BIOINFORMATICS

Vol. 22 no. 14 2006, pages e90–e98
doi:10.1093/bioinformatics/btl246

CONTRAfold: RNA secondary structure prediction without physics-based models

Chuong B. Do[1,*], Daniel A. Woods[1] and Serafim Batzoglou[1]

[1]Computer Science Department, Stanford University, Stanford, CA 94305, USA

ABSTRACT

Motivation: For several decades, free energy minimization methods have been the dominant strategy for single sequence RNA secondary structure prediction. More recently, stochastic context-free grammars (SCFGs) have emerged as an alternative probabilistic methodology for modeling RNA structure. Unlike physics-based methods, which rely on thousands of experimentally-measured thermodynamic parameters, SCFGs use fully-automated statistical learning algorithms to derive model parameters. Despite this advantage, however, probabilistic methods have not replaced free energy minimization methods as the tool of choice for secondary structure prediction, as the accuracies of the best current SCFGs have yet to match those of the best physics-based models.

Results: In this paper, we present CONTRAfold, a novel secondary structure prediction method based on *conditional log-linear models* (CLLMs), a flexible class of probabilistic models which generalize upon SCFGs by using discriminative training and feature-rich scoring. In a series of cross-validation experiments, we show that grammar-based secondary structure prediction methods formulated as CLLMs consistently outperform their SCFG analogs. Furthermore, CONTRAfold, a CLLM incorporating most of the features found in typical thermodynamic models, achieves the highest single sequence prediction accuracies to date, outperforming currently available probabilistic and physics-based techniques. Our result thus closes the gap between probabilistic and thermodynamic models, demonstrating that statistical learning procedures provide an effective alternative to empirical measurement of thermodynamic parameters for RNA secondary structure prediction.

Availability: Source code for CONTRAfold is available at http://contra.stanford.edu/contrafold/.

Contact: chuongdo@cs.stanford.edu

1 INTRODUCTION

In many RNA-related studies—ranging from noncoding RNA detection [13] to folding dynamics simulations [24] to hybridization stability assessment for microarray oligo probe selection [19]—knowing the secondary structure of an RNA sequence reveals important constraints governing the molecule's physical properties and function. To date, experimental assays for base-pairing in RNA sequences constitute the most reliable method for secondary structure determination [3]; however, their difficulty and expense are often prohibitive, especially for high-throughput applications. For this reason, computational prediction provides an attractive alternative to empirical discovery of RNA secondary structure [4].

Traditionally, the most successful techniques for single sequence computational secondary structure prediction have relied on physics models of RNA structure. Methods belonging to this category identify candidate structures for an RNA sequence by free energy minimization [22] through dynamic programming (e.g., Mfold [26] and ViennaRNA [7]) or alternative optimization schemes (e.g., RDfolder [25]).

Parameters used in energy-based methods typically come from empirical studies of RNA structural energetics. For example, parameters for nearest neighbor interactions in stacking base pairs are derived from melting curves of synthesized oligonucleotides [23]. In some cases, however, the difficulty of experimental procedures places severe restrictions on what parameters are measurable, and hence, the scoring models used. For instance, most secondary structure programs ignore the sequence dependence of hairpin, bulge, internal, and multi-branch loop energies due to the inability to quantify these effects experimentally. Similarly, the energies of multi-branch loops in modern secondary structure prediction programs rely on ad hoc scoring rules due to the lack of experimental techniques for assessing their free energy contribution [11].

Recently, stochastic context-free grammars (SCFGs) have emerged as an alternative probabilistic methodology for modeling RNA structure [2,8,9]. These models specify formal grammar rules that induce a joint probability distribution over possible RNA structures and sequences. In particular, the parameters of SCFG models specify probability distributions over possible transformations that may be applied to a ''nonterminal'' symbol, and thus are subject to the standard mathematical constraints of probability distributions (i.e. parameters may not be negative, and certain sets of parameters must sum to one). Though these parameters do not have direct physical interpretations, they are easily learned from collections of RNA sequences annotated with known secondary structures, without the need for external laboratory experiments [1].

While fairly simple SCFGs achieve respectable prediction accuracies, attempts in recent years to improve their performance using more sophisticated models have thus far yielded only modest gains. As a result, a significant performance separation still remains between the best physics-based methods and the best SCFGs [1]. Consequently, one might assume that such a gap is the inevitable price to be paid for using easily learnable probabilistic models, which are unable to provide an adequate representation of the physics underlying RNA structural stability. We assert that this is not the case.

In this paper, we present CONTRAfold, a new secondary structure prediction tool based on a flexible probabilistic model called a *conditional log-linear model* (CLLM). CLLMs generalize upon SCFGs in the sense that any SCFG has an equivalent representation

*To whom correspondence should be addressed.

as an appropriately parameterized CLLM. Like SCFGs, CLLMs enjoy the ease of computationally-driven parameter learning. Unlike vanilla SCFGs, however, CLLMs also have the generality to represent complex scoring schemes, such as those used in modern energy-based secondary structure predictors such as Mfold. CONTRAfold, a CLLM based on a simplified Mfold-like scoring scheme, not only achieves the highest single sequence prediction accuracies to date but also provides users with a new mechanism for controlling the sensitivity and specificity of the prediction algorithm.

2 METHODS

In this section, we motivate the use of CLLMs for RNA secondary structure prediction by showing how they arise as a natural extension of SCFGs. We then describe the CONTRAfold secondary structure model, which extends and simplifies traditional energy-based scoring schemes while retaining the parameter learning ease of common probabilistic methods. Finally, we describe a maximum expected accuracy decoding algorithm for secondary structure prediction which allows the user to adjust the desired sensitivity/specificity of the returned predictions via a single parameter γ.

2.1 Modeling secondary structure with SCFGs

In the RNA secondary structure prediction problem, we are given an input sequence x, and our goal is to predict the best structure y. For probabilistic parsing techniques, this requires a way to calculate the conditional probability $P(y \mid x)$ of the structure y given the sequence x.

2.1.1 Representation Stochastic context-free grammars (SCFGs) provide a compact representation of a joint probability distribution over RNA sequences and their secondary structures. An SCFG for secondary structure prediction defines (1) a set of transformation rules, (2) a probability distribution over the transformation rules applicable to each nonterminal symbol, and (3) a mapping from parses (derivations) to secondary structures.

