nature machine intelligence

Review article

https://doi.org/10.1038/s42256-025-01007-9

# Transformers and genome language models

Received: 29 January 2024

Accepted: 31 January 2025

Published online: 13 March 2025

Check for updates

Micaela E. Consens[1,2,3], Cameron Dufault[1], Michael Wainberg[2,4,5,6,7], Duncan Forster[2,8,9], Mehran Karimzadeh[2,10,11,12], Hani Goodarzi[10,11,12], Fabian J. Theis[13,14,15,16], Alan Moses[1,17] & Bo Wang[1,2,3,18] ✉

Large language models based on the transformer deep learning architecture have revolutionized natural language processing. Motivated by the analogy between human language and the genome's biological code, researchers have begun to develop genome language models (gLMs) based on transformers and related architectures. This Review explores the use of transformers and language models in genomics. We survey open questions in genomics amenable to the use of gLMs, and motivate the use of gLMs and the transformer architecture for these problems. We discuss the potential of gLMs for modelling the genome using unsupervised pretraining tasks, specifically focusing on the power of zero- and few-shot learning. We explore the strengths and limitations of the transformer architecture, as well as the strengths and limitations of current gLMs more broadly. Additionally, we contemplate the future of genomic modelling beyond the transformer architecture, based on current trends in research. This Review serves as a guide for computational biologists and computer scientists interested in transformers and language models for genomic data.

In the past decade deep learning has been applied to complex tasks, including generating art[1,2], representing language[3–6], and predicting protein structures from amino acid sequences[7]. The success of deep learning is attributed to the size, accessibility and multimodality of available datasets along with the push to generate larger and larger models[8].

In genomics, novel techniques[9,10] such as chromatin accessibility[11,12], methylation[13,14], transcriptional status[15], chromatin structure[16], and bound molecules[12] have yielded a large and varied source of data[17]. Deep learning tools have been widely applied to genomics due to their potential to solve many challenges posed by omics datasets[18]. For example, a primary application of deep learning in genomics is to predict high-dimensional modalities (transcription factor binding, RNA binding, chromatin accessibility, contact-maps, gene expression, RNA sequencing (RNA-seq) coverage, promoter/enhancer regions, and more[19–39]) from DNA sequence. Typically, these deep learning models have been dominated by convolutional neural network (CNN) structures[20–26,29,32]. However, driven by the success of transformer models[40] in computer vision and natural language processing (NLP), transformers are now being applied to genomic modelling problems[30,31,39–43].

Although transformers were originally conceived for sequence-to-sequence problems, they have since been adapted for diverse genomics tasks, such as predicting a quantitative assay or performing

[1]Department of Computer Science, University of Toronto, Toronto, Ontario, Canada. [2]Vector Institute for Artificial Intelligence, Toronto, Ontario, Canada. [3]Peter Munk Cardiac Center, University Health Network, Toronto, Ontario, Canada. [4]Prosserman Centre for Population Health Research, Lunenfeld-Tanenbaum Research Institute, Toronto, Ontario, Canada. [5]Department of Psychiatry, University of Toronto, Toronto, Ontario, Canada. [6]Biostatistics Division, Dalla Lana School of Public Health, University of Toronto, Toronto, Ontario, Canada. [7]Institute of Medical Science, University of Toronto, Toronto, Ontario, Canada. [8]Department of Molecular Genetics, University of Toronto, Toronto, Ontario, Canada. [9]The Donnelly Centre, University of Toronto, Toronto, Ontario, Canada. [10]Arc Institute, Palo Alto, CA, USA. [11]Department of Biochemistry and Biophysics, University of California, San Francisco, San Francisco, CA, USA. [12]Helen Diller Family Comprehensive Cancer Center, University of California, San Francisco, San Francisco, CA, USA. [13]Institute of Computational Biology, Department of Computational Health, Helmholtz Munich, Munich, Germany. [14]TUM School of Life Sciences Weihenstephan, Technical University of Munich, Munich, Germany. [15]Department of Mathematics, School of Computation, Information and Technology, Technical University of Munich, Garching, Germany. [16]Munich Center for Machine Learning, Technical University of Munich, Garching, Germany. [17]Department of Cell and Systems Biology, University of Toronto, Toronto, Ontario, Canada. [18]Department of Laboratory Medicine and Pathobiology, University of Toronto, Toronto, Ontario, Canada. ✉e-mail: bowang@vectorinstitute.ai

classification[31,43]. Computational advances continue to be made in improving the efficiency of the transformer, allowing the size of these models to increase along with their predictive power[44,45]. This has accelerated the application of transformer models for genomics. More recently, genomic models are being proposed with novel architectures that claim to outperform the transformer[46].

This Review will discuss the trajectory of deep learning approaches in genomics, including transformers, with a detailed discussion of the applications, successes and challenges of gLMs. gLMs, as referred to here, are models pretrained on sequences of genomic tokens such as nucleotides or *k*-mers. Numerous review papers have explored deep learning models in genomics, with topics spanning from general introductions to specific discussions on model interpretation, understanding gene regulation, predicting the impact of genetic variation, and unveiling new applications[18,47–57]. Our Review contributes to the field by specifically and exclusively focusing on transformers for genomic sequence prediction, transformers as gLMs, and gLMs with alternative architectures to transformers. We will not discuss transformers or language models for protein sequences as these topics have been reviewed elsewhere[58]. Given the rapid pace of the field, we acknowledge we cannot be comprehensive and instead focus on select models and advancements up to July 2024.

In this Review, we introduce an open problem in genomics and discuss the potential of the transformer model and, more broadly, the potential of gLMs solve it. We then present an overview of the transformer architecture in the context of genomics, along with a briefing on new approaches based upon state space models (SSMs), which have been claimed to outperform transformers. We also introduce hybrid models, which we define as models that include transformer modules but are not language models, instead being directly trained to predict assay data. We then introduce transformer-based gLMs and alternative-architecture gLMs. This Review is intended for computational biologists with deep learning experience interested in understanding gLMs and similar tools, and computer scientists keen on gaining insights into the research opportunities within this exciting field.

## An open problem in genomics

Only about 2% of the human genome encodes proteins, leaving the vast majority as non-coding regions whose functions remain poorly understood[55]. A key goal in genomics is deciphering the regulatory grammar of the genome: understanding how regulatory elements interact with one another and the genes they influence to modulate gene expression. This includes understanding how these interactions vary across environmental conditions, developmental stages, and cell types.

Deep learning is one of many tools applied to this challenge[56]. As regulation in the genome is large and complex, researchers focus on smaller tasks such as identifying regulatory elements like promoters or enhancers, classifying mutations as deleterious or benign, predicting transcription-factor binding sites, splice sites, gene expression, and chromatin accessibility[19–39].

### What data do deep learning models for genomics train on?

The genome of an organism is defined by its complete set of DNA sequences. Using only four different nucleotides (often represented by A, T, G and C), the information necessary for life is encoded in continuous stretches of DNA[58]. DNA has a double-stranded structure, with two complementary strands bonded together and read in the opposite directions[59]. For many complex organisms, including humans, DNA is wound up and packaged into chromosomes that are millions of base pairs long. Each chromosome contains many genes, regions of DNA transcribed into RNA. During transcription, many different protein complexes are recruited in a specific order, often through patterns of nucleotides known as motifs[58]. These motifs are found within the non-coding DNA sequences surrounding a gene. There are examples of non-coding DNA sequences that can affect the regulation of genes thousands of base pairs away, acting as promoters, enhancers, silencers or insulators.

DNA is sequenced in fragments[58], assembled into full-genome references for well-studied species, and stored in public databases like GenBank[60] and RefSeq[61]. For less-studied species, complete assemblies may not yet exist. Vast quantities of short DNA reads can be found in sequence read databases such as the Sequence Read Archive (SRA)[62]. Other data modalities capture additional genomic information beyond sequence identity, such as 3D genome organization and transcriptional activity. Examples of this include assay for transposase-accessible chromatin with sequencing (ATAC-seq)[11] and DNase-seq[63] for DNA accessibility, Hi-C[64] for 3D contact maps, chromatin immunoprecipitation followed by high-throughput sequencing (ChIP–seq)[12] for protein–DNA interactions, and RNA-seq[14] or single-cell RNA-seq[15] for transcriptional activity. Experiments such as CRISPR (clustered regularly interspaced short palindromic repeats) perturbations[65] help to identify regions within non-coding DNA that when knocked down or out result in gene expression changes. These regions are candidates for enhancers, regions involved in transcription that promote the recruitment of transcription machinery, and therefore increase gene expression. Publicly available sources of these other modalities of genomic data include ENCODE[66], Roadmap[67], GTEx[68] and the 1000 Genomes Project[69].

DNA sequence data is either one-hot encoded or tokenized when inputted to a deep learning model. One-hot encoding DNA transforms an *N* length sequence into a 4 × *N* matrix of zeros and ones indicating the presence or absence of a specific nucleotide at each position in the sequence[52]. Tokenization strategies vary, with single-nucleotide tokens, *k*-mers or byte-pair encoding (BPE)[70] creating vocabularies of different sizes (Fig. 1). *k*-mer tokenization borrows from bioinformatic principles, treating *k*-length nucleotide subsequences as the 'words' of the genomic language. BPE[70] iteratively merges the most commonly co-occurring nucleotides in the genome to build a genomic vocabulary of a specified size, with varying length 'words'. While chromosomes are millions of base pairs long, deep learning models are limited to shorter inputs, typically up to 1,000 tokens. However, clever tokenization (for example, if some tokens represent 100 bp length motifs) can extend the effective context to 10,000 bp. Full-chromosome contexts remain an open challenge. Instead, as chromosome-length sequences are too long, they are often split into manageable lengths before tokenization or one-hot encoding.

### Why transformers and gLMs?

Transformers excel in DNA representation largely due to their attention mechanism, which captures relationships across entire sequences, independent of nucleotide proximity[31,42,71] (Supplementary Information, appendix A). This contrasts with recurrent neural networks (RNNs), which struggle with long-range dependencies, and CNNs, which are limited to capturing relationships in fixed window sizes (Fig. 2). Transformers, originally developed as encoder-decoders[72] (see 'Architecture' section), naturally build on the success of earlier encoder-decoder models for genomic sequences (Supplementary Information, 'Encoder-decoders' section).

Most DNA sequence data are unlabelled[60–62]. gLMs address this through pretraining, learning generalizable representations from genomic tokens without relying on human annotations (Fig. 1). This reduces bias and improves downstream task performance. Pretraining tasks, when well designed[59,73], enhance a model's ability to perform 'few-shot' and 'zero-shot' tasks (see 'pretraining' in Box 1). Few-shot performance is particularly valuable in genomics, where biologists often work with limited, well-characterized examples (for example, microRNAs or enhancers). Strong zero-shot performance has the potential to discover new regulatory grammar within the genome.

