Genome language models for Enhancer-Promoter distal regulation

Evo2 (HiC)

Charles VIELZEUF

Statistical Genetics Team
RIKEN AIP

charles-vzf.eu

June 9, 2026

Outline

- **Brief introduction on enhancers, promoters, and available datasets**
- **Genome language models overview**
- **Presentation of Evo2 model and framework; exon boundary test and embedding probes**
- **Hi-C context and Evo2HiC model; CAGE (and joint DNase/CAGE) track prediction using Evo2HiC**
- **EPInformer model for distal regulation prediction**
- **Tested pipelines for distal regulation prediction**
- **Conclusions**

Introduction on why enhancer–promoter interactions matter

Most GWAS hits and disease variants are **noncoding** and likely act by perturbing **cis-regulatory logic**: **which enhancer (increases gene transcription) controls which promoter (where RNA polymerase binds to start transcription)**, in **which cell state**, and at **what distance in 3D**.

CAGE (Cap Analysis of Gene Expression) solves part of the problem by making it relatively easy to identify promoters. It was pioneered at RIKEN and scaled in the FANTOM5 consortium to map **TSS** activity (and enhancer RNA) genome-wide. This gives a practical **track-level target** (**bigWig**: *b*inary *i*ndexed *wiggle* file storing continuous genomic signal along the reference assembly) for models that aim to predict **transcription initiation**.

Why it is hard. Distal regulation is **combinatorial** (many TFBS), **cell-type specific**, and depends on **chromatin contacts**. Even with catalogs like ENCODE SCREEN/cCRE, linking enhancers to genes requires integrating **activity + 3D proximity** (e.g. ABC) and large reference maps (e.g. ENCODE-rE2G: **>13 million enhancer-gene regulatory interactions across 352 cell types and tissues, by integrating predictive models**).

Long-range links are weakest. For distances $\gtrsim$**500 kb** (*kilo-base pairs*), experimental signals often **fade with genomic distance** (distance-decay in contact maps; sparse functional validation), making true enhancer–promoter links harder to detect. This motivates learning **more robust and targeted representations** that integrate **multiple modalities** (sequence, epigenomics, 3D).

Enhancer/promoter: short context

Promoters (TSS-proximal): recruit Pol II (RNA polymerase II) and define transcription start; sequence grammar is comparatively **compact and stereotyped** (core motifs: **TATA**, **Inr**, **DPE**, **BRE**, **MTE**, etc.).

Enhancers (often distal): tune expression level, timing, and **cell-type** usage; many **TFBS** combinations, more **context-dependent** than core promoters.

Key fact: E–P wiring is **cell-type specific**; **which enhancer contacts which promoter** varies more than promoter activity alone.

Object	**Typical value**	**Comment**
Human gene locus	20–30 kb med.; 67 kb mean	Protein-coding span (RefSeq)
Core promoter / PLS	∼200 bp	TSS core; PLS = promoter-like cCRE
Enhancer element	∼800 bp mean	Standard enhancers; ≥3 kb = stretch
E–P distance	peak 20–50 kb; most <200 kb	Can reach Mb in GWAS loci
Enh./TSS; prom./enh.	∼4–5; ∼2	Many enh. → one prom. (common); one enh. → few prom.
Annotation & activity (datasets)		
Reference maps	> 2M+ cCREs	SCREEN/ cCRE; FANTOM5 CAGE; ENCODE-rE2G
Typical signal tracks (bigWig / peaks)		
Accessibility	DNase-seq; ATAC-seq	open chromatin / TF footprint
Histone ChIP-seq	H3K27ac, H3K4me1, H3K4me3	active enh. / poised enh. / active prom.
Expression	CAGE, RNA-seq (mRNA)	TSS initiation; steady-state abundance
Interactions (causal / 3D)		
Assays	screens / maps	CRISPRi; Hi-C / micro-C; ABC (activity + contact); MPRAs

Datasets useful for distal regulation (enhancer/promoter)

Dataset	Approx. size	Source / content	Typical uses
ENCODE cCRE / SCREEN	>2M cCREs	SCREEN / ENCODE; K562/GM12878/HepG2/H1-hESC benchmark panel	cCRE priors; DNase/H3K27ac/Hi-C tracks for E–G models
FANTOM5 CAGE	atlas-scale across many biosamples	cap-analysis gene-expression (TSS activity) + enhancer RNA signals; FANTOM5 portal	promoter usage labels, enhancer activity targets, transcript-initiation supervision
GTEx v8 eQTL	49 tissues (large variant-gene catalog)	tissue-level eQTL associations from GTEx	distal variant-to-gene validation, disease interpretation, external benchmarking
ABC maps	genome-wide per biosample	Activity-by-Contact enhancer-gene predictions (accessibility + H3K27ac + contact), pipeline: ABC	strong practical baseline for enhancer-promoter linking and model supervision
ENCODE-rE2G encyclopedia	>13M enhancer-gene links, 352 cell types/tissues	integrated predictive map + benchmark framework, preprint bioRxiv 2023	large reference for distal regulation, noncoding variant-to-gene interpretation, model comparison
CRISPR E-G	∼10k tested element-gene pairs (compiled)	aggregated perturbation sets (Nasser/Gasperini/Schraivogel-style collections used in ENCODE/Engreitz benchmarking)	causal evaluation set for enhancer-gene prediction quality (AUPRC/precision-recall)
OpenGenome (v1)	∼300B tokens; 2.7M genomes	prokaryotic + phage whole-genome corpus (LongSafari/open-genome); Arc/Together release for Evo1 pretraining	DNA LM pretraining at molecular–genome scale; prokaryotic regulatory sequence grammar
OpenGenome2	∼8.8–9T bp; >128k genomes	all domains of life (bacteria, archaea, eukaryotes, viruses); Arc HF dataset (FASTA + JSONL w/ phylogenetic tags) for Evo2	large-scale genomic foundation-model training; long-context autoregressive DNA modeling
RefSeq	curated gene/transcript models	NCBI Reference Sequence: non-redundant annotated assemblies, genes, mRNAs, proteins (e.g. GRCh38/hg38)	gene locus boundaries, TSS/transcript annotation, promoter windows, variant interpretation

Formats: FASTQ, BAM, bigWig, BED/narrowPeak/broadPeak, metadata.

