SQUARNA - an RNA secondary structure prediction method based on a greedy stem formation model

Davyd R. Bohdan[1], Grigory I. Nikolaev[1], Janusz M. Bujnicki[1], Eugene F. Baulin[1,*]

1. International Institute of Molecular and Cell Biology in Warsaw, Warsaw, Poland;

**corresponding author: ebaulin@iimcb.gov.pl, efbaulin@gmail.com*

ABSTRACT

Non-coding RNAs play a great variety of roles in many cellular processes with their spatial structure known to dictate the functioning. RNA secondary structure largely determines the molecule's global fold, which together with the paucity of experimentally determined RNA 3D structures makes its knowledge crucial for determining the function of the molecule. Currently, there is no one good solution for de novo RNA secondary structure prediction, with the existing methods getting more and more complicated without substantial progress in accuracy and being subject to drastic limitations such as ignoring pseudoknots. In this work, we present SQUARNA, a new approach for de novo RNA secondary structure prediction based on a straightforward greedy stem formation model overcoming many limitations of the existing tools. The benchmarks show that SQUARNA is on par with state-of-the-art methods for a single sequence input and significantly outperforms existing tools for a sequence alignment input.

MAIN

Hundreds of structured RNA molecules have been discovered and described to the present day [1], ranging from ligand-binding riboswitches regulating gene expression [2] to structured elements of viral RNAs making the molecule resistant to degradation [3]. The functioning of these RNAs is dictated by their spatial structure [4] which is in turn determined by the sequence of ribonucleotides and the environment conditions [5]. According to the widely accepted view, the RNA secondary structure (a set of Watson-Crick and wobble base pairs) forms first with all other interactions forming on top of it [6]. Thus, the RNA secondary structure dictates the molecule's global fold, which makes its knowledge crucial for determining the 3D structure and the function of

the molecule [7]. Despite recent great advancements in the experimental techniques of RNA structure determination, particularly in chemical probing methods [8] and cryo-electron microscopy (cryo-EM) [9], structures of many functional RNAs remain unknown [10]. Therefore, the computational problem of RNA secondary structure prediction remains highly relevant.

De novo RNA secondary structure prediction problem is formulated as the prediction of which pairs of nucleotides will form Watson-Crick and wobble base pairs from either a single RNA sequence input or a multiple sequence alignment input. Currently, one of the most popular approaches for single-sequence prediction is free energy minimization [11]. Here, the free energy of the structure is represented as the sum of the free energies of its elements, and the structure of the minimum free energy (MFE structure) is identified via dynamic programming [12]. Alternatively, some methods employ probabilistic approaches and attempt to predict the structure of maximum expected accuracy (MEA structure) [13]. The classical approach for the alignment-input prediction is based on covariation analysis and attempts to derive the evolutionary conserved secondary structure by identifying the most covarying pairs of residue positions as conserved base pairs [14]. In a number of methods, the covariation analysis is coupled with an algorithm searching for either the MFE structure [15] or the MEA structure [13, 14, 16]. Recently, several deep learning methods were developed for both single-sequence and alignment-based predictions and were shown to outperform the conventional tools [17-19].

Despite the variety of different approaches, to the moment there is still no one good solution for the problem, especially when the input is a single RNA sequence or a sequence alignment of too few or too similar RNA sequences insufficient for covariation analysis [20]. The RNA secondary structure prediction methods are getting more and more complicated without substantial progress in accuracy [19]. Most methods ignore pseudoknots [16, 17, 21], which is a horrible simplification [22]. Also, most methods attempt to predict the single best structure and are not suitable for RNA molecules folding into alternating structures [13, 17, 18, 21], with just a few exceptions [21, 23]. Many tools do not allow to incorporate structural restraints and chemical probing data to guide the predictions [13, 14, 16-18]. A very limited number of tools predict the secondary structure formed by several RNA sequences [24]. Newly appearing deep

learning methods are prone to overfitting, lack interpretability, and do not generalize well to unfamiliar structured RNA families [25, 26].

In this work, we present SQUARNA, a new RNA secondary structure prediction method based on a greedy stem formation model that overcomes the mentioned limitations of the existing approaches. Benchmarks show that SQUARNA is on par with state-of-the-art tools for a single sequence input and significantly outperforms the existing methods for a sequence alignment input.

RESULTS

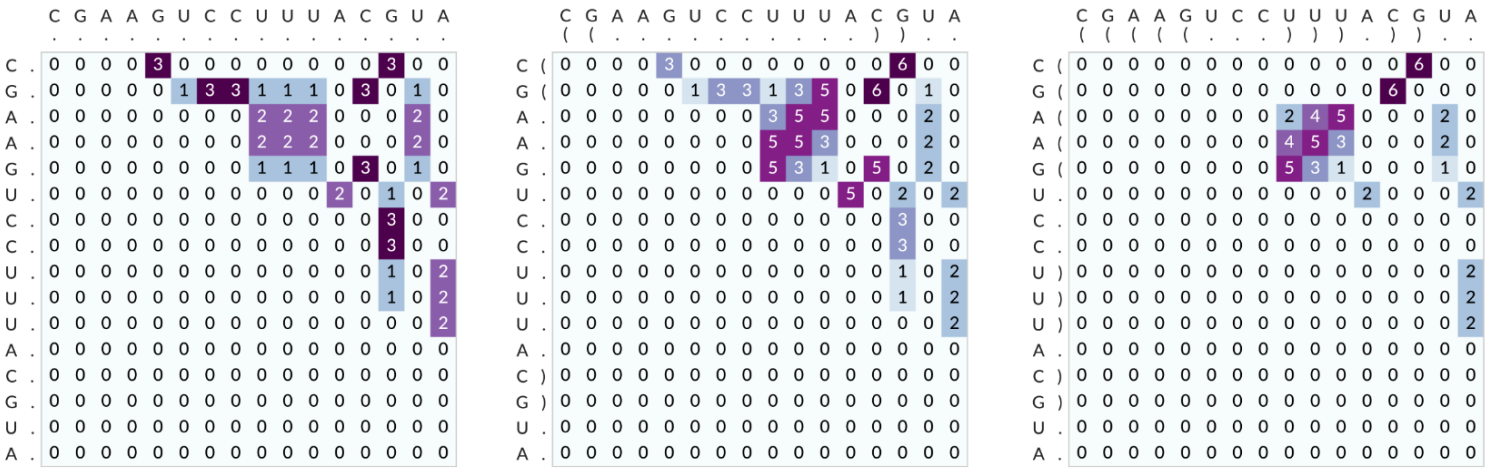
SQUARNA algorithm for the single-sequence prediction

We formulated the problem of de novo RNA secondary structure prediction based on a single sequence input as a variant of the partial assignment problem (https://en.wikipedia.org/wiki/Assignment_problem), where we have two sets of elements as input and an optimal matching between some of the elements of the two sets as output. To solve the given problem we designed a greedy stem formation model, see the Methods section for details.

We consider the input sequence of ribonucleotides as each of the two sets of elements simultaneously and predict the canonical (Watson-Crick and wobble) base pairs as the matches between the elements of the two sets (Figure 1A, left). The fact that the two sets represent the same RNA sequence and that RNA does not form hairpins shorter than three residues [27] limit us to finding (i, j) base pairs where $j \geq i + 4$. Given that the base pairs are not independent and the formation of base pair (i, j) is often accompanied by the formation of base pair $(i + 1, j - 1)$ and/or $(i - 1, j + 1)$, we used the scores of stems (consecutive stretches of canonical base pairs) as the weights of the potential matchings that guide the predictions (Figure 1A, middle). We defined the stem *bp-score* as the sum of the scores of its base pairs. For the prediction, we employed a greedy 2-step approach named SQUARNA (SQUAre RNA), where we first select the match (i, j) of the highest score, mark it as a base pair, and then recalculate the stem bp-scores with all the base pairs $(i, k \neq j)$ and $(m \neq i, j)$ removed. The 2-step procedure is iteratively repeated until we run out of matches above the defined threshold.

Since all the matches of a given stem have the same weight, we reformulated the approach into an equivalent and more straightforward form with stems being the unit of prediction. In this form, we annotate all possible stems, score them, select the stem with the highest score, remove all the base pairs inconsistent with the chosen stem, and iterate (Figure 1A, middle and right). To better reflect the nature of RNA secondary structure [27] we used adjusted stem scores (or just *stem scores*) instead of bp-scores. The stem score is the product of its bp-score and three factors dependent on the already-predicted stems (Figure 1B): distance factor (dependent on the number of residues confined by the stem), pseudoknot factor (dependent on the number of different pseudoknot levels having their stems intersecting the stem), and loop factor (the bonus for either GNRA tetraloop or inner or outer short near-symmetric internal loop). If we face other stems inconsistent with the highest-score stem that have a score within the defined range from the highest score we bifurcate and continue the predictions separately for each such stem being added to the current secondary structure. As a result, SQUARNA then returns a number of alternative secondary structures ranked based on their stem bp-scores and/or reactivity scores (Figure 1C). Also, SQUARNA can be run with several different parameter sets (reflecting e.g., different folding conditions) with the following aggregation and unified ranking of the predicted structures.

A



B

```
..(((.......(((((.....)))))).[[......)))].]]
.....((((.......................))))......
```

$$\text{score} = \text{bp-score} * F_{distance} * F_{order} * F_{loop}$$

$$\text{bp-score} = \sum_{XY \in base\ pairs} Number_{XY} * Weight_{XY}$$

$$F_{distance} = \left(\frac{1}{1 + |Number_{dots} + Number_{brackets} * Weight_{brackets} - Distance_{ideal}|}\right)^{Coefficient_{distance}}$$

$$F_{order} = \left(\frac{1}{1 + Order}\right)^{Penalty_{order}}$$

$$F_{loop} = (1 + Bonus_{GNRA} * GNRA) * (1 + Bonus_{inner\ loop} * inner\ loop) * (1 + Bonus_{outer\ loop} * outer\ loop)$$

C

```
..(((((((...(((((.....)))))).[[..)))))..))).]] #1
..(((((((...(((((.....)))))..))))).....)))... #2
```

$$score_{structure} = \sum_{stems} (bp_score_{stem})^{Power_{rank}}$$

$$score_{reactivity} = \frac{\sum_{i \leq Length} \begin{cases} Reactivity_i & \text{if } Base_i \text{ is paired} \\ 1 - Reactivity_i & \text{if } Base_i \text{ is unpaired} \end{cases}}{Length}$$

***Figure 1**. Details of the single-sequence SQUARNA algorithm. (A) demonstration example of the partial assignment problem being solved. Left: a square matrix of base pair weights (GC = 3, AU = 2, GU = 1). Middle: the square matrix of stem bp-scores, the stem consisting of the two GC pairs has the highest score; we select the stem and remove the base pairs inconsistent with it. Right: the square matrix of stem bp-scores recalculated after the base pair removal, the stem AAG-UUU has the highest score among the remaining stems; we select it and stop the procedure due to having zero other stems of at least two base pairs consistent with the two predicted stems. (B) The stem scoring formula used in SQUARNA. The first dot-bracket line shows the already-predicted stems, and the second dot-bracket line shows the current candidate stem. (C) The structure scoring formulas used for ranking in SQUARNA.*

For the analysis of SQUARNA performance, we manually compiled the SRtrain dataset of 274 RNA secondary structures derived from the PDB data bank [28]. To train the parameters of the single-sequence SQUARNA algorithm we used the subset SRtrain150 of 266 RNA sequences ≤ 150 nucleotides in length. In the training procedure, we maximized the harmonized F-score metric - the harmonic mean of the mean F-score among the sequences and the total F-score calculated for the total number of base pairs in the dataset. We used the harmonized F-score to balance the contribution of RNA sequences of different lengths and make the comparisons less dependent on the length distribution of the dataset. For an independent evaluation of SQUARNA performance, we used the TS1reducedWC dataset of 62 RNA sequences described in [18].

We compared the single-sequence SQUARNA with five state-of-the-art tools (according to recent benchmarks [18, 19]): three conventional tools (RNAfold [21], IPknot [13], and ShapeKnots [23]), and the two most efficient deep-learning-based tools MXfold2 [17] and SPOT-RNA [18] (Supplementary Table S1). The best performance of SQUARNA was achieved with two parameter sets (SQUARNA default), but we also included the best single parameter set in the benchmarking (SQUARNA alternative). Since unlike the other tools SQUARNA is able to return ranked alternative secondary structures, in the benchmarks we measured its performance separately based on the single top-ranked structure (top-1) or based on the best of the top-5 structures (top-5). The consideration of the top-5 structures is justified by the fact that the most popular RNA 3D structure prediction contests RNA-Puzzles [29] and CASP [30] accept up to

five models for each of the targets. For more details on the datasets and performed benchmarks see the Methods section.

The best performance on the TS1reducedWC was demonstrated by SPOT-RNA (all three F-scores of 82%), whereas on the SRtrain dataset, MXfold2 showed the best harmonized and total F-scores (79% and 76%) and the best mean F-score of 85% was achieved by SQUARNA-default-top-5 (Figure 2, Supplementary Table S2). However, on the TS1reducedWC dataset, the only statistically significant difference in the performance was identified between SPOT-RNA and ShapeKnots, and on the SRtrain dataset the statistically significant differences were identified between but not within the two groups - (i) RNAfold, IPknot, ShapeKnots, SQUARNA-default-top-1, SQUARNA-alternative-top-1, and (ii) MXfold2, SPOT-RNA, SQUARNA-default-top-5, SQUARNA-alternative-top-5 (Figure 2, Supplementary Table S3).

