A Neural Model of RNA Splicing: Learning Motif Distances with Self-Attention and Toeplitz Max Pooling

Thomas P. Quinn^{1*}, Dang Nguyen¹, Sunil Gupta¹, and Svetha Venkatesh¹

¹Applied Artificial Intelligence Institute, Deakin University, Geelong, Australia

* contacttomquinn@gmail.com

Abstract

Alternative RNA splicing is an important regulator of tissue development and specificity, and a relevant mechanism in cancer progression. There exists a strong motivation to disentangle the rules that govern RNA splicing, in part because this knowledge may one day yield new clinically relevant diagnostic tools and therapeutics. It is no easy task to reverse engineer how the spliceosome machinery choreographs the removal and addition of RNA elements following transcription. Here, we propose an interpretable neural network called the Toeplitz ATtention Architecture (TATA), which learns distance-dependent motif interactions through a novel Toeplitz max pool layer that captures the relative distance between interacting CNN filters. TATA is a completely transparent “clear-box” solution: every model parameter is human-interpretable. We validate TATA on simulated data, then apply it to real data to identify putative cis-regulatory elements that interact with primary RNA splice sites.

1 Introduction

Within a cell, DNA is transcribed as RNA which is eventually translated to make proteins. Before a protein is made, the RNA may be altered through post-transcriptional modification. One such modification, called *RNA splicing*, involves “cutting out” parts of the RNA and “glueing together” the remaining parts. In *alternative splicing*, a cell can turn a single RNA precursor into several different RNAs by changing which parts are “cut out”. Alternative splicing is an important regulator of tissue development and specificity [Baralle and Giudice, 2017], and a key driver of cancer progression [Bonnal et al., 2020]. Computational models of RNA splicing can help researchers discover novel splice mechanisms that may be targetable by therapeutic agents [Garcia-Blanco et al., 2004], *but interpretable models are needed to succeed in this endeavour*.

The convolutional neural network (CNN) is a natural choice for modelling RNA splicing. The first layer of a CNN meaningfully encodes the RNA through translation-invariant filters called *motifs* [Eraslan et al., 2019]. The deeper layers of a CNN then encode the nested interactions between the motifs. Deep CNNs perform well empirically (e.g., [Koo and Eddy, 2019, Wang et al., 2019, Jaganathan et al., 2019, Bao et al., 2019, Albaradei et al., 2020]), but are opaque [Rudin, 2019]: the neural connections do not readily provide a concise knowledge representation of the learned motif interactions. This opacity is intrinsic to the deep CNN itself, arising from a mismatch between the operations of the model and the nature of human-scale reasoning [Burrell, 2016]. There is a need for new interpretable models of RNA splicing, capable of learning a concise knowledge representation of motifs and their interactions.

RNA splicing depends on more than just motifs: it also depends on *context*, most notably the distance between cooperating motifs [Ule and Blencowe, 2019]. Quinn et al. [2020] define the *distance-dependent motif interaction problem* which states that a genomic event will occur if and only if (i) two (or more) motifs are present and (ii) the motifs are separated by a gap of a specific size (see Figure 1). Learning an interpretable solution to this problem is challenging. Although one recent study proposed an interpretable motif interaction model, it uses an expensive

optimization procedure that makes a strong assumption about the distributional nature of inter-motif distances [Quinn et al., 2020]. In contrast to [Koo et al., 2018, Avsec et al., 2020], who use post-hoc explainable AI (XAI), *we seek to describe distance-dependence directly through model parameters, whereby interpretability is built-in a priori as an intentional design choice.*

Figure 1: A visual summary of the distance-dependent motif interaction problem. In the first column, two motifs are separated by the correct distance and therefore $y = 1$. In the second column, one motif is missing and therefore $y = 0$. In the third column, two motifs are separated by the correct distance but have the wrong orientation and therefore $y = 0$. In the fourth column, two motifs are separated by the incorrect distance and therefore $y = 0$. An interpretable model should explicitly identify the relevant motifs and their distance dependence.

We propose the Toeplitz ATtention Architecture (TATA) as an explicitly interpretable solution to the distance-dependent motif interaction problem. TATA adapts self-attention to learn (i) the identity of the relevant motifs and (ii) the importance of the relative distances between them (including orientation). Furthermore, it advances RNA modelling along 2 dimensions:

1. **TATA does not assume an inter-motif distance distribution:** Our novel Toeplitz max pool layer sits on top of a self-attention head, encoding the relative distance between motifs as an interpretable hidden layer. This requires no prior knowledge about the distribution of the relative distances.
2. **TATA can learn interactions between discrete sets of motifs:** A continuous relaxation of motif set membership allows the neural network to learn discrete motif sets, which in turn allows the neural network to learn the interaction between sets, thus capturing more complex motif interactions.

TATA learns distance-dependent motif interactions through a novel Toeplitz max pool layer, built on top of a modified self-attention head, that captures the relative distance between interacting CNN filters. TATA is a completely transparent “clear-box” solution: every model parameter is human-interpretable. We validate TATA on simulated data, then apply it to real data to identify putative cis-regulatory elements that interact with primary RNA splice sites.

2 Related Background

Biological motifs: A DNA/RNA sub-sequence is biologically relevant if it is recognized at a molecular level. Different sub-sequences can cause the same genomic event. A motif is an abstraction of multiple related sub-sequences. Thus, a motif is more like a regular expression than

an ASCII string. For example, the U1 snRNP protein could initiate RNA splicing by binding to any sub-sequence captured by the regular expression ‘(C|A)AGGU(G|A)AG’ [Mount et al., 1992]. Motif activity may depend on the relative distance between interacting motifs [Ule and Blencowe, 2019], for example in the case of multivalent binding by RNA-binding proteins [Sohrabi-Jahromi and Söding, 2021]. Often, there is no “perfect” inter-motif distance; rather, there exists a range of valid inter-motif distances. For example, in RNA splicing, the inter-motif distance between the branch point and the 3’ consensus splice site is usually 16-to-25 bases apart [Ohno et al., 2018].

