Prospects for recurrent neural network models to learn RNA biophysics from high-throughput data

Michelle J Wu[1], Johan OL Andreasson[2,3], Wipapat Kladwang[2], William J Greenleaf[3,4], Eterna participants, Rhiju Das[2,5*]

[1] Biomedical Informatics Training Program, Stanford University School of Medicine
[2] Department of Biochemistry, Stanford University School of Medicine
[3] Department of Genetics, Stanford University School of Medicine
[4] Department of Applied Physics, Stanford University
[5] Department of Physics, Stanford University

* Corresponding author
Email: rhiju@stanford.edu

Abstract

RNA is a functionally versatile molecule that plays key roles in genetic regulation and in emerging technologies to control biological processes. Computational models of RNA secondary structure are well-developed but often fall short in making quantitative predictions of the behavior of multi-RNA complexes. Recently, large datasets characterizing hundreds of thousands of individual RNA complexes have emerged as rich sources of information about RNA energetics. Meanwhile, advances in machine learning have enabled the training of complex neural networks from large datasets. Here, we assess whether a recurrent neural network model, Ribonet, can learn from high-throughput binding data, using simulation and experimental studies to test model accuracy but also determine if they learned meaningful information about the biophysics of RNA folding. We began by evaluating the model on energetic values predicted by the Turner model to assess whether the neural network could learn a representation that recovered known biophysical principles. First, we trained Ribonet to predict the simulated free energy of an RNA in complex with multiple input RNAs. Our model accurately predicts free energies of new sequences but also shows evidence of having learned base pairing information, as assessed by *in silico* double mutant analysis. Next, we extended this model to predict the simulated affinity between an arbitrary RNA sequence and a reporter RNA. While these more indirect measurements precluded the learning of basic principles of RNA biophysics, the resulting model achieved sub-kcal/mol accuracy and enabled design of simple RNA input responsive riboswitches with high activation ratios predicted by the Turner model from which the training data were generated. Finally, we compiled and trained on an experimental dataset comprising over 600,000 experimental affinity measurements published on the Eterna open laboratory. Though our tests revealed that the model likely did not learn a physically realistic representation of RNA interactions, it nevertheless achieved good performance of 0.76 kcal/mol on test sets with the application of transfer learning and novel sequence-specific data augmentation strategies. These results suggest that recurrent neural network architectures, despite being naïve to the physics of RNA folding, have the potential to capture complex biophysical information. However, more diverse datasets, ideally involving more direct free energy measurements, may be necessary to train *de novo* predictive models that are consistent with the fundamentals of RNA biophysics.

Author Summary

The precise design of RNA interactions is essential to gaining greater control over RNA-based biotechnology tools, including designer riboswitches and CRISPR-Cas9 gene editing. However, the classic model for energetics governing these interactions fails to quantitatively predict the behavior of RNA molecules. We developed a recurrent neural network model, Ribonet, to quantitatively predict these values from sequence alone. Using simulated data, we show that this model is able to learn simple base pairing rules, despite having no *a priori* knowledge about RNA folding encoded in the network architecture. This model also enables design of new switching RNAs that are predicted to be effective by the "ground truth" simulated model. We applied transfer learning to retrain Ribonet using hundreds of thousands of RNA-RNA affinity measurements and demonstrate simple data augmentation techniques that improve model performance. At the same time, data diversity currently available set limits on Ribonet's accuracy. Recurrent neural networks are a promising tool for modeling nucleic acid biophysics and may enable design of complex RNAs for novel applications.

Background

RNA is involved in myriad functional roles in biological systems, from control of gene expression to splicing to protein synthesis, and serves as a key player in biotechnology tools, from RNA silencing to CRISPR/Cas9 to reengineered ribosomes.[1–8] These RNAs interact with each other and with other biomolecules in complex, dynamic ways that are highly dependent on the cellular environment in which they reside.[9,10] Understanding and controlling these processes requires a quantitative understanding of the energetics that govern these interactions. The classic nearest-neighbor model of nucleic acid energetics developed by Turner and colleagues is based upon careful parameterization of the energies of each possible motif formed.[11–13] However, this model makes strong assumptions about motif energies and interactions and relies on modest amounts of optical melting data.[11,14] Limited by the need for expert tuning and low measurement throughput, this model falls short in accurately predicting these interactions in many applications.[15,16]

Recently, technological developments have enabled high-throughput biophysical measurements of tens of thousands if individual RNAs, a rich source of data for refining models of RNA energetics. These techniques, which make use of repurposed Illumina sequencing tools, enable the measurement of RNA-RNA and RNA-protein affinities over millions of individual clusters on an RNA array in a wide variety of experimental conditions.[17–20] However, the complex relationship of these measurements to underlying energetic parameters makes it difficult to refine individual motif energies using these rich datasets. Thus, alternative models and optimization methods are needed to fully leverage these datasets for building quantitative models of RNA energetics.

Toward this end, advances in training neural network models have enabled extraction of relevant features from large, complex datasets. These deep learning models have recently propelled major improvements in many machine learning tasks, such as image classification, machine translation, and speech recognition.[21–24] They have also been successfully applied to modeling biological systems, although less work has been carried out in regression as opposed to classification tasks.[25–28] Among neural network architectures that have been successful in deep learning, recurrent neural networks (RNNs) have had a particularly large impact on processing temporal or sequence data, such as natural language or speech.[22,29,30] This architecture might be expected to naturally transfer to processing biological sequence data and has the potential to encode the level of complexity involved in making quantitative predictions of RNA interactions. Furthermore, such neural networks may be able to learn and predict non-nearest-neighbor effects and tertiary interactions that are not captured with classic physics-based nearest-neighbor models. Neural networks also offer the prospect of dramatic decreases in calculation speed for long RNA sequences or multi-RNA complexes, which currently incur computational expenses that scale with worse than polynomial time in sequence length.[31]

Here, we evaluate the effectiveness of a recurrent neural network, Ribonet, for predicting the energetics governing RNA interactions and recovering behavior consistent with biophysical principles. Specifically, our goal is to predict the energetics of RNAs in complex with other input RNAs at various concentrations. Using simulated free energy data, we demonstrate that, given enough training examples, Ribonet can capture base pairing information, despite having no *a priori* information about RNA biophysics and no explicit secondary structure information for the training molecules. In contrast, simulated affinity data, which involve free energy differences over multiple states and thus are a more complex function of secondary structure, were too indirect to allow Ribonet to learn this base-pairing level information. Nevertheless, these affinity-trained Ribonet models were accurate enough to facilitate design of novel RNA switches that modulate their affinity to a reporter in response to binding of an input. Further, we use transfer learning and data augmentation to train models on experimental RNA-RNA and RNA-protein affinity measurements, achieving good predictive performance on held-out test sets but failing to recover known principles of RNA biophysics. Together, these results suggest that RNNs are a promising architecture for use in predicting the energetics of RNA interactions and could potentially be extended to other complex biophysical measurements that are a function of sequence. However, more direct experimental measurements are necessary to enable training of physically realistic models.

Results

A recurrent network model of complex RNA interactions

RNNs are designed specifically for data that take the form of a sequence of elements and thus can be naturally applied to nucleic acid sequences. In fact, a direct parallel can be drawn between the mathematical operations applied in a classic partition function algorithm and those of a long short term memory (LSTM) unit, an architecture common in machine translation tasks.[23,32] In this mapping, the cell state stores the partition functions of subsequences, although the activation functions typically used in an LSTM are different from those in the Turner model parallel (Figure 1A & 1B).[33] To make our data compatible with these existing components of deep learning architectures, we encoded each measurement as a sequence of one-hot vectors to serve as input to an RNN. Each experimental measurement is characterized by the design sequence as well as the concentrations and sequences of zero or more input RNAs. The one-hot vectors representing the sequences of each RNA were multiplied by the log concentration of that species. Input RNA sequences were placed at the beginning of the input to the RNN, followed by the design RNA, with each strand separated by a zero vector (Figure 1C). The resulting sequence of vectors, representing each base in the sequence, was fed from 5′ to 3′ end as input to a 2-layer, 1024-unit LSTM RNN. The final states of each layer were then passed through two fully connected layers to aggregate the features extracted, resulting in a final numerical Δ*G* prediction (Figure 1D).

Exploring the expressivity of the RNN model on simulated free energies

As a first test of a recurrent neural network representation of RNA folding, we evaluated the expressivity of Ribonet on simulated energetic measurements from the existing nearest-neighbor model. Success in this task would demonstrate that an RNN can mimic a model that is constrained by known biophysical principles, suggesting the architecture may be suitable for modeling RNA energetics.

We began by training on free energy values for each complex relative to the fully unfolded and dissociated RNA, the most direct energetic characterization of an RNA fold (Figure 2A). Such data can be obtained empirically through optical melting experiments, albeit not at the throughput possible for other, less direct energetic measurements. We trained the model on simulated folding free energies for complexes consisting of randomly generated designs in 15 different conditions chosen to match available experimental measurements (1, 10–20 in Table 2). Designs were generated by introducing the reverse complements of the input RNAs into a random sequence at random positions (see Methods for details). For each design, the free energy was computed over a full ensemble of simulated secondary structures, as would best mimic an actual experimental scenario. As a standard of comparison for this dataset, two different parameter sets of the Turner model available in the NUPACK software gave calculations with root-mean-square error (RMSE) of 7.65 kcal/mol, giving a rough estimate of the error resulting from these parameter estimates. A 1-nearest neighbor model, which predicts based on the closest measurement in the training set, yielded an RMSE of 18.1 kcal/mol, setting a rough upper on our performance (Figure S1). To determine how much data are necessary to parameterize this model, we trained on datasets of varying size from 10,000 to 500,000 sequences. We found that 500,000 training sequences were needed in training to avoid overfitting (Figure 2B). Using this training set, we were able to achieve excellent predictive performance of 5.46 kcal/mol, better than the error between two Turner parameter sets (Figure 2C).

