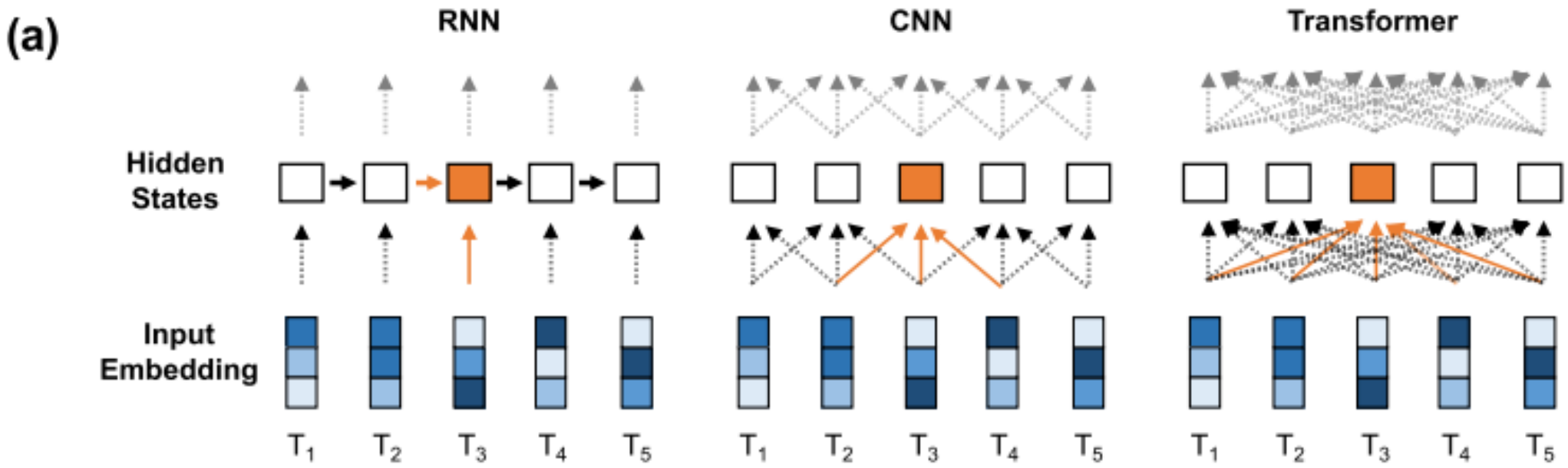
Foundation models for DNA sequences

Based on research internship at Shanghai AI lab.

Sun, Jianle

Modeling sequences

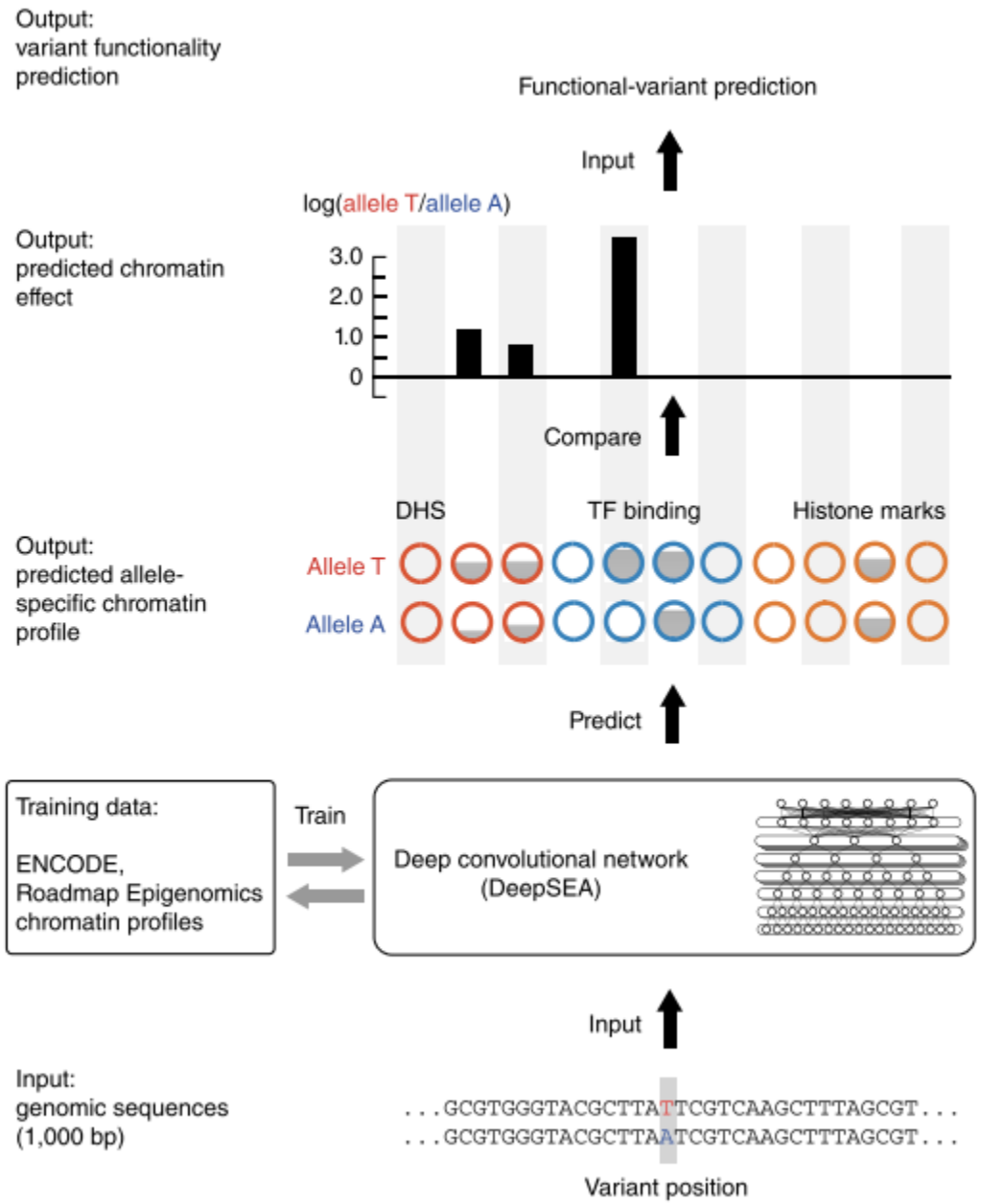


Two perspectives for DNA sequences

- Nucleotides as Pixels in images
  - CNN based model
  - Long sequences as input (~10kb-~mb)
  - Supervised (Mainly)
  - Task-driven: training on specific sequences (Mainly)
- Nucleotides as Words in natural languages
  - Transformer based (pretrained) model
  - Short sequences as input (~100bp-~kb)
  - Self-supervised: auto-encoding (mask language modeling), auto-regressive, encoder-decoder
  - General embedding: pretraining (on the whole genome) – finetune, few-shot, zero-shot

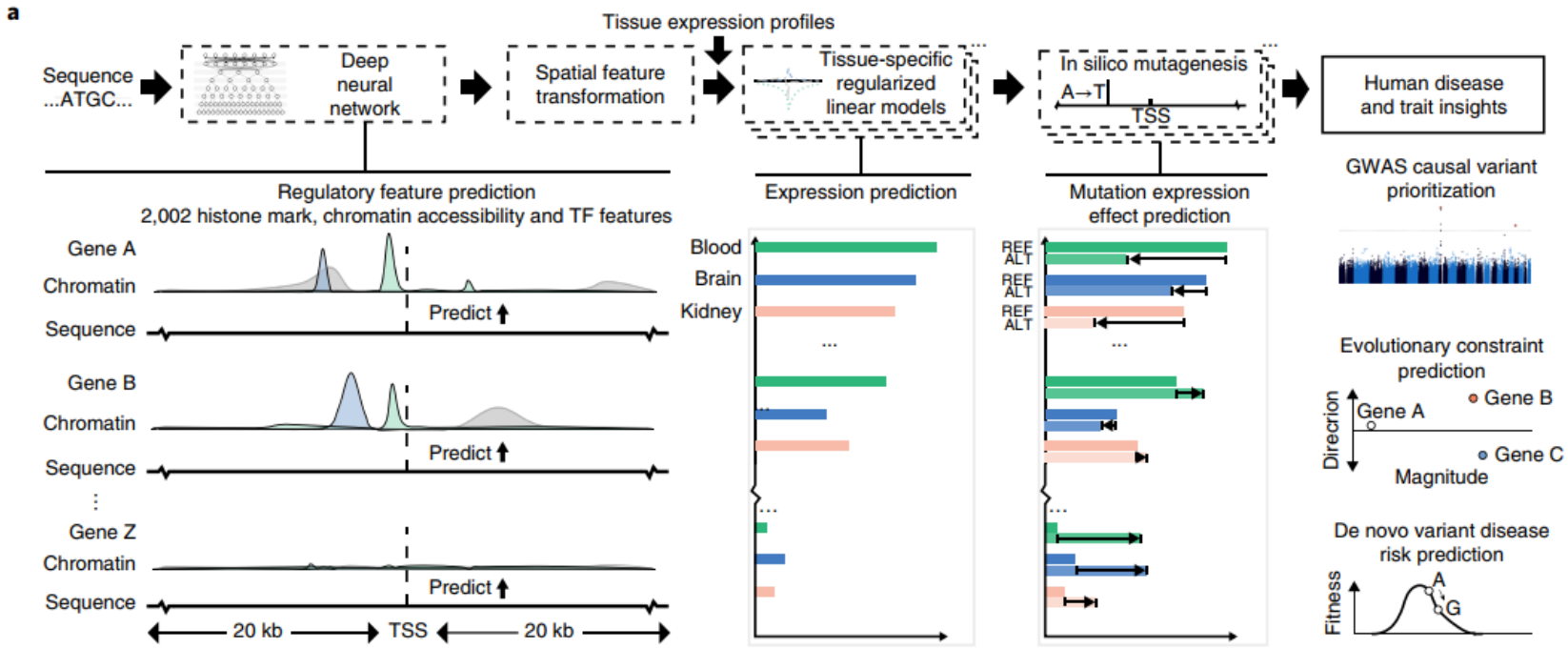
CNN based models

- DeepSEA (Nature Genetics, 2015)
- predict large-scale chromatin-profiling data, including 690 TF binding profiles for 160 different TFs, 125 DHS profiles and 104 histone mark profiles
- Each training sample consists of a 1,000-bp sequence centered on each 200-bp bin with the label for all 919 chromatin features; a chromatin feature was labeled 1 if more than half of the 200-bp bin is in the peak region and 0 otherwise.
- noncoding-variant (especially rare variant) effect prediction



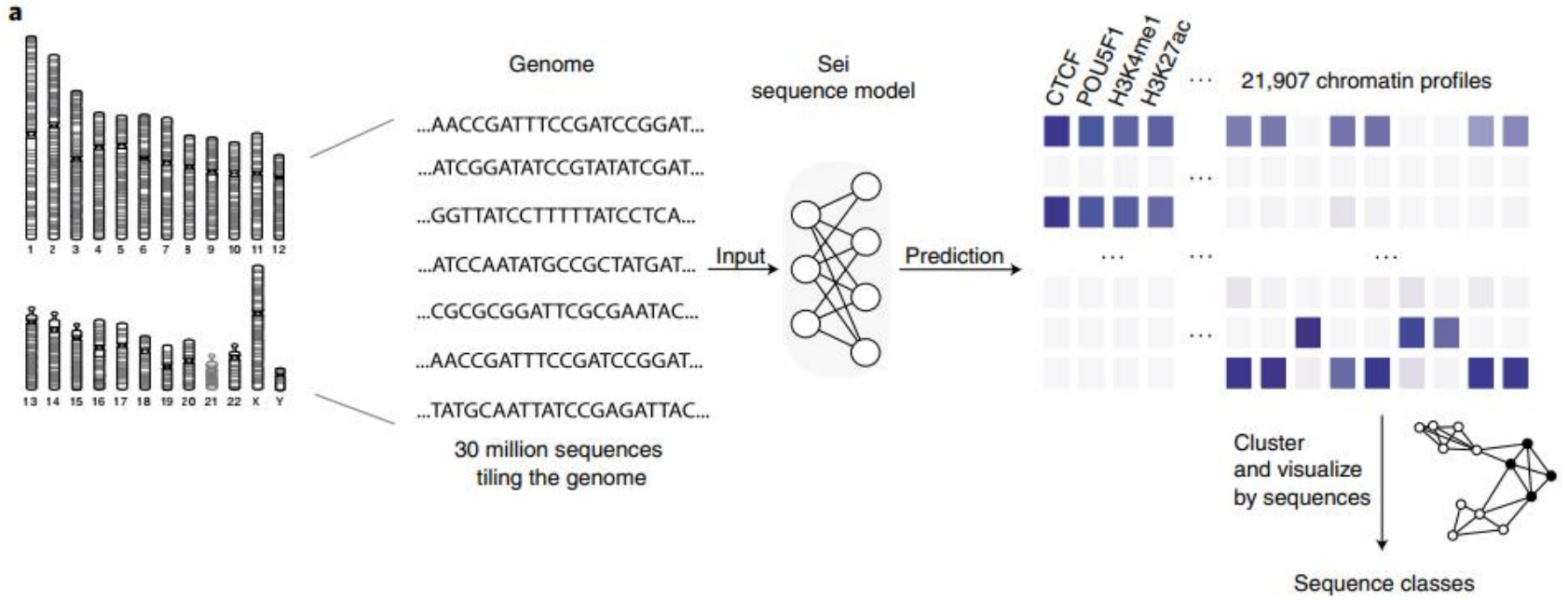
CNN based models

- Expecto (Nature Genetics, 2018)
- predicted the epigenomic features (2,002 different histone mark, transcription factor and DNA accessibility profiles for >200 tissues and cell types) of a 200-bp region, while also using the 1,800-bp surrounding context sequence
- Pol II–transcribed genes expression: scanned the genomic sequence between+20 kb upstream and –20 kb downstream to predict spatial chromatin organization patterns by using a moving window with a 200-bp step size, which yielded 200 spatial bins with a total number of 400,400 features



