A review on deep learning for regulatory genomics

Writing Assignment

MSc Bioinformatics and Biocomplexity

Abstract

Advances in deep learning have revolutionized the omics field, including genomics, epigenomics and transcriptomics. Many deep learning models have integrated multiple types of omics data to study genomic regulation and predict different signals of regulatory activity from DNA sequence. These models differ from each other in many aspects, such as the training data, the model architecture, the training approach, or their interpretation method. In this review, we provide a comprehensive overview of the current state of the field of deep learning in regulatory genomics by examining each part of these models. We start by describing the differences in the data used by each model and then explain the most commonly used architectures and the different training approaches these models take. We also provide a concise overview of the different model interpretation methods available with their advantages and disadvantages. Furthermore, three main applications of these models are described: motif discovery, non-coding variant effect and synthetic construct design. Finally, we conclude with a discussion of the limitations of these models nowadays. This survey is intended to serve as a guideline for omics researchers to gain an overview of the current landscape of deep learning methods in genomics and to guide them to focus new efforts on solving the limitations.

Empar Baltasar Pérez

Supervisor: Lucía Barbadilla

Group leader: Jeroen de Ridder

Table of Contents

Introduction

The development of the Sanger technique for sequencing DNA in 1977 opened the doors to the field of genomics[1]. The first DNA-based genome to be sequenced using that technique was the genome of the bacteriophage PhiX174. A few years later, in 1990, the Human Genome Project started with the goal of obtaining the sequence of the entire human genome, with the expectation that it would lead to cures and treatments for human diseases. But when the full sequence was published for the first time in 2003, it became clear that there was still much work to be done to understand the function of the sequence in its entirety. It consisted of 3 billion base pairs (bp), of which only 2% encoded proteins (20,000 – 25,000 genes). The rest of the sequence (98%), misleadingly referred to as "junk DNA", still had unclear functions.

It is now extensively known that non-coding DNA has structural and regulatory purposes[2]. It can regulate gene expression through various mechanisms (reviewed by Zrimec et al. (2021)[3] and Misteli (2020)[4]). For instance, it contains elements like promoters and enhancers that guide transcription initiation. This is achieved through the binding of transcription factors (TF) with activating or repressing effect to specific DNA patterns (also called motifs). Gene expression is also regulated through sequence-guided nucleosome positioning. Nucleosomes are basic structural units of chromatin that are formed by 147 bp-long DNA stretches wrapped around eight histone proteins. They regulate gene expression by competing with TFs for DNA binding. Other epigenetic mechanisms (functional changes in the genome that do not involve a sequence change), such as DNA methylation or histone modification, can also regulate gene expression by affecting DNA accessibility for TFs. In addition, some motifs affect gene expression by controlling the 3D structure of the chromatin. For example, the architectural chromatin protein CTCF binds to specific motifs and creates topologically associated domains (TADs). These chromatin domains restrict interactions between regulatory elements to genes within the domain and contain countless internal loop interactions that bring regulatory elements like enhancers and promoters close in space to facilitate the activation of gene expression.

Mutations in functional elements of the non-coding DNA can therefore result in gene deregulation and lead to diseases. For instance, mutations in two positions in the promoter of *TERT*, encoding the reverse transcriptase subunit of telomerase, introduce *de novo* TF binding sites and cause an increase in activity that avoids shortening of telomeres, driving a tumorigenic mechanism[5]. Some efforts to study the link between non-coding mutations and their consequence have been carried out in **expression quantitative trait loci (eQTL)** studies and **genome-wide association studies (GWAS)** that look for associations between specific single nucleotide point mutations (SNPs) in non-coding DNA and gene expression levels (eQTL studies) or a particular phenotype (GWAS) measured in hundreds or thousands of individuals[6]. However, these studies present some limitations. The first one originates from the phenomenon of linkage disequilibrium: some SNPs always appear simultaneously due to the lack of recombination between them. Because of this, eQTL and GWAS cannot discern which one is the cause of the phenotype under study. Second, they cannot determine how a particular SNP eventually causes the phenotype in question (e.g., through a drop/gain in TF binding affinity or the creation or disruption of a TF binding site). Third, they are limited to the study of SNPs present in the population of individuals that participate in the study, which caps their statistical power to pinpoint genetic signatures underlying rare traits. Furthermore, they are unable to predict the effect of unobserved mutations. This is an important limitation, since many variants are unique or rare: a recent study performed whole-genome sequencing of >53,000 individuals with rich phenotypic data and diverse backgrounds, and found 400 million variants, of which 97% had frequencies of less than 1%, and 46% occurred only in one individual[7].

The progress and feasibility of next-generation sequencing technology allowed the development and extended use of genome-wide methods that determine the state of the genome at multiple levels. For instance, chromatin immune-precipitation sequencing (ChIP-seq) is a technique that determines the binding profile of different DNA-binding proteins, histone modifications or nucleosomes[8]. DNase-seq identifies active or accessible (nucleosome-depleted) regions of the DNA[9]. RNA-seq and Cap Analysis Gene Expression (CAGE) quantify gene expression: RNA-seq can detect different transcript isoforms generated by alternative splicing, while CAGE measures expression only at the transcription start site[10,11]. Massively parallel reporter assays (MPRAs) have been used to screen candidate regulatory sequences for promoter/enhancer activity by placing them in plasmids with a reporter construct that are introduced in thousands or millions of cells from which the activity readout is obtained[12]. All these methods opened the opportunity to study in detail how non-coding DNA regulates gene expression, and the detailed mechanisms by which mutations in these regions could result in gene expression deregulation. However, understanding the complex interactions between different combinations of regulatory

elements and their function is not trivial. The increasing amounts of omics datasets available posed deep learning (DL) as a powerful tool to study the regulatory activity of non-coding DNA.

DL has replaced classical algorithms (linear regression, support vector machines, random forests and feedforward neural networks), commonly referred to as "shallow" methods, to unravel the syntax that dictates genomic regulation[3]. DL is a form of machine learning that learns how to perform tasks as humans do: by example. By providing a model with big amounts of labelled data, it is trained to detect patterns and annotate unseen data. Unlike more shallow machine learning methods, DL does not require manual feature extraction to make predictions, but it automatically detects significant features without the need to make strong biological assumptions. The ability of DL models to extract patterns and their ability to capture non-linear relationships makes them the ideal tool to study the regulatory function of non-coding DNA. They can be used to identify functional units in the DNA (enhancers, promoters, transcription factor binding sites (TFBS), transcription start sites (TSS), histone modification sites, etc.), investigate how they drive gene expression and predict the effect of mutations in these regions.

The potential of DL for regulatory genomics caused a continuous stream of published papers since 2015 describing DL models to predict genome-wide signals of regulatory activity (transcription factor binding profiles, histone modification profiles, chromatin accessibility, enhancer-promoter interactions or RNA expression levels) from DNA sequence alone (Figure 1, Table 1). These models can be used to make predictions from new sequences. This is particularly interesting for many reasons. First, unlike GWAS and eQTL studies, DL models have the ability to exhaustively inspect the effect of all possible mutations in a sequence, even if they have never been observed. Second, by using DNA sequence as the only predictor variable, they facilitate the application in a clinical context, where DNA sequence is a feasible data modality to be acquired from patients. In addition, DL models can be inspected with different interpretation methods to understand how predictions were made, which reveals how DNA sequence could possibly determine the different levels of genomic regulation (from TF binding affinities to gene expression levels). Thus, they are not only interesting for clinical applications, but also for basic molecular biology research.

The multiple models developed so far use different data, architectures, training and validation schemes, and model interpretation approaches. The diversity in these choices imply that there is not one golden standard to design, evaluate and interpret DL models in genomics. This review intends to examine the differences and similarities in their design, while critically assessing their possible limitations. We will also discuss challenges that need to be addressed. We hope that the reader obtains a clear picture of the diversity in the design and validation of DL models that could help them, if needed, in the development of their own algorithm. We also hope this review helps researchers focus on the current limitations and the gaps instead of reinventing models that have already been developed by others previously.