Genome language models

Core idea: treat DNA (or multi-omics token streams) like a sequence modeling problem and learn $p(x_t \mid x_{<t})$ at scale.

- **Representation learning**: hidden states encode motifs, regulatory syntax, and long-range dependencies (via convolutions/attention).
- **Transfer setting**: one pretrained foundational model, many downstream tasks (variant effect, promoter/ enhancer prediction, splice/exon boundary tasks).
- **Main technical challenge**: sequence length. Genomes need context windows far beyond classic NLP lengths: around 500k for promoter/enhancer long distances
- **NLP context (order of magnitude):** classical stacks often $T \sim 10^2$–10^3 tokens (BERT-era ∼512–2k); frontier LMs today are often 10^4–10^5+, with million-token-class advertised contexts for some SOTA models (effective use still hardware- and implementation-dependent).

$$\mathcal{L}_{\mathrm{AR}}(\theta) = -\sum_{t=1}^{T} \log p_\theta(x_t \mid x_{<t}), \qquad h_t = f_\theta(x_{\leq t})$$

h_t: contextual hidden state at position t, i.e., a compressed representation of sequence information up to t used to predict the next token/base.

downstream head: $\hat{y} = g_\phi(h_{1:T})$ (classification or regression)

Comparative table: selected DNA / genome sequence models

	DNABERT / DNABERT-2 2021 / 2023	**Nucleotide Transformer** 2023	**HyenaDNA** 2023	**Evo2** 2025	**GENERator** 2025 arXiv:2502.07272	**Enformer** 2021	**Evo2HiC** 2025 [github]
Size / scale	small (∼100M-class BERT-scale)	large (multi-hundred-M → multi-B checkpoints)	large (long-model family)	**base ∼7B; large ∼40B**	**1.2B / 3B** (long-ctx gen.)	medium (∼216M-class)	compact encoder (distilled from Evo2 7B)
Use cases (examples)	promoter/enhancer classification, motifs	seq. classification, variant-effect benchmarks	long DNA LM; **HyenaDNA** instantiates the **Hyena** hierarchy (long implicit conv. / gated filters; often gains with T)	**generation, embeddings, heads (exon/splice demos)**	gen. DNA; **enh./prom.** + distal links (long 1D); cis-reg. design; long 6-mer ctx. *vs.* char-LM (e.g. Evo2)	regulatory activity / gene-expression prediction from sequence	Hi-C maps, epigenomes, Hi-C resolution enhancement
Training dataset (reported)	human reference-genome MLM corpus (DNABERT) + broader multi-species benchmark corpus (DNABERT-2)	large multi-species genomic corpus (Nucleotide Transformer releases)	long genomic sequence chunks from reference genomes (human-focused + multi-species variants)	OpenGenome2 (∼8.84T bp reported; Arc Institute)	∼**386B nt** euk. DNA (reported)	human/mouse regulatory assays (CAGE, DNase/ATAC, ChIP; Enformer target collections)	Evo2-distilled embeddings + paired sequence/Hi-C supervision (project-specific multimodal sets) (`Evo2HiC/train/pretra`
Backbone	Transformer encoder	Transformer encoder	**Hyena**-family stack (Poli et al. '23): subquadratic long conv. ($\ll T^2$ vs. dense attention)	**StripedHyena hybrid**	causal **Transformer** dec.; **6-mer** tok. (len $\equiv 0 \pmod 6$ for gen.)	conv tower + Transformer encoder	lightweight multimodal encoder + Hi-C-guided training
Context strategy	bidirectional encoder; short fixed T	fixed attention span; **chunking** = windows + overlap when T exceeds model or GPU	Hyena mixing: no full $T \times T$ map; implicit kernel $K_\theta(\tau)$ over lags	**hybrid**: global attention blocks + Hyena long conv. layers; long T (checkpoint-dependent)	**causal AR**; ∼98k **bp** ≡ "98k ctx" → ∼16k tok. (6-mer); ≠ 98k×6	fixed-window bidirectional encoding for distal regulatory interactions	multimodal **2D Hi-C patch + 1D DNA** (one-hot/mappability or Evo2 memmaps @ 2 kb); tiled along chromosome
Context length	DNABERT ∼512 **bp (base pairs)**; **DNABERT-2: MSL 2048** tokens (paper default; ckpt-dep.)	∼6k–30k+ bp (checkpoint-dependent)	up to ∼1M bp (family-dependent)	**100k–1M+ class (checkpoint-dependent)**	∼**98k bp** genome; ∼**16k tok.** (6-mer)	∼196,608 bp input window	Hi-C patch **200–320 kb** (@ 2 kb; N=100/160); DNA: **60 kb** Evo2-7B memmap ctx. (2 kb bins); contact span $\lesssim$**2 Mb**

