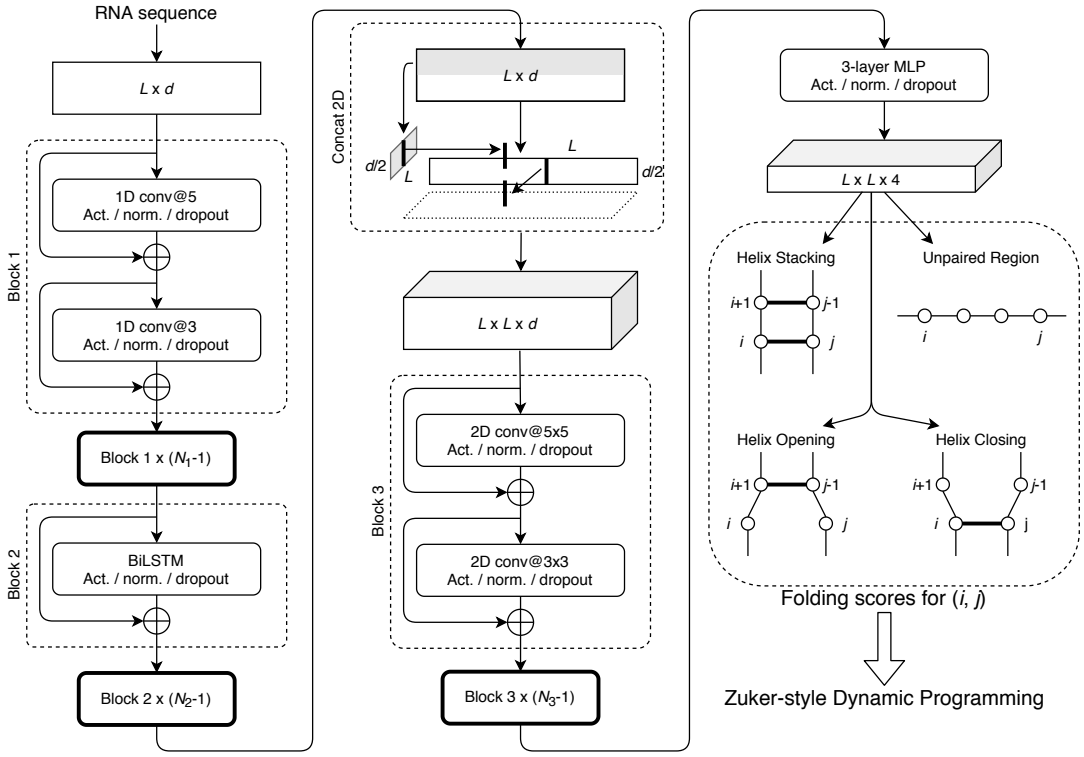

Figure 2. The network structure of our algorithm, where act. indicates the activation function, and norm. indicates the normalization function. The network calculates four types of folding scores for each pair of nucleotides (i, j) from a given RNA sequence of length L.

scoring function of a secondary structure y given an RNA sequence x as the following:

$$f(x,y) = f_T(x,y) + f_W(x,y), \quad (\mathbf{1})$$

where $f_T(x,y)$ is the contribution of the thermodynamic parameters, i.e. negative of the free energy of y, and $f_W(x,y)$ is the contribution of the deep neural network, which is calculated by the sum of the folding scores over the decomposed nearest neighbor loops as described above. The Zuker-style dynamic programming can be applied to find an optimal secondary structure that maximizes the scoring function (**1**).

Training parameters by the max-margin framework

To optimize the network parameters λ, we employ a max-margin framework called structured support vector machines (SSVM) (26). Given a training dataset $\mathcal{D} = \{(x^{(k)}, y^{(k)})\}_{k=1}^{K}$, where $x^{(k)}$ is the k-th RNA sequence and $y^{(k)} \in \mathcal{S}(x^{(k)})$ is the reference secondary structure for the k-th sequence $x^{(k)}$, we aim to find λ that minimizes the objective function:

$$\mathcal{L}(\lambda) = \sum_{(x,y)\in\mathcal{D}} \left\{ \left(\max_{\hat{y}\in\mathcal{S}(x)} [f(x,\hat{y}) + \Delta(y,\hat{y})] - f(x,y) \right) + C_1 [f(x,y) - f_T(x,y)]^2 + C_2 ||\lambda||_2 \right\}. \quad (\mathbf{2})$$