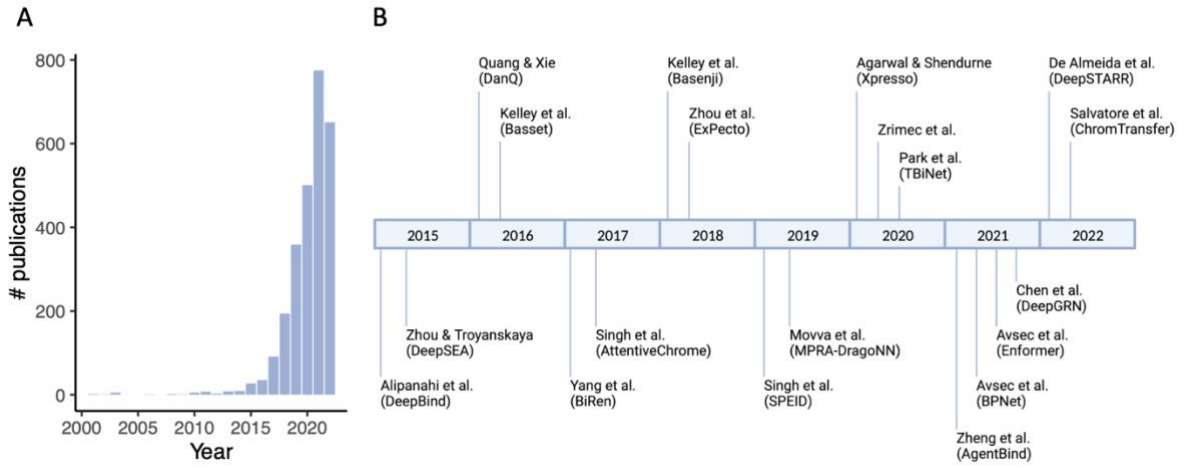

Figure 1. A) Histogram showing the increase over the last years in the number of publications about deep learning in regulatory genomics. B) Timeline of DL models for the study of regulatory genomics reviewed here.

Table 1. Overview of deep learning models for regulatory genomics.

Model	Input DNA length	Prediction	Organism / cell type	Architecture
DeepBind *Alipanahi et al. (2015)*[13]	14–101 bp	TF binding affinity	Mouse & human	CNN
DeepSEA *Zhou & Troyanskaya (2015)*[14]	1 kb	Epigenetic profiles: - 690 TF binding profiles for 160 different TFs - 125 DHS (DNase Hypersensitive Sites) - 104 histone mark profiles	Human	CNN
DanQ *Quang & Xie (2016)*[15]	1 kb	Epigenetic profiles - 690 TF binding profiles for 160 different TFs - 125 DHS (DNase Hypersensitive Sites) - 104 histone mark profiles	Human	CNN + RNN + Bidirectional LSTM (BLSTM)
Basset *Kelley et al. (2016)* [16]	600 bp	Chromatin accessibility	164 human cell types	CNN
BiRen *Yang et al. (2017)*[17]	1 kb	Enhancer ability	Mouse & human	CNN + RNN (GRU)
AttentiveChrome *Singh et al. (2017)*[18]	10 kb	Gene expression	56 human cell types	LSTM + Attention layers
Basenji *Kelley et al. (2018)*[19]	131 kb	Epigenetic profiles (genomic tracks): - 949 Chromatin accessibility profiles - 2307 histone modification profiles - 973 transcription start site (TSS) profiles	Human	CNN + Dilated layers
ExPecto *Zhou et al. (2018)*[20]	40 kb	Gene expression	Human (218 tissues and cell types)	CNN + linear regression
SPEID *Singh et al. (2019)*[21]	3 kb for enhancer 2 kb for promoter	Enhancer-promoter interaction	6 human cell lines (GM12878, HeLa-S3, HUVEC, IMR90, K562, and NHEK)	CNN + RNN
MPRA-DragoNN *Movva et al. (2019)*[22]	145 bp	Enhancer activity	Human K562 and HepG2 cell lines	CNN
Xpresso *Agarwal & Shendurne (2020)*[23]	10.5kb	Gene expression	- Human myelogenous leukemia cells (K562) - Human lymphoblastoid cells (GM12878) - Mouse embryonic stem cells (mESCs)	CNN
Zrimec et al. (2020)[24]	150 bp	Median gene expression	- S. cerevisiae - E. coli - D. melanogaster - M. musculus - H. sapiens - A. thaliana - D. rerio	CNN
TBiNet *Park et al. (2020)*[25]	1 kb	690 TF binding profiles for 160 different TFs	Human	CNN + Attention layers + BLSTM
AgentBind *Zheng et al. (2021)*[26]	1 kb	TF binding profiles of 38 TFs	Human lymphoblastoid	CNN or CNN + RNN
BPNet *Avsec et al. (2021)*[27]	1 kb	TF-binding: -Profile shape: probability of observing a particular read at a particular position in the input sequence -Total read count	Mouse embryonic stem cell (ESC)	CNN + Dilated layers
Enformer *Avsec et al. (2021)*[28]	200 kb	Genomic tracks: - 600 human RNA expression profiles - 4713 human auxiliary measurements (TF binding, DNA accessibility) - 357 mouse RNA expression profiles - 1286 mouse auxiliary measurements (TF binding, DNA accessibility)	Mouse & human	CNN + Attention layers
DeepGRN *Chen et al. (2021)*[29]	1 kb	TF binding profiles	Different human cell types (from the 2016 ENCODE-DREAM *in vivo* Transcription Factor Binding Site Prediction Challenge)	CNN + BLSTM + Attention layers
DeepSTARR *De Almeida et al. (2022)*[30]	249 bp	Enhancer activity in combination with a developmental (Dev) or a housekeeping (Hk) promoter	Drosophila melanogaster S2 cells	CNN
ChromTransfer *Salvatore et al. (2022)*[31]	600 bp	Chromatin accessibility	Pre-training with many human cell types (n not specified) Fine-tuning with 6 human cell lines: GM12878, K562, HCT116, A549, HepG2 & MCF7	CNN

Training data

Epigenomic data

Models like MPRA-DragoNN[22] and DeepSTARR[30] use data from reporter assays to predict the enhancer or promoter activity of DNA sequences. These assays enable the evaluation of the effect of isolated SNPs[32], solving the problem of linkage disequilibrium. However, they present some limitations: 1) These experiments often have low reproducibility, which caps the performance of models trained on the data obtained from them because a model can never be more accurate than the training data[22]. 2) Due to the location of putative sequences in plasmids, these assays cannot account for the effect of distal enhancers, local chromatin context and three-dimensional conformation of the genome. Thus, they only account for the regulatory activities that are intrinsic to the DNA sequence[33].

In contrast, other models overcome the limitations of reporter assays by using endogenous data obtained from genome-wide assays (ChIP-seq, DNase-seq, RNA-seq, CAGE). Different models use different combinations of data derived from these assays to predict diverse aspects of genomic regulation. For instance, some models have been trained with ChIP-seq data in order to predict TF binding or histone modification profiles (DeepBind[13], TBiNet[25], AgentBind[26], BPNet[27], DeepGRN[29,31]). Others are trained with DNase-seq data to predict chromatin accessibility (Basset[19], ChromTransfer[31]). These two features are highly interconnected, as TFs bind to accessible regions of the genome. Thus, some models predict both types of data simultaneously (DeepSEA[14], DanQ[15]). Other models use RNA-seq or CAGE data to predict gene expression levels (AttentiveChrome[18], ExPecto[20], Xpresso[23], Zrimec et al.[24]). As gene expression levels are dependent on TF binding, histone modification profiles and chromatin accessibility, some methods use all these kinds of data to predict all levels of genomic regulation (Basenji[19], Enformer[28]).

Input sequence data

All models analysed here make predictions from input DNA sequence in order to understand how it determines genomic regulatory activity. Raw DNA sequences need to be encoded before they are given to a model. ***One-hot encoding*** is the most used coding method to convert a DNA sequence into a matrix that the network can work with. This method encodes each nucleotide binarily as A = (1 0 0 0), G = (0 1 0 0), C = (0 0 1 0) and T = (0 0 0 1), resulting in a DNA sequence represented by a 4 x *L* matrix, where *L* is the sequence length. The length of DNA sequences used differs between models (Table 1). While the first models only used 600 bp – 1 kb, more recent algorithms used up to 200 kb (Enformer[28]). Such increase was possible thanks to developments in DL architectures, which allowed models to integrate information from more distant loci in the input sequence (see Deep learning architectures in genomics).

Deep learning architectures in genomics

Since deep learning was first theorized, different architectures have been developed for diverse purposes. When applying deep learning in regulatory genomics, one must decide which architecture fits better the purpose of the model. Given that there is a lack of consensus regarding what is the best architecture for each task, researchers should test multiple architectures and decide which one gives better results. In this section, we will describe the most used architectures for regulatory genomics applications: convolutional neural networks and recurrent neural networks. We will also introduce the attention mechanism, a technique that can be combined with one of the previous architectures.

Convolutional Neural Networks

Convolutional neural networks (CNNs) are the most widely used models for image recognition. They were developed by LeCun et al. (1998) to recognise handwritten digits[34]. Their ability to identify patterns makes them suitable in genomics to find DNA motifs that determine, for instance, transcription factor binding, RNA polymerase binding or histone modification.