As a test of whether the trained model was able to represent detailed secondary structure information, we evaluated the model on all double mutants of a randomly generated sequence distinct from the training data (Figure 2D). This analysis revealed that the RNN is appropriately sensitive even to small changes in the input sequence. Moreover, it detects base pairing effects, seen as diagonal segments on a heat map of double mutants; two mutations can restore the formation of a Watson-Crick base pair broken by single mutations (Figure 2D, insets). However, not all such base-pairing signals computed in the Turner model simulations (Figure 2D, top left) were captured by Ribonet and signals predicted by the RNN model were typically weaker, indicating less sensitivity to base-pairing than an explicit representation. Finally, we made predictions for input concentrations outside the range seen in the training data, to test if the model learned biophysically reasonable concentration effects. This free energy model produced an irregular pattern that does not match that of the simulated pattern over concentration space (Figure 2E). This suggests that, given enough data, the RNN model is expressive enough to capture the behavior of the Turner model of RNA energetics but cannot correctly extrapolate outside the concentration range seen in training.

Applying Ribonet to simulated affinity measurements

To more closely replicate how existing high-throughput experimental datasets might enable Ribonet training, we then generated a simulated dataset of predicted affinities to a reporter RNA molecule. For each sequence, we simulated the RNA-reporter affinities for 15 different conditions (1, 10–20 in Table 2). Sequences were randomly generated as in the folding free energy simulations, but affinities were computed by finding the difference in free energy between the complexes with and without the reporter molecule (Figure 3A, see Methods for details). As a comparison point, two different Turner parameter sets differ by an RMSE of 1.08 kcal/mol over the same sequences, while a simplistic 1-nearest neighbor model based on sequence similarity achieves an RMSE of about 3.3 kcal/mol (Figure S2). A learning curve revealed that more than 300,000 sequences were necessary to avoid overfitting the model (Figure 3B). Performance of 1.54 kcal/mol was achieved even with only 10,000 sequences (Figure S3A), less than the number of data collected in high-throughput experiments. The model trained on 500,000 sequences achieved even better predictive performance, with a test RMSE of 1.15 kcal/mol (Figure 3C). Although this model required fewer data to reach convergence than the absolute free energy study above, the resulting model does not exhibit base-pairing effects, as was the case in the free energy-trained model, and again did not extrapolate well to concentrations outside those seen in training (Figure 3D & E). We also tested a model trained on randomly generated input RNA sequences, rather than a fixed set of input sequences, to determine if a more diverse set of data would provide enough additional biophysical information. While these models did not perform better in terms of display of base pairing signals in double mutant tests or extrapolation to out-of-range input RNA concentrations, they did make more accurate predictions for complexes involving input RNAs not seen in training (Figure S3B & S3C). These tests suggest that Ribonet is able to capture affinity information with strong predictive power but without accurately capturing base-pairing effects or the functional form of affinity expected from statistical mechanics.

Despite its inability to recover physically realistic aspects of RNA folding, the accuracy of Ribonet for estimating affinities may still be useful for guiding design. We used a Monte Carlo algorithm to design an RNA sensor for an input RNA molecule, using the RNN as a scoring function and the Turner calculations as gold standard (Figure 4A, see Methods for details). In sequence optimization, we allowed both mutations and shifts of the input and reporter binding sites (Figure 4B). While shifts were accepted more often, mutations often preceded score increases, suggesting they are essential to sampling sequence space effectively (Figure 4C). We found that almost all designs improved over the initial sequence, as predicted by the Turner model, with the best designs achieving a fold change in affinity of almost 10^5 (Figure 4D). The RNN-predicted fold changes agreed well with those simulated by the Turner model. These simulation results suggest that a model trained on sufficiently large datasets can represent the behavior of the RNA well enough to design new sequences from scratch. However, design became less effective with a model trained on a smaller dataset (Figure 4E). Further, it failed for design of sensors for input RNAs not seen in the training dataset (Figure S4A), even using the model trained on random input sequences (Figure S4B).

Evaluating Ribonet on experimental affinity measurements

Having gained an understanding of the nuances of Ribonet's model architecture in relation to existing computational models, we trained the model on a rich dataset of affinity measurements. The Eterna massive open laboratory enables citizen scientists to design complex, multi-state RNAs using an intuitive graphical user interface.[20] Details of player strategies and performance are being reported in a separate manuscript; a brief summary follows. Various design challenges were posed as in-game puzzles with secondary structure constraints over multiple conditions defined by the concentrations of one or more input RNA strands. Players started with switch puzzles, where the aim was to design a molecule whose binding to a reporter RNA is dependent upon the presence or absence of a single input RNA. Next, players designed logic gates, which modulate their reporter binding based on the presence or absence of two input RNAs. Lastly, players moved to designing RNAs with an analog response, tuning the reporter affinity based on the ratio of concentrations of two to three input RNAs. Over 65,000 designs were made for these four puzzle types, collected over 9 different rounds (Table 1). While the library of designs contains a fair number of highly similar sequence clusters, it still represents a diverse collection of sequences targeting these 3 different types of puzzles (Figure 5A). Each sequence was synthesized on an RNA array, and its affinity to a fluorescent reporter RNA was measured by quantifying the fluorescence over varying reporter concentrations using a repurposed Illumina sequencing machine (see Methods for more details).[17] Each design was measured in up to 19 different input conditions, resulting in over 600,000 distinct affinity measurements.

Notably, these data are limited in range due to experimental constraints and contain less diverse sequences than the random simulated data (Figure 5A). To set a baseline for how well Ribonet might learn from realistic experimental datasets, we applied constraints to our simulated data to more closely match the experimental data. First, we filtered out points in the training data outside the experimentally measurable range, which reduced the 500,000 sequence dataset by about 5-fold. Test performance for data within the filtered range was good, with an RMSE of 0.92 kcal/mol, although worse than a similarly sized, unfiltered dataset, with an RMSE of 1.24 kcal/mol over values with nearly five-fold higher variance (Figure S5A). For test points outside the window, ΔG values were predicted to be near the edge of the measurable range (Figure S5B). This pileup is the expected behavior, because the model is unlikely to predict values beyond the range seen in the training data; however, the result underscores the inability of the model to learn physically reasonable extrapolations. In addition, we tested the model on simulated ΔGs for the sequences probed in available experiments instead of randomly generated sequences. The performance was far poorer than that for a comparably-sized randomly generated dataset, most likely due to the lower diversity of sequences (Figure S5C).

We then moved to experimental data, starting with a Ribonet model trained only on data from a single type of puzzle, with a fixed reporter sequence and a fixed set of three input RNAs (round 7 in Table 1). We used separate rounds of design and experiments for our training and test sets, with the test set having 3 additional experimental conditions not seen in the test set. As a lower bound on expected performance, the RMSE between technical replicates, which is a lower bound on our performance, is 0.39 kcal/mol (Figure S6A). As an upper bound, the RMSE for a 1-nearest neighbor model based on sequence similarity is 1.20 kcal/mol (Figure S6B, see methods for details). This bound is significantly tighter than that for simulation studies above due to the low sequence diversity of the designs and experimental measurement limits, setting a stringent bound for the performance of Ribonet. Further, the classic Turner model of RNA folding predicts these data poorly, with an RMSE of 2.73 kcal/mol (Figure S7). Training on 120,348 measurements over 10,046 sequences, we found that the model typically converged after about 50 epochs and significantly overfit to the training data, as evidenced by a large gap between training and test accuracy (Figure 5B). In our analysis of test set prediction results, we applied a Levenshtein distance threshold to the closest sequence in the training set to ensure that we did not include measurements unfairly similar to those in the training data (see Methods for details). For this model, the best model reaches an RMSE of 0.91 kcal/mol (Figure S8A). including predictions for experimental conditions not seen in the training data (Figure S9).

Because we had already trained Ribonet on simulated data from the Turner model, we hypothesized that performing transfer learning from the simulated model would accelerate and improve training. Thus, we trained models initialized with weights from the 500,000-sequence trained affinity model and compared those to randomly initialized weights. The training loss converged much more quickly, as expected, and also to a better test accuracy of 0.83 kcal/mol (Figure 5B). While the models were still overfit, as evidenced by a large gap between test and training accuracy, the performance was not only significantly better but also much more consistent across runs (Figure 5C). The preinitialized runs reached a median RMSE of 0.85 kcal/mol, with the best model reaching 0.83 kcal/mol (Figure 5D). For the rest of the models described, the transfer learning approach was used to initialize weights.

We then expanded our RNN models to increasing levels of generality, beyond the “single puzzle” setup used as an initial test case. As models become more general, the number of distinct puzzles and the associated amount of data available to train the model increases. However, the space of interactions that the model must represent also increases as the data involve more diverse inputs and reporters. For example, 29,542 additional sequence designs involve binding of an MS2 viral coat protein to specific RNA binding sites, but this protein-RNA interaction is captured through a new energetic parameter instead of those of RNA-RNA binding in designs tested previously. We tested three additional levels of model generality. “Single reporter” models generalized over multiple puzzles with the same reporter sequence, “all RNA” models included all measurements with RNA reporters over many puzzles and reporters, and “all reporter” models further included measurements with protein-based reporters. Empirically, we found that the addition of more complex RNA reporter data had no effect on performance (Figure 5E & S8), but predictive performance for protein reporters was poor, with RMSE of 1.06 kcal/mol (Figure S10). Our results suggest that training a more general model can allow for broader usability of the resulting model without compromising the predictive performance. In the case of including non-RNA reporters, however, additional information may need to be encoded to help the RNN understand how the reporter interacts with the design molecule.