Comparative table: distal regulation + long-context DNA LMs

	Basenji 2018	**Borzoi** 2024	**ENCODE-rE2G** 2023	**EPInformer** 2024	**Caduceus (Schiff et al.) 2024**	**Mamba (Gu & Dao) 2023**
Size / scale	medium (∼50–60M-class; Basenji2-style)	large regulatory foundational model (multi-assay releases)	atlas pipeline (>13M predicted **enhancer–gene (E–G)** links)	medium (cell-line training variants)	long-context DNA foundational-model family (public checkpoints; HF)	selective **state space model (SSM)** *operator* (*Mamba*; in DNA LMs / benchmarks)
Use cases (examples)	multi-assay track prediction from sequence alone (CAGE-like profiles); joint modeling across loci	RNA-seq tracks from seq.; variant-effect (Calico successor line)	E–G linking; variant-to-gene	enhancer/promoter expression; E–G prioritization	long-range DNA representations; strong on variant tasks; **PlantCaduceus** FT (Zhai et al. 2024, 65 angiosperms)	subquadratic LM primitive; multitask DNA suites vs. Transformers; block inside Caduceus (Bi-Mamba)
Training dataset (reported)	ENCODE / Roadmap-style human (mouse) assay maps (CAGE, DNase, ChIP, . . . ; release-dependent)	ENCODE+GTEx RNA + Enformer-style tracks	CRISPR eQTL/GWAS + chromatin/contact features (many biosamples)	K562/GM12878: DNase, H3K27ac, CAGE, RNA-seq, Hi-C, ABC	long human genome pretraining (paper); plant data for PlantCaduceus FT	N/A (no canonical corpus); DNA papers use ref. genomes / tasks per study
Backbone	deep **dilated CNN** tower over tiled DNA (Basenji family)	deep conv. sequence model → dense tracks	feature stack + supervised E–G scoring (ABC-style)	Transformer + multimodal regulatory inputs	**BiMamba** + **MambaDNA** (RC-equivariant SSM stack)	selective **SSM**: input-dependent gating; recurrence uses fixed state dim. (not $T \times T$ maps)
Context strategy	long receptive field via dilations over sequence bins; multi-track heads	long sequence → base/bin outputs	logistic regression on **E–G pairs**; features = DNase/H3K27ac activity, Hi-C contact, ABC score, distance (not seq. LM)	promoter–enhancer pairs + activity/contact channels	**BiMamba** (fwd+rev SSM, shared weights) + **MambaDNA** (RC-equivariant); 1 bp tokens	selective **SSM operator**: input-dependent gating, linear-time recurrence; block in Caduceus/HybriDNA (not a DNA LM)
Context length	∼131kb-class tiled inputs (Basenji2-class; release-dependent)	**524,288 bp** input (∼524kb; Borzoi default seq. length)	**5 Mb** ABC search window per TSS; ∼**500 bp** DHS elements; most links <**100 kb** (86.8% atlas median)	region windows (task-dependent)	**131,072 bp** pretrain / eval (also 1k/32k ablations); 131k in LRB benchmarks	no fixed bp (architecture op.); cost $O(T)$/layer vs. $O(T^2)$ attention; state width N fixed (typ. 16–256)
Refs / code	Genome Biol., GH	Nat. Genet., GH	bioRxiv, GH	bioRxiv, GH	ICML, GH	arXiv

Hyena operator: detailed view

Goal: approximate long-range token interactions with convolutional-style operators whose cost scales better than dense attention

Notations

- $x_{1:T} \in \mathbb{R}^{T \times d}$: input hidden sequence.
- $u, v, g \in \mathbb{R}^{T \times d}$: three projections (value streams + gate).
- $\bar{u}_t = g_t \odot u_t$, $z_t = \bar{u}_t \odot v_t$: gated multiplicative stream.
- $y_t = \sum_{\tau \le t} K_\theta(\tau)\, z_{t-\tau}$: lag-weighted long convolution output.

From sequence to gated streams

$$x_{1:T} \in \mathbb{R}^{T \times d} \xrightarrow{W_u, W_v, W_g} u, v, g \in \mathbb{R}^{T \times d}$$

$$\bar{u}_t = g_t \odot u_t, \qquad z_t = \bar{u}_t \odot v_t$$

Implicit long filter (per channel, schematic)

$$y_t = \sum_{\tau=0}^{t} K_\theta(\tau)\, z_{t-\tau}, \qquad K_\theta(\tau) = \mathrm{MLP}_\theta(\tau) \cdot w(\tau)$$

where K_θ is generated implicitly (parameter function over lag τ), not stored as a dense $T \times T$ interaction matrix.

Why this is efficient

- **Memory:** no explicit attention map; avoids $O(T^2)$ activation footprint.
- **Compute path:** long convolutions can be implemented with FFT/structured kernels.
- **Expressivity:** multiplicative gating + learned long filters capture content-dependent interactions.
- **Stacking effect:** repeated Hyena blocks expand effective receptive field over genomic-scale spans.

StripedHyena: key ideas

What it is: a hybrid sequence architecture combining sparse attention-style mixing with Hyena implicit long convolutions to handle very long contexts efficiently.

Notation in the schematic below

- h: hidden activations at the input of one hybrid sublayer (full sequence tensor).
- $\tilde{h}$: after local/sparse mixing (attention- or Hyena-style *mixing* sublayer), still pre-FFN; residual path: $h \mapsto h + \mathrm{Mix}(\cdot)$ inside LN as in the detailed block later.
- (u, v): gated streams derived from $\tilde{h}$ (same role as on the Hyena slide: inputs to the implicit long filter).
- y: output of the long-filter / Hyena branch at this abstraction level (sequence-shaped tensor before the feed-forward).
- h': output of the full block after residual + layer norm + FFN applied to $\tilde{h}$ (or equivalently after the y branch is fused back along the residual).

Core design principles

- **Hybrid blocks**: alternate/select between attention-like layers and Hyena-family layers.
- **Implicit long filters**: model long-range interactions without building a full $T \times T$ attention matrix.
- **Gated multiplicative mixing**: combine content streams before long filtering to keep expressive interactions.
- **Hardware-aware efficiency**: reduce memory pressure and improve throughput at large T.

High-level schematic

$$h \xrightarrow{\text{local/sparse mix}} \tilde{h} \xrightarrow{\text{gating}} (u, v) \xrightarrow{\text{implicit long filter } K_\theta} y \xrightarrow{\text{residual+MLP}} h'$$

Why used in Evo2: keeps useful global genomic dependencies while staying practical on long DNA windows.

Evo2 architecture: overview

Evo2 uses a StripedHyena-style hybrid stack mixing attention-style operations and long implicit filters.