Thus, with a much simpler method SQUARNA could achieve the level of the popular conventional tools if we consider its top-1 predicted structures, and if we consider the best of the top-5 predicted structures SQUARNA reaches the level of the best deep learning approaches.

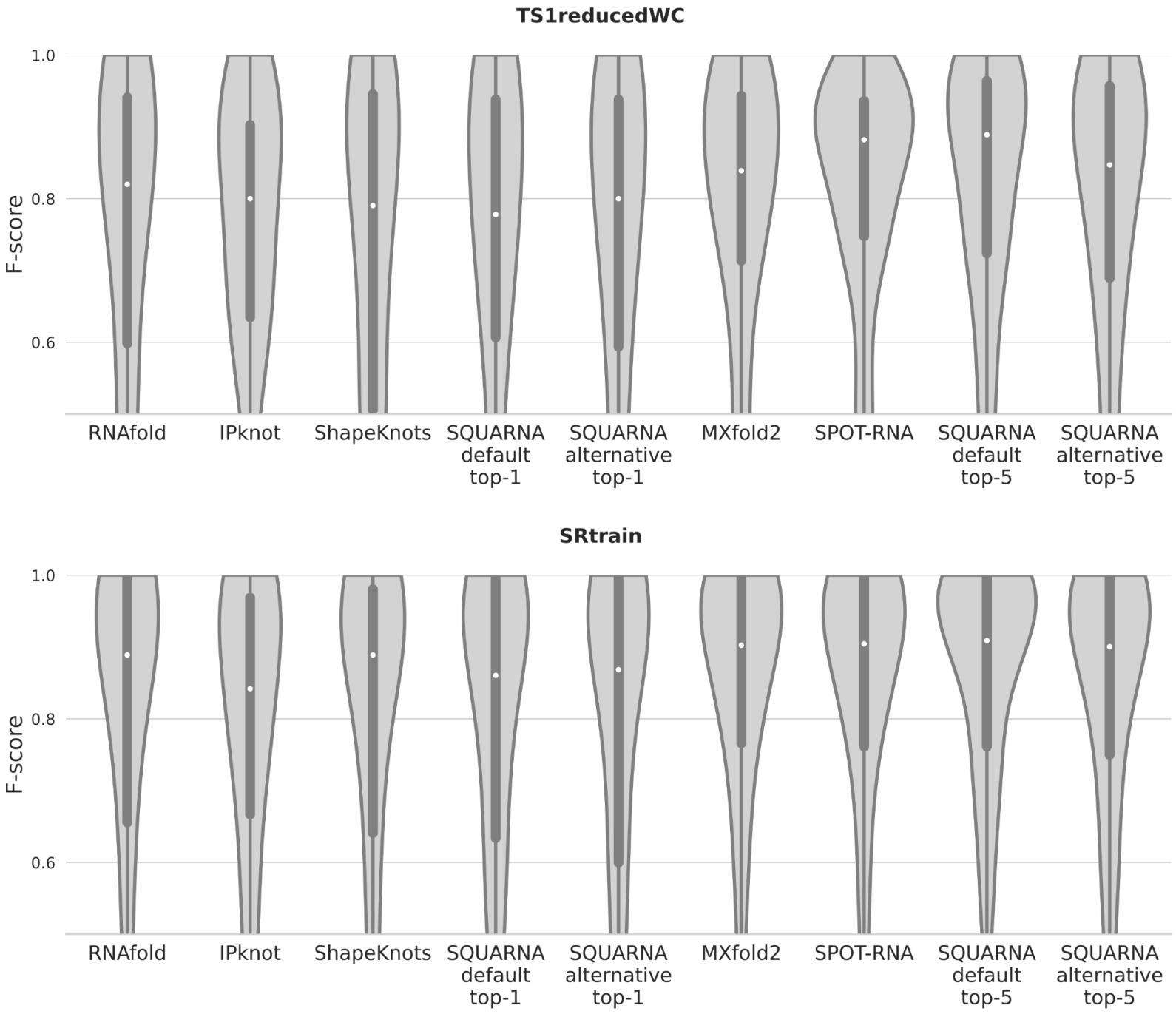


***Figure 2**. Single-sequence benchmark results. The violin plots represent F-score distributions resulting from the predictions of the four SQUARNA settings (default/alternative + top-1/top-5) and five state-of-the-art tools (RNAfold, IPknot, ShapeKnots, MXfold2, SPOT-RNA) on the two datasets of RNA sequences (TS1reducedWC of 62 sequences and SRtrain of 274 sequences). Boxplots are provided within the violins, and the white dots mark median points. The graph was*

prepared using the Seaborn Python library
(https://seaborn.pydata.org/generated/seaborn.violinplot.html).

SQUARNA algorithm for the alignment-based prediction

For the de novo RNA secondary structure prediction problem based on a multiple sequence alignment input, we designed a 3-step procedure largely utilizing the ideas of the single-sequence SQUARNA algorithm. Although, given that the demarcations of a stem are not clear in the alignment due to gaps, we used a base pair as a unit of prediction, see the Methods section for details.

Step 1 consists of two iterations (Figure 3). On the first iteration, we calculate matrices of stem bp-scores for each sequence in the alignment (Figure 3A), sum them up to obtain the total stem bp-score matrix (Figure 3B), and then greedily select a subset of consistent base pairs having the highest scores above the defined threshold. On the second iteration, the same procedure is performed but with the base pairs inconsistent with the subset selected at iteration 1 being ignored (Figure 3C). The second iteration serves to clean the predicted structure from mixes of base pairs belonging to mutually inconsistent stems of comparable scores. The set of base pairs selected after iteration 2 is considered the predicted structure after step 1 (SQUARNA step1).

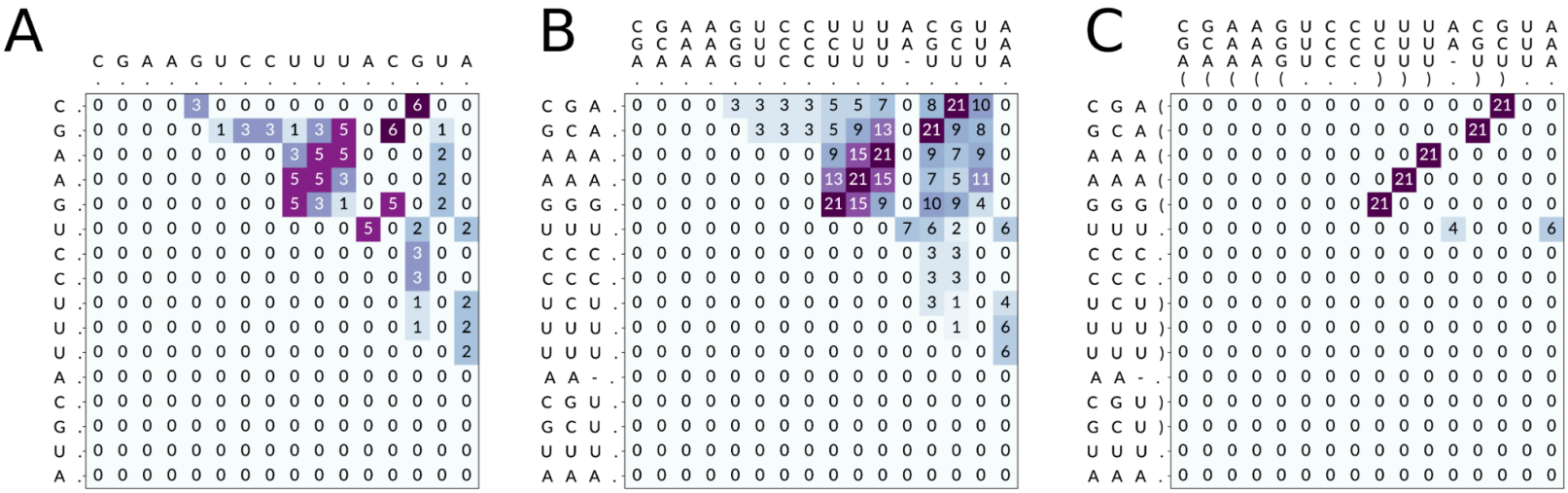


***Figure 3**. Demonstration example of step 1 of the alignment-based SQUARNA algorithm. (A) a square matrix of stem bp-scores (GC = 3, AU = 2, GU = 1) obtained*

from a single sequence. (B) The total stem bp-score matrix obtained from the sequence alignment at iteration 1; only the cells of score 21 are selected for the predicted structure, given that the score threshold is higher than 7. (C) The total stem bp-score matrix obtained from the sequence alignment at iteration 2; the structure obtained at iteration 1 and serving as structural restraints is shown in the dot-bracket line. Here, the predicted structure after iteration 2 is the same as the one after iteration 1.

In step 2 we run single-sequence SQUARNA predictions guided by the total stem bp-score matrix obtained at iteration 1 of step 1. The consensus of the predicted individual structures is then considered the predicted secondary structure after step 2 (SQUARNA step2). Step 2 serves two purposes: (i) to derive frequently predicted base pairs consistent with the step 1 structure but not perfectly conserved throughout the alignment and (ii) to identify potential alternative folds predicted for subsets of the sequences. In step 3 SQUARNA returns either the union of step 1 and step 2 base pairs (SQUARNA step3u) or their intersection (SQUARNA step3i).

To train the parameters of the alignment-based SQUARNA we compiled the RNAStralignExt dataset (Supplementary Table S4) - the set of seed alignments from Rfam [10] of eleven non-coding RNA families, including eight families from the RNAStralign dataset [31], and three additional families with well known pseudoknotted secondary structures (HDV ribozyme, SAM riboswitch, and Twister ribozyme). For an independent evaluation of SQUARNA performance, we used the full set of Rfam seed alignments (Rfam14.9, 4108 families) and the subset of Rfam families with known 3D structure (RfamPDB, 134 families).

We compared the alignment-based SQUARNA with four state-of-the-art tools: CentroidAlifold [16], RNAalifold [15], IPknot [13], and Rscape (CaCoFold) [14] (Supplementary Table S1). For Rscape we considered separately the nested part of the predicted structure (Rscape nested) and the total predicted structure (Rscape total). As the alignment-based predictions highly depend on the alignment depth (number of sequences), we analyzed the results with different depth thresholds (Supplementary Figure S1). We identified that Rscape starts to outperform other state-of-the-art tools at a depth threshold of around 100 sequences, and only closer to 1000 sequences Rscape catches up with SQUARNA which outperforms the other tools on the range between 100 and 1000 sequences. Surprisingly, the results of CentroidAlifold and RNAalifold

almost did not grow with the increase of depth threshold, and the IPknot results even decreased.

To avoid bias from the alignment depth distribution among Rfam families, we compiled the SubAli dataset with ten randomly sampled alignments for each of eleven different depths from 2 to 300 sequences for each of the eleven families from the RNAStralignExt set. On the SubAli dataset, only SQUARNA and Rscape demonstrated growth with the depth threshold increase (Supplementary Figure S1), and SQUARNA significantly outperformed all the other tools (Supplementary Table S6). We then further analyzed the results for each of the alignment depths separately (Figure 4). CentroidAlifold and RNAalifold started from a mean F-score of 60% on the depth of two sequences and stabilized on five sequences with a mean F-score of around 75% (Figure 4A, D). Rscape demonstrated a sharp rise from 30% to 80-85% mean F-score on the range from 2 to 100 sequences and then stabilized (Figure 4B, D). SQUARNA started with a mean F-score of 50% on 2 sequences, reached 80% on 10 sequences, and stabilized at 90% starting from 100 sequences (Figure 4C, D).

