Deep learning in regulatory genomics: from identification to design

Xuehai Hu¹, Alisdair R Fernie² and Jianbing Yan³,⁴

Genomics and deep learning are a natural match since both are data-driven fields. Regulatory genomics refers to functional noncoding DNA regulating gene expression. In recent years, deep learning applications on regulatory genomics have achieved remarkable advances so much so that it has revolutionized the rules of the game of the computational methods in this field. Here, we review two emerging trends: (i) the modeling of very long input sequence (up to 200 kb), which requires self-matched modularization of model architecture; (ii) on the balance of model predictability and model interpretability because the latter is more able to meet biological demands. Finally, we discuss how to employ these two routes to design synthetic regulatory DNA, as a promising strategy for optimizing crop agronomic properties.

Addresses
¹ College of Informatics, Agricultural Bioinformatics Key Laboratory of Hubei Province, Huazhong Agricultural University, Wuhan 430070, China
² Department of Molecular Physiology, Max-Planck-Institute of Molecular Plant Physiology, 14476 Potsdam-Golm, Germany
³ National Key Laboratory of Crop Genetic Improvement, Huazhong Agricultural University, Wuhan 430070, China
⁴ Hubei Hongshan Laboratory, Wuhan 430070, China

Corresponding authors: Hu, Xuehai (hxuehai@mail.hzau.edu.cn), Yan, Jianbing (jianbing@mail.hzau.edu.cn)

Current Opinion in Biotechnology 2023, 79:102887
This review comes from a themed issue on Plant Biotechnology
Edited by Alisdair Fernie and Jianbing Yan

Available online xxxx
https://doi.org/10.1016/j.copbio.2022.102887
0958–1669/© 2022 Elsevier Ltd. All rights reserved.

Transcriptional factor-binding site (TFBS)
TFBS are a subset of DNA fragments that transcription factors specifically bind to.

cis-regulatory elements (CREs)
CREs are noncoding DNA sequences serving as individual TFBSs to regulate the transcription of their target genes.

cis-regulatory modules (CRMs)
CRMs are assemblies of CREs, including promoters, enhancers, and silencers, which integrate the active transcription factors and the associated cofactors in a time- and place-specific manner to regulate their target genes.

Fully connected neural networks (FCNN)
FCNNs are a type of artificial neural networks that are composed of a series of fully connected layers. Each neuron in one layer receives the values from every neuron in the preceding layer, thus leading to the full connections.

Convolutional neural networks (CNN)
CNN inspired by the receptive field mechanism in biology, are a type of artificial neural networks and are utilized to process image data originally. Neurons in convolutional neural networks only accept the signals in a restricted region of the visual field (i.e., the receptive field), thus leading the local connections compared with FCNN.

Recurrent neural networks (RNN)
RNNs applicable to processing sequential data, are a class of neural networks, where the connections between neurons from one layer form a cycle, enabling outputs from some nodes at previous time steps to affect subsequent input to the same nodes, and therefore they are considered to have a memory ability.

A CNN layer
A CNN layer is composed of a series of filters whose parameters are learned from the training process. The filter used as a feature detector transforms the input data into a feature map, achieved by sliding the filter across the input data and performing a dot product of the filter with the same-sized input data. For DNA sequences, in the first CNN layer, filters could be considered as PWMs, each of which scans across the sequence, and calculate a nonlinear similarity score at each position. A CNN layer usually contains a set of filters to capture different patterns in the data.

* Given the role as Guest Editor, Jianbing Yan and Alisdair Fernie had no involvement in the peer-review of the article and has no access to information regarding its peer review. Full responsibility for the editorial process of this article was delegated to ().
Introduction
Since the turn of the century, genomics is a rising data-driven discipline [1], which aims to elucidate the function of all of the nucleotide sequences using high-throughput technologies such as genome sequencing and transcriptome profiling. Deep learning is a data-driven information technology that has made great successes in the artificial intelligence community, including computer vision and natural-language processing (NLP) [2]. In plant biology, deep learning is starting to be used in a wide range of different fields, including plant breeding [3–7] and fruit taste [8]. Genomics and deep learning, both being data-driven, are a natural match. Indeed, their combined use has already achieved considerable progresses in the fields of regulatory genomics [9–11], gene expression modeling [12–14], and cancer diagnosis [15] in the past decade. As such, we will focus our review on the combination of these approaches.