The first term is the structured hinge loss function (26). Here, $\Delta(y,\hat{y})$ is a margin term of $\hat{y}$ for y defined as:

$$\Delta(y,\hat{y}) = \delta^{\mathrm{FN}} \times (\text{\# of false negative base-pairs}) + \delta^{\mathrm{FP}} \times (\text{\# of false positive base-pairs}),$$

where δ^{FN} and δ^{FP} are tunable hyperparameters to control the trade-off between sensitivity and specificity for learning the parameters. We used $\delta^{\mathrm{FN}} = 8.0, \delta^{\mathrm{FP}} = 1.0$ by default. The margin term for structured models enables robust predictions by maximizing the margin between $f(x,y)$ and $f(x,\hat{y})$ for $y \neq \hat{y}$. The second and third terms of Eq. (**2**) are regularization terms to penalize parameter values that are extremely out of a postulated distribution. The second term is our proposed thermodynamic regularization, which avoids that the folding score of the secondary structure differs significantly from the free energy of thermodynamic parameters. The third term of Eq. (**2**) is the ℓ_2 regularization. We used $C_1 = 0.125$ and $C_2 = 0.01$ by default. To minimize the objective function (**2**), we employ the Adam optimizer (27).

Our algorithm was implemented with Python and C++ using PyTorch (28) and pybind11 (https://github.com/pybind/pybind11/). The source code of our algorithm is available at https://github.com/keio-bioinformatics/mxfold2/.

cture(0,0)(-30,0)10 (-30,-5)(0,1)10 (-35,0)(1,0)30 (0,-30)10 (-5,-30)(1,0)10 (0,-35)(0,1)30 picturepicture(0,0)(30,0)10 (30,-5)(0,1)10 (35,0)(-1,0)30 (0,

cture(0,0)(-30,0)10 (-30,-5)(0,1)10 (-35,0)(1,0)30 (0,30)10 (-5,30)(1,0)10 (0,35)(0,-1)30 picturepicture(0,0)(30,0)10 (30,-5)(0,1)10 (35,0)(-1,0)30 (0,30

Table 1. The summary of datasets used in our experiments.

		# sequences	length	sources
Rivas Dataset	TrainSetA	3,166	10–734	(13)
	TestSetA	592	10–768	(13)
	TestSetB	430	27–244	(13)
bpRNA-1m	TR0	10,814	33–498	(18)
	TS0	1,305	22–499	(18)
RNAStrAlign	train	20,923	30–600	(19)
ArchiveII		3,967	28–1,800	(19)
bpRNA-new		5,401	33–489	this work

Table 2. Comparison between the base model and the use of the thermodynamic integrations.

	TestSetA			TestSetB		
	PPV	SEN	F	PPV	SEN	F
base model	0.794	0.824	0.804	0.461	0.545	0.494
+int.	0.799	0.825	0.808	0.500	0.571	0.527
+reg.	0.765	0.796	0.776	0.542	0.647	0.583
final model	0.754	0.778	0.761	0.571	0.650	0.601

+int. = the use of the thermodynamic-integrated folding scores., +reg. = the use of the thermodynamic regularization.

Datasets

To evaluate our algorithm, we performed computational experiments on several datasets. The first dataset has been carefully established by Rivas *et al.* (13), including TrainSetA, TestSetA, TrainSetB and TestSetB. TrainSetA and TestSetA were collected from the literature (10, 15, 16, 29, 30, 31). TrainSetB and TestSetB were extracted from the Rfam 10.0 (32), which contain 22 families with 3D structure annotations. The sequences in Train/TestSetB have less than 70 % sequence identity with the sequences in TrainSetA. We excluded a number of sequences that contain pseudoknotted secondary structures in the original data sources from all the four sub-datasets since all algorithms evaluated in this study were designed for RNA secondary structure prediction without pseudoknot. It is important to note that literature-based TrainSetA and Rfam-based TestSetB are structurally diverse, whereas it can be observed structural similarity between TrainSetA and TestSetA.