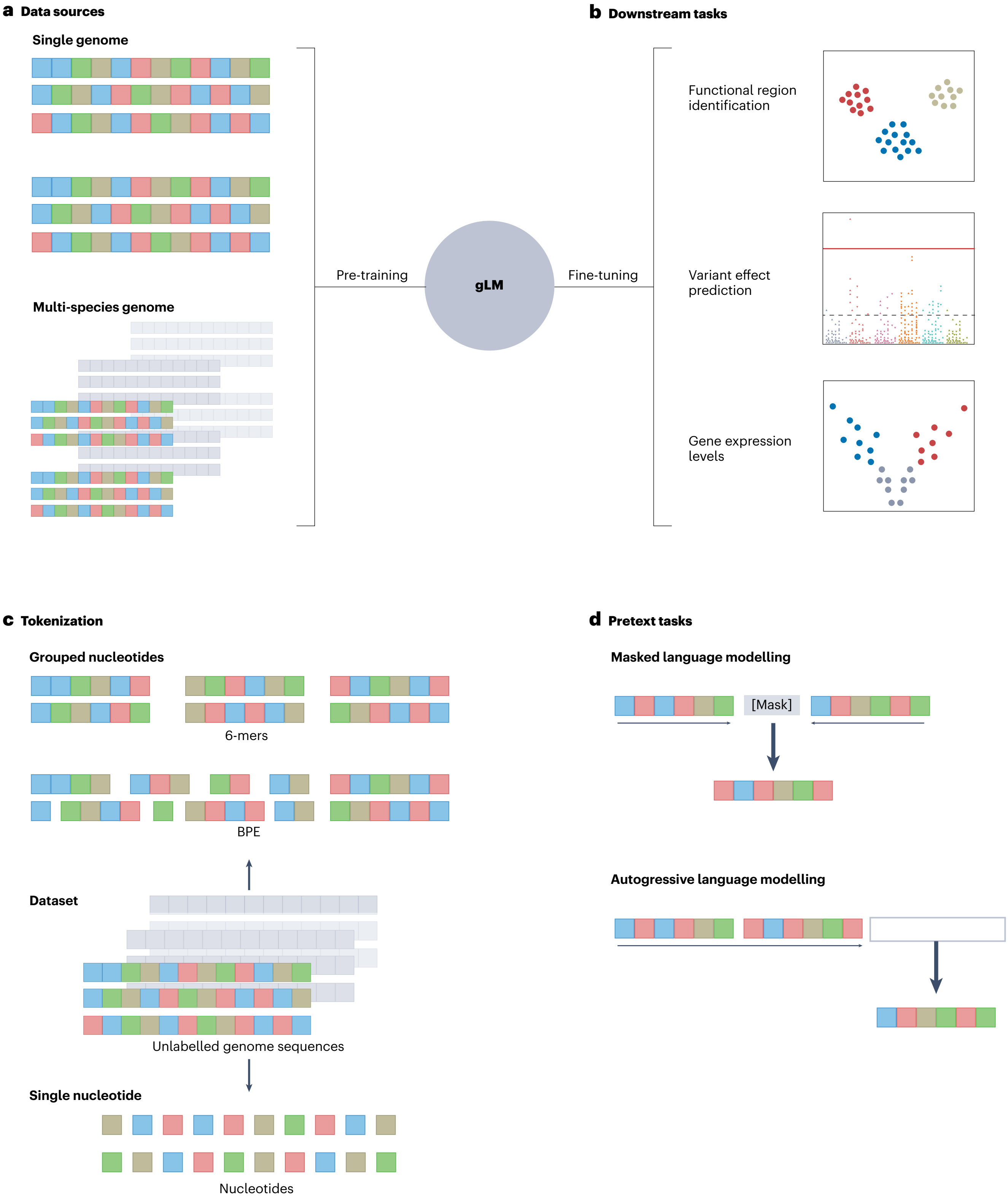

**Fig. 1 | A big-picture look at the diverse applications of gLMs. a**, gLMs can be trained to process DNA sequence data from one (often human) or more species and extract signals to make predictions on downstream tasks. Pretraining allows gLMs to learn the underlying structure of a dataset. **b**, Downstream task performance is evaluated after fine-tuning. Downstream tasks can include functional region identification, variant effect prediction, and gene expression level prediction. **c**, DNA sequences can be tokenized on the single-nucleotide level, multiple nucleotides can be grouped into *k*-mers, or learned vocabularies (such as BPE) can be used. **d**, gLMs are often pretrained using the masked language modelling pretext task, where a percentage of tokens are masked at different random positions in each sequence of the dataset. The model can use information from preceding and succeeding tokens to predict each masked token. Alternatively, in autoregressive language modelling, each token is predicted in order, and therefore only preceding information is used.

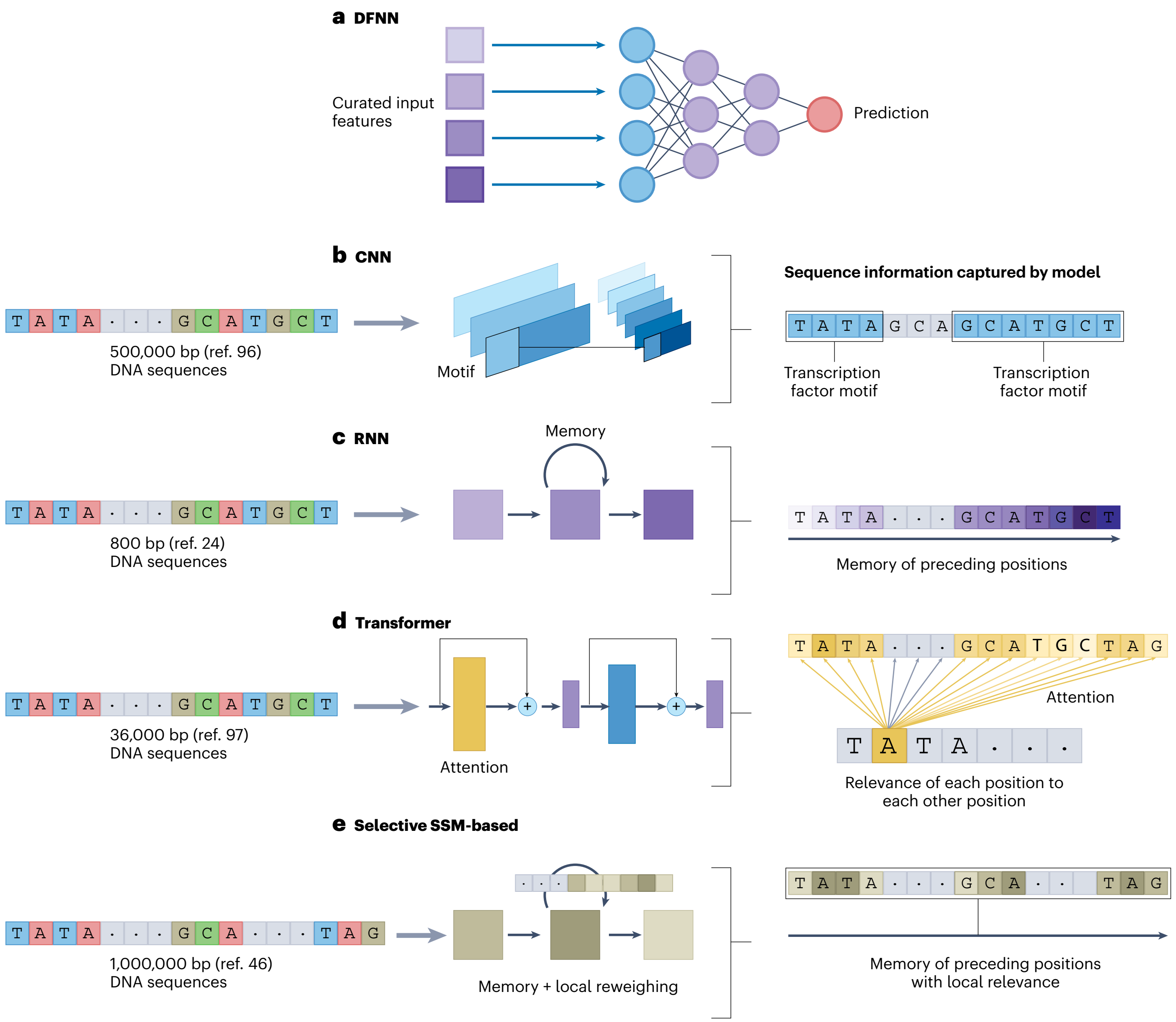
a DFNN
Curated input features
Prediction
b CNN
T A T A · · · G C A T G C T
500,000 bp (ref. 96)
DNA sequences
Motif
Sequence information captured by model
T A T A G C A G C A T G C T
Transcription factor motif
Transcription factor motif
c RNN
Memory
T A T A · · · G C A T G C T
800 bp (ref. 24)
DNA sequences
T A T A · · · G C A T G C T
Memory of preceding positions
d Transformer
T A T A · · · G C A T G C T
36,000 bp (ref. 97)
DNA sequences
Attention
T A T A · · · G C A T G C T A G
Attention
T A T A · · ·
Relevance of each position to each other position
e Selective SSM-based
· · ·
T A T A · · · G C A · · · T A G
1,000,000 bp (ref. 46)
DNA sequences
Memory + local reweighing
T A T A · · · G C A · · T A G
Memory of preceding positions with local relevance

**Fig. 2 | A comparison of how different genomic deep learning models operate on DNA sequence data.** All DNA sequence lengths are given as the lengths used by a recently proposed model in the Review (in order from top to bottom: Borzoi[95] for CNN, DeepMILO[24] for RNN, GENA-LM[96] for transformer, HyenaDNA[46] for S4-based). **a**, A DFNN is capable of taking a curated feature set, either manually curated or taken from the output of another model and making a prediction. **b**, A CNN can directly take as input a DNA sequence, and use convolutions to scan across a sequence to capture local patterns, or motifs, within a DNA sequence. This allows a CNN to pick up motifs that repeat across DNA, like promoters and genes. **c**, An RNN can take as input a DNA sequence and scan along its entirety while retaining a memory of what it has already seen. The RNN can use its memory as context to inform sequence information it has yet to see. **d**, A transformer can take as input a relatively short DNA sequence and use attention to 'attend' to every position within the inputted sequence. Attention allows the transformer to capture medium-range dependencies by modelling global context as the relationship between every token of the input sequence. **e**, A selective SSM layer, like the RNN, can scan along an entire sequence maintaining a memory of previous input tokens, but unlike the RNN, recontextualizes new tokens dynamically based on previous sequence information.

## Transformers

Understanding the use of transformers in the context of genomic modelling requires a foundational understanding of the architecture and its training regime. In explaining the transformer, we assume readers have a grasp of several prominent concepts in machine learning, including the architectures of deep feed-forward neural networks (DFNNs), CNNs and RNNs. For a comprehensive introduction to deep learning, specifically in the context of genomics, we direct readers to other reviews[18,47,49,51]. We refer readers to previous publications to familiarize themselves with DFNNs, CNNs[74] and RNNs[75,76] (Box 1 provides concise definitions).

### Architecture

The transformer architecture, introduced in 2017, features layers of stacked self-attention mechanisms, typically multi-headed attention, alongside addition and normalization layers, skip-connections and fully connected layers for final output predictions[77]. Here, we define encoder-only and decoder-only transformers. For a detailed breakdown of the attention formula and multi-head attention, see Supplementary Information, appendix A. For detailed information on the historical application of the subcomponents within the transformer module, and the discussion of the transformer as an encoder-decoder framework, see Supplementary Information.

The original transformer model was introduced as an encoder-decoder framework, but transformers can also be implemented as encoder-only or decoder-only models[77]. Bidirectional encoder representations from transformers (BERT)[3] is an encoder-only model with 12 layers and 12 attention heads, pretrained using the masked language modelling (MLM) task (see 'Pretraining' section). Generative

BOX 1

## Glossary of key terms

Attention
A mechanism in transformers that dynamically weights the importance of different input elements in a sequence. The input sequence is projected to a set of learned key, query and value matrices. The dot product is computed between the keys and queries to determine the similarity of every pair of input elements. The resulting matrix is then softmaxed to get a relevancy score between 0–1 of every pairwise element in the input sequence, known as the attention matrix. This attention matrix is multiplied by the values matrix so that every input element's representation is updated by every other input element's representation, according to their pairwise relevance.

CNN
Convolutional neural network. A neural network with convolutional layers that aggregate information in spatially proximate regions according to learned parameters. In genomics, these models detect patterns in DNA sequences using one-dimensional convolutions.

Decoder
The part of a model that generates output sequences from encoded representations of the input.

DFNN
Deep feed-forward neural network. A multi-layered deep learning model where each neuron connects to all neurons in the next layer.

Embedding
A numeric representation of data learned to capture semantic relationships, typically through a task-specific objective. Also called a learned representation.

Encoder
The part of a model that processes input data into learned representations or embeddings.