Data augmentation to enhance model performance

Our analysis on simulated data suggested that a dataset of tens of thousands sequences, such as that used in each of our models is not sufficient to obtain optimal performance. To better understand the data dependence of this RNN for real data, we trained the model on varying sized subsets of the data to build a learning curve.[34] To generate these datasets, we clustered the sequences based on Levenshtein distances and cut the tree to form 10 subsets of unequal size but similar distance between clusters (see Methods for more details). Training on subsets of data containing increasing numbers of clusters revealed that a strong downward trend in RMSE continues up to the maximum dataset size (Figure 6B). These learning curves suggest that the performance of Ribonet remains strongly data-limited.

Data augmentation is a method commonly used in deep learning to increase the effective dataset size.[21,35,36] Although the techniques for augmentation are most well-established for image data, we hypothesized that the same ideas might be expanded to our datasets. We tested four approaches to augmenting our dataset (Figure 6A). First, we introduced additional copies of each data point for each possible ordering of the input sequences, since the chosen ordering is arbitrary. Secondly, we added a copy of each data point with a circular permutation of the design sequence. We have observed that top solutions of Eterna players are often circular permutations of each other (unpublished work). Thus, we hypothesize that this transformation may produce designs with similar behavior. Thirdly, reflection of input sequences has been used in machine translation to transform input data, despite the lack of semantic meaning in such data.[23] We introduced an additional copy of each data point with each sequence reversed, passed from 3′ to 5′ end instead of 5′ to 3′. This augmentation has the potential to increase signal for base pairing, as the equivalent helices can still form in a reflected secondary structure, even though the energetics of these alternative helices are different according to the classical Turner model.[11] Lastly, in a similar effort to encode additional base pairing information, we reverse complement each sequence to create an additional copy of each data point. To evaluate the effect of these augmentations on model performance, we used the "single puzzle" model, again using 10 trials with different random seeds to assess the variance. Although only the reordering approach worked in simulation (Figure S11), we observed a significant improvement in the performance for experimental data with the models trained on every type of augmented data, with *p*-values of 3.4×10^{-2} for reordering of inputs, 1.2×10^{-2} for circular permutations, 1.3×10^{-7} for reversed sequences, 9.4×10^{-10} for reverse complements, and 2.8×10^{-6} for all methods applied simultaneously (Figure 6C). The reverse complement augmentation method was able to achieve an RMSE of as low as 0.76 kcal/mol, compared to 0.83 kcal/mol without augmentation (Figure 6D). Performance with varying distance cutoffs is shown in Figure S12. These results suggest that simple transformations are able to encode additional information in model training, and reverse complementation is particularly effective, potentially by introducing additional base pairing information. These types of strategies may enable models to learn more biophysically relevant information from less data.

The best model trained with data augmentation still does not learn base pairing information in a test on double mutants (Figure S13). However, the behavior over wide concentration ranges is relatively smooth, with the exception of a discontinuity at 100 nM of the second input RNA (Figure 6E). This discontinuity may be a result of this being a frequent concentration choice in our set of experimental conditions. Overall, these tests suggest that Ribonet has the potential to produce biophysically realistic representations over concentration extrapolation, despite containing no explicit constraints to force this behavior. It remains unclear whether Ribonet can learn base-pairing information from simulated or experimental affinity data.

Discussion

Here, we describe a novel RNN-based method, Ribonet, for predicting the affinity of RNA-RNA interactions in the context of different environmental conditions. We show that it exhibits the expected base pair effects when trained with simulated folding free energy data generated using a classic Turner model, despite the lack of *a priori* information about RNA folding. When instead trained with simulated affinity data, models lack this explicit biophysical information but still perform well enough to facilitate design of an RNA sensor. We successfully apply transfer learning to train Ribonet on data generated through the Eterna platform and high-throughput array experiments. We find that the models generally benefit from more diverse data but are unable to learn from a naïve representation of protein reporters. Finally, we introduce data augmentation techniques specific to nucleic acid sequence data that significantly improve performance.

This work represents one of the first applications of deep learning to a regression task in biology.[37,38] While we were able to achieve fairly good performance, as compared to prior biophysical models or 1-nearest neighbor models, we have shown that we are currently limited by the size of our dataset, a common theme in machine and deep learning.[39,40] Despite the use of high-throughput design and measurement, one limitation of our methodology is that, when design constraints become difficult to satisfy, Eterna players tend to produce additional submissions by modifying existing designs. This often results in clusters of sequences that are highly similar, differing only by single mutations or base pair swaps. Although these near redundancies have the potential to improve sensitivity to small changes and provide useful base pairing information, they also result in a smaller effective dataset size. Our data augmentation approach partially mitigated this effect, but further efforts would benefit from more diverse designs as well as development of experimental methods able to probe a much larger variety of input RNAs and reporter RNAs. Additional work is necessary to develop computational methods to enable us to take full advantage of the specific features of this dataset and to understand why the reverse complement augmentation aids accuracy in experimental but not simulated datasets.

Through our simulation studies, we also found that the indirect nature of the affinity measurements prevented learning of explicit information about base pairing, a feature that is fundamental to RNA interactions. Nevertheless, free energy values were sufficient for learning low level biophysical information, suggesting high-throughput approaches for making these kinds of measurements would open the door for more biophysically realistic models of RNA energetics.[41,42]

In addition, the architecture chosen here does not take advantage of existing knowledge about RNA folding. Many successful applications of deep learning have taken this approach of learning from scratch.[21,23] Although we have demonstrated here that the RNN is able to extract base pairing information *de novo*, it fails to do so from simulated or experimental affinity data, which is a more complex function of secondary structure interactions compared to free energies. While Ribonet-facilitated design was possible in simulation, it required an order of magnitude more data than currently available even with high-throughput experiments. Adding constraints to the weights or explicitly encoding additional information may accelerate model training or improve test performance.[43,44] Further work is necessary to optimize deep learning model architectures for specific biological systems.

Machine learning methods such as those described here may form the foundation for computational tools for RNA design. Given the ability to quantitatively predict RNA-reporter affinities, optimization algorithms could be applied to design RNAs that sense and interact with other biomolecules in their environments, as demonstrated by our design tests in simulation.[45–47] This level of predictive performance would enable us to precisely design interactions in biological systems, including those central to modern biotechnological tools.

Materials and Methods

Eterna massive open laboratory

The Eterna online platform (https://www.eternagame.org) enables citizen scientists to design RNA molecules with complex, multi-state folding behavior. The game allows for the specification of design challenges, or puzzles, with secondary structure constraints in multiple conditions defined by the concentrations of various input RNAs. Players can view folds and energies predicted by ViennaRNA 1.8.4 [48], Vienna 2.1.9 [49], and NUPACK 3.0.4. [31,47]

RNA array experiments

Clonal clusters of RNA were generated on an Illumina sequencing flow cell as previously described.[17,18] Binding curves were collected for each cluster by incubating with input RNA oligos at defined concentrations and progressively higher concentrations (by factors of 2) of fluorescently labeled reporter oligo or recombinant SNAP-tag-labeled MS2 coat protein (Tables 1–3). The starting concentration was between 0.09 and 0.75 nM, depending on the experiment and the reporter, and the final concentration was 3000 nM. The fluorescent cluster images were aligned to sequencing data and quantified as previously described.[17,18] The fluorescent signal for each cluster, *F,* was fit to a simple binding curve:

$$F([MS2]) = F_{\max}\frac{[\text{reporter}]}{[\text{reporter}] + K_d}$$

The median K_d and $F_{\max}$ for each sequence variant was subsequently used and only measurements with at least 5 clusters were included in downstream analysis. For subsequent analysis, all K_d measurements were converted to ΔG at 1M standard state for the reporter, using the expression $\Delta G = -k_B T \log\frac{K_d}{1M}$.

Simulated data

Sequences were produced by generating random RNA sequences, introducing the reverse complements to the input and reporter RNA strands at random positions, and generating *n*/5 additional random mutations, where *n* is the length of the sequence. The length was chosen to be the same as those of the Eterna designs for which measurements were made. For the Turner or nearest-neighbor model, predictions were generated using the *complexes* and *concentrations* executables in NUPACK 3.0.6.[31,47] For the free energy model, the simulated free energy values were computed for the design strand along with the relevant input and reporter strands. Epistatic effects were computed by finding the difference in $\Delta\Delta G$ between double mutants and the sum of the two single mutants:

$$\varepsilon = \Delta\Delta G_{\text{double}} - \Delta\Delta G_{\text{single 1}} - \Delta\Delta G_{\text{single 2}}$$

For the affinity model, complex concentrations were computed for the design strand along with the relevant input RNAs and reporter. A concentration of 1 pM was used for the design strand and 10 nM for the reporter strand, although the resulting affinity is independent of this choice as long as the reporter is in excess. These values were used to calculate the simulated ΔG value:

$$\Delta G = -k_B \mathrm{T} \cdot \log\left(\frac{[\text{complexes with reporter}]}{[\text{complexes without reporter}][\text{free reporter}]}\right)$$

Data were generated for all 15 experimental conditions used in experimental round 9, which was used as a test set for analyses of experimental data (Table 1 & 2).

1-nearest neighbor model

The 1-nearest neighbor model, not to be confused with the classic Turner nearest neighbor model for RNA energetics, is a *k*-nearest neighbors model with *k*=1.[50] This model predicts the energy for a new RNA sequence as the value observed for the closest sequence in the training set. The distance metric used was Levenshtein distance between the design sequences. For input RNA concentrations not seen in the training data, the closest condition was selected based on Euclidean distance.