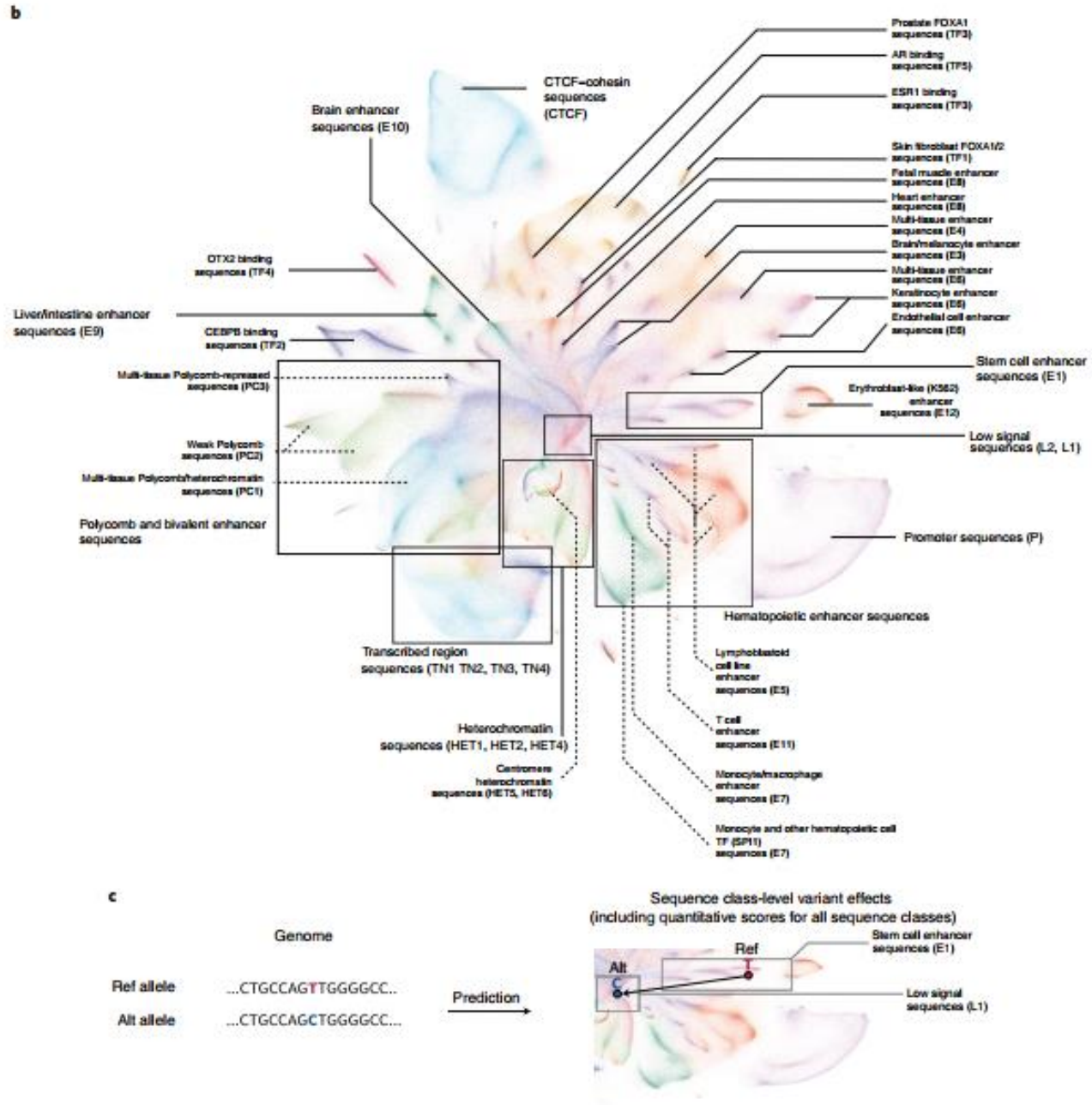
CNN based models

- Sei (Nature Genetics, 2022)
- takes as input a 4-kb length sequence and predicts the probabilities of 21,907 cis-regulatory targets (chromatin profiles across >1,300 cell lines and tissues) at the center position



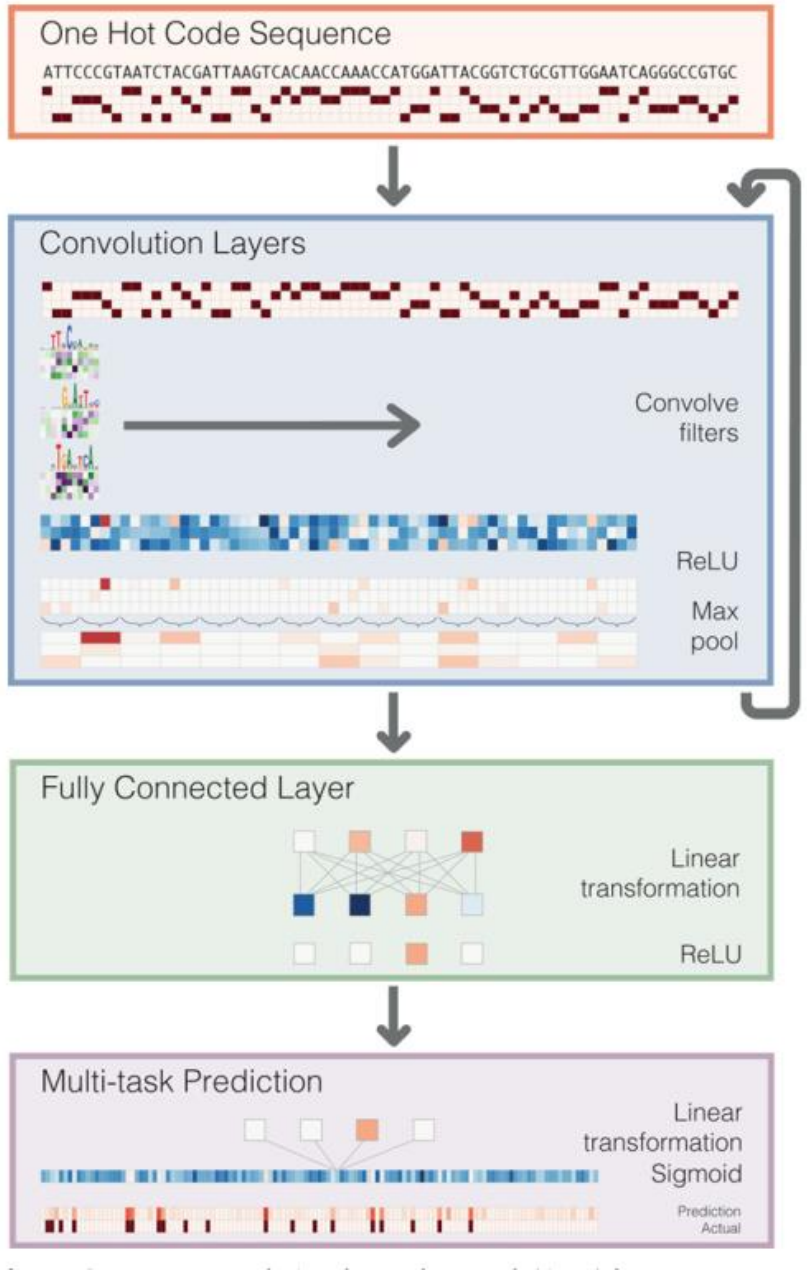
CNN based models

- Sei (Nature Genetics, 2022)
- Sequence classes provide a global classification and quantification of sequence and variant effects based on diverse regulatory activities, such as cell type-specific enhancer functions.



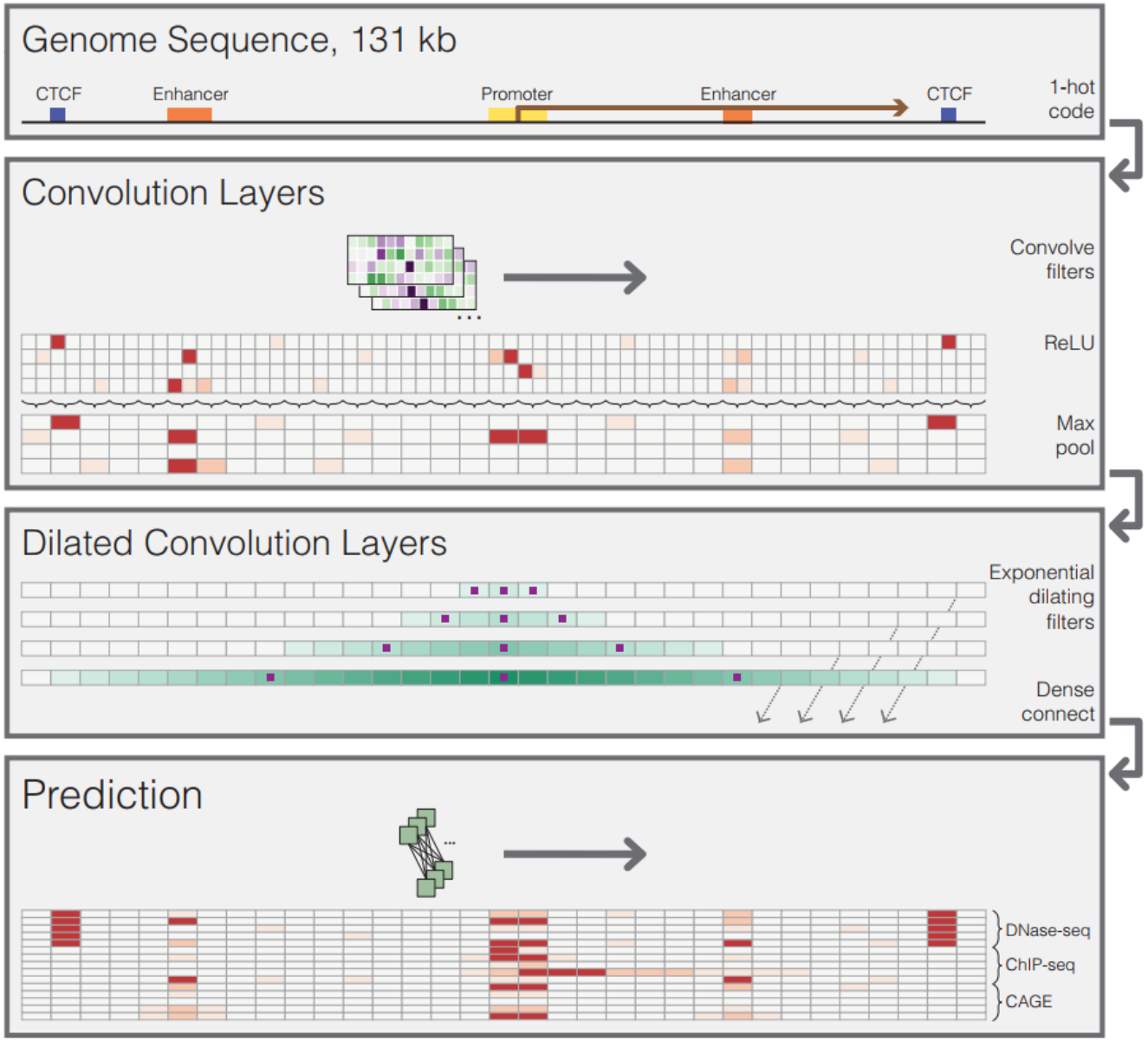
CNN based models

- Basset (Genome Research, 2016)
- trained Basset on a compendium of accessible genomic sites mapped in 164 cell types by DNase-seq
- the input data to training for each site include its 600-bp DNA sequence and a binary vector to indicate the presence of a significant peak in each of the 164 cell types (binary prediction)