CNNs for motif discovery: The idea of using computer algorithms to learn biological motifs is not new. For example, Stormo et al. [1982] learned a weight matrix called a *position weight matrix* that could predict whether a gene is valid. These weights, when normalized, represent the expected relative importance of each DNA base at each position, and thus define a *consensus motif*. Recently, deep learning methods have been used for motif discovery. Among model-agnostic approaches, Zhou and Troyanskaya [2015] proposed *in silico mutagenesis*, a technique that perturbs the DNA input and measures how the perturbation changes prediction. Among model-specific approaches, CNNs are a popular choice because the first layer kernel weights represent a position weight matrix. Studies have looked at how exponential activation [Koo and Ploenzke, 2019], Gaussian noise injection [Koo et al., 2019], and kernel regularization and constraints [Ploenzke and Irizarry, 2018] could make the resultant position weight matrices more robust. In TATA, we also use a convolutional layer to learn consensus motifs.

CNNs for motif interactions: Jaganathan et al. [2019] proposed a 32-layer deep CNN with 696,739 learnable parameters to predict splice site presence from DNA input. The high accuracy of this model suggests that motif context, including inter-motif distance, can be learned implicitly by the hidden layers of a CNN. We know of two studies that infer motif context rules from a deep CNN, though neither studied RNA splicing directly. Koo et al. [2018] and Avsec et al. [2020] both take a second-order *in silico mutagenesis* approach where one motif is fixed while another is inserted systematically at different relative distances to give a model-agnostic explanation of distance-dependent interactions. Although their implementations are intuitive, model-agnostic explanations can be unreliable: *an explanation of how a model appears to work may not be how the model actually works* (c.f., [Alvarez-Melis and Jaakkola, 2018a, Rudin, 2019]). In studying RNA splicing, Quinn et al. [2020] propose an explicitly interpretable model of inter-motif distance, but use an expensive nested optimization procedure, and restrict their analysis to pairwise motif interactions. In TATA, we also propose an explicitly interpretable model of inter-motif distance, but remove the distributional assumption and further capture interactions between motif sets.

Attention for biological sequences: In the quest for understanding how a model works, researchers have proposed methods for *probing* the black-box [Azodi et al., 2020]. Notable examples include methods that assign *saliency*, a measure of feature importance, to input variables based on gradients [Simonyan et al., 2014]. By selecting a reference input, importance scores can be computed for individual samples [Shrikumar et al., 2019]. In contrast, [Bahdanau et al., 2016] uses the attention mechanism to allow a model to automatically search for the important parts of an input. Attention, being related to the concept of *self-explanation* [Alvarez-Melis and Jaakkola, 2018b], can assign saliency from within the model, and has been previously used in genomics research to produce interpretable sample-specific importance scores [Beykikhoshk et al., 2020]. Attention is now an important part of transformer architectures [Vaswani et al., 2017], a light-weight alternative to CNNs for big data like whole genomes [Clauwaert and Waegeman, 2020]. Within transformers are *self-attention heads* that compute the attention between different positions of the same input. In TATA, we adapt self-attention to learn distance-dependent motif interactions.

3 Model Architecture

3.1 Overview

TATA uses self-attention to solve the distance-dependent motif interaction problem. It has 4 stages that learn to represent (1) motif discovery, (2) their interactions, (3) their context dependence, and (4) their contribution to the final prediction (see Figure 2).

- **Stage 1: Motif discovery** A convolutional layer learns to represent motifs, where the weights from one filter correspond to one motif.

- **Stage 2: Motif interaction** The motifs from Stage 1 are grouped into sets, then self-attention is used to capture multiplicative interactions along the RNA sequence.
- **Stage 3: Context dependence** The self-attention head is summarized to represent the relative distance between interacting motifs.
- **Stage 4: Final regression** A generalized linear regression makes a final prediction.

Figure 2: An overview of the TATA model, showing a single *Toeplitz head*. Stage 1 discovers the identity of the motifs and then groups them into motif sets (see Section 3.2). Stage 2 captures the interactions between motif sets (see Section 3.3). Stage 3 assigns weight to a motif set interaction based on the relative distance between the sets (see Section 3.4). Stage 4 makes the final prediction (see Section 3.5). All parameters are explicitly interpretable. K Toeplitz heads can be stacked in parallel to learn K independent interacting sets.

3.2 Stage 1: Motif discovery

Let us consider a total of N DNA sequences $\mathbf{X}$ with p bases having 4 possible states, 1-hot encoded as a $[N \times 4 \times L]$ tensor (the 4 states being adenine, cytosine, guanine, or thymine). We pass the input through a convolutional layer having F filters of size $[M \times 4]$, where F is the number of motifs and M is the maximum motif length:

$$\mathbf{H} = f_{CNN}(\mathbf{X}; \theta_m) \quad (1)$$

where the parameters θ_m represent the motif identities (as visualized after normalization by a softmax transform). The resultant $\mathbf{H}$ has the dimension $[N \times F \times L]$ and describes motif presence at each position along the DNA sequence.

We then pass $\mathbf{H}$ through a max pool layer to summarize motif presence in “chunks”. This motif compression step reduces the resolution at which the architecture detects motif interactions, but advantageously decreases the number of downstream parameters and the runtime of each epoch.

$$\hat{\mathbf{H}} = f_{pool}(\mathbf{H}, S) \quad (2)$$

where S is both the kernel size and the stride. The resultant $\hat{\mathbf{H}}$ has the dimension $[N \times F \times \frac{L}{S}]$ (or rounded up in the case that S is not a multiple of L).

3.3 Stage 2: Motif interaction

Since motifs carry meaning to domain experts in the field of genomics, $\mathbf{H}$ is an interpretable embedding of the DNA sequence. One could pass this embedding through a self-attention head, as done in natural language processing [Bahdanau et al., 2016].