Symbols in the pipeline and in one block

- $x_{1:T}$: token IDs over the window; $e_{1:T}$: embeddings (dimension d); $h_{1:T}$: hidden states after stacked hybrid blocks.
- $\ell_{1:T}$: per-position logits over vocabulary $\mathcal{V}$; ℓ_t predicts the next token distribution $p_\theta(x_{t+1} \mid x_{\leq t})$.
- $\tilde{h}$: post-$\mathrm{Mix}_{\mathrm{hyena/attn}}$ (mixing sublayer inside LN + residual); h': block output after FFN sublayer (feed-forward expands/contracts then projects back to d).
- y_t (filter line): causal long-filter contribution at t built from past u, v streams and implicit K_θ (same intuition as Hyena slide; L is effective filter support width here).

$$x_{1:T} \xrightarrow{\text{tokenizer}} e_{1:T} \in \mathbb{R}^{T\times d} \xrightarrow{\text{hybrid blocks}} h_{1:T} \xrightarrow{W_{\mathrm{lm}}} \ell_{1:T} \in \mathbb{R}^{T\times|\mathcal{V}|}$$

$$p_\theta(x_{t+1} \mid x_{\leq t}) = \mathrm{softmax}(\ell_t)$$

One block (conceptual decomposition):

$$\tilde{h} = \mathrm{LN}(h + \mathrm{Mix}_{\mathrm{hyena/attn}}(h)), \qquad h' = \mathrm{LN}(\tilde{h} + \mathrm{FFN}(\tilde{h}))$$

Hyena-like long filter term (schematic):

$$y_t = \sum_{\tau=0}^{L-1} K_\theta(\tau)\left(u_{t-\tau} \odot v_{t-\tau}\right)$$

with K_θ parameterized implicitly (not a dense explicit full attention matrix).

Why this matters: lower memory pressure than naive full attention at very long T, while preserving rich sequence interactions.

Evo2 architecture overview (7B checkpoint)

Core dimensions

- $L = 32$ blocks (stack depth: 32 hybrid layers)
- hidden size $d = 4096$ (main hidden/embedding width before LM head); for probing we usually extract embeddings from **layer 27** (close to network center)
- inner MLP size $d_{\text{ff}} = 11008$ (FFN expansion inside each block)
- attention heads $H = 32$ ($d_h = d/H = 128$)
- vocab size $|\mathcal{V}| = 512$, char/byte-level token space, practical DNA tokens mostly used: `A,C,G,T,N` + special tokens (PAD, BOS, EOS, SEP)
- context: some Evo2 checkpoints are reported up to $\sim 10^6$ tokens; on one GH200, practical runs are around 1.5×10^5 tokens for 7B and 4000 for 40B (batch/precision dependent)

Layer-type schedule (index sets from config)

$$\mathcal{I}_{\text{attn}} = \{3, 10, 17, 24, 31\}, \quad |\mathcal{I}_{\text{attn}}| = 5$$

$$\mathcal{I}_{\text{hcs}}, \mathcal{I}_{\text{hcm}}, \mathcal{I}_{\text{hcl}} : \; 9 + 9 + 9 \text{ Hyena-family layers}$$

$\text{hcs}/\text{hcm}/\text{hcl}$: short/medium/long Hyena-convolution variants

Block equations (schematic)

$$h^{(l+\frac{1}{2})} = \text{LN}\Big(h^{(l)} + \text{Mix}^{(l)}(h^{(l)})\Big)$$

$$h^{(l+1)} = \text{LN}\left(h^{(l+\frac{1}{2})} + \text{FFN}^{(l)}(h^{(l+\frac{1}{2})})\right)$$

$l + \frac{1}{2}$ is the intermediate state after the Mix sublayer (attention or Hyena) and before FFN; this notation highlights the two residual substeps inside one block. where $\text{Mix}^{(l)}$ is attention for $l \in \mathcal{I}_{\text{attn}}$, else a Hyena variant.

$$y_t^{\text{hyena}} = \sum_{\tau=0}^{M-1} K_\theta^{(l)}(\tau)\, \phi(h_{t-\tau})$$

with implicit long filters $K_\theta^{(l)}$; $\phi(\cdot)$ is the local feature-mixing map (projection/nonlinearity) applied before long-filter aggregation.

Evo2-base (7B) parameter count: detailed breakdown

From runtime log + notebook marker `EVO2_STATS_JSON`:

$$N_{\text{total}} = 6{,}481{,}059{,}584, \qquad N_{\text{trainable}} = 6{,}481{,}059{,}584$$

Per-block parameter counts follow a repeating profile:

$$(202{,}426{,}112,\ 202{,}461{,}184,\ 202{,}559{,}488,\ 202{,}387{,}456)$$

up to small schedule-dependent permutations across indices.

Block-level decomposition (conceptual)

$$N_{\text{block}}^{(l)} \approx N_{\text{mix}}^{(l)}(d, H, \text{groups, filter len}) + N_{\text{ffn}}(d, d_{\text{ff}}) + N_{\text{norm/proj}}$$

$$N_{\text{ffn}} \sim 2\, d\, d_{\text{ff}} \text{ (dominant term)}$$

while $N_{\text{mix}}^{(l)}$ differs between attention and Hyena variants.

Global model

$$N_{\text{total}} \approx \sum_{l=1}^{32} N_{\text{block}}^{(l)} + N_{\text{embed/unembed}} + N_{\text{final norm}}$$

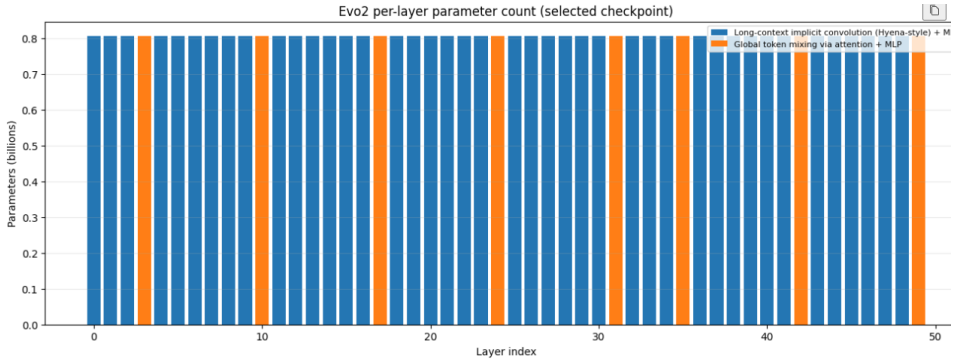
Why per-layer parameter counts are slightly different

Evo2 follows a hybrid block schedule (attention + Hyena variants), so $\text{Mix}^{(l)}$ is not strictly identical at every layer (projection/filter/gating tensors vary by block type). This induces small count deltas around the same $\sim 2.02 \times 10^8$ per-block scale; here the observed spread is 172,032 params ($< 0.1\%$ of one block), i.e. expected and modest.