CNNs are composed of several convolutional layers, pooling layers and fully connected (FC) layers[35] (Figure 2). Convolutional layers are the main part of these models. Each of them contains filters that extract patterns from the previous layer. For instance, in image recognition, the filters of the first convolution layer can detect edges with different orientations from the input image. In genomics, instead of edges these filters recognize DNA motifs. They can be thought of as *position weight matrices (PWMs)*[a] that scan the input sequence to find motif matches. Applying a filter throughout the length of the input sequence in sequential steps results in a vector, with each entry representing the similarity between each position in the raw input sequence and the filter applied. After applying *X* filters, the resulting vectors are combined into a matrix that serves as the input to the next layer. This one detects co-occurrences of motifs at specific positions of the input sequence. Each convolutional layer of the CNN adds a level of abstraction to detect complex combinations of DNA patterns across the full input sequence. Pooling layers (typically max-pooling) are typically employed after convolution layers to reduce the number of parameters and computational cost. Pooling also simplifies the output of each convolutional layer and allows the model to integrate more distant regions of the input sequence, i.e., it enables bigger *receptive fields*[b]. Finally, one or more fully connected layers convert the output of the last pooling layer to the desired output. This can be, as we saw in the previous section, TF binding affinity, gene expression, enhancer-promoter interaction, enhancer/promoter activity, histone modification, chromatin accessibility, or a combination of those. The output can be a unique value for the whole input sequence or different values for sequential sub-sequences of different sizes up to the single-nucleotide resolution. These values can be discrete (classification task) or continuous (regression task).

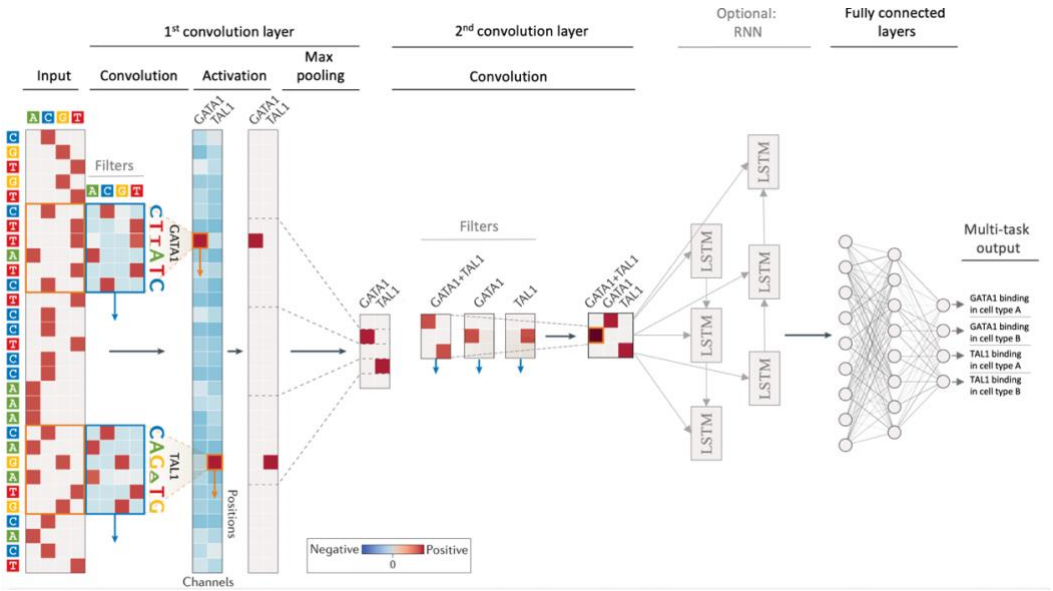

Figure 2. Illustration of a deep learning model. The input sequence is first one-hot encoded. The first convolutional layer scans the input sequence using filters that look for motifs and produce an output matrix with a column for each filter and a row for each position in the input sequence. In this example, the motifs represent TF binding motifs for GATA1 and TAL1. Negative values are transformed into zeros with an activation function. Max-pooling simplifies the output of the first convolutional layer. The second convolutional layer detects position and combination of motifs found in the first layer. The (optional) bidirectional LSTM (BLSTM) layer detects orientation and distances between motifs. Finally, a fully connected network produces the output (in this case, TF binding affinity for GATA1 and TAL1 in two different cell types). Figure adapted from Eraslan et al. (2019)[36].

The number of layers, the number of filters per layer, and their size (known as kernel size) are hyperparameters that need to be tuned to achieve the best performance. The number of layers defines the depth of the model, which, in turn, determines the complexity of motif combinations that it can find. The number of filters per layer determines how many different patterns will be scanned in the output from the previous layer. In the first layer, it represents the number of motifs (or partial motifs) that the model will look for in the input DNA sequence. The filter/kernel size represents the length of these motifs. The weights and coefficients in each filter are parameters that are optimized during the training phase.

[a] Position weight matrices are representations of motifs in DNA sequences with one column for each position filled with symbols (A, T, C, G) representing the four nucleotides with size proportional to the importance of each nucleotide in each position for the feature in question (TF binding, promoter activity…).

[b] The receptive field is the maximum length between two features that the model can account for to produce the output.

CNNs were first used in regulatory genomics by Alipanahi et al. (2015) to predict sequence specificities of transcription factors (DeepBind). Their model outperformed existing non-DL methods that participated in the DREAM5 TF-DNA Motif Recognition Challenge[13]. Zhou and Troyanskaya (2015) developed DeepSEA based also on CNNs to predict TF binding affinity, chromatin accessibility and histone modifications[14]. They achieved a median area under the curve (AUC) for TFBSs of 0.958, surpassing the performance of the best existing method at the time, which was based on a gapped k-mer support vector machine (AUC = 0.896)[37]. Since then, CNNs have been continually and successfully used in other models to predict different phenotypes from DNA sequence (Table 1). Kelley et al. (2016) used this architecture to predict chromatin accessibility in 164 human cell types (Basset)[16]. Movva et al. (2019) predicted enhancer activity based on MPRA experiments in two human cell lines (MPRA-DragoNN)[22]. Agarwal and Shendurne (2020) predicted gene expression levels (Xpresso)[23], and more recently Salvatore et al. (2022) predicted the open or closed chromatin state on a wide range of tissues, cell types and cellular states (ChromTransfer)[31].

CNN architectures have been widely used for different purposes, achieving high performances and allowing interesting findings (see Applications). They can learn properties like motif specificity, orientation and co-occurrence[3]. However, they are limited in the size of the receptive field, thus not being able to model long-range interactions between, for instance, promoters and enhancers. **Dilated convolutional layers** allow CNNs to capture information across longer spans of the sequence, therefore expanding the receptive field[38]. These layers make use of filters with gaps to achieve such an increase. They were used by Avsec et al. (2021) in their BPNet model, which predicted TF binding[27], and by Kelley et al. (2018) in their Basenji model, which predicted cell-type–specific epigenetic and transcriptional profiles[19]. Dilated layers proved to increase the accuracy of model predictions for all data types included in their model.

Recurrent Neural Networks

Recurrent neural networks (RNNs) are widely used for natural language processing due to their capacity to model sequential data. In RNNs, nodes are arranged in a chain in such a way that each node takes as input a subsequence from the previous layer but also the output of the previous node. That way, the output of each node integrates both current and previous sequence information[39,40]. Bidirectional RNNs (BRNN) do this in both directions, therefore integrating current, previous and future sequence information (Figure 2). Thanks to this ability, RNNs can model dependencies between different parts of the input sequence.

RNNs suffer from the “vanishing gradient” problem. This problem arises from the use of the gradient descent algorithm to update the network weights in order to minimize an error function during the training phase. Because of the recurrent architecture this gradient can become smaller and smaller as it is propagated back through the network. The consequence is that the model is not trained properly. Long Short-Term Memory (LSTM) networks and Gated Recurrent Units (GRU) are two variants of RNNs that reduce this problem.

In the regulatory genomics field, RNNs have been mostly used in combination with convolutional layers to predict TF binding affinity, DNase I sensitivity and histone modifications (DanQ[15]), enhancer activity (BiRen[17]), enhancer-promoter interactions (SPEID[21]) and TF binding scores (AgentBind[26]). RNNs can learn sequential properties like motif multiplicity, distance between elements (e.g., a TF binding site from the TSS) and relative order of patterns[3], but their training requires a lot of computational power, and they do not always perform better than CNNs. For instance, Zrimec et al. (2020) tested different neural network architectures that combined CNN layers, RNN layers and fully connected layers, achieving the best performance with a CNN (3 layers) – FC (2 layers) architecture[24]. This showcases that the most complex model is not always the most accurate, and different architectures should be tested to choose the most adequate for each purpose.

Self-attention mechanism

Due to the problem of vanishing gradients, RNNs are also limited in the length of the receptive field. This issue can be circumvented by using self-attention layers in CNN or RNN architectures. These layers transform each position (nucleotide, or *k*-mer) in the input sequence by applying a function that depends on the interaction score between the position in question and every other position in the same input sequence. These interaction scores are optimized during the training phase to bring the best results. The fact that each position can directly attend to all other positions in the input sequence allows the model to account for long-range dependencies (for

example between distant promoters and enhancers)[41]. This mechanism has been applied in AttentiveChrome[18] (predicts histone modification), DeepGRN[29] and TBiNet[25] (predict TF binding profiles). Thanks to the self-attention layers, AttentiveChrome outperformed DeepChrome in most of the human cell types included in the study (average AUC of 0.81 vs. 0.80), and TBiNet outperformed DeepSEA and DanQ (average AUC of 0.95 vs. 0.90 and 0.93, respectively).

