cDeepbind: A context sensitive deep learning model of RNA-protein binding

Shreshth Gandhi[1,2] **Leo J. Lee**[1] **Andrew Delong**[1] **David Duvenaud**[1] **Brendan J. Frey** [1,2]

[1] University of Toronto [2]Deep Genomics

Abstract

We present cDeepbind, a novel approach to model sequence and structure-specific binding of RNA binding proteins (RBPs). Recent deep learning approaches modelling RBP binding have used aggregate structural context vectors to represent RNA structure. Here we present an approach that generates a high-dimensional embedding of the base-pairing matrix and incorporates it into a deep neural network that predicts binding intensities for all 244 probes in RNAcompete simultaneously in a multi-task prediction setup. We observe an improvement in in-vitro prediction performance for our model compared to previous approaches and validate its ability to identify the effects of splicing mutations in real genomic contexts.

1 Introduction

RNA binding proteins (RBPs) are crucial bio-molecules that play a key role in regulating gene expression by regulating various steps of pre-mRNAs processing, including splicing, editing and polyadenylation. They allow for the generation of a large diversity of processed RNAs from the genome by regulating their maturation, stability, transport and degradation.

Many RBPs are known to have a preference for both a specific sequence and secondary structure of the target RNA [11]. The structure of RNA can affect accessibility to target sites, influencing the binding of RBPs ([7]). Several computational methods predict the secondary structure of an RNA sequence from the order of its bases based on thermodynamic stability constraints [24, 18]. It has been shown that local secondary structure restricts access to a large subset of sequence motifs that would otherwise be bound by RBPs [25].

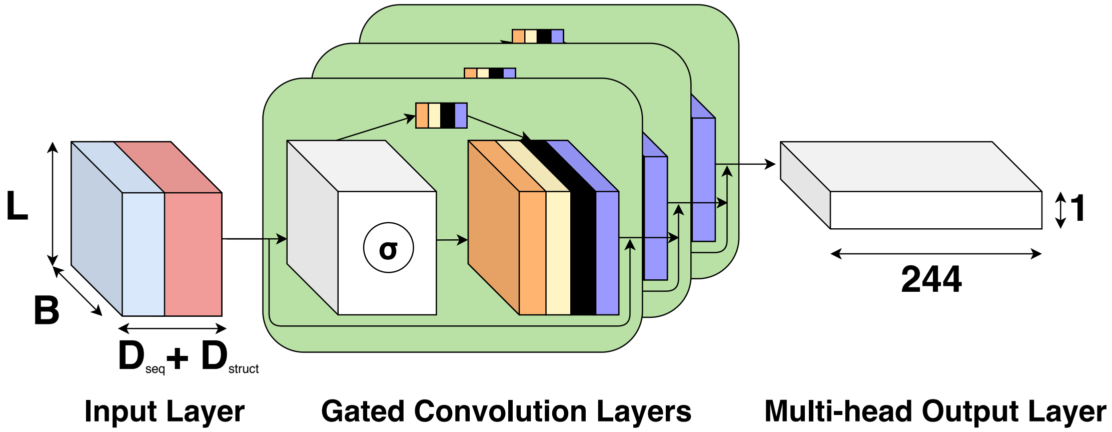

Figure 1: Model architecture for cDeepBind. Here L represents the length of the sequence, D_{seq} and D_{struct} represent the dimensions of the sequence and structure encodings, and B represents the batch-size of inputs fed to the model.

Several methods have been developed to model RBP binding preferences including methods that model both sequence and secondary structure specificity. The RNAcompete assay provided a comprehensive analysis of binding preferences

Machine Learning in Computational Biology (MLCB 2019), Vancouver, Canada.

of RBPs covering 205 genes in 24 different eukaryotes [23] and has been used as a standard benchmark for comparing in-vitro binding predictors. Deepbind [2] was the first approach to use deep convolutional neural networks(CNNs) to model RBP binding directly from sequences. However, Deepbind did not incorporate RNA secondary structure as an input feature and thus did not model secondary structure explicitly. Other methods RNAcontext [15], RCK [21] and GraphProt [19] incorporated RNA secondary structure but did not have the same expressive power as deep neural networks. DLPRB [3] improved upon RCK by using deep neural networks that use concatenated sequence and structure vectors as their input and outperformed all previous approaches on RNAcompete. The DeepRiPe model [9] jointly models RBP binding across multiple RBPs from CLIP-Seq but does not model secondary structure explicitly.

2 Methods

Inspired by previous approaches to this problem, we propose an architecture that aims to achieve the following objectives and adress shortcomings of prior work:

- Given a sequence, jointly predict binding intensities for all RBPs in RNAcompete. Multitask learning encourages the model to learn useful shared representations and should improve generalization [4]
- Generate a high-dimensional trainable representation of the RNA base-pairing matrix instead of average structure vectors.
- Use advances in neural network architecture design to explore a larger space of models that are significantly faster than Recurrent Neural Networks (RNNs) while having the same expressive power.

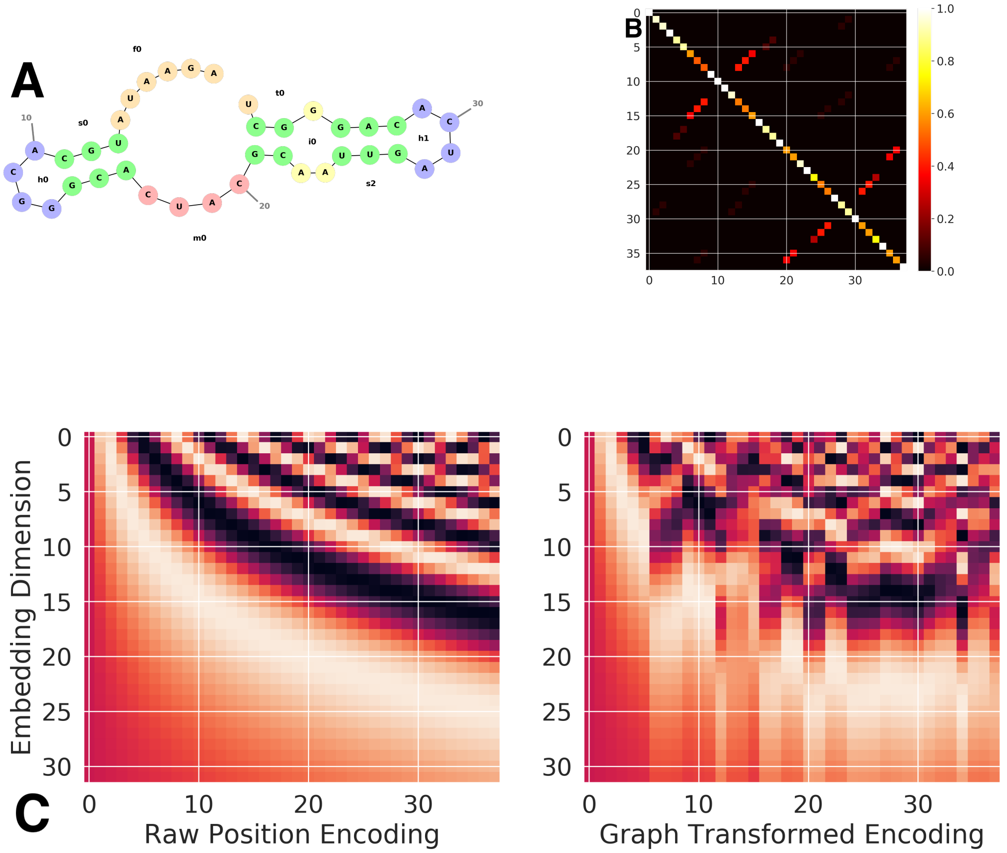

Figure 2: (A)Folding structure for a sequence from the RNAcompete dataset (B) Pairing matrix representing the folding structure (C) Transformed positional encoding obtained by multiplying the matrix in (B) with the raw position encodings

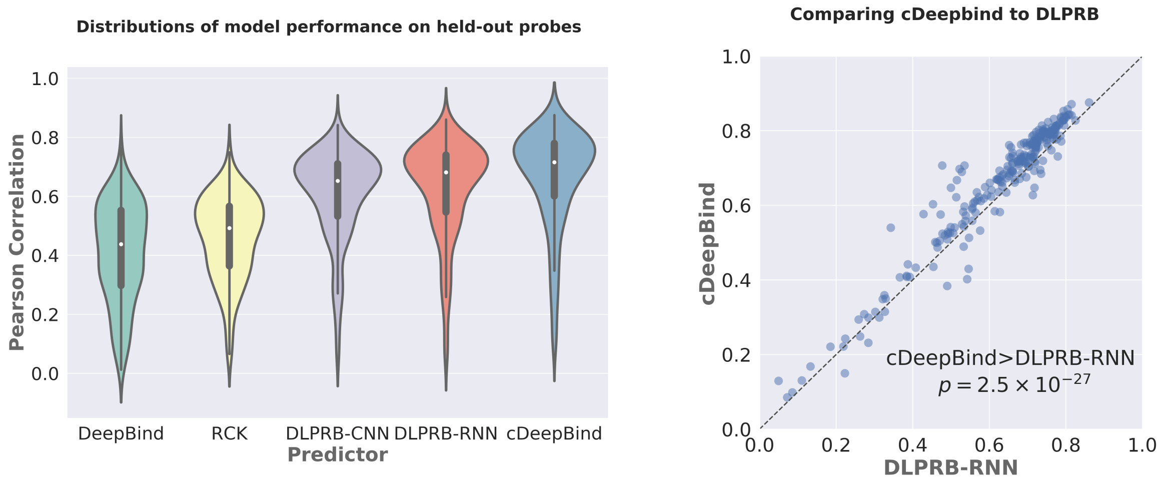

Figure 3: Evaluating cDeepbind predictions on held-out probes in RNAcompete

2.1 Architecture

The cDeepBind model employs a stack of gated convolutions with residual and skip connections. Gated convolutions have been proposed as an alternative to RNNs, that compute the gating signal in parallel for the entire input, as opposed to an RNN that process the input sequentially. Gated convolutions have outperformed RNNs on tasks such as language modelling [5] and phoneme recognition [20], and are much faster for training and inference. The encoded input is fed into a stack of 6-18 residual dilated convolution blocks [12]. We add skip connections after every 3 residual blocks. The residual blocks include a Squeeze-Excitation(SE) operation [13] that explicitly models the interdependence between the input channels. Intuitively we expect the SE operator to encourage the model to learn the dependence between the multiple output heads corresponding to each prediction target.

To allow the model to have access to the relative ordering of bases within the sequence, we use sinusoidal positional embeddings [27, 8]. This approach allows the model to generalize to sequence lengths not seen during training.

2.2 Input and Preprocessing

To encode the structural context we sample an ensemble of 10 structure graphs using Boltzman sampling [6], using the Forgi package [26].We encode the graph into a pairing matrix by first setting all diagonal values to 1. Then for paired bases, we set the value of the nucleotide they are paired with to 1, and normalize the row. Finally, we average the pairing matrices obtained for each graph. An example is shown in Figure 2.

We then multiply the pairing matrix with the sinusoidal positional encoding matrix to obtain the final structure representation.

We encode the RNA sequence as a one-hot-encoded vector and concatenate it with the transformed graph embedding representing the structure. The target RNAcompete probe intensities are clamped at the 99.95^{th} percentile and normalized to zero mean and unit variance, as done by other benchmarks on this dataset.

2.3 Training

We used random sampling to generate hyperparameters defining our model. We set the number of residual convolution layers between 6 to 18, the number of channels between 32 to 128, and the filter size between 16 and 32. For each residual block, the dilation rate is chosen randomly between $1, 2$ and 4. We randomly choose between using a gated tanh [5] or ReLU activations in our residual blocks and then use the same activation for each block within the model. We use a reduction factor of 16 in the SE unit and batch-normalization [14] before the activation. To make our training robust to outliers, we used the Huber loss function, which we confirmed empirically to work better than mean squared error. We use the mean loss across probes for each input while masking the contribution that would arise from targets with missing values. We use 5-fold cross-validation to identify the hyperparameters for our model. The final model is an ensemble of the 5 best performing models from the hyperparameter search. For each probe, we use the mean prediction from each model in the ensemble as the final predicted value. We used the Adam [16] optimizer with a batch size of 128 and a learning rate of 0.001 for 50 epochs with early stopping.

2.4 Computational resource requirements

We implemented our model in Tensorflow [1] and ran our experiments on a single machine with 2 NVIDIA TITAN V GPUs and a 12 core Intel(R) Core(TM) i7-5930K CPU. The total training time for a single multi-task model for all experiments in RNAcompete is about 20 minutes. For generating all predictions using a pre-trained model, we can encode the sequence and structure at 50 sequences/second and then generate predictions from our neural network ensemble at 1500 sequences/second.