Regulatory genomics refers to the study of functional noncoding DNA that contributes to the regulation of gene expression. The simplest units of regulatory genomics are transcriptional factor-binding site (TFBS) and cis-regulatory elements (CREs), which are often 5–20-bp DNA fragments recognized by a specific transcriptional factor (TF) protein [7]. In 2015, the pioneering work of DeepBind was the first successful deep learning application in genomics, and amazingly almost completely solved the long-standing problem of TFBS predictions [10]. Based on basic units of TFBS, larger genomic regions assembled by a combination of spaced TFBSs are called cis-regulatory modules, these include both gene-proximal promoters and distal enhancers [16]. Such elements are believed to act as master regulators of target gene expression and are naturally core objects of regulatory genomics [17]. Deep learning usually characterizes promoters and enhancers by modeling their associated epigenomic signals, including chromatin accessibility and histone modifications [11,18].

A considerable number of elegant reviews have comprehensively demonstrated the fundamental network structures of deep learning, including fully connected neural networks (FCNN), convolutional neural networks (CNN), and recurrent neural networks, demonstrating how to apply these modeling approaches to solve regulatory genomics problems [1,4,9,19–21]. For example, the first successful case of DeepBind took short DNA fragments (varying lengths of 14–101 bp) as the inputs and employed their binding intensities as the outputs to learn adjustable parameters of filter matrix in the CNN layer and weight matrix in the FCNN layer [10]. However, two important research trends seem to be emerging in view of new advances in recent years: (i) accurate modeling of more complex input of very long regulatory DNA sequence, which requires more complex model architecture that needs to be assembled from modules or blocks [12,22]; (ii) increased attention for the biological interpretability of the models [23–26]. This would help to define critical nucleotide bases with regulatory effects, which could then meet the specific biological demand and are the ideal target of downstream bioengineering applications such as drug targets in humans [27] and breeding-by-editing in plants [4].

Here, we first review recent innovations of model architecture on deep learning modeling in the field of regulatory genomics, and subsequently summarize existing biological interpretability methods for the identification of CREs. Finally, we discuss how to employ deep learning models and interpretability methods as we move from identification to design of genomic regulatory elements.

New trends of deep learning modeling of regulatory DNA: utilization of longer input sequences via modularization of model architecture
Deep learning modeling on regulatory DNA was initially carried out by using short genomic fragments (100–500 bp) as input of a basic unit of TFBS [10]. It has, by now, been expanded to modeling longer genomic regions (up to 200 kb), which are long enough to include the most determinants of gene expression [12]. During
the transition to using longer sequences, researchers had to adopt model architecture that was appropriately adapting to input lengthy sequences. At the early stage, when the input genomic sequence is of 100–500-bp length, the task of learning and training is to capture the local features with spatially invariant patterns among these short sequences. This is the main advantage of the use of a CNN layer, an architecture that was validated and proven in the field of image classification [2]. Therefore, early deep learning tools, including DeepBind [10] and DeepSEA [11], simply used a layer-based architecture, which in turn consists of a convolutional layer, a rectification layer, a pooling layer (the first three layers usually being referred to as a CNN block), and a fully connected layer (Figure 1a). The design principle behind this is that the convolutional layer can effectively detect local sequence features, whose noise is removed with the rectification layer; the pooling layer only reserves the most prominent features that will be used for classification or prediction with the last fully connected layer.

Later studies focused on larger functional genomics regions ranging from open-chromatin region with 600 bp of Basset [18] to gene expression of 10.5-kb region flanking transcription start sites (TSS) of Xpress [28] and of 3 kb of maize gene expression prediction [3]. To model a kilobase-scale input sequence, the corresponding design started to adopt the modularization idea by stacking two [28] or three [18] CNN blocks before the final fully connected layer. Under this design, the first CNN layer represents the recognition of position weight matrices (PWMs) and subsequent CNN blocks consider the spatial distances and combinations between PWMs recognized in the previous layer [18]. Models trained using kilobase-scale input sequences together with several CNN blocks had stronger predictability than short DNA fragments on more complex regulatory phenomena such as chromatin accessibility [18] and variant expression effects [28].

Understanding gene expression is the core objective of regulatory genomics, thus, accurate and robust prediction of gene expression is the core task of deep learning in regulatory genomics. To this end, researchers need to model longer input sequences, which includes most determinants (not only gene-proximal promoters, but also gene-distal enhancers) affecting gene expression. This is especially relevant for large genomes such as human [18] and maize [13]. Indeed, recent studies extended their input sequence length from kilobase scale to hundred kilobase scale [12–14,22,29] (Table 1).