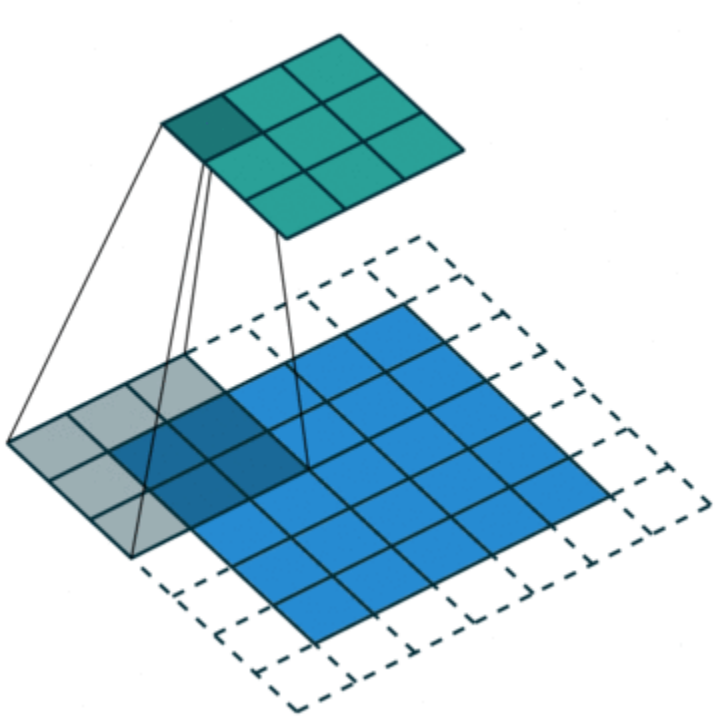
CNN based models

- Basenji (Genome Research, 2018)
- Modeling quantitative profile (a multitask Poisson regression on normalized counts of aligned reads to that region) instead of binary peak
- 131-kb regions as input, predict 529 unique cells/tissues profiled by DNase-seq, 1136 unique cells/tissues profiled by ChIP-seq, and 595 unique cells/tissues profiled by CAGE.
- Dilated Convolutions: model distal regulatory interaction

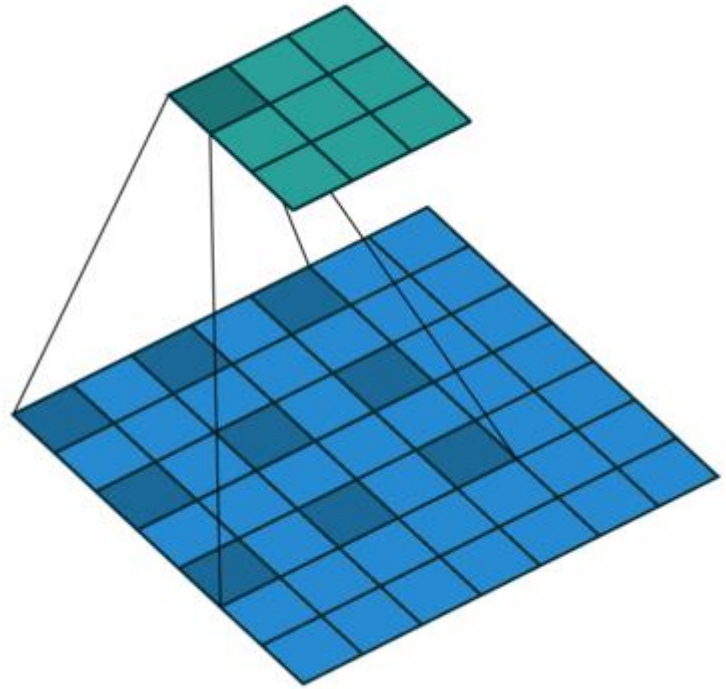


CNN based models

- Dilated Convolutions: increase Receptive Filed



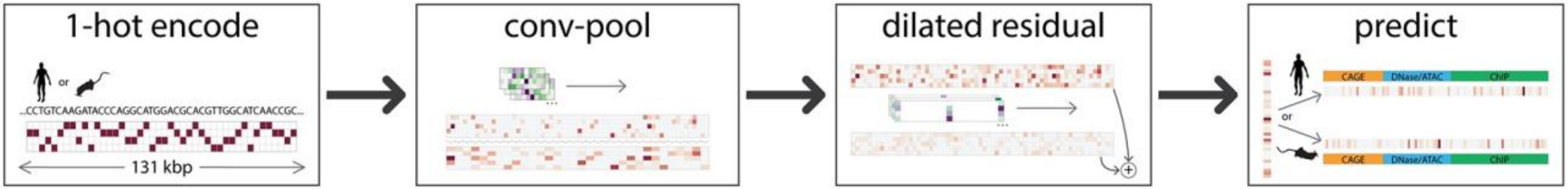
Standard Convolution (l=1)



Dilated Convolution (l=2)

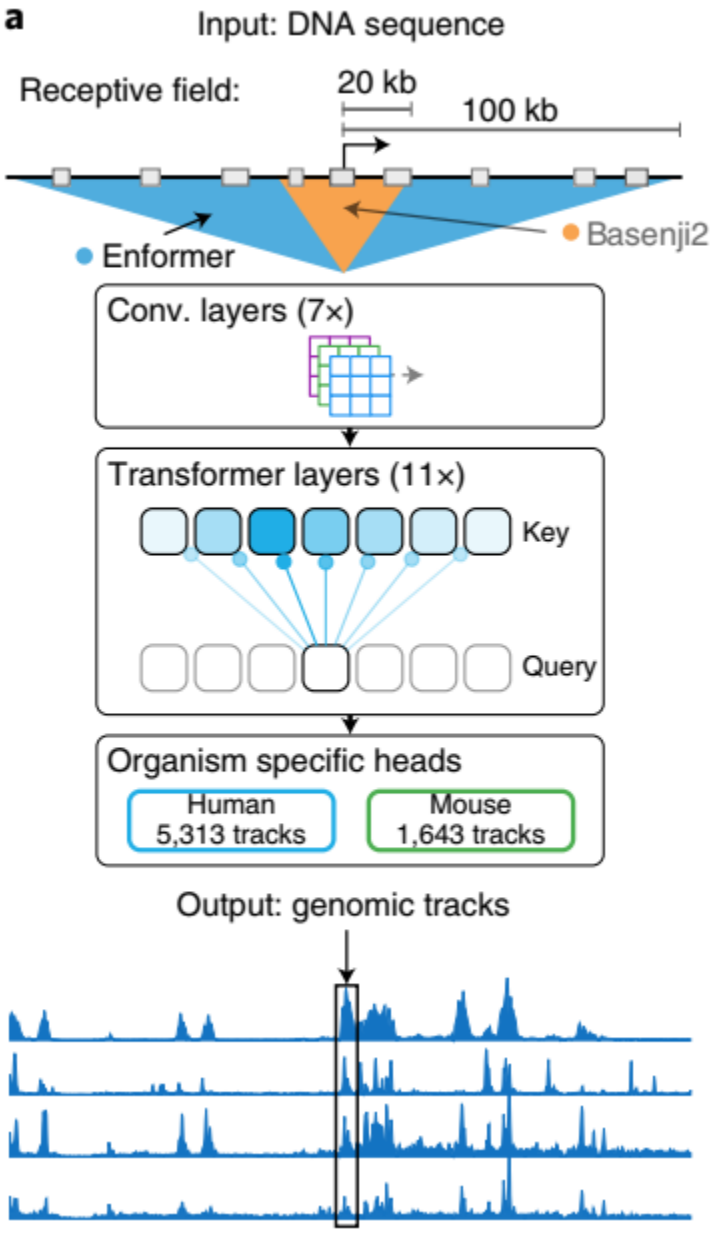
CNN based models

- Basenji2 (PLOS Computational Biology, 2020)
- The neural network takes as input a 131,072 bp sequence, transforms its representation with iterated convolution layers, and makes predictions in 128bp windows across the sequence for the normalized signal derived from many datasets
- training data consisting of 6,956 human and mouse quantitative sequencing assay signal tracks from the ENCODE and FANTOM consortiums



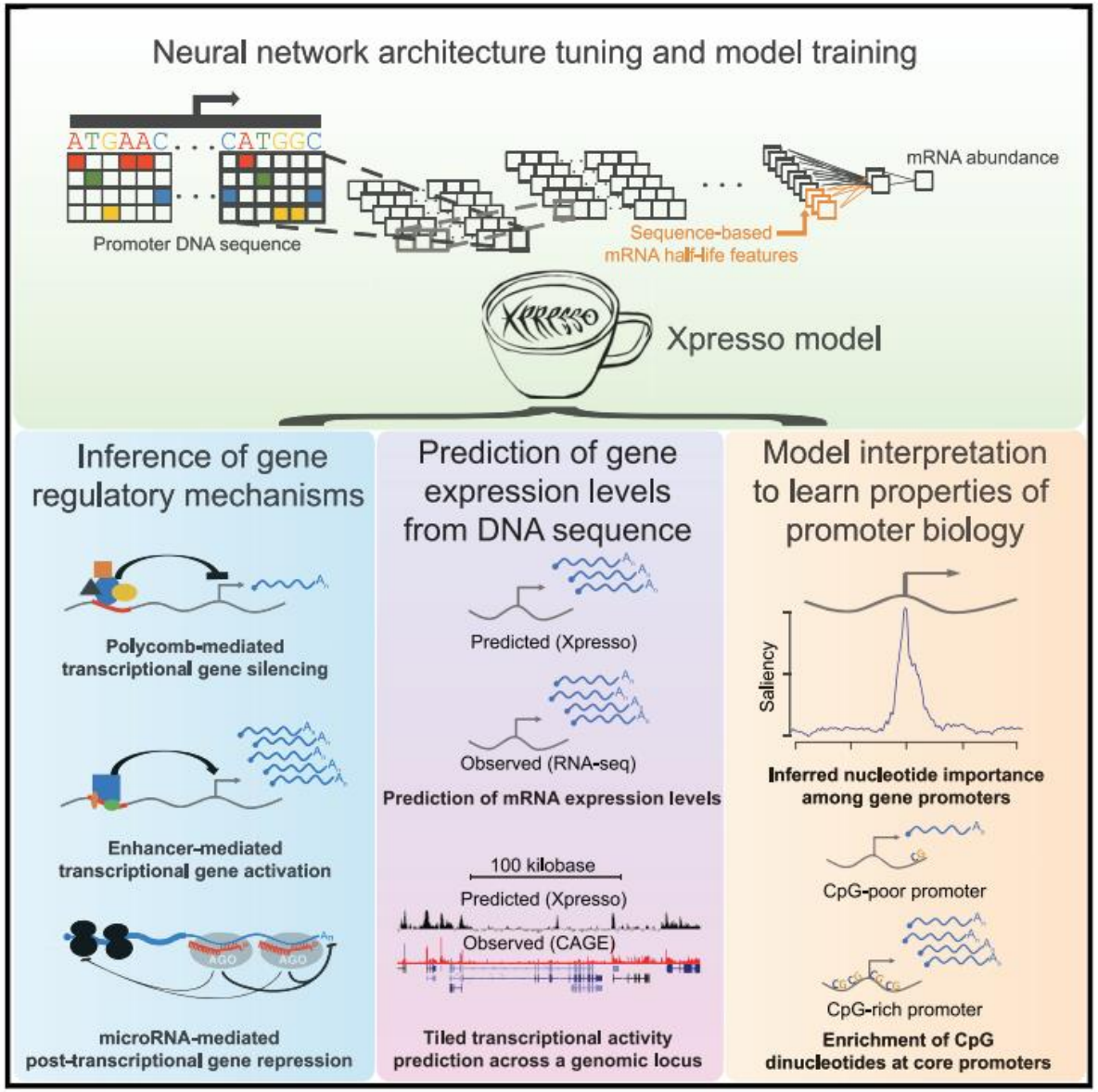
CNN based models

- Enformer (Nature Genetics, 2021)
- Enformer takes as input one-hot-encoded of length 196,608 bp and predicts 5,313 genomic tracks for the human genome and 1,643 tracks for the mouse genome, each of length 896 corresponding to 114,688 bp aggregated into 128-bp bins.
- Self-attention after convolution: tokenized by convolution



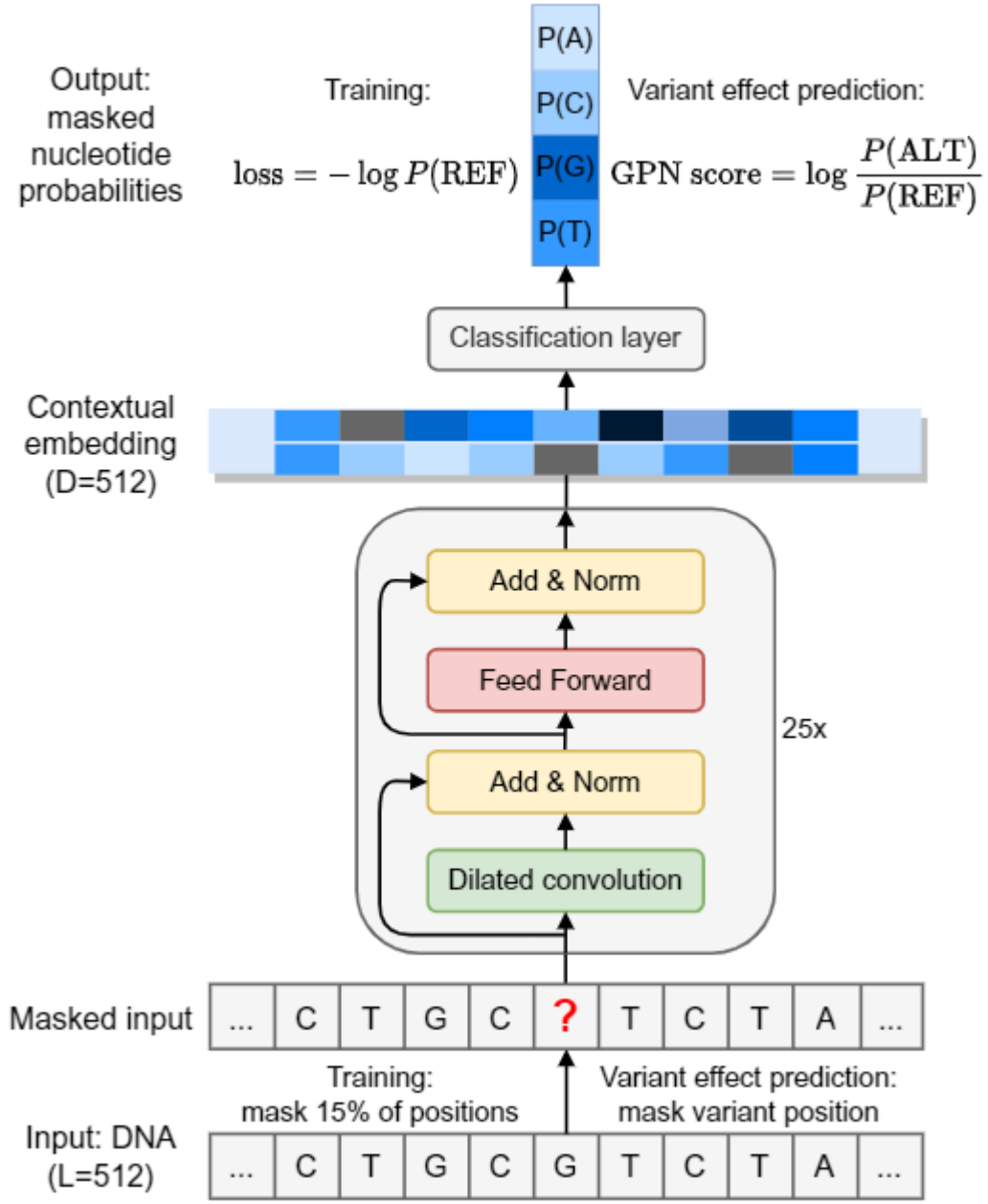
CNN based models

- Xpresso (Cell Reports, 2020)
- a deep convolutional neural network that jointly models promoter sequences and features associated with mRNA stability to predict steady-state mRNA levels.
- The ± 10 kilobase sequence centered at the TSS was extracted as the putative promoter region to consider.