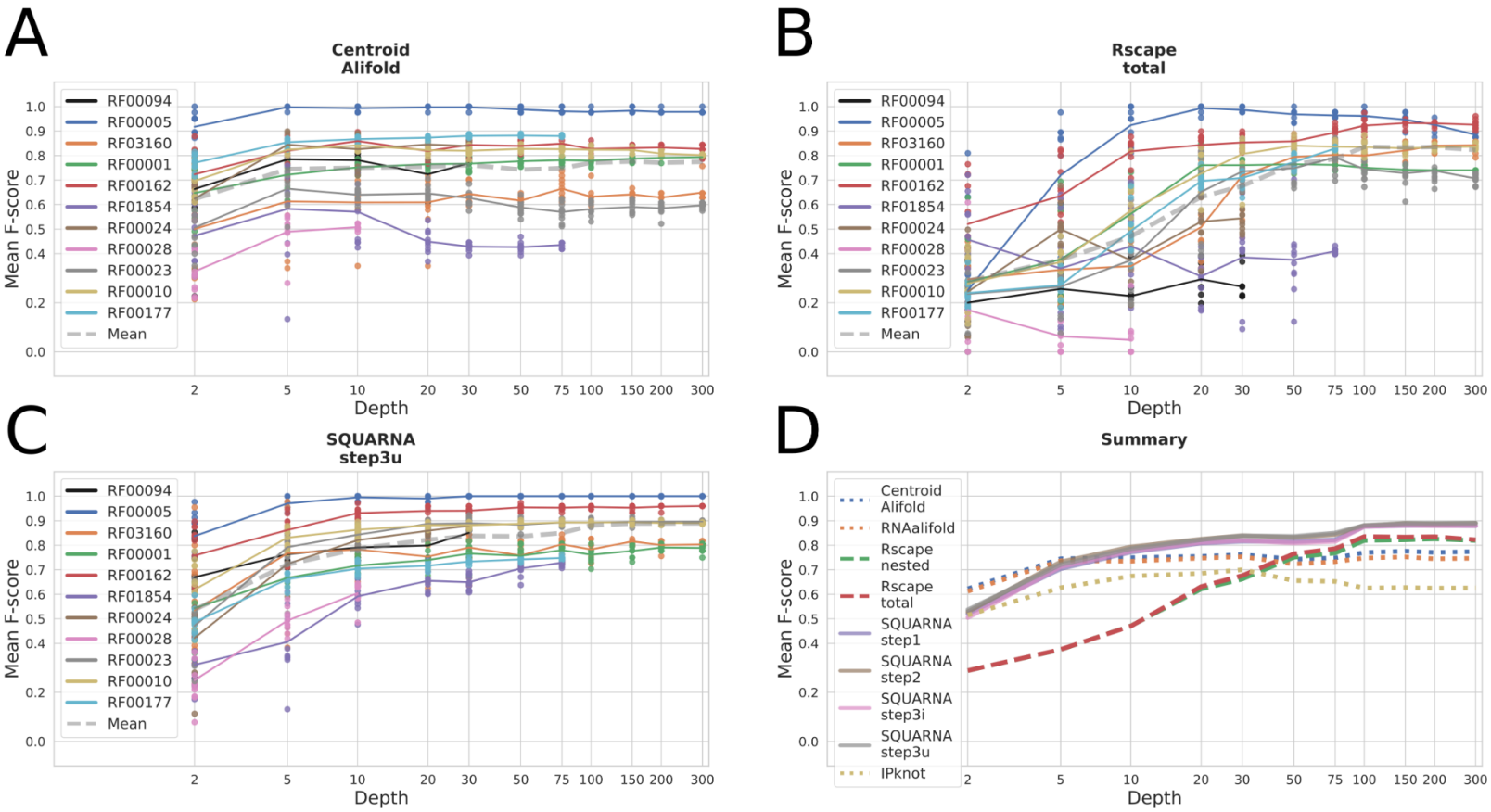


***Figure 4**. Analysis of the alignment-based prediction quality dependence from the alignment depth. (A, B, C) Results of three different tools (CentroidAlifold, Rscape total, SQUARNA step3u). Each dot is the F-score of a single prediction, solid lines represent the mean F-scores among the ten alignments of each depth for a particular Rfam family. The dashed gray line shows the mean F-scores among the eleven Rfam families. (D) Mean F-scores for each depth for each of the alignment input tools used.*

To analyze the dependence of the alignment-based predictions on the sequence similarity we randomly sampled alignments of different mean pairwise sequence similarity for the eleven families for each of the three depths of 5, 10, and 20 sequences. Interestingly, only SQUARNA demonstrated a small but clear negative correlation between the prediction quality and the mean pairwise sequence similarity in the alignment, which decreased with the increase of the alignment depth (Supplementary Figure S2).

We then looked into examples of SQUARNA fails on the deepest alignments of the Rfam14.9 dataset. On the algC RNA motif (RF02929) SQUARNA missed the only correct stem and predicted a two-stem pseudoknot instead which resulted in an F-score of 0.0 despite the alignment depth of 492 sequences (Figure 5A). The inspection of the sequences showed that the alignment includes only 90 unique sequences, with 3′-half of the sequences showing unusually high sequence conservation. This made SQUARNA identify the pseudoknot despite the correct stem being the only one to show base pair covariation. However, SQUARNA was able to derive the correct stem from shallow alignments of 5 to 40 most diverse sequences of the family. Moreover, the correct stem can be spotted in the SQUARNA verbose output on the entire alignment (Figure 5A).

Another example is the ribosomal S15 leader sequence (RF00114), where SQUARNA showed an F-score of only 35% on the alignment of 78 sequences (Figure 5B). The inspection of the family showed that the S15 leader sequence is known to fold into two alternative structures, one with consecutive hairpins and another with a pseudoknot [32]. The Rfam structure includes the two hairpins, whereas SQUARNA was the only tool to identify the correct pseudoknot. Moreover, the second hairpin can be spotted in the SQUARNA tool verbose output (Figure 5B).

A

```
RF02929
........(((((.........)))))............................ Rfam structure
.(((..((((((............[[[[.)))))))...)))(]]]].....).... SQUARNA-step3u
.......(((((...........)))))(.(((........))))............ Rscape-Total
......((((((.................))))))....(((((.......))))). IPknot
......(((((((.......)))))))).(((........)))............. SQUARNA-step3u, Depth 5
.......((((((.........))))))..(((........)))............. SQUARNA-step3u, Depth 10
........(((((.........)))))((.(.(........))))............ SQUARNA-step3u, Depth 20
........(((((.........)))))((.(.(........))))............ SQUARNA-step3u, Depth 40
......((((((............[[[[.)))))).......]]]]........... SQUARNA-step3u, Depth 80
>Step 1, Iteration 1
>Conserved base pairs (one by one)
..........(....................)......................... 6703.75
...........(..................)......................... 6703.75
........(......................)........................ 6698.5
.......(........................)....................... 6624.25
.........(......................)........................ 6620.0
......(........................)....................... 6116.0
.........................(..................)............ 5790.75
...........................(..............)............. 5705.75
.............................(...........)............... 5698.75
..........................(...............)............. 5683.0
.........................(...................)........... 5593.25
.........(...............)............................... 5362.0
..........(.............)................................ 5362.0
...........(...........)................................ 5362.0
........(..................)............................ 5353.5
.......(...................)........................... 5095.25
......(....................)............................ 4754.5
>Conserved base pairs (assembled)
......(((((((..[......)..[[[[.)))))))...((((]]]]].....).)))
......((((((....[.....])))))).(.(........))((]]...).).....
>URS0000D69DA4_596155/1-51
ACGAUGCUGGCC--AUUUAC---GGUCAG-GGCUAACUAUUGCCUGAAGAA-GGGUAA
......((((((...........)))))).(((........)))..............
>URS0000D65A94_913069/1-51
ACGAUGUUGGCC--AUUUAC---GGUCAG-GGCUAACUGUUGCCUGAAGAA-AGGUAA
.....(((((((............[[[[.)))))))..((((]]]].......))))
```

B

```
RF00114
......((((((............))))))........................(.((.........((((.............)))))))................................................. Rfam structure
..((((((((((.[[.....[[[[[])))))))))).................................................]]]]]]]..............(((..........)))..........((...)).. SQUARNA-step3u
.((((((((((.............))))))))))..................................................................................................... Rscape-Total
..((((((((((.............)))))))))).................................(((((.............)))))................................................. IPknot
>Step 1, Iteration 1
>Conserved base pairs (one by one)
.......(....................)........................................................................................................... 924.0
........(...................)........................................................................................................... 924.0
.........(.................)............................................................................................................ 908.5
..........(...............)............................................................................................................ 884.0
...........(.............)............................................................................................................ 847.0
......(......................)............................................................................................................. 786.75
.....................(.......................................................................)...................................... 740.25
....................(........................................................................)..................................... 738.25
...................(..........................................................................).................................... 735.75
.....................(.......................................................................)...................................... 732.5
....................(........................................................................)..................................... 731.75
....................(........................................................................)..................................... 731.75
.............(.............................................................................)............................................ 721.5
.............(.............................................................................)............................................ 721.5
.......................................................................(..................)..................................... 388.5
.......................................................................(..............)....................................... 387.5
.....................................................................(....................)..................................... 380.75
>Conserved base pairs (assembled)
..((((((((((.[[.....[[[[[])))))))))).................................................]]]]]]]..............(((..........)))..................
.......[[[..............[((({<.................................((.(............].).))...............]]]...........).......}.))...>...
>CP000094.2/917832-917934
UGUUGAAACUGUUUU-----UGA-AACAGUUUCCUGAUAA-----------------AA---------GGAUUGCC----UCAUGGCUCUCGACGUUCAAGAAAAAGCUCAAAUCGUUGCCGACUACCAGCAAGCCGUU
....((((((((((........).)))))))))....................................((..((((........)))))).................((..........)).................
>AALF02000011.1/47337-47457
UUGGGUUGCUGAAUU-----AGAGAUCGGCGCCUGUUUUAAUUUA-UCUAUAUAUAU-UU---------GGAGUUGU---AUCAUGUCUCUAAGUGUUGAAGCGAAAGCUAAAAUUGUUGCUGACUUCGGUCGCGGUGCC
..(((..((((((.[[.....[[[[[]))))).)))...............................................]]]]]]]..........(((((((..........)))...))))...........
```

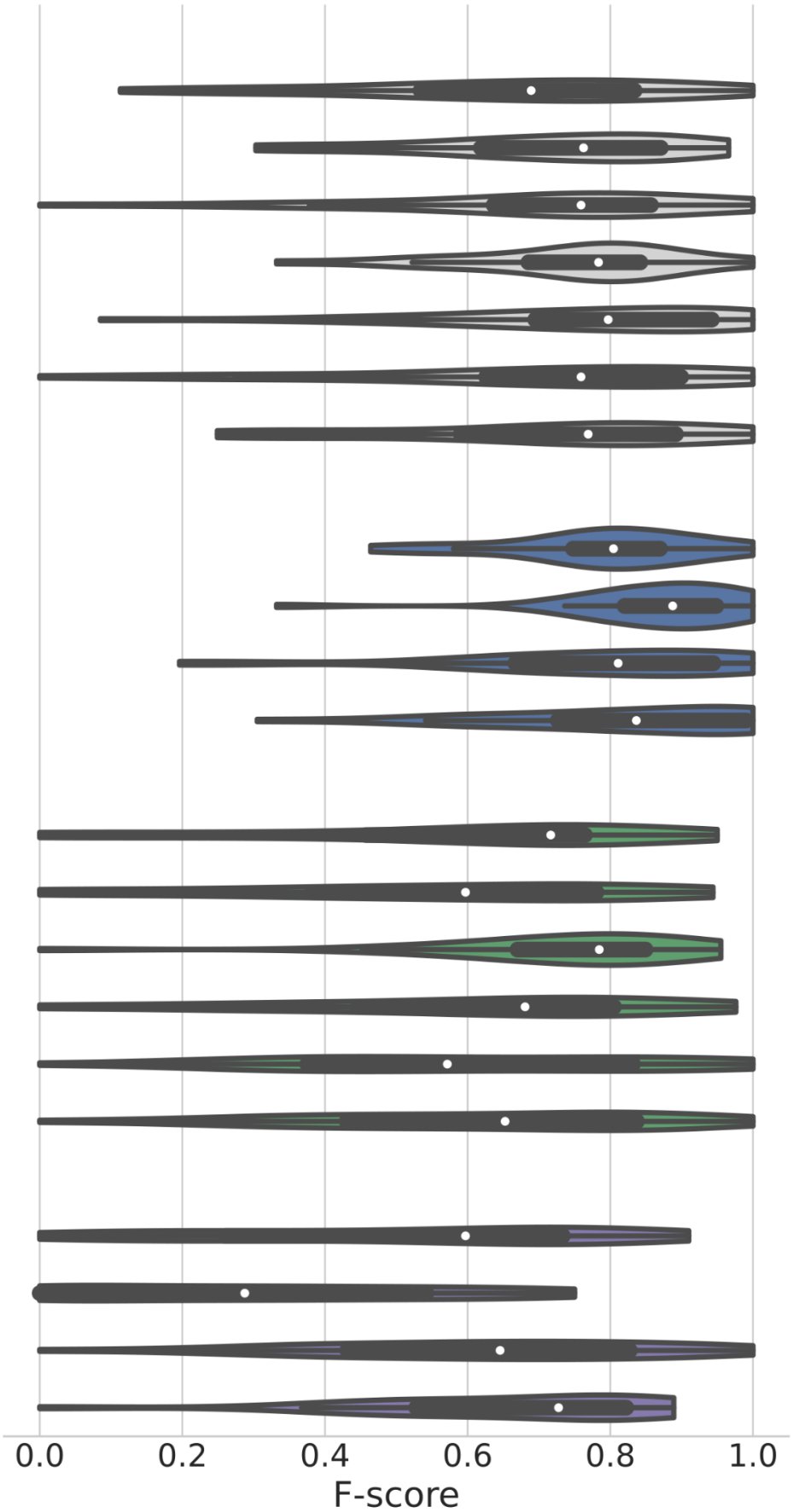
***Figure 5**. Selected examples of the alignment-based SQUARNA algorithm performance. (A) algC RNA motif, RF02929 Rfam family, seed alignment of 492*

sequences, (B) Ribosomal S15 leader sequence, RF00114 Rfam family, seed alignment of 78 sequences. The Rfam consensus structure and different predicted structures are shown in bold at the top of each panel and followed by fragments of the SQUARNA tool verbose output with individual sequence predictions at the bottom of each panel.

