DEEP MOTIF: VISUALIZING GENOMIC SEQUENCE CLASSIFICATIONS

Jack Lanchantin, Ritambhara Singh, Zeming Lin, & Yanjun Qi
University of Virginia, Department of Computer Science
{jjl5sw,rs3zz,zl4ry,y2qh}@virginia.edu

ABSTRACT

This paper applies a deep convolutional/highway MLP framework to classify genomic sequences on the transcription factor binding site task. To make the model understandable, we propose an optimization driven strategy to extract "motifs", or symbolic patterns which visualize the positive class learned by the network. We show that our system, Deep Motif (DeMo), extracts motifs that are similar to, and in some cases outperform the current well known motifs. In addition, we find that a deeper model consisting of multiple convolutional and highway layers can outperform a single convolutional and fully connected layer in the previous state-of-the-art.[1]

1 INTRODUCTION

Understanding genetic sequences is one of the fundamental tasks of health advancements due to the high correlation of genes with diseases and drugs. An important problem within genetic sequence understanding is related to transcription factors (TFs), which are regulatory proteins that bind to DNA. Each different TF binds to specific transcription factor binding sites (TFBSs) on the DNA sequence to regulate cell machinery. We focus on the task of accurately classifying and understanding the DNA subsequences that TFs bind to, which will allow us to better understand the underlying biological processes and potentially influence biomedical studies of human health. [2]

Chromatin immunoprecipitation (ChIP-seq) technologies and databases such as ENCODE (Consortium et al., 2012) have made binding site sequences available for hundreds of different TFs. Despite these advancements, there are two major drawbacks: (1) ChIP-seq experiments are slow and expensive, (2) although ChIP-seq experiments can find the binding site locations, they cannot find patterns that are common across all of the positive binding sites which can give insight as to why TFs bind to those locations. Thus, there is a need for large scale computational methods that can not only make accurate binding site classifications, but also produce clear patterns that represent the positive binding sites.

In order to computationally predict the binding sites, researchers initially used subset frequency counts (Stormo, 2000). Such generative frequency based searching techniques may, however, fail to generalize to unseen examples (Setty & Leslie, 2015). Discriminative techniques such as SVMs have shown to outperform the generative methods by using k-mer features (Ghandi et al., 2014; Setty & Leslie, 2015), but the string kernel based algorithms are limited by the computational complexity of the number of training and testing sequences.

Most recently, DeepBind (Alipanahi et al., 2015) has shown state-of-the-art results on the TFBS classification task by using a neural network based approach. A neural network model is particularly well suitable for the TFBS task considering that it is scalable to a large number of genomic sequences. Although DeepBind achieves better accuracy than previous methods, they use a shallow model with only one convolutional and one fully connected layer. It has been widely shown that the deeper, or multiple layer models outperform shallow models (Szegedy et al., 2015; Srivastava et al., 2015). For the TFBS task, there is a need to model long range dependencies. Therefore, we introduce a deeper model which is able to detect higher level features from the raw nucleotide sequences and make more accurate binding site classifications.

[1] An earlier version of this work was presented at the ICLR 2016 Workshops (Lanchantin et al., 2016). This paper shows the same methods, with a slight improvement on TF prediction and motif generation by tuning different models for each TF.

[2] This task classifies whether or not there is a binding site for a particular TF of interest when given an input DNA sequence.