In 2021, Avsec et al. published Enformer[27], a deep learning model that uses convolutional and self-attention layers to predict genomic tracks for 5313 epigenetic marks in humans and 1643 in mice from 200 kb-long DNA sequences. The receptive field of this model is five times longer than the previous state-of-the-art model Basenji2[19]. This increase was possible thanks to the use of self-attention layers. Despite architectures with self-attention being the most powerful to account for distant interactions, the flip side is the high computational costs required to implement them, and some claim that such long sequences (200 kb) are not strictly necessary to explain most of the variance in RNA levels[42] (see Discussion and limitations).

Training approach

Multi-task learning

We can observe among the reviewed models that while some of them predict only one type of data (Basset, AgentBind, BPNet, ChromTransfer, Expecto, Xpresso, Zrimec et al.), others perform multiple tasks at the same time (DeepSEA, DanQ, Basenji2, Enformer). Due to the dependencies between the different data types (eventually, they are all determined by TF binding to different DNA motifs) it is logical to think that models trained to predict multiple tasks will perform better than those that only learn one task because multi-task architectures allow learning shared features. This improves the generalisation of models and reduces the computational cost compared to training separate models for each of the tasks[39]. However, *multi-task learning* can cause optimization imbalances, meaning that some tasks can have a bigger influence on the network weights[31]. This might result in a worse performance for weaker tasks (e.g., predicting the epigenomic profile of a cell type that is significantly different from the rest), a problem that might be solved by giving more weight to weaker tasks in order to compensate the imbalance. In any case, one should be aware of this possible inconvenience when working with multi-task models.

Similarly, some models use the same network to predict traits from different organisms. Agarwal and Shendurne (2020) showed that their Xpresso model designed separately for mouse and human cell lines achieved similar performances when applied to the test set from the other species[23], which suggests that the learned regulatory syntax can be generalised between human and mouse. This is also supported by the fact that Enformer achieves the best performances with a model trained simultaneously on human and mouse data[28]. Kelley (2020) also showed that joint training on human and mouse data improved model performance for both species and suggested that the addition of more diverse sequences to the model is probably driving this improvement[43]. Humans and mice are evolutionarily close enough to have orthologous transcription factors and share regulation mechanisms, but distant enough that merging data from both species leads to an increase in sequence diversity that improves the model. This sweet balance is probably the key to reaching more accurate models but as discussed above, we should be careful with optimization imbalances.

To perform multi-task predictions, some have employed **transfer learning**. This concept is inspired by the way in which humans can apply knowledge acquired after solving a problem to tackle new tasks. Similarly, transfer learning allows pre-trained models to be adapted to new tasks. For instance, Salvatore et al. pre-trained a model in a cell-type agnostic way to predict open chromatin regions from DNA sequence and fine-tuned it in a cell-type specific manner[31]. In that way, the model can leverage big amounts of data to learn general rules that apply to all cell types, and tailor these rules towards each specific cell type with less required data. Transfer learning has also been used by Zheng et al. (2021)[26] and Novakovsky et al. (2021)[44] to adapt a model that predicted the binding of many TFs simultaneously to predict each of them separately. The advantage of using transfer learning is that it requires fewer amounts of cell type- or TF-specific data, whereas training independent models for each cell type or each TF results in lower performances due to insufficient data. Transfer learning can also be used to tailor models pre-trained on multiple species towards species-specific predictions. That way, species with less available genomics data can also make use of the benefits of deep learning.

Data splitting for validation

DL models must be trained, selected and evaluated. To avoid overfitting and ensure generalisability, these tasks must be performed on different splits of the dataset. The normal procedure involves partitioning the dataset in training, validation and test sets. The training set is used to train models with different hyperparameters to learn the parameters in the filters (i.e., for the filters in the first layer, the importance of each nucleotide). These models are then assessed on the validation set to choose the best hyperparameters and the best one is finally evaluated on the test set to get a final measure of its performance. Common splitting proportions are 60% for training, 10% for validation and 30% for testing, although when the amount of data is limited the training set can be increased. Partitioning the dataset in training, validation and test sets can be done in different ways. Most authors hold whole chromosomes out as test and validation sets. Others take extra precautions and keep homologous sequences in the same set[19,28]. This ensures that the model has never seen the test data during training, thus it is a measure that should be broadly taken to prevent data leakage between training and test sets, which would result in overfitting. This means the model would memorise the entire input sequence as relevant instead of just the region (e.g., the nucleotides that create a TFBS) that drives the observed feature and that can be applicable to other parts of the genome.

Preferably, each model should be evaluated multiple times to take the average of its performance as a more statistically robust measure. However, this is not always possible when data is scarce. *K*-fold cross-validation presents an alternative. It consists of splitting the data into training, validation and test sets *k* times, and repeating the training and evaluation process to take an average of the model's performance. Although *k*-fold cross-validation is preferred to achieve more certainty about the model's accuracy, training a model can require vast amounts of time, therefore this approach is not always employed.

Model interpretation

DL is often used to teach computers how to do a task that humans can perform easily (e.g., recognizing cats in images, or classifying whether a movie review is positive or negative) but in genomics, DL models go beyond human capabilities and achieve high performances doing very complex tasks. Therefore, humans can leverage these models to gain knowledge about the basis of genomic regulation, but unfortunately their parameters are hard to interpret because of the non-linearity, making it difficult to understand how they arrive to the output. Understanding how these models do their tasks is crucial for two reasons. First, the underlying bases on which models make predictions are, on some occasions, of greater value for researchers than the predictions themselves, as they enable insights into the basic biology driving genome regulation. Second, humans are often reluctant to accept what they do not understand, therefore opening the "black box" would facilitate the acceptance of these models in a clinical setting. *Model interpretation* aims to shed light into the basis underlying a model's predictions. Talukder et al. (2021)[45] and Novakovsky et al. (2022)[46] have extensively reviewed many methods that are used for model interpretation in genomics. Here, we will provide a short overview of these methods (Figure 3, Table 2).

Convolution kernel analysis was common among the first developed methods[13,15,16]. In convolutional neural networks that predict regulatory features from DNA sequences, the filters of the first layer learn short subsequences that drive such features (e.g., a motif to which a TF binds). To visualize and identify such subsequences, convolution kernel analysis takes these filters and looks for input sequences that activated them. After alignment, the nucleotide frequencies corresponding to each position are computed and converted to PWMs (Figure 3A). These PWMs can be interpreted as motifs that potentially drive the regulatory feature under study (TF binding, histone modification, chromatin accessibility, etc.). However, in neural networks multiple filters can cooperate to describe a single motif, being therefore only partial representations of it. These partial motifs could find no matches on databases of known motifs, making it difficult to interpret the results biologically.

Another approach for model interpretation, and the most used approach so far, is ***in silico* mutagenesis (ISM)**. This technique consists of mutating the input sequence one nucleotide at a time and comparing the model's output for the mutated sequence with that of the reference. The difference is used to create heatmaps that highlight important positions for the model's prediction (Figure 3B). A drawback of this technique is that it can miss redundant motifs. Consider a sequence with two binding motifs for a TF, where the presence of each of them individually is sufficient to drive TF binding. A network could succeed in annotating the sequence as

positive, but the model interpretation would not give a high importance score for these two regions, because mutation of any of them separately would not result in a difference in the output.

In a variation of ISM, longer stretches of the input sequence can be mutated to introduce motifs and assess their impact on the output. The difference between the model's output with and without the motif is used to determine motif importance (Figure 3C). This approach can also determine the effect of different backgrounds on known motifs.