Chemical probing input data

To assess the impact of chemical probing input data on the prediction quality from the final user perspective we prepared the S01 dataset of 24 challenging RNAs [23] in four different settings: (i) RNA sequence only (sequence); (ii) RNA sequence and SHAPE data (sequence+shape); (iii) alignment of homologous sequences (alignment); (iv) alignment of homologous sequences and SHAPE data (alignment+shape). The multiple sequence alignments were prepared in a structure-unaware manner using BlastN [33] search for homologs followed by aligning the matches with MAFFT [34].

On sequence input, SQUARNA with “sk” parameter set (SQUARNAsk top-5) performed with a mean F-score of 71%, whereas on sequence+shape input it increased to 82% and was the second best tool after ShapeKnots (86%) (Figure 6). Notably, SQUARNA performed better for shorter sequences, showing the best mean F-score of 93% (SQUARNAsk top-5) for the 14 sequences under 200 nts with sequence+shape input, with ShapeKnots showing 90%. For alignment+shape input, SQUARNA performed 2.5-4.5% better than for alignment input, however, overall the prediction quality of the alignment-based tools was much poorer than those of the single-sequence tools due to the low quality of the alignments built in the structure-unaware manner.



	Mean F-score
sequence	
RNAfold	0.663
IPknot	0.718
ShapeKnots	0.705
MXfold2	0.752
SPOT-RNA	0.765
SQUARNA top-5	0.703
SQUARNAsk top-5	0.71
sequence + shape	
RNAfold	0.789
ShapeKnots	0.864
SQUARNA top-5	0.782
SQUARNAsk top-5	0.822
alignment	
CentroidAlifold	0.586
RNAalifold	0.549
IPknot	0.73
Rscape	0.608
SQUARNA step1	0.6
SQUARNA step2	0.614
alignment + shape	
RNAalifold	0.511
ShapeSorter	0.308
SQUARNA step1	0.624
SQUARNA step2	0.659

***Figure 6**. Benchmark results on the S01 dataset of 24 sequences. The violin plots represent F-score distributions resulting from the predictions with four different input types (sequence, sequence with shape data, alignment, alignment with shape data). Boxplots are provided within the violins, and the white dots mark median points. Mean F-scores are listed on the right side of the figure. The graph was prepared using the Seaborn Python library (https://seaborn.pydata.org/generated/seaborn.violinplot.html).*

DISCUSSION

In this work, we presented SQUARNA, a new approach to de novo RNA secondary structure prediction based on a stem formation model. The benchmarks show the single-sequence SQUARNA is on par with state-of-the-art methods, whereas the alignment-based SQUARNA significantly outperforms the existing tools.

The greedy approach used in SQUARNA is meant to reflect the RNA folding kinetics and, furthermore, it allows us to avoid global optimization procedures hindered by pseudoknots, e.g., free energy minimization. In turn, the use of stem scores instead of base pair scores enriches the “signal to noise ratio” that allows the alignment-based SQUARNA to reach the CentroidAlifold/RNAalifold baseline on as few as five sequences, whereas Rscape requires around 50 sequences to catch up with the baseline (Figure 4D). Notably, as many as 46.5% of the Rfam seed alignments consist of 5 to 50 sequences, whereas only 7.7% of the alignments include more than 50 sequences sufficient for a confident structure prediction by Rscape. Furthermore, SQUARNA step1 is the second fastest alignment-based tool after RNAalifold. On the other hand, the main limitation of SQUARNA is that, unlike Rscape, it can be misled by sequence conservation over covariation, which can be overcome by the curation of the input alignment.

SQUARNA achieved the level of the state-of-the-art deep-learning methods with a much simpler approach that does not bear the flaws characteristic of learning-based algorithms, e.g., overfitting, lack of interpretability, and poor generalization to unfamiliar data [25]. SQUARNA handles any natural or synthetic RNA sequence in the same way, and one can even trace the order in which the stems are being predicted. Furthermore, SQUARNA can easily process any type of modified RNA bases no matter whether frequent or rare [35], provided the user specifies the base pair weights for such bases.

The harmonized F-score metric used in this work for benchmarking showed some tools (in particular SPOT-RNA and SQUARNA) to perform better on shorter RNA sequences (Supplementary Table S2). This can be overcome in the future by designing separate SQUARNA settings for different RNA sequence length ranges.

SQUARNA can successfully utilize both RNA single sequence and sequence alignment inputs as well as chemical probing data (Figure 6). Furthermore, SQUARNA is able to catch alternating folds (Figure 5). Also, SQUARNA can take structural restraints as additional input and predict RNA secondary structure formed by several RNA sequences, allowing the user to analyze the ability of a given RNA to oligomerize. We believe the advantages of SQUARNA will have a substantial impact in the field of RNA computational biology.

METHODS

Theoretical background

The SQUARNA approach is rooted in the ARTEM algorithm for the superposition of two arbitrary RNA 3D structure fragments without a priori knowledge of the nucleotide matchings between the fragments [36]. The two core ideas of ARTEM are the formulation of the problem as the partial assignment problem and the utilization of the dependencies in the data. In the case of de novo RNA secondary structure prediction, the partial assignment relates to base pairs formed between the nucleotides and the main dependency in the data is the tendency of base pairs to form continuous stems.

At the early stage of preliminary analysis, we discovered that the best results were achieved with negative weights for GU base pairs, which leads to an ambiguity in how to define a stem - as the longest sequence of consecutive base pairs or as the subsequence of consecutive base pairs having the highest bp-score. The stems from the known RNA secondary structures were found to be almost always the stems of the maximum length, with a small number of cases with one flanking GU/AU base pair missing (data not shown). We tested all possible stem definitions, including the definition of a stem as any possible subsequence of consecutive base pairs. The results didn’t change significantly, so eventually we defined a stem as the longest possible sequence of consecutive base pairs as the fastest for the algorithm.

For the description of RNA secondary structure elements, such as loops and pseudoknots, we used the definitions from [37]. A pseudoknot order of RNA secondary structure was defined as the minimum number of different bracket types required for the non-ambiguous representation of the secondary structure in the dot-bracket format. An assignment of bracket types for the base pairs of a given structure was performed greedily and followed by a bubble-like sorting to minimize the number of base pairs of the higher orders, see the PairsToDBN function in the code: https://github.com/febos/SQUARNA/blob/main/SQRNdbnseq.py. Two base pairs *(i, j)* and *(k, l)* were considered inconsistent (intersecting) with each other if they cannot be formed simultaneously (i.e., if $i = k$ or $i = l$ or $j = k$ or $j = l$), and considered consistent (non-intersecting) otherwise. Two stems were considered consistent with each other if all of their base pairs were consistent with each other, otherwise, two stems were considered intersecting.

RNA sequence datasets

The SRtrain dataset was prepared as follows. RNA 3D structures from PDB [28] were selected according to a representative set of RNA structures [38] (version 3.279 with a 3.0 Å cutoff). RNA secondary structure was annotated with DSSR [39] (version 2.0). Then, using a custom Python script, we derived from the PDB files RNA strands meeting three conditions: (i) the strand is continuous, i.e. with no chain breaks or missing residues; (ii) the strand does not involve modified residues; (iii) the strand does not form Watson-Crick or wobble base pairs with any other strand, but forms at least one such base pair within the strand. The set of derived strands was subject to a manual secondary structure redundancy reduction procedure resulting in a non-redundant subset of 274 RNA strands with their secondary structures. The SRtrain150 subset included 266 strands of length shorter than 150 nts.

The TS1reducedWC dataset was prepared as follows. The set of 62 RNA sequences populating the TS1reduced dataset was derived from [18] and converted from .ct format to dot-bracket. Unfortunately, the dataset included a mix of canonical and non-canonical base pairs without any annotation of which is which. Therefore, we didn't use the original secondary structures but derived the structures for the 62 sequences with DSSR (version 2.0) to ensure the inclusion of Watson-Crick and wobble base pairs only. The resulting dataset was named TS1reducedWC.

The S01 dataset was prepared as follows. The set of 24 sequences along with SHAPE data were downloaded from [23] (https://webshare.oasis.unc.edu/weeksgroup/data-files/ShapeKnots_DATA.zip). The RNA secondary structure data were converted from .ct format to dot-bracket. The original SHAPE data included values ranging from 0.0 to 1.5 with a number of outliers, and missing values set to -999. For two of the sequences the number of reactivity values was inconsistent with the sequence length: (i) for the entry “5' domain of 16S rRNA, H. volcanii” we removed one excess value from the 3'-end; (ii) for the entry “SARS corona virus pseudoknot” we added three missing -999 values at the 3'-end. For the two fixes, we used the supplementary information from [23] as a guiding reference (https://www.pnas.org/doi/suppl/10.1073/pnas.1219988110/suppl_file/sd01.pdf).

For the prediction with ShapeKnots, we used the original SHAPE data (after the fixes). For the prediction with RNAfold and RNAalifold, we scaled the values to range from 0.0 to 2.0 (dividing the values by 1.5 and multiplying by 2.0) and truncated the outliers (values under 0.0 were set to 0.0 and values higher than 2.0 were set to 2.0), with the missing values kept as -999. For the prediction with SQUARNA, we analogously scaled the values to range from 0.0 to 1.0, truncated the outliers, and set the missing values to 0.5.

For each of the prepared datasets we also prepared the NL (No-Lone) versions (SRtrainNL, TS1reducedWCNL, etc.) with the lone base pairs removed. However, the preliminary experiments showed almost no difference between the standard and NL versions of the datasets, thus the NL benchmarks were eventually omitted.

RNA multiple sequence alignment datasets

The RNAStralignExt, RfamPDB, and Rfam14.9 datasets were prepared by downloading seed alignments in Stockholm format (https://en.wikipedia.org/wiki/Stockholm_format) from Rfam [10] (version 14.9). The Stockholm files were then converted into one-line Fasta-like (https://en.wikipedia.org/wiki/FASTA_format) and Clustal (https://meme-suite.org/meme/doc/clustalw-format.html) formats with custom Python scripts. The 134 families of the RfamPDB dataset were defined via a search for “Families with 3D structure” on the Rfam web page (https://rfam.org/search?q=entry_type:%22Family%22%20AND%20has_3d_structure:%22Yes%22).

The SubAli dataset was prepared as follows. For each of the eleven Rfam families of the RNAStralignExt dataset, we randomly sampled ten alignments of each of the eleven depths (where the seed alignment depth allowed it) - 2, 5, 10, 20, 30, 50, 75, 100, 150, 200, and 300 sequences. The sampled sequences were not subject to any realigning procedures.

The SeqSim dataset used to analyze the correlation between the prediction quality and mean sequence similarity of the alignment was prepared as follows. For each of the eleven Rfam families of the RNAStralignExt dataset, we sampled 100 alignments of each of the three depths - 5, 10, and 20 sequences (where the seed alignment depth allowed it). Then for each alignment of a given (family, depth) pair we calculated mean pairwise sequence similarity and rounded it to three decimal digits. Finally, only one alignment for each (family, depth, similarity) triple was kept, resulting in about 50 different alignments for a (family, depth) pair. Hereinafter, we used the Hamming distance (https://en.wikipedia.org/wiki/Hamming_distance) between the aligned sequences as the measure of pairwise sequence similarity. The sampled sequences were not subject to any realigning procedures.

The S01AliUngap dataset used to measure the prediction quality on the S01 dataset with alignment and alignment+shape input was prepared as follows. For each of the 24 sequences, BlastN [33] (https://blast.ncbi.nlm.nih.gov/Blast.cgi?PROGRAM=blastn&PAGE_TYPE=BlastSearch&LINK_LOC=blasthome) was used to search for homologs with at least 90% query sequence coverage. Then unique homologs were derived from the search results. If the resulting number of sequences was higher than 100, we selected the 100 most diverse sequences as follows. First, the reference sequence was selected. Then, we iteratively added new sequences one by one until we had 100 sequences by selecting at each step the sequence having the lowest highest similarity to the previously selected sequences. The resulting set of sequences was then realigned with MAFFT [34] (version v7.453), and the columns with gaps in the reference sequence were removed.

The sub-alignments of the most diverse sequences of the RF02929 family were prepared as follows. First, the two most dissimilar sequences were selected. Then, we iteratively added new sequences one by one until the required depth by selecting at each step the sequence having the lowest highest similarity to the previously selected sequences.

Quality metrics

To measure the prediction quality we defined the number of true positives (*TP*) as the number of correctly predicted base pairs, the number of false positives (*FP*) as the number of wrongly predicted base pairs, and the number of false negatives (*FN*) as the number of wrongly missed base pairs. We avoided defining the number of true negatives (*TN*). The precision metric was defined as *Precision = TP / (TP + FP)*, the recall metric was defined as *Recall = TP / (TP + FN)*, and the F-score metric was defined as the harmonic mean of precision and recall, *F-score = 2 * Precision * Recall / (Precision + Recall) = 2 * TP / (2 * TP + FP + FN)*. Along with the commonly used *mean F-score* value among the sequences of a given dataset we also used the *total F-score* value, calculated from the *TP*, *FP*, and *FN* values summed up over the sequences of a given dataset. Finally, we defined the *harmonized F-score* as the harmonic mean between the *mean F-score* and *total F-score* values and used it as the main target metric.

