

# Interpretable Convolutional Neural Networks for Effective Translation Initiation Site Prediction

Jasper Zuallaert<sup>\*†</sup>, Mijung Kim<sup>\*†</sup>, Yvan Saeys<sup>‡§</sup>, and Wesley De Neve<sup>\*†</sup>

<sup>\*</sup>Center for Biotech Data Science, Ghent University Global Campus, Songdo, Incheon, 305-701, South Korea

<sup>†</sup>IDLab, ELIS, Ghent University - imec, Ghent, 9000, Belgium

<sup>‡</sup>VIB-UGent Center for Inflammation Research, Technologiepark 927, 9052 Zwijnaarde-Ghent, Belgium

<sup>§</sup>Department of Applied Mathematics, Computer Science and Statistics, Ghent University, Krijgslaan 281, S9, 9000 Ghent, Belgium

**Abstract**—Thanks to rapidly evolving sequencing techniques, the amount of genomic data at our disposal is growing increasingly large. Determining the gene structure is a fundamental requirement to effectively interpret gene function and regulation. An important part in that determination process is the identification of translation initiation sites. In this paper, we propose a novel approach for automatic prediction of translation initiation sites, leveraging convolutional neural networks that allow for automatic feature extraction. Our experimental results demonstrate that we are able to improve the state-of-the-art approaches with a decrease of 75.2% in false positive rate and with a decrease of 24.5% in error rate on chosen datasets. Furthermore, an in-depth analysis of the decision-making process used by our predictive model shows that our neural network implicitly learns biologically relevant features from scratch, without any prior knowledge about the problem at hand, such as the Kozak consensus sequence, the influence of stop and start codons in the sequence and the presence of donor splice site patterns. In summary, our findings yield a better understanding of the internal reasoning of a convolutional neural network when applying such a neural network to genomic data.

## I. INTRODUCTION

In recent years, with ever-improving sequencing methods, genomic data have become ubiquitous in the field of bioinformatics. For the analysis of gene function and regulation in these data, an overall gene structure is to be determined. This process highly depends on state-of-the-art machine learning approaches to distinguish different regions. In this context, an important task is translation initiation site (TIS) prediction.

A TIS indicates where the translation of genes into proteins starts. In addition to TISs, there are also stop codons (TAA, TAG, or TGA), acting as an endpoint for translation. Other important structures in the translation process are donor splice sites (boundaries between exons and introns) and acceptor splice sites (boundaries between introns and exons). In this paper, we will focus on TIS prediction.

TIS prediction has previously been conducted in a variety of ways. Early methods were mainly based on the consensus sequence (Kozak, 1987) and the scanning model (Kozak, 1989). Saeys *et al.* (2007) show that simple models based on these characteristics can already be highly effective. With machine learning techniques becoming more popular, approaches leveraging tools such as Support Vector Machines (SVMs) (Liu *et al.*, 2005; Chen *et al.*, 2014) and artificial

neural networks (Pedersen and Nielsen, 1997) have been used effectively.

Recently, the usage of deep convolutional neural networks (CNNs) has become popular for several pattern recognition tasks. Already achieving state-of-the-art results in the domains of image processing (Krizhevsky *et al.*, 2012) and natural language understanding (Kim, 2014), they are also being applied more and more frequently to genomic data (Alipanahi *et al.*, 2015; Kelley *et al.*, 2016; Zeng *et al.*, 2016). A main concern when using CNNs, however, is their black box nature, as the link between input and output may not be evident. Multiple studies have already been conducted on the visualization of the internals of CNNs (Zeiler and Fergus, 2014; Shrikumar *et al.*, 2016; Fong and Vedaldi, 2017).

In this paper, we introduce the usage of CNNs for effective TIS prediction, whilst also investigating the interpretability of these networks.