CNN based models

- GPN (PNAS 2023)
- Pre-trained Network (GPN), a model designed to learn genome-wide variant effects through unsupervised pretraining on genomic DNA sequences.
- Pretrain model based on CNN, zero-shot inference

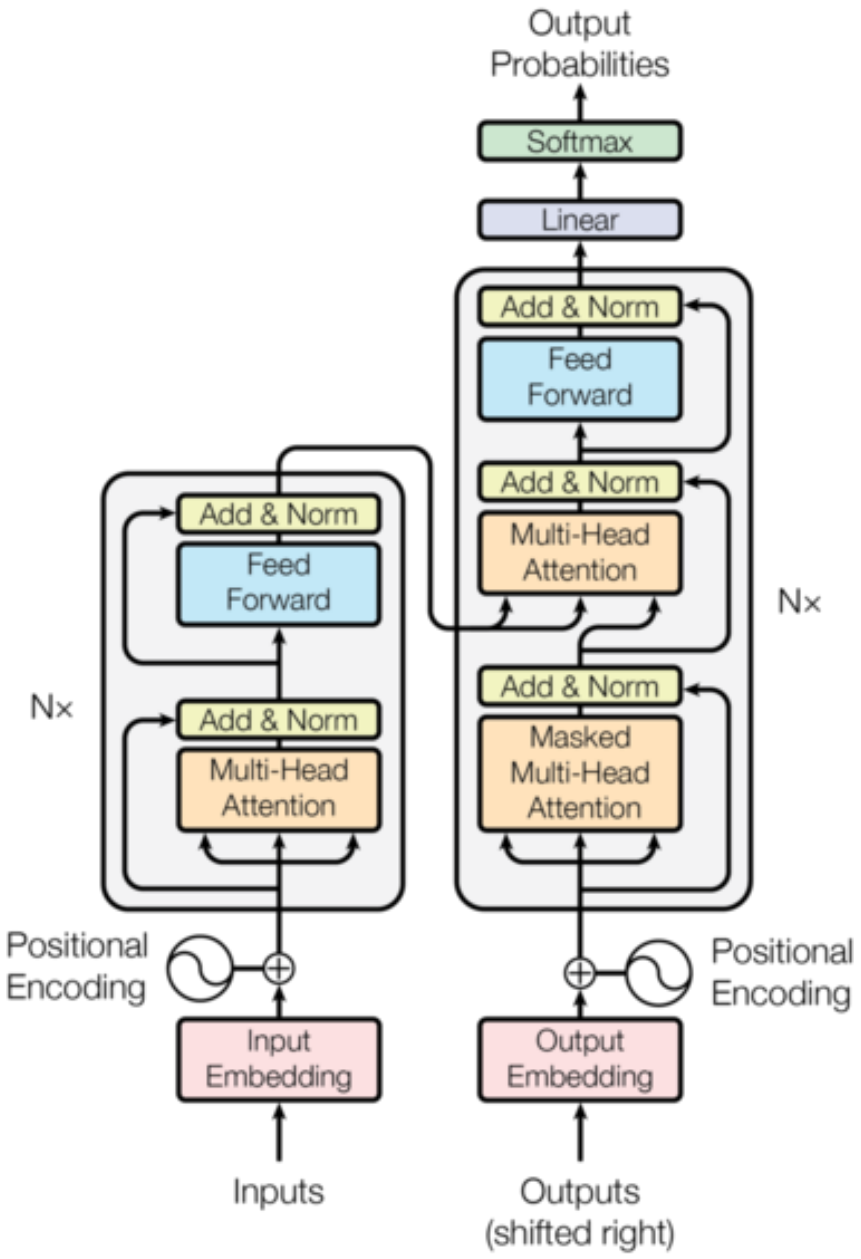


Transformer based models

- Token embedding:
  - Single nucleotide: e.g. one-hot
  - k-mer (overlap/non-overlap)
  - BPE
- Position embedding:
  - absolute position
  - ALiBi
- Model architecture and pretraining tasks
  - Transformer
    - Auto-encoding: BERT-like (Mask language modeling)
    - Auto-regressive: GPT-like (Generative)
  - others

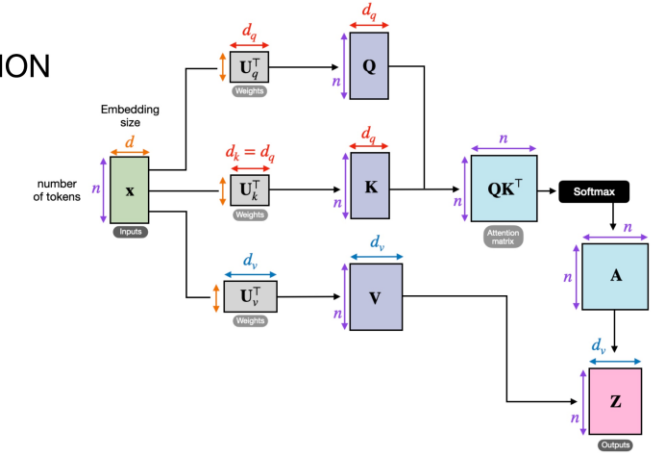
Transformer

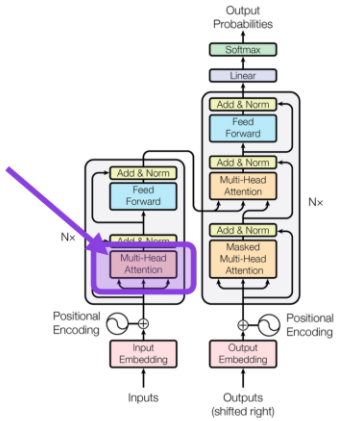
- Transformer
  - Encoder: multi-head self-attention
  - Decoder: multi-head masked self-attention, multi-head encoder-decoder cross-attention