Other tools used for benchmarking

RNAfold and RNAalifold were installed locally as a part of ViennaRNA package (version 2.5.0) and run with commands "*RNAfold --noPS [--shape=shapefile] < inputfile > outputfile*" and "*RNAalifold --noPS inputfile [--shape=shapefile] > outputfile*". IPknot was installed locally (from https://github.com/satoken/ipknot, version 1.0.0) and run with command "*ipknot inputfile > outputfile*". On the SeqSim dataset, IPknot stuck for more than ten days on the alignment *RF00177_5_729*, and on the Rfam14.9 dataset stuck for more than ten days on the RF02746 family alignment, thus we couldn't obtain IPknot benchmarks for these two datasets. ShapeKnots was installed locally as a part of RNAstructure package (version 6.4) and run with command "*DATAPATH=../data_tables ./ShapeKnots-smp inputfile outputfile [-sh shapefile]*". MXfold2 was installed locally (from https://github.com/mxfold/mxfold2, version 0.1.2) and run with command "*mxfold2 predict inputfile > outputfile*". SPOT-RNA was installed locally (from https://github.com/jaswindersingh2/SPOT-RNA, on 16 April 2023) and run with command "*python SPOT-RNA.py --inputs inputfile --outputs outputfolder --cpu 32*".

CentroidAlifold was installed locally (from https://github.com/satoken/centroid-rna-package, version v0.0.17) and run with command "*centroid_alifold inputfile > outputfile*". Rscape was installed locally (from

http://eddylab.org/R-scape/, version 1.6.1) and run with command "*R-scape --fold --covmin 4 --nofigures --rna inputfile > outputfile*". The "Rscape nested" structure was defined from the "SS_cons" line of the output, and the "Rscape total" structure was defined from the "SS_cons" line with the addition of consistent base pairs from the "SS_cons_N" lines of the output. To obtain ShapeSorter predictions we used the web server version (https://e-rna.org/shapesorter/, on 22 August 2023) as we failed to run it locally. As ShapeSorter predicts separate stems, the predicted structures were built via the greedy selection of the predicted stems below the recommended p-value threshold of $7 * 10^{-4}$ [40]. We didn't include the state-of-the-art deep-learning-based method SPOT-RNA2 [19] in the benchmarks because it takes a single sequence input and builds an alignment of homologous sequences on the fly. Hence, it's not clear to which category of the tools it should belong.

The benchmarks were prepared on an AMD Ryzen 9 5950X machine with 32 CPU cores and 128 GB RAM.

SQUARNA algorithm for the single-sequence prediction

The single-sequence SQUARNA algorithm consists of the following steps:

0) Initialization with an empty secondary structure as the only currently predicted structure;

1) annotation of all possible stems above the defined minimum length (minlen) and minimum bp-score (minbpscore) thresholds that are consistent with (but not a part of) the currently predicted structure;

2) calculation of stem scores according to the stem score formula (Figure 1B);

3) removal of the stems with the scores below the threshold defined as a the product of the minbpscore and a scaling factor (minfinscorefactor);

4) sorting the stems in the decreasing order of their stem scores;

5) selection of the stem with the highest score along with all the intersecting stems with a score within the suboptimality threshold (subopt);

6) adding the selected k stems to the currently predicted structure separately one by one, resulting in k new currently predicted structures;

7) repeat steps 1-6 for each of the currently predicted structures in the current pool and obtain a new pool. Repeat until the pool of currently predicted structures is empty. If for a given currently predicted structure the maximum allowed number of stems is reached (maxstemnum) or no stem can be added to the structure, the structure is moved from the pool to the set of predicted structures;

8) repeat steps 0-7 for each of the defined parameter sets;

9) all the predicted structures are ranked by their structure score, reactivity score, or the product of the two scores (Figure 1C).

The subopt parameter is being initialized with the starting value (suboptmin) and is increased by an increment value each time the size of the pool of the currently predicted structures grows (on step 7) until the maximum defined value is reached (suboptmax), The increment value is defined by a number of increment steps (suboptsteps) as (supoptmax - suboptmin) / suboptsteps.

Any hard restraints defined by the user are considered in step 1 allowing only the base pairs consistent with the restraints. In the case of a multi-chain input the distance factor is always set to 1 for the stems confining chain breaks, and $(i, i + 1)$, $(i, i + 2)$, and $(i, i + 3)$ base pairs are allowed for such stems. In the case of reactivities defined by the user, (i, j) base pairs of positive weights are multiplied by and of negative weights are divided by a reactivity factor defined as $(1 - (reactivity_i + reactivity_j) / 2) * 2$. The reactivities are expected to range from 0.0 to 1.0, with 0.5 being the neutral value, but outliers will be handled as well.

The theoretical time complexity of the algorithm is $O(log(K) * N^4)$, where N is the sequence length and K is the number of predicted structures. Step 1 takes $O(N^2)$ time to annotate all the stems, step 2 takes $O(N)$ time to score each of the $O(N^2)$ annotated stems, resulting in $O(N^3)$ time, and the number of times steps 1-6 are repeated is equal to the $O(N)$ number of stems in a predicted structure, resulting in $O(N^4)$. Overall, the algorithm runs pretty fast with *subopt = 1.0*, but significantly slows down with lower subopt values, as the number of alternatives K grows exponentially.

The initial training procedure was performed on the SRtrain150 dataset in the form of a sequence of grid-search runs. The target metric was the harmonized F-score of the best of the top-5 structures. After the training the following fixed values were chosen:

$Distance_{ideal}$ was set to 4 for hairpins and 2 for all other loops (distance factor formula in Figure 1B); the internal loops of the following sizes were defined as short near-symmetric loops deserving a bonus: *(0, 0), (0, 1), (1, 0), (1, 1), (0, 2), (2, 0), (2, 2), (1, 2), (2, 1), (3, 1), (1, 3), (2, 3), (3, 2), (3, 3), (3, 4), (4, 3), (4, 4), (4, 2), (2, 4)*; for symmetric loops *(x, x)* the bonus was doubled; for *(x - 1, x)* and *(x, x - 1)* loops the bonus was multiplied by 1.5; $Bonus_{GNRA}$ was set to 1.25; for the structure score (Figure 1C) the bp weights were set to *GU = -1.0, AU = 1.5, GC = 4.0*, and the $Power_{rank}$ was set to 1.7.

After the training the following parameter values were chosen: the ranking is performed based on the structure score (and this was set to default and used for all the benchmarks); the default config (def.conf) included two parameter sets - *def1* and *def2*; the *def1* parameter set included the following values: bp weights *GC = 3.25, AU = 1.25, GU = -1.25*; *suboptmax = 0.9*; *suboptmin = 0.65*; *suboptsteps = 1*; *minlen = 2*; *minbpscore = 4.5*; *minfinscorefactor = 1.25*; $Coefficient_{distance}$ (*distcoef*) = *0.09*; $Weight_{brackets}$ (*bracketweight*) = *-2*; $Penalty_{order}$ (*orderpenalty*) = *1.0*; $Bonus_{inner\ loop}$ & $Bonus_{outer\ loop}$ *(loopbonus) = 0.125*; the maximum number of stems to be predicted was not limited (*maxstemnum = 1e6*); the *def2* parameter set differed from *def1* in the following values: bp weights *GC = 2, AU = 1, GU = 1*; *minbpscore = 3*; *minfinscorefactor = 0.99*; *distcoef = 0.1*; *orderpenalty = 1.35*. We also saved an alternative config (alt.conf) with the best single parameter set "*alt*", which also works somehow better for short pseudoknotted RNAs. The *alt* parameter set differed from *def1* in a single value: *minfinscorefactor = 1.0*.

The *sk* config (sk.conf) was trained on a subset of the S01 dataset involving 14 sequences shorter than 200 nts along with their SHAPE data and included two parameter sets - *sk1* and *sk2*. The *sk1* parameter set differed from the *def1* set in the following values: *suboptmin = 0.75; suboptsteps = 2; minbpscore = 7; orderpenalty = 0.75*. The *sk2* parameter set differed from the *sk1* set in the following values: bp weights *GC = 2, AU = 1, GU = 1; minbpscore = 3; minfinscorefactor = 0.99; distcoef = 0.1; orderpenalty = 1.35*.

Due to slow performance of the trained SQUARNA configs on the longer RNA sequences we designed two faster configs, one for the sequences longer than 500 nts (500.conf) and another for the sequences longer than 1000 nts (1000.conf). The configs were used for the sequences of appropriate lengths in all the single-sequence

benchmarks. The 500 config included two parameter sets *long1* and *long2* that differed from the *def1* and *def2* sets only in the two values: *suboptmax = 0.95; suboptmin = 0.9*. The 1000 config included a single parameter set that differed from the *def1* set in the two values as well: *suboptmax = 0.99; suboptmin = 0.99*.

SQUARNA algorithm for the alignment-based prediction

The alignment-based SQUARNA algorithm works as described in the Results section. To calculate the total stem bp-score matrix on step 1 we un-align each individual sequence by removing all the gaps, calculate an individual stem bp-score matrix based on three parameters (bpweights, minlen, minbpscore), and re-align the individual matrix to the alignment size by introducing all-zero rows and columns in gap positions of the aligned sequence. An individual structure of a given sequence at step 2 is defined as the consensus of the top-N (1 by default) ranked predicted structures based on a defined config. The consensus structure at step 2 is calculated by greedy selection of a consistent set of the most populated (i.e., present in the highest number of sequences) base pairs from the individual structures present in at least *freqlim* share of the sequences. For each of the step 1 and step 2 predicted structures separately we apply a trimming procedure removing the base pairs of pseudoknot orders higher than the defined threshold (*levellim*). On step 3 the union of the step 1 and step 2 base pairs (in the step3u setting) is calculated by adding consistent step 2 base pairs to the base pairs obtained after step 1.

The parameter training procedure was performed on the RNAStralignExt dataset. The target metric was the mean F-score of SQUARNA step3u setting. After the training, the following parameter values were chosen: levellim = 3 for alignment length under 500 nts and 2 otherwise; freqlim was set to 0.35; step1 parameters: bp weights GC = 3.25, AU = 1.25, GU = -1.25, minlen = 2, minbpscore = 4.5; the config for the single-sequence SQUARNA predictions at step 2 (ali.conf) involved a single parameter set “ali” that included the following values: bp weights GC = 3.25, AU = 1.25, GU = -1.25; suboptmax = 1.0; suboptmin = 1.0; suboptsteps = 1; minlen = 2; minbpscore = 4.5; minfinscorefactor = 1.0; distcoef = 0.09; bracketweight = -2; orderpenalty = 0.75; loopbonus = 0.125; maxstemnum = 1e6. At step 2, base pair scores in the individual predictions were weighted with the total stem bp-score matrix obtained at iteration 1 of step 1 as follows. The matrix was normalized to have the maximum value of 1.0 (via dividing all the values by the maximum value), then all the matrix values were multiplied

by 5.0. The resulting matrix was then un-aligned for each individual sequence separately and used to weight the base pair scores.

The theoretical time complexity of SQUARNA step 1 is $O(D * N^2)$, where D is the alignment depth and N is the alignment length, as the calculation of individual stem bp-score matrices takes $O(N^2)$ time to annotate all the possible stems. In all the benchmarks SQUARNA step1 was the second fastest algorithm after RNAalifold. The theoretical time complexity of SQUARNA step 2 is $O(D * log(K) * N^4)$, which is the complexity of single-sequence SQUARNA multiplied by the number of sequences in the alignment.

SQUARNA tool implementation

A command-line SQUARNA tool is implemented in Python 3 with the use of the multiprocessing module from the Python standard library and is available at https://github.com/febos/SQUARNA. The single-sequence SQUARNA is parallelized by the pool of the currently predicted structures, whereas the alignment-based SQUARNA is parallelized by the individual sequences. SQUARNA can take as input all standard file formats - FASTA, Stockholm, and Clustal, along with its own fasta-like format that allows incorporating of restraints, reactivities and reference structure input data. In the case of running single-sequence predictions with an alignment input the predicted structures will be aligned to the input sequences.

DATA AVAILABILITY

An implementation of SQUARNA tool and the benchmarking data are available at https://github.com/febos/SQUARNA (DOI: 10.5281/zenodo.8292326).

FUNDING

E.F.B. was supported by the European Molecular Biology Organization [fellowship ALTF 525-2022 to E.F.B]. J.M.B. and G.I.N were supported by the NCN [grant 2017/26/A/NZ1/01083 to J.M.B.].

ACKNOWLEDGEMENTS

E.F.B. thanks his former supervisor Mikhail Roytberg for his visionary idea to choose a stem as a prediction unit, and Pavel Pevzner and Phillip Compeau for their online course "Bioinformatics Algorithms: An Active Learning Approach", which had a huge impact on this work.