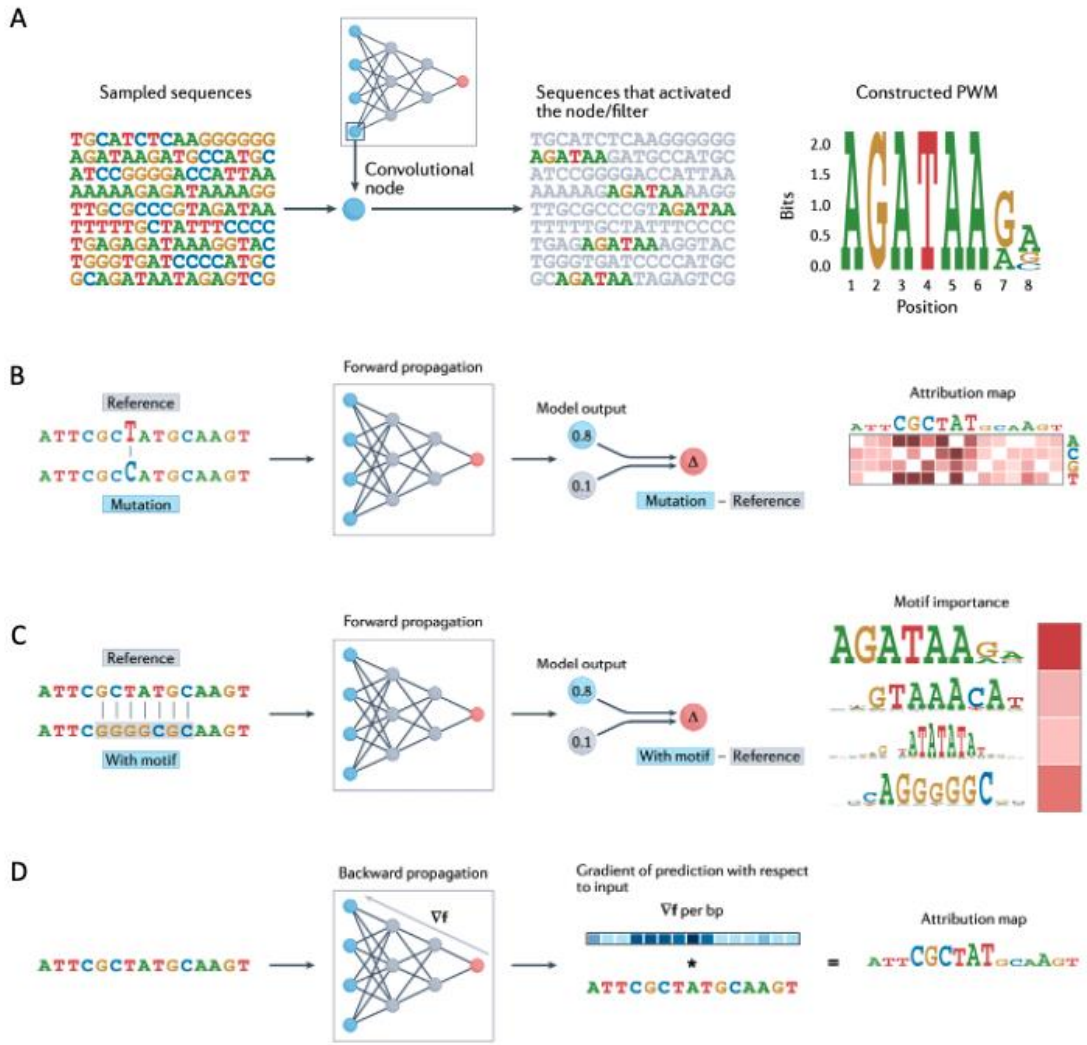

Figure 3. Approaches for model interpretation. A) Convolution kernel analysis. B) In silico mutagenesis. C) Motif insertion. D) Backpropagation-based approach. Figure from Novakovsky et al. (2022) [46] .

ISM is a very intuitive way of interpreting DL models, and despite the mentioned limitation it has been used by many authors[13,14,19–21,28,47] (see Table 2). However, it requires an iteration through the model for each sequence variation, resulting in high computational costs. Backpropagation-based approaches, like **saliency maps**, are computationally more efficient, as they only require one backward pass through the network (Figure 3D), but they present the same limitation as ISM: they can miss redundant motifs. To solve this issue, reference-based methods like **DeepLIFT** were developed [48]. With one backward pass, DeepLIFT compares the activation of each node to the reference activation and computes contribution scores to identify important features.

Finally, **attention mechanism** is a popular approach for model interpretation in RNNs and architectures with attention layers[18,25,28,29]. This technique uses the model's attention scores to emphasize parts of the input that are important to make predictions.

Table 2. Model interpretation techniques used by different deep learning models.

Convolution kernel analysis	Input modification (ISM)	DeepLIFT	Saliency maps	Attention mechanisms
- Alipanahi et al., 2015 (DeepBind) - Park et al. 2020 (TBiNet) - Kelley et al., 2016 (Basset) - Quang et al. 2016 (DanQ) - Singh et al., 2019 (SPEID)	- Alipanahi et al., 2015 (DeepBind) - Lanchantin et al, 2017 (DeMoDashboard) - Zhou et al., 2015 (DeepSEA) - Kelley et al., 2016 (Basset) - Avsec et al. 2021 (Enformer) - Singh et al., 2019 (SPEID) - Zhou et al. 2018 (ExPecto)	- Avsec et al. 2021 (BPNet) - Movva et al. 2019 (MPRA-DragoNN) - De Almeida et al. 2022 (DeepSTARR)	- Lanchantin et al, 2017 (DeMoDashboard) - Kelley et al., 2018 (Basenji)	- Park et al. 2020 (TBiNet) - Chen et al. 2021 (DeepGRN) - Avsec et al. 2021 (Enformer) - Singh et al., 2017 (AttentiveChrome)

Applications

The power of DL models is leveraged in different genomic areas such as variant calling and annotation, disease variant prediction, gene expression and regulation and epigenomics (reviewed by Alharbi and Rashid (2022)[49]). Here, we will describe in detail three key applications of the DL models reviewed (namely, motif discovery, non-coding variant effect prediction and synthetic construct design) and show examples that showcase the value of these models.

Motif discovery

As explained in the previous section, model interpretation enables the discovery of the most informative regions in the input DNA sequence for the model to determine the output. These regions often consist of motifs that drive the regulatory feature in question. For example, using three model interpretation approaches, Lanchantin et al. (2017) found different motifs that matched known TFBS motifs in the JASPAR motif database[47]. Similarly, Park et al. (2020) used convolution kernel analysis to interpret their TBiNet model (predicts TF binding affinities for different TF-cell type combinations). They saw that 142 out of 320 kernels of their model matched known TF binding motifs present in the JASPAR, jolma2013 and uniprobe databases[25]. Although the authors did not delve into the other 178 kernels, they might likely represent partial or undiscovered motifs. These examples show that DL models can be used to discover new DNA motifs that drive genome regulation.

In addition to the discovery of new motifs and important features for different regulatory features, these models offer an opportunity to understand how motif arrangement affects binding affinities. For instance, Avsec et al. (2021)[27] developed a model to predict TF binding profiles in mouse embryonic stem cells (TBNet) and inspected the results thoroughly to learn something about the motif syntax driving TF affinities. Interpretation of TBNet revealed a ~10.5-bp helical periodicity associated with Nanog binding. They also found composite motifs (two or more strictly spaced motifs) and cooperative TF interactions (the binding of a TF to a motif is affected by a second motif and the relative distance between them).

Similarly, Kelley et al. (2016) performed interpretation of their Basset model that predicts chromatin accessibility and they found that the model dedicated the most filters to overlapping parts of the CTCF's binding motif, showing that this was the most predictive pattern of accessibility. They also observed unrecognized filters that may be unknown protein binding sites. Analysis of these filters revealed that they matched proteins that were known to regulate the development of different cell types[16].

Finally, this approach can also be used to fine-tune representations of already known motifs. For example, De Almeida et al. (2022) used DeepLIFT to find regions in input sequences that were important for enhancer annotation by their DeepSTARR model, and they found a significant contribution of sequences adjacent to important known motifs up to ten or more nucleotides. This suggests that the current motif representation is only partially illustrating the entire sequence that drives enhancer activity.

Non-coding variant effect prediction

As stated in the introduction, an advantage of DL models over GWAS/eQTL studies is the possibility of studying the effect of all possible mutations in non-coding regions, without requiring the presence of these variants in the training data. For example, Alipanahi et al. (2015) used mutation maps to visualize the effect of genetic

variants on TF binding affinity. They found that a single nucleotide mutation in the LDL-R promoter disrupted an SP1 binding site, leading to familial hypercholesterolemia[13]. Similarly, Zhou et al. (2015) identified that a C-to-T mutation at a specific breast cancer risk locus led to increased affinity of FOXA1, and a T-to-C mutation at a locus associated with α thalassemia created a binding site for GATA1[14].

The added value of these models compared to GWAS/eQTL studies is not only in the possibility of studying all possible mutations, but also in the fact that they can infer the direct effect on the genome regulation instead of only associating them with an observed phenotype. For example, Kelley et al. (2018) analysed 1170 loci associated with autoimmune diseases and blood cell traits with their model Basenji. It predicted that a C-to-G mutation at one of the loci associated with multiple sclerosis increased transcription of GALC in immune cells and GPR65 in severely acute lymphoblastic leukaemia cell lines, thyroid cells, insular cortex cells, and immune cells[19]. They also showed that a SNP located in a 559-kb gene desert and associated by GWAS studies with vitiligo with high probability created a motif recognized by CTCF. It had been hypothesized that this SNP regulates TYR, a gene located 6.28 Mb away from the SNP that catalyses the conversion of tyrosine to melanin. Basset's prediction suggested a plausible mechanism by which the SNP can affect the regulation of such a distant gene.