To introduce self-attention into the model, we will use a variant of the attention head found in the transformer architecture (c.f., [Vaswani et al., 2017]). In transformers, the matrices $\mathbf{Q}$ and $\mathbf{K}$ are computed from the embedding matrix $\hat{\mathbf{H}}$ as

$$\mathbf{Q} = \hat{\mathbf{H}} \cdot \mathbf{W}_Q \quad (3)$$

$$\mathbf{K} = \hat{\mathbf{H}} \cdot \mathbf{W}_K \quad (4)$$

by the weights $\{\mathbf{W}_Q, \mathbf{W}_K\}$, each having the size $[F \times D]$. We interpret $\mathbf{Q}$ (and $\mathbf{K}$) to represent the presence of the motif sets. The weights $\mathbf{W}_Q$ and $\mathbf{W}_K$ define the sets, where a positive entry at coordinate $\{F, D\}$ indicates that motif F belongs to set D . A non-negative weights constraint ensures $\mathbf{W}_Q \geq 0$ (and $\mathbf{W}_K \geq 0$), reinforcing this intuition. Although these weights are not discrete, we can make them discrete with an algorithm described in the Implementation section.

The transformer attention head eventually uses 3 matrices to compute a latent representation of the attended input. In TATA, we are only interested in the query-key component

$$\mathbf{T} = \mathbf{Q} \cdot \mathbf{K}^\top. \quad (5)$$

The matrix $\mathbf{T}$ has the size $[N \times \frac{L}{S} \times \frac{L}{S}]$, and encodes the multiplicative interactions between motif sets. The term “interaction” follows from the statistical literature on non-additive effects [Tsang et al., 2018]. In $\mathbf{T}$, all possible pairwise multiplicative interactions are calculated at once by taking the dot product between the $\mathbf{Q}$ and $\mathbf{K}$ embeddings.

3.4 Stage 3: Context dependence

Each coordinate $\{i, j\}$ in $\mathbf{T}$ represents the magnitude of the interaction between the two sets at $j - i$ positions apart.¹ Due to the structure of the matrix, all elements of a given off-diagonal represent the same relative distance. For example, consider the $[4 \times 4]$ Toeplitz matrix

$$\begin{pmatrix} a & b & c & d \\ e & a & b & c \\ f & e & a & b \\ g & f & e & a \end{pmatrix} \quad (6)$$

Here, the off-diagonal b would represent all interactions at 1 position apart, c at 2 positions apart, and d at 3 positions apart (while e, f , and g represents the opposite orientation).

Our objective is to represent distance-dependent interactions *explicitly*. One way to do this is to take the maximum for each off-diagonal. This gives us a measure of the strongest interaction for a given relative distance. Put symbolically, we compute the maximum value γ_d for each integer $d \in \{-\frac{L}{S}, -\frac{L}{S} + 1, \dots, \frac{L}{S} - 1, \frac{L}{S}\}$

$$\gamma_d = \max \mathbf{T}_{i,j} \ni (i - j) = d \quad (7)$$

where each d represents one fixed inter-motif distance (e.g., $d = 2$ captures motif set interactions that are 2 bases apart). We then concatenate these as:

$$\Gamma = \text{ToeplitzMax}(\mathbf{T}) = [\gamma_{-\frac{L}{S}}, \gamma_{-\frac{L}{S}+1}, \dots, \gamma_{\frac{L}{S}-1}, \gamma_{\frac{L}{S}}] \quad (8)$$

¹The elements of $\mathbf{T}$ are always non-negative because $\mathbf{Q}$ and $\mathbf{K}$ are always non-negative. A ReLu activation is placed on $\hat{\mathbf{H}}$ while a non-negative weights constraint is placed on $\{\mathbf{W}_Q, \mathbf{W}_K\}$.

We call this operation the *Toeplitz max pool* because it reduces the matrix into a vector that describes a diagonal-constant matrix of the maximum off-diagonals. Thus, Toeplitz max pooling reduces the $[N \times \frac{L}{S} \times \frac{L}{S}]$ interaction matrix to a $[N \times (2\frac{L}{S} - 1)]$ vector.

We feed forward the results from the Toeplitz max pool layer to a single node $\mathbf{e}$ which represents the *evidence* provided by a single self-attention head. Although we could connect the relative distance vector Γ to $\mathbf{e}$ using any differentiable function, we simplify the interpretation by using a direct connection with linear activation so that

$$\mathbf{e} = \theta_e^\top \cdot \Gamma + \text{offset} \quad (9)$$

where the weights θ_e represent the importance of each off-diagonal.

3.5 Stage 4: Final regression

For $k \in \{1 \dots K\}$ attention heads stacked in parallel, each attention head will produce its own piece of evidence $\mathbf{e}^{(k)}$. We can concatenate these to form the penultimate layer

$$\mathbf{E} = [\mathbf{e}^{(1)}, \dots, \mathbf{e}^{(K)}] \quad (10)$$

which we use to estimate the final output $\mathbf{Y}$ as $\hat{\mathbf{Y}}$ via a generalized linear model

$$\hat{\mathbf{Y}} = \phi(\theta_o^\top \cdot \mathbf{E} + \text{offset}) \quad (11)$$

$$= \phi\left(\left[\theta_o^{(1)}\left(\theta_e^{(1)\top} \cdot \Gamma^{(1)}\right), \dots, \theta_o^{(K)}\left(\theta_e^{(K)\top} \cdot \Gamma^{(K)}\right)\right]\right) \quad (12)$$

where the weights θ_o represent the importance of each attention head, and ϕ is the link function chosen based on the nature of $\mathbf{Y}$ (e.g., continuous or binary). The parameters are then learned by minimizing the loss between $\mathbf{Y}$ and $\hat{\mathbf{Y}}$.