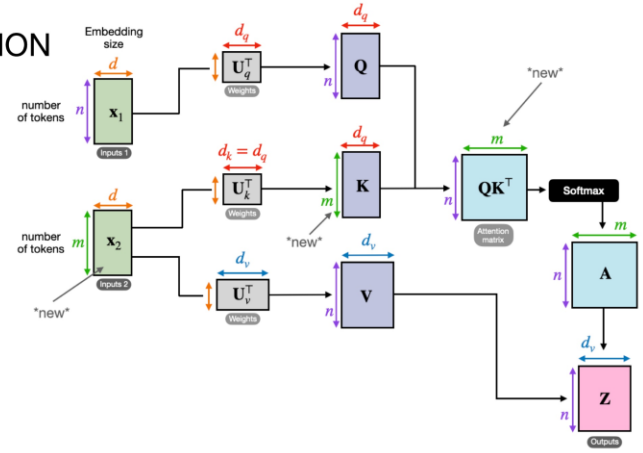
Attention

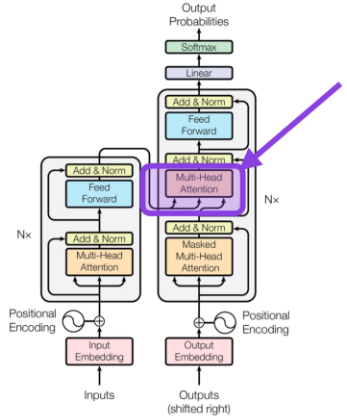
Self-Attention



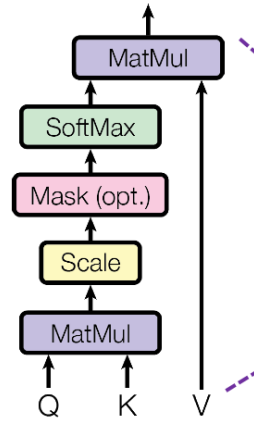


Cross-Attention

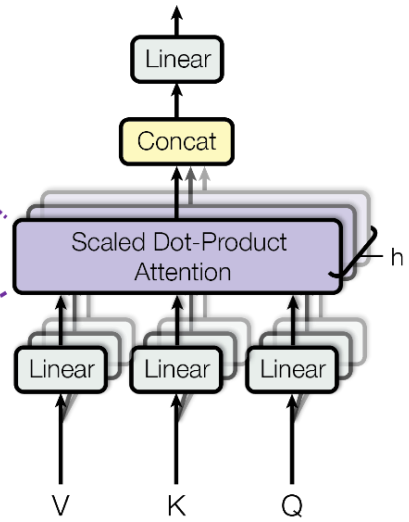




Scaled Dot-Product Attention



Multi-Head Attention



$$Attention(Q, K, V) = Softmax(\frac{QK^T}{d_k})V$$

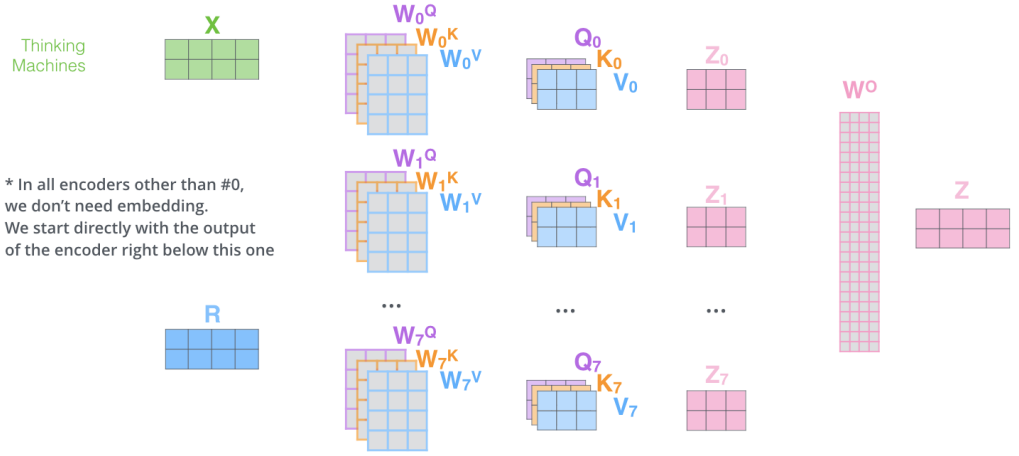
1) This is our input sentence*

2) We embed each word*

3) Split into 8 heads. We multiply X or R with weight matrices

4) Calculate attention using the resulting Q/K/V matrices

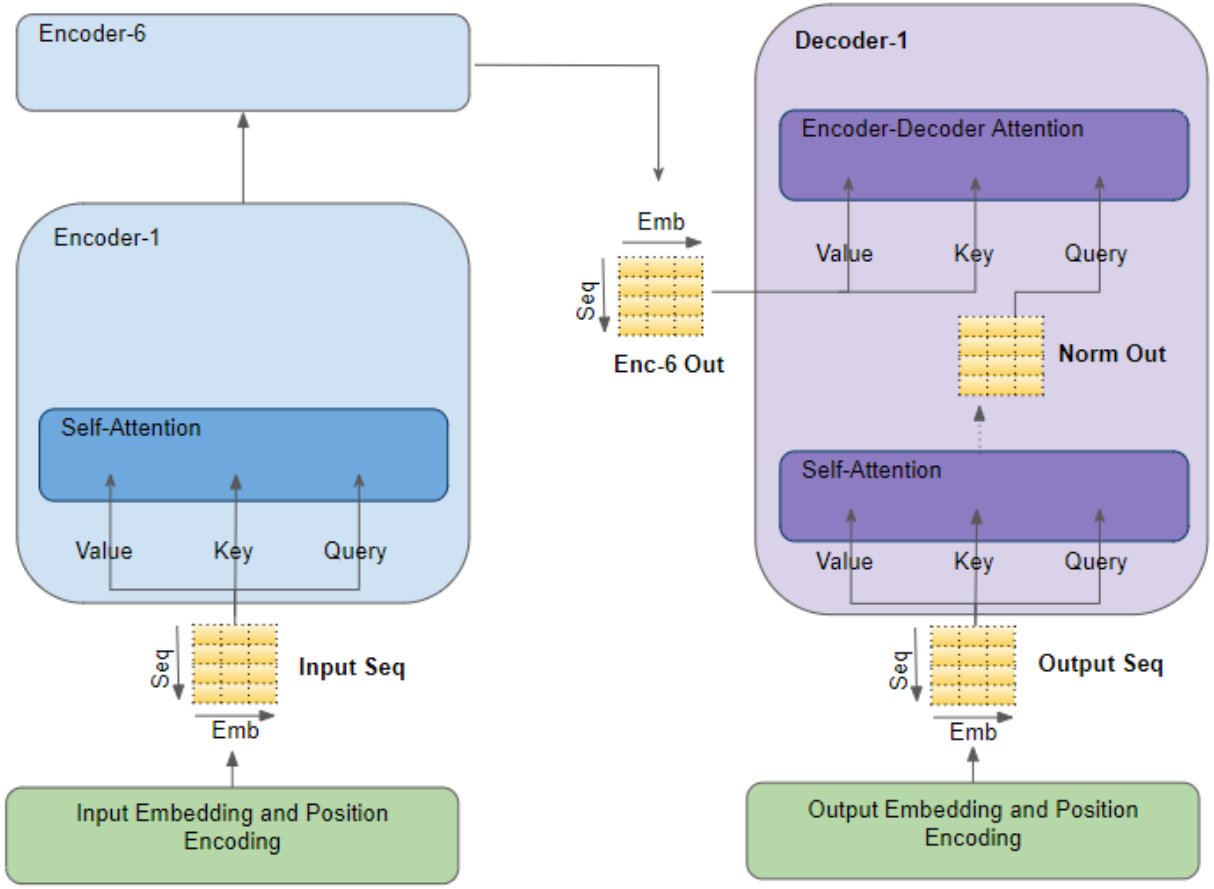
5) Concatenate the resulting Z matrices, then multiply with weight matrix W^O to produce the output of the layer

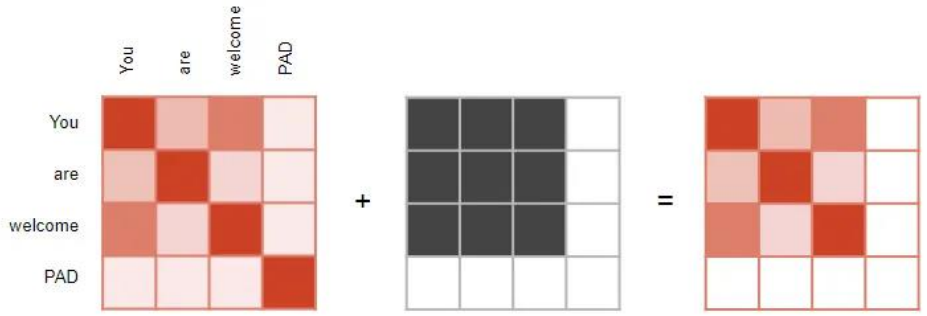


* In all encoders other than #0, we don't need embedding. We start directly with the output of the encoder right below this one

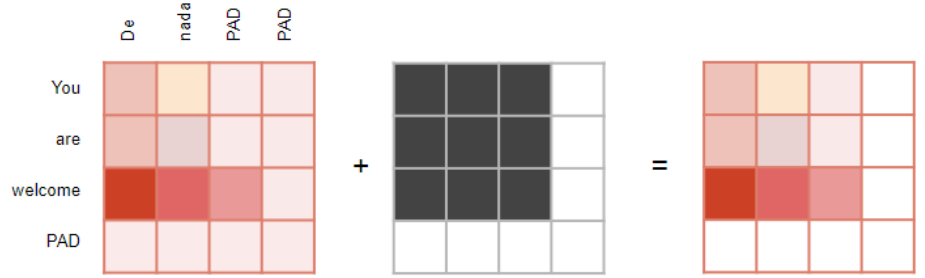
Attention

$$Softmax(QK^T + M)V$$

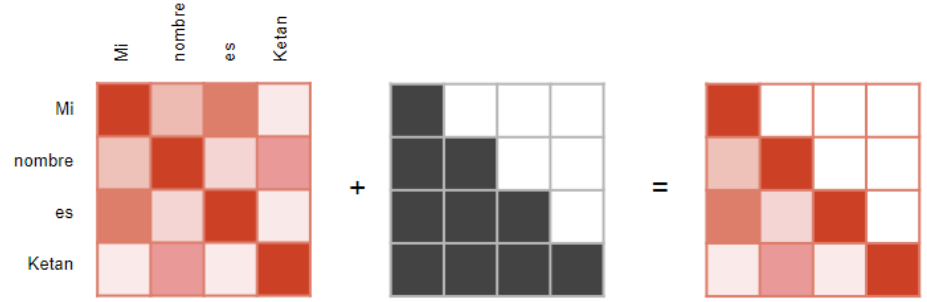




Encoder Self-Attention Scores



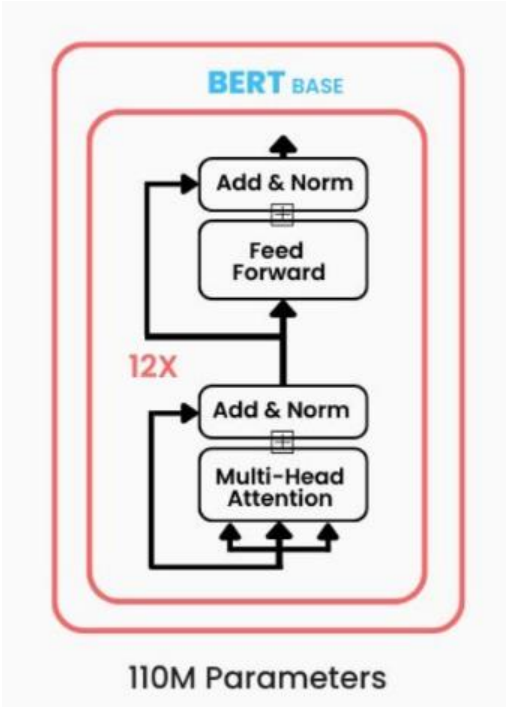
Encoder-Decoder Attention Scores

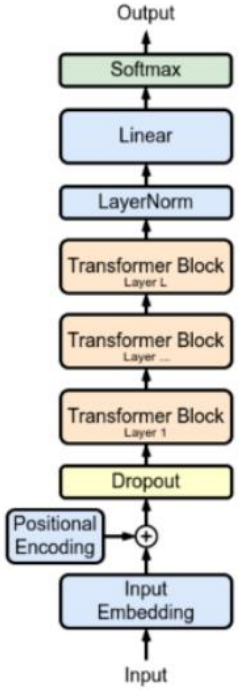


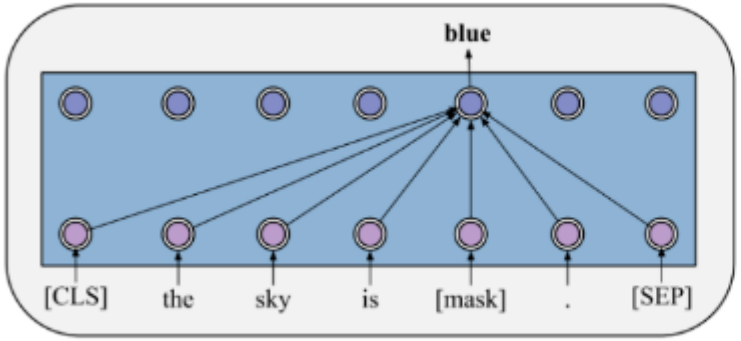
Decoder Self-Attention Scores

BERT, & GPT

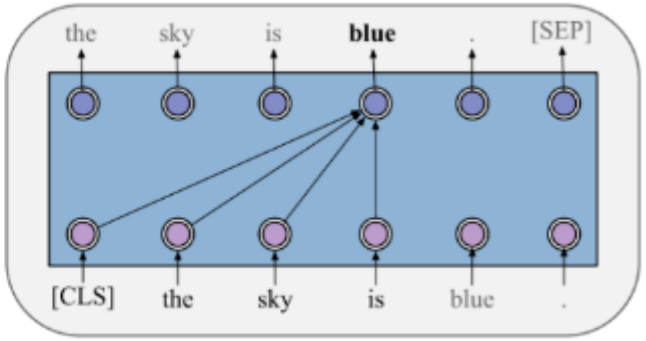
- Encoder-only
  - BERT: mask language model (predicts masked words based on the surrounding context)
- Decoder-only
  - GPT: causal language model (predicts the next word in a sequence)
- Encoder-decoder
  - BART





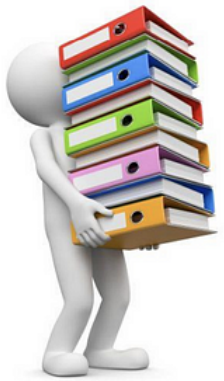


BERT



GPT

Pretrain-finetune



Self-supervised Learning

Pre-train

BERT

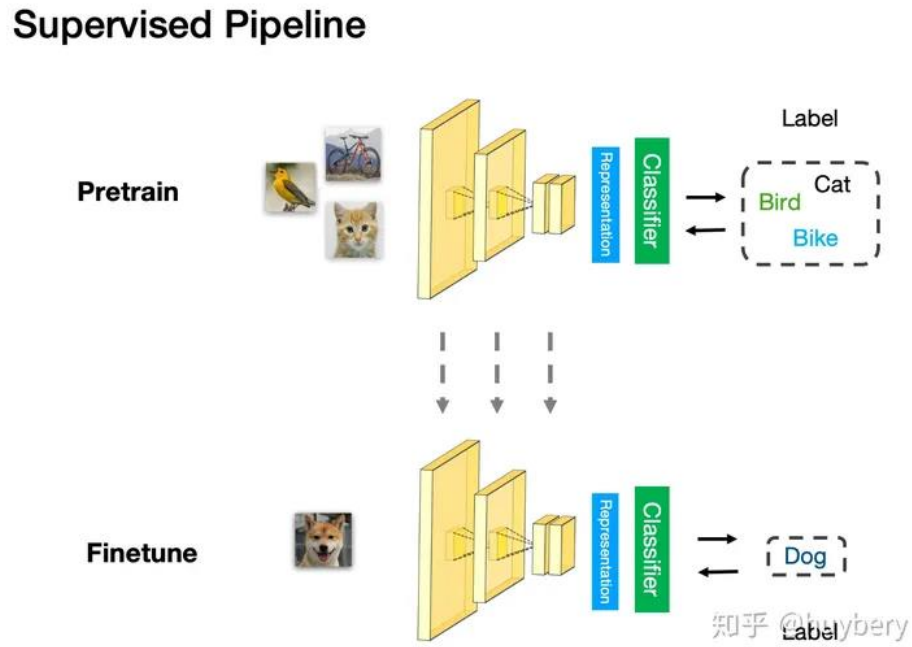
- Masked token prediction
- Next sentence prediction