To compare our algorithm with SPOT-RNA (18), we employed the same dataset extracted from the bpRNA-1m dataset (11), which is based on the Rfam 12.2 (33) with 2,588 families. After removing the redundancy based on the sequence identity, the bpRNA-1m dataset was randomly split into three sub-datasets: TR0, VL0, and TS0, for training, validation and test, respectively. Note that this partition is not family-wise, but sequence-wise. SPOT-RNA performed the initial training and validation using TR0 and VL0, then was tested using TS0, followed by the transfer learning using other PDB-based datasets.

To confirm the robustness against "unseen" families, we built a new dataset that includes new families from the most recent Rfam 14.2 (34). Since the Rfam 12.2 that bases bpRNA-1m, the Rfam project has been actively collecting about 1,500 new RNA families including the families detected by new techniques (35). We first extracted these new families. Then, as with SPOT-RNA, we removed the redundancy by CD-HIT-EST (36) with the cutoff threshold of 80 %, and discard sequences whose lengths are more than 500 bp. We call our new dataset bpRNA-new.

We also compared our algorithm with E2Efold (19) on the same experimental setting, in which the RNAStrAlign dataset (37) and the ArchiveII dataset (38) were employed for the training and test, respectively. Note that this setting is also not family-wise, but sequence-wise since both datasets contain the same families.

Table 1 shows the summary of datasets used in our experiments.

Performance measure

We evaluated the accuracy of predicting RNA secondary structures through the positive predictive value (PPV) and the sensitivity (SEN) of the base-pairs, defined as:

$$PPV = \frac{TP}{TP+FP}, \quad SEN = \frac{TP}{TP+TN},$$

where TP is the number of correctly predicted base-pairs (true positives), FP is the number of incorrectly predicted base-pairs (false positives), and FN is the number of base-pairs in the reference structure that were not predicted (false negatives). We also used the F-value as the balanced measure between PPV and SEN, which is defined as their harmonic mean:

$$F = \frac{2 \times SEN \times PPV}{SEN + PPV}.$$

RESULTS

Effectiveness of the thermodynamic integrations

In order to investigate the effectiveness of the thermodynamic integrations, the main contribution of this paper, we conducted an experiment to compare the cases with and without the thermodynamic integrations. We implemented a base model that employs only the deep neural network to compute the folding score, that is, not to use the thermodynamic parameters for the folding score and the thermodynamic regularization. Then, we compared the base model with the use of the thermodynamic-integrated folding score and/or the thermodynamic regularization. All the models were trained from TrainSetA, and then were evaluated on TestSetA and TestSetB. The result in Table 2 shows that the base model achieved extremely high accuracy for TestSetA, which is structurally similar to TrainSetA, but quite poor for TestSetB, which is structurally diverse from TrainSetA, suggesting that the base model might fall into heavy overfitting. On the other hand, the full model that employs all the thermodynamic integrations performs much better accuracy for TestSetB compared with the base model, indicating that the thermodynamic integrations enable us robust predictions.

Comparison with the existing methods

We compared our algorithm with seven available folding algorithms: MXfold version 0.0.2 (17), CONTRAfold version

cture(0,0)(-30,0)10 (-30,-5)(0,1)10 (-35,0)(1,0)30 (0,-30)10 (-5,-30)(1,0)10 (0,-35)(0,1)30 picturepicture(0,0)(30,0)10 (30,-5)(0,1)10 (35,0)(-1,0)30 (0,-

cture(0,0)(-30,0)10 (-30,-5)(0,1)10 (-35,0)(1,0)30 (0,30)10 (-5,30)(1,0)10 (0,35)(0,-1)30 picturepicture(0,0)(30,0)10 (30,-5)(0,1)10 (35,0)(-1,0)30 (0,30