Another example of this application is provided by Avsec et al. (2021), who also used their model Enformer to study the direct effect of mutations. In an eQTL study, a C-to-T mutation in an intron ~35 kb downstream of the TSS of the NLRC5 gene had been described to decrease the expression of that gene in whole blood. Enformer not only correctly predicted this change but also showed that this variant affects the TF binding motif of SP1, suggesting that the mechanism of action by which NLRC5 expression is decreased might be through a perturbed SP1 binding affinity. In a similar approach, Singh et al. (2019) used their SPEID model to predict the effects of somatic mutations on enhancer-promoter interaction in melanoma patients and they identified mutations that lower the interaction likelihood.

Finally, the ability to predict the effect of non-coding variants has been applied to prioritize SNPs identified in GWAS studies. Due to linkage disequilibrium, these studies result in many SNPs being associated with a pathology despite not being the causal mutation. DL models can be leveraged to predict the effect of each of these mutations and prioritize those that have a stronger effect for further experimental studies. For instance, Zhou et al. (2018) prioritized SNPs related to immune diseases with their ExPecto model and proved that their top three SNPs showed transcriptional regulatory activity, whereas none of the GWAS lead SNPs showed differences in transcriptional activity.

All in all, these examples show the potential of DL models as *in silico* perturbation tools to predict the direct effect of mutations that may eventually cause a pathology. The fact that they can evaluate non-coding mutations makes them particularly valuable, given that most variants identified by GWAS are non-coding[50]. Such predictions can be used as working hypotheses for further experimental validation.

Synthetic construct design

Finally, all these models further our understanding of how DNA sequences encode genome regulation and eventually, gene expression. This knowledge enables the design of synthetic constructs with desired characteristics. For example, de Almeida et al. (2022) used DeepSTARR to create synthetic enhancers de novo with specific activity levels. They designed 249 enhancers spanning different activity levels according to their model and measured their enhancer activity experimentally, achieving a Pearson correlation coefficient of 0.62. The advantage of using DL models over traditional methods based solely on experimental data on promoter or enhancer activity, is that DL models are not restricted to naturally occurring sequences, but instead can explore the entire sequence space to achieve the desired expression level. For instance, Vaishnav et al. (2022) used a genetic algorithm to design random sequences and predicted the resulting expression with a convolutional model. They chose the 500 sequences with the maximum or minimum expression levels and showed that the designed sequences drove more extreme expression levels than 99% of the naturally occurring sequences[51]. This is of high interest both for synthetic biology applications and for industrial purposes to increase the production of proteins of interest.

Discussion and limitations

Due to the success of DL as a tool in genomics and epigenomics, there has been since 2015 a steep increase in the number of published DL methods that model different aspects of genomic regulation (Figure 1). In this review, we have compared some of these published between 2015 and 2022 (Table 1, Figure 1), with an overview of the multiple alternatives available for each part of their design (input sequence length, predicted features, model architecture, training approach, model interpretation methods and downstream applications). Here, we will discuss some open questions and ideas that stemmed from this analysis and point out some limitations. For an in-depth review of general limitations of DL models in genomics, please see Whalen et al. (2022)[52].

Training data & training approach

An aspect of concern regarding the input data is the origin of the DNA sequence. Most of the models (DeepSEA[14], DanQ[15], DeepGRN[29], Basset[19], TBiNet[25], Enformer[28], Basenji[19], AgentBind[26], ChromTransfer[31]) make the simplifying assumption that the reference genome underlies the functional annotations studied. This poses two problems: 1) We cannot be sure if the features used to train the model came from that sequence, or if the cells from which the data was obtained carried a mutation or SNP that affected the output. This might have important consequences when assessing the impact of mutations, as the model could have learned the wrong rules. However, these models seem to succeed at finding known drivers of diseases, which might be explained by a low impact of common SNPs in the phenotypes predicted or by a low divergence between the reference genome and the genome of the cells from which the data originates. Despite the success of these models so far, this matter should not be disregarded, especially when using data derived from pathogenic cell lines (e.g., cancer cell lines) that most likely will have a set of SNPs more divergent from the reference genome. 2) These models do not account for the possibility of having allele variants. The input sequence contains only one allele, but the observed phenotype (e.g., RNA levels) could result from the effect of the other allele, or the combination of both. These variations cannot be detected with the current model designs, and further efforts should take them into account. New models should use datasets obtained from assays (ChIP-seq, DNase-seq, RNA-seq, etc.) in which the DNA sequence from the same sample is obtained and mapped to genome graphs or multiple genomes[53] instead of the reference genome to account for possible SNPs and structural variants and different alleles.

Given that these models only use the reference genome, it might seem surprising that they can solve the problem of linkage disequilibrium, as they are not leveraging the variability in SNPs between different genomes. This shows the power of DL models over statistical association studies like GWAS and eQTLs, as DL models do not try to learn the regulation of each gene independently, but instead they learn the shared rules that underlie regulation across the genome. We hypothesize that there might be enough sequence variability within the genome to discern the exact nucleotides that form important motifs, but to our knowledge, this argument has not been discussed or used to justify the usage of the reference genome. We anticipate that by using different genomes these models would incorporate more sequence variability and thus improve their performance.

Another aspect of controversy related to the input sequence comes from its length. We have seen that it varies between models, with a tendency to increase as the DL architectures improve their capacity to handle bigger receptive fields. The model with the longest input sequence is Enformer, with 200 kb. The authors of Enformer claim that this increase in input length compared to the previous state-of-the-art model Basenji2 resulted in a substantial performance increase when predicting gene expression. However, it has been shown that the model extracts most of the signal from regions proximal to the gene in question (the proximal 1/3rd of the sequence explains 99% of the variation in RNA levels)[42]. Thus, the increase in performance observed in Enformer could be driven mainly by the increased number of parameters[42]. The reason why Enformer does not attend to distant sequences is of interest for the field. A plausible explanation would be that enhancers tend to be closer to their paired gene, in which case such large receptive fields would not be strictly needed. This is supported by an enhancer-gene pair screening performed by Gasperini et al. (2019), where it was shown that enhancers are separated from their target genes by a median distance of 24.1 kb[54]. Although longer-distance enhancer-gene interactions are possible, perhaps their frequency is not enough for the model to increase the attention weights towards distant regions. Even if their frequency is not very high, Enformer is to this day the only model with a receptive field big enough to detect contributions from enhancers more than 20 kb away from the affected gene. It would be interesting to use the model to systematically predict distant enhancer-gene pairs and investigate if they share common features that determine this distant association.

The problem of the low number of examples of distant enhancer-gene pairs brings us to the topic of data imbalance. ***Data imbalance* or *class imbalance*** is a widespread problem in genomics. This can be intuitively understood if one thinks about the ratio of enhancer to non-enhancer sequences in the genome, the low percentage of the genome to which a TF can bind or the ratio of methylated to unmethylated regions. If the number of positive examples is very low due to the data imbalance, it can be insufficient for the model to learn the biological rules explaining the observed phenotype. Instead, the model could predict a negative output for every instance and yet achieve good accuracy. *Data augmentation* is used to alleviate the problem of class imbalance. It consists of increasing the number of examples with the minority label to reduce the imbalance. Data augmentation can be achieved by, for instance, shifting each positive sequence a few nucleotides right and left and adding the resulting sequences as new positive data[21] or adding the reverse complement of positive sequences to the pool of positive examples[22]. Training the model with augmented data can cause a high false positive rate when the model is applied to real data. A solution for this issue is the approach taken by Singh et al. (2019), who took advantage of transfer learning to pre-train the model on a dataset balanced with data augmentation and fine-tune it later with the original imbalanced data[21]. In addition, data augmentation can also result in overfitting (i.e., instead of learning just the motif that drives the feature under study, the model "memorizes" the larger sequence that has been repeatedly inputted due to the data augmentation technique). In short, although data augmentation alleviates the problem of data imbalance, it can have unwanted consequences, thus being still one of the limitations of DL in the genomics field.

Model evaluation approach

Although some researchers evaluate their models with an external validation experiment (e.g., CRISPR perturbation assay or GWAS studies)[22,27,28], the lack of a shared evaluation approach makes the comparison of the models difficult. A benchmark study, or the agreement to use a shared validation task for all models, would be of high value for the community. Perhaps this could be done in the shape of a competition or challenge like CAGI (Critical Assessment of Genome Interpretation)[55], in which participants had to predict the impact of mutations at every position in five enhancers and nine promoters. The activity of these mutated enhancers and promoters was assessed in a MPRA assay. This dataset was used by Enformer as an external validation experiment, and similar challenges could be designed to evaluate the performance of models that predict other epigenetic features.

Model interpretation

On the topic of model interpretation, a current limitation is the lack of ground truth datasets to evaluate which is the best technique for the discovery of motifs and their syntax. This problem does not come as a surprise, as otherwise we would not need deep learning tools to discover these regulatory features, but it can be partially circumvented by using synthetic datasets. Prakash et al. (2021) have described a pipeline to simulate realistic datasets for benchmarking interpretation models for their ability to discover motifs [56]. Their pipeline models the complexity of regulatory genomic DNA better than previous approaches[26]. They achieve that, for instance, by using dinucleotide shuffled sequences taken from a real dataset as background, instead of a randomly generated sequence, and by inserting known motifs in these background sequences at their original location (both in the negative and positive sets), therefore learning complex, co-operative TF binding patterns that characterise the positive set. With this dataset, they compared different interpretation tools and concluded that DeepLIFT and ISM perform best and second-best.