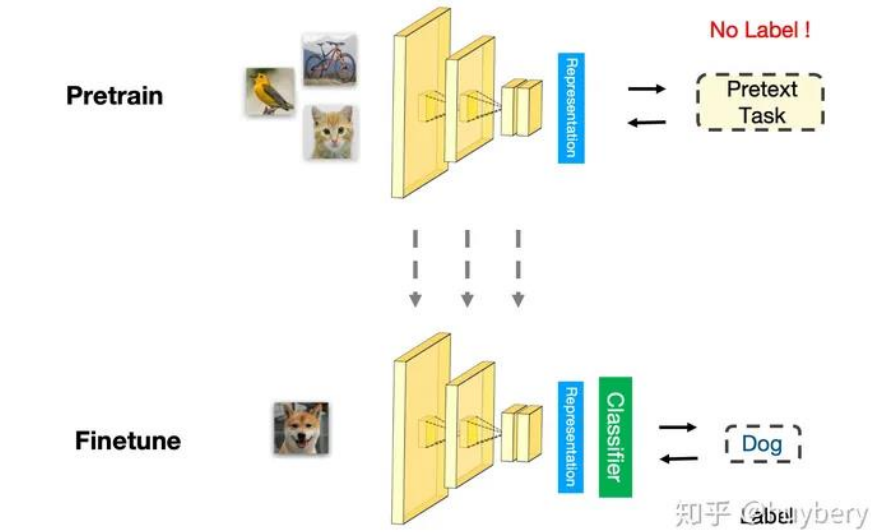
Fine-tune

- Model for Task 1
- Model for Task 2
- Model for Task 3

Downstream Tasks

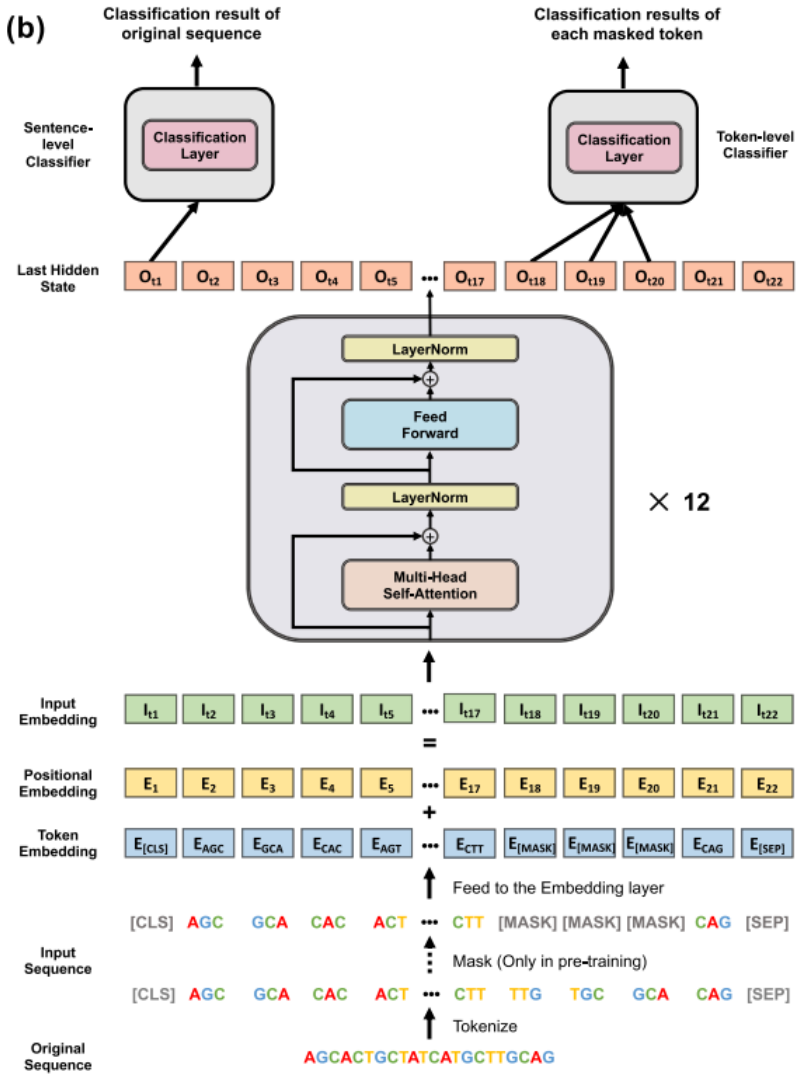
- The tasks we care
- We have a little bit labeled data.





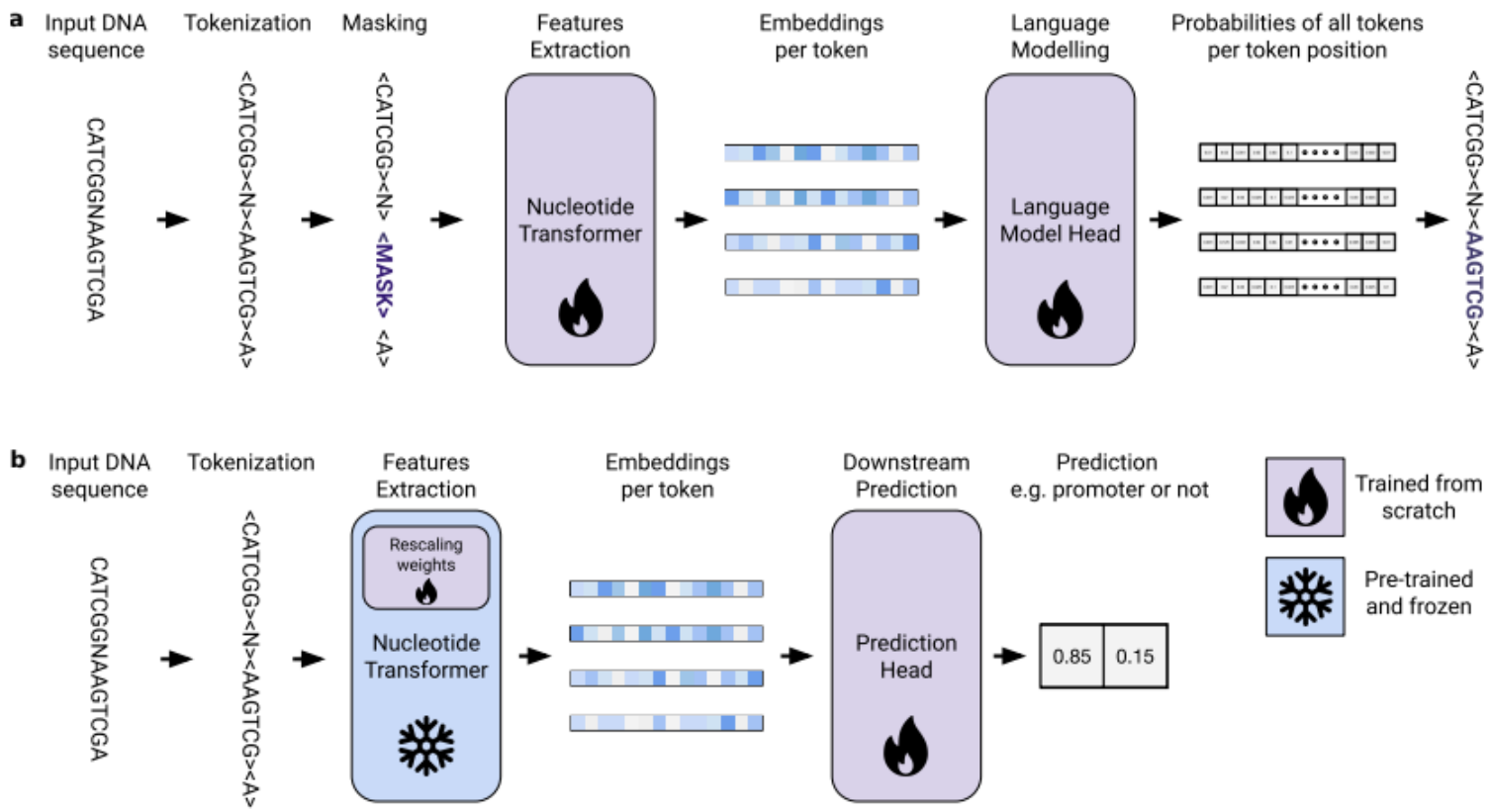
Transformer based models

- DNABERT (Bioinformatics, 2021)
- to capture global and transferrable understanding of genomic DNA sequences based on up and downstream nucleotide contexts.
- Overlapped k-mer, absolute position embedding, mask language model (MLM)
- Max input: 512 bp



Transformer based models

- Nucleotide Transformer (bioRxiv, 2023)
- pretraining (a) finetuning (b).
- non-overlapping 6mer, mask language model (MLM)
- Human genomics, 1000 genomics, multi-species
- Max input: 12 kb



Transformer based models

- DNABERT-2 (ICLR 2024)
- Token: Byte Pair Encoding (BPE)
- Position embedding: Attention with Linear Biases (ALiBi)
- Flash Attention
- Parameter-Efficient Tuning: Low-Rank Adaptation (LoRA)
- replace the Relu activation function with GEGLU
- Pretraining on genomics of multiple species
- Max length: 3 kb

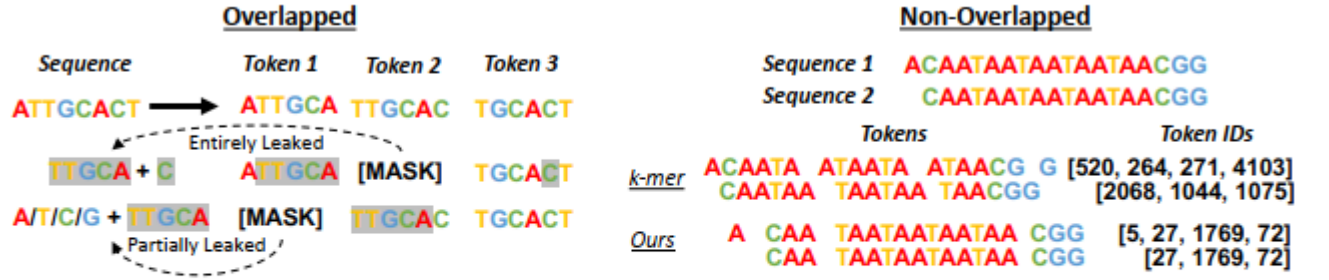


Figure 1: Illustration of the drawbacks of k-mer tokenization. In the overlapping setting, information about a masked token is leaked by its adjacent tokens, while in the non-overlapping setting, adding/deleting one nucleotide base leads to a dramatic change in the tokenized sequence.

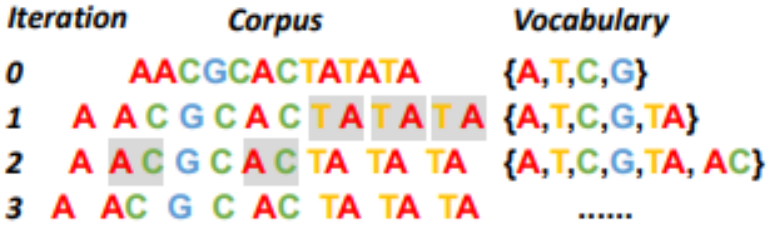
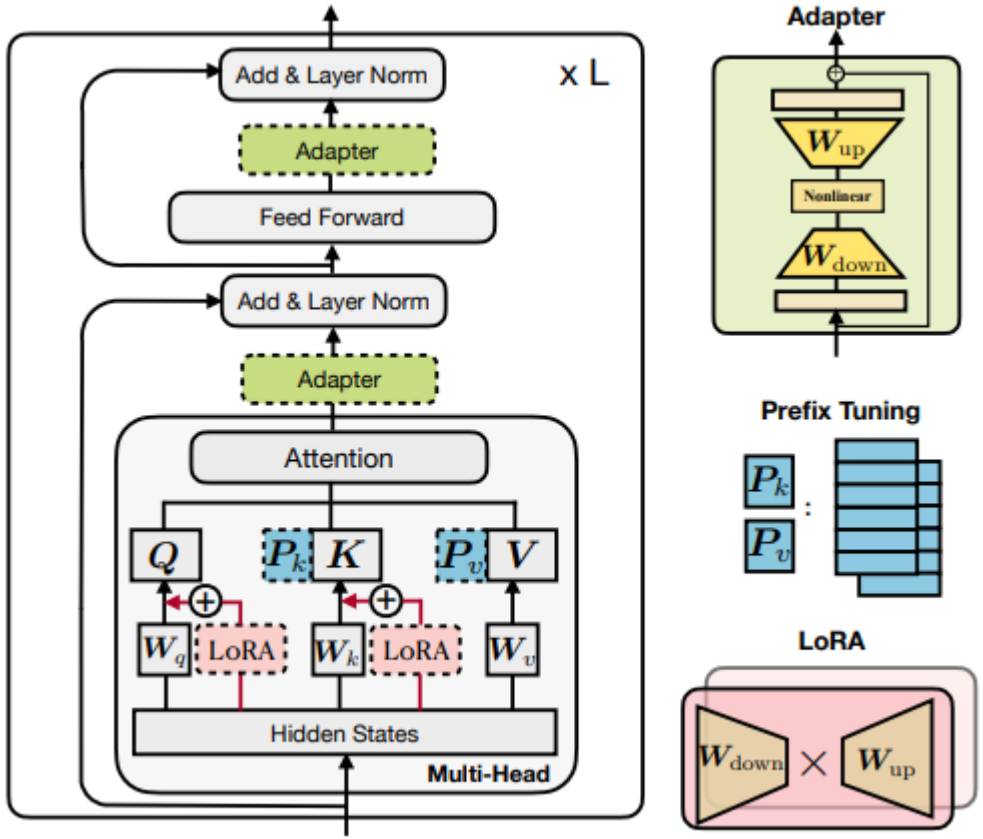


Figure 2: Illustration of the BPE vocabulary constructions.