Evo2-40B: checkpoint, parameter count, and layer-wise profile

Identifiers (same convention as `evo2_presentation.ipynb` / `EVO2_STATS_JSON`):

- `CKPT: evo2_40b_base`
- `HF_ID: arcinstitute/evo2_40b_base`
- **Total parameters:** 40,331,142,144 ($\approx$ 40.33B; trainable matches total in this run)



Per-layer parameter bars (Hyena-style blocks in blue, attention blocks in orange).

Dependencies and installation references

Upstream stack (container, Python toolchain, and model code should stay version-aligned on the target GPU node).

- **BioNeMo** — NVIDIA's BioNeMo *Framework*: training/inference tooling and containerized recipes for biomolecular and genomic foundation models; it standardizes the PyTorch/CUDA stack and pairs with pre-built images from NVIDIA's catalog. Official documentation: `docs.nvidia.com/bionemo-framework/latest/`.
- **NGC** (**NVIDIA GPU Cloud**) — NVIDIA's registry and hub for GPU-optimized *software artifacts*: container images, pretrained models, Helm charts, and resources for HPC/AI. Teams publish versioned BioNeMo images and related bundles there so deployments can `pull` a known-good stack instead of rebuilding from scratch. BioNeMo on NGC: `catalog.ngc.nvidia.com` (Clara BioNeMo collection).
- **Vortex** — open-source reference implementation of **StripedHyena**-style hybrid blocks (sparse or global mixing + implicit long convolutions). Evo2-family stacks use this so long DNA windows avoid a dense $T \times T$ attention budget in *every* layer; long-filter blocks scale more like convolutions than full attention. Source: `github.com/Zymrael/vortex`.
- **Runtime API (same stack)** — `model.generate(...)` returns continuations plus fields such as `sequences`, mean log-probs, and per-step `logits` (entropy / perplexity per step from logits). `return_embeddings=True` with named layers feeds lightweight task heads (e.g. exon classifier) without full finetuning.
- **Typical cluster deploy** — Singularity (or Apptainer) `exec` of an NGC/BioNeMo image, Python venv activated inside the container, explicit `-bind` mounts for weights and data; optional thin launch helpers (`run_command.sh`, notebook `utils.py`) reduce quoting/Slurm fragility.

Evo2 7B on GH200: first large context inference tests

Forward pass — full prompt teacher-forced; `model(input_ids)` → tuple (logits + aux).

- `example_sequence: ACGTACGTACGTACGT`
- `tokenizer_type: CharLevelTokenizer num_tokens: 16 input_ids_shape: [1, 16]`
- `forward_call: model(input_ids) return_type: tuple`

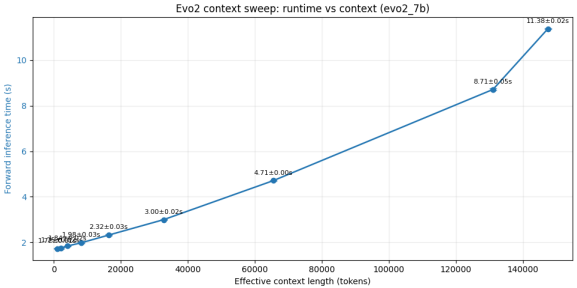
Autoregressive generation — `model.generate` (24 new tokens); vortex `scores_logprobs_mean`; logits $[1, 24, 512]$.

- `continuation[0]: ACGTACGTACGTACAATATATATA`
- seq. mean log p (sampled tokens): $-0.581880525\ldots$ $\exp(-\overline{\log p}) \approx 1.79$ (seq.-level perplexity analogue)
- `scores_logprobs_mean` (vortex): $[-0.581880569\ldots]$

Example:; H=entropy nats; $\mathrm{ppl}(\mathrm{tok}) = \exp(-\log p_{\mathrm{sampled}})$):

- step 1: `A` $|H$=0.646, $\exp(H)$=1.907, ppl=1.207 top: A:0.829, G:0.073, T:0.055, C:0.043
- step 8: `T` $|H$=0.407, $\exp(H)$=1.502, ppl=1.100 top: T:0.909, C:0.039, A:0.030, G:0.021

Full 24-step table: `EVO2_GENERATE_JSON` (notebook).



Long-context sweep (`evo2_7b`): forward time vs effective context length; 5 trials per size (mean ± std error bars). Refinement ⇒ ∼147k tokens last successful point.

Beyond ∼150k, observed failures include `CUDA_OOM` (e.g. 163,840 asks +40 GiB with only ∼31 GiB free) and `canUse32BitIndexMath` RuntimeError near higher lengths (e.g. 262,144).

On a single GH200, Evo2-40B is practically limited to ∼4,000 context for similar reasons.

Current investigation: container/Singularity versions first, then Vortex version; multi-GPU execution is being evaluated as a possible mitigation (and whether it is truly required).

Exon boundary test on evo2 embeddings

Goal: test whether Evo2 embeddings contain signal to discriminate exon-related windows.

(Non-biology recap.) DNA genes are transcribed to RNA; exons are retained after splicing, while introns are removed. Labels here mark exon/splice-context windows vs matched negatives.