Model architecture

Each measurement was identified by a design sequence, the concentrations and sequences of zero or more input RNAs, and a reporter RNA sequence. Each base in each sequence was represented as a four-element, one-hot vector. Input and design RNA vectors were multiplied by the log of their concentrations (in nM) to encode variable concentrations across conditions. For design RNA vectors, 10 μM was used as an estimate of their local concentration on the array. The input and design sequences, with each strand separated by a zero vector, served as the input to the RNN. For “single puzzle” and “single reporter” models, the sequence of the reporter was omitted from the input, as it was assumed to be the same across all data. For “all RNA” and “all reporter” models, the reporter was included as the last strand passed to the RNN (Figure S14). For the “all reporter” models that included measurements with the MS2 protein as a reporter, a fifth element was introduced into each one-hot vector to enable representation of the MS2 reporter (Figure S14). These inputs were passed through a 2-layer, 1024-unit LSTM RNN.

The final state of this RNN was then processed by two fully connected layers to result in a single numerical prediction for the ΔG of the interaction between design and reporter strands.

Model training

Hyperparameters were tuned using data from round 7, split into a training and validation set (Table 1). By clustering the sequences based on Levenshtein distance and splitting the resulting clusters, we ensured that similar sequences ended up in the same split of the dataset. Random search was conducted over learning rates and dropout rates to find the optimal parameter values. Various optimizers and batch sizes were also tested. All results shown were trained using the Adam optimizer, with a learning rate of 10^{-3}, a dropout rate of 0.5, and a batch size of 128. All models were trained until training loss converged. With the exception of those for learning curve analysis, each model was trained 10 times with different random seeds.

For randomly initialized models, weights were initialized from a truncated normal distribution. For pretrained models, weights were initialized using the final model for the 500,000-sequence simulated dataset.

"Single puzzle" models were trained on round 7. "Single reporter" models were trained on rounds 2 and 7. "All RNA" models were trained on rounds 2–3 and 7–8. "All reporter models" were trained on rounds 1–4, 7, and 8, with rounds 5 and 6 held out as additional MS2 reporter test sets. Reported test performance was computed for round 9 unless otherwise stated. See Table 1 for details of the data collected in each round.

Learning curves

Due to strong similarity between sequences in our dataset, clustering was used to group subsets of data instead of generating random subsets. Single-linkage hierarchical clustering was performed on the sequences to group similar designs together. The resulting tree was cut to form 10 clusters of variable numbers of sequences. The first model was then trained on only sequences in cluster 1, the second on clusters 1–2, and so on. All models were trained for 50 epochs. The test set remained the same for all models.

Base pairing and concentration extrapolation tests

Both of these tests of whether models are biophysically realistic representations used a randomly generated design with the following sequence: CAAUCGCUCAGAACGUAGU GUACGCAGGCAGGAAUGCGGCAGAAAAGGCACUAACCGAGCCGAACCAGGUAGCCCAAA ACCUCUCC. This sequence was not used in any of the training datasets. For the base pairing test, predictions were made for all double mutants of this sequence. For the concentration extrapolation test, predictions were made for all concentration combinations of two inputs in the range from 1 pM to 1 mM.

Design algorithm

For each round of sequence design, a random initial sequence was generated as a starting sequence. The reverse complements to the input and reporter RNA were introduced into this random starting sequence as binding sites. Each move was randomly chosen to be either a random mutation or a change in the binding sites, with equal probability. A modification to the binding sites was randomly chosen to be either a shift or change in length of the reverse complement sequence. The score was computed to be the difference between predicted affinities with and without the input RNA. The Metropolis criterion, with temperature 0.1, was applied to determine if a move was accepted or rejected. 20,000 iterations were applied to achieve the final designs. To evaluate the resulting designs, affinities were predicted using NUPACK as described above.

Data augmentation

For the reordering augmentation dataset, *n*! data points were included for each data point in the original dataset, where *n* is the number of inputs in that experimental condition. These represent each possible ordering of the *n* input strands. For cyclic augmentation, one circular permutation of each data point was added, with the first and second halves of the design sequence swapped. For reverse augmentation, an additional copy of each data point was introduced into the augmented dataset with all sequences — inputs and design — read from 3′ to 5′ end instead of 5′ to 3′. The order of the sequences is preserved, with inputs first and design subsequently. For reverse complement augmentation, all sequences were changed to their reverse complement in an additional copy of each data point. For the combined approach, all augmentation approaches were applied to each sequence.

Data Availability & Software

Models were implemented in Python with TensorFlow 1.0.0.[51] Data and code are available at https://github.com/wuami/ribonet.

Acknowledgments

We acknowledge funds from the Stanford Dean’s Innovation Award (R.D.), Joint Initiative for Metrology in Biology (R.D., W.G.), NIH R01 GM100953 (R.D.), NIH R31 GM122579 (R.D.), NIH R01 GM111990 (W.G), and NSF GRFP DGE-114747 (M.W.). This work used the Extreme Science and

Engineering Discovery Environment (XSEDE) [52], which is supported by National Science Foundation grant number ACI-1548562.

References

1. Gesteland R, Cech T, Atkins J. The RNA World. 3rd ed. Cold Spring Harbor Laboratory Press; 2006.
2. Geisler S, Coller J. RNA in unexpected places: long non-coding RNA functions in diverse cellular contexts. Nat Publ Gr. 2013;14. doi:10.1038/nrm3679
3. Jinek M, Chylinski K, Fonfara I, Hauer M, Doudna JA, Charpentier E. A Programmable Dual-RNA–Guided DNA Endonuclease in Adaptive Bacterial Immunity. Science (80-). 2012;337.
4. Mali P, Yang L, Esvelt KM, Aach J, Guell M, DiCarlo JE, et al. RNA-Guided Human Genome Engineering via Cas9. Science (80-). 2013;339.
5. Meister G, Tuschl T. Mechanisms of gene silencing by double-stranded RNA. Nature. Nature Publishing Group; 2004;431: 343–349. doi:10.1038/nature02873
6. Reynolds A, Leake D, Boese Q, Scaringe S, Marshall WS, Khvorova A. Rational siRNA design for RNA interference. Nat Biotechnol. Nature Publishing Group; 2004;22: 326–330. doi:10.1038/nbt936
7. Orelle C, Carlson ED, Szal T, Florin T, Jewett MC, Mankin AS. Protein synthesis by ribosomes with tethered subunits. Nature. Nature Research; 2015;524: 119–124. doi:10.1038/nature14862
8. Des Soye BJ, Patel JR, Isaacs FJ, Jewett MC. Repurposing the translation apparatus for synthetic biology. Curr Opin Chem Biol. 2015;28: 83–90. doi:10.1016/j.cbpa.2015.06.008
9. Brion P, Westhof E. HIERARCHY AND DYNAMICS OF RNA FOLDING. Annu Rev Biophys Biomol Struct. Annual Reviews 4139 El Camino Way, P.O. Box 10139, Palo Alto, CA 94303-0139, USA ; 1997;26: 113–137. doi:10.1146/annurev.biophys.26.1.113
10. Jankowsky E, Harris ME. RNA–protein interactions are critical for the regulation of gene expression. Nat Publ Gr. 2015;16. doi:10.1038/nrm4032
11. Xia T, John SantaLucia J, Burkard ME, Kierzek R, Schroeder SJ, Jiao X, et al. Thermodynamic Parameters for an Expanded Nearest-Neighbor Model for Formation of RNA Duplexes with Watson−Crick Base Pairs †. American Chemical Society; 1998; Available: http://pubs.acs.org/doi/full/10.1021/bi9809425
12. Serra MJ, Turner DH. [11] Predicting thermodynamic properties of RNA. Methods Enzymol. 1995;259: 242–261. doi:10.1016/0076-6879(95)59047-1
13. SantaLucia J. A unified view of polymer, dumbbell, and oligonucleotide DNA nearest-neighbor thermodynamics. Proc Natl Acad Sci U S A. 1998;95: 1460–5. Available: http://www.ncbi.nlm.nih.gov/pubmed/9465037
14. Mathews DH, Sabina J, Zuker M, Turner DH. Expanded sequence dependence of thermodynamic parameters improves prediction of RNA secondary structure. J Mol Biol. 1999;288: 911–40. doi:10.1006/jmbi.1999.2700
15. Doshi KJ, Cannone JJ, Cobaugh CW, Gutell RR. Evaluation of the suitability of free-energy minimization using nearest-neighbor energy parameters for RNA secondary structure prediction. BMC Bioinformatics. 2004;5. doi:10.1186/1471-2105-5-105
16. Anderson-Lee J, Fisker E, Kosaraju V, Wu M, Kong J, Lee J, et al. Principles for

Predicting RNA Secondary Structure Design Difficulty. J Mol Biol. 2016;428: 748–757. doi:10.1016/j.jmb.2015.11.013