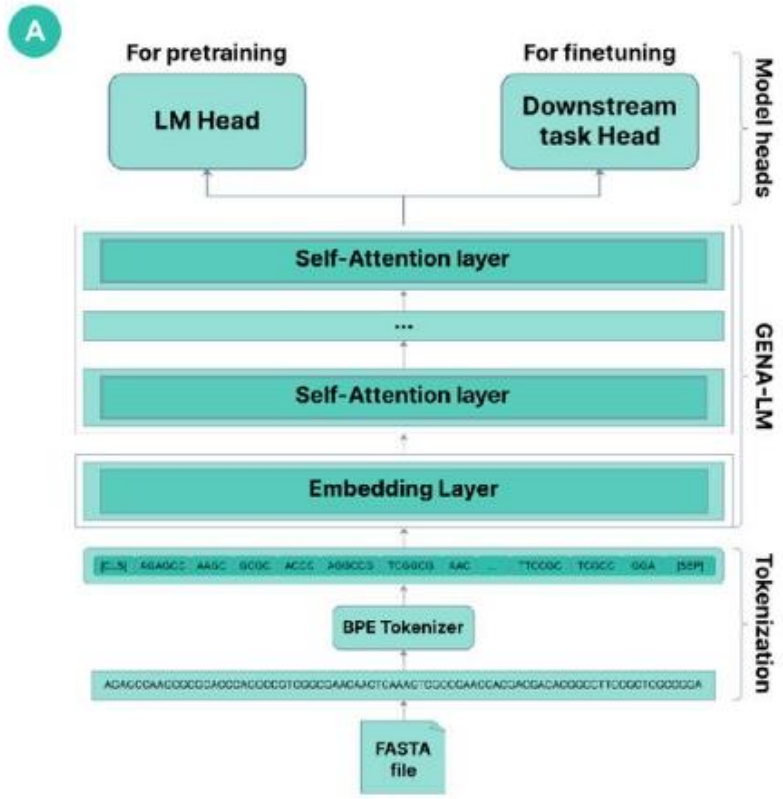
Parameter-Efficient Tuning (PEFT)

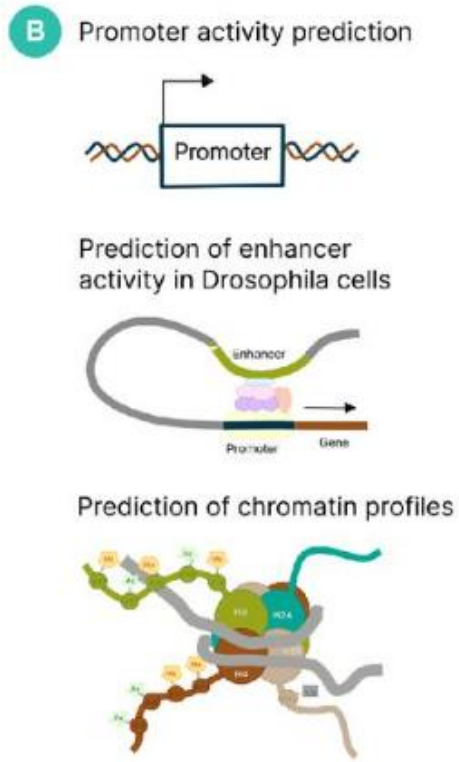
- Adapter tuning
  - Series adapter
  - Parallel adapter
- Prefix Tuning
  - Variants: Prompt Tuning, P-Tuning, etc.
- LoRA: Low-Rank Adaptation

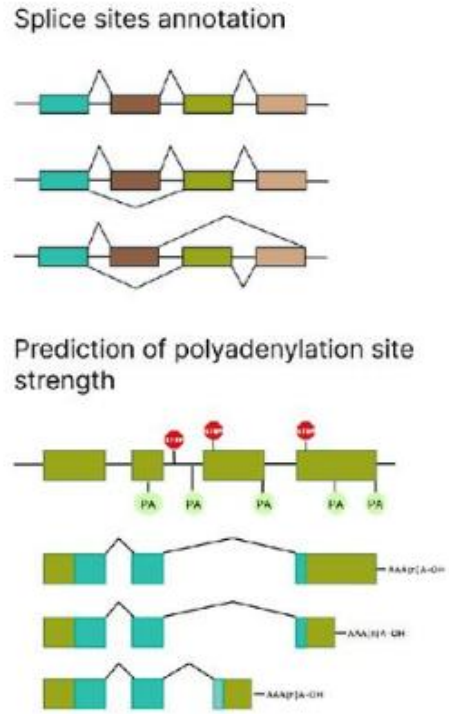


Transformer based models

- GENA-LM (bioRxiv, 2023)
- A series of models based on BERT and Bigbird
- BPE tokenization
- sparse attention mechanism
- Max input: approximately 4.5
- kb (512 tokens with full attention) and 36 kb (4096 tokens with sparse attention).

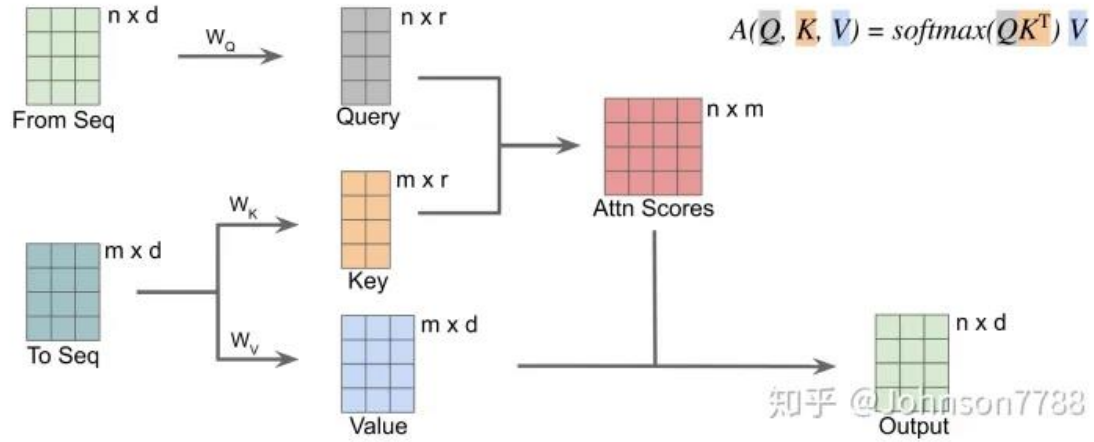


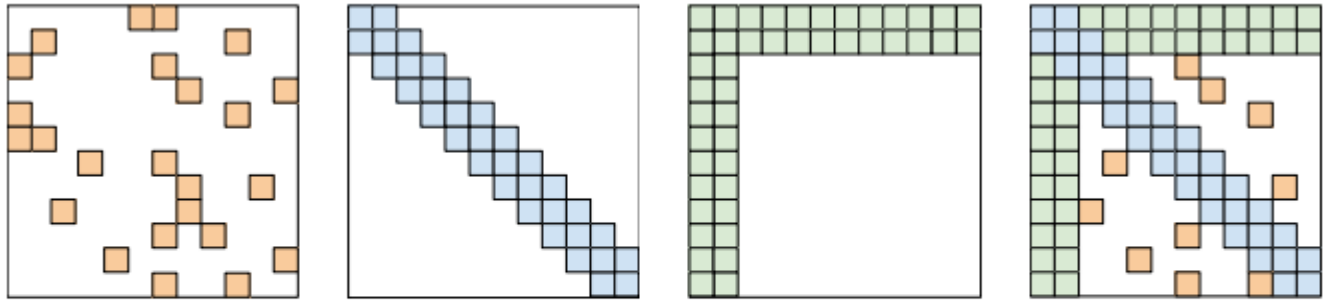




Transformer based models

- BigBird (NeurIPS, 2020): Sparse Attention Mechanism
- Long sequences





(a) Random attention (b) Window attention (c) Global Attention (d) BIGBIRD

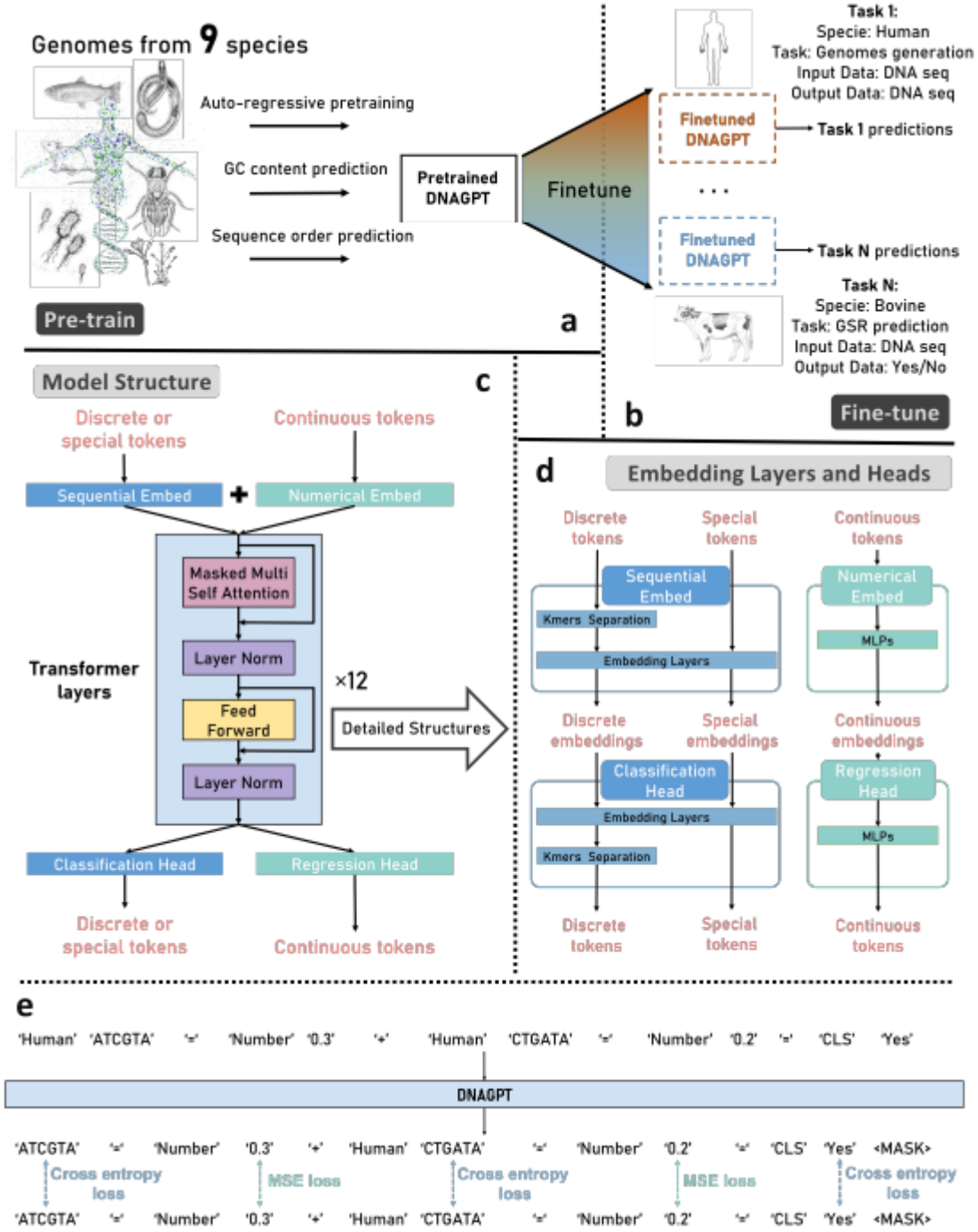
Figure 1: Building blocks of the attention mechanism used in BIGBIRD. White color indicates absence of attention. (a) random attention with $r = 2$, (b) sliding window attention with $w = 3$ (c) global attention with $g = 2$. (d) the combined BIGBIRD model.

Transformer based models

- DNAGPT (bioRxiv, 2023)
- DNA token: non-overlapped k-mers

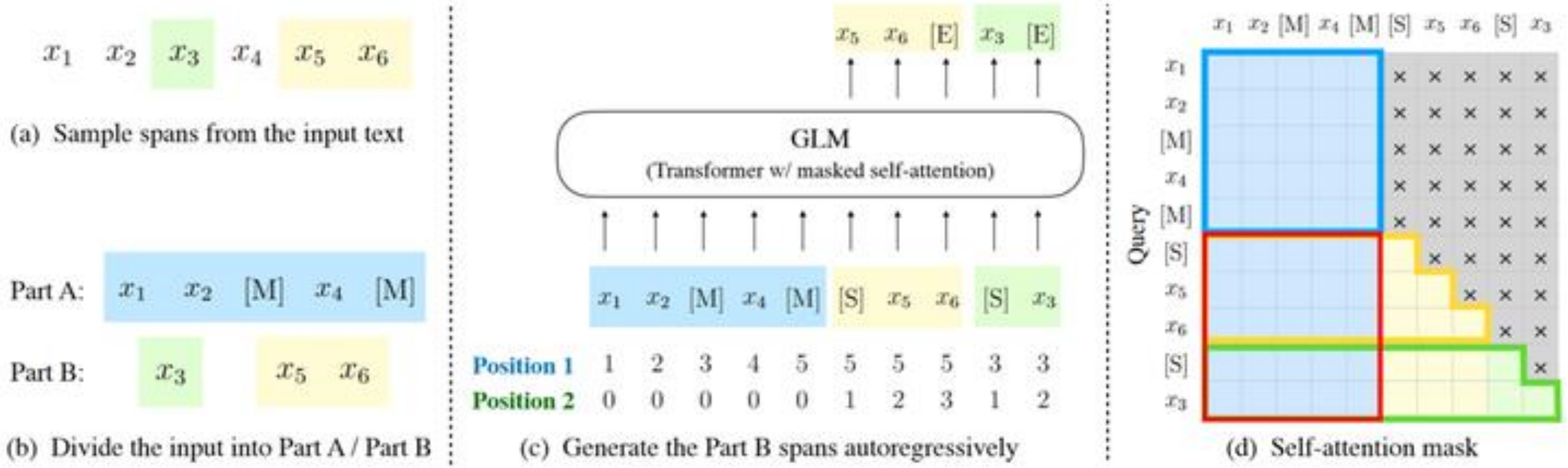
a. Tokens used in DNAGPT

DNA tokens Examples: 'ATCTAG'	Description: DNA sequence tokens	Instruction tokens Examples: 'Human', 'Bovine'	Description: Used to indicate the type of data generated afterwards	Classification tokens Examples: 'X', 'Y'	Description: Used to indicate the result of a binary classification problem
Continuous tokens Examples: '0.85', '-0.02'	Description: Number tokens	Connection tokens Examples: '+', '='	Description: Used to indicate the relationship between the two sequences before and after	Reserved tokens Examples: '0', '2', '*', 'K'	Description: Used to build up downstream tasks