## II. METHODOLOGY

### A. Network architecture

In our approach, we define a CNN-based architecture for TIS prediction, where CNNs are a specialized form of artificial neural networks (ANNs). The latter are able to learn hierarchical feature representations by utilizing multiple connected layers of neurons. Each neuron is responsible for the calculation and propagation of an output signal, using different input signals and corresponding weights. A typical characteristic of CNNs is the presence of convolutional layers, which use a sliding window to detect certain patterns in various positions. In contrast to machine learning approaches that require manual feature definition by human experts, convolutional layers allow for automatic feature extraction from raw data. In our research, DNA sequences are fed to the network using a one-hot vector encoding (i.e., A is represented as [1 0 0 0], C as [0 1 0 0], and so on).

Our architecture is shown in Fig. 1. Concretely, it consists of three convolutional layers containing 70, 100, and 150 filters, respectively, of size  $7 \times 4$ ,  $3 \times 1$ , and  $3 \times 1$ , respectively. Each of these layers is followed by a max-pooling layer (size  $3 \times 1$  or  $4 \times 1$ , depending on the input size) and a dropout layer ( $p = 0.2$ ). The output of this last dropout layer is then connected to a fully-connected layer with 512 neurons and another dropout

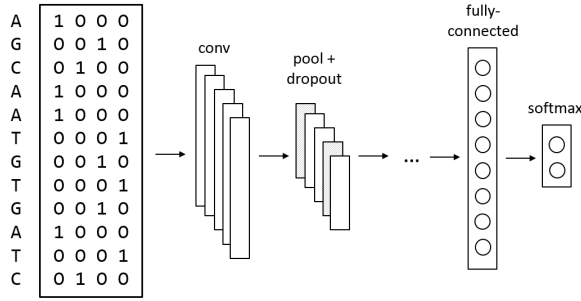


Fig. 1: Our network architecture. After converting the sequence in input to a one-hot encoding, we append a number of convolutional, pooling and dropout layers, followed by a fully-connected layer and a softmax layer.

layer ( $p = 0.2$ ). Finally, a softmax layer is added, generating a probability that leads to a positive or negative classification of a candidate TIS. In addition, every convolutional and fully-connected layer is succeeded by a rectified linear unit (ReLU) for non-linearization.

Given the aforementioned architecture, we train a predictive model by means of a training set. The weights of the model are initialized using a Glorot uniform distribution, while biases are initialized at zero. The training procedure minimizes the categorical cross-entropy cost function over the training set, using stochastic gradient descent with nesterov momentum. At the start of training, the learning rate is 0.05, whereafter it is halved every ten epochs, for a total of 50 epochs. After training, the validation set is used to select the predictive model that performed best during training. Finally, we evaluate the selected model against a testing set.

### B. Visualization

After evaluation, we want to interpret the choices made by our predictive model. To that end, we utilize the state-of-the-art DeepLIFT algorithm, as described in Shrikumar *et al.* (2016). This algorithm generates contribution scores for each nucleotide in the input sequence, indicating how important they are in determining the final prediction. Using these contribution scores, we can then investigate which nucleotide patterns the network is most sensitive to.

Before determining the nucleotide patterns that trigger a strong network response, we adjust the contribution scores of every individual input sequence to act on the same scale. To that end, we use the following formula to acquire weighted contribution scores ( $wcs$ ), which indicate the percentage of

how much each individual nucleotide contributes to the prediction, as compared to the other nucleotides:

$$wcs_i = \frac{cs_i}{\sum_{j=1}^n |cs_j|}, \quad (1)$$

with  $wcs_i$  denoting the weighted contribution score at position  $i$ , with  $cs_i$  denoting the contribution score for the nucleotide at position  $i$ , and with  $n$  denoting the sequence length.

Next, using the weighted contribution scores, we identify which patterns are important in the TIS prediction conducted by the network. The visualization of these results can be found in Section III.