17. Buenrostro JD, Araya CL, Chircus LM, Layton CJ, Chang HY, Snyder MP, et al. Quantitative analysis of RNA-protein interactions on a massively parallel array reveals biophysical and evolutionary landscapes. Nat Biotechnol. Nature Publishing Group, a division of Macmillan Publishers Limited. All Rights Reserved.; 2014;32: 562–8. doi:10.1038/nbt.2880
18. She R, Chakravarty AK, Layton CJ, Chircus LM, Andreasson JOL, Damaraju N, et al. Comprehensive and quantitative mapping of RNA-protein interactions across a transcribed eukaryotic genome. Proc Natl Acad Sci U S A. National Academy of Sciences; 2017;114: 3619–3624. doi:10.1073/pnas.1618370114
19. Boyle EA, Andreasson JOL, Chircus LM, Sternberg SH, Wu MJ, Guegler CK, et al. High-throughput biochemical profiling reveals sequence determinants of dCas9 off-target binding and unbinding. Proc Natl Acad Sci. National Academy of Sciences; 2017; 201700557. doi:10.1073/PNAS.1700557114
20. Lee J, Kladwang W, Lee M, Cantu D, Azizyan M, Kim H, et al. RNA design rules from a massive open laboratory. Proc Natl Acad Sci U S A. 2014;111: 2122–7. doi:10.1073/pnas.1313039111
21. Krizhevsky A, Sutskever I, Hinton GE. ImageNet Classification with Deep Convolutional Neural Networks. Adv Neural Inf Process Syst. 2012; 1097–1105.
22. Hinton G, Deng L, Yu D, Dahl G, Mohamed A, Jaitly N, et al. Deep Neural Networks for Acoustic Modeling in Speech Recognition: The Shared Views of Four Research Groups. IEEE Signal Process Mag. 2012;29: 82–97. doi:10.1109/MSP.2012.2205597
23. Sutskever I, Vinyals O, Le Q V. Sequence to Sequence Learning with Neural Networks. Adv Neural Inf Process Syst. 2014;
24. Szegedy C, Wei Liu, Yangqing Jia, Sermanet P, Reed S, Anguelov D, et al. Going deeper with convolutions. 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE; 2015. pp. 1–9. doi:10.1109/CVPR.2015.7298594
25. Alipanahi B, Delong A, Weirauch MT, Frey BJ. Predicting the sequence specificities of DNA- and RNA-binding proteins by deep learning. Nat Biotechnol. 2015;33: 3–1. doi:10.1038/nbt.3300
26. Zhou J, Troyanskaya OG. Predicting effects of noncoding variants with deep learning–based sequence model. Nat Methods. 2015;12: 931–934. doi:10.1038/nmeth.3547
27. Esteva A, Kuprel B, Novoa RA, Ko J, Swetter SM, Blau HM, et al. Dermatologist-level classification of skin cancer with deep neural networks. Nat Publ Gr. 2017;542. doi:10.1038/nature21056
28. Rui Xie, Quitadamo A, Cheng J, Xinghua Shi. A predictive model of gene expression using a deep learning framework. 2016 IEEE International Conference on Bioinformatics and Biomedicine (BIBM). IEEE; 2016. pp. 676–681. doi:10.1109/BIBM.2016.7822599
29. Graves A, Mohamed A-R, Hinton G. SPEECH RECOGNITION WITH DEEP RECURRENT NEURAL NETWORKS. arXiv. 2013;
30. Amodei D, Anubhai R, Battenberg E, Case C, Casper J, Catanzaro B, et al. Deep Speech 2: End-to-End Speech Recognition in English and Mandarin. arXiv. 2015; Available: https://arxiv.org/pdf/1512.02595.pdf
31. Dirks RM, Bois JS, Schaeffer JM, Winfree E, Pierce NA. Thermodynamic Analysis of Interacting Nucleic Acid Strands. SIAM Rev. Society for Industrial and Applied Mathematics; 2007;49: 65–88. doi:10.1137/060651100

32. Hochreiter S, Urgen Schmidhuber J. LONG SHORT-TERM MEMORY. Neural Comput. 1997;9: 1735–1780. Available: http://www7.informatik.tu-muenchen.de/~hochreit

33. McCaskill JS. The equilibrium partition function and base pair binding probabilities for RNA secondary structure. Biopolymers. 1990;29: 1105–1119. doi:10.1002/bip.360290621

34. Beleites C, Neugebauer U, Bocklitz T, Krafft C, Popp J. Sample size planning for classification models. Anal Chim Acta. 2013;760: 25–33. doi:10.1016/j.aca.2012.11.007

35. Yaeger L, Lyon R. Effective Training of a Neural Network Character Classifier for Word Recognition. Adv Neural Inf Process Syst. 1996;9: 807–813. Available: https://papers.nips.cc/paper/1250-effective-training-of-a-neural-network-character-classifier-for-word-recognition.pdf

36. Simard PY, Steinkraus D, Platt JC. Best Practices for Convolutional Neural Networks Applied to Visual Document Analysis. Proceedings of the Seventh International Conference on Document Analysis and Recognition. 2003. Available: https://pdfs.semanticscholar.org/7b1c/c19dec9289c66e7ab45e80e8c42273509ab6.pdf

37. Mamoshina P, Vieira A, Putin E, Zhavoronkov A. Applications of Deep Learning in Biomedicine. Mol Pharm. American Chemical Society; 2016;13: 1445–1454. doi:10.1021/acs.molpharmaceut.5b00982

38. Angermueller C, Pärnamaa T, Parts L, Stegle O. Deep learning for computational biology. Mol Syst Biol. 2016;12. doi:10.15252/msb

39. Cho J, Lee K, Shin E, Choy G, Do S. HOW MUCH DATA IS NEEDED TO TRAIN A MEDICAL IMAGE DEEP LEARNING SYSTEM TO ACHIEVE NECESSARY HIGH ACCURACY? arXiv. 2016; Available: https://arxiv.org/pdf/1511.06348.pdf

40. Raudys SJ, Jain AK. Small sample size effects in statistical pattern recognition: recommendations for practitioners. IEEE Trans Pattern Anal Mach Intell. 1991;13: 252–264. doi:10.1109/34.75512

41. De Vlaminck I, Henighan T, van Loenhout MTJ, Pfeiffer I, Huijts J, Kerssemakers JWJ, et al. Highly Parallel Magnetic Tweezers by Targeted DNA Tethering. Nano Lett. American Chemical Society; 2011;11: 5489–5493. doi:10.1021/nl203299e

42. Yang D, Ward A, Halvorsen K, Wong WP. Multiplexed single-molecule force spectroscopy using a centrifuge. Nat Commun. 2016;7: 11026. doi:10.1038/ncomms11026

43. Shrikumar A, Greenside P, Kundaje A. Reverse-complement parameter sharing improves deep learning models for genomics. bioRxiv. 2017; Available: http://biorxiv.org/content/early/2017/01/27/103663

44. Zhang S, Zhou J, Hu H, Gong H, Chen L, Cheng C, et al. A deep learning framework for modeling structural features of RNA-binding protein targets. Nucleic Acids Res. Oxford University Press; 2015;44: e32. doi:10.1093/nar/gkv1025

45. Höner Zu Siederdissen C, Hammer S, Abfalter I, Hofacker IL, Flamm C, Stadler PF. Computational design of RNAs with complex energy landscapes. Biopolymers. 2013;99: 1124–36. doi:10.1002/bip.22337

46. Espah Borujeni A, Mishler DM, Wang J, Huso W, Salis HM. Automated physics-based design of synthetic riboswitches from diverse RNA aptamers. Nucleic Acids Res. 2016;44: 1–13. doi:10.1093/nar/gkv1289

47. Zadeh JN, Steenberg CD, Bois JS, Wolfe BR, Pierce MB, Khan AR, et al. NUPACK: Analysis and design of nucleic acid systems. J Comput Chem. 2011;32: 170–3.

doi:10.1002/jcc.21596

48. Hofacker IL, Fontana W, Stadler PF, Bonhoeffer LS, Tacker M, Schuster P. Fast folding and comparison of RNA secondary structures. Monatshefte für Chemie Chem Mon. Springer-Verlag; 1994;125: 167–188. doi:10.1007/BF00818163

49. Lorenz R, Bernhart SH, Höner zu Siederdissen C, Tafer H, Flamm C, Stadler PF, et al. ViennaRNA Package 2.0. Algorithms Mol Biol. BioMed Central; 2011;6: 26. doi:10.1186/1748-7188-6-26

50. Cover T, Hart P. Nearest neighbor pattern classification. IEEE Trans Inf Theory. IEEE Press; 1967;13: 21–27. doi:10.1109/TIT.1967.1053964

51. Abadi M, Agarwal A, Barham P, Brevdo E, Chen Z, Citro C, et al. TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems. 2015; Available: http://download.tensorflow.org/paper/whitepaper2015.pdf

52. Towns J, Cockerill T, Dahan M, Foster I, Gaither K, Grimshaw A, et al. XSEDE: Accelerating Scientific Discovery. Comput Sci Eng. IEEE Computer Society; 2014;16: 62–74. doi:10.1109/MCSE.2014.80

Table 1: Summary of data collection rounds. Specific sequences and concentrations for each condition are available in Table 2 & 3. The number of measurements is less than the product of the number of sequences and the number of conditions due to filtering of low quality measurements.

round	puzzle type	inputs	reporter	number of sequences	conditions	number of measurements
1	logic gates	1 & 2	MS2 protein	9,354	1–4	37,364
2	switch	1	reporter 1	1,238	1–2	2,470
3	switch	1	reporter 2	2,119	1–2	4,236
4	arithmetic	2 & 3	MS2 protein	5,511	1, 3, 5–9	38,521
5	logic gates	1 & 2	MS2 protein	8,519	1–4	29,396
6	arithmetic	2 & 3	MS2 protein	6,158	1, 3, 5–9	42,966
7	arithmetic	4–6	reporter 1	10,046 (9,957 training, 89 validation)	1, 10–20	120,348
8	arithmetic	7–9	reporter 3	10,519	1, 21–38	197,199
9	arithmetic	4–6	reporter 1	11,841	1, 10–20, 39–41	168,939

Table 2: The specific concentrations of input RNAs for each experimental condition are summarized in this table.