3 Results

3.1 In-vitro evaluation

cDeepbind predictions have a higher correlation on held out probes in RNAcompete compared to other approaches. As done by other methods on this dataset we use Set A for training and hyperparameter search and report performance on probes in Set B. As shown in Figure 3 cDeepbind has an average Pearson correlation of 0.663, which is higher than state-of-the-art DLPRB-RNN that achieves 0.628 (p-value=2.5×10^{-27}, Wilcoxon signed rank test).

3.2 Predicting the effect of splicing mutations

To evaluate our model's ability to generalize to tasks outside of the in-vitro RNAcompete assay we use a set of 82 SNVs reported in [22] and recently used as an evaluation set by DeepClip [10], a deep neural network model that predicts RBP binding profiles trained on CLIP datasets. We use the binding score predicted for SRSF1 a which is known as a positive regulator of exon inclusion. We look at the difference in SRSF1 binding scores for the Wild Type and Mutant 15-mer sequences overlapping the variants and illustrate the difference in scores for the variants reported to cause skipping versus those reported to cause no exon skipping in Figure 4. For Deepbind we take the average score of the 6 RNAcompete models for SRSF1, the average of the 6 output heads for cDeepBind, and the scores provided by the authors for DeepClip. We observe a significant decrease in predicted SRSF1 binding scores (p-values computed using the Mann–Whitney U test) for mutations that cause exon-skipping. This suggests that cDeepBind can predict RBP binding in real genomic contexts as well as models trained directly on CLIP-seq.

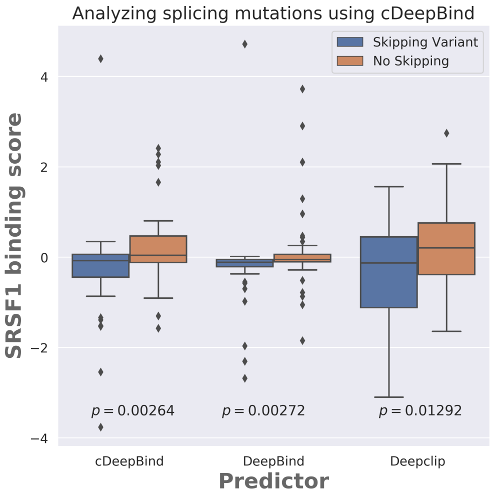

Figure 4: Evaluating RBP models for SRSF1

4 Discussion

We have presented a new approach for modelling RBP binding that addresses some of the limitations of prior work and obtains a higher pearson correlation on the RNAcompete dataset. We do not claim the superiority of our method based solely on the RNAcompete benchmark since over time it is likely that methods would have overfited to this metric. We have also evaluated our model on a set of splicing mutations to demonstrate that our model is useful beyond just in-vitro prediction. Due to the extensible nature of our method, we would like to explore training on other high-throughput binding datasets such as RNA-bind-n-seq [17] and eCLIP [29]. We are aware of the challenges associated with modelling in-vivo data but believe that our multi-task framework would be well-suited for modelling the competitive interactions of multiple RBPs. We would also like to explore integrating deep learning based methods for secondary structure prediction such as DMfold [28] for learning an end-to-end differentiable model that can incorporate secondary structure internally.

5 Software Availability

Code and Datasets used are available at https://github.com/PSI-Lab/cDeepbind

References

[1] Martin Abadi et al. “Tensorflow: Large-scale machine learning on heterogeneous distributed systems”. In: *arXiv preprint arXiv:1603.04467* (2016).

[2] Babak Alipanahi et al. “Predicting the sequence specificities of DNA-and RNA-binding proteins by deep learning”. In: *Nature biotechnology* 33.8 (2015), pp. 831–838.

[3] Ilan Ben-Bassat, Benny Chor, and Yaron Orenstein. “A Deep Learning Approach for Learning Intrinsic Protein-RNA Binding Preferences”. In: *bioRxiv* (2018), p. 328633.

[4] Rich Caruana. “Multitask learning”. In: *Machine learning* 28.1 (1997), pp. 41–75.

[5] Yann N Dauphin et al. “Language modeling with gated convolutional networks”. In: *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org. 2017, pp. 933–941.

[6] YE Ding, Chi Yu Chan, and Charles E Lawrence. “RNA secondary structure prediction by centroids in a Boltzmann weighted ensemble”. In: *Rna* 11.8 (2005), pp. 1157–1166.