Simplified exon schema (DNA → RNA)

Coding strand (sense)	`5' -[EXON 1]-(intron)-[EXON 2]-(intron)-[EXON 3]- 3'`
	`-> shown in 5' -> 3' direction`
Template strand (antisense)	`3' -[---]-(---)-[---]-(---)-[---]- 5'`
	`<- read by RNA polymerase in 3' -> 5' on template`
Transcription	`DNA template (read): 3' ------>------>------> 5'`
	`Pre-mRNA: 5' -[EXON 1]-(intron)-[EXON 2]-(intron)-[EXON 3]- 3'`
Splicing	`Mature mRNA: 5' -[EXON 1]---[EXON 2]---[EXON 3]- 3'`
	`-> ready for translation`

For each sample i, they used paired windows $s_i^{(f)}, s_i^{(r)}$ from the same locus (both DNA orientations) and concatenated the embeddings:

$$z_i^{(f)} = E_\theta(s_i^{(f)}, \ell), \quad z_i^{(r)} = E_\theta(s_i^{(r)}, \ell), \quad z_i = [z_i^{(f)}; z_i^{(r)}]$$

where $E_\theta(\cdot, \ell)$ extracts the last-token embedding at layer ℓ (here `blocks.26`).

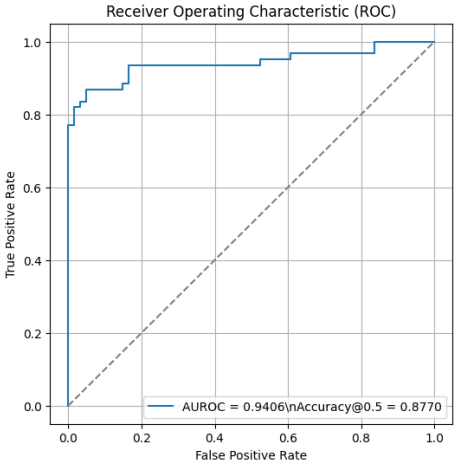
$$\hat{p}_i = \sigma\left(Wz_i + b\right)$$

with pretrained Hugging Face head `schmojo/evo2-exon-classifier`.

Pretraining format (head): frozen Evo-2 embeddings from paired DNA windows (`forward_seq`, `reverse_seq`; typically 8192 bp each) with binary exon labels.

How "exon detection" works here: no explicit donor/acceptor rule engine. We read last-token embeddings at layer ℓ, concatenate forward+reverse strands, then apply a linear head $\sigma(Wz+b)$. High scores indicate windows whose frozen representations match exon/splice-context positives.

Exon test results (reproduced from original Evo2 paper)



Setup: `samplePositions.tsv`, 122 rows from 8 species, balanced labels (61/61), paired 8192-bp windows (`forward_seq`/`reverse_seq`).

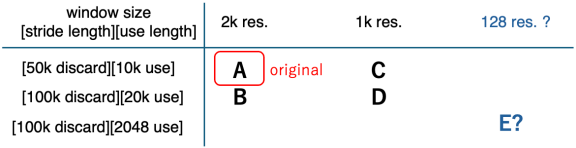
ROC is clearly above the random diagonal; $\mathrm{AUROC} \approx 0.9406$ and $\mathrm{Accuracy@0.5} \approx 0.8770$—strong exon/non-exon separation with **frozen** Evo2 + a **lightweight** (linear) head.

Reading the scores: high $\hat{p}$ means "embedding looks like the supervised positives" (exonic / splice-context windows in the training distribution), not that Evo2 was trained end-to-end on splice-site grammar. Because the **backbone is fixed**, high AUROC indicates the pretrained representation already organizes DNA patterns informative for this contrast.

Does the original paper interpret deeper? Brixi et al. (Evo2 preprint) mainly treat this as a **probing** / retrieval-style task on frozen embeddings. Separately, **sparse-feature / SAE interpretability** on comparable layers (Arc Institute + Goodfire) reports discoverable latent features aligned with **splicing and exon–intron structure**, among other genomic patterns— evidence that long-context DNA pretraining encodes biology-relevant sequence regularities, beyond a purely "token next-step" story .

Refs: Evo2 (bioRxiv) | Arc exon notebook | Goodfire/Arc: interpreting Evo2 (SAE features)

Evo2 embeddings and enhancer–promoter probing



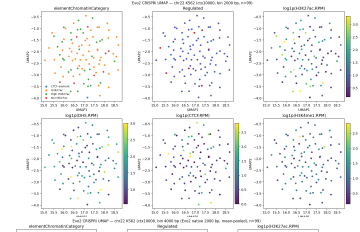
Goal. Fix a **single 320 kb window** on **hg38 chr22:20.0–20.32 Mb** (160 bins at **2 kb**) and a small **regulatory test set**, then vary context length and bin resolution to see how much promoter / enhancer / E–P geometry signal survives in frozen Evo2-7B embeddings (vs. being lost or over-smoothed).

Layer	Ground truth	Source	Note
Embeddings	8192-d / 2 kb bin; `blocks.27.pre_norm`	`hg38_2000_evo2_7b` memmaps	4096 fwd + 4096 rev; demo locus `chr22:20.0-20.32 Mb` (160 bins)
Per-bin class	`PLS` = promoter; `pELS`/`dELS` = enhancer	SCREEN cCRE v4 (ENCFF420VPZ)	epigenomic **annotation**, not CRISPR causality
E–P geometry	3 curated pairs (SORT1, MYC, Fig. 4d)	Fulco 2019 CRISPRi / literature pairs restricted to the chr22 window	benchmark pairs vs. unrelated positions in embedding space (distance / rank statistics)
QC	memmap vs live Evo2 forward	same chr22 locus; cosine similarity of per-bin embeddings	sanity check that precomputed memmaps faithfully reproduce the live model

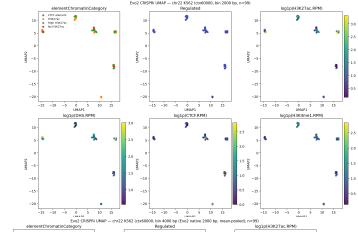
CRISPRi UMAPs at 2 kb / 4 kb resolution (different contexts)

K562 chr22 CRISPRi (n=99): mean-pooled Evo2 embeddings, UMAP colored by benchmark labels.
Top row: native **2 kb** bins | **Bottom row: 4 kb** aggregation (mean-pool adjacent 2 kb bins; same memmaps).