Future perspective

To conclude this review, we will discuss future perspectives in the field of deep learning for genomic regulation. We have seen a steep improvement in the capacities of DL models in genomics in the last years. The improvement of DL upon other shallow ML methods stems from the fact that they do not require feature extraction but instead use raw input directly and learn the important features during the training phase. This eliminates the necessity to make biological assumptions that can influence a model's results. Perhaps some of the few assumptions that the reviewed models make are the expected length of the motifs driving the regulatory features under study (determined by the kernel size in the first convolutional layer) or the order of motif combinations that the model should be able to integrate (defined by the number of layers, or depth, of the model). However, even these choices can be optimized during the hyperparameter tuning phase.

The capacity to extract important patterns from the input sequence has made them popular tools to unravel how DNA sequence controls regulatory features. Most models developed so far use only DNA sequence as input to predict a whole range of epigenetic features from it. The rationale behind this choice is the ambition to understand how DNA sequence alone determines epigenomic profiles and gene expression levels. However,

adding other types of epigenomic data as input could potentially improve the models by, for example, giving information about the cell type or cell state. It would also enable the inference of relationships between different epigenomic features (e.g., how does DNA methylation affect TF binding or histone modification profiles?). One can think, for instance, of transforming DNA methylation or chromatin accessibility data into a sequential feature and including it as input to a model. This strategy has been used by Chen et al (2021), who introduced chromatin accessibility and gene region annotation as input to their model to predict TF binding[29]. Karollus et al. (2022) also showed that adding extra information (tissue-specific exon-intron ratio of each gene) as input improved the performance of Enformer[42]. Such approaches have not been yet thoroughly explored, and it would be interesting to see which types of data are more valuable to improve model accuracy and what other regulation mechanisms can be derived from model interpretation.

Finally, we discuss the possibility of using other types of epigenomics data as the predicted feature, and the benefits that it could provide to the field. Multi-task models benefit from shared features and dependencies between different modalities. Most models reviewed here use ChIP-seq, DNase-seq, RNA-seq and CAGE[14,15,19,28], but we believe they could benefit further if more types of data were used simultaneously, e.g., DNA methylation or Hi-C sequencing (measures chromatin interactions). The possibilities will continue to increase as new omics assays are developed. In the future, we foresee that DL models will be able to account for a broad range of cell-type specific measurements from DNA sequence, ranging from enhancer-promoter interaction to gene expression, protein isoform levels and protein degradation rates. This could lead to methods able to predict the abundance of each protein in different cell types and states. The value of such methods will increase as advances in DL interpretation tools further improve our understanding of the rules that underlie their predictions.

Plain language summary

Our genome consists of 3 billion base pairs, of which only 2% contain protein-coding genes and the rest of the sequence (98%) consists of non-coding DNA with regulatory functions. The way in which this non-coding DNA affects the regulation of the genome is, for example, through small sequences (called motifs) to which some proteins (called transcription factors) bind to activate or repress the expression of genes. Other motifs determine if DNA is accessible for these transcription factors or the 3D conformation of the genome, which can also affect gene expression. Although many of these motifs are already known, it is still difficult to predict their integrated effect, and how mutations in these sequences would affect their functions.

In the last years, the explosion of datasets derived from assays that study different aspects of genomic regulation (transcription factor binding state, DNA accessibility, 3D conformation of the genome, etc.) has enabled the use of deep learning to understand how DNA sequence determines these regulatory features: to find the relevant motifs and understand their effect. Deep learning is a form of machine learning that mimics the way in which humans learn how to perform a task: by using examples. By providing a deep learning model with big amounts of labelled data, it learns to detect patterns that are used to annotate unseen data. Deep learning has already proved to be very successful in the fields of image and audio recognition, object detection and natural language processing, and it is nowadays present in our daily lives, from unlocking our phones with facial recognition to translating a text instantly with an online application. In the field of genomics, it has also proved successful to predict different regulatory features from DNA sequence, and to find the motifs driving these features.

To understand the success of deep learning to detect motifs and combinations of them in the DNA, let's first imagine a model that determines if an image contains a cat. The first layers of the model detect small patterns like eyes, mouth, whiskers, tail or paws, and subsequent layers detect combinations of these small patterns to determine whether there's a cat in the image. Similarly, we can imagine a model that instead of eyes or mouths in an image detects small motifs in a DNA sequence and combinations of those that determine whether a gene will be expressed or not.

Since 2015, several deep learning models have been developed to predict different regulatory features from DNA. In addition to their success to find DNA motifs, they are also great tools to predict what would happen if there was a mutation in these motifs, which makes them very valuable for predicting how mutations can cause a disease. All these models use different types of data, different model structures, and are trained and tested in different ways. In this review, we have compared them by analysing each step of their design, and we have outlined their limitations to help other researchers focus on the aspects that should be improved in future models.

Bibliography

1. Sanger F, Nicklen S, Coulson AR. DNA sequencing with chain-terminating inhibitors. *Proceedings of the National Academy of Sciences*. 1977;74(12):5463-5467. doi:10.1073/pnas.74.12.5463

2. Dunham I, Kundaje A, Aldred SF, et al. An integrated encyclopedia of DNA elements in the human genome. *Nature*. 2012;489(7414):57-74. doi:10.1038/nature11247

3. Zrimec J, Buric F, Kokina M, Garcia V, Zelezniak A. Learning the Regulatory Code of Gene Expression. *Front Mol Biosci*. 2021;8. doi:10.3389/fmolb.2021.673363

4. Misteli T. The Self-Organizing Genome: Principles of Genome Architecture and Function. *Cell*. 2020;183(1):28-45. doi:10.1016/j.cell.2020.09.014

5. Elliott K, Larsson E. Non-coding driver mutations in human cancer. *Nat Rev Cancer*. 2021;21(8):500-509. doi:10.1038/s41568-021-00371-z

6. Uffelmann E, Huang QQ, Munung NS, et al. Genome-wide association studies. *Nature Reviews Methods Primers*. 2021;1(1):59. doi:10.1038/s43586-021-00056-9

7. Taliun D, Harris DN, Kessler MD, et al. Sequencing of 53,831 diverse genomes from the NHLBI TOPMed Program. *Nature*. 2021;590(7845):290-299. doi:10.1038/s41586-021-03205-y

8. Orlando V. Mapping chromosomal proteins in vivo by formaldehyde-crosslinked-chromatin immunoprecipitation. *Trends Biochem Sci*. 2000;25(3):99-104. doi:10.1016/S0968-0004(99)01535-2

9. Song L, Crawford GE. DNase-seq: a high-resolution technique for mapping active gene regulatory elements across the genome from mammalian cells. *Cold Spring Harb Protoc*. 2010;2010(2):pdb-prot5384.

10. Nagalakshmi U, Wang Z, Waern K, et al. The Transcriptional Landscape of the Yeast Genome Defined by RNA Sequencing. *Science (1979)*. 2008;320(5881):1344-1349. doi:10.1126/science.1158441

11. Shiraki T, Kondo S, Katayama S, et al. Cap analysis gene expression for high-throughput analysis of transcriptional starting point and identification of promoter usage. *Proceedings of the National Academy of Sciences*. 2003;100(26):15776-15781. doi:10.1073/pnas.2136655100

12. Inoue F, Ahituv N. Decoding enhancers using massively parallel reporter assays. *Genomics*. 2015;106(3):159-164. doi:https://doi.org/10.1016/j.ygeno.2015.06.005

13. Alipanahi B, Delong A, Weirauch MT, Frey BJ. Predicting the sequence specificities of DNA- and RNA-binding proteins by deep learning. *Nat Biotechnol*. 2015;33(8):831-838. doi:10.1038/nbt.3300

14. Zhou J, Troyanskaya OG. Predicting effects of noncoding variants with deep learning-based sequence model. *Nat Methods*. 2015;12(10):931-934. doi:10.1038/nmeth.3547

15. Quang D, Xie X. DanQ: A hybrid convolutional and recurrent deep neural network for quantifying the function of DNA sequences. *Nucleic Acids Res*. 2016;44(11). doi:10.1093/nar/gkw226

16. Kelley DR, Snoek J, Rinn JL. Basset: Learning the regulatory code of the accessible genome with deep convolutional neural networks. *Genome Res*. 2016;26(7):990-999. doi:10.1101/gr.200535.115

17. Yang B, Liu F, Ren C, et al. BiRen: Predicting enhancers with a deep-learning-based model using the DNA sequence alone. *Bioinformatics*. 2017;33(13):1930-1936. doi:10.1093/bioinformatics/btx105

18. Singh R, Lanchantin J, Sekhon A, Qi Y. Attend and Predict: Understanding Gene Regulation by Selective Attention on Chromatin. *Adv Neural Inf Process Syst*. 2017;30:6785-6795.