To model such long sequences, one option is to stack more CNN blocks. Unfortunately, experiences from computer vision suggest that using more than 30 layers will lead to severe gradient-vanish problems. Furthermore, simply stacking CNN blocks does not guarantee an enlarged receptive field of hundred kilobase scale. To avoid the gradient-vanish problem, the ResNet program was proven to be successfully stacking more CNN blocks [29]. The enlarged receptive field problem can be improved by dilated convolution [14]. A successful approach therefore consists of combining ResNet with dilated convolution to create a novel design of ‘dilated residual blocks’, which theoretically can stack many blocks and solve both problems. Let us take Besenji2 [22] as an example to demonstrate its precise design (Figure 1a): to model DNA sequences with each length of 131 072 bp (=2^17), Besenji2 first adopts seven iterated CNN blocks to extract the relevant sequence motifs by reducing feature dimension to 1024 (=2^10) bins (Each bin represents a basic unit of 128-bp window, and each CNN block reduces a half-dimension with the max pooling layer of width of 2); it then applies eleven dilated residual blocks to model long-range interactions between the above 1024 bins. Besenji2 has improved gene expression prediction accuracy in human and mouse, and iterated CNN blocks combined with iterated dilated residual blocks have become a standard model architecture for modeling hundred kilobase-scale input sequences. Recently, the Enformer program [12] replaced the dilated residual block with a transformer block, which mainly uses multiple self-attention layers. With this modification, the authors successfully detect the relative position and coding of different words in NLP [30] as well as to effectively prioritize gene-distal enhancers [12]. For other possible approaches, we refer readers to a rich model resource of the Kipoi repository [31] (https://kipoi.org/), which is a community exchange platform that integrates a total of 2201 trained models for regulatory genomics from 35 research groups.

Model interpretability is the next step
Model interpretability refers to the prioritization of important features among all input features during prediction [26] and it is essential in biological applications of deep learning [32]. Robust model interpretability in regulatory genomics is receiving more attention in recent years because it may help us to identify causal variants affecting TF binding [16] or chromatin accessibility [18], as well as can aid in the identification of gene-distal enhancers [12] or critical CREs affecting enhancer activity [24]. Overall, interpretation strategies of deep learning in genomics can be summarized into three main categories: (i) perturbation-based interpretability, (ii) backpropagation-based interpretability, and (iii) attention mechanism-based interpretability.

Identifying important bases with perturbation-based interpretability
Perturbation-based interpretability quantifies the importance of each base of an input sequence using the
(a) Model Architecture

(b) Model Interpretability

(c) Design regulatory DNA

IPA1 (OsZH11G0821220960.01) rice core promoter region: chr8:25245456-25246525(-)

Figure 1
strategy of making small perturbations in the input sequence, and then monitors the corresponding change of the output using the deep learning model described in the last section. A simple but widely used perturbation method is \textit{in silico} mutagenesis (Figure 1b left panel), which mutates the current nucleotide into the other three nucleotides at each base and graphically demonstrates the significant effects on the output via means of a mutation map [10,12–14,18]. In addition, an emerging perturbation method is \textit{in silico} tiling deletion [24], which employs the deep learning model to simulate the tiling deletion of a given enhancer or a gene promoter. This approach is beginning to become popular in plants, for example, in gene promoters of maize FCP1 gene [33] and rice IPA1 gene [34]. Different from \textit{in silico} mutagenesis, \textit{in silico} tiling deletion (Figure 1b middle panel) removes a small sliding window (10 bp in DeepSTARR) with an overlapping stride (5 bp) from the input sequence (250 bp), and then monitors the significant changes in important bases using a variation curve or an overlapping histogram [24].

\textbf{Identification of critical \textit{cis}-regulatory elements with base importance score via backpropagation-based interpretability}

Backpropagation-based interpretability quantifies the importance of each base by means of a base contribution score (Figure 1b right panel), which represents the contribution this base makes to the difference prediction value. A typical backpropagation-based interpretability method is DeepLIFT [32], which first uses the deep learning model to compute the prediction values of the given input sequence and the reference sequence, respectively, and then decomposes the differences between them into contribution scores of all bases, by back-propagating the contributions of all neurons of the network to every feature of the input. Recent publications all chose “base contribution scores” to highlight base-resolution CREs that are usually visualized as highly colored characters [12,23,24]. A limitation of DeepLIFT is that its current version does not support the ResNet module, thus rendering it inapplicable to dilated residual block designs. An alternative strategy is to use a gradient-based method such as gradient×input or attention weight of transformer [12].