4 Implementation

4.1 Initialization by knowledge transfer from Deep CNN

Learning the motifs and relative distances simultaneously presents a difficult optimization problem (c.f., [Quinn et al., 2020]). Although we could train TATA end-to-end with back-propagation, the vanilla architecture has variable performance depending on the initialization of the motif identity layer weights θ_m . We bring greater stability to the training regime by transferring learned motif identities from a Deep CNN whose first layer is identical to TATA’s first layer. Since the first layers are identical, the weights can be transferred by direct copy.

Once transferred, the weights θ_m are frozen and TATA is trained end-to-end to learn $\mathbf{W}_Q$, $\mathbf{W}_K$, θ_e , and θ_o . After convergence, the weights θ_m are unfrozen and TATA is trained again to fine-tune all weights. The exact Deep CNN architecture is described in Experiments.

4.2 Set hardening

We interpret the weights $\mathbf{W}_Q$ and $\mathbf{W}_K$ to describe a set of motifs that interact together. Although we use a non-negative weights constraint to force positive values, the weights are otherwise free to take on any continuous value. As such, the sets described by $\mathbf{W}_Q$ and $\mathbf{W}_K$ are not really discrete sets *per se*. This complicates their interpretation.

We use a *weight hardening procedure* to discretize the weights. In weight hardening, the continuous weights $\mathbf{W}$ are replaced with discrete weights $\bar{\mathbf{W}}$ based on a cutoff c

$$\bar{W}_{fd} = \begin{cases} 1, & \text{if } W_{fd} \geq c \\ 0, & \text{otherwise} \end{cases} \quad (13)$$

for $f \in \{1 \dots F\}$ filters and $d \in \{1 \dots D\}$ sets. It follows that when the cutoff c is very large, the matrices $\mathbf{W}_Q$ and $\mathbf{W}_K$ will become very sparse. As such, set hardening also acts as a kind of regularization [Gordon-Rodriguez et al., 2021].

By conceptualizing the weights as a *continuous relaxation* of the ideal discrete weights [Maddison et al., 2017], we can use the training data to identify an *optimal cutoff* c^* , where replacing $\mathbf{W}$ with $\bar{\mathbf{W}}$ causes the smallest decrease in accuracy. To find the optimal cutoff, we trial C *candidate cutoffs* using the C largest unique values in $\{\mathbf{W}_Q, \mathbf{W}_K\}$. For each candidate cutoff, we discretize the weights and recompute the downstream layer activations.

$$\bar{\mathbf{Q}} = \hat{\mathbf{H}} \cdot \bar{\mathbf{W}}_Q \quad (14)$$

$$\bar{\mathbf{K}} = \hat{\mathbf{H}} \cdot \bar{\mathbf{W}}_K \quad (15)$$

$$\bar{\mathbf{T}} = \bar{\mathbf{Q}} \cdot \bar{\mathbf{K}}^\top \quad (16)$$

$$\bar{\Gamma} = \text{ToeplitzMax}(\bar{\mathbf{T}}) \quad (17)$$

Then, for each cutoff, we use the training data to fit a “new model” to predict the outcome from $\bar{\Gamma}$. Over 5 folds, we compute an average AUC and standard error, and select the largest cutoff (i.e., the most sparse model) whose highest AUC score is above the *maximum average AUC, minus 1 of its standard error*. Here, the “new model” is a random forest (RF), implemented using the randomForest R package with default arguments except `ntrees=25`. Once the optimal cutoff is chosen, the continuous weights are made discrete and frozen; then, the model is fine-tuned.

4.3 Training

The final loss is binary cross-entropy without any regularization. TATA is implemented in keras, and trained using the ADAM optimizer with a learning rate of 0.001. Most hyper-parameters are chosen *a priori* based on prior domain knowledge or intuition. Table 1 describes all relevant hyper-parameters, and the justification for their set value.

Module	Hyper-parameter	Value	Justification
Motif discovery	dropout	0.3	Adapted from Quinn et al. 2020
Motif discovery	F	8	Interactions usually involve ≤ 8 motifs
Motif discovery	M	8	Motifs usually have ≤ 8 bases
Motif discovery	activation	ReLU	Prevents negative motif presence
Motif compression	S	4 or 8 or 16	Tuned in experiments
Set definition	D	1	So Q and K represent single sets
Set definition	weights	non-negative	Prevents negative set membership
Set interaction	K	1 or 2 or 5	Tuned in experiments
Global	C	$6K$	Try more cutoffs with more heads
Global	epochs (initial)	500	Chosen by early simulations
Global	early stopping	15 or N/A	Stops overfit of certain data
Global	batch size	128 or 1024	Chosen by early simulations

Table 1: The hyper-parameter choices for our TATA architecture.

5 Experiments

5.1 Data

We demonstrate TATA on 2 series of simulated data sets (with 3 data sets per series), 2 real-world publicly available data sets (acceptor and donor), and 2 hypothesis-driven subsets of the real-world data. For each data set, we measure AUC on 5 independent training-test set splits (see Table 2).

- **Simulated Series 1:** These simulations represent *2-way Motif Interactions*. They are nucleotide FASTA files simulated to be an intron sequence, followed by a branch point (BP), more intron sequence, a 3' RNA splice site (3SS), then finally by an exon sequence. When $y = 1$, the BP is separated from the 3SS by an average of 21 bases (standard deviation 2). When $y = 0$, the BP is separated from the 3SS by an average of either 29, 37, or 45 bases (standard deviation 2). These correspond to *relative gaps* of 8, 16, or 24 bases. These gaps correspond to Hard, Medium, or Easy problems, respectively.