REFERENCES

1. Morris, K. V., & Mattick, J. S. (2014). The rise of regulatory RNA. Nature Reviews Genetics, 15(6), 423-437. DOI: 10.1038/nrg3722

2. Kavita, K., & Breaker, R. R. (2023). Discovering riboswitches: the past and the future. Trends in biochemical sciences. DOI: 10.1016/j.tibs.2022.08.009

3. MacFadden, A., O'Donoghue, Z., Silva, P. A., Chapman, E. G., Olsthoorn, R. C., Sterken, M. G., Pijlman, G.P., Bredenbeek, P.J. & Kieft, J. S. (2018). Mechanism and structural diversity of exoribonuclease-resistant RNA structures in flaviviral RNAs. Nature Communications, 9(1), 119. DOI: 10.1038/s41467-017-02604-y

4. Ganser, L. R., Kelly, M. L., Herschlag, D., & Al-Hashimi, H. M. (2019). The roles of structural dynamics in the cellular functions of RNAs. Nature reviews Molecular cell biology, 20(8), 474-489. DOI: 10.1038/s41580-019-0136-0

5. Capriotti, E., & Marti-Renom, M. A. (2010). Quantifying the relationship between sequence and three-dimensional structure conservation in RNA. BMC bioinformatics, 11(1), 1-10. DOI: 10.1186/1471-2105-11-322

6. Tinoco Jr, I., & Bustamante, C. (1999). How RNA folds. Journal of molecular biology, 293(2), 271-281. DOI: 10.1006/jmbi.1999.3001

7. Bailor, M. H., Mustoe, A. M., Brooks III, C. L., & Al-Hashimi, H. M. (2011). Topological constraints: using RNA secondary structure to model 3D conformation, folding pathways, and dynamic adaptation. Current opinion In structural biology, 21(3), 296-305. DOI: 10.1016/j.sbi.2011.03.009

8. Mitchell III, D., Assmann, S. M., & Bevilacqua, P. C. (2019). Probing RNA structure in vivo. Current opinion in structural biology, 59, 151-158. DOI: 10.1016/j.sbi.2019.07.008

9. Ma, H., Jia, X., Zhang, K., & Su, Z. (2022). Cryo-EM advances in RNA structure determination. Signal Transduction and Targeted Therapy, 7(1), 58. DOI: 10.1038/s41392-022-00916-0

10. Kalvari, I., Nawrocki, E. P., Ontiveros-Palacios, N., Argasinska, J., Lamkiewicz, K., Marz, M., ... & Petrov, A. I. (2021). Rfam 14: expanded coverage of metagenomic, viral and microRNA families. Nucleic Acids Research, 49(D1), D192-D200. DOI: 10.1093/nar/gkaa1047

11. Mathews, D. H., & Turner, D. H. (2006). Prediction of RNA secondary structure by free energy minimization. Current opinion in structural biology, 16(3), 270-278. DOI: 10.1016/j.sbi.2006.05.010

12. Eddy, S. R. (2004). How do RNA folding algorithms work? Nature biotechnology, 22(11), 1457-1458. DOI: 10.1038/nbt1104-1457

13. Sato, K., & Kato, Y. (2022). Prediction of RNA secondary structure including pseudoknots for long sequences. Briefings in Bioinformatics, 23(1), bbab395. DOI: 10.1093/bib/bbab395

14. Rivas, E. (2020). RNA structure prediction using positive and negative evolutionary information. PLoS computational biology, 16(10), e1008387. DOI: 10.1371/journal.pcbi.1008387

15. Bernhart, S. H., Hofacker, I. L., Will, S., Gruber, A. R., & Stadler, P. F. (2008). RNAalifold: improved consensus structure prediction for RNA alignments. BMC bioinformatics, 9, 1-13. DOI: 10.1186/1471-2105-9-474

16. Hamada, M., Sato, K., & Asai, K. (2011). Improving the accuracy of predicting secondary structure for aligned RNA sequences. Nucleic acids research, 39(2), 393-402. DOI: 10.1093/nar/gkq792

17. Sato, K., Akiyama, M., & Sakakibara, Y. (2021). RNA secondary structure prediction using deep learning with thermodynamic integration. Nature communications, 12(1), 941. DOI: 10.1038/s41467-021-21194-4

18. Singh, J., Hanson, J., Paliwal, K., & Zhou, Y. (2019). RNA secondary structure prediction using an ensemble of two-dimensional deep neural networks and transfer learning. Nature communications, 10(1), 5407. DOI: 10.1038/s41467-019-13395-9

19. Singh, J., Paliwal, K., Zhang, T., Singh, J., Litfin, T., & 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. DOI: 10.1093/bioinformatics/btab165

20. Matarrese, M. A., Loppini, A., Nicoletti, M., Filippi, S., & Chiodo, L. (2023). Assessment of tools for RNA secondary structure prediction and extraction: a final-user perspective. Journal of Biomolecular Structure and Dynamics, 41(14), 6917-6936. DOI: 10.1080/07391102.2022.2116110

21. Lorenz, R., Bernhart, S. H., Höner zu Siederdissen, C., Tafer, H., Flamm, C., Stadler, P. F., & Hofacker, I. L. (2011). ViennaRNA Package 2.0. Algorithms for molecular biology, 6, 1-14. DOI: 10.1186/1748-7188-6-26

22. Liu, B., Mathews, D. H., & Turner, D. H. (2010). RNA pseudoknots: folding and finding. F1000 biology reports, 2. DOI: 10.3410/B2-8

23. Hajdin, C. E., Bellaousov, S., Huggins, W., Leonard, C. W., Mathews, D. H., & Weeks, K. M. (2013). Accurate SHAPE-directed RNA secondary structure modeling, including pseudoknots. Proceedings of the National Academy of Sciences, 110(14), 5498-5503. DOI: 10.1073/pnas.1219988110

24. Bernhart, S. H., Tafer, H., Mückstein, U., Flamm, C., Stadler, P. F., & Hofacker, I. L. (2006). Partition function and base pairing probabilities of RNA heterodimers. Algorithms for Molecular Biology, 1, 1-10. DOI: 10.1186/1748-7188-1-3

25. Szikszai, M., Wise, M., Datta, A., Ward, M., & Mathews, D. H. (2022). Deep learning models for RNA secondary structure prediction (probably) do not generalize across families. Bioinformatics, 38(16), 3892-3899. DOI: 10.1093/bioinformatics/btac415

26. Grigorashvili, E. I., Chervontseva, Z. S., & Gelfand, M. S. (2022). Predicting RNA secondary structure by a neural network: what features may be learned? PeerJ, 10, e14335. DOI: 10.7717/peerj.14335

27. Mathews, D. H., Sabina, J., Zuker, M., & Turner, D. H. (1999). Expanded sequence dependence of thermodynamic parameters improves prediction of RNA secondary structure. Journal of molecular biology, 288(5), 911-940. DOI: 10.1006/jmbi.1999.2700

28. Burley, S. K., Bhikadiya, C., Bi, C., Bittrich, S., Chao, H., Chen, L., ... & Zardecki, C. (2023). RCSB Protein Data Bank (RCSB. org): delivery of experimentally-determined PDB structures alongside one million computed structure models of proteins from artificial intelligence/machine learning. Nucleic acids research, 51(D1), D488-D508. DOI: 10.1093/nar/gkac1077

29. Miao, Z., Adamiak, R. W., Antczak, M., Boniecki, M. J., Bujnicki, J., Chen, S. J., ... & Westhof, E. (2020). RNA-Puzzles Round IV: 3D structure predictions of four ribozymes and two aptamers. RNA, 26(8), 982-995. DOI: 10.1261/rna.075341.120

30. Das, R., Kretsch, R. C., Simpkin, A., Mulvaney, T., Pham, P., Rangan, R., ... & Westhof, E. (2023). Assessment of three-dimensional RNA structure prediction in CASP15. bioRxiv, 2023-04. DOI: 10.1101/2023.04.25.538330

31. Tan, Z., Fu, Y., Sharma, G., & Mathews, D. H. (2017). TurboFold II: RNA structural alignment and secondary structure prediction informed by multiple homologs. Nucleic acids research, 45(20), 11570-11581. DOI: 10.1093/nar/gkx815

32. Benard, L., Philippe, C., Ehresmann, B., Ehresmann, C., & Portier, C. (1996). Pseudoknot and translational control in the expression of the S15 ribosomal protein. Biochimie, 78(7), 568-576. DOI: 10.1016/S0300-9084(96)80003-4

33. Johnson, M., Zaretskaya, I., Raytselis, Y., Merezhuk, Y., McGinnis, S., & Madden, T. L. (2008). NCBI BLAST: a better web interface. Nucleic acids research, 36(suppl_2), W5-W9. DOI: 10.1093/nar/gkn201

34. Katoh, K., & Standley, D. M. (2013). MAFFT multiple sequence alignment software version 7: improvements in performance and usability. Molecular biology and evolution, 30(4), 772-780. DOI: 10.1093/molbev/mst010

35. Boccaletto, P., Stefaniak, F., Ray, A., Cappannini, A., Mukherjee, S., Purta, E., ... & Bujnicki, J. M. (2022). MODOMICS: a database of RNA modification pathways. 2021 update. Nucleic acids research, 50(D1), D231-D235. DOI: 10.1093/nar/gkab1083

36. Bohdan, D. R., Voronina, V. V., Bujnicki, J. M., & Baulin, E. F. (2023). A comprehensive survey of long-range tertiary interactions and motifs in non-coding RNA structures. Nucleic Acids Research, 2023, gkad605. DOI: 10.1093/nar/gkad605

37. Baulin, E., Yacovlev, V., Khachko, D., Spirin, S., & Roytberg, M. (2016). URS DataBase: universe of RNA structures and their motifs. Database, 2016, baw085. DOI: 10.1093/database/baw085

38. Leontis, N. B., & Zirbel, C. L. (2012). Nonredundant 3D structure datasets for RNA knowledge extraction and benchmarking. RNA 3D structure analysis and prediction, 281-298. DOI: 10.1007/978-3-642-25740-7_13

39. Lu, X. J., Bussemaker, H. J., & Olson, W. K. (2015). DSSR: an integrated software tool for dissecting the spatial structure of RNA. Nucleic acids research, 43(21), e142-e142. DOI: 10.1093/nar/gkv716

40. Tsybulskyi, V., & Meyer, I. M. (2022). ShapeSorter: A fully probabilistic method for detecting conserved RNA structure features supported by SHAPE evidence. Nucleic Acids Research, 50(15), e85-e85. DOI: 10.1093/nar/gkac405

SUPPLEMENTARY MATERIALS

Supplementary Table S1. Overview of the RNA secondary structure prediction tools used in this work. SQUARNA provides eight of the ten listed features, whereas the other tools provide five features at most.

	RNAfold	IPknot	Shape Knots	MXfold2	SPOT-RNA	SQUARNA	RNAalifold	CentroidAlifold	Rscape (CaCoFold)
single-sequence input									
alignment input									
structure restraints input									
residue reactivities input									
reference structure input									
circular molecule input									
multi-chain input									
predicts alternative structures									
predicts pseudoknots									
predicts non-canonical base pairs									
Number of features	4	4	5	1	3	8	4	4	4

Supplementary Table S2. Comparisons of the single-sequence input tools.

Dataset	Metric	RNAfold	IPknot	Shape Knots	MXfold2	SPOT-RNA	SQUARNA			
							default top-1	alternative top-1	default top-5	alternative top-5
TS1reducedWC	Harmonized F-score	0.723	0.742	0.704	0.793	**0.819**	0.711	0.709	0.789	0.781
	Total F-score	0.721	0.743	0.697	0.795	**0.82**	0.707	0.703	0.772	0.775
	Mean F-score	0.725	0.742	0.711	0.792	**0.819**	0.716	0.715	0.806	0.788
	Time in sec	0.5	2.3	2785.6	26.5	1203.5* (25.1)	17	10.4	17.1	10.4

SRtrain	Harmonized F-score	0.72	0.751	0.723	**0.789**	0.774	0.691	0.665	0.771	0.735
	Total F-score	0.694	0.746	0.697	**0.762**	0.734	0.63	0.603	0.705	0.672
	Mean F-score	0.748	0.757	0.75	0.819	0.818	0.766	0.742	**0.851**	0.812
	Time in sec	6.2	8394.6	16715.4	267.3	7010.2* (1424.7)	1918.8**	1771.6**	1883.1**	1817.9**

*SPOT-RNA has a ~20-sec overhead time for a single run. We ran the predictions sequence by sequence, and this way the overhead took place at each run. If we run all the sequences in a single bulk run, the runtime would be close to the time taken for the prediction of the longest sequence (shown in parentheses)

**In the benchmarks SQUARNA utilized the faster-running configs 500.conf for the sequences longer than 500 nts, and 1000.conf for the sequences longer than 1000 nts, as the def.conf and alt.conf configs can take days to run on long sequences

Supplementary Table S3. Student's t-test p-values calculated for each pair of the single-sequence input tools. Significant p-values ≤ 0.01 are shown in bold font.