### C. Data and metrics

To evaluate our approach, we compare our predictive model with two different state-of-the-art techniques. The datasets used are detailed in Table I, accompanied by the metrics used. The sensitivity ( $Se$ ), specificity ( $Sp$ ), false positive rate ( $FPR$ ), accuracy ( $Acc$ ), and Mathew's correlation coefficient ( $MCC$ ) are defined as follows:

$$Se = \frac{TP}{TP + FN} \quad Sp = \frac{TN}{TN + FP} \quad FPR = \frac{FP}{TN + FP}$$

$$Acc = \frac{TN + TP}{TN + TP + FP + FN}$$

$$MCC = \frac{TP * TN - FP * FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}$$

where  $TP$  denotes the number of true positives,  $TN$  the number of true negatives,  $FP$  the number of false positives, and  $FN$  the number of false negatives.

We will also make use of the notation  $FPR.80$ , which denotes the  $FPR$  for a fixed  $Se$  of 0.80.

## III. RESULTS AND DISCUSSION

### A. Comparison with other approaches

In Saeys *et al.* (2007), the authors use a combination of three simple TIS prediction mechanisms: the position weight matrix, stop codon frequencies, and an interpolated context model to investigate the composition of the regions around the investigated TIS. Evaluation is conducted on human chromosome 21, extracted from NCBI build 36 version 1, after training on the CCDS (consensus CDS) dataset (excluding chromosome 21).

Authors	dataset	# pos	# neg	pos:neg	seq length	metrics
Saeys <i>et al.</i> (2007)	CCDS (train)	13 917	350 578	1 : 25.2	203	$FPR.80$
	NCBI for chr. 21 (test)	258	1 267 443	1 : 4912.6		
Chen <i>et al.</i> (2014)	TISdb	1159	1159	1 : 1	399	Se, Sp, Acc, MCC

TABLE I: Characteristics of the datasets used in our experiments.

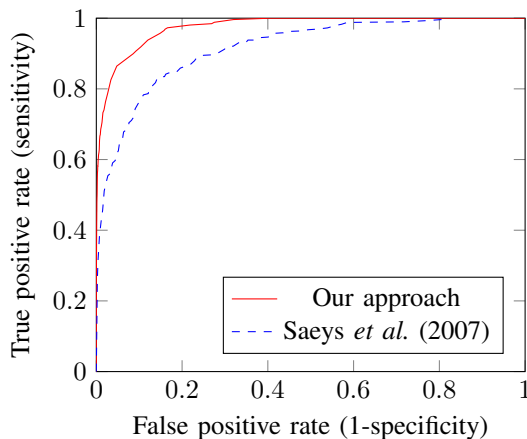


Fig. 2: ROC curves, plotting the relation between true and false positive rates, when classifying using different decision thresholds.

approach	<i>FPR.80</i>
Saeys <i>et al.</i> (2007)	0.125
Our approach	<b>0.031</b>

TABLE II: Results obtained for the NCBI chromosome 21 dataset, after training on the CCDS dataset.

Table II shows the results obtained. Our model was able to classify the candidate TISs more effectively, decreasing the *FPR.80* from 0.125 to 0.031, resulting in a relative decrease of 75.2%. This performance increase is also shown in Figure 2, where we show the ROC curves for both models.

This result encourages the usage of CNNs for DNA analysis. Convolutional layers are very suited for the recognition of different patterns. When searching for decisive patterns, the first convolutional layer gives every nucleotide an individual weight within that pattern, yielding an activation score at every position. By reducing the dimensionality of the activations as we propagate through the network, those activations then get combined to achieve a high-level evaluation of the input.

Additionally, we compare our approach to the approach used in Chen *et al.* (2014), which uses SVMs in combination with manually defined feature vectors, taking into account nucleotide frequencies whilst retaining considerable sequence order information, but also leveraging physicochemical properties of codons. In this case, evaluation is conducted on a much smaller dataset (see Table I), by making use of leave-one-out cross-validation.

approach	<i>Se</i>	<i>Sp</i>	<i>Acc</i>	<i>MCC</i>
Saeys <i>et al.</i> (2007)	0.9532	0.9643	0.9602	0.921
Chen <i>et al.</i> (2014)	0.9749	<b>0.9842</b>	0.9792	0.958
Our approach	<b>0.9905</b>	0.9783	<b>0.9843</b>	<b>0.969</b>

TABLE III: Results obtained for the cross validation tests executed on the TISdb dataset.