\textbf{Discovering the transcriptional factor motif syntax with transcriptional factor motif analysis}

To investigate the biological implications of several successive bases with high base contribution scores, the new motif discovery algorithm — TF-MoDISco [35] was developed. This tool can identify high-quality, non-redundant TF motifs. It takes base contribution scores as input and identifies DNA segments with substantial contributions (significantly higher than the background distribution of scores). As a result, TF-MoDISco will provide as output the nonredundant and known TF [36]

\begin{table}
\begin{center}
\begin{tabular}{|l|l|l|l|}
\hline
\textbf{Model name} & \textbf{Input sequence length} & \textbf{Model architecture} & \textbf{Model interpretability method} \\
\hline
DeepBind & Varying lengths of 14–101 bp & A CNN block & \textit{in silico} mutagenesis \\
DeepSEA & 1kbp & Three CNN blocks & \textit{in silico} mutagenesis \\
Basset & 600 bp & Six CNN blocks & \textit{in silico} mutagenesis \\
EnPecto & 40kbp & Four CNN blocks + seven dilated CNN blocks & Saliency maps \\
Basenji & 131kbp & Seven CNN blocks + eleven dilated residual blocks & \textit{in silico} mutagenesis \\
Basenji2 & 131kbp & Four CNN blocks & \textit{in silico} mutagenesis \\
Xpresso & 10.5kbp & Two CNN blocks & None \\
ExpResNet & 95kbp & Four residual units & None \\
Enformer & 200kbp & Seven CNN blocks + eleven transformer blocks & Gradient × input or attention weight \\
BPNet & 1kbp & Nine dilated residual blocks & DeepLIFT contribution score \\
DeepSTARR & 250 bp & Four CNN blocks & \textit{in silico} tiling deletion \\
\hline
\end{tabular}
\end{center}
\end{table}
An example of maize FCP1 gene for demonstrating the whole workflow. First, we constructed a training dataset of maize B73, which contains 36,906 maize genes for training and 4,101 maize genes for validation. For each gene, we took the 120-kb genomic sequence around its TSS as the input and adopted the maximal gene expression across 422 maize RNA-seq experiments as the output. And we subsequently employed a modularization of model architecture that successively contains seven CNN blocks and eleven dilated residual blocks for training. After training, we input the genomic sequence of FCP1 gene into the deep learning model and obtained the direct predictions of five events of promoter editing of FCP1. Comparing with the experimental expression levels, the predicted expression levels have a consistent result, implying that the deep learning model can prioritize critical regions, which have potentials to reappear the promoter editing results. Additionally, two model interpretability methods of in silico tiling deletion and DeepLIFT contribution scores on the FCP1 gene promoter both suggest several activating elements (positive contributions) and repressive elements (negative contributions), implying that the deep learning model might provide additional information of critical regulatory elements.
motifs enriched within the whole input sequences. One can thereby easily locate the position of each motif by scanning each input sequence with the above PWMs and standard motif tools such as FIMO [37], and then perform downstream syntax analysis.

Finally, there are some other interpretability tools or modifications on the above-mentioned tools that we would like to discuss. One of these is SHAP [38] (SHapley Additive exPlanations), another interpretability tool, the idea of which is to obtain interpretability by using a simpler explanation model with the additive feature attribution property to approximate the original model. SHAP claims that it outperforms DeepLIFT in some computer vision problems. However, there are no SHAP-based genomic applications to date, probably due to its huge computational burden. Another two technical modifications are worthy of note: (i) the use of exponential activation to first-layer filters, which may lead to interpretable and robust representations of motifs [39]; and (ii) the directed correction of gradient×input, which can lead to small, but significant improvements in gradient-based contribution scores [40].

Designing synthetic cis-regulatory elements: knowledge-guided design and generative adversarial-based design

One goal of future biological studies is to transition from understanding life to transforming life. For regulatory genomics, it would be interesting to design novel and new-to-nature CREs, which would go beyond the sequence space limitation of existing organisms. Below, we list a number of recent advances on the design and engineering of regulatory DNA, which can be broadly classified into two categories: (i) knowledge-guided design and (ii) generative adversarial-based design (Figure 1c).