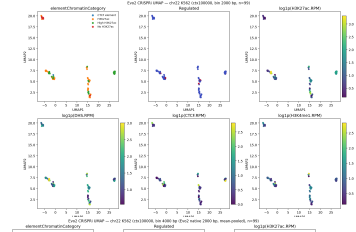
ctx 10 kb



ctx 60 kb



ctx 100 kb



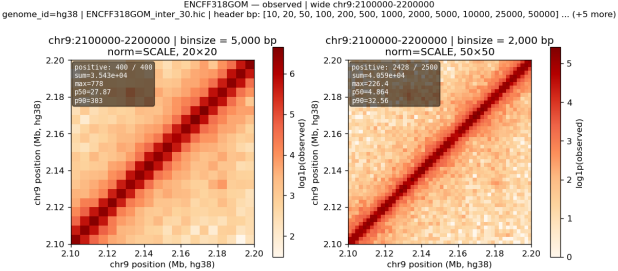
Hi-C: motivation and contact maps

Hi-C: genome-wide **pairwise contact frequencies** in a symmetric **contact matrix** (kb-scale bins). Captures **TADs**, loops, and **A/B compartments**.

Links **sequence** to **3D folding**; costly assay ⇒ denoising, super-resolution, sequence-to-contact models.

Juicer (documentation) → `.hic` files.

Reading the heatmap (`log1p` counts): **0** ≈ no interactions; **large** (∼3) ≈ many interactions. *Unfortunately*, contact signal **weakens with genomic distance** (bright diagonal, fading off-diagonal).

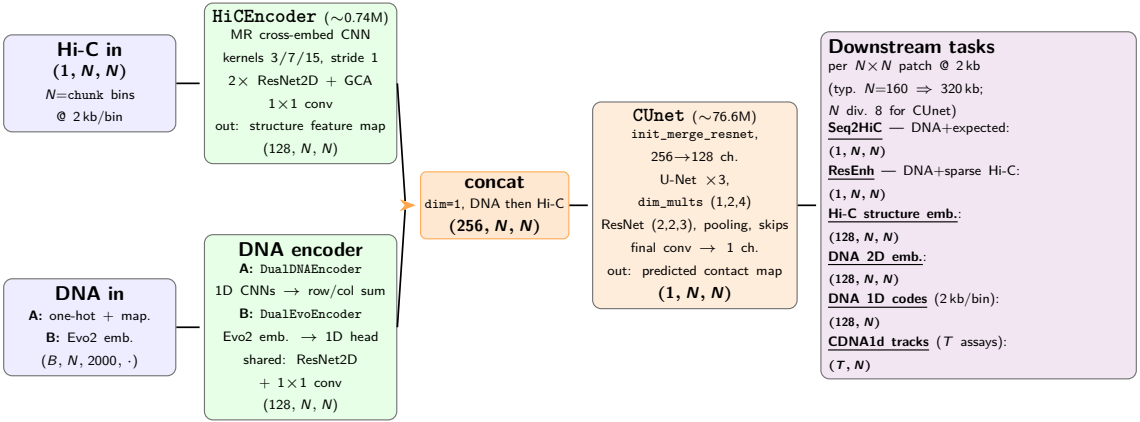


Observed Hi-C, ENCODE `ENCFF318GOM` (GM12878), **hg38 chr9:2.10–2.20 Mb**: **same 100 kb** at 5 kb (left) and 2 kb (right) resolution (`log1p`, `norm=SCALE`).

Evo2HiC architecture: big picture

Evo2HiC. Distill frozen DNA embeddings with **Hi-C structure supervision**.

Code: `Evo2HiC/train/pretrain.py`, `Evo2HiC/model/CDNA2d.py`, `Evo2HiC/model/siglip.py`.



Evo2HiC distilled model: objective and data flow

Overall idea. Learn a compact **DNA encoder** from two **frozen** teachers— no joint fine-tuning of Evo2 (7B):

sequence semantics from Evo2, 3D organization from the Hi-C encoder.

Joint pretraining.

A single **DNA encoder** is optimized on $\mathcal{L} = \mathcal{L}_{\text{seq}} + \mathcal{L}_{\text{struct}}$—two **SigLIP** contrastive losses (not cosine+MSE on one fused vector): **sequence distillation** aligns each 2 kb DNA embedding with **frozen Evo2** at the same bin (positives/negatives across bins); **structure distillation** (next slide) aligns each **2D DNA** patch with the **Hi-C encoder** at the same contact pixel, injecting **3D organization** (loops, TADs) alongside sequence semantics.

Sequence loss — implementation

Inputs: precomputed Evo2 embeddings per 2 kb bin (paper layer; code `blocks.27.pre_norm`); DNA encoder → 128-D/bin; linear heads → shared 512-D space.

Pairs: **positive** = same bin (Evo2 vs DNA); **negative** = different bins (paper adds genome-wide Evo2 negatives).

Sampling / step: six 200 kb windows on train chromosomes ⇒ 480 focal bins after 20 kb edge trim.