- **Simulated Series 2:** These simulations represent *3-way Motif Interactions*. They are nucleotide FASTA files simulated to be an intron sequence, followed by a branch point (BP), more intron sequence, a 3' RNA splice site (3SS), then finally by an exon sequence. This series differs from the first series in that, when $y = 1$, the intron also contains a polypyrimidine tract (PPT). When $y = 0$, the sequence either (a) does not have a PPT, (b) does not have a 3SS, or (c) has a BP that is separated from the 3SS by a relative gap of 8, 16, or 24 bases. These gaps correspond to Hard, Medium, or Easy problems, respectively.
- **Acceptor and Donor:** These data are nucleotide FASTA files of (a) acceptor vs. non-acceptor sites or (b) donor vs. non-donor sites. The data originally come from [Albaradei et al. \[2020\]](#), as separated into 5 training and test sets by [Quinn et al. \[2020\]](#).
- **Acceptor and Donor subsets:** These data are derived from [Albaradei et al. \[2020\]](#) to study whether and how context-dependence predicts splicing among select non-canonical motifs. Most acceptor sites have the trinucleotide CAG. The first subset includes acceptor (and non-acceptor) sites with a TAG instead. Similarly, most donor sites have the trinucleotide GGT. The second subset includes donor (and non-donor) sites with an AGT instead. These non-representative cases, called non-canonical sites, are chosen because the literature suggests that they may more likely depend on motif interactions.

Data Set	Max Length (L)	Total Folds (N)	Training (N)	Test (N)
2x Interactions	161	15	2744	1352
3x Interactions	159	15	2744	1352
Acceptor	102	5	38461	18944
TAG subset	102	5	8685	4277
Donor	102	5	40358	19878
AGT subset	102	5	9000	4433

Table 2: The dimensions of the data sets used to train our TATA architecture.

5.2 Baselines

We consider 9 baselines:

- **NLP-like methods:** For comparison with [Quinn et al. \[2020\]](#), we ran all baselines according to their specifications. These baselines are natural language processing (NLP) methods. They are bag-of-words (BOW), term frequency-inverse document frequency (TFIDF) [[Arora et al., 2018](#)], and Sqn2Vec, a neural embedding method [[Nguyen et al., 2019](#)].
- **Shallow CNN:** This CNN resembles the first layers of TATA: Input $\rightarrow$ 30% Dropout $\rightarrow$ CNN (8 motifs, sized $[8 \times 4]$) $\rightarrow$ Global Max Pool. TATA also uses 8 motifs sized $[8 \times 4]$.
- **Wide CNN:** This CNN also resembles the first layers of TATA, but has more parameters: Input $\rightarrow$ 30% Dropout $\rightarrow$ CNN (32 motifs, sized $[32 \times 4]$) $\rightarrow$ Global Max Pool.
- **Wider CNN:** This CNN also resembles the first layers of TATA, but has more parameters: Input $\rightarrow$ 30% Dropout $\rightarrow$ CNN (64 motifs, sized $[64 \times 4]$) $\rightarrow$ Global Max Pool.
- **Deep CNN:** The first layers of the Deep CNN resemble the first layers of TATA: Input $\rightarrow$ 30% Dropout $\rightarrow$ CNN (8 motifs, sized $[8 \times 4]$). The following layers are: Strided Max Pool (sized $[4 \times 1]$) $\rightarrow$ 30% Dropout $\rightarrow$ CNN (8 motifs sized $[8 \times 1]$) $\rightarrow$ Strided Max Pool² (sized $[4 \times 1]$) $\rightarrow$ Output. This is also the network used to initialize TATA as described in the Implementation section.

The similarities between TATA and the CNNs make it possible to evaluate the contribution of the Toeplitz Max Pool module as compared with having zero or more hidden layers.

²The acceptor and donor data are artefactually centered on the labelled splice site. For these data, the final strided max pool layer is replaced with a global max pool layer to erase the positional information.

6 Results

6.1 TATA distills knowledge about motif interactions

TATA has 5 sets of weights. They each represent different aspects of RNA splicing. The constraints placed on these weights, along with the operations that relate them to the output, ensure that the weights are biologically relevant and human-interpretable:

- $\{\theta_m\}$ from Eq 1 define the motifs themselves. A softmax transform normalizes the weights as a position weight matrix, which can be visualized via a seqLogo plot.
- $\{\mathbf{W}_Q, \mathbf{W}_K\}$ from Eq 4 define the motif membership within the interacting sets. They are, by design, non-negative, and can be discretized through our *weight hardening procedure*.
- $\{\theta_e, \theta_o\}$ from Eq 12 define the importance of each relative distance in predicting the outcome $\mathbf{y}$. Although θ_e and θ_o are separate parameters within the model, they can be multiplied together to get a single importance score (see Eq. 12).

There are no additional parameters within the model. Thus, these parameters together provide a complete description of model behavior.

6.2 TATA learns 2- and 3-way interactions

Figure 3 shows the performance of the TATA architecture compared with baselines run on 6 simulated data sets. For the 2-way interactions, only the Deep CNN and TATA models achieve high accuracy. Here, TATA performs best, perhaps because the Toeplitz head can explicitly represent the exact mechanism we simulated, yielding a more generalizable prediction than the Deep CNN layer. For the 3-way interactions, TATA matches the Shallow, Wide, and Deep CNNs. The 3-way interactions may be easier for the CNN to model because many examples in the negative class have 1 of the 3 motifs missing.

Figure 3: Test set AUC across 6 simulated data sets, using 5 independent training sets each. Our TATA architecture performs as well as, or better than, several state-of-the-art baselines. However, its parameters are explicitly interpretable. Here, we show TATA when $S = 4$ and $K = 5$. Hard, Medium, and Easy correspond to relative gaps of 8, 16, or 24 bases.

6.3 Continuous relaxation enhances mechanistic interpretations

The weights $\{\mathbf{W}_Q, \mathbf{W}_K\}$ define motif set membership. To solidify the interpretation of $\{\mathbf{W}_Q, \mathbf{W}_K\}$ as sets, we use a *weight hardening procedure* to discretize these parameters as 0 or 1. Figure 4 shows the relative difference in test set AUC before and after hardening, showing that hardening does not considerably impair model performance. Visually, we see that AUC drops by about 2% AUC on average, with a few outliers having worse performance.