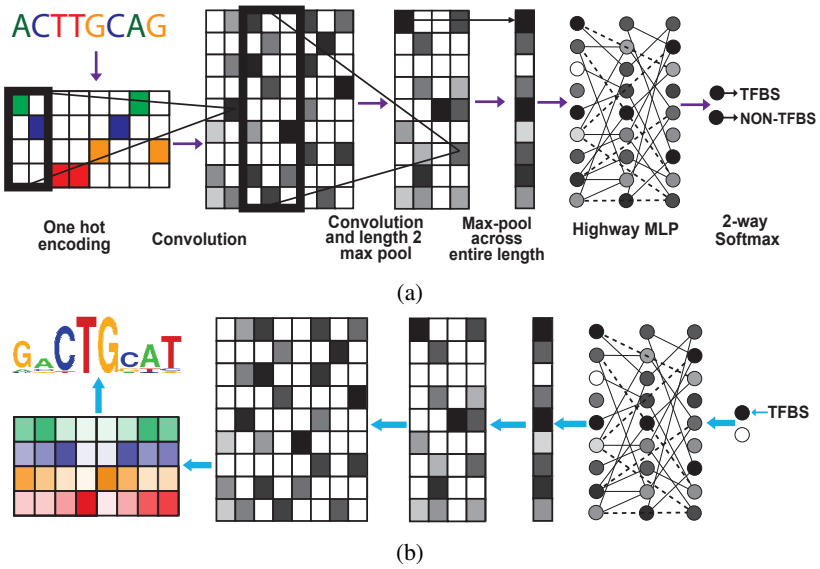

Figure 1: **(a)** DeMo model overview for TFBS classification. Shown with 2 convolutional layers and 2 Highway MLP layers. Our final model has 3 convolutional layers with 128 filters of length-5 at each layer, and 5 fully connected highway MLP layers with 32 nodes at each layer. **(b)** Method for motif generation via class optimization. We find the input matrix which corresponds to the highest locally optimum TFBS probability via backpropagation, and generate a PWM from the matrix.

As with many biomedical tasks, obtaining high accuracy results is not of sole importance. There is a need to find interpretable visualizations which help understand the biological process of interest. For TFBS classifications, this is typically done by finding "motifs", or consensus sequences which define the positive binding sites for a particular TF. Motifs are represented by position weight matrices (PWMs) corresponding to the probability of each character occurring at a specific position (see Fig. 2) (Stormo, 2013). DeepBind finds motifs by mapping the strongest activations of each feature map in the convolutional layer back to the input space for each positive example in their test set, and then counts the nucleotide frequencies of all subsequences to compute a PWM. However, this method is dependent on the specific testing sequences used, and does not represent positive TFBS patterns in general. We present a method (section 2), which finds motifs that depict the notion of a positive TFBS *class* learned by our model, and is not specific to any particular sequence. We argue that this method is more applicable for the biomedical task where it is important to get a general understanding of what a positive TFBS site looks like rather than the strong subsequences of specific positive samples.

The two major contributions of our Deep Motif (DeMo) model are: (1) we are able to achieve state-of-the-art TFBS classification accuracies by using a deep convolutional/highway MLP network, (2) we show that we can extract visual representations of positive binding sites from our model.

2 Network Details and Motif Extraction

Since ChIP-seq experiments output the binding sites in the format of sequences of nucleotide base pairs (i.e. strings with characters A,T,C,G), we can use similar sequence learning models to those used in NLP classification tasks such as sentiment analysis. We introduce the DeMo model (fig. 1a) which uses multiple convolutional layers and a highway multi-layer perceptron (MLP) to make binary classifications. Our experimental results prove that DeMo outperforms the previous state-of-the-art model on the TFBS classification task.

2.1 Deeper Model for TFBS Classification

We use the raw nucleotide characters as inputs to our network, which are encoded into a one-hot encoding. The encoded input then gets fed through several convolutional layers containing convolutions of 128 feature maps and rectified linear units (ReLUs). Certain convolutional layers contain a length 2 max-pooling. All of our filter sizes are length 5, which is much shorter than the 24 length filters of the one convolutional layer in DeepBind (Alipanahi et al., 2015). However, we note that since we use a

Hyperparameter	Values
# Convolutional layers	{3,4}
# Convolutional hidden units	{128}
Max-pooling at each convolutional layer	{2,1}
# Highway MLP layers	{5,7}
# MLP hidden units	{32}

Table 1: Model hyperparameters. Tuned and selected for each TF based on the training set AUC scores.

length 2 max-pooling in each of the convolutional layers, the final convolution actually "sees" a large subsequence of characters from the input sequence, so it is simply a deeper representation of their one layer of filters. The output of the convolutional layers are then max-pooled across the temporal domain resulting in a 128-dimensional vector. We use dropout (Srivastava et al., 2014)for regularization in the convolutional layers.

Traditionally, following the convolutional layers are fully connected MLP layers. Recently, a new technique called highway networks (Srivastava et al., 2015) have proven effective for deeper representations. Highway networks use gating units which learn to regulate the flow of information through a network. Kim et al. (2015) showed that a highway MLP was more effective than a standard MLP when used after a series of convolutions, hypothesizing that highway networks are especially well-suited to work with convolutional layers due to their ability to adaptively combine local features. We use a fully connected highway network after the max-pooled output of the convolutional layers. The output of the highway MLP is fed to a 2-way softmax function.

In our experiments, we train a different model for each TF dataset, and we vary the hyperparameters for each model. Table 1 shows the hyperparameters which were are tuned and selected on the training set for each TF dataset. We found that the TF datasets with fewer training samples had better AUC scores for the smaller (fewer layer) models.