A typical representative of knowledge-guided design is DeepSTARR, which first built a CNN model for accurate enhancer activity prediction, and subsequently discovered TF motif syntax of combination and spacing (Figure 1c left panel) using the base importance score of DeepLIFT, and finally employed the learned syntax knowledge to design a more optimal enhancer with maximal activity [24]. In another example, Jores et al. [41] created synthetic promoters, from native plant core promoters, whose promoter strengths are comparable or even exceed that of the strong 35S minimal promoter, using the deep learning-based strategy of in silico evolution. More recent design ideas on regulatory DNA, which can be broadly classified into two categories: (i) knowledge-guided design and (ii) generative adversarial-based design (Figure 1c).

Unlike TF motif syntax-based design from learned biological knowledge, generative adversarial nets (GAN)-based design is a more data-driven strategy, which simultaneously constructs two models: a generator and a discriminator. The generator aims to generate sequences as outputs, whose fidelities are then measured by the discriminator model, which is pretrained with natural sequences [43]. The training process of the whole GAN model is the adversarial process between generator and discriminator [19, 44] (Figure 1c right panel). For example, Wang et al. implemented a GAN-based design of 50-bp E. coli promoters, most of which were found to have low sequence similarity with natural sequences and were experimentally validated to be functional [45].

An example of the maize FCP1 gene for demonstrating the whole workflow

Finally, we use an example of the maize FCP1 gene to demonstrate the whole workflow of this review (Figure 2). Maize FCP1 gene acts in the CLAVATA (CLV)-WUSCHEL (WUS) feedback pathway, and mutations of FCP1 might lead to enlarged inflorescence stems and fasciated ears in maize [33]. The details can be found in the figure legend of Figure 2.

Conclusions and future perspectives

Deep learning applications in regulatory genomics have undergone a complete volte face from the early goal of improving model predictability to the current goal of balancing predictability and interpretability. Consequently, there have been remarkable advances in the past few years with this change in perspective, which we feel is completely changing the ‘rules of the game’. Future studies should focus on two aspects: (i) model interpretability should be further improved because this will help us performing more precise identification; (ii) designing synthetic CREs, which is a promising route for future crop improvement. However, it is important to note that this field is in its infancy, and as such, more novel strategies and methods to design synthetic regulatory DNA, capable of implementing any desired gene expression, will be required.

Funding
This work was supported by the National Natural Science Foundation of China (32070689 to Xuehai Hu and U1901201 to Jianbing Yan).

Conflict of interest statement
Nothing declared.

Data availability
Data will be made available on request.
References and recommended reading

Papers of particular interest, published within the period of review, have been highlighted as:

• of special interest
•• of outstanding interest.


DeepBind is a pioneering work that successfully applied the deep learning method in genomics. It was the first to design a proper way to model DNA sequence with CNN, and it was also first created the mutation map via in-silico mutagenesis.


Enformer is a breakthrough software that first introduced the transformer model into the block model of architecture generation of gene expression prediction. It also extended the input sequence to 200-kb length, which makes it the first report to identify gene-distal enhancers via gene expression prediction model.


DeepSTAR is a breakthrough work that comprehensively assessed regulatory genomic elements to DNA-protein and protein-DNA interactions. It furthermore demonstrated how to use base importance scores for systematically studying TF motif syntax and how to design stronger enhancers based on the learned knowledge.


DeepSTARR is a breakthrough work that comprehensively interpreted model interpretability. It furthermore demonstrated how to use base importance scores for systematically studying TF motif syntax and how to design stronger enhancers based on the learned knowledge.


DeepLIFT is a breakthrough study that first assigned contribution scores into each base by decomposing the prediction difference through deep learning network. Many subsequent publications chose to use DeepLIFT contribution score to highlight critical bases affecting gene expression.


enhances rice grain yield by decoupling panicle number and size. Nat Biotechnol 2022, 40:1403-1411.

This paper is highly important in that it used the strategy of promoter tiling deletion to target a 54-base pair cis-regulatory region, deletion of which balances two opposite traits of grains per panicle and tiller number, leading to substantially enhanced grain yield per plant.


This paper is highly important in that it first noted the necessary conditions that a robust design of regulatory DNA should satisfy. It also first introduced fitness into considerations and it reminds us to balance adaptation and complexity