condition	**input RNA concentrations**
1	none
2	100 nM input 1
3	100 nM input 2
4	100 nM input 1, 100 nM input 2
5	100 nM input 3
6	5 nM input 3
7	5 nM input 2, 5 nM input 3
8	5 nM input 2, 100 nM input 3
9	100 nM input 2, 100 nM input 3
10	100 nM input 4
11	100 nM input 5
12	100 nM input 6
13	100 nM input 4, 100nM input 5
14	100 nM input 4, 100nM input 6
15	100 nM input 5, 100nM input 6
16	5 nM input 4, 100 nM input 6
17	5 nM input 5, 100 nM input 6
18	50 nM input 4, 50 nM input 5, 100 nM input 6
19	50 nM input 4, 50 nM input 5, 300 nM input 6
20	200 nM input 4, 200 nM input 5, 300 nM input 6
21	100 nM input 7
22	100 nM input 8
23	100 nM input 9
24	100 nM input 7, 100 nM input 8
25	100 nM input 7, 100 nM input 9
26	100 nM input 8, 100 nM input 9
27	5 nM input 7, 5 nM input 8
28	5 nM input 7, 100 nM input 8
29	5 nM input 7, 100 nM input 9
30	5 nM input 8, 100 nM input 9
31	100 nM input 7, 5 nM input 8
32	5 nM input 7, 5 nM input 8, 100 nM input 9
33	25 nM input 7, 100 nM input 8, 100 nM input 9
34	50 nM input 7, 50 nM input 8, 25 nM input 9
35	50 nM input 7, 50 nM input 8, 100 nM input 9
36	50 nM input 7, 50 nM input 8, 300 nM input 9
37	100 nM input 7, 25 nM input 8, 100 nM input 9
38	200 nM input 7, 200 nM input 8, 300 nM input 9
39	5 nM input 4, 5 nM input 5, 100 nM input 6
40	100 nM input 4, 25 nM input 5, 100 nM input 6
41	25 nM input 4, 100 nM input 5, 100 nM input 6

Table 3: Input and reporter RNA sequences used for all puzzles are shown here.

RNA	**Sequence**
reporter 1	UAAGUUCUGA
reporter 2	ACCCCACAAUAAAGAAUAAG
reporter 3	GACUAAGUUCUGAC
input 1	CUAAGCAGUUCCCUCAUU
input 2	ACCCCACAAUAAAGAAUAAG
input 3	CUGAUCCCAUCUCACUC
input 4	ACAGCUCAGCACAACAUUCC
input 5	GUUGGUGCCUUUUGUGCCAC
input 6	UUUUGGGCUACCUGGUUCGU
input 7	ACAGAUGCAGGAACAGGCUG
input 8	CCAUGGUGAUGGAUGGUUUG
input 9	GUACAUAGAGAGACAGGUGG

Table 4: All training and test RMSEs (kcal/mol) for simulated data are summarized in this table.

number of training sequences	**train RMSE**	**test RMSE**
simulated free energy		
100,000	1.98	6.77
200,000	2.84	5.46
300,000	3.39	6.08
400,000	3.94	5.42
500,000	4.58	5.05
simulated affinity		
10,000	0.19	1.54
20,000	0.16	1.40
50,000	0.26	1.29
100,000	0.36	1.24
200,000	0.59	1.24
300,000	0.97	1.15
400,000	0.86	1.15
500,000	0.98	1.10

Table 5: All training and test RMSEs (in kcal/mol) for experimental data are summarized in this table. All test values are filtered by a Levenshtein distance of 5 from the training data.

replicate	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
single puzzle, random initialization (train)	0.29	0.12	0.14	0.15	0.15	0.13	0.13	0.15	0.44	0.19
single puzzle, random initialization (test)	1.01	0.93	1.03	0.97	0.91	0.98	0.97	0.94	1.07	0.97
single puzzle (train)	0.07	0.07	0.07	0.07	0.06	0.09	0.08	0.07	0.07	0.07
single puzzle (test)	0.85	0.86	0.85	0.89	0.85	0.87	0.87	0.85	0.87	0.83
single reporter (train)	0.07	0.08	0.07	0.07	0.08	0.07	0.07	0.08	0.07	0.08
single reporter (test)	0.87	0.86	0.86	0.87	0.85	0.87	0.86	0.85	0.84	0.89
all RNA (train)	0.12	0.11	0.10	0.14	0.11	0.11	0.11	0.12	0.11	0.11
all RNA (test)	0.85	0.88	0.84	0.81	0.88	0.84	0.86	0.92	0.81	0.92
all reporter (train)	0.13	0.14	0.17	0.14	0.14	0.15	0.14	0.14	0.16	0.14
all reporter (test)	0.83	0.85	0.90	0.83	0.81	0.83	0.79	0.88	0.87	0.84
single puzzle, reorder augment (train)	0.06	0.07	0.05	0.05	0.06	0.06	0.05	0.06	0.06	0.05
single puzzle, reorder augment (test)	0.82	0.86	0.86	0.85	0.86	0.82	0.85	0.84	0.83	0.86
single puzzle, cycle augment (train)	0.09	0.08	0.08	0.09	0.10	0.08	0.08	0.08	0.09	0.08
single puzzle, cycle augment (test)	0.84	0.83	0.82	0.87	0.80	0.85	0.84	0.82	0.85	0.85
single puzzle, reverse augment (train)	0.09	0.09	0.08	0.09	0.09	0.09	0.09	0.10	0.09	0.09
single puzzle, reverse augment (test)	0.77	0.77	0.81	0.81	0.80	0.77	0.82	0.77	0.77	0.80
single puzzle, reverse complement augment (train)	0.09	0.09	0.09	0.09	0.09	0.10	0.08	0.10	0.09	0.09
single puzzle, reverse complement augment (test)	0.79	0.76	0.80	0.78	0.80	0.76	0.80	0.78	0.77	0.76
single puzzle, all methods augment (train)	0.12	0.14	0.11	0.11	0.11	0.11	0.11	0.11	0.11	0.13
single puzzle, all methods augment (test)	0.82	0.84	0.81	0.82	0.81	0.79	0.83	0.82	0.79	0.78

Figure 1: (A) Parallels can be drawn between the components of an LSTM unit (top) and those of a partition function calculation (bottom). The cell state (C_i) can store the partition functions for subsequences (Q_{de}), with the outputs (o_i) containing the desired prediction value. The weights w in the LSTM contain the values that parameterize different motif energies (G). (B) The dynamic programming matrix can be computed horizontally rather than diagonally to account for the fact that the sequence is read in order, so that the 3′ end of the sequence is not needed at the start of the computation. (C) Each base in each sequence is encoded as an n-hot vector where n is the log of the RNA concentration. Input RNAs, followed by the design RNA, and, when relevant, the reporter RNA are taken in that order, and each strand is separated by a zero-vector. (D) The input is fed through a 2-layer LSTM RNN, followed by 2 fully connected layers.

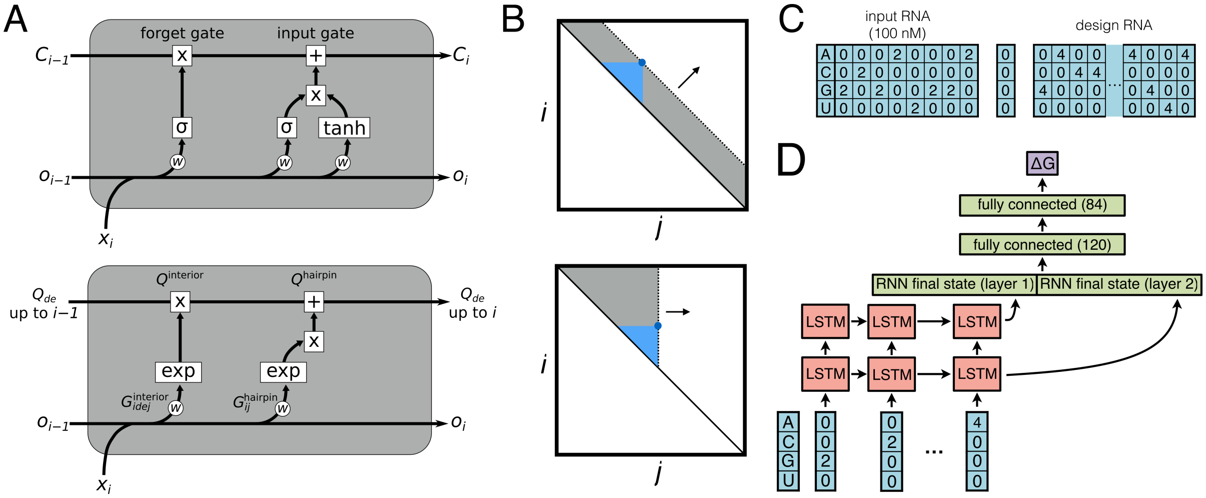


Figure 2: (A) The free energy computed is the difference between the folded complex and unfolded, dissociated individual strands. (B) For the free energy model, the learning curve for varying numbers of simulated sequences reveals that training and test performance converges at around 500,000 sequences. (C) For the largest dataset, the model achieves predictions of affinity with an RMSE of 5.46 kcal/mol. (D) For the free energy model, predictions were made for free energy of folding for a set of double mutants (lower right) and compared to those from the Turner model (upper left). To highlight the effects of double mutants, these are shown as epistatic effects (ε), which is the predicted free energy of the double mutant minus those for the two corresponding single mutants. Diagonal signals (inset) suggest that the model detects compensatory rescue of base pairs. The secondary structure suggested by the helices is shown on the middle right. (E) Extrapolating to predictions outside the training conditions results in an irregular, unrealistic pattern (right), notably different from the simulated behavior (left). The boxed region highlights the range of concentrations seen in the training data, showing similar simulated and predicted behavior.