2.2 Class Visualization for Motif Generation

Upon training completion, we propose a strategy to extract class specific visualizations, providing an easy interpretation of what the model has learned (fig. 1b). Similar to the methods used in Simonyan et al. (2013) and yosinski2015understanding, we seek to optimize the following equation where $P_+(S)$ is the probability of the input sequence S (matrix of $input\,length \times 4$, where 4 is our alphabet size) being a positive TFBS computed by the softmax output of our trained model for a specific TF:

$$\arg\max_S P_+(S) + \lambda \|S\|_2^2 \tag{1}$$

where λ is the regularization parameter. We find a locally optimal S through backpropagation, where the optimization is with respect to the input sequence and the model weights remain unchanged. Each element of the input matrix S is uniformly initialized to 0.25, and then S is optimized using (1). We clip the optimized values to the interval $[0, 1]$ and convert S into a PWM, using Laplace smoothing. Although we are generating a dense matrix S when the model was trained on a one-hot encoded input matrix, the experiments show promising results of motif generation.

2.3 Connecting to Previous Studies

Simonyan et al. (2013) and Yosinski et al. (2015) showed that visualisations of a certain class can be obtained from a ConvNet by optimizing the input, where the samples are images rather than sequences. Vidovic et al. (2015) showed that it is possible to extract the underlying "motif" from a discriminative model, but do it on kernel machines. Lastly, Zhang et al. (2015) show that deep character-level ConvNets can outperform other models for sequence classification. However, they do not do any type of visual analysis to understand why it works well. DeMo connects all three of these works into a single model which can make high accuracy predictions on biomedical sequences, and also produce a motif which represents a positive binding site class.

3 Experiments and Results

In order to prove the effectiveness of a deeper model on the TFBS task, we ran DeMo on the same 108 leukemia cell TF datasets used in Alipanahi et al. (2015). Each TF dataset has an average of 30,819 training sequences, and each sequence consists of 101 DNA-base characters (A,C,G,T). Due to the separate train/test data for each TF, we train a separate model for each individual TF dataset.

For the TFBS classification task, our model outperforms DeepBind's by achieving a higher AUC for 92 out of the 108 TF datasets. A comparison is shown in figure 2. In addition, our model achieves a median AUC of 0.951 whereas DeepBind's is 0.931.

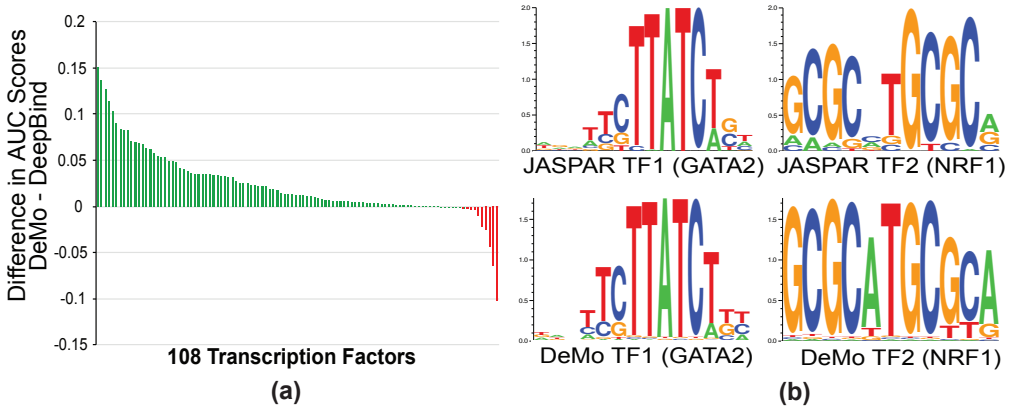

Figure 2: **(a)** DeMo AUC - DeepBind AUC for each of the 108 TF datasets. DeMo outperforms DeepBind on 92 of the 108 datasets. **(b)** Comparison of DeMo motifs vs JASPAR motifs for 2 different TFs. Motifs are shown using information content in bits.

To evaluate our motifs, we performed two comparison strategies against JASPAR motifs (Mathelier et al., 2015), which are widely known within the biological community to be the "gold standard" representations of positive binding sites for hundreds of TFs. We were limited to a comparison of 57 out of our 108 TF datasets by the TFs which JASPAR has motifs for.

For our first strategy, in order to compare the similarity of our motifs, we use a tool called Tomtom (Gupta et al., 2007; Bailey et al., 2009), which compares a specific motif against JASPAR motifs and returns significant matches using their defined statistical measure of motif-motif similarity. Out of the 57 tested, we find that 36 of our motifs (using the windowing approach) significantly match JASPAR motifs (q-value < 0.5). A comparison of motifs can be seen in figure 2.