Figure 4: Change in test set AUC, before hardening vs. after hardening, across 6 simulated data sets, using 5 independent training sets each. Set hardening does not considerably impair performance, but can substantially improve interpretability.

With discretized sets, it becomes possible to describe each Toeplitz Max Pool head using a concise *visual-semantic representation*. Figure 5 provides one example, where every model parameter is combined into a single “dashboard” description of model behavior. Motifs, shown with seqLogo plots, are bracketed into sets. In this case, one set contains a single motif whose elaborate identity is better expressed as multiple distinct position weight matrices, perhaps as a compensation for motif compression. The line plot shows the importance of each relative distance, centered at a relative distance of 0. This figure, learned on simulated data, perfectly describes the rules used to simulate the data: the genomic event is most likely to occur when the branchpoint sequence (BPS, with regex ‘(C|T)T*A(C|T)’) is separated from the 3’ splice site (3SS, with regex ‘*(C|T)AGG’) by ~ 21 bases, and it is least likely to occur when the motifs are separated by ~ 45 bases.

Figure 5: A concise visual-semantic representation of one Toeplitz Max Pool Head and all associated parameters. This representation, learned from the simulated data, accurately captures the ground-truth mechanism. The motifs, shown as seqLogo plots and regular expressions, are bracketed into sets that interact. The line plot shows the importance of the interaction for each relative distance. Here $K = 1$, but there would be K representations for K Toeplitz heads.

We note that high negative weight is also assigned when the BPS and 3SS are separated by ~ 150 bases in either direction. This appears to be an artefact of the simulation, whereby the positive class sequences tend to be shorter than the negative class sequences. As such, when the BPS and something 3SS-like are very far apart, for example by chance, the sequence must be very long and therefore assigned to the negative class. Such examples seem to appear rarely enough that the ground-truth mechanism is still detected.

6.4 Application to RNA Splice Site Prediction

Having established the reliability of the interpretations using simulated data, we benchmark TATA performance on real-world data, then carve out a hypothesis-driven subset of the data for further exploration. Figure 6 shows the average test set AUC for the 2 data sets and their subsets. Visually, we see that the TATA architecture is largely comparable with Shallow, Wide, and Deep CNNs, but with the added advantage of its parameters being explicitly interpretable.

Figure 6: Test set AUC across 2 real data sets, using 5 independent training sets each. Our TATA architecture performs as well as, or better than, several state-of-the-art baselines. However, its parameters are explicitly interpretable. Here, we show TATA when $S = 4$ and $K = 5$.

In the case of the real splicing data, one could achieve decent accuracy by considering only the presence or absence of canonical splice motif, without considering context-dependent motif interactions. Non-canonical motifs occur less rarely, and are a less precise predictor of RNA splicing (i.e., a non-canonical motif may appear in a sequence without causing a splice may event). We hypothesize that non-canonical splice sites are harder to predict without also considering context-dependent motif interactions. Therefore, we subset the [Albaradei et al. \[2020\]](#) data to include only specific non-canonical splice sites in the positive class. Although we do not know the ground-truth mechanism, we can inspect the parameters of our neural network directly to generate hypotheses about how non-canonical splice sites might get recognized. For this, we viewed the visual-semantic representations of the model across 5 folds and $K = \{1, 2, 5\}$ Toeplitz heads. Figure 7 presents a clear example of a pattern that appears more generally within the TAG-only acceptor subset data. Here a TA-rich sequence precedes a GC-rich sequence by about 50 base pairs. The poly-pyrimidine tract is defined by a repetition of Ts and As, and is known to mediate acceptor site splicing [[Ohno et al., 2018](#)]. Splice enhancers are often defined by Gs and Cs [[Cartegni et al., 2003](#)], and are known to play an important role in splicing when other canonical elements are lacking [[Murray et al., 2008](#)]. From this, we can formulate a testable hypothesis that having a GC-rich splice enhancer ~ 50 bases ahead of a poly-pyrimidine tract signals causes the RNA splicing machinery to recognize the non-canonical splice motif as a valid splice site.

Figure 7: A concise visual-semantic representation of one Toeplitz Max Pool Head and all associated parameters. This representation, learned from the TAG-only acceptor subset data, can inform a testable hypothesis about a mechanism in non-canonical splice site detection: having a GC-rich splice enhancer ~ 50 bases ahead of a poly-pyrimidine tract signals causes the RNA splicing machinery to recognize the non-canonical splice motif as a valid splice site.

7 Conclusions

Here, we propose the TATA architecture for learning distance-dependent motif interactions through a novel Toeplitz max pool layer, built on top of a modified self-attention head. As a software tool, TATA can advance research into the basic biology of RNA splicing by generating a concise, human-interpretable knowledge representation of RNA splicing. This knowledge representation can help generate hypotheses about putative splice mechanisms, feeding back into the research cycle via a bit-to-bedside pipeline in which computational discoveries inform subsequent biomedical experiments. Although we focus on RNA splicing here, the method is generally applicable to other problems where distance-dependent motif interactions trigger genomic events.

8 Availability of Data and Materials

All data and code will become publicly available after peer review.

References

Francisco E Baralle and Jimena Giudice. Alternative splicing as a regulator of development and tissue identity. *Nature reviews. Molecular cell biology*, 18(7):437–451, July 2017. ISSN 1471-0072. doi: 10.1038/nrm.2017.27. URL <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6839889/>.