Pairwise t-test p-values on TS1reducedWC dataset results

TS1reduced WC	**RNAfold**	**IPknot**	**Shape Knots**	**SQUARNA default top-1**	**SQUARNA alternative top-1**	**MXfold2**	**SPOT-RNA**	**SQUARNA default top-5**	**SQUARNA alternative top-5**
RNAfold	1.00E+00	7.11E-01	7.82E-01	8.61E-01	8.46E-01	1.34E-01	2.47E-02	6.39E-02	1.62E-01
IPknot	7.11E-01	1.00E+00	4.96E-01	5.69E-01	5.59E-01	2.03E-01	3.21E-02	9.33E-02	2.47E-01
ShapeKnots	7.82E-01	4.96E-01	1.00E+00	9.19E-01	9.37E-01	6.53E-02	**8.58E-03**	2.72E-02	8.27E-02
SQUARNA default top-1	8.61E-01	5.69E-01	9.19E-01	1.00E+00	9.83E-01	8.45E-02	1.24E-02	3.69E-02	1.06E-01
SQUARNA alternative top-1	8.46E-01	5.59E-01	9.37E-01	9.83E-01	1.00E+00	8.47E-02	1.30E-02	3.75E-02	1.05E-01
MXfold2	1.34E-01	2.03E-01	6.53E-02	8.45E-02	8.47E-02	1.00E+00	4.15E-01	6.87E-01	9.16E-01
SPOT-RNA	2.47E-02	3.21E-02	**8.58E-03**	1.24E-02	1.30E-02	4.15E-01	1.00E+00	6.96E-01	3.60E-01
SQUARNA default top-5	6.39E-02	9.33E-02	2.72E-02	3.69E-02	3.75E-02	6.87E-01	6.96E-01	1.00E+00	6.14E-01

SQUARNA alternative top-5	1.62E-01	2.47E-01	8.27E-02	1.06E-01	1.05E-01	9.16E-01	3.60E-01	6.14E-01	1.00E+00
Pairwise t-test p-values on SRtrain dataset results									
SRtrain	**RNAfold**	**IPknot**	**Shape Knots**	**SQUARNA default top-1**	**SQUARNA alternative top-1**	**MXfold2**	**SPOT-RNA**	**SQUARNA default top-5**	**SQUARNA alternative top-5**
RNAfold	1.00E+00	7.15E-01	9.58E-01	4.74E-01	8.18E-01	**3.24E-03**	**4.28E-03**	**4.75E-06**	**7.23E-03**
IPknot	7.15E-01	1.00E+00	7.54E-01	7.06E-01	5.33E-01	**5.10E-03**	**6.84E-03**	**3.69E-06**	1.18E-02
ShapeKnots	9.58E-01	7.54E-01	1.00E+00	5.04E-01	7.74E-01	**3.39E-03**	**4.49E-03**	**4.34E-06**	**7.63E-03**
SQUARNA default top-1	4.74E-01	7.06E-01	5.04E-01	1.00E+00	3.25E-01	1.53E-02	1.97E-02	**2.24E-05**	3.27E-02
SQUARNA alternative top-1	8.18E-01	5.33E-01	7.74E-01	3.25E-01	1.00E+00	**9.92E-04**	**1.38E-03**	**5.20E-07**	**2.43E-03**
MXfold2	**3.24E-03**	**5.10E-03**	**3.39E-03**	1.53E-02	**9.92E-04**	1.00E+00	9.38E-01	8.99E-02	7.30E-01
SPOT-RNA	**4.28E-03**	**6.84E-03**	**4.49E-03**	1.97E-02	**1.38E-03**	9.38E-01	1.00E+00	7.80E-02	7.93E-01
SQUARNA default top-5	**4.75E-06**	**3.69E-06**	**4.34E-06**	**2.24E-05**	**5.20E-07**	8.99E-02	7.80E-02	1.00E+00	3.42E-02
SQUARNA alternative top-5	**7.23E-03**	1.18E-02	**7.63E-03**	3.27E-02	**2.43E-03**	7.30E-01	7.93E-01	3.42E-02	1.00E+00

Supplementary Table S4. Content of the RNAStralignExt dataset.

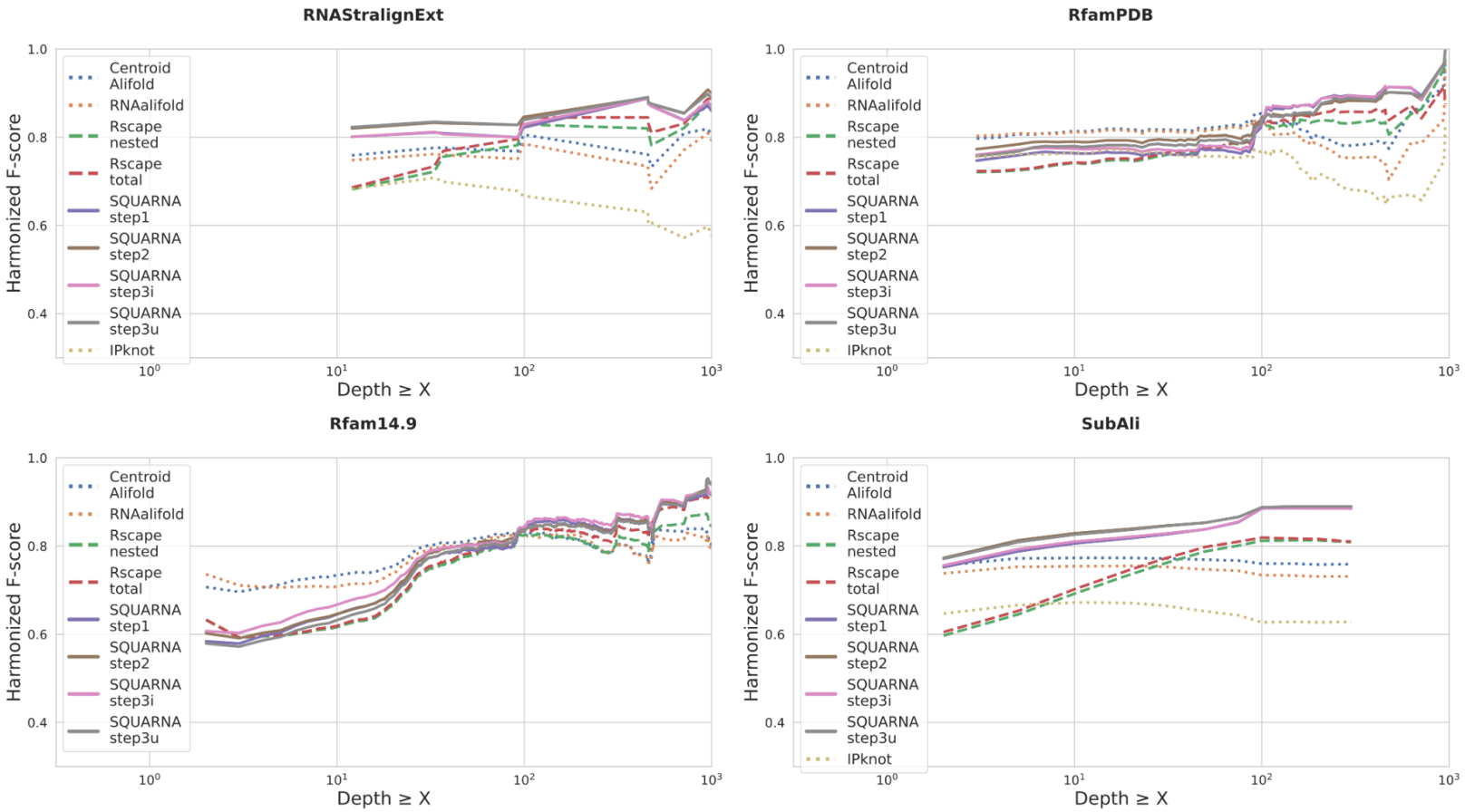
RNAStralignExt				
N	**Family**	**Rfam ID**	**Length**	**Depth**
1	5S rRNA	RF00001	230	712
2	tRNA	RF00005	118	954
3	RNaseP	RF00010	996	458
4	tmRNA	RF00023	954	477
5	telomerase RNA	RF00024	777	37
6	group I intron	RF00028	892	12
7	HDV ribozyme	RF00094	115	33
8	SAM riboswitch	RF00162	239	457
9	16S rRNA	RF00177	1980	99
10	SRP RNA	RF01854	315	92
11	Twister ribozyme	RF03160	189	1613

Supplementary Table S5. Comparisons of the alignment input tools. IPknot values are missing for the Rfam14.9 dataset as IPknot took more than 10 days to run on the RF02746 family and was eventually aborted.

Dataset	**Metric**	**Centroid Alifold**	**RNAali fold**	**IPknot**	**Rscape nested**	**Rscape total**	**SQUARNA step1**	**SQUARNA step2**	**SQUARNA step3i**	**SQUARNA step3u**
RNAStralignExt	Harmonized F-score	0.759	0.748	0.683	0.682	0.686	0.801	0.82	0.801	**0.823**
	Total F-score	0.785	0.775	0.697	0.711	0.709	0.782	**0.807**	0.785	0.806
	Mean F-score	0.735	0.723	0.67	0.655	0.664	0.82	0.834	0.818	**0.84**
	Time in sec	1093.4	7.9	495.5	115.1	114.8	53.4	1072	1088.2	1096.6
RNAStralignExt, Depth ≥ 30	Harmonized F-score	0.776	0.762	0.708	0.721	0.734	0.81	0.833	0.812	**0.835**
	Total F-score	0.798	0.786	0.722	0.744	0.755	0.788	**0.815**	0.792	0.814
	Mean F-score	0.756	0.74	0.695	0.7	0.715	0.834	0.851	0.833	**0.858**

RNAStralignExt, Depth ≥ 300	Harmonized F-score	0.761	0.734	0.63	0.82	0.845	0.888	0.89	0.888	**0.891**
	Total F-score	0.75	0.723	0.632	0.814	0.84	**0.893**	0.89	**0.893**	0.892
	Mean F-score	0.772	0.745	0.629	0.827	0.85	0.883	**0.89**	0.883	**0.89**
RfamPDB	Harmonized F-score	0.797	**0.803**	0.756	0.72	0.723	0.747	0.773	0.76	0.757
	Total F-score	0.799	**0.805**	0.73	0.718	0.716	0.714	0.757	0.731	0.737
	Mean F-score	0.795	**0.801**	0.784	0.723	0.73	0.783	0.79	0.792	0.778
	Time in sec	29858.8	134.1	2921.4	8681.1	8055.1	766.6	26678.7	26782	26576.9
RfamPDB, Depth ≥ 30	Harmonized F-score	**0.817**	0.813	0.763	0.759	0.763	0.763	0.794	0.772	0.783
	Total F-score	0.82	**0.821**	0.743	0.752	0.75	0.726	0.779	0.739	0.761
	Mean F-score	**0.814**	0.806	0.784	0.766	0.776	0.803	0.81	0.807	0.807
RfamPDB, Depth ≥ 300	Harmonized F-score	0.785	0.755	0.676	0.837	0.857	0.89	0.882	**0.892**	0.884
	Total F-score	0.77	0.745	0.663	0.829	0.848	**0.888**	0.878	**0.888**	0.882
	Mean F-score	0.801	0.766	0.689	0.846	0.867	0.893	0.887	**0.896**	0.887
Rfam14.9	Harmonized F-score	0.707	**0.736**		0.632	0.633	0.584	0.602	0.607	0.579
	Total F-score	0.738	**0.761**		0.65	0.651	0.6	0.614	0.634	0.589
	Mean F-score	0.679	**0.713**		0.615	0.616	0.569	0.59	0.583	0.57
	Time in sec	31542.4	212.5		12000.2	11787.3	1482.4	28858.6	28777.5	28604.9
Rfam14.9, Depth ≥ 30	Harmonized F-score	**0.8**	0.79		0.744	0.75	0.769	0.779	0.784	0.769
	Total F-score	**0.811**	0.802		0.747	0.751	0.762	0.782	0.778	0.769
	Mean F-score	0.789	0.778		0.741	0.749	0.777	0.777	**0.791**	0.77
Rfam14.9,	Harmonized	0.799	0.802		0.803	0.827	0.848	0.846	**0.862**	0.84

Depth ≥ 300	F-score									
	Total F-score	0.8	0.797		0.812	0.834	0.858	0.857	**0.869**	0.85
	Mean F-score	0.799	0.807		0.794	0.82	0.839	0.836	**0.855**	0.831
SubAli	Harmonized F-score	0.756	0.738	0.647	0.597	0.606	0.752	**0.773**	0.756	0.771
	Total F-score	**0.772**	0.75	0.657	0.58	0.589	0.727	0.753	0.735	0.749
	Mean F-score	0.741	0.726	0.638	0.615	0.623	0.779	**0.795**	0.778	0.794
	Time in sec	23315.1	259.1	86705.3	8483.4	8692.7	1606	24222.8	24154.4	24204.2
SubAli, Depth >= 30	Harmonized F-score	0.772	0.753	0.665	0.76	0.771	0.826	**0.846**	0.828	0.845
	Total F-score	0.783	0.763	0.68	0.75	0.759	0.806	**0.831**	0.811	0.827
	Mean F-score	0.762	0.743	0.65	0.77	0.783	0.847	0.862	0.846	**0.863**



***Supplementary Figure S1**. Benchmarking results of the alignment-based SQUARNA algorithm. X-axis - alignment depth threshold, Y-axis - harmonized F-score metric value. The four plots represent the four datasets: RNAStralignExt, RfamPDB, Rfam14.9, and SubAli.*