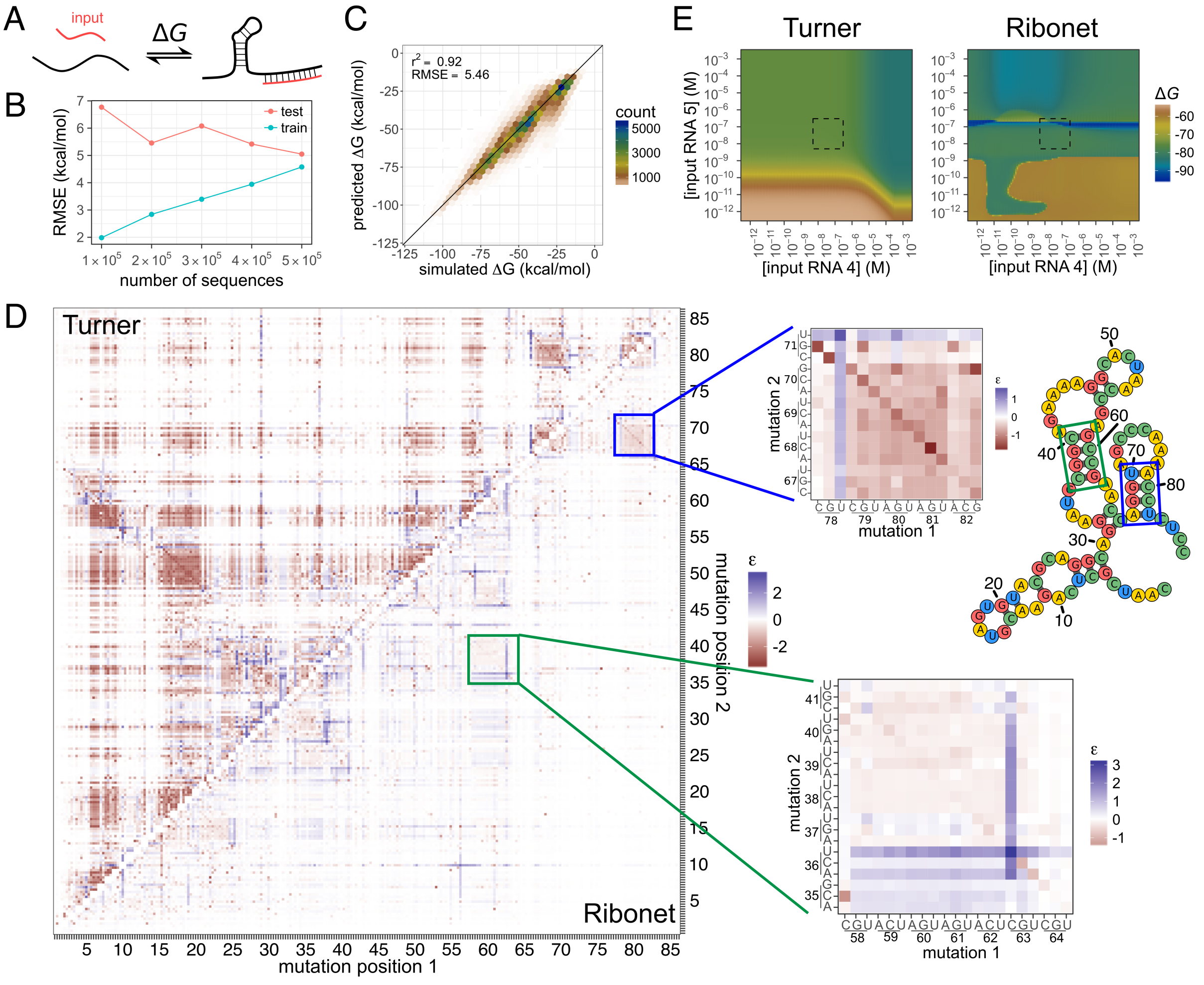


Figure 3: (A) The affinity is computed as the difference in free energy between the complexes with and without the reporter (blue). (B) For the simulated affinity data, a learning curve demonstrated that 300,000 sequences are necessary for convergence of training and test performance, fewer than for the free energy data. (C) For the largest dataset, the model achieves an RMSE of 1.15 kcal/mol. (D) The model (right) shows more realistic, smooth concentration dependence than the free energy model, even though the observed pattern differs greatly from the simulated one (left). The dotted box highlights the concentration range seen in the training set. (E) While the simulated affinities for a set of double mutants shows diagonal helix signals (upper left), this is not seen in the RNN predictions (lower right). The values shown are the predicted energy for the double mutant minus those for the two corresponding single mutants.

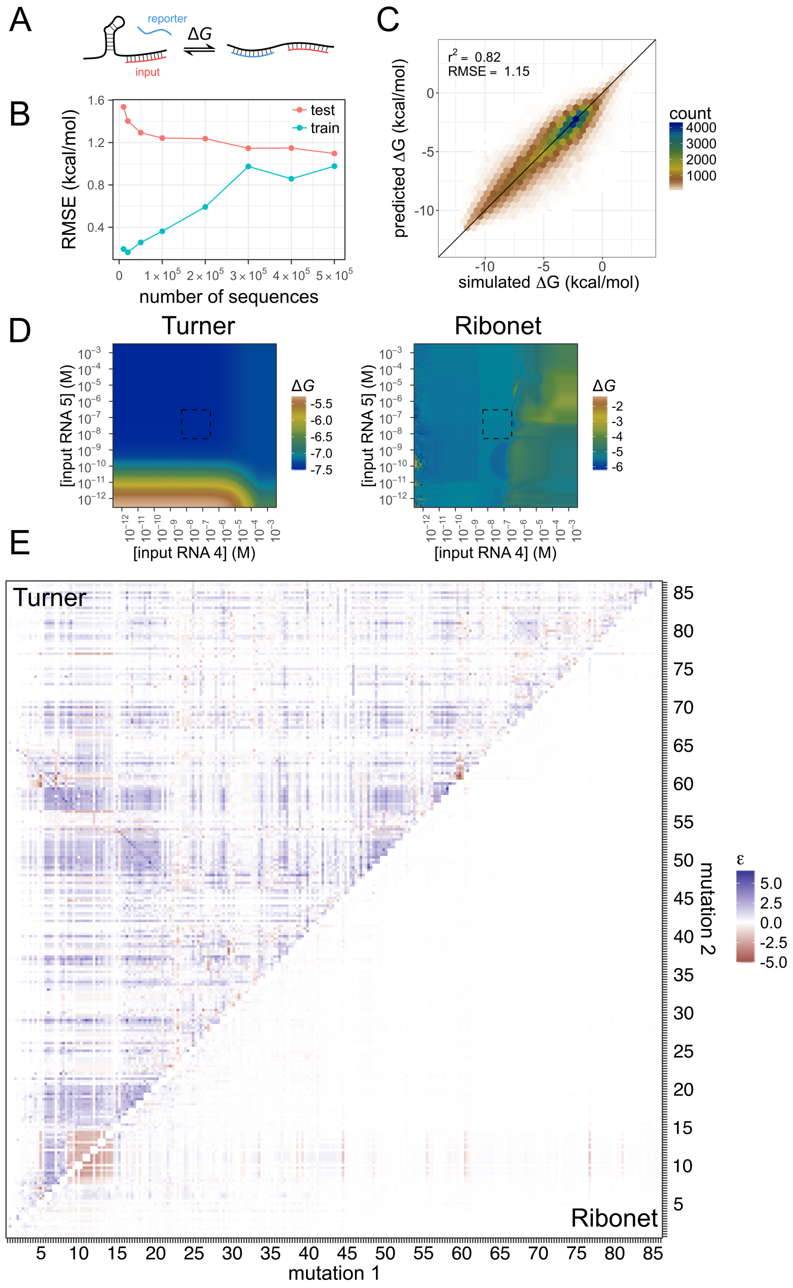


Figure 4: (A) For design, sequences were optimized using a score based on the ratio of affinities between the conditions with and without the input molecule. (B) Our design method initializes sequences with the reverse complements of the input and reporter sequences. Allowed moves include mutations and shifts of the reverse complements. (C) Our design method enables optimization of the RNN-predicted score. While most accepted moves are shifts (blue), significant jumps in score are often preceded by mutations (red). (D) The ground truth fold change between the two states, that predicted by the Turner model, increases significantly from initialization (purple) to after optimization (green), reaching a fold change as high as 10^6. Lines connect the start and end sequences from the same design run. Predicted and ground truth values correlate well ($r^2 = 0.63$), and a paired one-sided *t*-test shows that the optimized fold changes are significantly greater than the initial ones (p=2.8x10^{-7}). (E) For the model trained on 10,000 sequences, the improvement is not as consistent, not reaching statistical significance (p=0.08), but most runs still yield an improvement in nearest-neighbor predicted fold change, our ground truth in these simulations.