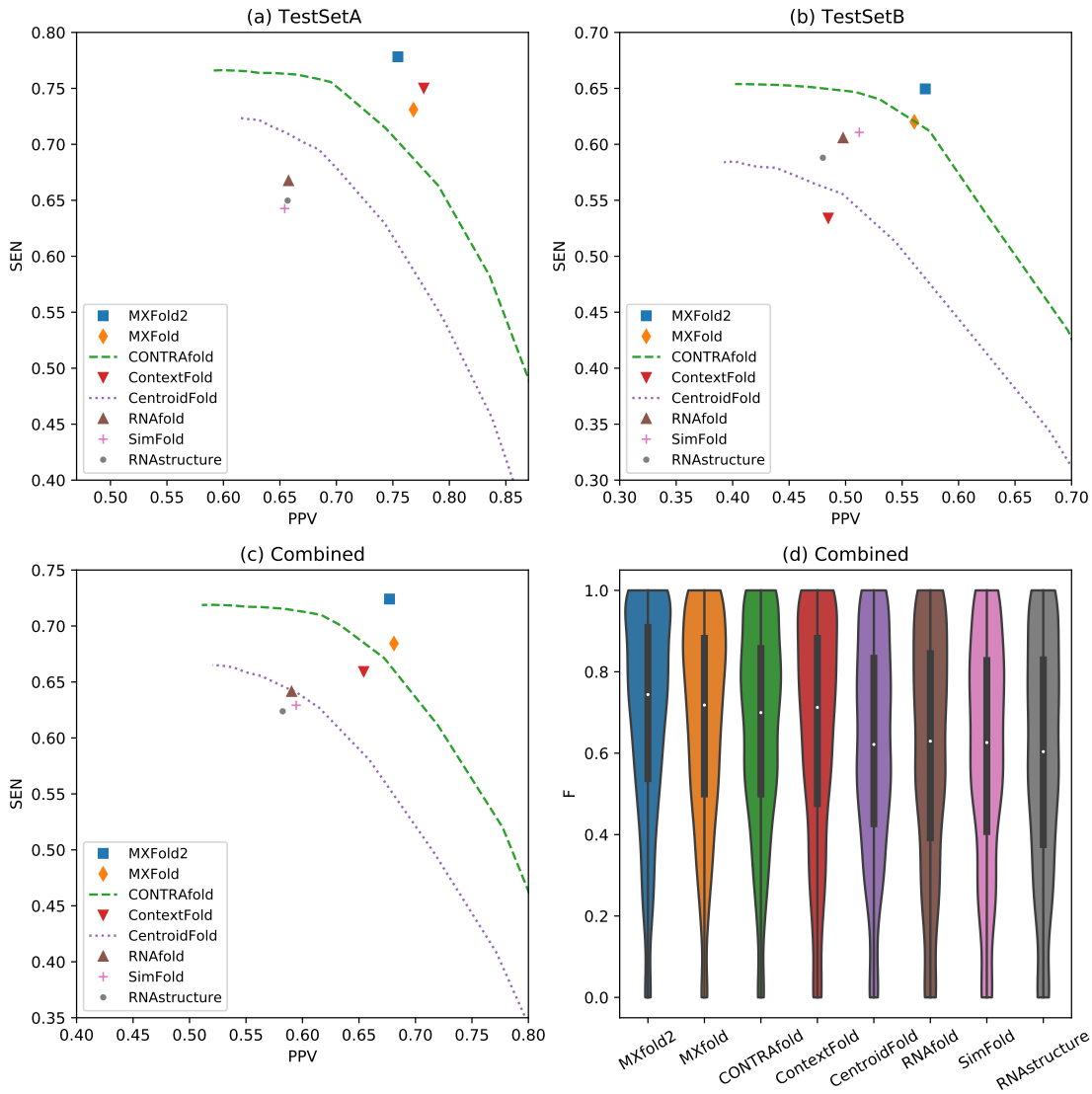

Figure 3. PPV-SEN plots on (a) TestSetA, (b) TestSetB and (c) the combined data set with TestSetA and TestSet B. (d) F-value on the combined test data for all the methods.

2.02 (10, 11), ContextFold version 1.00 (12), CentroidFold version 0.0.16 (39, 40), RNAfold in Vienna RNA package 2.4.14 (6, 7), SimFold version 2.1 (15, 16), and RNAstructure version 6.2 (8, 9). Three trainable algorithms, MXfold, CONTRAfold and ContextFold trained their parameters from TrainSetA.

Figures (3a) and (3b) show that PPV-SEN plots of prediction accuracy on TestSetA and TestSetB, respectively. The maximum expected accuracy (MEA)-based secondary structure predictions, CONTRAfold and CentroidFold, can control PPV and SEN by the parameter γ, and therefore, their accuracies are shown by the curves for multiple $\gamma \in \{2^n \mid -5 \leq n \leq 10\}$.