Fine-tuning
Adjusting a pretrained model on a specific task using a smaller, task-specific dataset.

gLM
Genomic language model. A model that has been pretrained in a self-supervised manner on tokenized genomic sequence data.

Hybrid
A model architecture that incorporates a transformer module along with other layer types (for example, convolutional or recurrent layers) to predict assay data.

Multi-head attention
A variation of attention where multiple sets of key, query and value vectors are projected to compute attention in parallel.

Pretraining
Training a model on a larger more general dataset, usually in a reconstruction pretext task to learn the underlying structure of the data, before fine-tuning on a specific task.

Pretext
A training task during self-supervised pretraining that helps the model learn a dataset's underlying structure, usually through reconstruction of the input data.

RNN
Recurrent neural network. A neural network that captures temporal or sequential dependencies between successive inputs with 'memory' mechanisms, including long short-term memory (LSTM) networks and gated recurrent units (GRUs).

Self-attention
A form of attention, where key, query and value vectors originate from the same input sequence.

Self-supervised learning
A method for training a model on unlabelled data using a pretext task, where the data's structure is exploited to generate labels the model must predict. Examples of this include MLM and ALM.

SSM
State space model. A class of models that can represent sequential data, traditionally used in control theory to model dynamic systems using state variables (the minimum number of variables that defines all possible states of a problem). These models, and recent modifications of them like the selective SSM, aim to be competitive with transformer performance, with a subquadratic cost in compute.

Token
The smallest unit of data a language model trains on. For gLMs, a token can be a DNA *k*-mer (where for *k*=6, a token is ATGATT), a single nucleotide (like 'A') or a BPE-summarized DNA token.

Transformer
A neural network composed of a series of stacked layers with data-dependent global context. Global context is achieved by having every token attend to every other token, and attention is determined dynamically (through the attention calculation) by the inputted data.

Transformer module
A subset of layers in the transformer that includes the attention mechanism, usually forming either the encoder or decoder of a transformer.

Zero-shot generalization
The ability of a model to perform well on tasks it was not explicitly trained on.

pretrained transformer (GPT)[4] is a decoder-only model pretrained with the autoregressive language modelling (ALM) task (see 'Pretraining' section). BERT models process tokens bidirectionally, considering both left-to-right and right-to-left contexts, while GPT models generate text sequentially, predicting the next token based on previous ones. In genomics, BERT predicts masked nucleotides using both upstream and downstream information, while GPT predicts subsequent nucleotides based on preceding ones.

BERT-style and GPT-style models excel in different tasks[3,6]. Encoder-only models, like BERT, are useful in cases where final predictions have high accuracy when based on only an embedding, or feature representation of the inputted sequence. This is often true for classification tasks. BERT-based models pretrained with MLM are useful for understanding genomic sequences where the overall context (upstream and downstream) is important, such as identifying genomic features or classifying sequences where directionality is not critical. The BERT framework has been successfully applied for many genomic modelling problems[30].

Decoder-only transformers pretrained with ALM, like GPT-3 and GPT-4[6,8], are appropriate for tasks that involve predicting sequences where directionality is important. This includes modelling anything co-transcriptional or co-translational, such as RNA splicing or protein folding, where these sequences are biologically synthesized in a unidirectional manner. Additionally, decoder-only models generally have superior zero-shot generalization[78].

### Training

Transformers can be trained in a supervised manner on labelled data for specific tasks, or pretrained in a semi-supervised manner on unlabelled data. Here, we discuss different methods of pretraining and fine-tuning transformer models, along with their motivation.

**Pretraining.** The attention mechanism in transformers is often credited with their success[79]. However, perhaps just as important, is the transformer's capacity to be pretrained. While pretraining is not exclusive to transformers and benefits other architectures[80,81], transformers are the most commonly pretrained architecture.

Unsupervised pretraining is the area of most research interest, particularly in biology, where available data are mostly not labelled[82]. The goal of unsupervised learning is for the model to uncover the underlying structure and signal of a dataset through pretext tasks (Box 1). These tasks, such as reconstruction (for example, predicting masked regions) or contrastive learning (for example, forcing semantically similar data points to be close in representation space), help the model learn meaningful data representations despite the lack of labels. In genomics, pretraining exposes the model to diverse sequences, enabling it to understand various sequence patterns, contextual relationships, and general nuances of genomic data.

MLM is the most common pretraining task in genomics[39,41,43,83] (Fig. 1). MLM involves randomly masking a subset of input tokens (usually 15%), with the model's task being to predict these masked tokens. In genomics, masking out nucleotide tokens within the genome and having the model 'fill in the gaps' allows the model to learn bidirectional relationships between nucleotides.

ALM is a training technique used in models such as GPT[4,6], where the task is to predict the next token in a sequence based on all previously observed tokens (Fig. 1). For genomic sequence modelling, this means predicting the next base (or group of bases) in a DNA sequence given the preceding bases[46]. Thus, ALM enables the model to directly generate new sequences. However, an artefact of this training is unidirectionality, as only previous tokens are considered when predicting the next one (Fig. 1), which contrasts with bidirectional models trained using MLM, which consider both past and future tokens. Thus, ALM and MLM offer different types of sequence understanding. At the time of writing this Review, the ALM pretext task has only been applied to the HyenaDNA[46] model and the Evo model[84], neither of which are transformer based.

**Fine-tuning.** After pretraining, the model undergoes fine-tuning, a form of transfer learning, on a smaller, task-specific (usually labelled) dataset. The pretrained parameters are adjusted so the model specializes for the target task. In some cases, earlier model layers may be frozen, with only later layers being fine-tuned. The expectation is that the model's weights will not change dramatically, as pretrained weights provide useful information for the downstream task. However, this assumption is often not explicitly evaluated. Most genomic models have not been closely examined to assess what is learned during pretraining versus fine-tuning[85]. Ideally, fine-tuning benefits from the broad representations learned in pretraining, allowing the model to specialize with less labeled data. For example, in genomics, MLM pretraining on the human genome could be followed by fine-tuning for classifying TATA promoters. This combination of pretraining and fine-tuning balances unsupervised learning from large unlabelled datasets with supervised learning from smaller, task-specific labelled datasets.

## SSMs and beyond

SSMs represent sequential data and were traditionally used in control theory to model dynamic systems using state variables, the minimum number of variables that define all possible states of a problem[86]. SSMs, at a given time $t$, map an input sequence $x_t$, to a hidden state or embedding space $h_t$, to make a prediction $y_t$. Like RNNs, the prediction $y_t$ depends on the hidden state $h_{t-1}$ of the last input or $x_{t-1}$. An RNN is actually a special case of an SSM (Fig. 2). The state space described by an SSM is similar to the embedding space, where embeddings of genomic deep learning models (whether they are RNNs, CNNs or transformers) can also describe the 'state' of an input sequence. While RNNs face slow training times but fast inference, transformers improve training time but struggle with slow inference due to their quadratic cost at longer sequences. Even for non-generative transformers with increased sequence lengths during training, the memory needed to store the attention weights can become prohibitive. Modified SSMs, or selective SSMs, aim to address these issues by enabling parallelizable training and fast inference. For a detailed breakdown of SSM equations, the differences between the original SSM[86] and selective SSMs[87], see Supplementary Information, appendix B.

Selective SSMs, of which Mamba is an example, are a class of models that adapt the original SSM for high-accuracy sequence-to-sequence prediction. Selective SSMs, unlike regular SSMs, selectively 'propagate' or 'forget' information by learning a different matrix for each input that determines how much of the current input enters the hidden state. This modification, along with some clever memory and storage access manipulations, allows selective SSMs to compete with attention's global context and data dependency[79], at a fraction of the cost of a full pair-wise attention calculation.

Models like the Hyena layer[79] also build off the idea of selective SSMs to balance global subquadratic context and data dependency to achieve very long contexts (Fig. 2). For a more detailed description of the Hyena layer, see Supplementary Information, appendix C.

So far, the Hyena layer, and the Mamba layer are the only new model types to be applied to genomic data[46,84,88]. However, if trends in NLP and protein modelling continue to be predictive of DNA modelling trends, these models are likely to continue to be adapted for DNA sequences. Additionally, the benefits of these models in terms of matching the performance of transformers, but scaling back on computational cost, could help increase context window size further for genomic sequence modelling.

## Hybrid models and gLMs

The main focus of this section will be to explore the transformer model and similar architectures (including HyenaDNA), but more general reviews of deep learning models applied to genomic data are available elsewhere[54–57,89–91] as is background on model interpretation in genomics[52,53].

### The transformer

In genomic modelling, transformers are used either as a subsequent module after initial layers or as a standalone model. In the hybrid approach, initial layers compress broad context into a shorter-sequence length but high-dimensional embedding space to mitigate attention's quadratic computational cost. When extended context windows are not needed, transformers with vanilla self-attention can operate standalone, processing input directly as transformer gLMs.

Therefore, we split transformer-based models for genomics into two classes: hybrids and transformer gLMs, and leave other gLMs as a separate category of genomic models[56] (Supplementary Fig. 1). Hybrid models incorporate transformers into more complex architectures, and are designed for tasks like predicting experimental assays (for example, cap analysis gene expression (CAGE) tracks and ChIP–seq). These models are specialized for high accuracy on assay-prediction tasks similar to CNN-based models like DeepBind[20,21]. Unlike gLMs, hybrid models typically do not pretrain, which is a characteristic feature of gLMs, making hybrid models more task-specific and less generalizable.

### Hybrid models: assay prediction

SATORI (self-attention-based model to detect regulatory element interactions) combines a convolutional layer and a self-attention mechanism to model the interactions between regulatory elements in DNA sequences. The model uses the sparsity in the attention matrix as a proxy for covariance, and was proposed as 'interpretable' due to direct analysis of the attention matrix. However, treating the attention mechanism within transformers as directly interpretable has drawbacks[92] (see 'Limitations' section).

Enformer[31], developed shortly after SATORI, predicts various genomic track signals (gene expression, DNA accessibility, histone modifications and transcription factor-binding). It combines convolutional and transformer blocks to capture long-range dependencies in sequences up to around 200 kb, with organism-specific prediction heads for human and mouse data. This architecture overcomes the context window limitations of convolution-only models like Basenji[32] and ExPecto[25]. However, recent work by Karollus et al.[93] scrutinized models like Enformer and Basenji2[94] (which also increased context window size over its predecessor), emphasizing the need for more and better-curated training data. They noted that despite Enformer's dramatically increased context window, it still encountered considerable limitations in predicting the impact of distal regulatory elements, such as enhancers. Enformer's predictive power remains comparably robust even with a severely restricted input window, suggesting its receptive field size is not the primary determinant of its success. This success can instead be attributed to innovations in model architecture such as combinations of various layer types, overall parameter number, or the quantity of data it was trained on. Karollus et al.[93] propose that models able to accurately account for distal regulators' contributions must train on datasets curated with an emphasis on long-range signals.

Borzoi[95] builds on the Enformer architecture by doubling the size of its context window. Borzoi further expands the number of experimental assay predictions compared to Enformer and introduces predictions of RNA-seq coverage. The model's main innovation is using an architecture styled after U-Net to upsample and increase prediction resolution following the transformer module. The convolutional blocks preceding the transformer summarize the longer sequence input and transform it into the same resolution as Enformer. This allows attention to be calculated with similar compute, thereby avoiding the quadratic memory complexity by decreasing sequence resolution. To make final predictions at a higher resolution, the information outputted by the transformer is upsampled using deconvolutional layers.

C.Origami is the hybrid transformer iteration on 3D genome prediction[42]. The model makes de novo predictions of cell-type-specific chromatin architecture from DNA sequence and genomic signals (CTCF-binding and ATAC-seq). Like its predecessor, Orca[72], C.Origami uses an encoder-decoder design, but adds an additional encoder for its multi-modal input types (one for DNA sequence and one for genomic signals). C.Origami leverages a transformer module to integrate the embeddings of the dual encoders before a task-specific decoder. The transformer module facilitates the multimodal integration and also enables long-range information exchange across these modalities, allowing C.Origami to outperform Orca[74] and Akita[36]. The C.Origami model enables in silico experiments that examine the impact of genetic perturbations on chromatin interactions and identifies a compendium of putative cell-type-specific regulators of 3D chromatin architecture.

### Transformers: gLMs

One of the earliest gLMs, DNABERT, adapted the original BERT model for genomic sequence modelling[3,30,39]. DNABERT is pretrained on overlapping *k*-mers using the MLM task, then fine-tuned for specific tasks. These include predicting proximal and core promoter regions and the presence of transcription factor binding sites with high accuracy. However, DNABERT's limited context window (512 tokens) restricts its ability to model long-range dependencies. To address this, a variation on the model, DNABERT-XL, splits longer sequences into smaller pieces, which are independently fed into the model. While this approach to increasing context-window size was able to identify between TATA and non-TATA promoters well, DNABERT did not demonstrate an end-to-end approach in modelling complex long-range dependencies, due to the limitations of the cost of attention.

The major advantage of DNABERT was the introduction of self-supervised pretraining for genomic data. As seen in Table 1, the model is pretrained extensively before any fine-tuning tasks. This alleviates the need for large amounts of labelled task-specific data later on, and highlights the power of gLMs.

Like DNABERT, the Nucleotide Transformer is pretrained in a self-supervised manner and adopts *k*-merization for tokenization[43]. This family of models varies in size, ranging from 500 million to 2.5 billion parameters, and unlike DNABERT, uses non-overlapping *k*-mers to avoid issues with token leakage from overlapping *k*-mers[39]. However, the non-overlapping *k*-mer approach has limitations, primarily that insertion or deletion of a single nucleotide base leads to dramatic changes in how a sequence is tokenized[39].

The smallest of the Nucleotide Transformer models is five times larger than DNABERT, and the authors' benchmarking experiments (predicting enhancers, promoters, TATA promoters, splice sites, and so on) show that increasing model size yields better performance. This is the same intuition that has led assay-prediction models like Enformer and its predecessors to increase their parameter sizes. Nucleotide Transformer also showed that training with intra-species variability (using multiple genomes of a single species, such as thousands of human genomes) did not perform as well as training with inter-species variability (their multi-species training regime). This is likely due to multi-species models capturing functional importance conserved across evolution, allowing them to generalize better even on human-based prediction tasks. The Nucleotide Transformer models strongly suggest that models leveraging evolutionarily diverse data in pretraining will improve capacity to capture functional relevance.

DNABERT-2[39] follows the multi-species training approach and uses BPE instead of *k*-mers for tokenization[70]. This approach bypasses the issues associated with overlapping and non-overlapping *k*-mer tokenization. BPE iteratively merges frequent pairs of nucleotides or segments within the genome instead of using a specific *k*-mer. This results in the model's vocabulary comprising a set of variable-length tokens representing the entire genome dataset across species (Box 1). BPE tokenization of DNA sequences has been observed to result in biologically significant tokens, with the longest tokens corresponding to elements of the genome known to be repetitive[96]. By contrast, *k*-mer tokenization treats all regions of the genome equally. Furthermore, the BPE method remains as computationally efficient as non-overlapping tokenization. The DNABERT-2 authors employ several other methods to improve computational efficiency over DNABERT, including the use of Flash Attention[97], among others[98,99]. These modifications allow DNABERT-2 to perform comparably to the Nucleotide Transformer models in several tasks, despite 21 times fewer parameters and significantly less computational cost.

Another recently introduced family of transformer-based DNA gLMs is GENA-LM[96]. Like DNABERT-2 it uses BPE tokenization, and like Nucleotide Transformer there are both human-only and multi-species models with varying parameters. However, a significant difference between GENA-LM and other models is the use of sparse attention to help mitigate the quadratic complexity in the context length of the transformer's attention mechanism. This results in GENA-LM models having increased maximum sequence length over other transformer-based gLMs, with a maximum tokenized sequence length of 4,096 tokens. The median token length after BPE tokenization is nine base pairs, thus GENA-LM models can process sequences of up to 36,000 base pairs.

### Beyond the transformer

While transformers are the dominant architecture for gLMs, alternatives can also match their performance and undergo similar pretraining

**Table 1 | A summary of the recently proposed deep learning models covered in this Review**

| Model name | Primary architecture (input, parameters) | Encoder/decoder | Tokenization/encoding | Date published | Pre- training? | Human or multispecies? | Trained to predict | Interpretability method |
|---|---|---|---|---|---|---|---|---|
| DNABERT[30] | Transformer (512 bp input, 110 million parameters) | Encoder-only | Overlapping *k*-mer | 1 February 2021 | MLM | Human | Based on fine-tuning can predict sequence classification task such as:<br>• Promoter recognition<br>• Transcription factor-binding site prediction<br>• Splice site prediction<br>• Functional genetic variants classification | Attention visualization method called DNABERT-viz |
| SATORI (Self Attention Based Model to detect Regulatory Element Interactions)[123] | CNN + (RNN) + transformer (up to 1 kb input, if RNN is included 674,270 parameters) | Encoder-only | One-hot encoding | 1 July 2021 | No | Multispecies | Transcription factor–transcription factor interactions | Uses filter–filter interactions from the self-attention layer to infer cooperativity between regulatory features, built as an 'interpretable' model |
| Enformer[31] | CNN + transformer (196 kb input, 228 million parameters + 16 million parameters for the human output head and 5 million for the mouse output head) | Encoder-only | One-hot encoding | 4 October 2021 | No | Multispecies | Experimental tracks for human and mouse from ENCODE | Attention matrices were inspected around specific regions of sequences |
| GPN (Genomic Pretrained Network)[83] | Modified-transformer architecture, replaces attention mechanism with dilated convolutions (512 bp input, over 65 million parameters) | Encoder-only | Single nucleotide token | 23 August 2022 | MLM | Multispecies (non-human) | Not trained in any supervised way, did perform unsupervised or zero-shot variant effect prediction in coding regions | Motif analysis of convolutions |
| C.Origami[42] | CNN + transformer (approx 200 kbp input, 10 million parameters) | Encoder-only | One-hot encoding | 9 January 2023 | No | Multispecies | De novo, cell-type-specific prediction of genome interaction contact maps | Visualization of all attention weights revealed that different attention heads attend to specific regions, using DNABERT-viz |
| Nucleotide Transformer[43] | Transformer (max input of 1,000 tokens, 2.5 billion parameters) | Encoder-only | Non-overlapping *k*-merization | 15 January 2023 | MLM | Multispecies | Based on fine-tuning can predict sequence classification tasks such as:<br>• Epigenetic marks prediction<br>• Promoter sequence prediction<br>Enhancer sequence prediction<br>• Splice site prediction | Analysed attention maps gathered from the pre-trained models and used BERTology paper to guide analysis of how attention is distributed to different genomic elements across different heads |
| GENA-LM[96] | Transformer (max 4.5 kb input and 336 million parameters with full attention, 36 kb input and 110 million parameters with sparse attention) | Encoder-only | BPE | 13 June 2023 | MLM | Multispecies | Based on fine-tuning can predict sequence classification tasks such as:<br>• Promoter prediction<br>• Splice site prediction<br>• *Drosophila* enhancers prediction<br>• Chromatin profiling<br>• Polyadenylation sites prediction | None |

**Table 1 (continued) | A summary of the recently proposed deep learning models covered in this Review**

| Model name | Primary architecture (input, parameters) | Encoder/decoder | Tokenization/encoding | Date published | Pre-training? | Human or multispecies? | Trained to predict | Interpretability method |
|---|---|---|---|---|---|---|---|---|
| DNABERT-2[39] | Transformer (128 bp input, 117 million parameters) | Encoder-only | BPE | 26 June 2023 | MLM | Multispecies | Based on fine-tuning can predict sequence classification tasks such as:<br>• Core promoter prediction<br>• Proximal promoter prediction<br>• Splice site prediction<br>• Transcription factor-binding site prediction for human and mouse<br>• Epigenetic marks prediction | None |
| HyenaDNA[46] | Hyena (up to 1 million token input, up to 6.6 million parameters) | Decoder-only | Single nucleotide token | 27 June 2023 | ALM | Human | Based on fine-tuning can predict sequence classification tasks such as:<br>• Epigenetic marks prediction Promoter sequence prediction<br>• Enhancer sequence prediction<br>• Splice site prediction | None |
| Borzoi[95] | CNN + transformer (524 kb input, 186 million parameters) | Encoder-only | One-hot encoding | 1 September 2023 | No | Multispecies | More experimental tracks for human and mouse from ENCODE as well as RNA-seq tracks | Attention map exploration across TSS regions, exon boundaries, and polyadenylation signals for example regions |
| Evo[84] | StripedHyena | Decoder-only | Single nucleotide token | 27 February 2024 | ALM | Multispecies (prokaryotic genomes) | Zero shot evaluated on:<br>• Predicting mutational effects on non-coding RNA function<br>• Predicting gene expression from promoter–RNA binding site pairs<br>• Generating mobile genetic elements<br>• Gene essentiality prediction<br>Fine-tuned to generate protein–RNA complexes | None |

regimes (for example, MLM or ALM pretexts[79,80]). It remains unclear whether the success of the transformer model lies in an artefact of the architecture, like the attention mechanism, or whether this mechanism simply allowed these models to scale up more quickly than their counterparts. It could be that the pretraining capabilities of the transformer, which are not restricted to this architecture, contribute the most to its success. If this is the case, the transformer model could be replaced by another model in NLP, proteomics, and genomics[79,100,101].

The genomic pretrained network, or GPN[83], copies the exact architecture of a transformer encoder module but replaces the attention mechanism with a convolution operation across the sequence. The idea behind this came from recent work that showed that pretrained CNNs are competitive with transformers in NLP[80], and protein modelling[81]. The GPN model leverages the MLM pretext task in pretraining and is trained solely on individual nucleotides, rather than using BPE or any *k*-merization strategy. The genomes used in pretraining consisted of eight Brassicales reference genome assemblies from National Center for Biotechnology Information (NCBI) Genome. Instead of sampling the whole genome equally in 512 bp windows during pretraining, the authors took the union of exons (with a small intronic flank), promoters (1,000 base pairs upstream of the TSS (transcription start site)), and a complementary number of random windows from the whole genome. While the authors state this may have improved performance, they do not show any experiments to validate this claim.

The authors demonstrate that GPN learns non-coding variant effects from unsupervised pretraining solely on genomic DNA sequences, outperforming supervised deep learning models such as DeepSEA[21].

Another non-transformer gLM, HyenaDNA[46], achieves a context size of 1 million nucleotides, 500× larger than the largest of the gLMs utilizing full pairwise attention, the Nucleotide Transformer[43]. Instead of relying on the quadratic-bound attention mechanism, which compares each pair of points in a sequence, the authors of the original Hyena paper designed a subquadratic-time layer. HyenaDNA is based on the structure of the decoder-only transformer architecture, replacing the attention mechanism directly with the Hyena operator. HyenaDNA is trained generatively using the ALM pretext task. The HyenaDNA model was only trained on one reference human genome, providing an obvious direction for future work. The model boasts state-of-the-art performance on all eight datasets from GenomicBenchmarks[102].

The recently proposed Evo model[84] is trained on whole prokaryotic genomes. Evo uses the StripedHyena architecture, a hybridization of attention layers and Hyena layers. The authors are the first to provide scaling laws experiments motivating the use of the StripedHyena architecture as opposed to Mamba, Hyena, or a set of efficient Transformer variations. Scaling law analysis aims to determine the relationship between the size of pretraining datasets, the model architecture used, and performance metrics. Scaling laws of language models in NLP show increasing training dataset size and model size results in proportional increases in performance[103]. This is a strong motivation for working with these models in NLP, and suggests the training task of these models learns the underlying structure of the data. However, whether gLMs and protein language models adhere to this kind of a scaling law has yet to be robustly demonstrated[104,105]. Evo is capable of predicting whether a mutation in a non-coding RNA (RNA that does not encode for proteins but instead might regulate activity in the cell) results in a drop in fitness as measured by non-coding RNA deep mutational scanning (ncRNA DMS) experiments[84]. Evo shows an ability to predict gene expression given promoter–RNA binding site sequence pairs, and can predict gene essentiality, as mutations in essential genes result in larger negative log likelihood changes than non-essential genes. Overall, the Evo authors showed that Evo has strong performance on a variety of tasks, but only tested on prokaryotic data and compared to models pretrained on both eukaryotic and prokaryotic DNA. Additionally, the prokaryotic genome is substantially less complex than the eukaryotic genome, making the performance of Evo at genome-scale, and the efficacy of Evo's pretraining, not necessarily translatable to eukaryotic DNA.

## A comparison

Hybrid models with transformer elements are not always evaluated on the same tasks as transformer gLMs or alternative gLM architectures. Hybrid models are supervised, predicting assay data, while gLMs are typically assessed on self-supervised representations or after supervised fine-tuning. When gLMs are evaluated on their pretrained embeddings, they often underperform compared to supervised models[59,73]. This performance gap can be partly explained by the different training objectives: gLMs aim to provide general representations of DNA sequences for diverse downstream tasks, whereas hybrid models target high accuracy for specific tasks. The disparity may also result from ineffective pretraining task design for gLMs, as many models do not report zero-shot performance relative to expert supervised methods. Given that gLMs typically have many more parameters and require more data to pretrain, if a smaller, less computationally intensive model can outperform a gLM, there is little incentive to train a gLM. Therefore, a gLM's pretrained or zero-shot performance should be comparable to hybrid models, and their fine-tuned predictions should outperform hybrid models across a broader range of downstream tasks. Thus, we recommend comparing gLMs in zero-shot contexts, using supervised hybrid models as a baseline.

Many of the models explored here report strong performance on a single task (hybrid model), or curated series of tasks (gLM), compared against a subset of similar models. Hybrid models can show a measurable improvement over prior methods on the specific task they were trained on. However, comparisons between gLMs are less straightforward. The pretrained embeddings of gLMs may capture different information within the genome[59], making it difficult to assess which representation is more meaningful. Additionally, fine-tuned performance can vary in task difficulty, biological relevance, and dataset application (for example, genome segmentation is more challenging than sequence classification). To standardize gLM comparisons, several benchmarking task collections have been proposed: GenomicBenchmarks[102], Genome Understanding Evaluation (GUE)[39], and the BEND Benchmarking paper[59]. GenomicBenchmarks and GUE evaluate fine-tuned performance, while BEND assesses zero and few-shot performance, either directly through gLM embeddings or by training a shallow CNN on top of them. A well-pretrained gLM should embed similar sequences proximally and dissimilar ones distally in the embedding space. For variant effect prediction, cosine distances between reference and variant sequence embeddings can indicate functional differences. For tasks like enhancer region annotation, evaluating zero-shot embeddings is less straightforward. This is why shallow CNNs are trained on top of gLM embeddings to predict specific annotations from the frozen gLM pretrained weights in tasks like gene finding, chromatin accessibility, histone modification, and more. The BEND paper found that current gLMs show promise but do not consistently outperform supervised baselines, aligning with results from other studies on zero-shot performance[103]. The Nucleotide Transformer Multi-Species and original DNABERT models had the best zero- and few-shot embeddings, with the Nucleotide Transformer excelling in gene finding and enhancer annotation, and DNABERT in chromatin accessibility and histone modification prediction. These models approached the performance of hybrid models like Enformer and Basset[33]. However, the design and curation of benchmarking tasks and datasets for gLMs remain an area for future research[106].

## Limitations

As previous review papers have focused on the limitations inherent in applying deep learning models to genomic data, including cell-type-specificity debates and training data limitations[55], we focus on the limitations inherent to applying novel architectures to genomics, such as the transformer and SSM-like models.

### Long-range interactions

Hybrid models remain unable to capture long-range dependencies within the genome. This is despite most improvements in assay prediction models increasing context window size to better model these long-range dependencies. However, Karollus et al.[93] suggest a larger receptive field might not be the key factor driving success in more recent models, including Enformer[31]. While trends show models with increased context window size have had higher accuracy in predicting experimental assays, this does not necessarily suggest context window size is the driving force behind improved predictive capacity. Karollus et al. showed that significantly reducing the input size given to the Enformer model has a minimal effect on its performance, suggesting that transformer-based models, like Enformer, may achieve state-of-the-art performance simply due to the addition of the transformer module, or through increased parameter size. Enformer's successor, Borzoi[95] appears to better integrate long-distance information, as measured by ranking distal regulatory elements for their gene-specific enhancer activity (data from CRISPR screens). However, experiments similar to those performed on Enformer have not yet evaluated Borzoi's performance after a significant reduction of the input size. Likely, context window size has to be combined with better-curated datasets, ones that are curated to capture the effects of distal expression quantitative trait loci (eQTLs), distal enhancers, and distal repressors[93].

If such a dataset can be curated, then the context-window size will be the driving factor for modelling long-range dependencies. Transformer-based gLMs that do not use downsampling techniques to reduce dimensionality before calculating self-attention, or do not apply a more efficiently implemented attention method[97], will be limited by their context window. Even the Nucleotide Transformer[43], with 2.5 billion parameters, could only extend the context window to a maximum of 1,000 tokens, which remains 1,000 times smaller than the HyenaDNA[46] context window.

Other gLMs that forgo the use of the quadratic attention mechanism have the potential to better capture long-range interactions in the genome. As long as the attention mechanism itself is not the driving force of success in these models, which does not appear to be the case given the recent success of HyenaDNA[46], Evo[84], and GPN[83], this is a strong avenue for potential research.

### Cell type specificity

Hybrid models, which aim to predict experimental assays, are usually trained with either ENCODE[107] data from multiple cell lines, or fine-tuned non-specifically across cell types. The tasks they are evaluated on make predictions ignoring the inter-cell-type variability within genomic annotations. While this allows the models to leverage huge amounts of data available on ENCODE[66] and Roadmap[67] by pooling cell types together, many findings in the field of genomics and the increased use of single-cell specific sequencing show there is cell-specific heterogeneity in regulatory annotations like chromatin accessibility, chromatin conformation, gene expression, and transcription factor binding[26,108,109]. Some transformer models have recently been proposed to mitigate the bias of cross-cell-line gene expression prediction using transfer learning, but this approach has yet to be commonly adopted[110]. A future area of research could be moving away from hybrid models for predicting cell-type-specific experimental assays. Instead, the prompting capabilities of generative gLMs could be leveraged to create cell-type-specific contexts for predictions.

### Data privacy

As mentioned, most of the models discussed in this Review are trained on public datasets like ENCODE[107] and Roadmap[111]. However, genetic information is unique, so there is a risk of re-identifying individuals even from anonymized genomic datasets. As more data continue to be integrated into these models, and these models are potentially leveraged for use on private datasets including in clinical settings, there is an increasing need for developing and implementing more robust de-identification techniques and privacy-preserving algorithms, such as differential privacy[112] and federated learning[113].

### Interpretability

A key limitation of applying deep learning models to genomic data is their black-box nature, which becomes more pronounced as models grow in size and complexity. This is especially problematic in the context of genomics where the underlying 'language' of the genome is unclear to us[53]. While previous work has extensively explored deep learning interpretability in genomics[18,52,55], this section focuses specifically on interpreting transformer models and similar architectures.

Attention scores have been proposed as a solution to the interpretability problem in genomics[30,31,114]. Models like Enformer, DNABERT, and C.Origami have used attention scores to demonstrate their ability to capture biological signals. However, studies outside genomics show that attention scores are not inherently interpretable[115,116]. Reporting only raw attention scores misses key information by focusing solely on the inner product of queries and keys, and ignoring the full computation of queries, keys, and values. Aggregating attention scores across layers or heads, as is commonly done, also neglects the complexities of how attention passes through the model, including through add-and-norm layers and skip-connections[77,116]. Models that consider the mean of attention heads across multi-headed attention also dilute the information captured by the model as different heads contribute differently in each layer, and not all heads contribute equally[115–117].

To address these issues, methods like attention flow and attention rollout[118] have been used to interpret transformers. Layer-wise relevance propagation (LRP), widely applied to interpret CNN-based models[92], has been adapted for transformers as well to explain multi-headed attention and highlight 'redundant' heads[103]. A recent adaptation of LRP for transformers, incorporating attention scores across multiple heads[102], has outperformed other attribution methods like classic LRP[119], partial LRP[103], rollout[118] and Grad-CAM[120] on transformers.

Ultimately, the interpretation of transformers in genomics beyond attention-score visualization has been limited. We acknowledge that while previous papers provide some insight into transformers' interpretability specifically in genomics[114], they have not compared methods beyond attention scores or acknowledged the limitations of this method.

Beyond transformers, other gLMs also require interpretability assessments. For example, GPN[83] reported motifs captured in the convolutional blocks of the architecture as a metric of interpretability, similar to previous CNN-based models for genomics[20,21,23]. These motifs were then compared to experimentally determined and validated motifs and reported as position weight matrices (PWMs) or logos. HyenaDNA, despite its success, lacks a dedicated interpretability section, likely due to the novelty of the Hyena layer and the absence of established interpretability methods for this architecture.

Model-agnostic methods like SHAP (Shapley additive explanations)[121], and WeightedSHAP[122], offer alternative interpretability approaches for models with or without attention. SHAP quantifies the contribution of each feature to a model's prediction on a specific data point. While SHAP uniformly averages the contribution of each feature on the model's prediction across different subsets of the total number of features, WeightedSHAP allows for using weights to emphasize more important feature contributions.

All transformer-based hybrid models covered in this review have reported attention scores for their interpretability metrics, with the SATORI[123] authors even claiming the model was inherently interpretable due to the attention mechanism alone. While these models have not employed more sophisticated methods for model interpretability, attention mechanisms themselves may encourage the exploration of