For example, consider the following simple unambiguous SCFG for a restricted class of RNA secondary structures:

(1) *Transformation rules*.

$$S \rightarrow \mathtt{a}S\mathtt{u} \mid \mathtt{u}S\mathtt{a} \mid \mathtt{c}S\mathtt{g} \mid \mathtt{g}S\mathtt{c} \mid \mathtt{g}S\mathtt{u} \mid \mathtt{u}S\mathtt{g} \mid \mathtt{a}S \mid \mathtt{c}S \mid \mathtt{g}S \mid \mathtt{u}S \mid \epsilon.$$

(2) *Rule probabilities*. The probability of transforming a nonterminal S into $\mathtt{a}S\mathtt{u}$ is $p_{S\rightarrow \mathtt{a}S\mathtt{u}}$, and similarly for the other transformation rules.

(3) *Mapping from parses to structures*. The secondary structure y corresponding to a parse σ contains a base pairing between two letters if and only if the two letters were generated in the same step of the derivation for σ.

For a sequence $x = \mathtt{agucu}$ with secondary structure[1] $y = (\,(\,.\,)\,)$, the unique parse σ corresponding to y is

$$S \rightarrow \mathtt{a}S\mathtt{u} \rightarrow \mathtt{ag}S\mathtt{cu} \rightarrow \mathtt{agu}S\mathtt{cu} \rightarrow \mathtt{agucu}. \quad (1)$$

The SCFG models the *joint* probability of generating the parse σ and the sequence x as

$$P(x,\sigma) = p_{S\rightarrow \mathtt{a}S\mathtt{u}} \cdot p_{S\rightarrow \mathtt{g}S\mathtt{c}} \cdot p_{S\rightarrow \mathtt{u}S} \cdot p_{S\rightarrow \epsilon}. \quad (2)$$

It follows that[2]

$$P(y \mid x) = \sum_{\sigma \in y} P(\sigma \mid x) = \frac{\sum_{\sigma \in y} P(x,\sigma)}{\sum_{\sigma' \in \Omega(x)} P(x,\sigma')}, \quad (3)$$

where $\Omega(x)$ is the space of all possible parses of x.

[1]The secondary structure of a sequence can be represented in *nested parenthesis* format, in which pairs of matching parentheses represent base pairings in the sequence.

[2]Here, we regard y as a ''set'' of parses σ sharing the same secondary structure. Note that in ambiguous grammars, the mapping from parses to secondary structures may be many-to-one.

2.1.2 Parameter estimation One of the chief advantages of SCFGs as a language for describing RNA secondary structure is the existence of well-understood algorithms for parameter estimation. Given a set $\mathcal{D} = \{(x^{(1)}, y^{(1)}), \ldots, (x^{(m)}, y^{(m)})\}$ of m pairs of RNA sequences $x^{(i)}$ with experimentally-validated secondary structures $y^{(i)}$, the training task involves finding the set of parameters $\boldsymbol{\theta} = \{p_1, \ldots, p_n\}$ (i.e., the probabilities for each of the n transformation rules) that maximize some specified objective function.

In the popular maximum likelihood approach, $\boldsymbol{\theta}$ is chosen to maximize the *joint likelihood* of the training sequences and their structures,

$$\ell_{\mathrm{ML}}(\boldsymbol{\theta} : \mathcal{D}) = \prod_{i=1}^{m} P(x^{(i)}, y^{(i)}; \boldsymbol{\theta}), \quad (4)$$

subject to the contraints that all parameters must be nonnegative, and certain group of parameters must sum to one. For unambiguous grammars, the solution $\boldsymbol{\theta}_{\mathrm{ML}}$ to this constrained optimization problem exists in closed form. Consequently, the maximum likelihood technique is by far the most commonly used method for SCFG parameter estimation in practice.

2.2 From SCFGs to CLLMs

Like SCFGs, *conditional log-linear models* (CLLMs) are probabilistic models which have the goal of defining the conditional probability of an RNA secondary structure y given a sequence x. Here, we motivate the CLLM framework by comparison to SCFGs.

2.2.1 Representation To understand how CLLMs generalize upon the representation of conditional probabilities for SCFGs, we first consider a feature-based representation of SCFGs that highlights several important assumptions made when modeling with SCFGs. Removing these assumptions leads directly to the CLLM framework.

For a particular parse σ of a sequence x, let $\mathbf{F}(x,\sigma) \in \mathbb{R}^n$ be an n-dimensional *feature vector* (where n is the number of rules in the grammar) whose ith dimension, $F_i(x,\sigma)$, indicates the number of times the ith transformation rule is used in parse σ. Furthermore, let p_i denote the probability for the ith transformation rule. We rewrite the joint likelihood of the sequence x and its parse σ in *log-linear* form as

$$\begin{aligned} P(x,\sigma) &= \prod_{i=1}^{n} p_i^{F_i(x,\sigma)} = \exp\left(\ln\left(\prod_{i=1}^{n} p_i^{F_i(x,\sigma)}\right)\right) \\ &= \exp\left(\sum_{i=1}^{n} F_i(x,\sigma) \ln p_i\right) = \exp(\mathbf{w}^T \mathbf{F}(x,\sigma)), \end{aligned} \quad (5)$$

where $w_i = \ln p_i$. Substituting this form into equation 3,

$$P(y \mid x) = \frac{\sum_{\sigma \in y} \exp(\mathbf{w}^T \mathbf{F}(x,\sigma))}{\sum_{\sigma' \in \Omega(x)} \exp(\mathbf{w}^T \mathbf{F}(x,\sigma'))}. \quad (6)$$

In this alternate form, we see that SCFGs are actually log-linear models with the restrictions that

(1) the parameters $w_1, \ldots, w_n$ correspond to log probabilities and hence obey a number of constraints (e.g., all parameters must be negative), and

(2) the features $F_1(x,\sigma), \ldots, F_n(x,\sigma)$ derive directly from the grammar; thus the types of features are restricted by the complexity of the grammar.

In both cases, the imposed restriction is unnecessary if we simply wish to ensure that the conditional probability in equation 6 is well-defined. Removing these restrictions, thus, is the basis for the CLLM framework. More generally, CLLMs are probabilistic models defined by equation 6, in the case that the parameters $w_1, \ldots, w_n$ may take on any real values, and the feature vectors are similarly unrestricted.[3]

[3]Note that conditional random fields (CRFs) are a specialized class of CLLMs whose probability distributions are defined in terms of graphical models [10].

2.2.2 Parameter estimation By definition, CLLMs parameterize the conditional probability $P(y \mid x)$ as a log linear function of the model's features $\mathbf{F}(x, \sigma)$, but they provide no manner for calculating $P(x, y)$. As a side effect, straight maximum likelihood techniques, which optimize this joint probability, do not apply to CLLMs.