- Sophie C. Bonnal, Irene López-Oreja, and Juan Valcárcel. Roles and mechanisms of alternative splicing in cancer - implications for care. *Nature Reviews. Clinical Oncology*, 17(8):457–474, August 2020. ISSN 1759-4782. doi: 10.1038/s41571-020-0350-x.
- Mariano A. Garcia-Blanco, Andrew P. Baraniak, and Erika L. Lasda. Alternative splicing in disease and therapy. *Nature Biotechnology*, 22(5):535–546, May 2004. ISSN 1546-1696. doi: 10.1038/nbt964. URL <https://www.nature.com/articles/nbt964>.
- Gökçen Eraslan, Žiga Avsec, Julien Gagneur, and Fabian J. Theis. Deep learning: new computational modelling techniques for genomics. *Nature Reviews Genetics*, 20(7):389, July 2019. ISSN 1471-0064. doi: 10.1038/s41576-019-0122-6. URL <https://www.nature.com/articles/s41576-019-0122-6>.
- Peter K. Koo and Sean R. Eddy. Representation learning of genomic sequence motifs with convolutional neural networks. *PLOS Computational Biology*, 15(12):e1007560, December 2019. ISSN 1553-7358. doi: 10.1371/journal.pcbi.1007560. URL <https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1007560>.
- Ruohan Wang, Zishuai Wang, Jianping Wang, and Shuaicheng Li. SpliceFinder: ab initio prediction of splice sites using convolutional neural network. *BMC Bioinformatics*, 20(23):652, December 2019. ISSN 1471-2105. doi: 10.1186/s12859-019-3306-3. URL <https://doi.org/10.1186/s12859-019-3306-3>.
- Kishore Jaganathan, Sofia Kyriazopoulou Panagiotopoulou, Jeremy F. McRae, Siavash Fazel Darbandi, David Knowles, Yang I. Li, Jack A. Kosmicki, Juan Arbelaez, Wenwu Cui, Grace B. Schwartz, Eric D. Chow, Efsthios Kanterakis, Hong Gao, Amirali Kia, Serafim Batzoglou, Stephan J. Sanders, and Kyle Kai-How Farh. Predicting Splicing from Primary Sequence with Deep Learning. *Cell*, 176(3):535–548.e24, January 2019. ISSN 0092-8674. doi: 10.1016/j.cell.2018.12.015. URL <http://www.sciencedirect.com/science/article/pii/S0092867418316295>.
- Suying Bao, Daniel F. Moakley, and Chaolin Zhang. The Splicing Code Goes Deep. *Cell*, 176(3):414–416, January 2019. ISSN 0092-8674. doi: 10.1016/j.cell.2019.01.013. URL <http://www.sciencedirect.com/science/article/pii/S0092867419300467>.
- Somayah Albaradei, Arturo Magana-Mora, Maha Thafar, Mahmut Uludag, Vladimir B. Bajic, Takashi Gojobori, Magbubah Essack, and Boris R. Jankovic. Splice2Deep: An ensemble of deep convolutional neural networks for improved splice site prediction in genomic DNA. *Gene: X*, 5:100035, December 2020. ISSN 2590-1583. doi: 10.1016/j.gene.2020.100035. URL <http://www.sciencedirect.com/science/article/pii/S2590158320300097>.
- Cynthia Rudin. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, 1(5):206–215, May 2019. ISSN 2522-5839. doi: 10.1038/s42256-019-0048-x. URL <https://www.nature.com/articles/s42256-019-0048-x>.
- Jenna Burrell. How the machine ‘thinks’: Understanding opacity in machine learning algorithms. *Big Data & Society*, 3(1):2053951715622512, June 2016. ISSN 2053-9517. doi: 10.1177/2053951715622512. URL <https://doi.org/10.1177/2053951715622512>.
- Jernej Ule and Benjamin J. Blencowe. Alternative Splicing Regulatory Networks: Functions, Mechanisms, and Evolution. *Molecular Cell*, 76(2):329–345, October 2019. ISSN 1097-2765. doi: 10.1016/j.molcel.2019.09.017. URL <http://www.sciencedirect.com/science/article/pii/S1097276519307026>.
- Thomas P. Quinn, Dang Nguyen, Phuoc Nguyen, Sunil Gupta, and Svetha Venkatesh. Learning distance-dependent motif interactions: an interpretable CNN model of genomic events. *bioRxiv*, page 2020.08.27.270967, August 2020. doi: 10.1101/2020.08.27.270967. URL <https://www.biorxiv.org/content/10.1101/2020.08.27.270967v1>.
- Peter K. Koo, Praveen Anand, Steffan B. Paul, and Sean R. Eddy. Inferring Sequence-Structure Preferences of RNA-Binding Proteins with Convolutional Residual Networks. *bioRxiv*, page 418459, September 2018. doi: 10.1101/418459. URL <https://www.biorxiv.org/content/10.1101/418459v1>.