For the predictions on TestSetA, our algorithm, MXfold2 achieved the best accuracy (F=0.761), followed by ContextFold (F=0.759), MXfold (F=0.739), CONTRAfold (F=0.719 at γ=4.0). On the other hand, for the predictions on TestSetB, which is structurally diverse from TrainSetA, we can observe that ContextFold achieved the worst precision (F=0.502), as Rivas *et al.* (13) has pointed out that ContextFold might fall in the overfitting. In contrast, we cannot observe heavy overfitting as in ContextFold for MXfold2 (F=0.601), MXfold (F=0.581) and CONTRAfold (F=0.573 at γ=4.0).

Figures (3c) and (3d) show the PPV-SEN plot and the distribution of F-value on the combined dataset with TestSetA and TestSetB. These results indicate that MXfold2 (F=0.693) achieved the best accuracy, followed by the trainable methods, MXfold (F=0.673), CONTRAfold (F=0.658 at γ=4.0) and ContextFold (F=0.651), outperforming the thermodynamic based methods.

Comparison with DNN-based methods

We compared our algorithm, MXfold2, with two recently published DNN-based methods, SPOT-RNA (18) and E2Efold (19). We trained MXfold2 using their training

cture(0,0)(-30,0)10 (-30,-5)(0,1)10 (-35,0)(1,0)30 (0,-30)10 (-5,-30)(1,0)10 (0,-35)(0,1)30 picturepicture(0,0)(30,0)10 (30,-5)(0,1)10 (35,0)(-1,0)30 (0,

cture(0,0)(-30,0)10 (-30,-5)(0,1)10 (-35,0)(1,0)30 (0,30)10 (-5,30)(1,0)10 (0,35)(0,-1)30 picturepicture(0,0)(30,0)10 (30,-5)(0,1)10 (35,0)(-1,0)30 (0,30)10 (-5,30)(1,0)10 (0,35)(0,-1)30 pictu

Table 3. Comparison with SPOT-RNA on TS0 as the sequence-wise cross-validation and the bpRNA-new dataset as the family-wise cross-validation. All the methods were trained from TR0.

	Sequence-wise CV			Family-wise CV		
	PPV	SEN	F	PPV	SEN	F
MXfold2	0.520	0.682	0.575	0.585	0.710	0.632
SPOT-RNA	0.652	0.578	0.597	0.599	0.619	0.596

Table 4. Comparison with E2Efold on ArchiveII as the sequence-wise cross-validation and the bpRNA-new dataset as the family-wise cross-validation. All the methods were trained from the RNAStrAlign dataset.

	Sequence-wise CV			Family-wise CV		
	PPV	SEN	F	PPV	SEN	F
MXfold2	0.790	0.815	0.800	0.575	0.712	0.628
E2Efold	0.734	0.660	0.686	0.0474	0.0307	0.0361

datasets, then tested the prediction accuracy on their test datasets for direct comparison. Furthermore, since their datasets were designed only for the sequence-wise cross-validation, we additionally performed the family-wise cross-validation using the bpRNA-new dataset that does not contain any families in their training datasets.

Table 3 shows the comparison of the prediction accuracy between MXfold2 and SPOT-RNA, in which MXfold2 was trained on the TR0 dataset, whereas SPOT-RNA used the pre-trained model trained on the TR0 dataset and an additional dataset for the transfer learning, then tested on the TS0 dataset for the sequence-wise cross-validation, and the bpRNA-new dataset for the family-wise cross-validation. We can observe that MXfold2 is slightly worse in the sequence-wise cross-validation than SPOT-RNA, but more accurate in the family-wise cross-validation.

Table 4 shows the comparison of the prediction accuracy between MXfold2 and E2Efold, in which MXfold2 was trained on the RNAStrAlign dataset, whereas E2Efold used the pre-trained model trained on the RNAStrAlign dataset, then tested on the ArchiveII dataset for the sequence-wise cross-validation, and the bpRNA-new dataset for the family-wise cross-validation. The result indicates that E2Efold almost completely failed to predict secondary structures for unseen families, while MXfold2 accurately predicted their secondary structures.