Instead, CLLM training relies on the *conditional maximum likelihood* principle, in which one finds the parameters $\mathbf{w}_{\mathrm{CML}} \in \mathbb{R}^n$ that maximize the *conditional likelihood*[4] of the structures given the sequences,

$$\ell_{\mathrm{CML}}(\mathbf{w} : \mathcal{D}) = \prod_{i=1}^{m} P(y^{(i)} \mid x^{(i)}; \mathbf{w}). \tag{7}$$

Arguably, for prediction problems, conditional likelihood (or *discriminative*) training is more natural than joint likelihood (or *generative*) training as it focuses on finding parameters that give good predictive performance without attempting to model the distribution over input sequences x.

The mechanics of performing the probabilistic inference tasks required in the optimization of equation 7 follow closely the traditional inside and outside algorithms for SCFGs [2].

2.3 From energy-based models to CLLMs

Converting an SCFG to a CLLM by removing restrictions on the parameter vector $\mathbf{w}$ and training via conditional likelihood allows SCFGs to obtain many of the benefits of the discriminative learning approach. Straightforward conversions of this sort are routine in the machine learning literature and have recently been applied to RNA secondary structure alignment [21]. Such conversions, however, do not take full advantage of the expressivity of CLLMs. In particular, the ability of CLLMs to use generic feature representations means that in some cases, CLLMs can conveniently represent models which do not have compact parameterizations as SCFGs.

For example, the QRNA algorithm [18] attempts to capture the salient properties of standard thermodynamic models for RNA secondary structure, such as loop lengths and base-stacking, via an SCFG. This conversion, however, is only approximate. In particular, the usual energy rules [23,11] contain *terminal mismatch* terms describing the interaction between closing base pairs of helices and nucleotides in the adjacent loop. These interactions are ignored in QRNA, and more generally, are difficult to incorporate in SCFG models without considerably increasing grammar complexity. As the authors themselves note, QRNA underperforms compared to standard folders, highlighting the difficulty of building SCFGs on par with energy-based methods [18].

Contrastingly, the complex scoring terms of thermodynamic models transfer to CLLMs with no difficulties. In the standard model, the energy of a folding σ decomposes as the sum of energies for hairpin, interior, bulge, stacking pair, and multi-branch loops. In turn, the energy of each type of loop further decomposes as the sum of interaction energies over individual features of the sequence x and its parse σ. Thus, in the CLLM equivalent of standard thermodynamic scoring, the parameters $w_1, \ldots, w_n$ replace the interaction energy contributions for various secondary elements, and the features $F_1(x, \sigma), \ldots, F_n(x, \sigma)$ count the number of times a particular interaction term appears in the parse σ. This procedure is illustrated in Figures 1 and 2.

2.4 The CONTRAfold model

The CONTRAfold program implements a CLLM for RNA secondary structure prediction, following the general strategy for model construction outlined in the previous section. The features in CONTRAfold (see Figure 3) include:

(1) base pairs,
(2) helix closing base pairs,

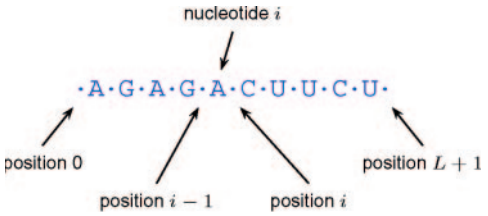


Fig. 1. Positions in a sequence of length $L = 10$. Here, let x_i denote the ith nucleotide of x. For ease of notation, we say that there are $L + 1$ *positions* corresponding to x—one position at each of the two ends of x, and $L - 1$ positions between consecutive nucleotides of x. We assign indices ranging from 0 to L for each position.

(3) hairpin lengths,
(4) helix lengths,
(5) bulge loop lengths,
(6) internal loop lengths,
(7) internal loop asymmetry,
(8) full two-dimensional table of internal loop scores,
(9) helix base pair stacking interactions,
(10) terminal mismatch interactions,
(11) single (dangling) base stacking,
(12) affine multi-branch loop scoring, and
(13) free bases.

To a large extent, the features above closely mirror the features employed in traditional thermodynamic models of RNA secondary structure. We point out a few key differences:

(1) CONTRAfold makes use of generic feature sets without incorporating ''special cases'' typical of complex thermodynamic scoring models, such as the popular Turner energy rules [11]. For instance, CONTRAfold
   – omits the bonus free energies for special case hairpin loops (specifically items (d) through (f) from the list in Figure 2).
   – does not contain a table exhaustively enumerating all possible 1×1, 1×2, 2×2, and 2×3 internal loops.

   While such features may be useful, they are more likely to lead to overfitting due to the large number of parameters that must be trained.[5] Incorporation of a small number of specially selected interactions which are known to be particularly important *a priori* is more feasible.

(2) Internal and bulge loop lengths are scored separately as a function of the lengths ℓ_1 and ℓ_2 of each side of the loop:

$$f_{\text{single length}}(\ell_1, \ell_2) = \begin{cases} w_{\text{bulge length}}[\ell_1 + \ell_2] & \text{if } \ell_1 \ell_2 = 0 \\ w_{\text{internal length}}[\ell_1 + \ell_2] & \text{otherwise} \\ \quad + w_{\text{internal asymmetry}}[|\ell_1 - \ell_2|] & \\ \quad + w_{\text{internal correction}}[\ell_1][\ell_2]. & \end{cases} \tag{8}$$

   In most thermodynamic models, only bulge and internal loop length score tables exist, whereas internal loop asymmetry is scored according to the Ninio equations [14]. Here, CONTRAfold learns an explicit scoring table $w_{\text{internal asymmetry}}[\cdot]$ for internal loop asymmetry in addition to a two-dimensional correction matrix $w_{\text{internal correction}}[\cdot][\cdot]$ for representing dependencies not captured by total loop length and asymmetry alone.

[4] In practice, we avoid overfitting by placing a zero-mean Gaussian regularization prior on the parameters, and selecting the variance of the prior using holdout cross-validation on training data only (see Results).

[5] This may be considered an advantage of physics-based methods; a hybrid approach which combines machine learning with physics-based prior knowledge may help alleviate the burden on the learning algorithm.

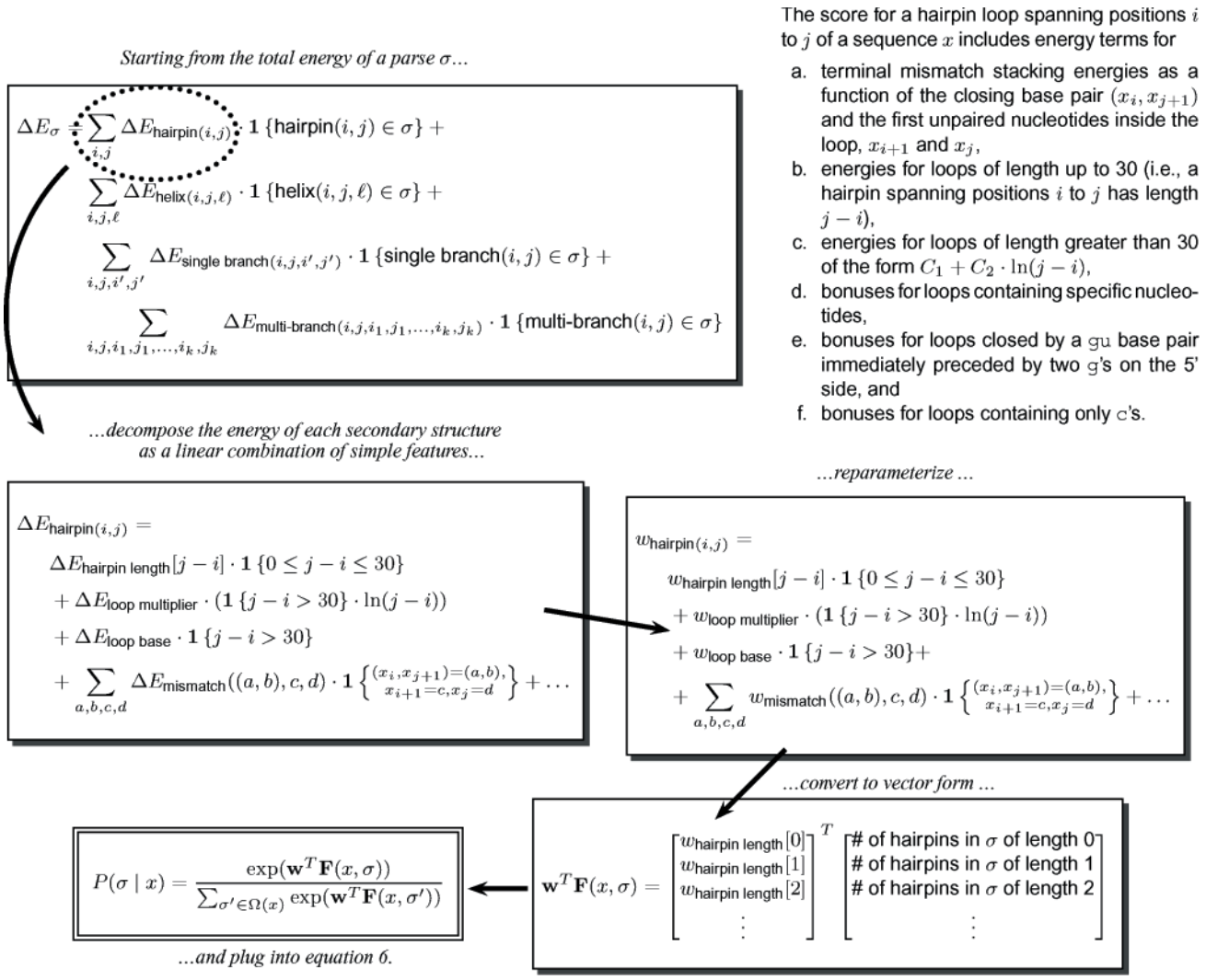


Fig. 2. The construction of a CLLM from an energy-based model. In short, the conversion process involves expressing the total energy of a parse σ as a linear function of counts for joint features $F_i(x, \sigma)$ of the sequence x and the parse σ. Once this is done, substituting into equation 6 gives a probabilistic model whose Viterbi parse is the minimum energy parse.

(3) Unlike typical energy minimization schemes, the energy of a helix consists not only of stacking interactions but also direct base pair interactions. Also, all combinations of nucleotide pairs are allowed, unlike the standard nearest neighbor model in which only canonical Watson-Crick or wobble gu pairs are permitted. Finally, CONTRAfold introduces new scoring terms for helix lengths (via an explicit scoring table for helices of length up to 5 and affine afterwards), which are not part of the standard nearest neighbor model.

(4) Since little is currently known about the energetics of free bases (bases which do not belong to any other loop in the secondary structure), they are typically ignored by energy-based folders. Here, CONTRAfold introduces two scoring parameters: $w_{\text{outer unpaired}}$ for scoring each free base, and $w_{\text{outer paired}}$ for scoring each base pair adjacent to a free base.

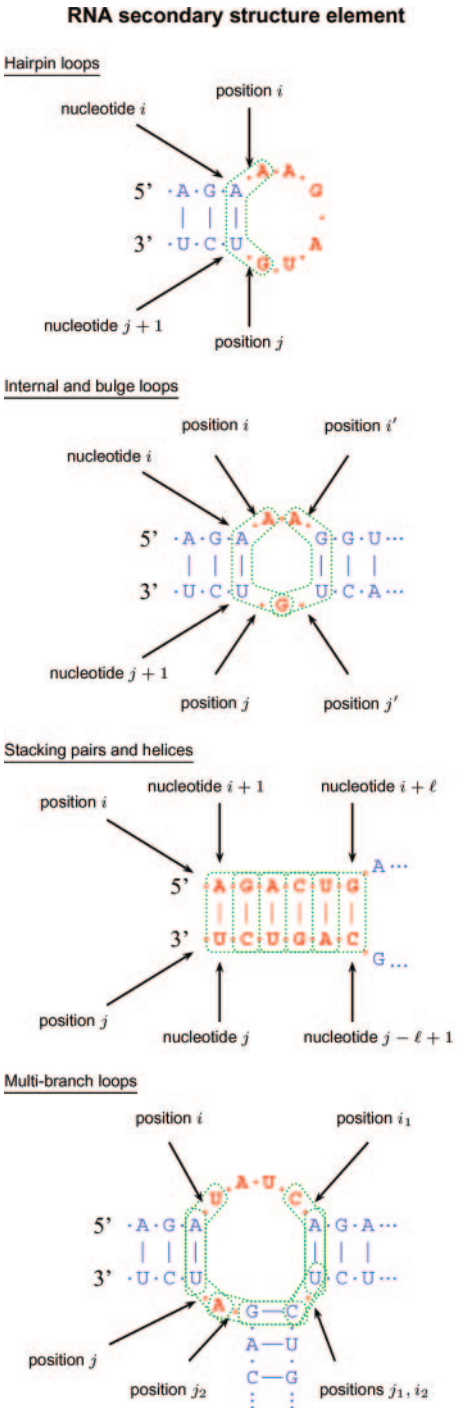
(5) For simplicity, CONTRAfold scores terminal mismatches for hairpins, bulges, and internal loops using the same parameters. CONTRAfold also does not account for coaxial stacking dependencies when scoring multi-branch loops. Like the special case hairpin loops mentioned earlier, making more specific scoring models by differentiating between these terminal mismatches may improve prediction accuracy.

2.5 Maximum expected accuracy parsing with sensitivity/specificity tradeoff

Most physics-based approaches to secondary structure prediction use dynamic programming to recover the structure with minimum free energy [26,7]. For probabilistic methods, the Viterbi algorithm (known as the CYK algorithm [2] for SCFGs) fulfills this function by finding the most likely parse,[6]

$$\hat{\boldsymbol{\sigma}}_{\text{viterbi}} = \arg\max_{\boldsymbol{\sigma} \in \Omega(x)} P(\boldsymbol{\sigma} \mid x; \mathbf{w}). \quad (9)$$

[6]For unambiguous grammars, the most likely parse is also the most likely secondary structure; however, this is not the case for ambiguous grammars [1,16].



Main components of CLLM scoring

$$w_{\text{hairpin}}(i,j) = w_{\text{terminal mismatch}}((x_i, x_{j+1}), x_{i+1}, x_j) + \begin{cases} w_{\text{hairpin length}}[j-i] & \text{if } 0 \le j-i \le 30 \\ w_{\text{loop base}} + w_{\text{loop multiplier}} \cdot \ln(j-i) & \text{if } j-i > 30. \end{cases}$$

Hairpin loop scoring in CONTRAfold accounts for terminal mismatch interactions between the hairpin closing base pair and the adjacent nucleotides in the hairpin loop (see left diagram), and hairpin lengths. Following energy-based models, we set the score for hairpins of length beyond 30 to be affine in the logarithm of the hairpin size.

$$\begin{aligned} w_{\text{single}}(i,j,i',j') &= f_{\text{single length}}(i'-i, j-j') \\ &+ w_{\text{terminal mismatch}}((x_i, x_{j+1}), x_{i+1}, x_j) \\ &+ w_{\text{terminal mismatch}}((x_{j'}, x_{i'+1}), x_{j'+1}, x_{i'}) \end{aligned}$$

Internal and bulge loops (also known as single branch loops) have exactly two adjacent base pairs. CONTRAfold scores internal and bulge loops by accounting for loop lengths and terminal mismatches (see left diagram). Loop lengths are scored according to equation 8. Terminal mismatches are scored in the same way as terminal mismatches in hairpin loops.

$$\begin{aligned} w_{\text{helix}}(i,j,\ell) &= \textstyle\sum_{k=1}^{\ell} w_{\text{base pair}}((x_{i+k}, x_{j+1-k})) \\ &+ \textstyle\sum_{k=1}^{\ell-1} w_{\text{stacking}}((x_{i+k}, x_{j+1-k}), (x_{i+k+1}, x_{j-k})) \\ &+ w_{\text{helix closing}}(x_{i+1}, x_j) + w_{\text{helix closing}}(x_{j-\ell+1}, x_{i+\ell}) \end{aligned}$$

CONTRAfold scores helices (regions of adjacent stacking base pairs) by including terms for the contribution of closing base pairs as well as each stacking and base pair interaction. Unlike the standard thermodynamic model, which allows only canonical Watson-Crick or wobble gu pairs, CONTRAfold allows for all possible nucleotide combinations.

$$\begin{aligned} w_{\text{multi}}(i,j,i_1,j_1,\ldots,i_m,j_m) &= w_{\text{multi base}} \\ &+ w_{\text{multi paired}} \cdot (m+1) + w_{\text{multi unpaired}} \cdot \ell \\ &+ w_{\text{stacking left}}((x_i, x_{j+1}), x_{i+1}) + w_{\text{stacking right}}((x_i, x_{j+1}), x_j) \\ &+ \textstyle\sum_{k=1}^{m} w_{\text{stacking left}}((x_{j_k}, x_{i_k+1}), x_{j_k+1}) \\ &+ \textstyle\sum_{k=1}^{m} w_{\text{stacking right}}((x_{j_k}, x_{i_k+1}), x_{i_k}) \end{aligned}$$

Multi-branch loops have at least three adjacent base pairs. Scoring of multi-branch loops in CONTRAfold accounts for the single base stacking interactions and the length dependence of multi-branch loops. For computational efficiency reasons [11], multi-branch loop length scores are affine in the number of unpaired nucleotides (ℓ) and the number of adjacent base pairs ($m+1$).

Fig. 3. Correspondence between energy-based model scoring and CLLM potentials in CONTRAfold. In each diagram, the nucleotides comprising the indicated RNA secondary structure element are shown in red. Green dotted lines indicate the groups of nucleotides involved in the terminal mismatch, helix stacking, or single base stacking interactions considered by CONTRAfold.

Here, we describe an alternative scheme that, for a given setting of a sensitivity/specificity tradeoff parameter γ, identifies the structure with *maximum expected accuracy*.

In particular, for a candidate structure $\hat{y}$ with true structure y, let $\text{accuracy}_\gamma(\hat{y}, y)$ denote the number of correctly unpaired positions in $\hat{y}$ (with respect to y) plus γ times the number of correctly paired positions in $\hat{y}$. Then, we wish to find,

$$\hat{y}_{\text{mea}} = \arg\max_{\hat{y}} \mathbb{E}_y[\text{accuracy}_\gamma(\hat{y}, y)], \tag{10}$$

where the expectation is taken with respect to the conditional distribution over structures of the sequence x.

To do this, let p_{ij} denote the conditional probability that the ith and jth nucleotides of sequence x base pair. Similarly, let $q_i = 1 - \sum_j p_{ij}$ be the conditional probability that the ith nucleotide is unpaired. The following recurrence computes $M_{1,L} = \max_y(E_y[\text{accuracy}_\gamma(\hat{y}_{\text{mea}}, y)])$:

$$M_{i,j} = \max \begin{cases} q_i & \text{if } i = j \\ q_i + M_{i+1,j} & \text{if } i < j \\ q_j + M_{i,j-1} & \text{if } i < j \\ \gamma \cdot 2p_{ij} + M_{i+1,j-1} & \text{if } i + 2 \leq j \\ M_{i,k} + M_{k+1,j} & \text{if } i \leq k < j. \end{cases} \quad (11)$$

Including the traceback for recovering the optimal structure, the parsing algorithm takes $O(L^3)$ time and $O(L^2)$ space.

Note that in the above algorithm, γ controls the balance between the sensitivity and specificity of the returned structure—i.e., higher values of γ encourage the parser to predict more base pairings whereas lower values of γ restrict the parser to predicting only base pairs for which the algorithm is extremely confident. When $\gamma = 1$, the algorithm maximizes the expected number of correct positions and is identical to the parsing technique used in Pfold [9]. As shown in the Results section, by allowing γ to vary, we may adjust the sensitivity and specificity of the parsing algorithm as desired.