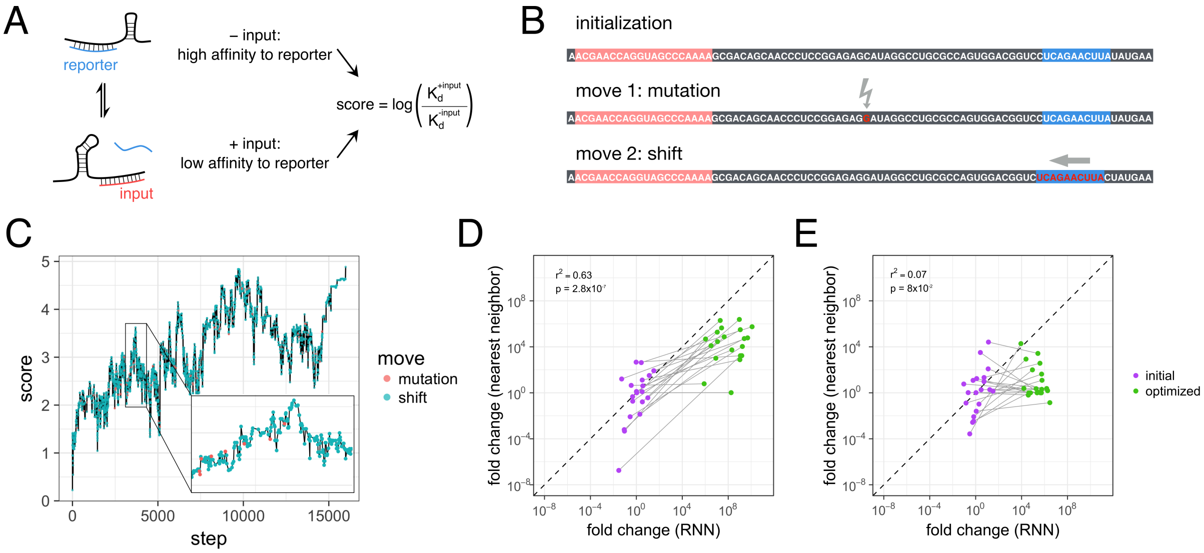


Figure 5: (A) Hierarchical clustering of all sequences designed over 9 rounds of collection show many tight clusters of similar sequences. (B) Initializing weights from a model pretrained on simulated data allows the loss to converge more quickly and to a lower final train and test RMSE than the random initialized model. The pretrained model shown was only trained for an additional 50 epochs, and the final loss is extended horizontal for ease of comparison. (C) Pretraining the model significantly improves performance (one-sided *t*-test, $p = 6.46\text{x}10^{-6}$). (D) With transfer learning, the model is able to achieve performance of 0.83 kcal/mol. (E) Test results are shown for 10 replicate training experiments for each model type. No statistically significant differences are seen across models. All RMSEs are for only test set points with Levenshtein distance of at least 5 from all sequences in the training set.

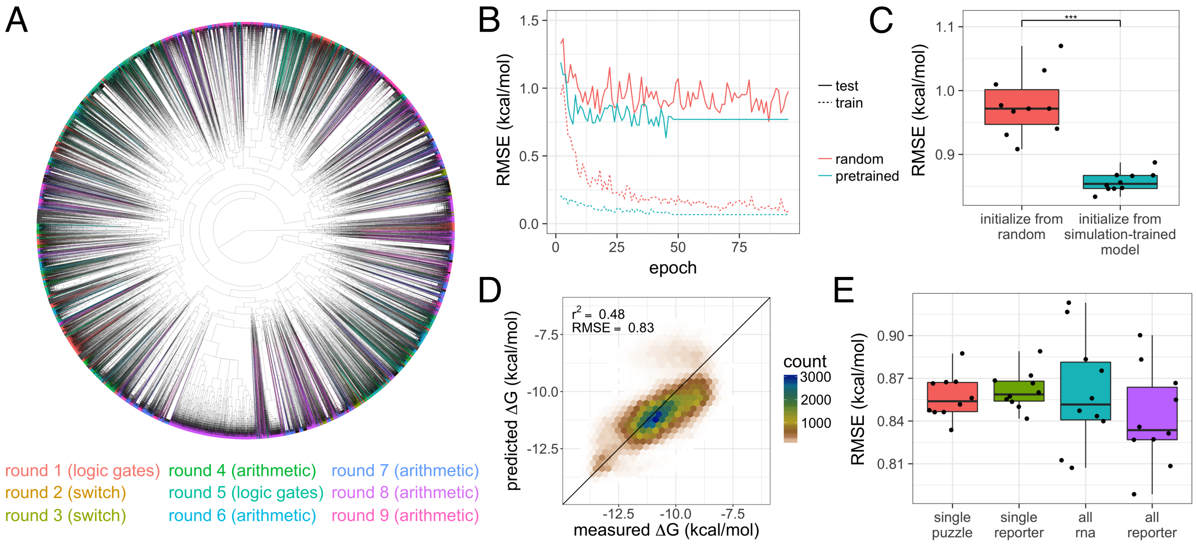
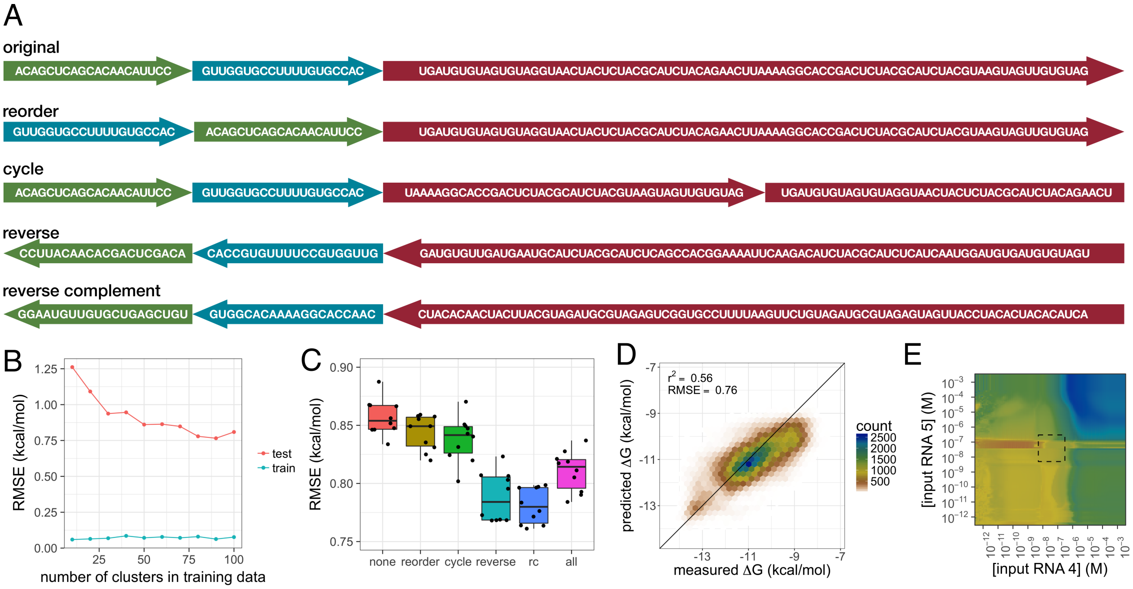


Figure 6: (A) Four different transformations were evaluated for data augmentation. The visualization shows the design sequence in red and two input sequences in green and blue. (B) A learning curve for the "single puzzle" reveals that the model is overfit, with test set RMSE exceeding train set RMSE, and lies in a high variance regime. (C) Augmentation of the training set significantly improves performance for the single puzzle model, with one-sided *t*-test *p*-values of 3.4×10^{-2} for reorder, 1.2×10^{-2} for cycle, 1.3×10^{-7} for reverse, 9.4×10^{-10} for reverse complement (RC) and 2.8×10^{-6} for all augmentation methods. (D) With reverse complement augmentation, the model achieves an RMSE of 0.76 kcal/mol. (E) In concentration extrapolation tests, the best model exhibits relatively smooth behavior compared to previous simulated models. The dotted box highlights the concentration range seen in training.


A
original
ACAGCUCAGCACAACAUUCC
GUUGGUGCCUUUUGUGCCAC
UGAUGUGUAGUGUAGGUAACUACUCUACGCAUCUACAGAACUUAAAAGGCACCGACUCUACGCAUCUACGUAAGUAGUUGUGUAG
reorder
GUUGGUGCCUUUUGUGCCAC
ACAGCUCAGCACAACAUUCC
UGAUGUGUAGUGUAGGUAACUACUCUACGCAUCUACAGAACUUAAAAGGCACCGACUCUACGCAUCUACGUAAGUAGUUGUGUAG
cycle
ACAGCUCAGCACAACAUUCC
GUUGGUGCCUUUUGUGCCAC
UAAAAGGCACCGACUCUACGCAUCUACGUAAGUAGUUGUGUAG
UGAUGUGUAGUGUAGGUAACUACUCUACGCAUCUACAGAACU
reverse
CCUUACAACACGACUCGACA
CACCGUGUUUUCCGUGGUUG
GAUGUGUUGAUGAAUGCAUCUACGCAUCUCAGCCACGGAAAAUUCAAGACAUCUACGCAUCUCAUCAAUGGAUGUGAUGUGUAGU
reverse complement
GGAAUGUUGUGCUGAGCUGU
GUGGCACAAAAGGCACCAAC
CUACACAACUACUUACGUAGAUGCGUAGAGUCGGUGCCUUUUAAGUUCUGUAGAUGCGUAGAGUAGUUACCUACACUACACAUCA
B
RMSE (kcal/mol)
number of clusters in training data
test
train
C
RMSE (kcal/mol)
none
reorder
cycle
reverse
rc
all
D
r^2 = 0.56
RMSE = 0.76
predicted ΔG (kcal/mol)
measured ΔG (kcal/mol)
count
E
[input RNA 5] (M)
10^{-3}
10^{-4}
10^{-5}
10^{-6}
10^{-7}
10^{-8}
10^{-9}
10^{-10}
10^{-11}
10^{-12}
10^{-12}
10^{-11}
10^{-10}
10^{-9}
10^{-8}
10^{-7}
10^{-6}
10^{-5}
10^{-4}
10^{-3}
[input RNA 4] (M)
ΔG