**Introduction**

RNA binding proteins (RBPs) bind to sequence of pre-mRNAs to direct their processing and bind mature mRNAs to control their translation, localization, and stability. Mapping protein-RNA interactions can contribute to the understanding of posttranscriptional regulation.

An RNA sequence typically contains a motif, usually a substring of 6-12 bp, to which the protein binds. In order to predict whether a specific protein binds to a given RNA sequence, a motif should be detected. Given a specific RBP, biological methods are able to produce a set of RNA sequences that are bound to the respective RBP. The identification of motifs by non-DNN methods is performed by finding motif enrichment, using PWM (position weight matrix) or by counting k-mers.

**Goal of the project**Create a DNN model predicting how likely is it for a specific protein to bind to an RNA sequence using a dataset created by RBNS technology (see description below). Use this model to predict binding of the same protein on a dataset created using RNCMPT (see description below) technology.

**Datasets**

The learning process is performed on the RNA Bind-n-Seq (see description below), and the testing is performed on a different dataset of RNA-compete (see description below). The binding model for each RBP is used to rank RNA-compete probes.

Training Files

There are 16 proteins (RBP1-RBP16).

**RBNS:** 6 files for each RBP: one file with zero RBP concentration, and five files with increasing RBP concentrations, typically starting from 5nM where each file has four times higher concentration than the previous one. Each file contains between 10 and 20 million RNA sequences of 20 base-pairs.

**RNCMPT:** One file for each RBP containing ~250,000 RNA sequences of length ~40, sorted by binding intensity. The first 100 sequences are considered as positives, the other are considered as negatives.

Test files

There are 15 proteins (RBP17-RBP31), different from the training dataset.

**RBNS:** Same as in the training dataset

**RNCMPT:** Same as in the training dataset, only not sorted.

**Technical description of the datasets**

**RNA Bind-n-Seq (RBNS)**  
RNA Bind-n-Seq dataset (RBNS) is designed to dissect the sequence and RNA structural preferences of RBPs. An RBP is incubated with a pool of randomized RNAs at several different protein concentrations, typically ranging from low nanomolar to low micromolar.

The RNA pool typically consists of random RNAs of length 40 nt flanked by short primers used to add the adapters needed for deep sequencing.

RBPbound RNA is reverse-transcribed into cDNA, and barcoded sequencing adapters are added by PCR to produce libraries for deep sequencing. Libraries corresponding to the input RNA pool and to five or more RBP concentrations (including zero RBP concentration as an additional control) are sequenced in a single Illumina HiSeq 2000 lane, typically yielding at least 15– 20 million reads per library.

Most RBPs bind single-stranded RNA sequence motifs 3–8 nt in length.

The modeled enrichment profiles show that R values (motif enrichment) of high-affinity motifs decrease as RBP concentrations become very high under all conditions tested. This effect is readily understood by considering that high RBP concentrations will tend to drive binding toward lower affinity RNAs (and high-affinity motifs may become saturated), resulting in a lower fraction of high-affinity motifs in RBP-bound RNA. These simulations also showed that even a small amount of nonspecific binding to the apparatus greatly reduces R values at very low RBP concentrations, because nonspecifically recovered RNA dilutes the small amount of specifically bound RNA. Together, these two effects produce a characteristic unimodal curve that peaks at intermediate RBP concentrations under a wide range of assumptions about affinities

**RNA-compete (RNCMPT)**

A sequence library covering all 9-mers, each at least 16 times. RNA-compete consists of three basic steps: (i) generation of an RNA pool comprising a variety of RNA sequences and structures; (ii) a single pulldown of the RNAs bound to a tagged RBP of interest; and (iii) microarray and computational interrogation of the relative enrichment of each RNA in the bound fraction relative to the starting pool.

In each RNA-compete experiment, the bindings of one protein to around 240,000 short synthetic RNAs (30 to 40 nucleotides long) are measured.

**Model description**

We would like for the model to detect motifs among the sequences. Hence, a convolutional neural network is a right fit for the problem – a kernel can represent a motif similarly to a PWM.

**Dataset preprocessing**

An RNA sequence of length ℓ is a string of nucleotides over the alphabet Σ={A,G,C,U}. We encode every nucleotide as a one-hot vector of dimension 4. An additional nucleotide N represents an unknown base and is encoded by {0.25, 0.25, 0.25, 0.25}.

We created a balanced set containing XX sequences:

**Positive set**: 2 files with the highest concentrations

**Negative set**: for each sequence in the positive set, we created a shuffled version

**Input**

The encoding of an RNA sequence

**Output**

Score – a value between 0 and 1, using the sigmoid function

The general overview of our network:

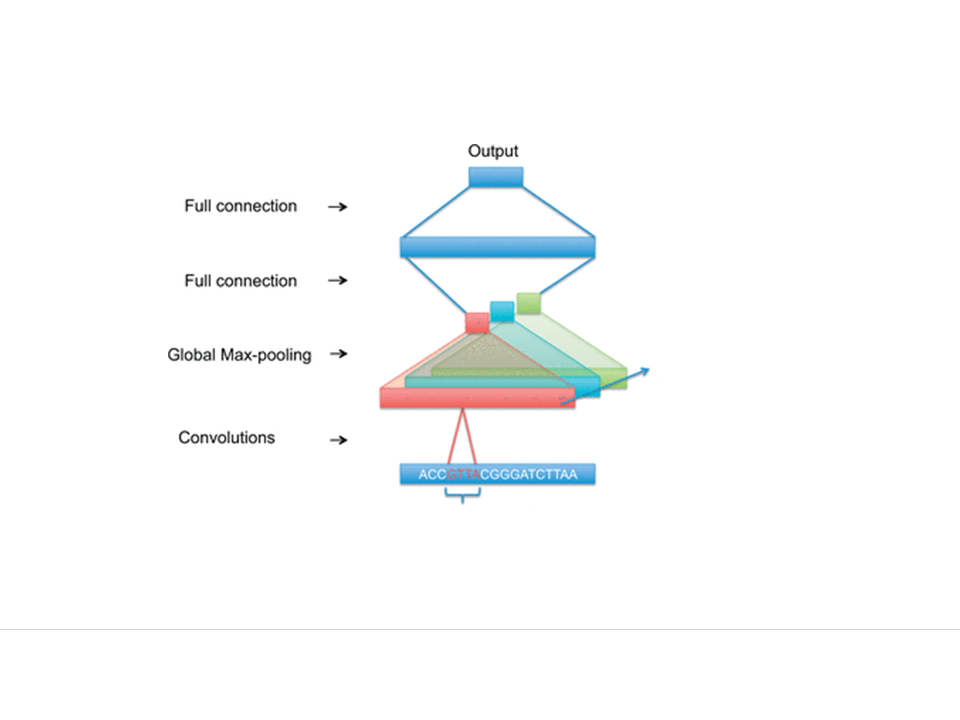


Figure taken from: Bioinformatics, Volume 32, Issue 12, 15 June 2016, Pages i121–i127, https://doi.org/10.1093/bioinformatics/btw255

The kernels should locate motifs, more kernels provide the opportunity to find different motifs or different variants of the same motif. The purpose of the global max-pooling is to determine if a motif exists in the input sequence or not.

We found that different kernel sizes perform differently on different proteins. This can be explained by the fact that motifs can be of varying length. To overcome this, we decided to use three different kernel sizes.

To work with different kernel sizes, for each sequence we created three one hot encodings, with different lengths of padding regions (Depends on the kernel size). We fed each one hot encoding to the corresponding convolution layer, and applied max pooling. Afterwards we concatenated all the results from the convolution layer to a single vector, and fed this vector to one fully connected layer.

|  |  |
| --- | --- |
| Kernel sizes | 6, 8, 10 |
| Number of kernels | 32, 32, 32 |
| Dense layer size | 32 |
| Optimizer, learning rate |  |
| Batch size | 264 |
| Epochs |  |

**What did not work**

The following methods yielded random results on RNCMPT

* Prediction of 6 classes: random sequences file and 5 concentration files. The score for RNCMPT is the sum of scores for each concentration file.
* Binary model of the random sequences file versus one of the first three files
* Creating random sequences for the negative set

The following methods did not improve results:

* Filtering to only sequences that appear more than one time
* Counting k-grams that appeared more in the positive dataset than in the negative one, and filtering the positive dataset accordingly. Made slight improvements on some proteins, but harmed the others.

**Results on the train RNCMPT set**

**Random results**

100 random runs yielded an average AUPR of 0.004 (1-10 positives).

**Benchmark**

For a performance baseline, 7-mer z-scores (defined in the RNA bind-n-seq paper) were used to score RNAcompete probes. The score of each probe is the sum of 7-mer z-scores in it.

The AUPR results for RBP1 to RBP16 achieved by this method:

|  |  |  |
| --- | --- | --- |
| Protein | Benchmark | Our best result |
| RBP1 | 0.07473610521979981 | 0.0455388926186549 |
| RBP2 | 0.04254748703001486 | 0.022516612005553653 |
| RBP3 | 0.007844351178510244 | 0.009072449948776679 |
| RBP4 | 0.00486103693158827 | 0.004244341033218847 |
| RBP5: | 0.007071545992948212 | 0.004548838977976385 |
| RBP6: | 0.04031246050463821 | 0.014826312026258753 |
| RBP7: | 0.157394086104544 | 0.19456106376025395 |
| RBP8: | 0.009758432348908962 | 0.004008459404957258 |
| RBP9: | 0.003731964318077849 | 0.006248319230634378 |
| RBP10: | 0.01106059960001746 | 0.005744016723631379 |
| RBP11: | 0.0070360302753930655 | 0.042577303190533634 |
| RBP12 | 0.10825262758132842 | 0.07050639730061498 |
| RBP13 | 0.07125136106156829 | 0.06519439302182424 |
| RBP14 | 0.016075801633795135 | 0.016044323482680513 |
| RBP15 | 0.023859220487953848 | 0.005589651952765266 |
| RBP16 | 0.0711428765901181 | 0.03734592695723653 |
| Average | 0.041 | 0.0342854563522232 |

**Performance (time, memory, CPU)**

Running time for one epoch:

Total running time:

Memory:

CPU: