GENA-LM: A Family of Open-Source Foundational Models for Long DNA Sequences

Veniamin Fishman [1,2,+,*], Yuri Kuratov [1,3,+], Maxim Petrov [1], Aleksei Shmelev [1,4], Denis Shepelin [1], Nikolay Chekanov [1], Olga Kardymon [1,4*] and Mikhail Burtsev [5*]

[1]AIRI, Moscow, Russia,
[2]Institute of Cytology and Genetics, Novosibirsk, Russia,
[3]Moscow Institute of Physics and Technology, Dolgoprudny, Russia,
[4]HSE University, Moscow, Russia and
[5]London Institute for Mathematical Sciences, London, UK

*corresponding authors.
+equal contribution

Abstract

The field of genomics has seen substantial advancements through the application of artificial intelligence (AI), with machine learning revealing the potential to interpret genomic sequences without necessitating an exhaustive experimental analysis of all the intricate and interconnected molecular processes involved in DNA functioning. However, precise decoding of genomic sequences demands the comprehension of rich contextual information spread over thousands of nucleotides. Presently, only a few architectures exist that can process such extensive inputs, and they require exceptional computational resources. To address this need, we introduce GENA-LM, a suite of transformer-based foundational DNA language models capable of handling input lengths up to 36 thousands base pairs. We offer pre-trained versions of GENA-LM and demonstrate their capacity for fine-tuning to address complex biological questions with modest computational requirements. We also illustrate diverse applications of GENA-LM for various downstream genomic tasks, showcasing its performance in either matching or exceeding that of prior models, whether task-specific or universal. All models are publicly accessible on GitHub `https://github.com/AIRI-Institute/GENA_LM` and as pre-trained models with `gena-lm-` prefix on HuggingFace `https://huggingface.co/AIRI-Institute`.

Contacts: minja-f@ya.ru, kardymon@airi.net, am@lims.ac.uk

1 Introduction

The process by which DNA encodes genetic information is a paramount inquiry within the field of biology. While the system of protein translation employs a relatively simple and universal genetic code to decipher messenger RNA into amino acid sequences, other encoding forms, such as the epigenetic code, are considerably more complex (Kim and Wysocka, 2023). Undoubtedly, functional genome elements like promoters, enhancers, transcription factor binding sites, insulators, splicing, and polyadenylation sites are dictated by DNA sequences. However, the inherent diversity and redundancy of underlying motifs pose a challenge for detection within expansive eukaryotic genomes. This complicates our understanding of non-coding genome evolution and interpretation of human genomic variants as the complexity of the epigenetic code has yet to be fully dissected.

The advent of next-generation sequencing and other high-throughput technologies has led to the creation and public deposition of vast databases containing functional genomic elements. This development has fostered the application of computational methods for large-scale genomic data analysis (Sean Whalen, 2022). Toward this goal, we and others (Libbrecht and Noble, 2015) have effectively employed machine learning methodologies such as ensemble learning (Belokopytova *et al.*, 2020) and convolutional neural networks (Sindeeva *et al.*, 2023; Penzar *et al.*, 2022). Although powerful, these approaches face limitations in identifying long-range dependencies between DNA sequences, which are common in humans and other eukaryotic genomes (Belokopytova and Fishman, 2021a). However, recently introduced transformer neural network-based strategies strive to overcome these limitations (Belokopytova and Fishman, 2021a). Cutting-edge transformer architectures have demonstrated an

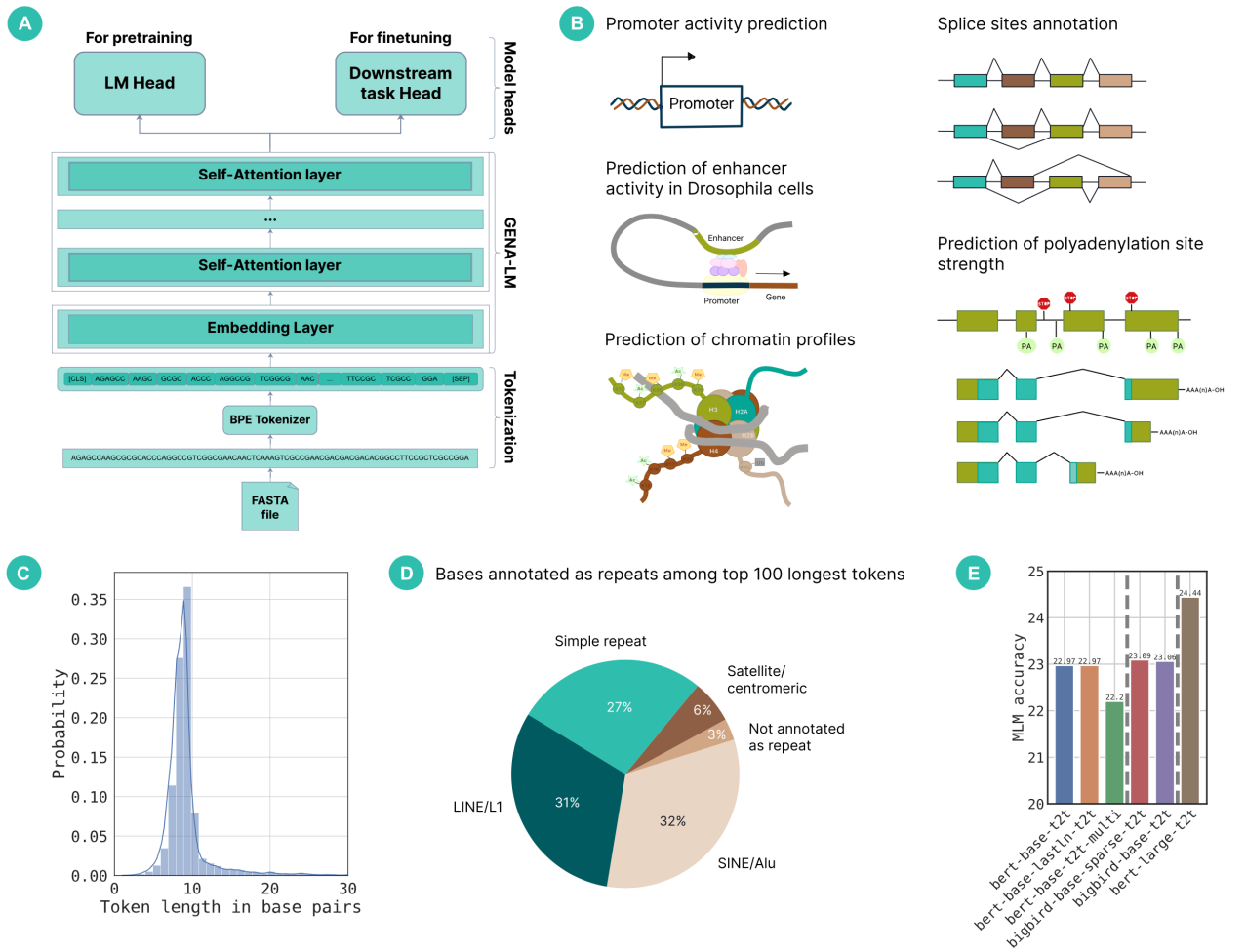

Fig. 1. The GENA-LM Family: Foundational DNA Models. A. GENA-LM is a transformer-based architecture, pre-trained using a masked language modeling (MLM) objective on DNA sequences. The input sequence is divided into tokens using the BPE algorithm and then fed into the transformer layers. After pre-training, the resulting foundational DNA model is augmented with a downstream head and fine-tuned for a specific task. B. Downstream tasks for model evaluation encompass the prediction of promoter and enhancer activity, splicing sites, chromatin profiles, and polyadenylation site strength. C. The distribution of token lengths following BPE tokenization indicates that the median token length is nine base pairs. D. The BPE vocabulary encompasses biologically significant tokens, with the longest tokens representing well-known repetitive elements such as LINEs or simple repeats. E. MLM accuracies for GENA-LM pre-training reveal that sparse attention models achieve higher accuracy than their full-attention counterparts of relatively shorter length.

ability to infer specific epigenetic properties and gene expression levels from DNA sequences with unparalleled precision (Avsec *et al.*, 2021). A significant drawback is that training these models demands substantial computational resources and, once trained, the model cannot infer features not incorporated into the training dataset.

In the realm of natural language processing, transfer learning (Pan and Yang, 2010), particularly pre-training (Dai and Le, 2015; Peters *et al.*, 2018; Howard and Ruder, 2018; Radford *et al.*, 2018; Devlin *et al.*, 2019), has become the primary approach for saving computational resources and achieving high performance even with limited target data. A model that is pre-trained on a large unlabeled dataset acquires general-purpose representations and can subsequently be fine-tuned or used as a feature extractor to address new tasks using a different dataset. Compared to models trained solely on a task-specific dataset, pre-trained models are more efficient computationally and often outperform, especially with smaller datasets. The transfer learning approach has recently been used to train a BERT-like (Devlin *et al.*, 2019) Transformer neural network (Vaswani *et al.*, 2017) on DNA sequences, culminating in the DNABERT model (Ji *et al.*, 2021). This model was first trained to predict human genome subsequences using context and then fine-tuned for multiple downstream tasks, including promoter activity prediction and transcription factor binding. Despite the promise shown by DNABERT, its predictive capabilities are constrained by an input size limited to 500 base pairs, insufficient for capturing longer contexts required for numerous genomic applications.

Several recent developments have addressed increasing input size for transformer models: sparse attention, effective attention, and recurrence. Sparse attention techniques employ predefined or learned attention patterns (e.g., sliding window, block-diagonal, and global attention) to mitigate the quadratic dependence of full attention on input length to linear (Guo *et al.*, 2019; Beltagy *et al.*, 2020; Ainslie *et al.*, 2020; Zaheer *et al.*, 2020; Kitaev *et al.*, 2020). Linear attention methods approximate full token-to-token interactions using softmax linearization (Choromanski *et al.*, 2021; Katharopoulos *et al.*, 2020). Recurrent models segment the input sequence and process these sequentially, passing information between segments via the reuse of previous hidden states (Dai *et al.*, 2019; Rae *et al.*, 2020) or via special memory (Wu *et al.*, 2022; Hutchins *et al.*, 2022; Bulatov *et al.*, 2022). The recently proposed Recurrent Memory Transformer architecture enables the aggregation of information from long (thousands) (Bulatov

et al., 2022) and extremely long (millions) input sequences (Bulatov *et al.*, 2023).

In this paper, we demonstrate the successful application of these advanced transformer-based neural networks to DNA sequences to effectively predict various functional elements. These include promoter activity, splicing, polyadenylation sites, enhancer annotations, and chromatin profiles. We offer the research community a family of open-source models, GENA-LM [1], and illustrate that fine-tuning these models often yields superior results compared to existing state-of-the-art architectures.

2 Methods

2.1 Datasets

2.1.1 Genomic datasets for language model pre-training

Human T2T v2 genome assembly was downloaded from NCBI (acc. GCF_009914755.1). Genomic datasets that were used to train multispecies models were downloaded from ENSEMBL release 108[2]. The list of species is provided in Supplementary Table 1[3]. For the 1000-genome dataset, we used gnomAD v. 3.1.2 data.

2.1.2 Genomic datasets preprocessing

To convert genomic datasets to the training corpus we processed each record in genomic fasta files, omitting contigs that include the substring "*mitochondrion*" in their identifier and contigs shorter than 10 kb. We split the remaining sequences into "sentences" of length 500–1000 base pairs (the sentence length was chosen randomly for each sentence) and prepared "documents" containing 50–100 consecutive sentences. "Sentences" and "documents" pipeline is similar to how the data was processed in BigBird (Zaheer *et al.*, 2020). We augmented data by including reverse-complement sequences and used stochastic shift so that some documents contain overlapping genomic sequences. For the 1000-genome SNP augmentation, we substituted reference nucleotides with alternative alleles listed in the gnomAD dataset. To preserve haplotype structure, we processed each gnomAD sample individually, i.e., for each genomic region we obtained several sequences, each produced by substituting reference alleles with alternative variants from one particular gnomAD sample. We only used genomic regions where the fraction of genomic positions with reported common SNPs (AF > 0.5) for a particular gnomAD sample was above 0.01.

2.1.3 Train and test split

For our first models (*bert-base* and bigbird-base-sparse) we hold out human chromosomes 22 and Y (CP068256.2 and CP086569.2) as the test dataset for the masked language modeling task. For all other models, we hold out human chromosomes 7 and 10 (CP068271.2 and CP068268.2); these models have the suffix "t2t" in their names. Other data was used for training. Human-only models were trained on pre-processed Human T2T v2 genome assembly and its 1000-genome SNP augmentations making in a total of $\approx 480 \times 10^9$ base pairs. Multispecies models were trained on human-only and multispecies data making in a total of $\approx 1072 \times 10^9$ base pairs. We follow the data split strategy for downstream tasks according to the previously used design. We provide the details in a section dedicated to each of the tasks.

2.1.4 Sequence tokenization

We used Byte-Pair Encoding (BPE) tokenization (Sennrich *et al.*, 2016) for our models, with dictionary size set to 32 000 and initial character-lvl vocabulary [’A’, ’T’, ’G’, ’C’, ’N’]. We used two tokenizers: one trained on human T2T v2 genome assembly only (refered as T2T split v1 in Table 2) and the tokenizer that was trained on both human-only and multispecies data sampled equally (refered as T2T+1000G SNPs+Multispieces). The tokenizers include special tokens: CLS, SEP, PAD, UNK, and MASK. The second tokenizer, trained on T2T+1000G SNPs+Multispieces, includes a preprocessing step for long gaps: more than 10 consecutive ’N’ are replaced by a single ’–’ token.

2.1.5 Downstream task datasets

A brief summary of downstream task dataset parameters can be found in Table 1. We also provide a more detailed description is provided below.

Promoters prediction. For the promoter prediction task, we downloaded human sequences upstream TSS (transcriptional start sites) from EPDnew[4]. We extracted 300-, 2000-, or 16000-bp length sequences, and each of these dataset was processed and scored separately. We prepared negative samples as described in (Zaheer *et al.*, 2020): briefly, negative samples were generated from positive samples by splitting the promoter sequence into consecutive non-overlapping 20 subsequences and shuffling the order of these 20 subsequences. Train, validation, and test chunks were obtained by splitting the dataset in a by-sequence manner. The task is a binary classification: the region contains a promoter or not.

Splice site prediction. For the prediction of the splice donor and acceptor sites, the dataset described in (Jaganathan *et al.*, 2019) was reproduced using original scripts provided by the authors. The train and test splits were the same as in the (Jaganathan *et al.*, 2019). In this dataset, the target 5000-bp region is flanked by a 10000-bp context. We map per-base splice site annotations within the target region to token positions so that if the token overlaps the splice-donor or splice-acceptor site, it is considered a positive sample for this splicing annotation class. Next, the target and context were tokenized separately, and if their total length did not match the model input size, padding or truncation was applied. When truncating, we start from the sequences farthest from the middle of the target region. We inserted SEP tokens between context and target sequences. After processing data this way, we obtain the target with a size equal to the number of the model’s input tokens, but the loss was not computed for tokens representing context or padding. The task for this challenge is multi-class token-level classification with three classes: splice donor, splice acceptor, and none.

Drosophila enhancers prediction. Candidate sequences and their housekeeping and tissue-specific activity in *Drosophila* cells were downloaded from [5]. These data already include train/validation/test splits previously used to train the DeepSTARR model (de Almeida *et al.*, 2022). This challenge is two-class regression, i.e., predicting for every 249-bp sequence two float values, one for housekeeping and one for developmental enhancer scores.

Chromatin profiling. The original DeepSEA (Zhou and Troyanskaya, 2015) dataset was downloaded from [6]. DeepSEA dataset describes the chromatin binding/occupancy profile for multiple genomic features, including histone marks, transcription factors, and DNAse I

[1] GitHub: `https://github.com/AIRI-Institute/GENA_LM` and pre-trained models with `gena-lm-` prefix in `https://huggingface.co/AIRI-Institute`

[2] `https://ftp.ensembl.org/pub/release-108/`

[3] `https://github.com/AIRI-Institute/GENA_LM/blob/main/data/full_ensembl_genomes_metadata.cvs`

[4] `https://epd.epfl.ch/EPDnew_select.php`

[5] `https://data.starklab.org/almeida/DeepSTARR/Data/`

[6] `http://deepsea.princeton.edu/media/code/deepsea_train_bundle.v0.9.tar.gz`

Table 1. Parameters of downstream task datasets.

Downstream task	Input length, bp	Number of targets	Task
Promoters prediction (300)	300	2	classification
Promoters prediction (2000)	2000	2	classification
Promoters prediction (16000)	16000	2	classification
Splice site prediction	15000	3 per token / bp	multi-class classification
Drosophila enhancers prediction	249	2	regression
Chromatin profiling (1000)	1000	919	multi-label classification
Chromatin profiling (8000)	8000	919	multi-label classification
Polyadenylation sites prediction	443	1	regression

hypersensitivity region. The dataset contains DNA sequences of length 1000 bp, which includes the target 200-bp region and a 400-bp left and right context. The occupancy of each feature is measured for each of the 200-bp target regions. We also experimented with a longer context of 7800 base pairs (total input length 8000-bp). To extend the DNA context for this dataset, we searched the input DNA segments in the hg19 genome using *bwa fastmap* and collected the required portion of sequence around mapped locations. We discarded the sequences that failed to remap or mapped too close to the end of the chromosome to be extended (less than 1% of the data). We used the same train/validation/test split as was provided in the original DeepSEA dataset. This task is multi-label classification, with the number of classes matching the number of unique epigenetic profiles in the DeepSEA dataset (919).

Polyadenylation sites prediction. We downloaded the APARENT (Bogard *et al.*, 2019) dataset from [7] for polyadenylation site prediction. This dataset describes how often a specific nucleotide sequence is recognized as a polyadenylation signal by transcription machinery. We extracted target values and train/test splits using the scripts provided by the authors. We also extracted APARENT predictions (stored as field *iso_pred*) to benchmark the performance of the APARENT model. The sequences for upstream and downstream segments of 5'-untranslated regions were tokenized independently and separated by a SEP token. This task is a regression with a single target for 256-bp sequences.

2.2 Models architecture and training

2.2.1 DNA language models based on transformer architecture

We trained several transformer models reproducing and extending the BERT (Devlin *et al.*, 2019) and the BigBird (Zaheer *et al.*, 2020) architectures, which are all referred to as `GENA-LM` through the manuscript. The main differentiations between architectures are presented in the Table 2. More details and exact combinations of the aforementioned parameters for each model are provided in Supplementary Table 2[8]. We modified BERT-based models with Pre-Layer normalization (Xiong *et al.*, 2020), for some models we explicitly specify *lastln* in models naming that the layer norm is also applied to the last layer output (see exact parameters in Supplementary Table 2).

All models were trained to solve the masked language modeling (MLM) task. The sequence was tokenized and inserted between the special tokens CLS and SEP. Following the BERT pre-training procedure, we use 15% random tokens for predictions: in 80% of the cases they are replaced by MASK tokens, in 10% they are replaced by random tokens, and in 10% the same token is kept. The models were trained for 1-2 million steps with batch size 256 using 8 or 16 NVIDIA A100 GPUs. We use Nvidia Apex [9] FusedAdam implementation of AdamW (Loshchilov and Hutter, 2019) optimizer, the initial learning rate is 1×10^{-4} with warm-up. We used linear learning rate decay for most of the models. In the case of pre-training divergence, we manually decayed the learning rate.

2.2.2 GENA-LM finetuning

By default, we tokenized input sequences and flank them by service tokens CLS and SEP to process with GENA-LM. We padded or truncated the sequence when this was required to fit into the mode's input. If the structure of data requires specific tokenization, the preprocessing procedure is explained in the corresponding dataset paragraph.

The tokenized sequence was used as an input to a downstream model, composed of one of the pre-trained GENA-LM architectures and a single fully-connected layer. This fully-connected layer has the shape of `(hidden_size, target_size)`, where `hidden_size` is the size of the hidden unit specific for the GENA-LM model (as provided in Supplementary Table 2) and `target_size` is provided in the description of each of the downstream task datasets above. For single-label classification tasks, we used softmax as the activation function on the last layer and cross-entropy as the loss. For multi-label classification tasks, we used sigmoid as the activation function on the last layer and binary cross-entropy with logits as the loss. For regression tasks, the activation function on the last layer was not applied and mean squad error was used as the loss. To solve sequence classification and regression tasks we used the hidden state of the CLS token from the last layer. We used all hidden states from the last layer to solve the token-lvl classification task (splice site prediction). Both the last fully connected layer weights and all GENA-LM parameters were updated during finetuning. We use learning rate warm-up for all tasks (Goyal *et al.*, 2017). The number of training and warm-up steps was selected empirically for each task separately.

3 Results

3.1 Development of the GENA-LM models

This study presents the development of a novel, universal transformer model for nucleic acid sequence analysis, introducing several improvements compared to existing models like DNABERT (Ji *et al.*, 2021) or BigBird (Zaheer *et al.*, 2020). Firstly, we employed Byte-Pair Encoding (BPE) for sequence tokenization (Fig.1, A). In brief, the BPE algorithm constructs a sequence dictionary that identifies the most frequent subsequences in the genome, generating tokens of varying lengths, from one to 64 base pairs, with a median token length of 9 in our experiments (Fig.1, C). Intriguingly, our observation showed that the BPE vocabulary encompasses biologically significant tokens. For instance, the lengthiest tokens represent well-known repetitive elements such as LINE or simple repeats (Fig. 1, D). The tokenization of the input sequence is greedy; the dictionary initially searches for the longest sequence, and if found, it is converted into a token. The use of BPE instead of overlapping k-mers

[7] `https://github.com/johli/aparent`

[8] `https://github.com/AIRI-Institute/GENA_LM/blob/main/manuscript_data/Suplementary_Table_2.csv`

[9] `https://github.com/NVIDIA/apex`

Table 2. Characteristics of GENA-LM foundational DNA language models. The pre-trained GENA-LM models vary in terms of pre-training data, the number of layers, the type of attention, and sequence length. The same naming convention is maintained for models uploaded to the HuggingFace model hub, all prefixed with `AIRI-Institute/gena-lm-`. BERT-based models employ Pre-Layer Normalization (Xiong et al., 2020), and `lastln` explicitly denotes that layer normalization is also applied to the final layer. T2T split v1 refers to preliminary experiments with a non-augmented T2T human genome assembly split. 'DS Sparse' refers to the DeepSpeed sparse attention implementation, while 'HF Sparse' refers to the HuggingFace BigBird implementation. 'RoPE' indicates the use of rotary position embeddings (Su et al., 2021) in place of BERT-like absolute positional embeddings. The models have been trained with 12 (BERT-12L) and 24 (BERT-24L) layers, comprising a total of 110M and 336M parameters respectively.

Model	Architecture	Maximum seq len, tokens (≈ bp)	Tokenizer data	Training data
DNABERT	BERT-12L	512 (512)	3,4,5,6-mer	GRCh38.p13 (GENCODE release 33)
GENA-LM models:				
bert-base	BERT-12L	512 (4500)	T2T split v1	T2T split v1
bert-base-t2t	BERT-12L	512 (4500)	T2T+1000G SNPs+Multispieces	T2T+1000G SNPs
bert-base-lastln-t2t	BERT-12L	512 (4500)	T2T+1000G SNPs+Multispieces	T2T+1000G SNPs
bert-base-t2t-multi	BERT-12L	512 (4500)	T2T+1000G SNPs+Multispieces	T2T+1000G SNPs+Multispieces
bert-large-t2t	BERT-24L	512 (4500)	T2T+1000G SNPs+Multispieces	T2T+1000G SNPs
bigbird-base-sparse	BERT-12L, DS Sparse Att, RoPE	4096 (36000)	T2T split v1	T2T split v1
bigbird-base-sparse-t2t	BERT-12L, DS Sparse Att, RoPE	4096 (36000)	T2T+1000G SNPs+Multispieces	T2T+1000G SNPs
bigbird-base-t2t	BERT-12L, HF Sparse Attention	4096 (36000)	T2T+1000G SNPs+Multispieces	T2T+1000G SNPs

allows for the inclusion of longer sequence fragments. For instance, 512 6-mers encode 512 base pairs, while 512 BPE tokens encode around 4.5 kb, a key consideration for large and complex genomes such as the human genome. The model's resolution, however, is limited to individual tokens, which might be drawback of this approach.

Our second improvement involved the use of several implementations of the attention mechanism. In the base models, we employed a classic attention mechanism allowing the model to learn interconnections between each pair of tokens in the input sequences. In the sparse models, we utilized a sparse attention mechanism. This mechanism increases the length of the input sequence by limiting the total number of connections but still preserving the ability to learn interconnections between distant sequence elements.

These two improvements, BPE tokenization and sparse attention mechanism, enabled us to construct models with input sequences of 4.5 kb (512 tokens, full attention) and 36 kb (4096 tokens, sparse attention). This considerably exceeds the 512 bp inputs of DNABERT(Ji *et al.*, 2021), the only publicly available universal transformer model for DNA sequences.

For model training, we employed a masked language modeling task, a standard technique in natural language processing where the model is trained to predict a masked token based on its sequence context. We trained all models using the latest human T2T genome assembly - a unique feature of our experiment design as previous research used the hg38 genome (Ji *et al.*, 2021; Zaheer *et al.*, 2020). To minimize overfitting on the reference genome, we augmented it with common variants derived from the 1000-genome project database for some model options. In another training setup, we included genomes from different species, including standard model organisms like mice, fruit flies, nematode worms, baker's yeast, and other species spanning all eukaryotic taxa (for details see section 2 Methods).

Throughout the manuscript, we refer to all developed models as GENA language models (GENA-LM), with each specific model being named according to its label provided in Table 2. Each model has its own strengths and limitations, but we would like to spotlight the *gena-lm-bert-base-t2t* model, which replicates the BERT transformer architecture and can be considered as a baseline for other models; the *gena-lm-bert-large-t2t* model, possessing the highest number of parameters (336M) and an input size of 4.5 kb; and the *gena-lm-bigbird-base-sparse-t2t* models, which have fewer parameters than the *gena-lm-bert-large-t2t*, but a larger input sequence length (36 kb). We note that the *gena-lm-bert-base* model was developed during our initial trials using a tokenizer, train/test split, and other parameters that slightly differ from other models. Despite these differences, we chose to include this model in our comparisons as it is the first model publicly released by our group, and we wish to inform the research community of its capabilities.

When comparing the performance of the models on the masked language modeling task (Fig. 1, E), we found that the sparse attention models achieve higher accuracy than their relatively shorter (512 tokens) full-attention counterparts. This suggests that contextual information plays a significant role in this training task. An increase in model size also contributes to better pre-training metrics. However, the masked language model task may not compel models to utilize the full spectrum of dependencies present in biological data. Thus, our aim was to thoroughly assess the performance of GENA-LMs in various biologically meaningful challenges to understand their strengths and limitations.

3.2 GENA-LM performance on different genomic tasks

Next, our goal was to benchmark the pre-trained foundational DNA language models against various biological tasks (Fig. 1, B). We chose a selection of critical genomic challenges that have recently been tackled using artificial intelligence. These include: 1) predicting polyadenylation site strength; 2) estimating the promoter activity of a DNA sequence; 3) forecasting splicing sites; 4) predicting chromatin profiles, including histone modifications, DNAse I hypersensitivity sites, transcription factor binding sites, and more. These tasks (1-4) were all evaluated using human genomic datasets. To test the applicability of the model pre-trained on human data to non-vertebrate species, we added a fifth task: predicting the activity of housekeeping and developmental enhancers in a STARR-seq assay for Drosophila sequences.

For each challenge, we set up several benchmarks. Firstly, we fine-tuned the publicly available DNABERT transformer model. Secondly, we utilized the performance scores of state-of-the-art machine-learning models specifically developed for each of the selected challenges. These include SpliceAI (Jaganathan *et al.*, 2019) for splicing, APARENT (Bogard *et al.*, 2019) for polyadenylation, DeepSEA (Zhou and Troyanskaya, 2015) for chromatin profiles and gene expression, and DeepSTARR (de Almeida *et al.*, 2022) for predicting enhancer activity in Drosophila cells. We were careful in composing the training, validation, and test sets for each challenge and only compared our models with state-of-the-art methods when we could accurately reproduce the composition of these datasets. Nevertheless, we did allow discrepancies in input sequence length: for example, the original DeepSEA model was trained to predict

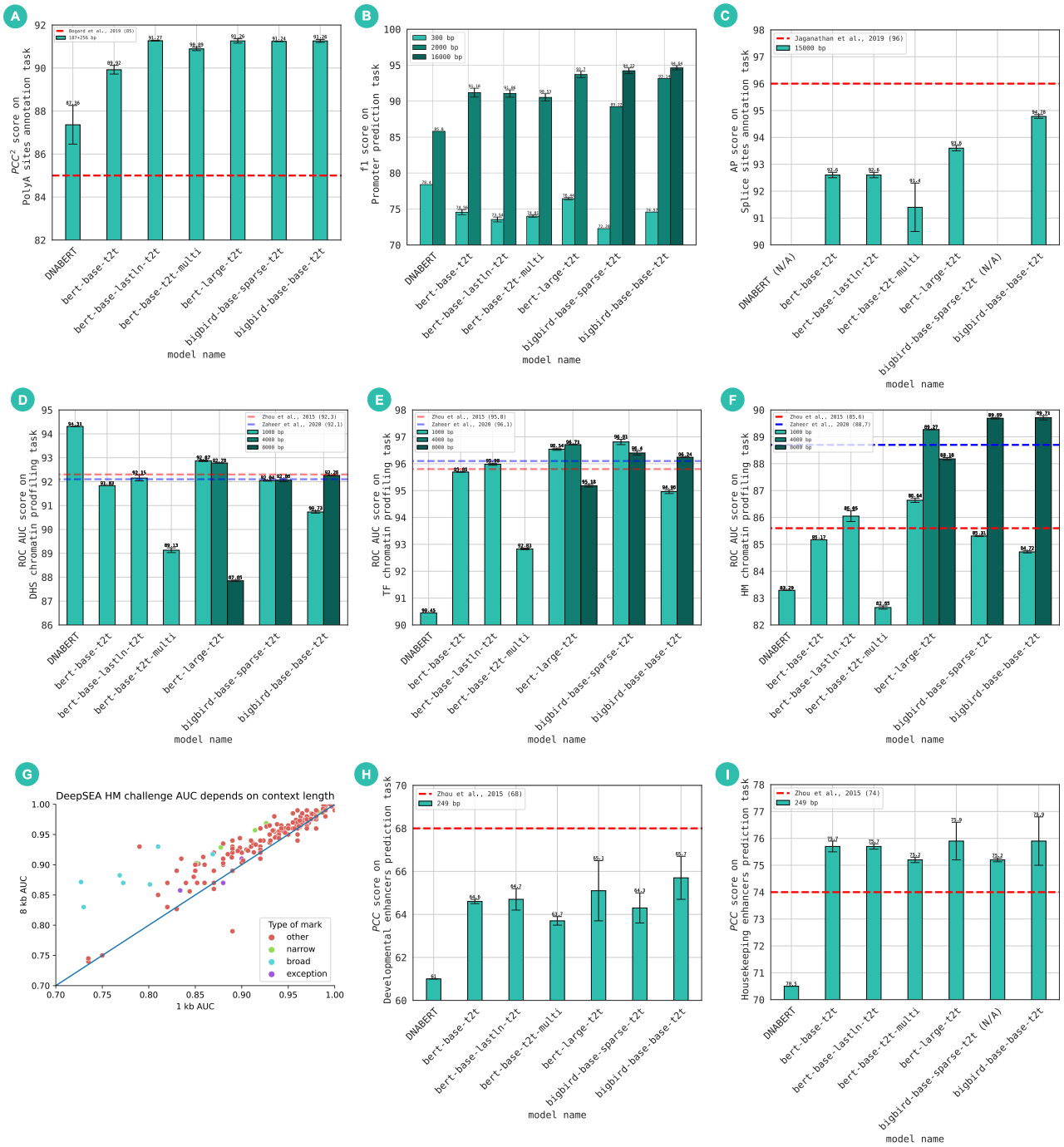

Fig. 2. GENA-LMs achieve state-of-the-art performance across various genomic tasks. A. Performance scores for polyadenylation site prediction. B. Promoter prediction. C. Splice site annotation. D-F. Epigenetic feature annotation. G. Comparison of AUC scores for the prediction of chromatin occupancy, detailed for each individual chromatin mark, between GENA-LM trained on 1-kb and 8-kb contexts. H-I. Classification of Drosophila enhancers: developmental (H) and housekeeping (I). For each panel, the Y-axis denotes the metric used for scoring the models, and the dashed vertical line represents the metric of the state-of-the-art model reported in previous literature. Each model was fine-tuned three times using different random seeds, with error bars reporting the standard deviation.

chromatin occupancy of 256 bp loci using an 800 bp context. In contrast, we also provided a model trained on 8 kb sequences. This approach allowed us to demonstrate how the length of the DNA context influences the quality of the models.

Prediction of polyadenylation site strength. Polyadenylation of DNA is a crucial process, with variants impacting polyadenylation site selection known to cause diseases. Recently, (Bogard *et al.*, 2019) developed a reporter assay to measure the strength of the proximal polyadenylation signal for millions of short sequences. They demonstrated that the strength of the polyadenylation signal has specific nucleotide determinants and can be inferred from the DNA sequence using the convolutional neural network APARENT. We fine-tuned GENA-LMs to perform the task of predicting polyadenylation signal strength. As evident from Fig. 2, A, all GENA-LM architectures significantly outperform APARENT (best GENA-LM

Pearson $R^2 = 0.91 \pm 0.0002$ vs APARENT Pearson $R^2 = 0.85$). The DNABERT model, fine-tuned on the same dataset, also slightly surpassed APARENT ($R^2 = 0.87 \pm 0.01$ vs. APARENT Pearson $R^2 = 0.85$), albeit scoring significantly lower than GENA-LMs. These results suggest that the GENA-LM architecture can achieve state-of-the-art results for certain genomic tasks. However, sequences profiled in the polyadenylation signal strength assay are relatively short (187 bp proximal part plus 256 bp distal part) and may not fully exploit the advantages of long-input GENA-LMs. Therefore, we decided to test these models against challenges where the benefits of a longer context can be assessed.

Promoter activity prediction. We benchmarked machine-learning models on promoter sequences taken from the EPD dataset and non-promoter control samples, discovering that expanding the input sequence length from 300 bp to 16 kb notably enhanced model performance, as depicted in Fig. 2, B. For the shorter 300 bp sequences, the DNABERT architecture achieved the best result (F1 score of 78.5), slightly surpassing the best GENA-LM result (F1 score of 76.44 $\pm$ 0.16). However, for longer sequences, GENA-LM demonstrated markedly superior performance, attaining an F1 score of 93.7 $\pm$ 0.44 for 2 kb inputs. This score was substantially greater than that for the DNABERT model fine-tuned on the same input sequence length (F1 score of 85.8).

Upon comparing the performance of the GENA-LMs, we discovered that for promoter activity prediction task: 1) GENA-LM models with more parameters surpassed models with fewer parameters (i.e., *gena-lm-bert-large-t2t* outperformed *gena-lm-bert-base-t2t*); 2) models that allowed for longer input sequence length due to the sparse attention mechanism outperformed traditional full-attention BERT models (i.e., *gena-lm-bigbird-base-sparse* surpassed *gena-lm-bert-base-t2t*). However, models with shorter inputs and more parameters outperformed both (i.e., *gena-lm-bert-large-t2t* was superior to *gena-lm-bigbird-base-sparse*); 3) multispecies training (i.e., enriching the dataset with genomic sequences other than human during pretraining) did not improve performance (*gena-lm-bert-base-t2t-multi* versus *gena-lm-bert-base-t2t*). Importantly, all GENA-LM models surpassed DNABERT results, presumably because the BPE tokenization strategy enables these models to process sequences of characteristic lengths of around 4-5 kb, while DNABERT splits long sequences into 512 bp chunks and processes these segments independently.

We observed that the model with more parameters outperformed the sparse model with a longer input sequence length. However, this could be due to the fact that the 2 kb sequence can be processed by both short- and long-input models without truncation. To investigate whether extending the input length further (i.e., providing more context) would result in even higher promoter classification accuracy, we tested the models with a 16 kb context. As hypothesized, for these longer inputs, the maximal performance exceeded any performance seen for any model on the 2 kb dataset: F1 score of 94.64 $\pm$ 0.3 for the *gena-lm-bigbird-base-t2t* model on the 16 kb dataset versus an F1 score of 93.7 $\pm$ 0.44 for the *gena-lm-bert-large-t2t* on the 2 kb dataset. This result underscores the significance of contextual information for this task. Notably, this high score was obtained using a sparse-attention model with fewer parameters than the model that displayed the best performance on the 2 kb dataset. Therefore, this task demonstrates that increasing context length can be more beneficial than the number of parameters.

Splice site annotation. We then fine-tuned GENA-LMs to predict splice-donor and splice-acceptor sites in the human genome, comparing the results with those of the state-of-the-art SpliceAI model as shown in Fig. 2, C. This task also necessitated the processing of extensive contexts: a total input of 15 kb, where the 5 kb target region in the middle is flanked by 5 kb context sequences on both sides. We observed that the task-specific convolutional neural network SpliceAI slightly outperformed GENA-LM models on this dataset (mean PR AUC of 0.960 vs. 0.947 $\pm$ 0.002).

For this task architectures with longer sequence input (*gena-lm-bigbird-base-t2t*) performed better than models with shorter sequence input, even when the latter possessed more parameters (*gena-lm-bert-large-t2t*). These results corroborate our previous conclusion that longer contextual information might be a good trade-off to the number of parameters. Analogous to our findings with promoters, we did not observe any performance enhancement when using multispecies models (*gena-lm-bert-base-t2t-multi* mean PR AUC of 0.914 vs. *gena-lm-bert-base-t2t* mean PR AUC of 0.926).

Prediction of chromatin profiles. Predicting the epigenetic states of a locus from its sequence is another pervasive challenge in genomics. We utilized the widely recognized DeepSEA dataset to evaluate the capacity of GENA-LM transformers to tackle this challenge (Fig. 2, D-F). This dataset contains over 900 cell-type specific chromatin profiles, categorized as DNAse I hypersensitivity sites (DHS), histone marks (HM), and transcription factor binding sites (TF). In the original DeepSEA challenge, the signals of these chromatin marks were predicted for each 200 bp genomic bin, using its sequence and the sequence of the 800 bp context ($\pm$400 bp flanking regions).

Applying GENA-LMs to this task, we found that transformer models significantly outperformed the results previously obtained using the convolutional neural network DeepSEA. Moreover, for TF and DHS profiles, the results obtained by GENA-LMs surpassed the metrics reported for the BigBird architecture, despite the latter utilizing a larger 8 kb context (best GENA-LM average AUC on 1 kb for TF: 96.81 $\pm$ 0.1 vs. BigBird: 96.1; for DHS: 92.8 $\pm$ 0.03 vs. BigBird: 92.3). The GENA-LMs scores were also comparable to or higher than those recently reported by (Dalla-Torre *et al.*, 2023) for the Nucleotide Transformer, and superior to the DNABERT architecture trained on 1 kb input lengths.

To facilitate a more accurate comparison between BigBird and GENA-LM architectures, we reprocessed the DeepSEA dataset to include additional context sequences, thereby achieving an input length similar to the BigBird dataset (8000 bp). Interestingly, increasing context size impacted the prediction of epigenetic profiles in various ways. For histone marks, we observed a substantial increase in performance, resulting in an AUC of 89.71$\pm$ 0.08, significantly higher than for shorter context (86.64$\pm$ 0.08), original DeepSEA results (85.6), or BigBird performance (88.70). Conversely, enlarging the input length only slightly affected performance for TF and DHS.

We scrutinized how AUC varied for individual histone marks to ascertain which epigenetic profiles contributed to the AUC increase. Notably, we identified a significant distinction between narrow and broad histone marks: while the former displayed the same AUC for 1 kb and 8 kb models, the latter showed a drastic increase with enlarging context length (Fig. 2, G). These findings support our earlier suggestion (Sindeeva *et al.*, 2023) that broad histone marks require a larger context for accurate prediction and underscore the importance of large-input models for this task.

The performance metrics of different GENA-LMs for various epigenetic profiles and contexts clearly demonstrate that no single model achieves optimal performance across all tasks. For TFs, *gena-lm-bigbird-base-sparse-t2t* yields the best performance on 1 kb inputs, with an increase in input size resulting in only a slight decrease in performance. For DHS, the optimal model is *gena-lm-bert-large-t2t*, which has the highest number of parameters; however, extending the context length unexpectedly causes a substantial drop in performance. Lastly, for HM, the best strategy is to process a longer context using the *gena-lm-bigbird-base-t2t* model. Overall, a combination of GENA-LMs outperforms existing models (BigBird, DNABERT, or Nucleotide Transformer), achieving

state-of-the-art results for this task. We hypothesize that the observed variance is likely a consequence of the design of the loss function, which doesn't prioritize maximizing the prediction accuracy of a particular type of chromatin marks.

Prediction of enhancer activity in Drosophila cells. So far, our tasks were exclusive to human data. We next sought to determine whether GENA-LMs could be extended to other species. For this, we engaged the recently published DeepSTARR dataset (de Almeida *et al.*, 2022), which reports enhancer activity for millions of short sequences measured in *Drosophila* cells. Each sequence was classified according to its housekeeping (i.e., cell-type unspecific) and developmental (i.e., cell-type specific) enhancer strength. The authors demonstrated that the convolutional neural network DeepSTARR could predict these activities based on nucleotide determinants. When we applied the GENA-LMs to this task, we obtained mixed results (Fig. 2, H-I). For the developmental enhancers, the task-specific DeepSTARR convolutional neural network outperformed GENA-LMs (best GENA-LM Pearson R = 0.657 $\pm$ 0.01 vs. DeepSTARR reported 0.68); however, for the housekeeping enhancers, the outcome was reversed, with GENA-LM outperforming DeepSTARR (best GENA-LM Pearson R = 0.768 $\pm$ 0.01 vs. DeepSTARR reported 0.74). Importantly, GENA-LMs demonstrated superior scores for both tasks in comparison to DNABERT and the Nucleotide Transformer.

While this challenge was based on non-human sequencing data, we did not observe any performance improvement with the multispecies model. The highest performing GENA-LM results were achieved by the *gena-lm-bert-large-t2t* model, which has the most parameters. Therefore, we concluded that GENA-LM models trained on human data produce sequence embeddings that are sufficiently universal to be applied in non-human genomic tasks.

4 Conclusions

Transformer model architectures have recently garnered attention across a broad spectrum of research and technology fields, including genomic studies. These architectures exhibit state-of-the-art performance in a variety of biological tasks, such as learning gene expression regulation in mammals (Avsec *et al.*, 2021) and *E. coli* (Clauwaert *et al.*, 2021), predicting phenotypes based on gene expression (Khan and Lee, 2021) (Zhang *et al.*, 2022), inferring DNA methylation (Le and Ho, 2022), completing missing genotypes (Mowlaei *et al.*, 2023), among others. However, these task-specific models require training from scratch for each unique biological problem, a process that is both time and resource-consuming. Pre-trained foundational DNA models, such as DNABERT, mitigate this issue by enabling the fine-tuning of publicly available multi-purpose models. Notable models developed since the original DNABERT (Ji *et al.*, 2021) include BigBird (Zaheer *et al.*, 2020) by Google and the Nucleotide Transformer (Dalla-Torre *et al.*, 2023) by InstaDeep. However, BigBird does not publicly provide its trained model and the Nucleotide Transformer does not support sequence inputs longer than 6 kb. To bridge this gap, we present GENA-LMs which comprise several models that support the longest input size among publicly available DNA transformers [10]. Additionally, the GENA-LM suite offers a range of publicly available architectures, allowing researchers to select the best model for their specific task. Our intensive benchmarking demonstrated that GENA-LMs consistently outperform previously published pre-trained models and, on occasion, even task-specific convolutional neural networks.

Through comparing different models, we explored the impacts of context length and the number of model parameters on prediction performance. We discovered that the trade-off between these two factors is task-dependent. A longer context is crucial for predicting promoter activity or inferring broad histone mark distribution, as previously suggested (Sindeeva *et al.*, 2023). For other tasks, a shorter context is sufficient, thus favoring an increase in the number of model parameters. The diverse array of GENA-LM models enables users to tailor their model selection to the task at hand.

While GENA-LMs, to our knowledge, support the largest input size among publicly available foundational DNA language models, they still do not reach the size required for maximal accuracy in some biological tasks. For instance, it has been demonstrated that gene expression can be affected by variants located hundreds or even millions of base pairs away from the promoter, due to loop extrusion (Kabirova *et al.*, 2023) or other 3D-genomic mechanisms (Fishman *et al.*, 2018). Several strategies can help circumvent this limitation of GENA. Firstly, the 3D proximity of chromatin could be inferred using specialized models (Belokopytova and Fishman, 2021b), which can then be explicitly passed to the transformer models. Secondly, BPE tokenization constrains the granularity of model predictions to tokens. Alternative approaches to DNA tokenization and the creation of low-level nucleotide embeddings could address this limitation for some applied tasks. Lastly, a more elegant approach would involve increasing the model's input size to allow the analysis of longer DNA sequences. For this purpose, future iterations could incorporate recurrent memory transformers (Bulatov *et al.*, 2022) or other memory-based architectures with GENA-LM.

[10] Models with `gena-lm-` prefix: `https://huggingface.co/AIRI-Institute/`

References

Ainslie, J. *et al.* (2020). Etc: Encoding long and structured data in transformers.

Avsec, Ž. *et al.* (2021). Effective gene expression prediction from sequence by integrating long-range interactions. *Nature methods*, **18**(10), 1196–1203.

Belokopytova, P. and Fishman, V. (2021a). Predicting genome architecture: Challenges and solutions. *Frontiers in Genetics*, **11**.

Belokopytova, P. and Fishman, V. (2021b). Predicting Genome Architecture: Challenges and Solutions. *Front. Genet.*, **11**.

Belokopytova, P. S. *et al.* (2020). Quantitative prediction of enhancer–promoter interactions. *Genome research*, **30**(1), 72–84.

Beltagy, I. *et al.* (2020). Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*.

Bogard, N. *et al.* (2019). A deep neural network for predicting and engineering alternative polyadenylation. *Cell*, **178**(1), 91–106.e23.

Bulatov, A. *et al.* (2022). Recurrent memory transformer. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, volume 35, pages 11079–11091. Curran Associates, Inc.

Bulatov, A. *et al.* (2023). Scaling transformer to 1m tokens and beyond with rmt. *arXiv preprint arXiv:2304.11062*.

Choromanski, K. M. *et al.* (2021). Rethinking attention with performers. In *International Conference on Learning Representations*.

Clauwaert, J. *et al.* (2021). Explainability in transformer models for functional genomics. *Briefings Bioinf.*, **22**(5), bbab060.

Dai, A. M. and Le, Q. V. (2015). Semi-supervised sequence learning. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc.

Dai, Z. *et al.* (2019). Transformer-XL: Attentive language models beyond a fixed-length context. In *Proceedings of the 57th Annual Meeting of the*

Association for Computational Linguistics, pages 2978–2988, Florence, Italy. Association for Computational Linguistics.

Dalla-Torre, H. *et al.* (2023). The Nucleotide Transformer: Building and Evaluating Robust Foundation Models for Human Genomics. *bioRxiv*, page 2023.01.11.523679.

de Almeida, B. P. *et al.* (2022). DeepSTARR predicts enhancer activity from DNA sequence and enables the de novo design of synthetic enhancers. *Nat. Genet.*, **54**(5), 613–624.

Devlin, J. *et al.* (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *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)*, pages 4171–4186.

Fishman, V. S. *et al.* (2018). Interpreting Chromosomal Rearrangements in the Context of 3-Dimentional Genome Organization: A Practical Guide for Medical Genetics. *Biochemistry (Mosc.)*, **83**(4), 393–401.

Goyal, P. *et al.* (2017). Accurate, large minibatch SGD: training imagenet in 1 hour. *CoRR*, **abs/1706.02677**.

Guo, Q. *et al.* (2019). Star-transformer.

Howard, J. and Ruder, S. (2018). Universal language model fine-tuning for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 328–339, Melbourne, Australia. Association for Computational Linguistics.

Hutchins, D. *et al.* (2022). Block-recurrent transformers. In A. H. Oh, A. Agarwal, D. Belgrave, and K. Cho, editors, *Advances in Neural Information Processing Systems*.

Jaganathan, K. *et al.* (2019). Predicting splicing from primary sequence with deep learning. *Cell*, **176**(3), 535–548.e24.

Ji, Y. *et al.* (2021). DNABERT: pre-trained Bidirectional Encoder Representations from Transformers model for DNA-language in genome. *Bioinformatics*, **37**(15), 2112–2120.

Kabirova, E. *et al.* (2023). Function and Evolution of the Loop Extrusion Machinery in Animals. *Int. J. Mol. Sci.*, **24**(5), 5017.

Katharopoulos, A. *et al.* (2020). Transformers are RNNs: Fast autoregressive transformers with linear attention. In H. D. III and A. Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 5156–5165. PMLR.

Khan, A. and Lee, B. (2021). Gene Transformer: Transformers for the Gene Expression-based Classification of Lung Cancer Subtypes. *arXiv*.

Kim, S. and Wysocka, J. (2023). Deciphering the multi-scale, quantitative cis-regulatory code. *Molecular Cell*, **83**(3), 373–392. Reimagining the Central Dogma.

Kitaev, N. *et al.* (2020). Reformer: The efficient transformer. In *International Conference on Learning Representations*.

Le, N. Q. K. and Ho, Q.-T. (2022). Deep transformers and convolutional neural network in identifying DNA N6-methyladenine sites in cross-species genomes. *Methods*, **204**, 199–206.

Libbrecht, M. W. and Noble, W. S. (2015). Machine learning applications in genetics and genomics. *Nature Reviews Genetics*, **16**(6), 321–332.

Loshchilov, I. and Hutter, F. (2019). Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Mowlaei, M. E. *et al.* (2023). Split-Transformer Impute (STI): Genotype Imputation Using a Transformer-Based Model. *bioRxiv*, page 2023.03.05.531190.

Pan, S. J. and Yang, Q. (2010). A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, **22**(10), 1345–1359.

Penzar, D. *et al.* (2022). Legnet: resetting the bar in deep learning for accurate prediction of promoter activity and variant effects from massive parallel reporter assays. *bioRxiv*.

Peters, M. E. *et al.* (2018). Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Radford, A. *et al.* (2018). Improving language understanding with unsupervised learning. Technical report.

Rae, J. W. *et al.* (2020). Compressive transformers for long-range sequence modelling. In *International Conference on Learning Representations*.

Sean Whalen, Jacob Schreiber, W. S. N. . K. S. P. (2022). Navigating the pitfalls of applying machine learning in genomics. *Nature Reviews Genetics*, **23**, 169–181.

Sennrich, R. *et al.* (2016). Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Sindeeva, M. *et al.* (2023). Cell type–specific interpretation of noncoding variants using deep learning–based methods. *GigaScience*, **12**, giad015.

Su, J. *et al.* (2021). Roformer: Enhanced transformer with rotary position embedding. *ArXiv*, **abs/2104.09864**.

Vaswani, A. *et al.* (2017). Attention is All you Need. In *Advances in neural information processing systems*, pages 5998–6008.

Wu, Q. *et al.* (2022). Memformer: A memory-augmented transformer for sequence modeling. In *Findings of the Association for Computational Linguistics: AACL-IJCNLP 2022*, pages 308–318, Online only. Association for Computational Linguistics.

Xiong, R. *et al.* (2020). On layer normalization in the transformer architecture. In H. D. III and A. Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 10524–10533. PMLR.

Zaheer, M. *et al.* (2020). Big bird: Transformers for longer sequences. In H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 17283–17297. Curran Associates, Inc.

Zhang, T.-H. *et al.* (2022). Transformer for Gene Expression Modeling (T-GEM): An Interpretable Deep Learning Model for Gene Expression-Based Phenotype Predictions. *Cancers*, **14**(19), 4763.

Zhou, J. and Troyanskaya, O. G. (2015). Predicting effects of noncoding variants with deep learning-based sequence model. *Nat. Methods*, **12**(10), 931–934.