DISCUSSION

Our experiments showed that several methods with rich parameterization achieved excellent performance for test datasets structurally similar to training datasets, but poor performance for test datasets structurally diverse from training datasets. Rivas *et al.* (13) have already pointed out that rich-parameterized methods might easily fall into the overfitting. Such secondary structure prediction methods would be useful if we could know a priori whether the sequence we want to predict is a member of a family included in the training data. However, if we have a knowledge about the family of the sequence to predict, the homology search for RNA sequences such as Infernal (41) is a better choice. In other words, the lack of prior knowledge of them is a major reason why we would like to predict secondary structures from single sequences. In principle, regardless of whether a given sequence is of a family included in the training data or not, any methods should be able to properly predict its secondary structure, since the secondary structure forms under the same physical laws for both cases. In general, machine learning-based methods can be expected to improve the prediction accuracy by increasing the number of training samples. While the development of sequencing techniques has made it easier to obtain sequence data, it is still not easy to increase the amount of training data because of the significant effort required to obtain those secondary structures. Therefore, it is not practical to predict secondary structures of newly discovered sequences using the methods with poor generalization ability, such as only memorizing sequences and their secondary structures within a large number of parameters of neural networks. We designed our experiments for predicting secondary structures of newly discovered RNAs using test datasets structurally diverse from training datasets, e.g., TestSetB for TrainSetA (originally developed by Rivas *et al.* (13)), and bpRNA-new dataset for TR0. This setting is one of the most essential situations for RNA secondary structure prediction from single sequences, and our method, which is able to achieve high accuracy under this setting, is useful in practice.

CONCLUSION

We proposed a deep learning-based algorithm for predicting RNA secondary structures with the thermodynamic integrations. Our experiments showed that the thermodynamic integrations have a significant effect on robust predictions. We conducted the benchmarks that compare our algorithm with the existing algorithms on not only the sequence-wise cross-validation, but also the family-wise cross-validation, which assumes one of most practical situations for the RNA secondary structure prediction from single sequences. The results showed that our algorithm achieved the best prediction accuracy without the overfitting to training data as shown in other rich-parameterized methods such as ContextFold and E2Efold. The proposed algorithm (MXfold2) should be useful for improving RNA structure modeling, especially for newly discovered RNAs.

ACKNOWLEDGEMENTS

This work was partially supported by Grant-in-Aid for Scientific Research (B) (No. 19H04210) and Challenging Research (Exploratory) (No. 19K22897) from the Japan Society for the Promotion of Science (JSPS) to K. S., and Grant-in-Aid for JSPS Fellows (No. 18J21767) from JSPS to M. A. This work was also supported by Grant-in-Aid for Scientific Research on Innovative Areas "Frontier Research on Chemical Communications." The super-computer system was provided by the National Institute of Genetics (NIG), Research Organization of Information and Systems (ROIS).

Conflict of interest statement. None declared.

cture(0,0)(-30,0)10 (-30,-5)(0,1)10 (-35,0)(1,0)30 (0,-30)10 (-5,-30)(1,0)10 (0,-35)(0,1)30 picturepicture(0,0)(30,0)10 (30,-5)(0,1)10 (35,0)(-1,0)30 (0,-30)10 (-5,-30)(1,0)10 (0,-35)(0,1)30 p

"draft" — 2020/8/11 — 14:48 — page 7 — #7
cture(0,0)(-30,0)10 (-30,-5)(0,1)10 (-35,0)(1,0)30 (0,30)10 (-5,30)(1,0)10 (0,35)(0,-1)30 picturepicture(0,0)(30,0)10 (30,-5)(0,1)10 (35,0)(-1,0)30 (0,30