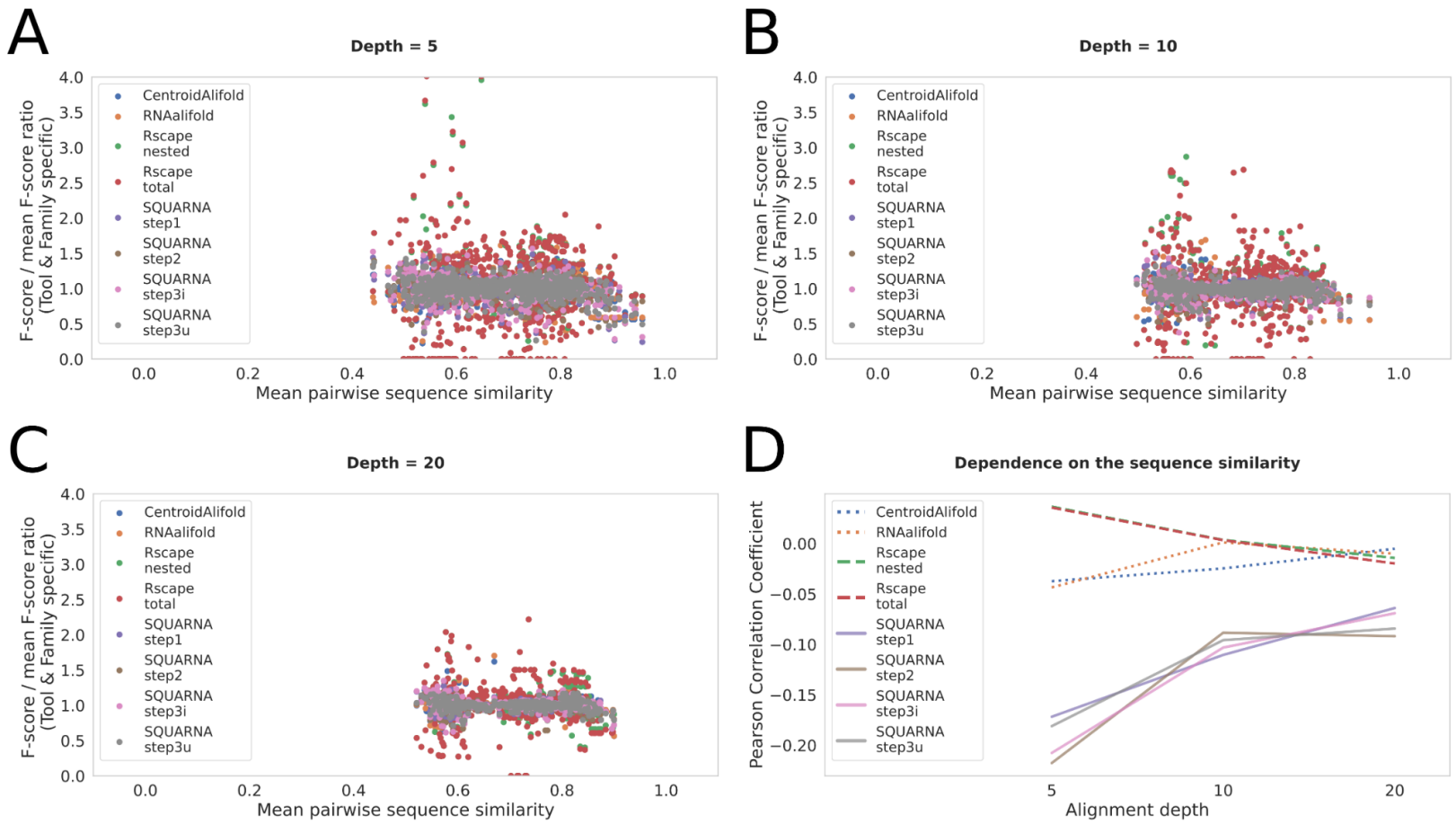
Supplementary Table S6. Student's t-test p-values calculated for each pair of the alignment input tools. Significant p-values ≤ 0.01 are shown in bold font.

Pairwise t-test p-values on SubAli dataset results									
SubAli	**CentroidAlifold**	**RNAalifold**	**IPknot**	**Rscape nested**	**Rscape total**	**SQUARNA step1**	**SQUARNA step2**	**SQUARNA step3i**	**SQUARNA step3u**
CentroidAlifold	1.00E+00	5.04E-02	**4.04E-40**	**2.16E-32**	**1.14E-28**	**3.73E-06**	**7.62E-12**	**8.74E-06**	**7.35E-12**
RNAalifold	5.04E-02	1.00E+00	**1.87E-29**	**2.07E-25**	**3.86E-22**	**1.53E-10**	**6.20E-18**	**5.90E-10**	**5.09E-18**
IPknot	**4.04E-40**	**1.87E-29**	1.00E+00	2.91E-02	1.54E-01	**3.24E-63**	**2.10E-81**	**1.86E-60**	**1.93E-82**

Rscape nested	**2.16E-32**	**2.07E-25**	2.91E-02	1.00E+00	5.33E-01	**4.88E-49**	**3.84E-60**	**1.13E-47**	**1.93E-60**
Rscape total	**1.14E-28**	**3.86E-22**	1.54E-01	5.33E-01	1.00E+00	**1.24E-44**	**3.17E-55**	**2.26E-43**	**1.74E-55**
SQUARNA step1	**3.73E-06**	**1.53E-10**	**3.24E-63**	**4.88E-49**	**1.24E-44**	1.00E+00	4.70E-02	9.31E-01	5.22E-02
SQUARNA step2	**7.62E-12**	**6.20E-18**	**2.10E-81**	**3.84E-60**	**3.17E-55**	4.70E-02	1.00E+00	4.15E-02	9.47E-01
SQUARNA step3i	**8.74E-06**	**5.90E-10**	**1.86E-60**	**1.13E-47**	**2.26E-43**	9.31E-01	4.15E-02	1.00E+00	4.61E-02
SQUARNA step3u	**7.35E-12**	**5.09E-18**	**1.93E-82**	**1.93E-60**	**1.74E-55**	5.22E-02	9.47E-01	4.61E-02	1.00E+00

Pairwise t-test p-values on SubAli, Depth ≥ 30 dataset results

SubAli, Depth ≥ 30	**CentroidAlifold**	**RNAalifold**	**IPknot**	**Rscape nested**	**Rscape total**	**SQUARNA step1**	**SQUARNA step2**	**SQUARNA step3i**	**SQUARNA step3u**
CentroidAlifold	1.00E+00	4.64E-02	**1.12E-29**	4.31E-01	3.38E-02	**1.41E-22**	**2.45E-33**	**2.31E-21**	**7.68E-35**
RNAalifold	4.64E-02	1.00E+00	**6.70E-21**	**8.08E-03**	**5.89E-05**	**1.72E-31**	**4.25E-44**	**5.64E-30**	**8.09E-46**
IPknot	**1.12E-29**	**6.70E-21**	1.00E+00	**5.05E-30**	**6.92E-38**	**5.46E-93**	**1.81E-114**	**8.04E-90**	**3.19E-117**
Rscape nested	4.31E-01	**8.08E-03**	**5.05E-30**	1.00E+00	2.16E-01	**1.41E-16**	**1.18E-24**	**1.05E-15**	**8.80E-26**
Rscape total	3.38E-02	**5.89E-05**	**6.92E-38**	2.16E-01	1.00E+00	**6.71E-13**	**2.50E-20**	**3.87E-12**	**2.16E-21**
SQUARNA step1	**1.41E-22**	**1.72E-31**	**5.46E-93**	**1.41E-16**	**6.71E-13**	1.00E+00	3.63E-02	8.58E-01	1.77E-02
SQUARNA step2	**2.45E-33**	**4.25E-44**	**1.81E-114**	**1.18E-24**	**2.50E-20**	3.63E-02	1.00E+00	2.52E-02	7.74E-01
SQUARNA step3i	**2.31E-21**	**5.64E-30**	**8.04E-90**	**1.05E-15**	**3.87E-12**	8.58E-01	2.52E-02	1.00E+00	1.20E-02
SQUARNA step3u	**7.68E-35**	**8.09E-46**	**3.19E-117**	**8.80E-26**	**2.16E-21**	1.77E-02	7.74E-01	1.20E-02	1.00E+00



Supplementary Figure S2*. Analysis of the alignment-based prediction quality dependence on the sequence similarity within the input alignment. (A, B, C) The relative performance of the alignment-input tools against the mean pairwise sequence similarity within the input alignment of depth 5, 10, and 20. The relative performance is measured as a ratio of an F-score and the mean F-score demonstrated by the tool on the alignments of a particular family of the given depth. (D) Pearson correlation coefficients between the prediction quality of a tool and mean pairwise similarity of the input alignment plotted against the alignment depth.*

Supplementary Table S7. Comparisons of the RNA secondary structure prediction tools on the S01 dataset with different input types.

Metrics type		Harmonic (Mean, Total)			Total			Mean			Time in sec
Metrics		F-score	Precision	Recall	F-score	Precision	Recall	F-score	Precision	Recall	
Input	Tool										
sequence	RNAfold	0.639	0.647	0.637	0.617	0.609	0.626	0.663	0.689	0.648	0.9
	IPknot	0.701	0.766	0.65	0.684	0.745	0.633	0.718	0.788	0.668	17.7
	ShapeKnots	0.681	0.678	0.688	0.659	0.645	0.674	0.705	0.714	0.702	9473.7
	MXfold2	**0.744**	0.774	**0.721**	**0.736**	0.749	**0.724**	0.752	0.801	**0.719**	26.4
	SPOT-RNA	0.719	**0.837**	0.638	0.678	**0.818**	0.579	**0.765**	**0.856**	0.71	852.7* (72.7)
	SQUARNA top-1	0.602	0.613	0.595	0.571	0.576	0.566	0.637	0.655	0.628	1032.5**
	SQUARNAsk top-1	0.605	0.603	0.609	0.576	0.567	0.585	0.636	0.644	0.636	136.3**
	SQUARNA top-5	0.658	0.677	0.644	0.618	0.628	0.609	0.703	0.734	0.683	1016.6**
	SQUARNAsk top-5	0.666	0.676	0.661	0.627	0.628	0.627	0.71	0.733	0.699	136.4**
sequence + shape	RNAfold	0.778	0.802	0.76	0.768	0.78	0.756	0.789	0.825	0.764	1.2
	ShapeKnots	**0.852**	**0.844**	**0.862**	**0.841**	**0.831**	**0.852**	**0.864**	**0.858**	**0.872**	292.6
	SQUARNA top-1	0.649	0.635	0.666	0.631	0.618	0.646	0.668	0.653	0.687	1927.8**
	SQUARNAsk top-1	0.737	0.725	0.75	0.709	0.698	0.72	0.767	0.755	0.783	214.4**
	SQUARNA top-5	0.734	0.72	0.749	0.691	0.678	0.705	0.782	0.768	0.799	1914.7**
	SQUARNAsk top-5	0.779	0.778	0.781	0.74	0.737	0.743	0.822	0.823	0.824	212.8**
alignment	CentroidAlifold	0.606	0.841	0.496	0.627	**0.855**	0.494	0.586	0.828	0.498	239.8
	RNAalifold	0.565	0.729	0.489	0.583	0.717	0.491	0.549	0.741	0.488	8
	IPknot	**0.719**	**0.835**	**0.636**	**0.709**	0.827	**0.62**	**0.73**	**0.844**	**0.653**	239.8
	Rscape nested	0.589	0.652	0.542	0.571	0.62	0.529	0.608	0.687	0.555	89.8
	Rscape total	0.589	0.652	0.542	0.571	0.62	0.529	0.608	0.687	0.555	89.2
	SQUARNA step1	0.558	0.638	0.501	0.522	0.621	0.45	0.6	0.656	0.566	21.6

	SQUARNA step2	0.58	0.592	0.573	0.55	0.569	0.532	0.614	0.617	0.622	124.2
	SQUARNA step3i	0.581	0.775	0.482	0.551	0.772	0.428	0.615	0.779	0.551	125.7
	SQUARNA step3u	0.554	0.558	0.554	0.52	0.532	0.509	0.593	0.587	0.608	124.7
alignment + shape	**RNAalifold**	0.528	0.696	0.454	0.547	0.675	0.46	0.511	0.719	0.449	9.5
	ShapeSorter* **	0.316	**0.863**	0.208	0.325	**0.841**	0.201	0.308	**0.887**	0.216	
	SQUARNA step1	0.591	0.651	0.546	0.562	0.642	0.501	0.624	0.661	0.6	26
	SQUARNA step2	**0.638**	0.637	**0.644**	**0.619**	0.629	**0.609**	**0.659**	0.646	**0.684**	132.6
	SQUARNA step3i	0.629	0.794	0.531	0.603	0.798	0.484	0.658	0.791	0.588	134.7
	SQUARNA step3u	0.603	0.598	0.609	0.58	0.587	0.572	0.627	0.61	0.652	134.1

*SPOT-RNA has a ~20 sec overhead time for a single run. We ran the predictions sequence by sequence, and this way the overhead took place at each run. If we run all the sequences in a single bulk run, the runtime would be close to the time taken for the prediction of the longest sequence (shown in parentheses).
**In the benchmarks SQUARNA utilized the faster-running configs 500.conf for the sequences longer than 500 nts, and 1000.conf for the sequences longer than 1000 nts, as the def.conf and alt.conf configs can take days to run on long sequences.
***We used the ShapeSorter web server and thus couldn't measure the running time.