arXiv:2407.11435v1 [q-bio.GN] 16 Jul 2024

Genomic Language Models: Opportunities and Challenges

Gonzalo Benegas[1,*], Chengzhong Ye[2,*], Carlos Albors[1,*], Jianan Canal Li[1,*], Yun S. Song[1,2,3,†]

[1]Computer Science Division, University of California, Berkeley
[2]Department of Statistics, University of California, Berkeley
[3]Center for Computational Biology, University of California, Berkeley

July 17, 2024

Abstract

Large language models (LLMs) are having transformative impacts across a wide range of scientific fields, particularly in the biomedical sciences. Just as the goal of Natural Language Processing is to understand sequences of words, a major objective in biology is to understand biological sequences. Genomic Language Models (gLMs), which are LLMs trained on DNA sequences, have the potential to significantly advance our understanding of genomes and how DNA elements at various scales interact to give rise to complex functions. In this review, we showcase this potential by highlighting key applications of gLMs, including fitness prediction, sequence design, and transfer learning. Despite notable recent progress, however, developing effective and efficient gLMs presents numerous challenges, especially for species with large, complex genomes. We discuss major considerations for developing and evaluating gLMs.

*These authors contributed equally to this work.
†To whom correspondence should be addressed: yss@berkeley.edu

INTRODUCTION

Recent advances in AI/ML have profoundly impacted a wide range of scientific disciplines, revolutionizing approaches to modeling, data analysis, interpretation, and discovery. One of the key pillars of this development is self-supervised learning, in which training on massive amounts of unlabeled data enables the learning of complex features and their interactions. This paradigm has particularly transformed Natural Language Processing (NLP), allowing AI models to match human performance on several challenging tasks, including translation [1], speech recognition [2], and even answering questions from standardized professional and academic exams [3].

Just as the aim of NLP is to understand sequences of natural language, a major aim of computational biology is to understand biological sequences. As such, there has been intense recent interest in adapting modern techniques from NLP for biological sequences (DNA, RNA, proteins). In particular, protein sequence databases (e.g., UniProt [4]) have grown exponentially over the past decade, and protein language models (pLMs) trained on these immense data have achieved impressive performance on complex problems such as structure prediction [5] and variant effect prediction [6, 7], to name just a few examples (see [8, 9] for reviews on pLMs and their applications). This success aligns with the intuition that billions of years of evolution have explored portions of the protein sequence space that are relevant to life, so large unlabeled datasets of protein sequences are expected to contain significant biological information.

In a similar vein, large language models (LLMs) trained on DNA sequences have the potential to transform genomics, but training an effective model for genomes presents several additional challenges. For instance, unlike proteins, which are functionally important units and relatively small in size, most genomes are much larger and often contain vast amounts of complex, non-functional regions that overshadow the amount of functional elements. In addition, the number of available whole-genome sequences across the tree of life is minuscule compared to the hundreds of millions of protein sequences, limiting the diversity of functionally important DNA elements in training data. Despite these issues, we believe that language models trained on genomes – referred to as genomic language models (gLMs) – hold great promise for biology. In this article, we review some of the key opportunities and challenges in this domain, and outline major considerations that should be addressed to develop and evaluate gLMs that would be useful to the genomics community.

APPLICATIONS

The general language model framework is summarized in Box 1. Below, we elaborate on three main application areas of gLMs: Fitness prediction, Sequence design, and Transfer learning.

Fitness prediction

An intriguing application of gLMs is the unsupervised prediction of the fitness (specifically, deleteriousness) of genetic variants. The underlying idea is that reference genomes, typically derived from healthy individuals, are relatively depleted of deleterious variants. Consequently, models trained on these data are predisposed to assigning lower probabilities to harmful variants. This observation underpins the strategy of using the log-likelihood ratio (LLR) between two alleles (e.g., $\log[\mathbb{P}(X_i = a \mid X_{-i})/\mathbb{P}(X_i = b \mid X_{-i})]$ in the MLM setting described in Box 1) to estimate their relative fitness. A significant benefit of this approach is its independence from supervised labels, such as whether a variant is disease-causing, which are often limited and subject to a wide range of biases.

Box 1: General Language Model Framework.

At a high level, a language model is trained to learn the conditional probability distribution of the form $\mathbb{P}[X_i \mid X_{-\text{Masked}}]$ for $i \in$ Masked (in **Masked Language Modeling, MLM)** or $\mathbb{P}[X_k \mid X_{1:k-1}]$ (in **Causal Language Modeling, CLM**), where $X = (X_1, X_2, \ldots)$ denotes a sequence of "tokens" (e.g., nucleotides or amino acids) and "Masked" denotes a collection of masked positions. The key to recent advances in NLP is that, instead of fitting a parametric distribution that one designs by hand, one lets the dataset speak for itself and fit more complex models as more data are observed. This nonparametric density estimation is achieved by leveraging deep learning. Figure 1 depicts the masked language modeling framework for DNA. While the model is trained to predict the nucleotide at each masked site using information from unmasked sites, it will learn position-specific contextual representation (called embedding, a high-dimensional vector in $\mathbb{R}^n$), which then gets converted into a probability distribution over {A,C,G,T}. These embeddings and probability distributions, both of which are position-specific, can be applied to many problems in genomics.

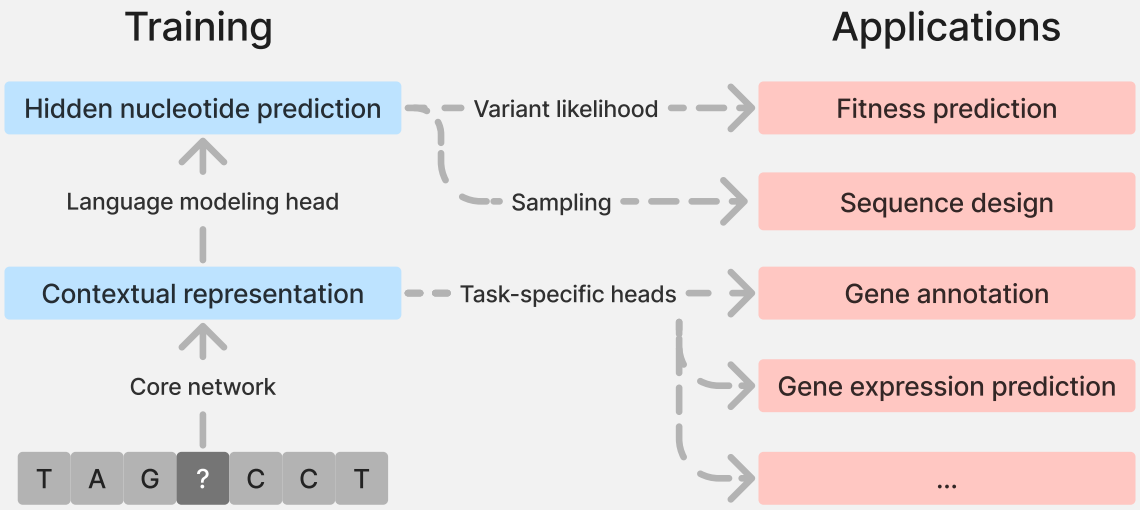

Figure 1: **Training and applications of gLMs.** The schematic on the left-hand side illustrates MLM training. The log-likelihood ratio (LLR) between two alleles (specifically, $\log[\mathbb{P}(X_i = a \mid X_{-i})/\mathbb{P}(X_i = b \mid X_{-i})]$) is a good unsupervised predictor of fitness or deleteriousness (Fitness prediction). New sequences can be generated by sampling from the learned probability distribution (Sequence design). A vector representation, called embedding, of each token in the input sequence can be extracted and adapted for different downstream tasks (Transfer learning).

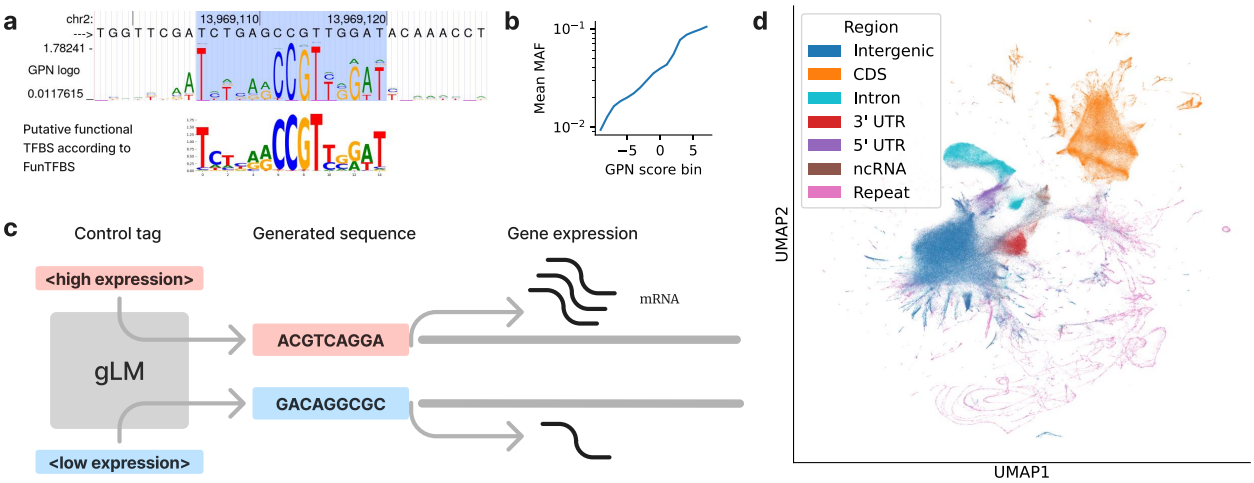

Figure 2: **Application examples.** **(a)** gLM GPN's predicted logo plot (top) at a promoter, highlighting a motif (bottom logo) that matches a putative functional TFBS. **(b)** Correlation between minor allele frequency (MAF) and GPN score (log-likelihood ratio). **(c)** A gLM can be prompted with different control tags to design promoter sequences driving high or low expression in a given cell type. **(d)** Visualization of GPN embeddings for different classes of genomic windows, illustrating that the learned representations contain useful information such as gene regions.