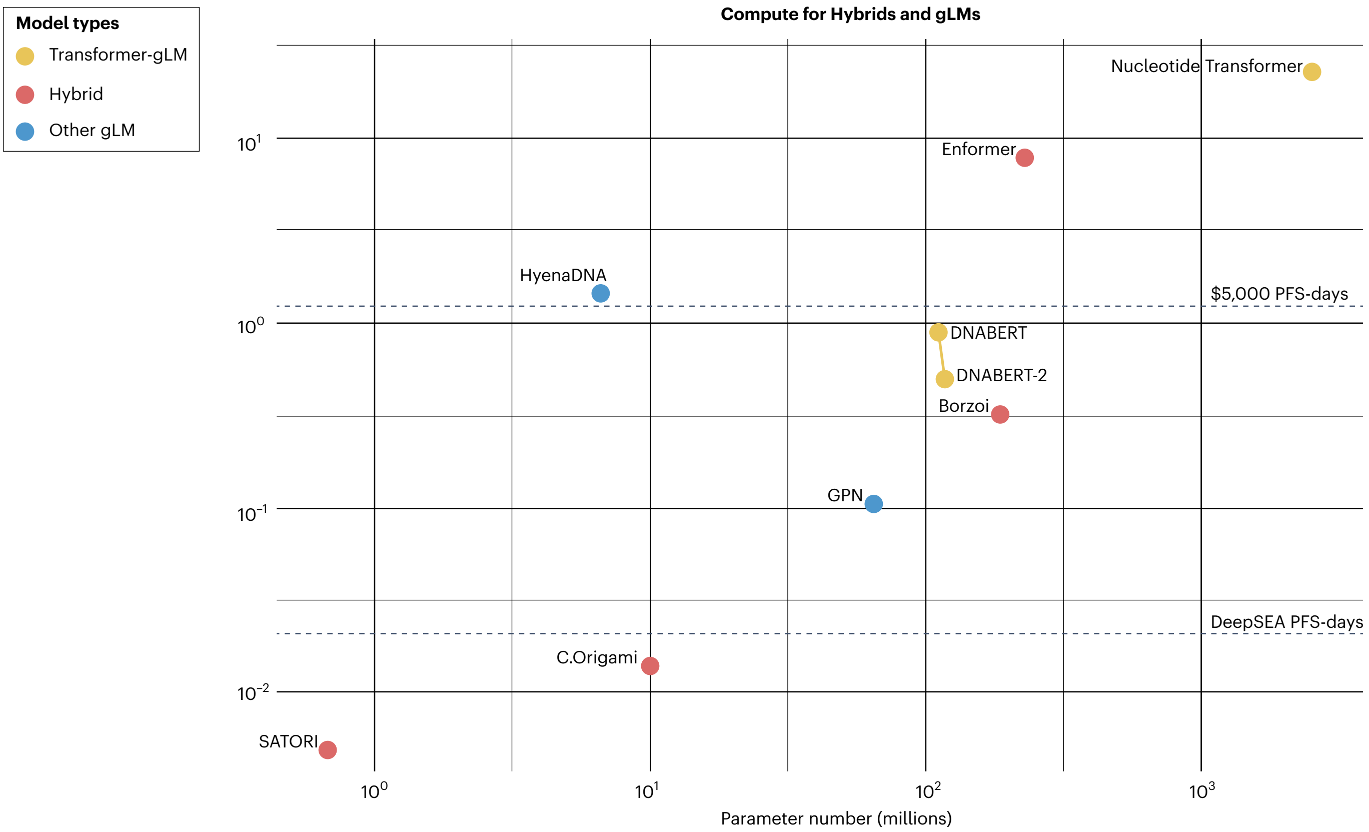

**Fig. 3 | The total amount of compute, in PFS-days used to train the various models discussed in the Review (all of the models for which parameter number, training time, and GPU usage were available).** A petaflops-day (PFS-day) consists of performing $10^{15}$ neural net operations per second for one day, or a total of ~$10^{20}$ operations. It is a compute-time measurement proposed by OpenAI to compare across model architectures, which can be thought of similarly to kW·h for energy. For context, we calculate the PFS-days for the original DeepSEA[21] model and the equivalent PFS-days that can be purchased to train a model with US$5,000, renting eight A100 GPUs at US$8.80 per hour. Calculations for PFS-days can be found in the 'Limitations' section.

model interpretability. In contrast, only one of the non-transformer gLM models (GPN) reported some kind of model interpretability. We suggest researchers working with transformer-based models consider incorporating methods like classic LRP[119], partial LRP[103], rollout[118], SHAP[121], or weightedSHAP[122] to discover biological motifs and tokens of interest these models attend to. Furthermore, we encourage the development of interpretability methods for the novel architectures proposed to this field beyond transformers, like HyenaDNA. We expect the use of interpretability methods for attention-based mechanisms in genomics to increase, given their increasing usage trend among transformers applied to other fields[102]. Similarly, we expect models that aim to outperform the transformer in genomics to develop interpretability methods for their novel architecture design. For this, we hope to see both perturbation-based interpretation methods[124,125], as well as interpretations of internal model representations[126].

## Compute requirements

Perhaps the most noteworthy limitation in the usage of gLMs is the computational cost of training them. Their pretraining often necessitates high-performance computing resources, which may not be readily accessible or affordable for all research teams. However, this may be changing with the introduction of novel architectures like Mamba[87].

Using a compute calculation derived by OpenAI, we can calculate the petaflops/days for each of the models discussed in the review (except for GENA-LM[96] and Evo[84], for which there was insufficient data) to compare across different model architectures and parameter count. The equation for PFS-days using GPU time is:

$$\text{PFS-days} = \text{number of GPUs} \times (\text{petaflops/hardware}) \\ \times \text{days trained} \times \text{estimated utilization}$$

Where OpenAI assumes a 33% utilization for GPUs. Looking at Fig. 3 we can see that most models proposed using transformers or similar architectures for genomics can be trained with USD 5,000 on eight A100 GPUs. However, of the largest and arguably most discussed models (DNABERT, Enformer, Nucleotide Transformer and HyenaDNA) only DNABERT could be trained within this budget. Note that DNABERT was trained before 2021, when GPU access was more limited and expensive.

Transformer-based models require substantial compute and memory due to their multi-layer, multi-headed attention mechanisms. Even though HyenaDNA is more efficient, it still demands more compute than most academic labs can afford (setting USD 5,000 as a baseline). Additionally, the long training times for these models slow down research progress and hinder model iteration (Fig. 3).

## Pretraining task design

Pretraining is a powerful and architecture-agnostic tool for gLMs, but its effectiveness depends on the quality of the pretraining task. Ideally, it enables models to capture universal data patterns[127], but if poorly designed, it becomes an unnecessary computational cost[128]. To maximize benefits, pretraining tasks should be biologically relevant for later applications.

Pretraining tasks from NLP have generally been applied to genomics with little consideration for their biological relevance and the inherent differences between DNA sequences and natural language. These tasks may not capture meaningful biological signals or align with the model's ultimate goal. A better approach may be to design a pretraining task based on biological insights. One recently proposed technique for biologically informed unsupervised pretraining is phylogenetic augmentation[129]. In this task, evolutionarily related sequences are treated as augmentations, or different views, of original sequence data. The goal of phylogenetic augmentation pretraining is to learn representations that maximize the mutual information between evolutionarily related sequences and their conserved function.

Ultimately, the gLMs discussed in this Review have applied ALM or MLM pretext tasks for pretraining, with minimal adaptations for biological context. The effectiveness of these models' pretraining has been largely unexplored. However, initial investigations on the performance of pretrained models, or investigating pretraining regimes for genome gLMs, have not been favourable[85,128]. While this remains a limitation of current gLMs, it also provides a promising direction for future research.

## Future directions

The success of deep learning models for the genome, specifically with the increasing use of gLMs, and the limitations they are currently bound by, provide a complex outlook for the future. Of the notable trends within deep learning genomic modelling, one of the most prominent is the potential of unsupervised pretraining regimes, specifically multi-species pretraining. This approach could capture evolutionarily conserved data in the genome and better model its underlying grammar. As the success of many of these models is considered to be contingent upon their expensive and time-consuming pretraining regimes, it is important to understand what exactly the models are learning in pretraining versus fine-tuning. Recent work[75] investigating BERT model behaviour in genomics shows that *k*-mer embeddings from random data have comparable performance on downstream tasks to *k*-mer embeddings pretrained on real biological sequences. This tells us that while pretraining and unsupervised learning could increase the power of genomic models, the pretraining tasks for these models must be well designed and validated to prove true genomic grammar is being captured. We suggest further experiments are conducted on these models to compare pretrained and fine-tuned versus randomly initialized and fine-tuned embedding spaces. Additionally, we question whether pretraining on entire genomes is the best way to leverage the power of pretraining[83,130]. Repetitive non-coding sections of DNA make up nearly half of the human genome[131], and could potentially overpower the ability of these models to learn more relevant signals from relatively less common but more important sequence regions.

While this Review focuses on DNA-based genome language models, single-cell RNA-seq language models follow many similar principles[40,41,132–136]. However, these models come with their own strengths and limitations, particularly in handling the non-sequential nature of transcriptomic data. An in-depth exploration of these models is out of scope, but we expect many advancements in architectural and pretraining task designs to be aligned between DNA and scRNAseq language models.

New gLM architectures like the Hyena layer are emerging, which do not rely on attention mechanisms yet still support pretraining. These models may offer better scalability for genomic data compared to traditional transformers[46,83]. The attention mechanism's quadratic complexity is a bottleneck for genomic modelling, especially if increasing the context window size is crucial. This opens the door for next-generation models like HyenaDNA, Mamba-based models and StripedHyena models, to potentially outperform transformers.

Efforts are underway to improve the scalability of attention mechanisms, such as introducing sliding windows[137], enhancing efficiency[44,138], and improving long-range interaction modelling in NLP[48]. Despite these advancements, transformers still lag in context window size compared to models with the Hyena layer, or models that use Selective SSMs. The hybridization of attention and other SSM-like layers in the future provides even more dials to adjust in model design.

As multi-omic data becomes more widely available, interest in training gLMs on sequence tokens alone may diminish. Large models capable of integrating multi-modal data could unify genomic, transcriptomic, proteomic and epigenomic data, offering a more holistic view of biological systems. If trends in deep learning models for proteins predict future trends in genomic sequence modelling[139], the next training paradigm in genomics will be diffusion[140]. If pretraining with evolutionarily varied data is important for modelling genomic information[129], perhaps diffusion, approximations of which have been widely applied in evolutionary theory[141–143], is the obvious choice. DNA-based diffusion models have already started to show promise in modelling regulatory elements in the genome[144].

The future of powerful and interpretable deep learning models in genomics is one of personalized medicine, understanding evolutionary dynamics, drug discovery, synthetic biology, and more. We are at an exciting time for the field, and we hope to see an increase in the use of multi-species pretraining, and more biologically motivated and downstream-task-aligned approaches to designing pretext tasks. Furthermore, we hope to see greater emphasis on the zero-shot performance of proposed gLMs. Research aligned with these tenets will lead to the greatest success in modelling the genome, whether it be through transformer models, Hyena layers, Mamba layers, or diffusion-style training. Additionally, we believe deep learning models will only succeed in modelling genomic data if strides continue to be made toward their interpretation within genomic contexts[116,126].

## References

1. Nichol, A. et al. GLIDE: towards photorealistic image generation and editing with text-guided diffusion models. Preprint at https://doi.org/10.48550/arXiv.2112.10741 (2021).
2. Ramesh, A., Dhariwal, P., Nichol, A., Chu, C. & Chen, M. Hierarchical text-conditional image generation with clip latents. Preprint at https://doi.org/10.48550/arXiv.2204.06125 (2022).
3. Devlin, J., Chang, M.-W., Lee, K. & Toutanova, K. BERT: pre-training of deep bidirectional transformers for language understanding. Preprint at https://doi.org/10.48550/arXiv.1810.04805 (2018).
4. Radford, A. et al. Language models are unsupervised multitask learners. *OpenAI Blog* **1**, 9 (2019).
5. Yang, Z. et al. XLNet: generalized autoregressive pretraining for language understanding. In *Proc. 33rd International Conference Neural Information Prcoessing Systems* **517**, 5753–5763 (2019).
6. Brown, T. B. et al. Language models are few-shot learners. Preprint at https://doi.org/10.48550/arXiv.2005.14165 (2020).
7. Jumper, J. et al. Highly accurate protein structure prediction with alphafold. *Nature* **596**, 583–589 (2021).
8. *GPT-4 Technical Report* (OpenAI, 2023).
9. Warr, A. et al. Exome sequencing: current and future perspectives. *G3 Genes Genomes Genet.* **5**, 1543–1550 (2015).
10. Ng, P. C. & Kirkness, E. F. Whole genome sequencing. *Genet. Var.* **628**, 215–226 (2010).
11. Buenrostro, J. D., Wu, B., Chang, H. Y. & Greenleaf, W. J. ATAC-seq: a method for assaying chromatin accessibility genome-wide. *Curr. Protoc. Mol. Biol.* **109**, 21–29 (2015).
12. Park, P. J. ChIP–seq: advantages and challenges of a maturing technology. *Nat. Rev. Genet.* **10**, 669–680 (2009).
13. Vaisvila, R. et al. Enzymatic methyl sequencing detects DNA methylation at single-base resolution from picograms of DNA. *Genome Res.* **31**, 1280–1289 (2021).
14. Wang, Z., Gerstein, M. & Snyder, M. RNA-Seq: a revolutionary tool for transcriptomics. *Nat. Rev. Genet.* **10**, 57–63 (2009).

15. Haque, A., Engel, J., Teichmann, S. A. & Lönnberg, T. A practical guide to single-cell RNA-sequencing for biomedical research and clinical applications. *Genome Med.* **9**, 75 (2017).
16. Lieberman-Aiden, E. et al. Comprehensive mapping of long-range interactions reveals folding principles of the human genome. *Science* **326**, 289–293 (2009).
17. Ecker, J. R. et al. ENCODE explained. *Nature* **489**, 52–54 (2012).
18. Zou, J. et al. A primer on deep learning in genomics. *Nat. Genet.* **51**, 12–18 (2019).
19. Quang, D., Chen, Y. & Xie, X. DANN: a deep learning approach for annotating the pathogenicity of genetic variants. *Bioinforma. Oxf. Engl*. **31**, 761–763 (2014).
20. Alipanahi, B., Delong, A., Weirauch, M. T. & Frey, B. J. Predicting the sequence specificities of DNA- and RNA-binding proteins by deep learning. *Nat. Biotechnol*. **33**, 831–838 (2015).
21. Zhou, J. & Troyanskaya, O. G. Predicting effects of noncoding variants with deep learning–based sequence model. *Nat. Methods* **12**, 931–934 (2015).
22. Pei, G., Hu, R., Jia, P. & Zhao, Z. DeepFun: a deep learning sequence-based model to decipher non-coding variant effect in a tissue- and cell type-specific manner. *Nucleic Acids Res.* **49**, W131–W139 (2021).
23. Hassanzadeh, H. R. & Wang, M. DeeperBind: enhancing prediction of sequence specificities of DNA binding proteins. In *Proc. IEEE International Conference on Bioinformatics and Biomedicine* Vol. 2016, 178–183 (2016).
24. Trieu, T., Martinez-Fundichely, A. & Khurana, E. DeepMILO: a deep learning approach to predict the impact of non-coding sequence variants on 3D chromatin structure. *Genome Biol*. **21**, 79 (2020).
25. Zhou, J. et al. Deep learning sequence-based ab initio prediction of variant effects on expression and disease risk. *Nat. Genet.* **50**, 1171–1179 (2018).
26. Wang, M., Tai, C., E, W. & Wei, L. Define: Deep convolutional neural networks accurately quantify intensities of transcription factor-DNA binding and facilitate evaluation of functional non-coding variants. *Nucleic Acids Res.* **46**, e69 (2018).
27. He, Z., Liu, L., Wang, K. & Ionita-Laza, I. A semi-supervised approach for predicting cell-type specific functional consequences of non-coding variation using MPRAs. *Nat. Commun.* **9**, 5199 (2018).
28. Wells, A. et al. Ranking of non-coding pathogenic variants and putative essential regions of the human genome. *Nat. Commun.* **10**, 5241 (2019).
29. Quang, D. & Xie, X. DanQ: a hybrid convolutional and recurrent deep neural network for quantifying the function of DNA sequences. *Nucleic Acids Res.* **44**, e107 (2016).
30. Ji, Y., Zhou, Z., Liu, H. & Davuluri, R. V. DNABERT: pre-trained bidirectional encoder representations from transformers model for DNA-language in genome. *Bioinformatics* **37**, 2112–2120 (2021).
31. Avsec, Z. et al. Effective gene expression prediction from sequence by integrating long-range interactions. *Nat. Methods* **18**, 1196–1203 (2021).
32. Kelley, D. R. et al. Sequential regulatory activity prediction across chromosomes with convolutional neural networks. *Genome Res.* **28**, 739–750 (2018).
33. Kelley, D. R., Snoek, J. & Rinn, J. L. Basset: learning the regulatory code of the accessible genome with deep convolutional neural networks. *Genome Res.* **26**, 990–999 (2016).
34. Tasaki, S., Gaiteri, C., Mostafavi, S. & Wang, Y. Deep learning decodes the principles of differential gene expression. *Nat. Mach. Intell.* **2**, 376–386 (2020).
35. Xiong, H. Y. et al. The human splicing code reveals new insights into the genetic determinants of disease. *Science* **347**, 1254806 (2015).
36. Fudenberg, G., Kelley, D. R. & Pollard, K. S. Predicting 3D genome folding from DNA sequence with Akita. *Nat. Methods* **17**, 1111–1117 (2020).
37. Avsec, Z. et al. Base-resolution models of transcription-factor binding reveal soft motif syntax. *Nat. Genet*. **53**, 354–366 (2021).
38. Vitsios, D., Dhindsa, R. S., Middleton, L., Gussow, A. B. & Petrovski, S. Prioritizing non-coding regions based on human genomic constraint and sequence context with deep learning. *Nat. Commun*. **12**, 1504 (2021).
39. Zhou, Z. et al. DNABERT-2: efficient foundation model and benchmark for multi-species genome. Preprint at https://doi.org/10.48550/arXiv.2306.15006 (2023).
40. Cui, H., Wang, C., Maan, H. & Wang, B. scGPT: towards building a foundation model for single-cell multi-omics using generative AI. *Nat. Methods* **21**, 1470–1480 (2024).
41. Theodoris, C. V. et al. Transfer learning enables predictions in network biology. *Nature* **618**, 616–624 (2023).
42. Tan, J. et al. Cell-type-specific prediction of 3D chromatin organization enables high-throughput in silico genetic screening. *Nat. Biotechnol*. **41**, 1140–1150 (2023).
43. Dalla-Torre, H. et al. Nucleotide Transformer: building and evaluating robust foundation models for human genomics. *Nat. Methods* **22**, 287–297 (2025).
44. Bolya, D., Fu, C.-Y., Dai, X., Zhang, P. & Hoffman, J. Hydra Attention: efficient attention with many heads. Preprint at https://doi.org/10.48550/arXiv.2209.07484 (2022).
45. Ma, X. et al. Mega: moving average equipped gated attention. Preprint at https://doi.org/10.48550/arXiv.2209.10655 (2022).
46. Nguyen, E. et al. HyenaDNA: long-range genomic sequence modeling at single nucleotide resolution. Preprint at https://doi.org/10.48550/arXiv.2306.15794 (2023).
47. Jones, W., Alasoo, K., Fishman, D. & Parts, L. Computational biology: deep learning. *Emerg. Top. Life Sci.* **1**, 257–274 (2017).
48. Ching, T. et al. Opportunities and obstacles for deep learning in biology and medicine. *J. R. Soc. Interface* **15**, 20170387 (2018).
49. Min, S., Lee, B. & Yoon, S. Deep learning in bioinformatics. *Brief. Bioinform*. **18**, 851–869 (2017).
50. Richards, B. A. et al. A deep learning framework for neuroscience. *Nat. Neurosci*. **22**, 1761–1770 (2019).
51. Wainberg, M., Merico, D., Delong, A. & Frey, B. J. Deep learning in biomedicine. *Nat. Biotechnol*. **36**, 829–838 (2018).
52. Novakovsky, G., Dexter, N., Libbrecht, M. W., Wasserman, W. W. & Mostafavi, S. Obtaining genetics insights from deep learning via explainable artificial intelligence. *Nat. Rev. Genet*. **24**, 125–137 (2023).
53. Talukder, A., Barham, C., Li, X. & Hu, H. Interpretation of deep learning in genomics and epigenomics. *Brief. Bioinform*. **22**, bbaa177 (2021).
54. Li, Z. et al. Applications of deep learning in understanding gene regulation. *Cell Rep. Methods* **3**, 100384 (2023).
55. Eraslan, G., Avsec, Z., Gagneur, J. & Theis, F. J. Deep learning: new computational modelling techniques for genomics. *Nat. Rev. Genet*. **20**, 389–403 (2019).
56. Routhier, E. & Mozziconacci, J. Genomics enters the deep learning era. *PeerJ* **10**, e13613 (2022).
57. Sapoval, N. et al. Current progress and open challenges for applying deep learning across the biosciences. *Nat. Commun.* **13**, 1728 (2022).
58. Muse, S. *Introduction to Biomedical Engineering* 2nd edn (eds Enderle, J. D. et al.) Ch. 13, 799–831 (2005).
59. Marin, F. I. et al. BEND: benchmarking DNA language models on biologically meaningful tasks. Preprint at https://doi.org/10.48550/arXiv.2311.12570 (2024).

60. Benson, D. A. et al. GenBank. *Nucleic Acids Res.* **41**, D36–D42 (2013).
61. O'Leary, N. A. et al. Reference sequence (RefSeq) database at NCBI: current status, taxonomic expansion, and functional annotation. *Nucleic Acids Res.* **44**, D733–D745 (2016).
62. Leinonen, R., Sugawara, H. & Shumway, M. The Sequence Read Archive. *Nucleic Acids Res.* **39**, D19–D21 (2011).
63. Song, L. & Crawford, G. E. DNase-seq: a high-resolution technique for mapping active gene regulatory elements across the genome from mammalian cells. *Cold Spring Harb. Protoc.* **2010**, pdb.prot5384 (2010).
64. Belton, J.-M. et al. Hi-C: a comprehensive technique to capture the conformation of genomes. *Methods* **58**, 268–276 (2012).
65. Yao, D. et al. Multicenter integrated analysis of noncoding CRISPRi screens. *Nat. Methods* **21**, 723–734 (2024).
66. ENCODE Project Consortium et al. Expanded encyclopaedias of DNA elements in the human and mouse genomes. *Nature* **583**, 699–710 (2020).
67. Satterlee, J. S. et al. The NIH Common Fund/Roadmap Epigenomics Program: successes of a comprehensive consortium. *Sci. Adv.* **5**, eaaw6507 (2019).
68. Lonsdale, J. et al. The Genotype-Tissue Expression (GTEx) project. *Nat. Genet.* **45**, 580–585 (2013).
69. Auton, A. et al. A global reference for human genetic variation. *Nature* **526**, 68–74 (2015).
70. Sennrich, R., Haddow, B. & Birch, A. Neural machine translation of rare words with subword units. Preprint at https://doi.org/10.48550/arXiv.1508.07909 (2016).
71. Chandra, A., Tünnermann, L., Löfstedt, T. & Gratz, R. Transformer-based deep learning for predicting protein properties in the life sciences. *eLife* **12**, e82819 (2023).
72. Zhou, J. Sequence-based modeling of three-dimensional genome architecture from kilobase to chromosome scale. *Nat. Genet.* **54**, 725–734 (2022).
73. Tang, Z. & Koo, P. K. Evaluating the representational power of pre-trained DNA language models for regulatory genomics. Preprint at *bioRxiv* https://doi.org/10.1101/2024.02.29.582810 (2024).
74. Krizhevsky, A., Sutskever, I. & Hinton, G. ImageNet classification with deep convolutional neural networks. In *NIPS'12: Proc. 26th International Conference on Neural Information Processing Systems* Vol. 1, 1097–1105 (NIPS, 2012).
75. Elman, J. L. Finding structure in time. *Cogn. Sci.* **14**, 179–211 (1990).
76. Hochreiter, S. & Schmidhuber, J. Long short-term memory. *Neural Comput.* **9**, 1735–1780 (1997).
77. Vaswani, A. et al. Attention is all you need. Preprint at https://doi.org/10.48550/arXiv.1706.03762 (2017).
78. Wang, T. et al. What language model architecture and pretraining objective works best for zero-shot generalization? In *Int. Conf. Machine Learning* 22964–22984 (PMLR, 2022).
79. Poli, M. et al. Hyena Hierarchy: towards larger convolutional language models. Preprint at https://doi.org/10.48550/arXiv.2302.10866 (2023).
80. Tay, Y. et al. Are pre-trained convolutions better than pre-trained transformers? Preprint at https://doi.org/10.48550/arXiv.2105.03322 (2022).
81. Yang, K. K., Lu, A. X. & Fusi, N. Convolutions are competitive with transformers for protein sequence pretraining. *Cell Syst.* **15**, 286–294.e2 (2024).
82. Greene, C. S. The future is unsupervised. *Sci. Transl. Med.* **8**, 346ec108 (2016).
83. Benegas, G., Batra, S. S. & Song, Y. S. DNA language models are powerful predictors of genome-wide variant effects. *Proc. Natl Acad. Sci. USA* **120**, e2311219120 (2023).
84. Nguyen, E. et al. Sequence modeling and design from molecular to genome scale with Evo. *Science* **386**, eado9336 (2024).
85. Zhang, Y., Bai, Z. & Imoto, S. Investigation of the BERT model on nucleotide sequences with non-standard pre-training and evaluation of different k-mer embeddings. *Bioinformatics* **39**, btad617 (2023).
86. Gu, A., Goel, K. & Ré, C. Efficiently modeling long sequences with structured state spaces. Preprint at https://doi.org/10.48550/arXiv.2111.00396 (2022).
87. Gu, A. & Dao, T. Mamba: linear-time sequence modeling with selective state spaces. Preprint at https://doi.org/10.48550/arXiv.2312.00752 (2024).
88. Schiff, Y. et al. Caduceus: bi-directional equivariant long-range dna sequence modeling. Preprint at https://doi.org/10.48550/arXiv.2403.03234 (2024).
89. Bishop, C. M. & Bishop, H. *Deep Learning: Foundations and Concepts* (Springer International, 2024).
90. Goodfellow, I., Bengio, Y. & Courville, A. *Deep Learning* (MIT Press, 2016).
91. MIT Deep Learning 6.S191. http://introtodeeplearning.com (accessed 11 July 2024).
92. Bach, S. et al. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PLoS One* **10**, e0130140 (2015).
93. Karollus, A., Mauermeier, T. & Gagneur, J. Current sequence-based models capture gene expression determinants in promoters but mostly ignore distal enhancers. Preprint at *bioRxiv* https://doi.org/10.1101/2022.09.15.508087 (2022).
94. Kelley, D. R. Cross-species regulatory sequence activity prediction. *PLoS Comput. Biol.* **16**, e1008050 (2020).
95. Linder, J., Srivastava, D., Yuan, H., Agarwal, V. & Kelley, D. R. Predicting RNA-seq coverage from DNA sequence as a unifying model of gene regulation. *Nat. Genet.* https://doi.org/10.1038/s41588-024-02053-6 (2025).
96. Fishman, V. et al. GENA-LM: a family of open-source foundational DNA language models for long sequences, *Nucleic Acids Res.* **53**, gkae1310 (2025).
97. Dao, T., Fu, D. Y., Ermon, S., Rudra, A. & Ré, C. FlashAttention: fast and memory-efficient exact attention with IO-awareness. Preprint at https://doi.org/10.48550/arXiv.2205.14135 (2022).
98. Press, O., Smith, N. A. & Lewis, M. Train short, test long: attention with linear biases enables input length extrapolation. Preprint at https://doi.org/10.48550/arXiv.2108.12409 (2022).
99. Hu, E. J. et al. LoRA: low-rank adaptation of large language models. Preprint at https://doi.org/10.48550/arXiv.2106.09685 (2021).
100. Katharopoulos, A., Vyas, A., Pappas, N. & Fleuret, F. Transformers are RNNs: fast autoregressive transformers with linear attention. Preprint at https://doi.org/10.48550/arXiv.2006.16236 (2020).
101. Sun, Y. et al. Retentive Network: a successor to transformer for large language models. Preprint at https://doi.org/10.48550/arXiv.2307.08621 (2023).
102. Gresova, K., Martinek, V., Cechak, D., Simecek, P. & Alexiou, P. Genomic benchmarks: a collection of datasets for genomic sequence classification. *BMC Genomic Data* **24**, 25 (2023).
103. Kaplan, J. et al. Scaling laws for neural language models. Preprint at https://doi.org/10.48550/arXiv.2001.08361 (2020).
104. Serrano, Y., Ciudad, A. & Molina, A. Are protein language models compute optimal? Preprint at https://doi.org/10.48550/arXiv.2406.07249 (2024).
105. Li, F.-Z., Amini, A. P., Yue, Y., Yang, K. K. & Lu, A. X. Feature reuse and scaling: understanding transfer learning with protein language models. Preprint at *bioRxiv* https://doi.org/10.1101/2024.02.05.578959 (2024).

106. Theodoris, C. V. Perspectives on benchmarking foundation models for network biology. *Quant. Biol.* **12**, 335–338 (2024).
107. Thurman, R. E. et al. The accessible chromatin landscape of the human genome. *Nature* **489**, 75–82 (2012).
108. Javierre, B. M. et al. Lineage-specific genome architecture links enhancers and non-coding disease variants to target gene promoters. *Cell* **167**, 1369–1384 (2016).
109. Fang, R. et al. Comprehensive analysis of single cell ATAC-seq data with SnapATAC. *Nat. Commun.* **12**, 1337 (2021).
110. Chen, Y., Xie, M. & Wen, J. Predicting gene expression from histone modifications with self-attention based neural networks and transfer learning. *Front. Genet.* **13**, 1081842 (2022).
111. Kundaje, A. et al. Integrative analysis of 111 reference human epigenomes. *Nature* **518**, 317–330 (2015).
112. Dwork, C. & Roth, A. The algorithmic foundations of differential privacy. *Found. Trends Theor. Comput. Sci.* **9**, 211–407 (2014).
113. McMahan, H. B., Moore, E., Ramage, D., Hampson, S. & Arcas, B. A. Y. Communication-efficient learning of deep networks from decentralized data. Preprint at https://doi.org/10.48550/arXiv.1602.05629 (2016).
114. Clauwaert, J., Menschaert, G. & Waegeman, W. Explainability in transformer models for functional genomics. *Brief. Bioinform.* **22**, bbab060 (2021).
115. Serrano, S. & Smith, N. A. Is attention interpretable? Preprint at https://doi.org/10.48550/arXiv.1906.03731 (2019).
116. Chefer, H., Gur, S. & Wolf, L. Transformer interpretability beyond attention visualization. Preprint at https://doi.org/10.48550/arXiv.2012.09838 (2020).
117. Voita, E., Talbot, D., Moiseev, F., Sennrich, R. & Titov, I. Analyzing multi-head self-attention: specialized heads do the heavy lifting, the rest can be pruned. Preprint at https://doi.org/10.48550/arXiv.1905.09418 (2019).
118. Abnar, S. & Zuidema, W. Quantifying attention flow in transformers. Preprint at https://doi.org/10.48550/arXiv.2005.00928 (2020).
119. Binder, A., Montavon, G., Lapuschkin, S., Müller, K.-R. & Samek, W. Layer-wise relevance propagation for neural networks with local renormalization layers. In *Artificial Neural Networks and Machine Learning–ICANN 2016: 25th International Conference on Artificial Neural Networks* 63–71 (Springer, 2016).
120. Selvaraju, R. R. et al. Grad-CAM: visual explanations from deep networks via gradient-based localization. In *Proc. IEEE International Conference on Computer Vision* 618–626 (IEEE, 2017).
121. Lundberg, S. & Lee, S.-I. A unified approach to interpreting model predictions. Preprint at https://doi.org/10.48550/arXiv.1705.07874 (2017).
122. Kwon, Y. & Zou, J. WeightedSHAP: analyzing and improving Shapley based feature attributions. Preprint at https://doi.org/10.48550/arXiv.2209.13429 (2022).
123. Ullah, F. & Ben-Hur, A. A self-attention model for inferring cooperativity between regulatory features. *Nucleic Acids Res.* **49**, e77 (2021).
124. Toneyan, S. & Koo, P. K. Interpreting cis-regulatory interactions from large-scale deep neural networks. *Nat. Genet.* **56**, 2517–2527 (2024).
125. Zhang, Z. et al. Protein language models learn evolutionary statistics of interacting sequence motifs. *Proc. Natl Acad. Sci. USA* **121**, e2406285121 (2024).
126. Vig, J. et al. BERTology meets biology: interpreting attention in protein language models. Preprint at https://doi.org/10.48550/arXiv.2006.15222 (2021).
127. Bommasani, R. et al. On the opportunities and risks of foundation models. Preprint at https://doi.org/10.48550/arXiv.2108.07258 (2022).
128. Kedzierska, K. Z., Crawford, L., Amini, A. P. & Lu, A. X. Assessing the limits of zero-shot foundation models in single-cell biology. Preprint at *bioRxiv* https://doi.org/10.1101/2023.10.16.561085 (2023).
129. Lu, A. X., Lu, A. X. & Moses, A. Evolution is all you need: phylogenetic augmentation for contrastive learning. Preprint at https://doi.org/10.48550/arXiv.2012.13475 (2020).
130. Benegas, G., Albors, C., Aw, A. J., Ye, C. & Song, Y. S. A DNA language model based on multispecies alignment predicts the effects of genome-wide variants. *Nat. Biotechnol.* https://doi.org/10.1038/s41587-024-02511-w (2025).
131. Belancio, V. P., Deininger, P. L. & Roy-Engel, A. M. LINE dancing in the human genome: transposable elements and disease. *Genome Med.* **1**, 97 (2009).
132. Yang, F. et al. scBERT as a large-scale pretrained deep language model for cell type annotation of single-cell RNA-seq data. *Nat. Mach. Intell.* **4**, 852–866 (2022).
133. Levine, D. et al. Cell2Sentence: teaching large language models the language of biology. Preprint at *bioRxiv* https://doi.org/10.1101/2023.09.11.557287 (2023).
134. Hao, M. et al. Large scale foundation model on single-cell transcriptomics. *Nat. Methods* **21**, 1481–1491 (2024).
135. Szałata, A. et al. Transformers in single-cell omics: a review and new perspectives. *Nat. Methods* **21**, 1430–1443 (2024).
136. Hao, M. et al. Current opinions on large cellular models. *Quant. Biol.* **12**, 433–443 (2024).
137. Hassani, A. & Shi, H. Dilated neighborhood attention transformer. Preprint at https://doi.org/10.48550/arXiv.2209.15001 (2022).
138. Bolya, D. et al. Token Merging: your ViT but faster. Preprint at https://doi.org/10.48550/arXiv.2210.09461 (2022).
139. Alamdari, S. et al. Protein generation with evolutionary diffusion: sequence is all you need. Preprint at *bioRxiv* https://doi.org/10.1101/2023.09.11.556673 (2023).
140. Ho, J., Jain, A. & Abbeel, P. Denoising diffusion probabilistic models. *Adv. Neural Inf. Process. Syst.* **33**, 6840–6851 (2020).
141. Kimura, M. Solution of a process of random genetic drift with a continuous model. *Proc. Natl Acad. Sci. USA* **41**, 144–150 (1955).
142. Kimura, M. Stochastic processes and distribution of gene frequencies under natural selection. *Cold Spring Harb. Symp. Quant. Biol.* **20**, 33–53 (1955).
143. Wakeley, J. The limits of theoretical population. *Genetics* **169**, 1–7 (2005).
144. DaSilva, L. F. et al. DNA-Diffusion: leveraging generative models for controlling chromatin accessibility and gene expression via synthetic regulatory elements. Preprint at *bioRxiv* https://doi.org/10.1101/2024.02.01.578352 (2024).

## Acknowledgements

We acknowledge the support of the Natural Sciences and Engineering Research Council of Canada (NSERC).

## Author contributions

M.E.C. selected the papers to review, summarized contributions from all papers, performed analysis, and designed all figures. A.M., B.W., M.W. and D.F. helped with figure design. C.D. contributed to paper selection and summarizing contributions; A.M., M.W., M.K., F.J.T. and H.G. contributed to manuscript writing. A.M. supervised and B.W. conceived and supervised the project.

## Competing interests

The authors declare no competing interests.

## Additional information

**Supplementary information** The online version contains supplementary material available at https://doi.org/10.1038/s42256-025-01007-9.

**Correspondence and requests for materials** should be addressed to Bo Wang.

**Peer review information** *Nature Machine Intelligence* thanks Jesper Tegner, Fan Yang, Xuegong Zhang and the other, anonymous, reviewer(s) for their contribution to the peer review of this work.

**Reprints and permissions information** is available at www.nature.com/reprints.