19. Kelley DR, Reshef YA, Bileschi M, Belanger D, McLean CY, Snoek J. Sequential regulatory activity prediction across chromosomes with convolutional neural networks. *Genome Res*. 2018;28(5):739-750. doi:10.1101/gr.227819.117

20. Zhou J, Theesfeld CL, Yao K, Chen KM, Wong AK, Troyanskaya OG. Deep learning sequence-based ab initio prediction of variant effects on expression and disease risk. *Nat Genet*. 2018;50(8):1171-1179. doi:10.1038/s41588-018-0160-6

21. Singh S, Yang Y, Póczos B, Ma J. Predicting enhancer-promoter interaction from genomic sequence with deep neural networks. *Quantitative Biology*. 2019;7(2):122-137. doi:10.1007/s40484-019-0154-0

22. Movva R, Greenside P, Marinov GK, Nair S, Shrikumar A, Kundaje A. Deciphering regulatory DNA sequences and noncoding genetic variants using neural network models of massively parallel reporter assays. *PLoS One*. 2019;14(6). doi:10.1371/journal.pone.0218073

23. Agarwal V, Shendure J. Predicting mRNA Abundance Directly from Genomic Sequence Using Deep Convolutional Neural Networks. *Cell Rep*. 2020;31(7). doi:10.1016/j.celrep.2020.107663

24. Zrimec J, Börlin CS, Buric F, et al. Deep learning suggests that gene expression is encoded in all parts of a co-evolving interacting gene regulatory structure. *Nat Commun*. 2020;11(1). doi:10.1038/s41467-020-19921-4

25. Park S, Koh Y, Jeon H, Kim H, Yeo Y, Kang J. Enhancing the interpretability of transcription factor binding site prediction using attention mechanism. *Sci Rep*. 2020;10(1). doi:10.1038/s41598-020-70218-4

26. Zheng A, Lamkin M, Zhao H, Wu C, Su H, Gymrek M. Deep neural networks identify sequence context features predictive of transcription factor binding. *Nat Mach Intell*. 2021;3(2):172-180. doi:10.1038/s42256-020-00282-y

27. Avsec Ž, Weilert M, Shrikumar A, et al. Base-resolution models of transcription-factor binding reveal soft motif syntax. *Nat Genet*. 2021;53(3):354-366. doi:10.1038/s41588-021-00782-6

28. Avsec Ž, Agarwal V, Visentin D, et al. Effective gene expression prediction from sequence by integrating long-range interactions. *Nat Methods*. 2021;18(10):1196-1203. doi:10.1038/s41592-021-01252-x

29. Chen C, Hou J, Shi X, Yang H, Birchler JA, Cheng J. DeepGRN: prediction of transcription factor binding site across cell-types using attention-based deep neural networks. *BMC Bioinformatics*. 2021;22(1). doi:10.1186/s12859-020-03952-1

30. de Almeida BP, Reiter F, Pagani M, Stark A. DeepSTARR predicts enhancer activity from DNA sequence and enables the de novo design of synthetic enhancers. *Nat Genet*. 2022;54(5):613-624. doi:10.1038/s41588-022-01048-5

31. Salvatore M, Horlacher M, Winther O, Andersson R. Transfer learning reveals sequence determinants of regulatory element accessibility. *bioRxiv*. Published online 2022. doi:10.1101/2022.08.05.502903

32. van Arensbergen J, Pagie L, FitzPatrick VD, et al. High-throughput identification of human SNPs affecting regulatory element activity. *Nat Genet*. 2019;51(7):1160-1169. doi:10.1038/s41588-019-0455-2

33. van Arensbergen J, FitzPatrick VD, de Haas M, et al. Genome-wide mapping of autonomous promoter activity in human cells. *Nat Biotechnol*. 2017;35(2):145-153. doi:10.1038/nbt.3754

34. LeCun Y, Boser B, Denker J, et al. Handwritten Digit Recognition with a Back-Propagation Network. In: *Advances in Neural Information Processing Systems*. Vol 2. ; 1989.

35. Al-Ajlan A, el Allali A. CNN-MGP: Convolutional Neural Networks for Metagenomics Gene Prediction. *Interdiscip Sci*. 2019;11(4):628-635. doi:10.1007/s12539-018-0313-4

36. Eraslan G, Avsec Ž, Gagneur J, Theis FJ. Deep learning: new computational modelling techniques for genomics. *Nat Rev Genet*. 2019;20(7):389-403. doi:10.1038/s41576-019-0122-6

37. Ghandi M, Lee D, Mohammad-Noori M, Beer MA. Enhanced Regulatory Sequence Prediction Using Gapped k-mer Features. *PLoS Comput Biol*. 2014;10(7):e1003711-. https://doi.org/10.1371/journal.pcbi.1003711

38. Gupta A, Rush AM. Dilated convolutions for modeling long-distance genomic dependencies. *arXiv preprint arXiv:171001278*. Published online 2017.

39. Angermueller C, Pärnamaa T, Parts L, Stegle O. Deep learning for computational biology. *Mol Syst Biol*. 2016;12(7):878. doi:10.15252/msb.20156651

40. Yue T, Wang H. Deep Learning for Genomics: A Concise Overview. Published online February 2, 2018. http://arxiv.org/abs/1802.00810

41. Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need. *Adv Neural Inf Process Syst*. 2017;30.

42. Karollus A, Mauermeier T, Gagneur J. Current sequence-based models capture gene expression determinants in promoters but mostly ignore distal enhancers. *bioRxiv*. Published online 2022. doi:10.1101/2022.09.15.508087

43. Kelley DR. Cross-species regulatory sequence activity prediction. *PLoS Comput Biol*. 2020;16(7). doi:10.1371/journal.pcbi.1008050

44. Novakovsky G, Saraswat M, Fornes O, Mostafavi S, Wasserman WW. Biologically relevant transfer learning improves transcription factor binding prediction. *Genome Biol*. 2021;22(1):280. doi:10.1186/s13059-021-02499-5

45. Talukder A, Barham C, Li X, Hu H. Interpretation of deep learning in genomics and epigenomics. *Brief Bioinform*. 2021;22(3). doi:10.1093/bib/bbaa177

46. Novakovsky G, Dexter N, Libbrecht MW, Wasserman WW, Mostafavi S. Obtaining genetics insights from deep learning via explainable artificial intelligence. *Nat Rev Genet*. Published online October 3, 2022. doi:10.1038/s41576-022-00532-2

47. Lanchantin J, Singh R, Wang B, Qi Y. Deep motif dashboard: Visualizing and understanding genomic sequences using deep neural networks. In: *Pacific Symposium on Biocomputing*. Vol 0. World Scientific Publishing Co. Pte Ltd; 2017:254-265. doi:10.1142/9789813207813_0025

48. Shrikumar A, Greenside P, Kundaje A. Learning important features through propagating activation differences. In: *International Conference on Machine Learning*. PMLR; 2017:3145-3153.

49. Alharbi WS, Rashid M. A review of deep learning applications in human genomics using next-generation sequencing data. *Hum Genomics*. 2022;16(1):26. doi:10.1186/s40246-022-00396-x

50. Maurano MT, Humbert R, Rynes E, et al. Systematic Localization of Common Disease-Associated Variation in Regulatory DNA. *Science (1979)*. 2012;337(6099):1190-1195. doi:10.1126/science.1222794

51. Vaishnav ED, de Boer CG, Molinet J, et al. The evolution, evolvability and engineering of gene regulatory DNA. *Nature*. 2022;603(7901):455-463. doi:10.1038/s41586-022-04506-6

52. Whalen S, Schreiber J, Noble WS, Pollard KS. Navigating the pitfalls of applying machine learning in genomics. *Nat Rev Genet*. 2022;23(3):169-181. doi:10.1038/s41576-021-00434-9

53. Paten B, Novak AM, Eizenga JM, Garrison E. Genome graphs and the evolution of genome inference. *Genome Res*. 2017;27(5):665-676.

54. Gasperini M, Hill AJ, McFaline-Figueroa JL, et al. A Genome-wide Framework for Mapping Gene Regulation via Cellular Genetic Screens. *Cell*. 2019;176(1):377-390.e19. doi:10.1016/j.cell.2018.11.029

55. Shigaki D, Adato O, Adhikari AN, et al. Integration of multiple epigenomic marks improves prediction of variant impact in saturation mutagenesis reporter assay. *Hum Mutat*. 2019;40(9):1280-1291. doi:https://doi.org/10.1002/humu.23797

56. Prakash E, Shrikumar A, Kundaje A. Towards More Realistic Simulated Datasets for Benchmarking Deep Learning Models in Regulatory Genomics. *bioRxiv*. Published online 2021. doi:10.1101/2021.12.26.474224