Results are listed in Table III. It shows that our approach, which does not need manually defined features, is more effective. Whereas Chen *et al.* (2014) achieves an accuracy of 0.9792, our approach reaches an accuracy of 0.9838, leading to a relative error rate reduction of 24.5%.

### B. Visualization

We visualize the model acquired for our experiments using the CCDS dataset. First, we calculate the *wcs* (as described in Section II) for every input sequence in the NCBI chromosome 21 dataset. Using these scores, we can then calculate the averages on each position for each nucleotide or pattern. That way, we are able to verify four hypotheses:

- 1) **In the proximity of a candidate TIS, the network looks for a pattern that resembles the well-known Kozak consensus sequence (Kozak, 1987).**

In Fig. 3a, we show the average *wcs* per nucleotide on each position in the proximity of the splice site. This is visualized by having higher bars for higher scores. As a comparison, we also show the aforementioned consensus sequence in Fig. 3b, which is based on the frequencies at each position (a higher frequency results in a bigger letter), over a known set of TISs (Cavener, 1987). When comparing both visualizations, we can observe that, for each of the nine positions in the region  $[-6, -1] + [3, 5]$ , the nucleotide with the highest frequency is the nucleotide that gets the highest average *wcs*.

- 2) **The network learns that genes are composed of nucleotide triplets (codons). In addition, the network is highly sensitive to the three stop codons TAA, TAG, and TGA. If a stop codon occurs behind the TIS, the chances of a false classification increase, as this would imply an unlikely short gene.**

Fig. 4 shows the average *wcs* per position for the stop codon pattern TAA. The spikes in the graph occur every three positions. This behavior can be attributed to the translation mechanism, where RNA is translated in codons. The network only steers towards a negative classification when the TAA pattern would indeed get translated into a protein, not if the pattern is split over two codons.

In Fig. 5, we present the average scores per stop codon, as well as the averages for the ATG pattern and the averages over all patterns (NNN). Clearly, the presence of any stop codon behind the TIS that would get translated, has a negative impact on the prediction outcome.

- 3) **The network is sensitive to the occurrence of another ATG pattern. This is justified by the Scanning Model (Kozak, 1989), which implies that the 40S ribosomal subunit, responsible for the initiation of translation, generally stops at the first AUG (ATG in DNA) it encounters after binding to the 5'-end of mRNA.**

As seen in Fig. 5, the scoring for occurrences of the pattern ATG shows a similar behavior, although inversed: the network

is negatively biased regarding ATG codons in front of the TIS, rather than behind it, which confirms that the Scanning Model is implicitly learnt by the network.

- 4) **The network is sensitive to donor splice sites. When it notices a typical donor splice site pattern behind the TIS, the chance of a positive classification increases, as this suggests the presence of an exon.**

Fig. 6 shows the average  $w_{CS}$  for each position of the pattern **MAGGTAAGT** (with  $M = A$  or  $M = C$ ), which is the typical consensus sequence for donor splice sites (Iwata and Gotoh, 2011). The  $w_{CS}$  values in front of the splice site vary between  $-6$  and  $-14$ , and behind the splice site between  $10$  and  $20$ . Consequently, we can conclude that the network learns that a donor splice site pattern should generally only occur behind the splice site.

#### IV. CONCLUSIONS AND FUTURE WORK

In this paper, we successfully applied convolutional neural networks to the problem of translation initiation site prediction. When comparing to the state-of-the-art, we were able to reduce the false positive rate for a sensitivity of  $0.80$  by  $75.2\%$  in a first experiment, and the classification error rate by  $24.5\%$  in a second experiment, with both experiments conducted on different datasets and compared to different approaches.

Furthermore, we were able to provide insight into the decisions taken by our network. By calculating the contribution scores of the individual nucleotides towards the final prediction, and by subsequently looking for reoccurring structures in these scores, we were able to verify four hypotheses. In particular, we found that our network learns the typical pattern for a TIS, that it is sensitive towards stop codons behind and other start codons in front of the TIS, and that it is negatively biased towards donor site patterns preceding the TIS and positively biased towards those succeeding it.