Transformer based models

- GLM (generalized language model): Autoregressive Blank Infifilling (ACL, 2022)



Mixture-of-Experts (MoE)

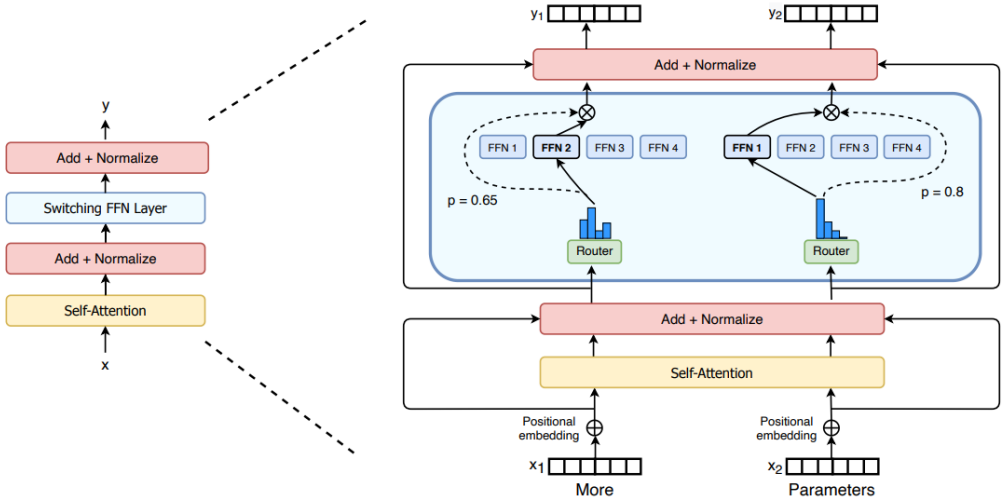
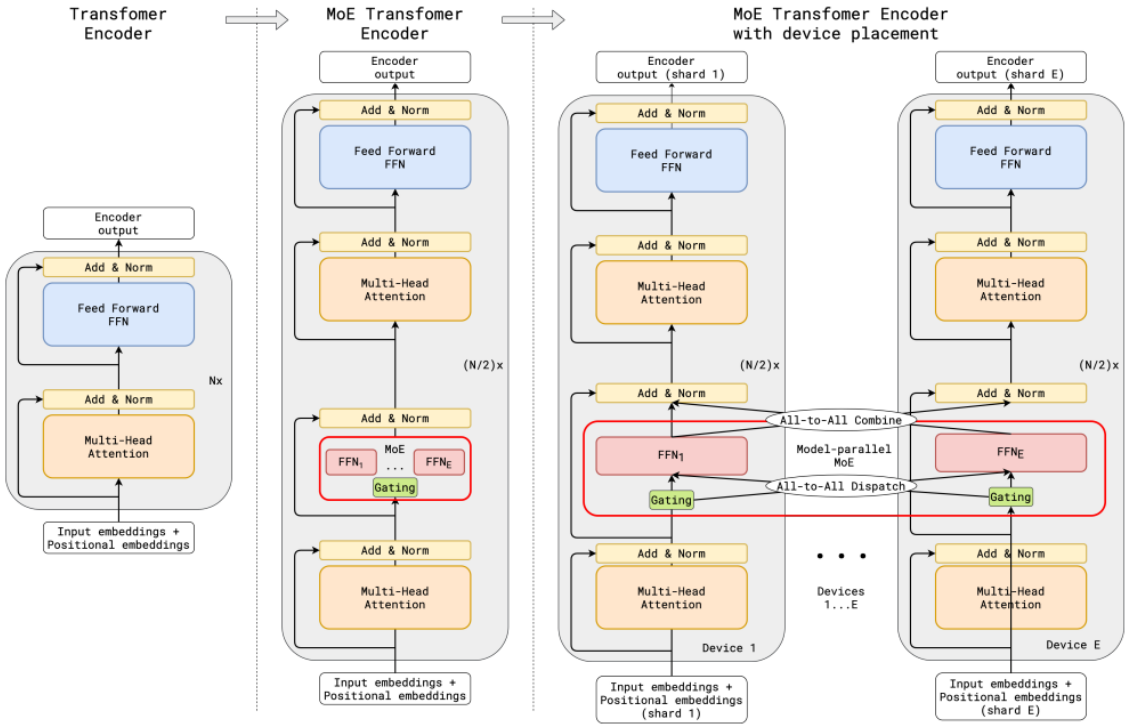


Figure 2: Illustration of a Switch Transformer encoder block. We replace the dense feed forward network (FFN) layer present in the Transformer with a sparse Switch FFN layer (light blue). The layer operates independently on the tokens in the sequence. We diagram two tokens (x_1 = "More" and x_2 = "Parameters" below) being routed (solid lines) across four FFN experts, where the router independently routes each token. The switch FFN layer returns the output of the selected FFN multiplied by the router gate value (dotted-line).

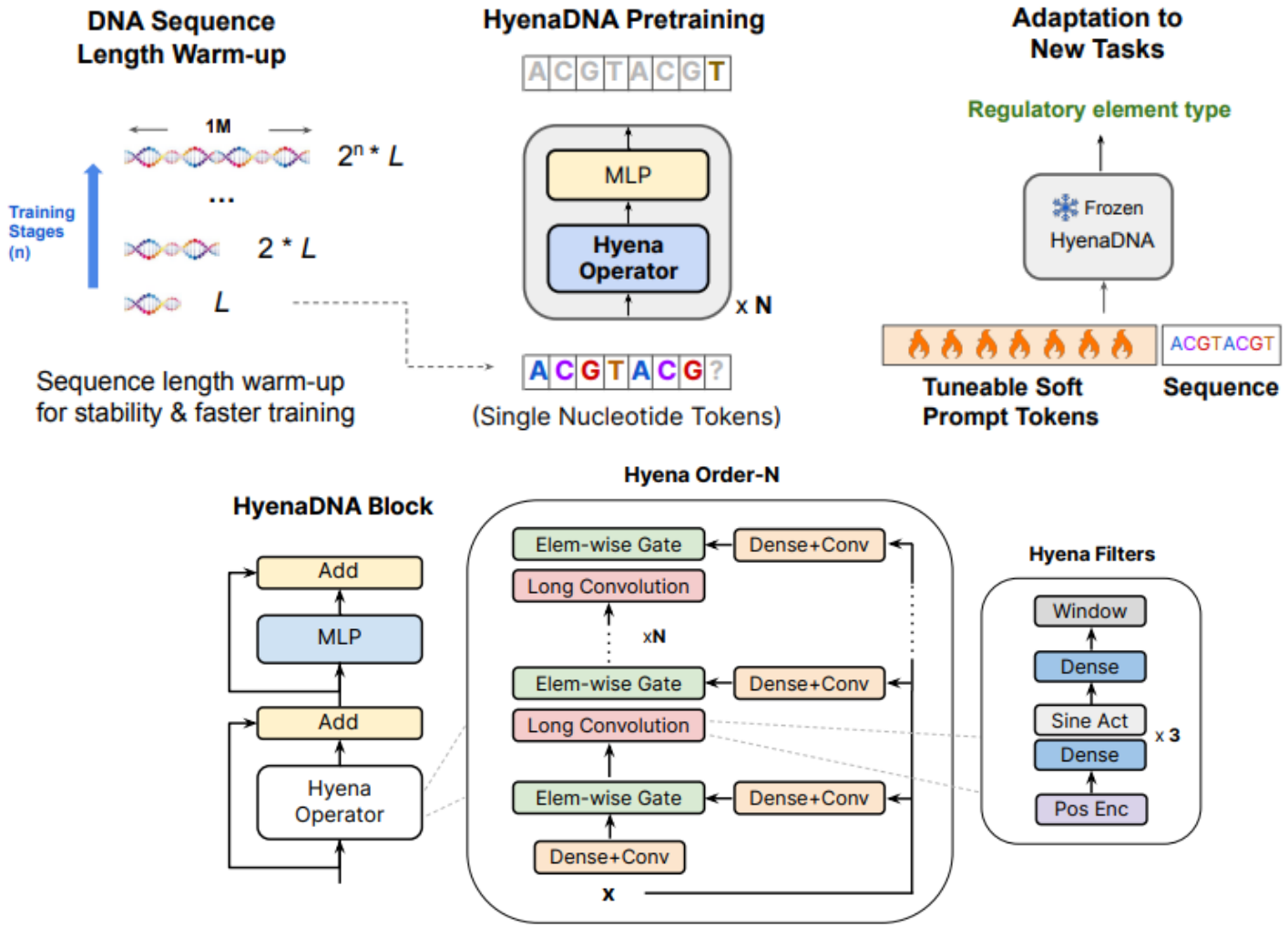
Switch Transformers



GShard

LLM beyond transformer

- HyenaDNA (NeurIPS, 2023)
- Long sequences: max 1 million bp
- Hyena uses a parameter-efficient global convolutional filter along with a data-controlled gating mechanism, which enables a context-specific operation over every token.



LLM beyond transformer

- Mamba (arXiv, 2023)
- Selective SSM: based on S4 (Structured State Spaces for Sequence Modeling)

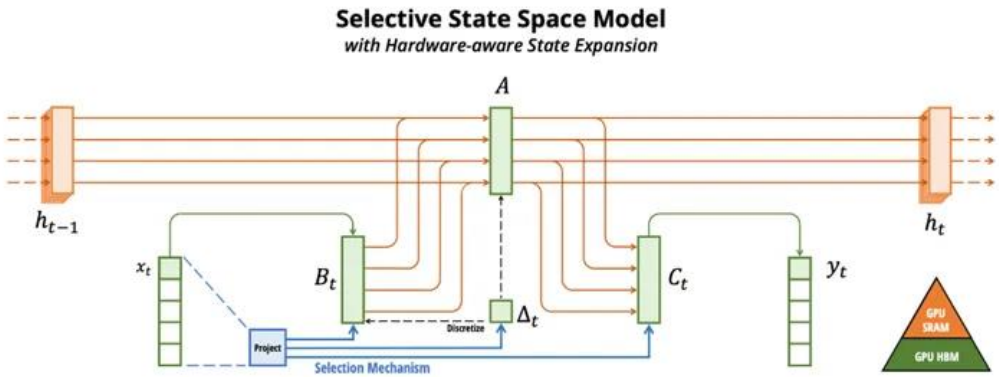


Figure 1: (**Overview**.) Structured SSMs independently map each channel (e.g. $D = 5$) of an input x to output y through a higher dimensional latent state h (e.g. $N = 4$). Prior SSMs avoid materializing this large effective state (DN, times batch size B and sequence length L) through clever alternate computation paths requiring time-invariance: the (Δ, A, B, C) parameters are constant across time. Our selection mechanism adds back input-dependent dynamics, which also requires a careful hardware-aware algorithm to only materialize the expanded states in more efficient levels of the GPU memory hierarchy.

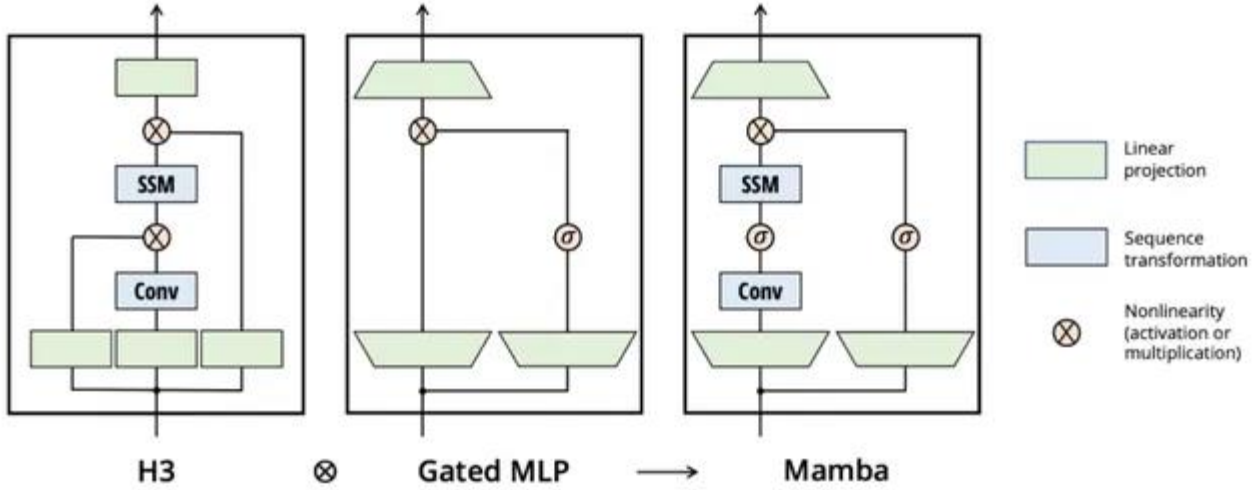


Figure 3: (**Architecture**.) Our simplified block design combines the H3 block, which is the basis of most SSM architectures, with the ubiquitous MLP block of modern neural networks. Instead of interleaving these two blocks, we simply repeat the Mamba block homogenously. Compared to the H3 block, Mamba replaces the first multiplicative gate with an activation function. Compared to the MLP block, Mamba adds an SSM to the main branch. For σ we use the SiLU / Swish activation (Hendrycks and Gimpel 2016; Ramachandran, Zoph, and Quoc V Le 2017).

Thank you!