Fitness prediction using the LLR was initially introduced in the context of protein sequence models, leading to outstanding results in predicting the effects of missense variants [6, 10–12]. Expanding this approach, genome-wide fitness prediction using a gLM was first undertaken by GPN [13], achieving state-of-the-art results in the model plant *Arabidopsis thaliana*. To illustrate how a gLM might be able to predict fitness, we note that gLMs can learn transcription factor binding site (TFBS) motifs, understanding which positions are under constraint and which are not (Figure 2a). In addition, GPN's LLR score is correlated with allele frequencies in natural *Arabidopsis thaliana* populations, even though the model was only trained on a single genome from this species (Figure 2b). Subsequently, AgroNT [14] and PlantCaduceus [15] have also obtained excellent results in other plant species. For the human genome, however, the LLR from the Nucleotide Transformer (NT) [16] fell short of existing baselines. Meanwhile, GPN-MSA [17], leveraging a whole-genome multiple sequence alignment (MSA) across diverse vertebrate species, was able to attain state-of-the-art performance (see Learning objective for further MSA considerations). It should be noted that the observed nucleotide distribution is driven not only by fitness but also by mutational biases; explicitly incorporating this information into fitness prediction is a promising avenue of future research.

There are two main kinds of variant effect predictors in genomics: *fitness predictors*, including gLMs but also conservation scores, and *activity predictors*, such as the gene expression predictor Enformer [18] or the splicing predictor SpliceAI [19]. These two kinds of models are related in the sense that if a variant changes fitness, it does so via a change in activity in some context (e.g., change in transcription of a certain gene during limb development), ultimately affecting a high-level trait (e.g., polydactyly). Fitness models cover all possible mechanisms and contexts that affect the overall organismal fitness, while activity models reflect only those they are explicitly trained on (some data, such as protein expression in the developing human brain, are just hard to obtain). On the other hand, activity models can nominate a specific mechanism and context through which a variant acts, while fitness models are more of a black box.

With regards to functional variant prioritization, there are some additional considerations. An

activity model would not be able to prioritize between two variants that induce a similar gene expression fold change but in two different genes, which might have completely different impacts on high-level traits. On the other hand, a trait not under detectable selection could still be of scientific or medical interest. In this case, a fitness model would have limited power to prioritize variants affecting it, especially if they have small effect sizes, as is the case in complex trait GWAS. However, while a gLM's LLR might not have high power in this setting, gLM's learned embeddings (Box 1) could still provide value with additional supervision on labeled data [20].

Sequence design

Designing novel biological sequences is of great interest to both the academic and industry research communities due to its immense potential in drug discovery and delivery; agricultural improvement; bioremediation; and the development of biological research tools. We here describe sequence generation with a CLM (Box 1) as it is the most common approach (see Learning objective for generation with MLMs). Specifically, the sequence generation task is decomposed into a series of next-token prediction problems. Starting with a given sequence fragment (referred to as prompts [21], or control tags [22]), the language model can predict the next token recursively and generate a whole new sequence. pLMs have been shown to be powerful tools for protein design [22–25]. Going beyond coding sequences, designing non-coding sequences is also crucial due to its applications such as gene and cell therapies [26], as well as synthetic biology [27]. Recent work has explored the use of gLMs for *de novo* DNA sequence generation.

The model regLM [26] was built upon the causal gLM HyenaDNA [28] and used to perform *de novo* generation of promoter and enhancer sequences. HyenaDNA models are trained or fine-tuned on regulatory sequences with control tags prepended. The trained model is then used to generate new regulatory sequences with given tags (Figure 2c). The authors performed *in silico* evaluation of the diversity and activity of the generated sequences in yeast and human cell lines, and demonstrated the sequences to have desired functionality as well as realistic and diverse sequence features.

gLMs have the unique potential for multi-modal design tasks such as generating protein-RNA complexes by unifying them as DNA sequence design. For instance, EVO, a gLM trained on prokaryote genomes, was used to design novel CRISPR-Cas systems [27]. The model was fine-tuned using a dataset of CRISPR-Cas sequences with Cas subtype-specific prompt prepended. The fine-tuned model was able to generate novel CRISPR-Cas sequences that matched the subtype prompt and had predicted structures that resemble naturally existing systems.

Additionally, gLMs can be potentially used to design organized, functional DNA sequences at the chromosome or genome-scale. Recently, two gLMs, MegaDNA and EVO, have explored such design tasks for prokaryote genomes [27, 29]. EVO pretrained model was used to generate 20 sequences of size about 650 Mbp. The generated sequences were found to have realistic coding sequence density, protein sequences with predicted secondary structure and globular folds, as well as plausible tRNA sequences. MegaDNA was used to generate full bacteriophage genomes up to 96 kbp. Apart from validating coding sequences, the author further identified functional regulatory elements including promoters and ribosome binding sites in the generated sequences. Yet, such mega-scale DNA sequence design tasks remain challenging. The generated sequences by EVO were found to lack highly conserved marker genes that typically exist in functional prokaryote genomes, and the predicted protein structures have limited matches to natural protein databases. A recent independent evaluation [30] revealed that the sequence composition of MegaDNA-generated genomes is still largely dissimilar to natural genomes. Therefore, further work is needed to refine the methods to enable *de novo* design of fully functional genomes with gLM.

Box 2: Transfer Learning in NLP

For NLP models to generalize on most tasks (including typical tasks, such as sentiment analysis, question answering, and part-of-speech tagging, to name only a few), models need to understand grammar and meaning. However, data specific to these tasks are limited. Utilizing LLMs trained on raw text data (sourced from articles, books, and websites) for transfer learning has enabled breakthrough progress on these problems [36]. Today, virtually every state-of-the-art NLP model is adapted from a LLM.

Transfer learning techniques have underpinned the recent boom in natural language models. In particular, the availability of pretrained models that are broadly adaptable to downstream tasks—termed "foundation models"—has yielded a major shift in how machine learning models are developed [37].

Transfer learning

Neural networks trained to predict annotations from functional genomics experiments have been widely utilized to interpret the functions of genomic elements. A significant application has been predicting variant effects on molecular phenotypes, such as gene expression [18, 31–34] and splicing [19, 35]. The ability of neural networks to interpret complex interactions between genomic sites has made them essential tools for tackling these important problems, but suitable training data are often difficult to collect and consequently limited. To generalize on prediction tasks, models need to develop an understanding of genomic grammar (such as sequence motifs and epistatic interactions), which requires substantial data and computation. To overcome the limitations of insufficient data for individual tasks, developers have employed *transfer learning* methods — techniques that leverage knowledge gained from training models on one task to improve performance on related tasks. Specifically, most neural networks trained to predict functional annotations have been trained to predict a wide array of annotations simultaneously, forcing these models to learn a single unifying representation. This, in turn, has improved their generalization performance.

Language models may also be utilized for transfer learning. See Box 2 for the general concept of transfer learning in NLP. One technique is *feature extraction*: while learning to predict the context-dependent distribution of nucleotides, gLMs transform input genomic sequences into intermediate vector representations (Box 1). These representations may distill relevant information, and, therefore, be utilized as features for another model. For example, visualization of GPN embeddings reveals that, without any supervision, the model has learned to distinguish different classes of genomic elements such as coding sequence and untranslated regions [13] (Figure 2d). Another way to utilize language models for transfer learning is to use them as *pretrained* models: that is, to continue training them on downstream tasks. This technique is called *fine-tuning*. Fine-tuned models develop representations that synthesize knowledge from both tasks, which may improve their generalization performance on the downstream task.

For example, SegmentNT [38], a fine-tuned version of a gLM called Nucleotide Transformer (NT) [16], was recently developed to annotate gene regions and cis-regulatory elements in the human genome at base-pair resolution. Without the MLM pretraining step, the obtained performance was substantially worse. Furthermore, the authors evaluated the accuracy of the model on species not seen during training and how it correlates with phylogenetic distance. AgroNT [14], another model of the NT family, was pretrained on diverse plant species and then fine-tuned to predict chromatin

accessibility and gene expression on select crop species.

Two recent studies evaluated several gLMs in prediction tasks in the human genome and found that they did generally not outperform non-gLM baselines [39, 40]. These results were based on frozen embeddings; evaluating full fine-tuning would provide additional insights. While gLMs are already well suited to demonstrate the value of transfer learning in less-studied organisms, further innovation may be required for them to offer significant value in human genetics, where high-quality labeled data and carefully crafted models already exist. An important question is how far the scaling hypothesis holds for gLMs, i.e., how much increasing unlabeled data and computation will keep improving model performance.

DEVELOPMENT

We now describe the key components of developing useful gLMs; a schematic diagram summarizing the development pipeline is illustrated in Figure 3. We first describe the importance of selecting and preparing training data, and then discuss architectural and training decisions. We then consider interpreting and benchmarking gLMs. Our aim is to provide insights into the methodologies and challenges encountered in developing gLMs that are both effective and efficient. To provide a comprehensive view of the current landscape in the field, we list in Table 1 some of the existing gLMs that we are aware of and summarize their design decisions.

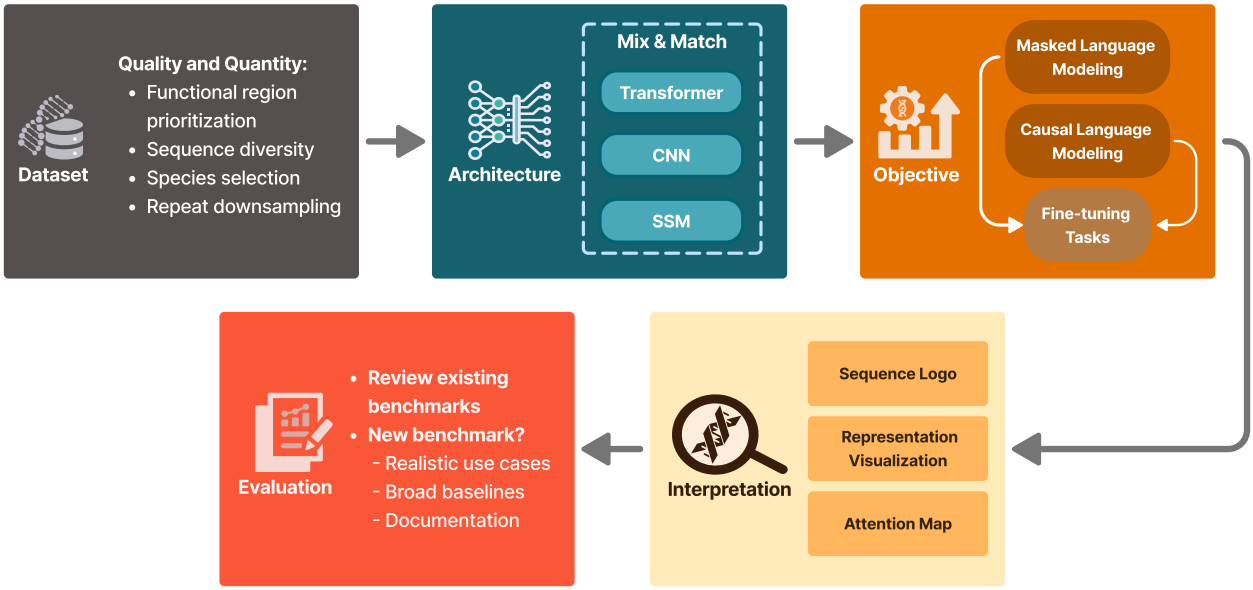

Figure 3: **Development Pipeline.** This figure illustrates the general gLM development pipeline described in this review, from model conception to deployment. We begin with the selection and preparation of the training dataset, emphasizing the importance of data quality and quantity (Training data). Subsequently, in Model architecture and Learning objective, we explore the various choices for designing and training gLMs, discussing the strengths and weaknesses of different approaches. We also examine how hybrid models combine elements from multiple architectures to mitigate specific limitations. In Interpretation, we discuss methods for analyzing and interpreting the outputs of gLMs. Finally, in Evaluation, we present evaluation methods through current benchmarks, emphasizing the complexities in aligning model performance with actual biological functions.

Model Name	Pretraining data sources	Task	Architecture	Tokenization	Notes
BigBird [41]	Human	MLM	Transformer	BPE	
DNABERT [42]	Human	MLM	Transformer	overlapping k-mer	
GeneBERT [43]	Human	MLM	Transformer	overlapping k-mer	Trained to also predict chromatin accessibility ATAC-seq data.
Epigenomic BERT [44]	Human	MLM	Transformer	non-overlapping k-mer	DNA sequences are paired with associated epigenetic state information (IDEAS) [45] during training.
LookingGlass [46]	Bacteria + archaea	CLM	RNN	nucleotide-level	Metagenomic sequences from diverse environments rather than assembled genomes are used for training.
LOGO [47]	Human	MLM	CNN + Transformer	overlapping k-mer	
ViBE [48]	Virus	MLM	Transformer	overlapping k-mer	
GPN [13]	*Arabidopsis thaliana* + 7 related Brassicales genomes	MLM	CNN	nucleotide-level	
FloraBERT [49]	Several hundred plants + selected maize genomes	MLM	Transformer	BPE	Only 1kb promoter sequences are used in training.
INHERIT [50]	Bacteria + bacteriophage	MLM	Transformer	overlapping k-mer	
GenSLMs [51]	Prokaryotic gene sequences + SARS-CoV-2 genomes	CLM	Transformer	non-overlapping k-mer	Pretrain on prokaryotic genes and fine-tune on SARS-CoV-2 genomes.
NT [16]	Human + 1000 Genomes Project + multi-species	MLM	Transformer	non-overlapping k-mer	
SpliceBERT [52]	Human + 71 veterbrate genomes	MLM	Transformer	nucleotide-level	Only RNA Transcripts are used in training.
species LM [53]	1500 fungal genomes	MLM	Transformer	overlapping k-mer	Only 5' and 3' UTR regions are used in training: the 5' species LM and 3' species LM.
GENA-LM [54]	Human + multi-species	MLM	Transformer	BPE	
DNABERT-2 [55]	Human + multi-species	MLM	Transformer	BPE	
HyenaDNA [28]	Human	CLM	SSM	nucleotide-level	
GROVER [56]	Human	MLM	Transformer	BPE	
DNAGPT [57]	Human + multi-species	CLM	Transformer	non-overlapping k-mer	
GPN-MSA [17]	Human + Multiple Sequence Alignment (MSA) with 100 vertebrate genomes	MLM	Transformer	nucleotide-level	
UTR-LM [58]	Human + 4 vertebrate genomes	MLM	Transformer	nucleotide-level	Only 5' UTR regions are used in training. Trained also to predict mRNA minimum free energy and secondary structures calculated by ViennaRNA [59].
hgT5 [60]	Human	T5 [61]	Transformer	Unigram model [62]	
AgroNT [14]	48 plant genomes focusing on edible plant species	MLM	Transformer	BPE	
MegaDNA [29]	∼100k bacteriophage genomes	CLM	Transformer	nucleotide-level	
EVO [27]	Bacteria + archaea + virus + plasmid	CLM	SSM + Transformer	nucleotide-level	
Caduceus [20]	Human	MLM	SSM	nucleotide-level	
ChatNT [63]	DNA sequences + English instructions	CLM	Transformer	overlapping k-mer	Combines the pretrained gLM NT [16] and the English LM Vicuna [64].
PlantCaduceus [15]	16 Angiosperm genomes	MLM	SSM	nucleotide-level	
CD-GPT [65]	Human + multi-species	CLM	Transformer	BPE	DNA/RNA/Protein multimodal pretraining.

Table 1: **A summary of existing gLMs.** An overview of various gLMs is provided, highlighting their pretraining datasets, tasks, architectures, tokenization methods, and unique features. The models are listed in the order of their public release dates.

Training data

The performance of a machine learning model is significantly influenced by both its architecture and its training data. Various model architectures such as convolutional neural networks (CNNs), Transformers, and state-space models have been successfully adapted to a wide range of domains, including natural language, images, audio, proteins, and genomics. However, selecting suitable training data requires a deep understanding of the specific domain, especially in genomics where there is no universally accepted, curated dataset comparable to those in NLP (e.g., the Pile [66, 67]) or proteins (e.g., UniProt [4]).

A key consideration is data quality. For example, in NLP this may refer to data sources that have undergone editing or peer review, such as scientific articles or books [67]. In the case of proteins, quality control involves removing predicted pseudogenes or truncated proteins that are no longer functional [4]. However, a recent study found only 3.3% of the bases in the human reference genome, the most popular gLM training dataset (Table 1), to be significantly constrained and likely functional [68]. Importantly, a typical genomic sequence used for training a gLM will contain a mix of functional and non-functional sites, and one cannot always separate training examples into high vs. low quality. A proposed solution is to have a base-pair-level weighting of the training loss according to the evidence for functionality [17].

It is standard in NLP and proteins to filter out duplicated sequences, which improves training efficiency and reduces memorization [69]. Despite the fact that a staggering 50% of the human genome is repetitive (a high proportion across eukaryotes), very few gLM studies propose a solution (downweighting [13, 15] or downsampling [52, 60]), let alone acknowledge the issue. It would be insightful if studies of language model perplexity [20, 28] would also report it separately for non-repetitive regions, to distinguish improvements due to generalization vs. memorization.

Another key question is how to ensure that the amount of data is enough. It is likely that a single genome might not be enough to train a large model, especially if non-functional regions are downsampled or downweighted. One approach is to add sequence variants from the same species [16]. However, in many cases, e.g., non-African human populations, there is relatively little variation between individuals. A more common approach is to train across multiple species (Table 1), as typically done for pLMs. As species become more distant, the grammar of the non-coding genome starts to change, faster so than the grammar of protein-coding regions. One proposed approach is to explicitly add a species identifier as an extra input to the model [53]. Notwithstanding, it is plausible that a large enough model, with enough genomic context, might be able to naturally model distant genomes, similarly to how LLMs handle multilingual datasets.

As mentioned earlier, in prokaryotes, there exist models (MegaDNA and EVO) that take an entire genome as context [27, 29]. This is currently infeasible for eukaryotes, and therefore leads to the question of how to partition the genome into context windows to be separately modeled. Many interactions are restricted to nearby positions, such as transcription factor binding site motifs, motivating the development of models with a relatively small context (< 6 kb) (Table 1). However, there are obvious long-range interactions, such as between exons of the same gene or between enhancers and promoters (up to 1 Mb) [70]. Such long context lengths introduce computational and statistical challenges, and efforts have been made to overcome them [20, 27–29]. Regardless of the chosen context length, it is still not easy to partition the genome into independent units (similarly to how proteomes are separated by protein). For instance, the enhancer of a gene can be located inside the intron of another gene [70].

The choice of training data may significantly influence gLMs' outputs and learned representations. DNA sequences observed in nature are the outcome of various evolutionary processes, the foremost of which are mutation and selection [71]. For certain applications, it may be desirable to

curate training data such that one of these processes is more manifest than the other. For example, for the sake of fitness prediction, it may be desirable to exclude/downweight hypermutable sites (such as CpG sites) and nonfunctional regions (such as certain classes of repetitive elements).

Model architecture

CNN models [31–34] have been widely used in genomics for supervised tasks prior to the emergence of the Transformer architecture [1]. CNNs are particularly effective at capturing local dependencies and motifs within genomic sequences through their ability to apply filters across the input data. These models have been successful in predicting DNA-protein binding sites, regulatory elements, and TFBS. GPN [13], the aforementioned gLM for genome-wide variant effect prediction in *Arabidopsis thaliana*, took inspiration from the success of language models with modified CNN layers in NLP [72] and protein modeling [73], and replaced self-attention layers in a Transformer encoder with dilated CNN layers.

Transformer models have revolutionized various machine learning domains, particularly in NLP [1], and have recently been widely adopted for genomics modeling. The self-attention mechanism allows each token to attend to all positions in the input sequence simultaneously, enabling the model to dynamically focus on relevant parts of the sequence. This capability has led to significant advancements in detecting regulatory mechanisms for supervised gene expression tasks [18, 74].

Despite their strengths, Transformer models face several challenges unique to genomic modeling. One significant issue is that Transformers have weak or no inductive biases regarding the locality of interactions [41, 75], making them less data-efficient at modeling local motifs such as TFBS. While this can be partially mitigated with CNN-Transformer hybrid architectures such as LOGO [47], further research is needed to enhance this aspect.

Another challenge is the context length: the self-attention mechanism results in computational time and memory scaling quadratically with the input sequence length, making it impractical to apply Transformers to very long genomic sequences [76]. Consequently, the longest input length that conventional attention-based gLMs can handle so far is 12 kb for NT-v2 [16]. To address this limitation, several Transformer-based gLMs have implemented approximate attention or hierarchical attention methods that sacrifice full pairwise attention between all tokens. These methods include the use of sparse attention [41] in GENA-LM [54], which extends the context length to 36 kb, and the MEGABYTE sub-quadratic hierarchical self-attention [77] employed in MegaDNA [29], achieving a context length of 96 kb.

To overcome the quadratic scaling issues of self-attention, various state-space models (SSMs) [78–80] have been proposed for gLMs as efficient alternatives to Transformers, offering nearly linear scaling with sequence length. HyenaDNA [28], based on the Hyena Hierarchy [79], can support input contexts as long as 1 million nucleotides. EVO [27], a hybrid model combining Hyena and Transformer architectures, is pretrained with 8 kb sequences and later fine-tuned with 131 kb sequences during the context extension stage. Caduceus [20], built on the Mamba-based SSM [80], is trained on 131 kb sequences while incorporating reverse-complementarity equivariance.

Learning objective

As described in Box 1, the MLM task (sometimes also called "masked token prediction") involves predicting the identities of tokens randomly omitted from sequences with a predetermined probability (a common choice is 15%) given the remaining tokens. This framework has been used to train the seminal LLM BERT [36] and pLM ESM-1b [81], and has since been widely used for training gLMs. The CLM task (also referred to as "autoregressive language modeling" or "next token

prediction") involves predicting the identities of tokens in sequences given their preceding tokens; it has been used to train the GPT series of LLMs [21]. In this task, the model predicts the next token given the previous tokens in a unidirectional, left-to-right order. A commonality between these two tasks is that they require models to predict components of data given other components as context. To generalize on these tasks, models must learn low-dimensional representations of the data. This capability enables the gLMs to understand and generate genomic sequences by capturing the underlying patterns and dependencies within the genome. In protein modeling, MLM tends to achieve better representations and transfer learning capabilities than CLM [82]. On the other hand, CLMs are the traditional choice for generation tasks, but excellent results have been recently obtained with MLMs via progressive unmasking [83, 84].

To reduce input sequence length and model longer context, both k-mer and byte-pair encoding [85] *tokenizations* create artificially defined nucleotide vocabularies larger than the natural nucleotide vocabularies of {`A, C, G, T`}. On the other hand, single-nucleotide tokenization simplifies model interpretation and attribution, and enhances the model's ability to handle genomic variations more effectively.

Several modifications to the training objective have been explored to provide additional signal and boost performance. For instance, GPN-MSA [17] enhances MLM training on the human reference genome with a whole-genome MSA [86, 87] of vertebrate species, leveraging conservation across related species for additional context. A limitation is that whole-genome MSAs have only been generated for certain species, and might require further development to be effective in plants [88]. Similarly, Species LM [53] directly integrates species information by assigning a dedicated token for each yeast species and appending the species token to the input sequence during training and inference. Training on nucleotide sequences has been expanded to enable cross-talk with additional modalities such as epigenetics [43, 44], RNA [65], proteins [65], and natural language [63].

Interpretation

Deep learning models, while having achieved remarkable performance in various prediction tasks, typically lack interpretability and are often used as "black boxes". However, understanding how these models generate such predictions is crucial for enabling broader applications and advancing model development. As a result, a series of methods have been developed to interpret deep learning models, including those specific to genomics [89–91]. While the interpretation of gLMs is still an emerging line of research, several models have been shown to have learned meaningful biological patterns.

The sequence embeddings extracted from language models are commonly used as representations that capture rich contextual information and sequence features. Unsupervised clustering of GPN's final-layer embeddings shows distinct clusters of input sequences that correspond to different genomic classes such as CDS, intronic, UTR, etc. [13] (Figure 2d). Similarly, unsupervised clustering of SpliceBERT embeddings of canonical splice sites and non-splice GT/AG sites reveals distinct clusters that correspond to the two groups, implying that the model has learned to capture key contextual patterns that determine functional elements in the genome [52].

The attention mechanism in the Transformer model is designed to capture the pattern of interaction between input tokens. Thus, interpreting the attention weights or the attention map for a given input sequence can reveal genomic features learned by the model. In SpliceBERT, attention weights between splice donors and acceptors are significantly higher than those between random pairs of sites; also, the strength of interaction tends to be higher within true donor-acceptor pairs compared to other combinations of donor and acceptor sites. These findings suggest that the model

has learned the relationship between functionally interacting sites [52].

The nucleotide reconstruction approach has also been used in several gLMs to discover sequence motifs learned by the models. Specifically, individual positions of the input sequence are masked one at a time and the probability distribution of the nucleotides is predicted by the trained model given the genomic context. The obtained distribution at each site can reveal motifs learned by the model. This approach has been used in GPN to find notable patterns in the distribution of the reconstructed nucleotides. In particular, the model's predictions are generally more confident in functionally important sites. For example, coding sequences and splice donor/acceptor sites are typically predicted with higher confidence than deep intronic sites. Moreover, within coding sequences, the third nucleotide position of a codon, the least determinant of the translated amino acid, is typically predicted with lower confidence than the first two nucleotide positions. Adapting TF-MoDISco [92], a dedicated tool to identify novel TFBS using model predictions, the authors also found sequence motifs that match known ones in TFBS databases and relevant literature [13] (Figure 2a). Similarly, the reconstructed sequence motifs from Species LM [53] also match the binding sites of known DNA- and RNA-binding proteins in species that are unseen during training, with the fidelity of motif reconstruction depending on the context and genomic regions that correctly reflect the *in vivo* binding sites. Furthermore, the reconstructed motifs' composition, existence, and location exhibit species-specific patterns, which suggests gLM as a potentially powerful tool for investigating the evolution of sequence motifs and regulatory code.

Evaluation

Here, we discuss how gLMs can be evaluated in regards to the three application areas described earlier: predicting alleles' fitness, generating novel viable sequences, and transfer learning.

One way to benchmark a fitness predictor is against experimentally obtained measurements from multiplexed assays of variant effect [93–95]. These assays couple functional differences between genetic variants to readouts (such as the expression of a reporter gene). In turn, these readouts may be used to rank variants by their functionality. Since variants that affect function also tend to affect fitness, we should expect that experimentally obtained and predicted ranks of variants should be correlated. One source for these data is ProteinGym, a widely-used collection of experimental data that may be used to benchmark missense variant effect predictors [96]. Another label that may reflect fitness is whether variants have evidence of pathogenicity – that is, can elevate the risk for diseases. Pathogenic variants may affect fecundity, and, therefore, be deleterious. As a result, we can benchmark fitness predictors by evaluating them as pathogenicity classifiers. In human genetics, primary sources of clinical labels for variants include the ClinVar [97], HGMD [98], and OMIM [99] databases. A third type of data that provides information on alleles' fitness values are variant frequencies. Since common variants are unlikely to be highly deleterious [100], their predicted fitness values generally should be relatively higher than those of rare variants. Therefore, we may benchmark predictors based on how well they identify common variants. A primary source of data on human allele frequencies in various ancestry groups is the gnomAD database [101].

The murky relationship between these data and fitness compounds the problem of extrapolating predictors' operational performance from their performance on benchmarks. Predictors may appear to excel, but may do so by exploiting the ways in which benchmarks fail to capture fitness. For example, a critical issue with using clinical labels is that variants are classified based on whether there is ample evidence that they are benign or pathogenic [106]. Since predictors may also utilize this evidence, their benchmarked performance on labeled variants may not reflect their true performance on unlabeled variants. (See Box 3 for a brief discussion of generalization performance.) There are also critical issues with using allele frequency data: for one, in addition to

Box 3: Evaluating Generalization Performance

The purpose of evaluating predictive models is to build trust in their capability to generalize – that is, to make satisfactory predictions for unlabeled data. A straightforward and standard way to estimate the generalization performance of a model is to evaluate its accuracy on a "test set" of labeled data that are representative of unlabeled data of interest [102]. This approach is the basis of most machine learning benchmarks.

Importantly, for this evaluation to be a reliable indicator of generalization performance, models must not be provided any information that may be used to differentiate test set data from the data they will ultimately be deployed on. Otherwise, they may decrease their test set error at the expense of their generalization performance. For this reason, machine learning contests that withhold their test data from participants are routinely organized [103–105].

the direct action of natural selection, allele frequencies are influenced by factors such as mutation rates, drift, background selection, and hitchhiking [107]. As a result, predictors may perform well on benchmarks by predicting the effects of these processes instead of fitness. These issues highlight a need to carefully interpret the causes of predictors' performance, and they have led to calls for greater transparency on which data and methods are used to train predictors [108].

Regarding sequence design evaluation, one should adopt a holistic approach and examine a broad range of properties of the generated sequences. For instance, Polygraph [109], a recent benchmark for regulatory sequence design, proposes a series of analyses that investigate the sequence composition, motif pattern, and predicted functional activity. In whole-genome or chromosome design tasks, one should further evaluate the existence and positioning of essential genes and functional regulatory elements, as well as the interactions between them. Ultimately, the designed sequences should be evaluated via *in vivo* or *in vitro* experiments to determine if they perform the desired function.

The problem of evaluating the utility of gLMs for transfer learning is uniquely challenging, as pretrained models are intended to be applied to a broad and perhaps indeterminate set of tasks. A solution is to evaluate models on a representative subset of tasks and extrapolate their broader utility. One approach is to process data from representative functional genomics experiments (such as those from the ENCODE [110] or Roadmap Epigenomics [111] projects) into genomic and variant annotations that models can be trained to predict from their sequence context. To facilitate comparison between models, these annotations have been consolidated into various standardized sets of training and test data [16, 39, 60, 112]. However, a problem with these benchmarks is that they are often either opaque and possibly say little about the adaptability of models, or they are relatively trivial, evaluating models against annotations that are themselves derived from simple computational methods (which may themselves be heuristic and are surely wrong some of the time). This makes it difficult to tell whether models that perform better on these benchmarks are genuinely more adaptable, and whether outstanding performance on these benchmarks is sufficient for a model to be broadly adaptable.

CONCLUDING REMARKS & FUTURE PERSPECTIVES

In an age of a vast and growing number of genomic sequences, gLMs are emerging as powerful tools to extract complex patterns useful for numerous applications, including fitness estimation, sequence design and transfer learning. However, they do not yet represent a magical, sudden breakthrough as the term "AI" may suggest. Instead, we view them as another useful modeling tool, much like Hidden Markov Models were when they were first introduced.

As we introduce new techniques into the field of biology, we can provide more value by using clear vocabulary and drawing connections to existing concepts rather than introducing unnecessary jargon that exaggerates novelty (which can confuse readers, editors and funding sources). While the distinction is occasionally useful, "self-supervision" is in many aspects just another unsupervised learning technique. Similarly, in most biological tasks "zero-shot" prediction could be rephrased as unsupervised prediction. Finally, designating a pretrained model as "foundation" or "frontier" gives the impression that the model has undeniably redefined a field. This might be applicable to GPT [113], but hardly to current generation models in genomics, which are often insufficiently benchmarked to begin with.

While earlier gLMs tend to be more or less direct adaptations from NLP models, we expect that further contextualization with deep genomics expertise will reap the highest rewards. We note that evaluating the capabilities of gLMs is challenging because metrics may be misleading, especially when over-optimized. A boon for NLP is that humans are experts in natural language and, therefore, can calibrate benchmarks to match their expertise. In genomics, however, we must rely on data and expert knowledge to falsify models. This aspect of the problem makes it especially challenging and may highlight a need for engagement with subject-matter experts and deliberate experimentation for the sake of developing benchmarks. We conclude this review with a set of

Box 4: Outstanding Questions for Future Research

1. How can we best model patterns across a wide range of scales, from motifs to genes to whole genomes?
2. For which applications is it important to model long-range interactions and how does one determine a suitable size of the receptive field?
3. How can we incorporate structural variations into gLMs?
4. What is the best way to utilize population genetic data when training gLMs?
5. How can we best integrate gLMs with other complex modalities, such as transcriptomic and epigenetic data?
6. For developing gLMs, can we better understand what makes some genomes harder to model than others?
7. Will the scaling hypothesis hold for gLMs, and for how long? Are there really that much data available, considering that most may be non-functional? If we do need scale, how can we organize funding and compute resources to ensure that models can be trained by academic labs and not solely depend on a few industry players with unclear incentives?

outstanding questions in Box 4 that we believe warrant further investigation.

Acknowledgments

This work is supported in part by an NIH grant R35-GM134922, a grant from the Koret-UC Berkeley-Tel Aviv University Initiative in Computational Biology and Bioinformatics, and a grant from the Noyce Initiative UC Partnerships in Computational Transformation Program.

References

1. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. In: Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R., eds. *Advances in Neural Information Processing Systems* vol. 30. Curran Associates, Inc. (2017):.

2. Gulati, A., Qin, J., Chiu, C.-C., Parmar, N., Zhang, Y., Yu, J., Han, W., Wang, S., Zhang, Z., Wu, Y. et al. (2020). Conformer: Convolution-augmented transformer for speech recognition. arXiv preprint arXiv:2005.08100. https://arxiv.org/abs/2005.08100.

3. Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F. L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S. et al. (2023). GPT-4 technical report. arXiv preprint arXiv:2303.08774. https://arxiv.org/abs/2303.08774.

4. Bateman, A., Martin, M.-J., Orchard, S., Magrane, M., Ahmad, S., Alpi, E., Bowler-Barnett, E. H., Britto, R., Cukura, A., Denny, P. et al. (2023). UniProt: the universal protein knowledgebase in 2023. Nucleic Acids Research *51*, D523–D531.

5. Lin, Z., Akin, H., Rao, R., Hie, B., Zhu, Z., Lu, W., Smetanin, N., Verkuil, R., Kabeli, O., Shmueli, Y. et al. (2023). Evolutionary-scale prediction of atomic-level protein structure with a language model. Science *379*, 1123–1130.

6. Meier, J., Rao, R., Verkuil, R., Liu, J., Sercu, T., and Rives, A. Language models enable zero-shot prediction of the effects of mutations on protein function. In: Ranzato, M., Beygelzimer, A., Dauphin, Y., Liang, P., and Vaughan, J. W., eds. *Advances in Neural Information Processing Systems* vol. 34. Curran Associates, Inc. (2021):(29287–29303). https://proceedings.neurips.cc/paper_files/paper/2021/file/f51338d736f95dd42427296047067694-Paper.pdf.

7. Truong Jr, T., and Bepler, T. PoET: A generative model of protein families as sequences-of-sequences. In: Oh, A., Naumann, T., Globerson, A., Saenko, K., Hardt, M., and Levine, S., eds. *Advances in Neural Information Processing Systems* vol. 36. Curran Associates, Inc. (2023):(77379–77415). https://proceedings.neurips.cc/paper_files/paper/2023/file/f4366126eba252699b280e8f93c0ab2f-Paper-Conference.pdf.

8. Bepler, T., and Berger, B. (2021). Learning the protein language: Evolution, structure, and function. Cell Systems *12*, 654–669.

9. Ruffolo, J. A., and Madani, A. (2024). Designing proteins with language models. Nature Biotechnology *42*, 200–202.

10. Riesselman, A. J., Ingraham, J. B., and Marks, D. S. (2018). Deep generative models of genetic variation capture the effects of mutations. Nature Methods *15*, 816–822.

11. Frazer, J., Notin, P., Dias, M., Gomez, A., Min, J. K., Brock, K., Gal, Y., and Marks, D. S. (2021). Disease variant prediction with deep generative models of evolutionary data. Nature *599*, 91–95.

12. Brandes, N., Goldman, G., Wang, C. H., Ye, C. J., and Ntranos, V. (2023). Genome-wide prediction of disease variant effects with a deep protein language model. Nature Genetics. https://doi.org/10.1038/s41588-023-01465-0. doi:10.1038/s41588-023-01465-0.

13. Benegas, G., Batra, S. S., and Song, Y. S. (2023). DNA language models are powerful predictors of genome-wide variant effects. Proceedings of the National Academy of Sciences *120*, e2311219120.

14. Mendoza-Revilla, J., Trop, E., Gonzalez, L., Roller, M., Dalla-Torre, H., de Almeida, B. P., Richard, G., Caton, J., Lopez Carranza, N., Skwark, M., Laterre, A., Beguir, K., Pierrot, T., and Lopez, M. (2024). A foundational large language model for edible plant genomes. Communications Biology *7*, 835. https://doi.org/10.1038/s42003-024-06465-2. doi:10.1038/s42003-024-06465-2.

15. Zhai, J., Gokaslan, A., Schiff, Y., Berthel, A., Liu, Z.-Y., Miller, Z. R., Scheben, A., Stitzer, M. C., Romay, C., Buckler, E. S., and Kuleshov, V. (2024). Cross-species plant genomes modeling at single nucleotide resolution using a pre-trained DNA language model. bioRxiv preprint. https://www.biorxiv.org/content/early/2024/06/05/2024.06.04.596709. doi:10.1101/2024.06.04.596709.

16. Dalla-Torre, H., Gonzalez, L., Mendoza Revilla, J., Lopez Carranza, N., Henryk Grywaczewski, A., Oteri, F., Dallago, C., Trop, E., Sirelkhatim, H., Richard, G. et al. (2023). The Nucleotide Transformer: Building and Evaluating Robust Foundation Models for Human Genomics. bioRxiv preprint. https://www.biorxiv.org/content/10.1101/2023.01.11.523679v3.

17. Benegas, G., Albors, C., Aw, A. J., Ye, C., and Song, Y. S. (2023). GPN-MSA: an alignment-based DNA language model for genome-wide variant effect prediction. bioRxiv preprint. https://www.biorxiv.org/content/10.1101/2023.10.10.561776v2.

18. Avsec, Ž., Agarwal, V., Visentin, D., Ledsam, J. R., Grabska-Barwinska, A., Taylor, K. R., Assael, Y., Jumper, J., Kohli, P., and Kelley, D. R. (2021). Effective gene expression prediction from sequence by integrating long-range interactions. Nature Methods *18*, 1196–1203.

19. Jaganathan, K., Panagiotopoulou, S. K., McRae, J. F., Darbandi, S. F., Knowles, D., Li, Y. I., Kosmicki, J. A., Arbelaez, J., Cui, W., Schwartz, G. B. et al. (2019). Predicting splicing from primary sequence with deep learning. Cell *176*, 535–548.

20. Schiff, Y., Kao, C.-H., Gokaslan, A., Dao, T., Gu, A., and Kuleshov, V. (2024). Caduceus: Bi-directional equivariant long-range DNA sequence modeling. arXiv preprint arXiv:2403.03234. https://arxiv.org/abs/2403.03234.

21. Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., and Amodei, D. Language models are few-shot learners. In: Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H., eds. *Advances in Neural Information Processing Systems* vol. 33. Curran Associates, Inc. (2020):(1877–1901). https://proceedings.neurips.cc/paper_files/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf.

22. Madani, A., Krause, B., Greene, E. R., Subramanian, S., Mohr, B. P., Holton, J. M., Olmos, J. L., Xiong, C., Sun, Z. Z., Socher, R. et al. (2023). Large language models generate functional protein sequences across diverse families. Nature Biotechnology *41*, 1099–1106.

23. Ingraham, J., Garg, V., Barzilay, R., and Jaakkola, T. Generative models for graph-based protein design. In: Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., and Garnett, R., eds. *Advances in Neural Information Processing Systems* vol. 32. Curran Associates, Inc. (2019):https://proceedings.neurips.cc/paper_files/paper/2019/file/f3a4ff4839c56a5f460c88cce3666a2b-Paper.pdf.

24. Hsu, C., Verkuil, R., Liu, J., Lin, Z., Hie, B., Sercu, T., Lerer, A., and Rives, A. Learning inverse folding from millions of predicted structures. In: *International Conference on Machine Learning*. PMLR (2022):(8946–8970).

25. Shin, J.-E., Riesselman, A. J., Kollasch, A. W., McMahon, C., Simon, E., Sander, C., Manglik, A., Kruse, A. C., and Marks, D. S. (2021). Protein design and variant prediction using autoregressive generative models. Nature Communications *12*, 2403.

26. Lal, A., Garfield, D., Biancalani, T., and Eraslan, G. regLM: Designing realistic regulatory DNA with autoregressive language models. In: *International Conference on Research in Computational Molecular Biology*. Springer (2024):(332–335).

27. Nguyen, E., Poli, M., Durrant, M. G., Thomas, A. W., Kang, B., Sullivan, J., Ng, M. Y., Lewis, A., Patel, A., Lou, A. et al. (2024). Sequence modeling and design from molecular to genome scale with Evo. bioRxiv preprint (2024–02). https://www.biorxiv.org/content/10.1101/2024.02.27.582234v2.

28. Nguyen, E., Poli, M., Faizi, M., Thomas, A., Wornow, M., Birch-Sykes, C., Massaroli, S., Patel, A., Rabideau, C., Bengio, Y., Ermon, S., Ré, C., and Baccus, S. HyenaDNA: Long-Range Genomic Sequence Modeling at Single Nucleotide Resolution. In: Oh, A., Naumann, T., Globerson, A., Saenko, K., Hardt, M., and Levine, S., eds. *Advances in Neural Information Processing Systems* vol. 36. Curran Associates, Inc. (2023):(43177–43201).

29. Shao, B. (2023). A long-context language model for deciphering and generating bacteriophage genomes. bioRxiv preprint. https://www.biorxiv.org/content/10.1101/2023.12.18.572218v3.

30. Ratcliff, J. D. (2024). Transformer model generated bacteriophage genomes are compositionally distinct from natural sequences. bioRxiv preprint. https://www.biorxiv.org/content/10.1101/2024.03.19.585716v1.

31. Alipanahi, B., Delong, A., Weirauch, M. T., and Frey, B. J. (2015). Predicting the sequence specificities of DNA-and RNA-binding proteins by deep learning. Nature Biotechnology *33*, 831–838.

32. Zhou, J., and Troyanskaya, O. G. (2015). Predicting effects of noncoding variants with deep learning–based sequence model. Nature Methods *12*, 931–934.

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

34. Kelley, D. R., Reshef, Y. A., Bileschi, M., Belanger, D., McLean, C. Y., and Snoek, J. (2018). Sequential regulatory activity prediction across chromosomes with convolutional neural networks. Genome Research *28*, 739–750.

35. Zeng, T., and Li, Y. I. (2022). Predicting RNA splicing from DNA sequence using Pangolin. Genome Biology *23*, 103. https://genomebiology.biomedcentral.com/articles/10.1186/s13059-022-02664-4. doi:10.1186/s13059-022-02664-4.

36. Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. BERT: Pre-training of deep bidirectional transformers for language understanding. In: Burstein, J., Doran, C., and Solorio, T., eds. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Minneapolis, Minnesota: Association for Computational Linguistics (2019):(4171–4186). https://aclanthology.org/N19-1423. doi:10.18653/v1/N19-1423.

37. Bommasani, R., Hudson, D. A. et al. (2021). On the opportunities and risks of foundation models. arXiv preprint arXiv:2108.07258. https://arxiv.org/abs/2108.07258.

38. de Almeida, B. P., Dalla-Torre, H., Richard, G., Blum, C., Hexemer, L., Gélard, M., Mendoza-Revilla, J., Pandey, P., Laurent, S., Lopez, M. et al. (2024). SegmentNT: annotating the genome at single-nucleotide resolution with DNA foundation models. bioRxiv preprint. https://www.biorxiv.org/content/10.1101/2024.03.14.584712v2.

39. Marin, F. I., Teufel, F., Horlacher, M., Madsen, D., Pultz, D., Winther, O., and Boomsma, W. BEND: Benchmarking DNA Language Models on Biologically Meaningful Tasks. In: *International Conference on Learning Representations* (2024):.

40. Tang, Z., and Koo, P. K. (2024). Evaluating the representational power of pre-trained DNA language models for regulatory genomics. bioRxiv preprint. https://www.biorxiv.org/content/10.1101/2024.02.29.582810v1.

41. Zaheer, M., Guruganesh, G., Dubey, K. A., Ainslie, J., Alberti, C., Ontanon, S., Pham, P., Ravula, A., Wang, Q., Yang, L., and Ahmed, A. Big Bird: Transformers for Longer Sequences. In: Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H., eds. *Advances in Neural Information Processing Systems* vol. 33. Curran Associates, Inc. (2020):(17283–17297).

42. Ji, Y., Zhou, Z., Liu, H., and Davuluri, R. V. (2021). DNABERT: pre-trained bidirectional encoder representations from Transformers model for DNA-language in genome. Bioinformatics *37*, 2112–2120.

43. Mo, S., Fu, X., Hong, C., Chen, Y., Zheng, Y., Tang, X., Lan, Y., Shen, Z., and Xing, E. Multi-modal Self-supervised Pre-training for Large-scale Genome Data. In: *NeurIPS 2021 AI for Science Workshop* (2021):.

44. Trotter, M. V., Nguyen, C. Q., Young, S., Woodruff, R. T., and Branson, K. M. (2021). Epigenomic language models powered by Cerebras. arXiv preprint arXiv:2112.07571. https://arxiv.org/abs/2112.07571.

45. Zhang, Y., An, L., Yue, F., and Hardison, R. C. (2016). Jointly characterizing epigenetic dynamics across multiple human cell types. Nucleic Acids Research *44*, 6721–6731.

46. Hoarfrost, A., Aptekmann, A., Farfañuk, G., and Bromberg, Y. (2022). Deep learning of a bacterial and archaeal universal language of life enables transfer learning and illuminates microbial dark matter. Nature Communications *13*, 2606.

47. Yang, M., Huang, L., Huang, H., Tang, H., Zhang, N., Yang, H., Wu, J., and Mu, F. (2022). Integrating convolution and self-attention improves language model of human genome for interpreting non-coding regions at base-resolution. Nucleic Acids Research *50*, e81–e81.

48. Gwak, H.-J., and Rho, M. (2022). ViBE: a hierarchical BERT model to identify eukaryotic viruses using metagenome sequencing data. Briefings in Bioinformatics *23*. doi:10.1093/bib/bbac204. Bbac204.

49. Levy, B., Xu, Z., Zhao, L., Kremling, K., Altman, R., Wong, P., and Tanner, C. FloraBERT: cross-species transfer learning withattention-based neural networks for gene expression prediction (2022). https://doi.org/10.21203/rs.3.rs-1927200/v1. doi:10.21203/rs.3.rs-1927200/v1.

50. Bai, Z., Zhang, Y.-z., Miyano, S., Yamaguchi, R., Fujimoto, K., Uematsu, S., and Imoto, S. (2022). Identification of bacteriophage genome sequences with representation learning. Bioinformatics. Btac509.

51. Zvyagin, M., Brace, A., Hippe, K., Deng, Y., Zhang, B., Bohorquez, C. O., Clyde, A., Kale, B., Perez-Rivera, D., Ma, H. et al. (2023). GenSLMs: Genome-scale language models reveal SARS-CoV-2 evolutionary dynamics. The International Journal of High Performance Computing Applications *37*, 683–705.

52. Chen, K., Zhou, Y., Ding, M., Wang, Y., Ren, Z., and Yang, Y. (2024). Self-supervised learning on millions of primary RNA sequences from 72 vertebrates improves sequence-based RNA splicing prediction. Briefings in Bioinformatics *25*, bbae163.

53. Karollus, A., Hingerl, J., Gankin, D., Grosshauser, M., Klemon, K., and Gagneur, J. (2024). Species-aware DNA language models capture regulatory elements and their evolution. Genome Biology *25*, 83.

54. Fishman, V., Kuratov, Y., Petrov, M., Shmelev, A., Shepelin, D., Chekanov, N., Kardymon, O., and Burtsev, M. (2023). GENA-LM: A Family of Open-Source Foundational Models for Long DNA Sequences. bioRxiv preprint. https://www.biorxiv.org/content/early/2023/06/13/2023.06.12.544594. doi:10.1101/2023.06.12.544594.

55. Zhou, Z., Ji, Y., Li, W., Dutta, P., Davuluri, R., and Liu, H. (2023). DNABERT-2: Efficient foundation model and benchmark for multi-species genome. arXiv preprint arXiv:2306.15006. https://arxiv.org/abs/2306.15006.

56. Sanabria, M., Hirsch, J., and Poetsch, A. R. (2023). The human genome's vocabulary as proposed by the DNA language model GROVER. bioRxiv preprint. https://www.biorxiv.org/content/10.1101/2023.07.19.549677v2.

57. Zhang, D., Zhang, W., He, B., Zhang, J., Qin, C., and Yao, J. (2023). DNAGPT: A generalized pretrained tool for multiple DNA sequence analysis tasks. bioRxiv preprint. https://arxiv.org/abs/2307.05628.

58. Chu, Y., Yu, D., Li, Y., Huang, K., Shen, Y., Cong, L., Zhang, J., and Wang, M. (2024). A 5' UTR language model for decoding untranslated regions of mRNA and function predictions. Nature Machine Intelligence *6*, 449–460.

59. Lorenz, R., Bernhart, S. H., Höner zu Siederdissen, C., Tafer, H., Flamm, C., Stadler, P. F., and Hofacker, I. L. (2011). ViennaRNA Package 2.0. Algorithms for Molecular Biology *6*, 1–14.

60. Robson, E. S., and Ioannidis, N. M. (2023). GUANinE v1. 0: Benchmark Datasets for Genomic AI Sequence-to-Function Models. bioRxiv preprint. https://www.biorxiv.org/content/10.1101/2023.10.12.562113v3.

61. Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. (2020). Exploring the limits of transfer learning with a unified text-to-text transformer. Journal of Machine Learning Research *21*, 1–67. http://jmlr.org/papers/v21/20-074.html.

62. Kudo, T. Subword regularization: Improving neural network translation models with multiple subword candidates. In: Gurevych, I., and Miyao, Y., eds. *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Melbourne, Australia: Association for Computational Linguistics (2018):(66–75). https://aclanthology.org/P18-1007. doi:10.18653/v1/P18-1007.

63. Richard, G., de Almeida, B. P., Dalla-Torre, H., Blum, C., Hexemer, L., Pandey, P., Laurent, S., Lopez, M. P., Laterre, A., Lang, M. et al. (2024). ChatNT: A Multimodal Conversational Agent for DNA, RNA and Protein Tasks. bioRxiv preprint. https://www.biorxiv.org/content/10.1101/2024.04.30.591835v1.

64. Chiang, W.-L., Li, Z., Lin, Z., Sheng, Y., Wu, Z., Zhang, H., Zheng, L., Zhuang, S., Zhuang, Y., Gonzalez, J. E., Stoica, I., and Xing, E. P. Vicuna: An Open-Source Chatbot Impressing GPT-4 with 90%* ChatGPT Quality (2023). https://lmsys.org/blog/2023-03-30-vicuna/.

65. Zhu, X., Qin, C., Wang, F., Yang, F., He, B., Zhao, Y., and Yao, J. (2024). CD-GPT: A Biological Foundation Model Bridging the Gap between Molecular Sequences Through Central Dogma. bioRxiv preprint. https://www.biorxiv.org/content/10.1101/2024.06.24.600337v1.

66. Gao, L., Biderman, S., Black, S., Golding, L., Hoppe, T., Foster, C., Phang, J., He, H., Thite, A., Nabeshima, N., Presser, S., and Leahy, C. (2020). The Pile: An 800GB Dataset of Diverse Text for Language Modeling. arXiv preprint arXiv:2101.00027. https://arxiv.org/abs/2101.00027.

67. Longpre, S., Biderman, S., Albalak, A., Schoelkopf, H., McDuff, D., Kapoor, S., Klyman, K., Lo, K., Ilharco, G., San, N. et al. (2024). The responsible foundation model development cheatsheet: A review of tools & resources. arXiv preprint arXiv:2406.16746. https://arxiv.org/abs/2406.16746.

68. Sullivan, P. F., Meadows, J. R., Gazal, S., Phan, B. N., Li, X., Genereux, D. P., Dong, M. X., Bianchi, M., Andrews, G., Sakthikumar, S. et al. (2023). Leveraging base-pair mammalian constraint to understand genetic variation and human disease. Science *380*, eabn2937.

69. Lee, K., Ippolito, D., Nystrom, A., Zhang, C., Eck, D., Callison-Burch, C., and Carlini, N. Deduplicating training data makes language models better. In: Muresan, S., Nakov, P., and

Villavicencio, A., eds. *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Dublin, Ireland: Association for Computational Linguistics (2022):(8424–8445). https://aclanthology.org/2022.acl-long.577. doi:10.18653/v1/2022.acl-long.577.

70. Schoenfelder, S., and Fraser, P. (2019). Long-range enhancer–promoter contacts in gene expression control. Nature Reviews Genetics *20*, 437–455.

71. King, J. L., and Jukes, T. H. (1969). Non-darwinian evolution. Science *164*, 788–798. doi:10.1126/science.164.3881.788.

72. Tay, Y., Dehghani, M., Gupta, J. P., Aribandi, V., Bahri, D., Qin, Z., and Metzler, D. Are pretrained convolutions better than pretrained transformers? In: *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing* (2021):(4349–4359). https://aclanthology.org/2021.acl-long.335/.

73. Yang, K. K., Fusi, N., and Lu, A. X. (2024). Convolutions are competitive with transformers for protein sequence pretraining. Cell Systems *15*, 286–294.

74. Linder, J., Srivastava, D., Yuan, H., Agarwal, V., and Kelley, D. R. (2023). Predicting RNA-seq coverage from DNA sequence as a unifying model of gene regulation. bioRxiv preprint. https://www.biorxiv.org/content/10.1101/2023.08.30.555582v1.

75. Su, J., Ahmed, M., Lu, Y., Pan, S., Bo, W., and Liu, Y. (2024). Roformer: Enhanced transformer with rotary position embedding. Neurocomputing *568*, 127063.

76. Dai, Z., Yang, Z., Yang, Y., Carbonell, J. G., Le, Q. V., and Salakhutdinov, R. Transformer-XL: Attentive Language Models beyond a Fixed-Length Context. In: Korhonen, A., Traum, D. R., and Màrquez, L., eds. *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*. Association for Computational Linguistics (2019):(2978–2988). https://doi.org/10.18653/v1/p19-1285. doi:10.18653/V1/P19-1285.

77. Yu, L., Simig, D., Flaherty, C., Aghajanyan, A., Zettlemoyer, L., and Lewis, M. MEGABYTE: Predicting million-byte sequences with multiscale transformers. In: Oh, A., Naumann, T., Globerson, A., Saenko, K., Hardt, M., and Levine, S., eds. *Advances in Neural Information Processing Systems* vol. 36. Curran Associates, Inc. (2023):(78808–78823). https://proceedings.neurips.cc/paper_files/paper/2023/file/f8f78f8043f35890181a824e53a57134-Paper-Conference.pdf.

78. Gu, A., Goel, K., and Re, C. Efficiently modeling long sequences with structured state spaces. In: *International Conference on Learning Representations* (2022):.

79. Poli, M., Massaroli, S., Nguyen, E., Fu, D. Y., Dao, T., Baccus, S., Bengio, Y., Ermon, S., and Ré, C. Hyena Hierarchy: Towards larger convolutional language models. In: *International Conference on Machine Learning*. PMLR (2023):(28043–28078).