REFERENCES

1. Hirose, T., Mishima, Y., and Tomari, Y. (May, 2014) Elements and machinery of non-coding RNAs: toward their taxonomy. *EMBO Rep.,* **15**(5), 489–507.
2. Schroeder, S. J. and Turner, D. H. (November, 2009) Optical melting measurements of nucleic acid thermodynamics. *Methods Enzymol.,* **468**, 371–387.
3. Turner, D. H. and Mathews, D. H. (January, 2010) NNDB: the nearest neighbor parameter database for predicting stability of nucleic acid secondary structure. *Nucleic Acids Res.,* **38**(Database issue), D280–2.
4. Zuker, M. (July, 2003) Mfold web server for nucleic acid folding and hybridization prediction. *Nucleic Acids Res.,* **31**(13), 3406–3415.
5. Markham, N. R. and Zuker, M. (2008) UNAFold: software for nucleic acid folding and hybridization. *Methods Mol. Biol.,* **453**, 3–31.
6. Hofacker, I. L. (July, 2003) Vienna RNA secondary structure server. *Nucleic Acids Res.,* **31**(13), 3429–3431.
7. Lorenz, R., Bernhart, S. H., Höner Zu Siederdissen, C., Tafer, H., Flamm, C., Stadler, P. F., and Hofacker, I. L. (November, 2011) ViennaRNA Package 2.0. *Algorithms Mol. Biol.,* **6**, 26.
8. Mathews, D. H., Andre, T. C., Kim, J., Turner, D. H., and Zuker, M. An Updated Recursive Algorithm for RNA Secondary Structure Prediction with Improved Thermodynamic Parameters. (1997).
9. Reuter, J. S. and Mathews, D. H. (March, 2010) RNAstructure: software for RNA secondary structure prediction and analysis. *BMC Bioinformatics,* **11**, 129.
10. Do, C. B., Woods, D. A., and Batzoglou, S. (July, 2006) CONTRAfold: RNA secondary structure prediction without physics-based models. *Bioinformatics,* **22**(14), e90–8.
11. Foo, C.-S., Do, C. B., and Ng, A. Y. (2008) Efficient multiple hyperparameter learning for log-linear models. In Platt, J. C., Koller, D., Singer, Y., and Roweis, S. T., (eds.), *Advances in Neural Information Processing Systems 20*, pp. 377–384 Curran Associates, Inc.
12. Zakov, S., Goldberg, Y., Elhadad, M., and Ziv-Ukelson, M. (November, 2011) Rich parameterization improves RNA structure prediction. *J. Comput. Biol.,* **18**(11), 1525–1542.
13. Rivas, E., Lang, R., and Eddy, S. R. (February, 2012) A range of complex probabilistic models for RNA secondary structure prediction that includes the nearest-neighbor model and more. *RNA,* **18**(2), 193–212.
14. Darty, K., Denise, A., and Ponty, Y. (August, 2009) VARNA: Interactive drawing and editing of the RNA secondary structure. *Bioinformatics,* **25**(15), 1974–1975.
15. Andronescu, M., Condon, A., Hoos, H. H., Mathews, D. H., and Murphy, K. P. (July, 2007) Efficient parameter estimation for RNA secondary structure prediction. *Bioinformatics,* **23**(13), i19–28.
16. Andronescu, M., Condon, A., Hoos, H. H., Mathews, D. H., and Murphy, K. P. (December, 2010) Computational approaches for RNA energy parameter estimation. *RNA,* **16**(12), 2304–2318.
17. Akiyama, M., Sato, K., and Sakakibara, Y. (December, 2018) A max-margin training of RNA secondary structure prediction integrated with the thermodynamic model. *J. Bioinform. Comput. Biol.,* **16**(6), 1840025.
18. Singh, J., Hanson, J., Paliwal, K., and Zhou, Y. (November, 2019) RNA secondary structure prediction using an ensemble of two-dimensional deep neural networks and transfer learning. *Nat. Commun.,* **10**(1), 5407.
19. Chen, X., Li, Y., Umarov, R., Gao, X., and Song, L. (March, 2020) RNA Secondary Structure Prediction By Learning Unrolled Algorithms. In *Proceedings of the 8th International Conference on Learning Representations*.
20. Ba, J. L., Kiros, J. R., and Hinton, G. E. Layer normalization. arXiv preprint arXiv:1607.06450 [stat.ML] (2016).
21. Barron, J. T. Continuously Differentiable Exponential Linear Units. arXiv preprint arXiv:1704.07483 [cs.LG] (April, 2017).
22. Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. R. Improving neural networks by preventing co-adaptation of feature detectors. arXiv preprint arXiv:1207.0580 [cs.NE] (July, 2012).