We believe the aforementioned findings are of high interest, as they prove that our network implicitly learns biologically relevant features, without having any prior knowledge about the problem. All these features are automatically learnt from scratch, based on a dataset consisting of raw DNA sequences with corresponding labels. In the future, leveraging the kind of visualizations presented in this paper could open the door for the discovery of new biological characteristics in various use-cases.

#### FUNDING

The research activities described in this paper were funded by Ghent University Global Campus, Ghent University, imec, Flanders Innovation & Entrepreneurship (VLAIO), the Fund for Scientific Research-Flanders (FWO-Flanders), and the EU. Yvan Saeys is a Marylou Ingram Scholar.

#### REFERENCES

Alipanahi, B., Delong, A., Weirauch, M. T., and Frey, B. J. (2015). Predicting the sequence specificities of DNA- and

RNA-binding proteins by deep learning. *Nature Biotechnology*, **33**(8), 831–838.

Cavener, D. R. (1987). Comparison of the consensus sequence flanking translational start sites in *Drosophila* and vertebrates. *Nucleic Acids Research*, **15**(4), 1353–1361.

Chen, W., Feng, P. M., Deng, E. Z., Lin, H., and Chou, K. C. (2014). iTIS-PseTNC: A sequence-based predictor for identifying translation initiation site in human genes using pseudo trinucleotide composition. *Analytical Biochemistry*, **462**(August), 76–83.

Fong, R. and Vedaldi, A. (2017). Interpretable Explanations of Black Boxes by Meaningful Perturbation. *arXiv preprint arXiv:1704.03296*.

Iwata, H. and Gotoh, O. (2011). Comparative analysis of information contents relevant to recognition of introns in many species. *BMC Genomics*, **12**(1), 45.

Kelley, D. R., Snoek, J., and Rinn, J. L. (2016). Basset: Learning the regulatory code of the accessible genome with deep convolutional neural networks. *Genome Research*, **26**(7), 990–999.

Kim, Y. (2014). Convolutional neural networks for sentence classification. *Proceedings of EMNLP*, pages 1476–1751.

Kozak, M. (1987). An analysis of 5'-noncoding sequences from 699 vertebrate messenger rNAS. *Nucleic Acids Research*, **15**(20), 8125–8148.

Kozak, M. (1989). The Scanning Model for Translation: An Update. *J. Cell Biol.*, **108**(February), 229–241.

Krizhevsky, A., Sutskever, I., and Geoffrey E., H. (2012). ImageNet Classification with Deep Convolutional Neural Networks. *Advances in Neural Information Processing Systems 25 (NIPS2012)*, pages 1–9.

Liu, H., Han, H., Li, J., and Wong, L. (2005). DNAFSMiner: A web-based software toolbox to recognize two types of functional sites in DNA sequences. *Bioinformatics*, **21**(5), 671–673.

Pedersen, A. G. and Nielsen, H. (1997). Neural network prediction of translation initiation sites in eukaryotes: perspectives for EST and genome analysis. *Proceedings / International Conference on Intelligent Systems for Molecular Biology*, **5**, 226–233.

Saeys, Y., Abeel, T., Degroove, S., and Van de Peer, Y. (2007). Translation initiation site prediction on a genomic scale: Beauty in simplicity. *Bioinformatics*, **23**(13), 418–423.

Shrikumar, A., Greenside, P., Shcherbina, A., and Kundaje, A. (2016). Not just a black box: Learning important features through propagating activation differences. *arXiv preprint arXiv:1605.01713*.

Zeiler, M. D. and Fergus, R. (2014). Visualizing and Understanding Convolutional Networks arXiv:1311.2901v3 [cs.CV] 28 Nov 2013. *European conference on computer vision (ECCV)*, **8689**, 818–833.

Zeng, H., Edwards, M. D., Liu, G., and Gifford, D. K. (2016). Convolutional neural network architectures for predicting DNA-protein binding. *Bioinformatics*, **32**(12), i121–i127.