80. Gu, A., and Dao, T. (2023). Mamba: Linear-time sequence modeling with selective state spaces. arXiv preprint arXiv:2312.00752. https://arxiv.org/abs/2312.00752.

81. Rives, A., Meier, J., Sercu, T., Goyal, S., Lin, Z., Liu, J., Guo, D., Ott, M., Zitnick, C. L., Ma, J., and Ferguson, A. L. (2021). Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. Proceedings of the National Academy of Sciences *118*, e2016239118.

82. Cheng, X., Chen, B., Li, P., Gong, J., Tang, J., and Song, L. (2024). Training compute-optimal protein language models. bioRxiv preprint. https://www.biorxiv.org/content/10.1101/2024.06.06.597716v1.

83. Samuel, D. (2024). BERTs are Generative In-Context Learners. arXiv preprint arXiv:2406.04823. https://arxiv.org/abs/2406.04823.

84. Hayes, T., Rao, R., Akin, H., Sofroniew, N. J., Oktay, D., Lin, Z., Verkuil, R., Tran, V. Q., Deaton, J., Wiggert, M. et al. (2024). Simulating 500 million years of evolution with a language model. bioRxiv preprint. https://www.biorxiv.org/content/10.1101/2024.07.01.600583v1.

85. Sennrich, R., Haddow, B., and Birch, A. Neural machine translation of rare words with subword units. In: Erk, K., and Smith, N. A., eds. *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Berlin, Germany: Association for Computational Linguistics (2016):(1715–1725). https://aclanthology.org/P16-1162. doi:10.18653/v1/P16-1162.

86. Blanchette, M., Kent, W. J., Riemer, C., Elnitski, L., Smit, A. F., Roskin, K. M., Baertsch, R., Rosenbloom, K., Clawson, H., Green, E. D. et al. (2004). Aligning multiple genomic sequences with the threaded blockset aligner. Genome Research *14*, 708–715.

87. Armstrong, J., Hickey, G., Diekhans, M., Fiddes, I. T., Novak, A. M., Deran, A., Fang, Q., Xie, D., Feng, S., Stiller, J. et al. (2020). Progressive Cactus is a multiple-genome aligner for the thousand-genome era. Nature *587*, 246–251.

88. Song, B., Buckler, E. S., and Stitzer, M. C. (2024). New whole-genome alignment tools are needed for tapping into plant diversity. Trends in Plant Science *29*, 355–369.

89. Linardatos, P., Papastefanopoulos, V., and Kotsiantis, S. (2020). Explainable AI: A review of machine learning interpretability methods. Entropy *23*, 18.

90. Zhang, Y., Tiňo, P., Leonardis, A., and Tang, K. (2021). A survey on neural network interpretability. IEEE Transactions on Emerging Topics in Computational Intelligence *5*, 726–742.

91. Talukder, A., Barham, C., Li, X., and Hu, H. (2021). Interpretation of deep learning in genomics and epigenomics. Briefings in Bioinformatics *22*, bbaa177.

92. Shrikumar, A., Tian, K., Avsec, Ž., Shcherbina, A., Banerjee, A., Sharmin, M., Nair, S., and Kundaje, A. (2018). Technical note on transcription factor motif discovery from importance scores (TF-MoDISco) version 0.5. 6.5. arXiv preprint arXiv:1811.00416. https://arxiv.org/abs/1811.00416.

93. Fowler, D. M., Adams, D. J., Gloyn, A. L., Hahn, W. C., Marks, D. S., Muffley, L. A., Neal, J. T., Roth, F. P., Rubin, A. F., Starita, L. M., and Hurles, M. E. (2023). An Atlas of Variant Effects to understand the genome at nucleotide resolution. Genome Biology *24*, 147.

94. Kircher, M., Xiong, C., Martin, B., Schubach, M., Inoue, F., Bell, R. J. A., Costello, J. F., Shendure, J., and Ahituv, N. (2019). Saturation mutagenesis of twenty disease-associated regulatory elements at single base-pair resolution. Nature Communications *10*. doi:10.1038/s41467-019-11526-w.

95. Findlay, G. M., Daza, R. M., Martin, B., Zhang, M. D., Leith, A. P., Gasperini, M., Janizek, J. D., Huang, X., Starita, L. M., and Shendure, J. (2018). Accurate classification of BRCA1 variants with saturation genome editing. Nature *562*, 217–222.

96. Notin, P., Kollasch, A. W., Ritter, D., Niekerk, L. V., Paul, S., Spinner, H., Rollins, N. J., Shaw, A., Orenbuch, R., Weitzman, R., Frazer, J., Dias, M., Franceschi, D., Gal, Y., and Marks, D. S. ProteinGym: Large-Scale Benchmarks for Protein Fitness Prediction and Design. In: *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track* (2023):https://openreview.net/forum?id=URoZHqAohf.

97. Landrum, M. J., Lee, J. M., Benson, M., Brown, G. R., Chao, C., Chitipiralla, S., Gu, B., Hart, J., Hoffman, D., Jang, W. et al. (2016). ClinVar: public archive of interpretations of clinically relevant variants. Nucleic Acids Research *44*, D862–D868.

98. Stenson, P. D., Mort, M., Ball, E. V., Evans, K., Hayden, M., Heywood, S., Hussain, M., Phillips, A. D., and Cooper, D. N. (2017). The Human Gene Mutation Database: towards a comprehensive repository of inherited mutation data for medical research, genetic diagnosis and next-generation sequencing studies. Human Genetics *136*, 665–677. doi:10.1007/s00439-017-1779-6.

99. Amberger, J. S., Bocchini, C. A., Schiettecatte, F., Scott, A. F., and Hamosh, A. (2015). OMIM.org: Online Mendelian Inheritance in Man (OMIM®), an online catalog of human genes and genetic disorders. Nucleic Acids Research *43*, D789–D798.

100. Pritchard, J. K., and Cox, N. (2002). The allelic architecture of human disease genes: common disease–common variant...or not? Human Molecular Genetics *11*, 2417–2423. doi:10.1093/hmg/11.20.2417.

101. Karczewski, K. J., Francioli, L. C., Tiao, G., Cummings, B. B., Alföldi, J., Wang, Q., Collins, R. L., Laricchia, K. M., Ganna, A., Birnbaum, D. P., Gauthier, L. D., Brand, H., Solomonson, M., Watts, N. A., Rhodes, D., Singer-Berk, M., England, E. M., Seaby, E. G., Kosmicki, J. A., Walters, R. K., Tashman, K., Farjoun, Y., Banks, E., Poterba, T., Consortium, G. A. D., and MacArthur, D. G. (2020). The mutational constraint spectrum quantified from variation in 141,456 humans. Nature *581*, 434–443. doi:10.1038/s41586-020-2308-7.

102. Vapnik, V. N. The Nature of Statistical Learning Theory. New York: Springer (1999).

103. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C., and Fei-Fei, L. (2015). ImageNet Large Scale Visual Recognition Challenge. International Journal of Computer Vision (IJCV) *115*, 211–252. doi:10.1007/s11263-015-0816-y.

104. Moult, J., Fidelis, K., Kryshtafovych, A., Schwede, T., and Tramontano, A. (2018). Critical assessment of methods of protein structure prediction (CASP)—round XII. Proteins: Structure, Function, and Bioinformatics *86*, 7–15.

105. Johnson, A. D., Handsaker, R. E., Pulit, S. L., Nizzari, M., O'Donnell, C. J., and de Bakker, P. I. (2017). CAGI: The Critical Assessment of Genome Interpretation. Genome Biology *18*, 1–5.

106. Grimm, D. G., Azencott, C.-A., Aicheler, F., Gieraths, U., MacArthur, D. G., Samocha, K. E., Cooper, D. N., Stenson, P. D., Daly, M. J., Smoller, J. W. et al. (2015). The evaluation of tools used to predict the impact of missense variants is hindered by two types of circularity. Human Mutation *36*, 513–523.

107. Hartl, D. L., Clark, A. G., and Clark, A. G. Principles of population genetics vol. 116. Sinauer associates Sunderland, MA (1997).

108. Livesey, B. J., Badonyi, M., Dias, M., Frazer, J., Kumar, S., Lindorff-Larsen, K., McCandlish, D. M., Orenbuch, R., Shearer, C. A., Muffley, L. et al. (2024). Guidelines for releasing a variant effect predictor. arXiv preprint. https://arxiv.org/abs/2404.10807.

109. Gupta, A., Lal, A., Gunsalus, L. M., Biancalani, T., and Eraslan, G. (2023). Polygraph: A software framework for the systematic assessment of synthetic regulatory DNA elements. bioRxiv preprint. https://www.biorxiv.org/content/10.1101/2023.11.27.568764v2.

110. Consortium, E. P. et al. (2012). An integrated encyclopedia of DNA elements in the human genome. Nature *489*, 57–74.

111. Kundaje, A., Meuleman, W., Ernst, J. et al. (2015). Integrative analysis of 111 reference human epigenomes. Nature *518*, 317–330.

112. Grešová, K., Martinek, V., Čechák, D., Šimeček, P., and Alexiou, P. (2023). Genomic benchmarks: a collection of datasets for genomic sequence classification. BMC Genomic Data *24*, Article number: 25.

113. Bubeck, S., Chandrasekaran, V., Eldan, R., Gehrke, J., Horvitz, E., Kamar, E., Lee, P., Lee, Y. T., Li, Y., Lundberg, S., Nori, H., Palangi, H., Ribeiro, M. T., and Zhang, Y. (2023). Sparks of Artificial General Intelligence: Early experiments with GPT-4. arXiv preprint. https://arxiv.org/abs/2303.12712. arXiv:2303.12712.