Daniel A. Woods

CS229 Final Project Writeup

December 16, 2005

Original Project Description

Title:
A Discriminative Learning Model for RNA Secondary Structure Prediction

Non-CS229 Collaborators:
Chuong (Tom) Do - Andrew Ng Lab

Empirical discovery of RNA secondary structure is expensive and time consuming, but is a necessary part of exploring function. Software tools exist for performing these predictions, the best of which either heuristic physics modeling or generative learning models. Currently, the best of each are approximately equal in performance.

While perfect predictions would require physics modeling well beyond our current computational capabilities, current levels of performance are much lower than perfect. I believe it may be possible to create a program better than the current machine learning methods by making two improvements:

1 - Current methods model RNA sequence and secondary structure as stochastic context-free grammars, and then use a generative learning model to find the most likely parse (and, therefore, the most likely structure). As we learned in class, discriminative models generally enjoy higher performance than generative learning models. This implies that performance may increase if discriminative learning were implied on top of the same stochastic context free grammar model of RNA sequence and secondary structure.

2 - Current software tools return the most likely structure of a given RNA molecule. However, it may be possible that a particular substructure is most likely among all possible structures, but it simply does not occur in the most likely overall structure. In order to increase overall predictive accuracy, I would prefer to return the most likely structure on a part-by-part basis rather than to return the most likely overall structure. I believe this would be more useful biologically because software predictions are never assumed accurate, but rather are the first step leading to manual refinement.

Background

The first thing we did was to find the most recent work done on the subject of using machine learning applied to RNA secondary structure prediction, which turns out to be a recent paper by Robin D Dowell and Sean R Eddy [1]. It models RNA secondary structure as a Stochastic Context-free Grammar (SCFG), and learns using a model very similar to an HMM, substituting the SCFG in place of the state machine. This is a generative model, as alluded to in the original proposal.

An HMM can be converted to a discriminative model using using Conditional Random Fields (CRFs) [2]. Our plan is to improve on Dowell and Eddy's algorithm by doing what CRFs do improve on to HMMs. The problem is very analogous and CRFs translate over nicely without any significant mathematical obstacles.

SCFGs

Here is a sample grammar from Eddy's paper, which is referred to as "G1":

$$\text{G1} : S \rightarrow aS\hat{a} \mid aS \mid Sa \mid SS \mid \varepsilon$$

Here, the $aS\hat{a}$ production refers to a pairing of two produced bases a and $\hat{a}$. This rule is actually shorthand for all the paired productions that are possible (A-A, A-C, A-G..., although some of these pairings are chemically impossible, we let the algorithm learn this fact rather than enforce it manually). This SCFG is a straightforward and simple representation of the structures possible with RNA. However, it performs extremely poorly as will be shown after introduction of another SCFG, "G6":

$$\begin{aligned} \text{G6} : S &\rightarrow LS \mid L \\ L &\rightarrow aF\hat{a} \mid a \\ F &\rightarrow aF\hat{a} \mid LS \end{aligned}$$

The performance of these, compared to *mfold*, is as follows:

Generative G1 : 17(12)
Generative G6 : 47(45)
mfold v3.1.2 : 56(48)

Note: these scores are given in the form *sensitivity (specificity)*, where *sensitivity* refers to the percentage of correct pairing that were predicted and *specificity* refers to the percentage of predicted pairings that are correct.

Eddy does evaluate several other grammars, but G6 is relatively simple and performs nearly as well as the best, so it was selected as the first candidate to create a discriminative model for.

Predictions

Given a set of weights, the probability of a parse y given a sequence x can be calculated as follows:

$$P(Y = y|X = x) = \frac{e^{w^T f(x,y)}}{\sum\limits_{y' \in \mathcal{Y}} e^{w^T f(x,y')}}$$

Here, $f(x, y)$ refers to a vector of feature counts for x and y, and w is the set of weights which must be learned (see "Training" section). $\mathcal{Y}$ refers to the set of all possible parses, as used in the denominator to form the partition function.