3 RESULTS

To assess the suitability of CLLMs as models for RNA secondary structure, we performed a series of cross-validation experiments using known consensus secondary structures of noncoding RNA families taken from the Rfam database [5,6]. Specifically, version 7.0 of Rfam contains seed multiple alignments for 503 noncoding RNA families, and consensus secondary structures for each alignment either taken from a previously published study in the literature or predicted using automated covariance-based methods.

To establish ''gold-standard'' data for training and testing, we first removed all seed alignments with only predicted secondary structures, retaining the 151 families with secondary structures from the literature. For each of these families, we then projected the consensus family structure to every sequence in the alignment, and retained the sequence/structure pair with the lowest combined proportion of missing nucleotides and non-{au, cg, gu} base pairs. The end result was a set of 151 independent examples, each taken from a different RNA family.

3.1 Comparison to generative training

In our first experiment, we took nine different grammar-based models (G1-G8, G6s) from a recent study by Dowell and Eddy on the performance of simple SCFGs for RNA secondary structure prediction [1]. For each grammar, we took the original SCFG and constructed an equivalent CLLM. We then applied a two-fold cross-validation procedure to compare the performance of SCFG (generative) and CLLM (discriminative) parameter learning.

In particular, we partitioned the 151 selected sequence-structure pairs randomly into two approximately equal-sized ''folds.'' For any given setting of the MEA trade-off parameter γ, we used parameters trained on sequences from one fold[7] to perform predictions for all sequences from the other fold. For each tested example, we computed sensitivity and specificity (PPV)[8], defined as

$$\text{sensitivity} = \frac{\text{number of correct base pairings}}{\text{number of true base pairings}} \quad (12)$$

$$\text{specificity} = \frac{\text{number of correct base pairings}}{\text{number of predicted base pairings}}. \quad (13)$$

By repeating this cross-validation procedure for values of $\gamma \in \{2^k : -5 \leq k \leq 10\}$, we obtained a receiver operating characteristic (ROC) curve for each grammar. We report the estimated area under each curve (see Table 1). In 7 out of 9 grammars, the CLLM outperforms its SCFG counterpart.

Table 1. Comparison of generative and discriminative model structure prediction accuracy.

Grammar	Generative	Discriminative	Difference
G1	0.0392	**0.2713**	+0.2321
G2	0.3640	**0.5797**	+0.2157
G3	**0.4190**	0.4159	−0.0031
G4	**0.1361**	0.1350	−0.0011
G5	0.0026	**0.0031**	+0.0005
G6	0.5446	**0.5600**	+0.0154
G7	0.5456	**0.5582**	+0.0126
G8	0.5464	**0.5515**	+0.0051
G6s	0.5501	**0.5642**	+0.0141

Each number in the table represents the area under the ROC curve of an MEA-based parser using the indicated model. As seen below, the discriminative model consistently outperforms its generative counterpart.

Using a similar cross-validation protocol, we also found that MEA parsing outperforms the Viterbi algorithm on average for both the generative and discriminative models. In particular, when an algorithm A achieves better sensitivity and specificity than algorithm B, we say that A *dominates* B. On 7 out of 9 generatively-trained grammars and 9 out of 9 discriminatively-trained grammars, we found a γ for which the MEA parsing algorithm dominates the Viterbi algorithm (see Table 2).

3.2 Comparison to other methods

Next, we compared the performance of CONTRAfold with a number of leading probabilistic and free energy minimization methods. In particular, we benchmarked Mfold v3.2 [26], ViennaRNA v1.6 [7], PKNOTS v1.05 [17][9], Pfold v3.2 [9], and ILM [20], using default parameters for each program.[10] Whenever a program returned multiple possible structures (e.g., Mfold), we scored only the structure with minimum predicted free energy.

[7]To determine smoothing parameters (for SCFGs) or regularization constants (for CLLMs), we used conditional log-likelihood on a holdout set taken from the training data as an estimate of the generalization ability of the learned model, and found the optimal setting of the desired parameter using a golden section search [15].

[8]We considered only au, cg, and gu base pairs since many of the energy-based folders cannot predict other types of base pairings as a consequence of the nearest neighbor model.

[9]Because of the large size of some of the sequences in our dataset, we disabled pseudoknot prediction for PKNOTS.

[10]Note that while all tools listed support single sequence RNA secondary structure prediction, not all were designed specifically for single sequence prediction. Pfold, for instance, was developed in the context of multiple sequence structure prediction; similarly, ILM and PKNOTS were developed for prediction of RNA structures with pseudoknots, and so might fare better on sequences where pseudoknot interactions play a more important role.

Table 2. Comparison of generative and discriminative model structure prediction accuracy

Grammar	Generative Viterbi Sens (spec)	Generative MEA Sens (spec)	Discriminative Viterbi Sens (spec)	Discriminative MEA Sens (spec)
G1	**0.41 (0.27)**	0.18 (0.11)	0.40 (0.28)	**0.48 (0.33)**
G2	0.53 (0.36)	0.53 (0.36)	0.63 (0.48)	**0.67 (0.64)**
G3	0.46 (0.48)	**0.56 (0.51)**	0.45 (0.46)	**0.54 (0.53)**
G4	0.21 (0.17)	**0.33 (0.23)**	0.21 (0.17)	**0.34 (0.23)**
G5	0.03 (0.04)	**0.06 (0.04)**	0.02 (0.03)	**0.06 (0.04)**
G6	0.60 (0.61)	**0.62 (0.63)**	0.61 (0.62)	**0.62 (0.67)**
G6s	0.60 (0.62)	**0.62 (0.64)**	0.62 (0.63)	**0.65 (0.65)**
G7	0.58 (0.63)	**0.63 (0.63)**	0.58 (0.62)	**0.63 (0.67)**
G8	0.58 (0.60)	**0.63 (0.62)**	0.58 (0.61)	**0.65 (0.62)**

In each case, γ was adjusted for MEA parsing to allow a direct comparison with Viterbi, and the dominant parsing method is shown in bold. Finally, note that the results for MEA reflect only a single choice of γ rather than the entire ROC curve, so one should refer to Table 1 for a more reliable comparison of generative and discriminative MEA accuracy.