For our second strategy, we compare how well our motifs score on the positive TFBS test sequences against JASPAR motifs using the Average Motif Affinity (AMA) tool (Buske et al., 2010; Bailey et al., 2009), which scores a set of sequences given a motif, treating each position in the sequence as a possible binding site. Although our method can generate motifs up to length 101 (size of our input sequences), JASPAR motifs are much shorter. In order to handle this issue, we split our motif into all possible windows which are the same size as the JASPAR motif. We then rank each window by average information content, and select the most informative motif to compare against JASPAR. We run the AMA tool on all positive test sequences for each TF, and compare the scoring of our motif vs the JASPAR motif. We find that our motifs are able to outscore ($> 50\%$ of test sequences) JASPAR motifs on 29 out of the 57 motifs. It is important to note that although the JASPAR motifs have been carefully generated using an ensemble approach with much larger TFBS datasets compared to ours, they are not guaranteed to be accurate representations of the positive binding sites.

4 Conclusion

We present Deep Motif (DeMo), a convolutional/highway MLP network which outperforms the state-of-the-art baseline for 92 different TFBS datasets, as well as generate motifs, or interpretable patterns that represent the important transcription factor binding patterns. Although our experiments are on genomic sequence classifcation, DeMo is a generic model for visualizing sequence classifcation tasks. We believe our model is applicable to other sequence classification tasks which demand a visual interpretation of the classes.

REFERENCES

Babak Alipanahi, Andrew Delong, Matthew T Weirauch, and Brendan J Frey. Predicting the sequence specificities of dna-and rna-binding proteins by deep learning. *Nature biotechnology*, 2015.

Timothy L Bailey, Mikael Boden, Fabian A Buske, Martin Frith, Charles E Grant, Luca Clementi, Jingyuan Ren, Wilfred W Li, and William S Noble. Meme suite: tools for motif discovery and searching. *Nucleic acids research*, pp. gkp335, 2009.

Fabian A Buske, Mikael Bodén, Denis C Bauer, and Timothy L Bailey. Assigning roles to dna regulatory motifs using comparative genomics. *Bioinformatics*, 26(7):860–866, 2010.

ENCODE Project Consortium et al. An integrated encyclopedia of dna elements in the human genome. *Nature*, 489(7414):57–74, 2012.

Mahmoud Ghandi, Dongwon Lee, Morteza Mohammad-Noori, and Michael A Beer. Enhanced regulatory sequence prediction using gapped k-mer features. 2014.

Shobhit Gupta, John A Stamatoyannopoulos, Timothy L Bailey, and William S Noble. Quantifying similarity between motifs. *Genome biology*, 8(2):R24, 2007.

Yoon Kim, Yacine Jernite, David Sontag, and Alexander M Rush. Character-aware neural language models. *arXiv preprint arXiv:1508.06615*, 2015.

Jack Lanchantin, Ritambhara Singh, Zeming Lin, and Yanjun Qi. Deep motif: Visualizing genomic sequence classifications. *ICLR Workshops*, 2016.

Anthony Mathelier, Oriol Fornes, David J Arenillas, Chih-yu Chen, Grégoire Denay, Jessica Lee, Wenqiang Shi, Casper Shyr, Ge Tan, Rebecca Worsley-Hunt, et al. Jaspar 2016: a major expansion and update of the open-access database of transcription factor binding profiles. *Nucleic acids research*, pp. gkv1176, 2015.

Manu Setty and Christina S Leslie. Seqgl identifies context-dependent binding signals in genome-wide regulatory element maps. 2015.

Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958, 2014.

Rupesh K Srivastava, Klaus Greff, and Jürgen Schmidhuber. Training very deep networks. In *Advances in Neural Information Processing Systems*, pp. 2368–2376, 2015.

Gary D Stormo. Dna binding sites: representation and discovery. *Bioinformatics*, 16(1):16–23, 2000.

Gary D Stormo. Modeling the specificity of protein-dna interactions. *Quantitative biology*, 1(2):115–130, 2013.

Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–9, 2015.

Marina M-C Vidovic, Nico Görnitz, Klaus-Robert Müller, Gunnar Rätsch, and Marius Kloft. Opening the black box: Revealing interpretable sequence motifs in kernel-based learning algorithms. In *Machine Learning and Knowledge Discovery in Databases*, pp. 137–153. Springer, 2015.

Jason Yosinski, Jeff Clune, Anh Nguyen, Thomas Fuchs, and Hod Lipson. Understanding neural networks through deep visualization. *arXiv preprint arXiv:1506.06579*, 2015.

Xiang Zhang, Junbo Zhao, and Yann LeCun. Character-level convolutional networks for text classification. In *Advances in Neural Information Processing Systems*, pp. 649–657, 2015.