More generally, what we need is to find the probability that y is part of some set of parses $\mathcal{A}$. For example (as will be important for posterior decoding), we can use this generalization to find the probability that the bases at locations i and j are paired by calculating the probability that y is in the set of all parses in which i and j are paired. This more general form is as follows:

$$P(Y \in \mathcal{A}|X = x) = \frac{\sum\limits_{y' \in \mathcal{A}} e^{w^T f(x,y)}}{\sum\limits_{y' \in \mathcal{Y}} e^{w^T f(x,y')}}$$

Training

Training refers to optimization of w according to a training set, and subject to the array of regularization parameters C (w has a Gaussian prior). Optimizing w requires taking a gradient of the likelihood of the correct parse (or parses, in the case of ambiguous grammars) with respect to w:

$$\begin{aligned} \bigtriangledown_w \ell(w) &= \sum_{y \in \mathcal{A}} f(x,y) \frac{e^{w^T f(x,y)}}{\sum\limits_{y' \in \mathcal{Y}} e^{w^T f(x,y')}} - \sum_{y \in \mathcal{Y}} f(x,y) \frac{e^{w^T f(x,y)}}{\sum\limits_{y' \in \mathcal{Y}} e^{w^T f(x,y')}} - 2C \cdot w \\ &= E_{y \sim P(Y|X=x, Y \in \mathcal{A})}[f(x,y)] - E_{y \sim P(Y|X=x)}[f(x,y)] - 2C \cdot w \end{aligned}$$

Although a simple gradient decent could be used here, we opted for L-BFGS, which performs the same function but converges more quickly.

Posterior Decoding

As mentioned in the original statement of the problem, we would like to maximize the percent accuracy of our predictions rather than simply returning the single parse which is most likely. The latter could be calculated by finding the parse with the highest probability, but our approach requires an additional step.

Using techniques alluded to in the "Predictions" section, we can find for any i and j the probability $p_{i,j}$ that the bases at those locations will be paired. Similarly, we can find for any location i the probability p_i that its corresponding base is unpaired.

It is a straightforward dynamic programming implementation to maximize the overall score given these probabilities according to the following recurrence:

$$\text{score}(i,j) = \max \begin{cases} 0 & \text{if } i = j \\ p_{i+1} + \text{score}(i+1, j) & \text{if } i < j \\ p_j + \text{score}(i, j-1) & \text{if } i < j \\ m(p_{i+1,j} + p_{j,i+1}) + \text{score}(i+1, j-1) & \text{if } i+1 < j \\ \text{score}(i,k) + \text{score}(k,j) & \text{if } i < k < j \end{cases}$$

Here, m is a parameters that can be used to adjust the overall propensity to create pairings. As will be seen in the "Results" section, we adjusted m until our discriminative model had the same sensitivity as Eddy's generative model, thus allowing a straightforward comparison on the basis of specificity.

Results

Our results showed a significant improvement on Eddy's generative model for the G6 grammar. We performed both of these tests ourselves, using separate data sets for training and testing.

Generative G6 : 47.9(44.6)
Discriminative G6 : 47.8(52.0)

Again, we adjusted m until the sensitivity levels were very close, so that an even comparison could be made of specificity levels.

Conclusions

Part of what originally caught our attention about this project was the observation that Eddy's performance was comparable to the best physics-based algorithms using something extremely simple and lightweight. Now that we have demonstrated that the change to a discriminative model really does offer significant improvement in this simple case, we will attempt to create a much richer model in hopes of achieving results similar to (or even better than) those provided by *mfold* (which has been the standard for over two decades).

RNA secondary structure prediction is critical to medical research, and an improved algorithm would be extremely helpful in aiding advancement of biology and medicine.

References

1. Robin D Dowell, Sean R Eddy; **Evaluation of several lightweight stochastic context-free grammars for RNA secondary structure prediction.** *BMC Bioinformatics* 2004, **5** 71
2. John Lafferty, Andrew McCallum, Fernando Pereira: **Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data.** *Proc. 18th International Conf. on Machine Learning* 2001
3. Michael Zucker: **Mfold web server for nucleic acid folding and hybridization prediction.** Nucleic Acids Research 2003, **31** 34-6-3415