23. He, K., Zhang, X., Ren, S., and Sun, J. Deep Residual Learning for Image Recognition. arXiv preprint arXiv:1512.03385 [cs.CV] (December, 2015).
24. Hochreiter, S. and Schmidhuber, J. (November, 1997) Long short-term memory. *Neural Comput.,* **9**(8), 1735–1780.
25. Tinoco, Jr, I., Borer, P. N., Dengler, B., Levin, M. D., Uhlenbeck, O. C., Crothers, D. M., and Bralla, J. (November, 1973) Improved estimation of secondary structure in ribonucleic acids. *Nat. New Biol.,* **246**(150), 40–41.
26. Tsochantaridis, I., Joachims, T., Hofmann, T., and Altun, Y. (2005) Large Margin Methods for Structured and Interdependent Output Variables. *J. Mach. Learn. Res.,* **6**(Sep), 1453–1484.
27. Kingma, D. P. and Ba, J. (2015) Adam: A method for stochastic optimization. In *Proceedings of the 3rd International Conference for Learning Representations*.
28. Adam, P., Sam, G., Soumith, C., Gregory, C., Edward, Y., Zachary, D., Zeming, L., Alban, D., Luca, A., and Adam, L. (2017) Automatic differentiation in PyTorch. In *Proceedings of Neural Information Processing Systems*.
29. Steinberg, S., Misch, A., and Sprinzl, M. (July, 1993) Compilation of tRNA sequences and sequences of tRNA genes. *Nucleic Acids Res.,* **21**(13), 3011–3015.
30. Dowell, R. D. and Eddy, S. R. (June, 2004) Evaluation of several lightweight stochastic context-free grammars for RNA secondary structure prediction. *BMC Bioinformatics,* **5**, 71.
31. Lu, Z. J., Gloor, J. W., and Mathews, D. H. (October, 2009) Improved RNA secondary structure prediction by maximizing expected pair accuracy. *RNA,* **15**(10), 1805–1813.
32. Gardner, P. P., Daub, J., Tate, J., Moore, B. L., Osuch, I. H., Griffiths-Jones, S., Finn, R. D., Nawrocki, E. P., Kolbe, D. L., Eddy, S. R., and Bateman, A. Rfam: Wikipedia, clans and the "decimal" release. (2011).
33. Nawrocki, E. P., Burge, S. W., Bateman, A., Daub, J., Eberhardt, R. Y., Eddy, S. R., Floden, E. W., Gardner, P. P., Jones, T. A., Tate, J., and Finn, R. D. (January, 2015) Rfam 12.0: updates to the RNA families database. *Nucleic Acids Res.,* **43**(Database issue), D130–7.
34. Kalvari, I., Argasinska, J., Quinones-Olvera, N., Nawrocki, E. P., Rivas, E., Eddy, S. R., Bateman, A., Finn, R. D., and Petrov, A. I. (January, 2018) Rfam 13.0: shifting to a genome-centric resource for non-coding RNA families. *Nucleic Acids Res.,* **46**(D1), D335–D342.
35. Weinberg, Z., Lünse, C. E., Corbino, K. A., Ames, T. D., Nelson, J. W., Roth, A., Perkins, K. R., Sherlock, M. E., and Breaker, R. R. (October, 2017) Detection of 224 candidate structured RNAs by comparative analysis of specific subsets of intergenic regions. *Nucleic Acids Res.,* **45**(18), 10811–10823.
36. Fu, L., Niu, B., Zhu, Z., Wu, S., and Li, W. (December, 2012) CD-HIT: accelerated for clustering the next-generation sequencing data. *Bioinformatics,* **28**(23), 3150–3152.
37. 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 Res.,* **45**(20), 11570–11581.
38. Sloma, M. F. and Mathews, D. H. (December, 2016) Exact calculation of loop formation probability identifies folding motifs in RNA secondary structures. *RNA,* **22**(12), 1808–1818.
39. Hamada, M., Kiryu, H., Sato, K., Mituyama, T., and Asai, K. (2009) Prediction of RNA secondary structure using generalized centroid estimators. *Bioinformatics,* **25**(4), 465–473.
40. Sato, K., Hamada, M., Asai, K., and Mituyama, T. (2009) CentroidFold: A web server for RNA secondary structure prediction. *Nucleic Acids Res.,* **37**(SUPPL. 2), 277–280.
41. Nawrocki, E. P. and Eddy, S. R. (November, 2013) Infernal 1.1: 100-fold faster RNA homology searches. *Bioinformatics,* **29**(22), 2933–2935.

cture(0,0)(-30,0)10 (-30,-5)(0,1)10 (-35,0)(1,0)30 (0,-30)10 (-5,-30)(1,0)10 (0,-35)(0,1)30 picturepicture(0,0)(30,0)10 (30,-5)(0,1)10 (35,0)(-1,0)30 (0,