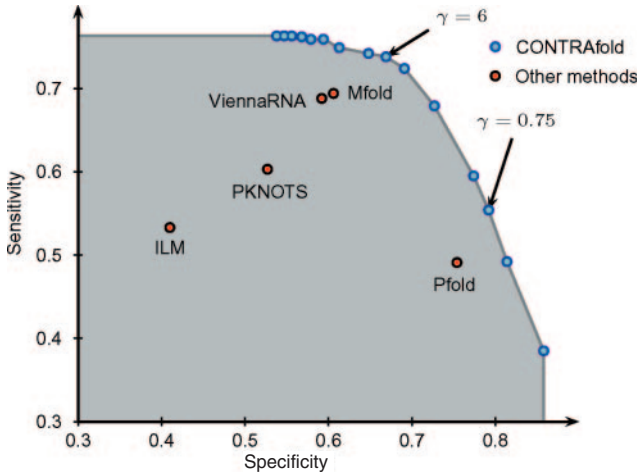


Fig. 4. ROC plot comparing sensitivity and specificity for several RNA structure prediction methods. CONTRAfold performance was measured at several different settings of the γ parameter, which controls the tradeoff between the sensitivity and specificity of the prediction algorithm. As shown above, CONTRAfold achieves the highest sensitivity at each level of specificity.

Unlike the other programs in our comparison, CONTRAfold's use of the maximum expected accuracy algorithm for parsing allows it to optimize for either higher sensitivity or higher specificity via the constant γ. In Figure 4, we varied the choice of γ for the parsing algorithm so as to allow CONTRAfold to achieve many different trade-offs between sensitivity and specificity; some of these trade-offs allow for unambiguous comparisons between CONTRAfold and existing methods.

As shown in Tables 3 and 4, CONTRAfold outperforms existing probabilistic and energy-based structure prediction methods without relying on the thousands of experimentally measured parameters common among free energy minimization techniques. For $\gamma = 6$ in particular, CONTRAfold achieves statistically significant improvements of over 4% in sensitivity and 6% in specificity relative to the best current method, Mfold. This demonstrates not only the quality of the underlying model but also the effectiveness of the parsing mechanism for providing a sensitivity/specificity trade-off.

Table 3. Accuracies of leading secondary structure prediction methods

Method	Sensitivity	Specificity	Time (s)
CONTRAfold (γ=6)	**0.7377**	**0.6686**	224
Mfold	0.6943	0.6063	62
ViennaRNA	0.6877	0.5922	**8**
PKNOTS	0.6030	0.5269	460
ILM	0.5330	0.4098	22
CONTRAfold (γ =0.75)	**0.5540**	**0.7920**	224
Pfold	0.4906	0.7535	**22**

Table 4. Performance of CONTRAfold relative to leading secondary structure prediction methods

Method	Sensitivity +	Sensitivity −	Sensitivity p-value	Specificity +	Specificity −	Specificity p-value
Mfold	34	69	0.00081	51	77	0.0271
ViennaRNA	30	72	4.9×10^{-5}	44	82	0.00098
PKNOTS	17	94	5.5×10^{-13}	26	104	1.5×10^{-11}
ILM	20	101	3.6×10^{-13}	12	126	6.8×10^{-22}
Pfold	38	72	0.0017	41	64	0.0318

Mfold, ViennaRNA, PKNOTS, and ILM were compared to CONTRAfold ($\gamma = 6$). Pfold was compared to CONTRAfold ($\gamma = 0.75$). The numbers in the +/− columns indicate the number of times the method achieved higher (+) or lower (−) sensitivity/specificity than CONTRAfold. p-values were calculated using the sign test.

3.3 Feature assessment

To understand the importance of various features to the CONTRAfold model, we performed an abrasion analysis in which we removed various sets of features from the model and assessed the change in total ROC area for the MEA parser. As seen in Table 5, the performance of CONTRAfold degrades as features are removed from the model.

Interestingly, even the weakest model from Table 5, which includes only features for hairpin, bulge, internal, multi-branch loops (without accounting for internal loop asymmetry), helix closing base pairs, and helix base pairs, achieves a respectable ROC area of 0.6003. In fact, this crippled version of CONTRAfold, which does not even account for helix stacking interactions, manages to obtain sensitivity and specificity values of 0.7006 and 0.6193, respectively, accuracy statistically indistinguishable from Mfold.

3.4 Learned versus measured parameters

In many respects, the general techniques employed by CLLMs are reminiscent of many previously described algorithms. For instance,

Table 5. Abrasion analysis of CONTRAfold model

Variant	ROC area	Decrease
CONTRAfold	0.6433	n/a
(without single base stacking)	0.6416	0.0017
(without helix lengths)	0.6370	0.0063
(without terminal mismatch penalties)	0.6362	0.0071
(without full internal loop table)	0.6336	0.0097
(without helix stacking)	0.6276	0.0157
(without outer)	0.6271	0.0162
(without internal loop asymmetry)	0.6134	0.0299
(without all of the above)	0.6003	0.0430

A large decrease in ROC area suggests that the corresponding removed features play an important role in RNA secondary structure. However, the reverse is not true: small decreases in accuracy (such as seen for single base stacking) may simply mean that CONTRAfold was less effective in leveraging that feature for prediction.

(a) Learned

$5' \longrightarrow 3'$: $\mathtt{a}X$ / $\mathtt{u}Y$; $3' \longleftarrow 5'$

X \ Y	a	c	g	u
a	0.48	0.38	0.34	-1.24
c	0.27	0.33	-1.74	0.34
g	0.34	-1.63	0.27	-0.74
u	-1.26	0.32	-0.89	0.32

(b) Experimental

$5' \longrightarrow 3'$: $\mathtt{a}X$ / $\mathtt{u}Y$; $3' \longleftarrow 5'$

X \ Y	a	c	g	u
a	.	.	.	-0.90
c	.	.	-2.20	.
g	.	-2.10	.	-0.60
u	-1.10	.	-1.40	.

Fig. 5. Comparison of learned and experimentally measured stacking energies. (**a**) A portion of the helix stacking parameters learned by CONTRAfold, scaled by $-RT$ at $T = 310.15$ K $= 37°$C. (**b**) A portion of the helix stacking energies from the Turner 3.0 energy rules [11], as taken from the Mfold package [26].

the inside-outside algorithms inspired by SCFGs bear close relation to McCaskill's procedure for computing base-pairing probabilities via the partition function [12]. Indeed, one may be tempted to draw direct analogies between the parameters of energy-based models and the parameters learned by the CLLM (appropriately scaled by $-RT$, the negated product of the universal gas constant and absolute temperature).

As shown in Figure 5, in some cases one can find a good correlation between parameters learned by CONTRAfold and those measured experimentally. Differences between learned parameters and measured values, however, are not necessarily diagnostic of errors in the laboratory measurements. Roughly speaking, the parameters learned by CLLMs reflect the degree of enrichment of their corresponding features in training set secondary structures. Therefore, parameters which do not appear often in training set structures will have smaller parameter values, regardless of their actual energetic contribution to real RNA structures. Additionally, Gaussian prior regularization (see footnote to Section 2.2.2), reduces the magnitude of less confident parameters to prevent overfitting. Finally, CLLM learning compensates for dependencies between parameters so as to maximize the overall conditional likelihood of the training set; thus, the values learned for one parameter will depend greatly on the other parameters in the model.