[7] Olivier Duss et al. “Molecular basis for the wide range of affinity found in Csr/Rsm protein–RNA recognition”. In: *Nucleic acids research* 42.8 (2014), pp. 5332–5346.

[8] Jonas Gehring et al. “Convolutional sequence to sequence learning”. In: *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org. 2017, pp. 1243–1252.

[9] Mahsa Ghanbari and Uwe Ohler. “Deep neural networks for interpreting RNA binding protein target preferences”. In: *bioRxiv* (2019).

[10] Alexander Gulliver Bjoernholt Groenning et al. “DeepCLIP: Predicting the effect of mutations on protein-RNA binding with Deep Learning”. In: *bioRxiv* (2019), p. 757062.

[11] Jörg Hackermüller et al. “The effect of RNA secondary structures on RNA-ligand binding and the modifier RNA mechanism: a quantitative model”. In: *Gene* 345.1 (2005), pp. 3–12.

[12] Kaiming He et al. “Deep residual learning for image recognition”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 770–778.

[13] Jie Hu, Li Shen, and Gang Sun. “Squeeze-and-excitation networks”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 7132–7141.

[14] Sergey Ioffe and Christian Szegedy. “Batch normalization: Accelerating deep network training by reducing internal covariate shift”. In: *arXiv preprint arXiv:1502.03167* (2015).

[15] Hilal Kazan et al. “RNAcontext: a new method for learning the sequence and structure binding preferences of RNA-binding proteins”. In: *PLoS computational biology* 6.7 (2010), e1000832.

[16] Diederik Kingma and Jimmy Ba. “Adam: A method for stochastic optimization”. In: *arXiv preprint arXiv:1412.6980* (2014).

[17] Nicole J Lambert, Alex D Robertson, and Christopher B Burge. “RNA Bind-n-Seq: measuring the binding affinity landscape of RNA-binding proteins”. In: *Methods in enzymology*. Vol. 558. Elsevier, 2015, pp. 465–493.

[18] Ronny Lorenz et al. “ViennaRNA Package 2.0”. In: *Algorithms for Molecular Biology* 6.1 (2011), p. 26.

[19] Daniel Maticzka et al. “GraphProt: modeling binding preferences of RNA-binding proteins”. In: *Genome biology* 15.1 (2014), R17.

[20] Aaron van den Oord et al. “Wavenet: A generative model for raw audio”. In: *arXiv preprint arXiv:1609.03499* (2016).

[21] Yaron Orenstein, Yuhao Wang, and Bonnie Berger. “RCK: accurate and efficient inference of sequence-and structure-based protein–RNA binding models from RNAcompete data”. In: *Bioinformatics* 32.12 (2016), pp. i351–i359.

[22] Michela Raponi et al. “Prediction of single-nucleotide substitutions that result in exon skipping: identification of a splicing silencer in BRCA1 exon 6”. In: *Human mutation* 32.4 (2011), pp. 436–444.

[23] Debashish Ray et al. “A compendium of RNA-binding motifs for decoding gene regulation”. In: *Nature* 499.7457 (2013), pp. 172–177.

[24] Peter Steffen et al. “RNAshapes: an integrated RNA analysis package based on abstract shapes”. In: *Bioinformatics* 22.4 (2005), pp. 500–503.

[25] J Matthew Taliaferro et al. “RNA sequence context effects measured in vitro predict in vivo protein binding and regulation”. In: *Molecular cell* 64.2 (2016), pp. 294–306.

[26] Bernhard C Thiel et al. “3D based on 2D: Calculating helix angles and stacking patterns using forgi 2.0, an RNA Python library centered on secondary structure elements.” In: *F1000Research* 8 (2019).

[27] Ashish Vaswani et al. “Attention is all you need”. In: *Advances in neural information processing systems*. 2017, pp. 5998–6008.

[28] Linyu Wang et al. “DMfold: A Novel Method to Predict RNA Secondary Structure With Pseudoknots Based on Deep Learning and Improved Base Pair Maximization Principle”. In: *Frontiers in genetics* 10 (2019), p. 143.

[29] Ei-Wen Yang et al. “Allele-specific binding of RNA-binding proteins reveals functional genetic variants in the RNA”. In: *Nature communications* 10.1 (2019), p. 1338.