- Žiga Avsec, Melanie Weilert, Avanti Shrikumar, Sabrina Krueger, Amr Alexandari, Khyati Dalal, Robin Fropf, Charles McAnany, Julien Gagneur, Anshul Kundaje, and Julia Zeitlinger. Deep learning at base-resolution reveals cis-regulatory motif syntax. *bioRxiv*, page 737981, March 2020. doi: 10.1101/737981. URL <https://www.biorxiv.org/content/10.1101/737981v2>.
- S M Mount, C Burks, G Hertz, G D Stormo, O White, and C Fields. Splicing signals in Drosophila: intron size, information content, and consensus sequences. *Nucleic Acids Research*, 20(16):4255–4262, August 1992. ISSN 0305-1048. URL <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC334133/>.
- Salma Sohrabi-Jahromi and Johannes Söding. Thermodynamic modeling reveals widespread multivalent binding by RNA-binding proteins. *bioRxiv*, page 2021.01.30.428941, February 2021. doi: 10.1101/2021.01.30.428941. URL <https://www.biorxiv.org/content/10.1101/2021.01.30.428941v1>.
- Kinji Ohno, Jun-Ichi Takeda, and Akio Masuda. Rules and tools to predict the splicing effects of exonic and intronic mutations. *Wiley interdisciplinary reviews. RNA*, 9(1), 2018. ISSN 1757-7012. doi: 10.1002/wrna.1451.
- G D Stormo, T D Schneider, L Gold, and A Ehrenfeucht. Use of the ‘Perceptron’ algorithm to distinguish translational initiation sites in E. coli. *Nucleic Acids Research*, 10(9):2997–3011, May 1982. ISSN 0305-1048. URL <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC320670/>.
- Jian Zhou and Olga G. Troyanskaya. Predicting effects of noncoding variants with deep learning-based sequence model. *Nature Methods*, 12(10):931–934, October 2015. ISSN 1548-7105. doi: 10.1038/nmeth.3547. URL <https://www.nature.com/articles/nmeth.3547>.
- Peter K. Koo and Matt Ploenzke. Improving Convolutional Network Interpretability with Exponential Activations. *bioRxiv*, page 650804, May 2019. doi: 10.1101/650804. URL <https://www.biorxiv.org/content/10.1101/650804v1>.
- Peter K. Koo, Sharon Qian, Gal Kaplun, Verena Volf, and Dimitris Kalimeris. Robust Neural Networks are More Interpretable for Genomics. *bioRxiv*, page 657437, June 2019. doi: 10.1101/657437. URL <https://www.biorxiv.org/content/10.1101/657437v1>.
- M. S. Ploenzke and R. A. Irizarry. Interpretable Convolution Methods for Learning Genomic Sequence Motifs. *bioRxiv*, page 411934, September 2018. doi: 10.1101/411934. URL <https://www.biorxiv.org/content/10.1101/411934v1>.
- David Alvarez-Melis and Tommi S. Jaakkola. On the Robustness of Interpretability Methods. *2018 ICML Workshop on Human Interpretability in Machine Learning*, June 2018a. URL <https://arxiv.org/abs/1806.08049v1>.
- Christina B. Azodi, Jiliang Tang, and Shin-Han Shiu. Opening the Black Box: Interpretable Machine Learning for Geneticists. *Trends in genetics: TIG*, 36(6):442–455, June 2020. ISSN 0168-9525. doi: 10.1016/j.tig.2020.03.005.
- Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps. *arXiv:1312.6034 [cs]*, April 2014. URL <http://arxiv.org/abs/1312.6034>. arXiv: 1312.6034.
- Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. Learning Important Features Through Propagating Activation Differences. *arXiv:1704.02685 [cs]*, October 2019. URL <http://arxiv.org/abs/1704.02685>. arXiv: 1704.02685.
- Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural Machine Translation by Jointly Learning to Align and Translate. *arXiv:1409.0473 [cs, stat]*, May 2016. URL <http://arxiv.org/abs/1409.0473>. arXiv: 1409.0473.
- David Alvarez-Melis and Tommi S. Jaakkola. Towards Robust Interpretability with Self-Explaining Neural Networks. June 2018b. URL <https://arxiv.org/abs/1806.07538v2>.

- Adham Beykikhoshk, Thomas P. Quinn, Samuel C. Lee, Truyen Tran, and Svetha Venkatesh. DeepTRIAGE: interpretable and individualised biomarker scores using attention mechanism for the classification of breast cancer sub-types. *BMC Medical Genomics*, 13(3):20, February 2020. ISSN 1755-8794. doi: 10.1186/s12920-020-0658-5. URL <https://doi.org/10.1186/s12920-020-0658-5>.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention Is All You Need. *arXiv:1706.03762 [cs]*, December 2017. URL <http://arxiv.org/abs/1706.03762>. arXiv: 1706.03762.
- J. Clauwaert and W. Waegeman. Novel transformer networks for improved sequence labeling in genomics. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, pages 1–1, 2020. ISSN 1557-9964. doi: 10.1109/TCBB.2020.3035021.
- Michael Tsang, Dehua Cheng, and Yan Liu. Detecting Statistical Interactions from Neural Network Weights. *Sixth International Conference on Learning Representations*, 2018. URL <http://arxiv.org/abs/1705.04977>. arXiv: 1705.04977.
- Elliott Gordon-Rodriguez, Thomas P. Quinn, and John P. Cunningham. Learning Sparse Log-Ratios for High-Throughput Sequencing Data. *bioRxiv*, page 2021.02.11.430695, February 2021. doi: 10.1101/2021.02.11.430695. URL <https://www.biorxiv.org/content/10.1101/2021.02.11.430695v1>.
- Chris J. Maddison, Andriy Mnih, and Yee Whye Teh. The Concrete Distribution: A Continuous Relaxation of Discrete Random Variables. *arXiv:1611.00712 [cs, stat]*, March 2017. URL <http://arxiv.org/abs/1611.00712>. arXiv: 1611.00712.
- Sanjeev Arora, Mikhail Khodak, Nikunj Saunshi, and Kiran Vodrahalli. A Compressed Sensing View of Unsupervised Text Embeddings, Bag-of-n-Grams, and LSTMs. February 2018. URL <https://openreview.net/forum?id=B1e5ef-C->.
- Dang Nguyen, Wei Luo, Tu Dinh Nguyen, Svetha Venkatesh, and Dinh Phung. Sqn2Vec: learning sequence representation via sequential patterns with a gap constraint. In *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2018 Dublin, Ireland, September 10–14, 2018 Proceedings, Part II*, pages 569–584. Springer, 2019. doi: 10.1007/978-3-030-10928-8_34. URL <https://research.monash.edu/en/publications/sqn2vec-learning-sequence-representation-via-sequential-patterns->.
- Luca Cartegni, Jinhua Wang, Zhengwei Zhu, Michael Q. Zhang, and Adrian R. Krainer. ESEfinder: A web resource to identify exonic splicing enhancers. *Nucleic Acids Research*, 31(13):3568–3571, July 2003. ISSN 1362-4962. doi: 10.1093/nar/gkg616.
- Jill I Murray, Rodger B Voelker, Kristy L Henscheid, M Bryan Warf, and J Andrew Berglund. Identification of motifs that function in the splicing of non-canonical introns. *Genome Biology*, 9(6):R97, 2008. ISSN 1465-6906. doi: 10.1186/gb-2008-9-6-r97. URL <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2481429/>.