4 DISCUSSION

In this paper, we presented CONTRAfold, a new RNA secondary structure prediction method based on conditional log-linear models (CLLMs). Like previous structure prediction methods based on probabilistic models, CONTRAfold relies on statistical learning techniques to optimize model parameters according to a training set. Unlike its predecessors, however, CONTRAfold uses a discriminative training objective and flexible feature representations in order to achieve accuracies exceeding those of the current best physics-based structure predictors.

As a modeling framework for RNA secondary structure prediction, CLLMs provide many advantages over physics-based models and previous probabilistic approaches, ranging from ease of parameter estimation to the ability to incorporate arbitrary features. It is only natural, then, to suspect that these advantages will carry over to related problems as well. For instance, most current methods for multiple sequence RNA secondary structure prediction either take a purely probabilistic approach or attempt to combine physics-based scoring with covariation information in an ad hoc way. In contrast, the CLLM methodology provides a principled framework for combining the rich feature sets of physics-based methods with the predictive power of sequence covariation.

To date, SCFGs and their extensions provide the foundation for many standard computational techniques for RNA analysis, ranging from modeling of specific RNA families to noncoding RNA detection to RNA structural alignment. In each of these cases, CLLMs provide principled alternatives to SCFGs which take advantage of complex features of the input data when making predictions. Extending the CLLM methodology to these cases provides an exciting avenue for future research.

ACKNOWLEDGEMENTS

We thank B. Knudsen for assisting us with Pfold benchmarking, S. R. Eddy and A. Laederach for helpful comments, S. S. Gross and G. Asimenos for helpful discussions regarding algorithms and implementation, and A. F. Novak for assistance in editing the manuscript. CBD was supported by an NDSEG fellowship. Work in the Batzoglou laboratory is supported in part by NSF grant EF-0312459, NIH grant U01-HG003162, the NSF CAREER Award, and the Alfred P. Sloan Fellowship.

REFERENCES

[1] R.D. Dowell and S.R. Eddy. Evaluation of several lightweight stochastic context-free grammars for RNA secondary structure prediction. *BMC Bioinformatics* 5(71), 2004.

[2] R. Durbin, S.R. Eddy, A. Krogh, and G. Mitchison. *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids*. Cambridge University Press, 1998.

[3] B. Furtig, C. Richter, J. Wohnert, and H. Schwalbe. NMR spectroscopy of RNA *Chembiochem*., 4(10): 936–962, 2003.

[4] P. P. Gardner and R. Giegerich. A comprehensive comparison of comparative RNA structure prediction approaches *BMC Bioinformatics* 5(140), 2004.

[5] S. Griffiths-Jones, A. Bateman, M. Marshall, A. Khanna, and S. R. Eddy. Rfam: an RNA family database. *Nucleic Acids Res*., 31(1):439–441, 2003.

[6] S. Griffiths-Jones, S. Moxon, M. Marshall, A. Khanna, S.R. Eddy, and A. Bateman. Rfam: annotating non-coding RNAs in complete genomes. *Nucleic Acids Res.*, 33:D121–D124, 2005.

[7] I.L. Hofacker, W. Fontana, P.F. Stadler, L.S. Bonhoeffer, and P. Schuster. Fast folding and comparison of RNA secondary structures (The Vienna RNA Package). *Monatsh Chem.*, 125:167–188, 1994.

[8] B. Knudsen and J. Hein. RNA secondary structure prediction using stochastic context-free grammars and evolutionary history. *Bioinformatics*, 15(6): 446–454, 1999.

[9] B. Knudsen and J. Hein. Pfold: RNA secondary structure prediction using stochastic context-free grammars. *Nucleic Acids Res.*, 31(13): 3423–3428, 2003.

[10] J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proc. 18th ICML*, pages 282–289, 2001.

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

[12] J.S. McCaskill. The equilibrium partition function and base pair binding probabilities for RNA secondary structure. *Biopolymers*, 29: 1105–1119, 1990.

[13] V. Moulton. Tracking down noncoding RNAs. *Proc. Nat Acad. Sci. USA,* 102(7):2269–2270, 2005.

[14] C. Papanicolaou, M. Gouy, and J. Ninio. An energy model that predicts the correct folding of both the tRNA and the 5S RNA molecules. *Nucleic Acids Res.*, 12(1 Pt 1):31–44, 1984.

[15] W.H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery. *Numerical Recipes in C: The Art of Scientific Computing* Cambridge UP, New York, NY, USA, 1992.

[16] J. Reeder, P. Steffen, and R. Giegerich. Effective ambiguity checking in biosequence analysis. *BMC Bioinformatics*, 6(153), 2005.

[17] E. Rivas and S.R. Eddy. A dynamic programming algorithm for RNA structure prediction including pseudoknots. *J. Mol. Biol.*, 285:2053–2068, 1999.

[18] E. Rivas and S.R. Eddy. Secondary structure alone is generally not statistically significant for the detection of noncoding RNAs. *Bioinformatics*, 16(7):583–605, 2000.

[19] J.M. Rouillard, M. Zuker, and E. Gulari. OligoArray 2.0: design of oligonucleotide probes for DNA microarrays using a thermodynamic approach. *Nucleic Acids Res.*, 31(12): 3057–3062, 2003.

[20] J. Ruan, G.D. Stormo, and W. Zhang. An iterated loop matching approach to the prediction of RNA secondary structures with pseudoknots. *Bioinformatics*, 20(1): 58–66, 2004.

[21] K. Sato and Y. Sakakibara. RNA secondary structural alignment with conditional random fields. *Bioinformatics*, 21(Suppl 2):ii237–ii242, 2005.

[22] I. Tinoco, O.C. Uhlenbeck, and M.D. Levine. Estimation of secondary structure in ribonucleic acids. *Nature*, 230:362–367, 1971.

[23] D.H. Turner, N. Sugimoto, and S.M. Freier. RNA structure prediction. *Ann. Rev. Biophys. Biophys. Chem.,* 17:167–192, 1988.

[24] M.T. Wolfinger, W.A. Svrcek-Seiler, C. Flamm, I.L. Hofacker, and P.F. Stadler. Efficient computation of RNA folding dynamics. *J. Phys. A: Math Gen,* 37: 4731–4741, 2004.

[25] X. Ying, H. Luo, J. Luo, and W. Li. RDfolder: a web server for prediction of RNA secondary structure. *Nucleic Acids Res.,* 32(Web Server Issue):W150–W153, 2004.

[26] M. Zuker. Mfold web server for nucleic acid folding and hybridization prediction. *Nucleic Acids Res.,* 31(13):3406–3415, 2003.