SigLIP weights $z_{ij} \in \{+1, -1\}$, learnable t_{seq}, b_{seq}, DNA x_i, Evo2 y_j:

$$\mathcal{L}_{\text{seq}} = \frac{1}{N_{\text{seq}}} \sum_{i=1}^{N_{\text{seq}}} \sum_{j=1}^{M_{\text{seq}}} \log \frac{1}{1 + \exp\left(z_{ij}\left(-t_{\text{seq}}\, x_i^{\top} y_j + b_{\text{seq}}\right)\right)}.$$

Code: `train/pretrain.py` (`loss1`); `model/siglip.py` (`siglip_DNA_evo2`); `dataset/evo2_embedding_loader.py`. **Ref:** bioRxiv | code

Evo2HiC distilled model: structure loss and training

(ii) Structure distillation — Hi-C encoder → DNA encoder

Per 2 *kb* × 2 *kb pixel:* Hi-C teacher embeds the local contact patch; DNA student builds a matching **2D** embedding from row/column bin sequences.

Pairs: **positive** = same pixel; **negative** = different pixels.

Sampling: six 200 kb×200 kb submatrices (2 kb bins) in a 500 kb diagonal window; eight random contact maps per submatrix, **averaged** per pixel for the Hi-C teacher; 20 kb borders dropped.

SigLIP weights $w_{ij} \in \{+1, -1\}$, learnable t_{struct}, b_{struct}, Hi-C u_i, DNA v_j:

$$\mathcal{L}_{\text{struct}} = \frac{1}{N_{\text{struct}}} \sum_{i=1}^{N_{\text{struct}}} \sum_{j=1}^{N_{\text{struct}}} \log \frac{1}{1 + \exp\left(w_{ij}\left(-t_{\text{struct}}\, u_i^{\top} v_j + b_{\text{struct}}\right)\right)}.$$

Total loss: $\mathcal{L} = \mathcal{L}_{\text{seq}} + \mathcal{L}_{\text{struct}}$ (slide 1: $\mathcal{L}_{\text{seq}}$).

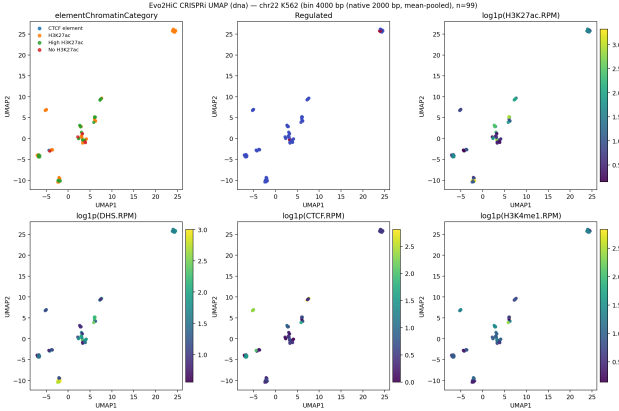
Training (paper). 4× A100, Adam, LR 10^{-4}, 50,000 steps (validation convergence).

Why frozen Evo2? Joint Evo2+Hi-C fine-tuning is too costly at genome scale.

Reported gain: ∼ 500× faster / lighter than direct Evo2 on paper tasks—re-benchmark on GH200/A100 .

Code: `train/pretrain.py` (`loss0+loss1`); `model/siglip.py` (`siglip_HiC_DNA`); `model/CDNA2d.py`. **Ref:** bioRxiv

Evo2HiC embedding quality



CRISPRi UMAP — distilled `DNA_encoder` (**128-d**), K562 `chr22` (n=99); mean-pooled over **4 kb** bins (native 2 kb, same locus as Evo2 EP bench).

What this shows. Mean-pooled Evo2HiC DNA-encoder vectors over CRISPRi-perturbed elements; colors from **EPCrisprBenchmark** (not inferred from embeddings).

Observations.

- Chromatin categories (`elementChromatinCategory`) partially separate in UMAP.
- `H3K27ac` / `DHS` signal tracks co-localize with enhancer-like clusters.
- `CTCF` elements form a distinct branch (low `CTCF.RPM` in the isolated cluster).

Quantitative vs Evo2 7B (chr22 160 × 2 kb; teacher **8192-d** fwd‖rev):

student	desc	stud→teach	CKA	R^2	lost %
`dna`	distilled `DNA_encoder` output (128-d internal)	128→8192	.192	.933	6.7
`dna_proj`	pretrain `projection0` Linear(128→512): maps `DNA_encoder` into shared space with projected Evo2 (SigLIP DNA–Evo2 loss)	512→8192	.300	.926	7.4
pca128	top-128 PCA of Evo2 (same-dim control)	128→128	.192	.934	6.6

lost % = $100(1-R^2)$: fraction of Evo2 variance **not** linearly recoverable from student.
CKA (geom.; 1 = same layout): **bad** here (∼0.2–0.3) — bins rearranged vs Evo2.
R^2 / lost % (info.): **good** — ∼93% retained, only ∼7% lost (same order as Evo2 PCA-128 control).
Conclusion.

Strong compressed proxy for Evo2 **information**; weak proxy for Evo2 **geometry**.

∼7% lost ⇒ **good** for info. (∼93% of Evo2 variance linearly recoverable; on par with PCA-128).
CKA ∼0.2–0.3 ⇒ **bad** for geometry (bins rearranged; UMAP ≠ Evo2).
`dna_proj` improves CKA (0.30) via `projection0`, not R^2.
Takeaway: Evo2HiC OK for linear probes; not drop-in Evo2 for neighborhood structure.

Comparaison with Borzoi: baseline for epigenomic track prediction

Borzoi (Kelley et al., Nat. Genet. 2025; code).

Input — **DNA sequence only.** One-hot **genomic DNA sequence** windows (**524 kb** tiled along hg38/mm10): **sequence alone**—no Hi-C, no precomputed LM embeddings—only nucleotides (A/C/G/T) along the reference.

Training data. Uniform ENCODE **RNA-seq** (866 human + 279 mouse biosamples), GTEx recount3 whole-tissue tracks, plus reprocessed **Enformer** assays (CAGE, DNase, ATAC, ChIP).

Architecture / scale. Enformer-style stack: **convolution** tower → **self-attention** at 128 bp → **U-Net** upsampling → **32 bp** resolution; four trained replicates for ensembling. Reported model size is about **186M parameters** (∼0.74 GB in FP32 weights; ∼0.37 GB in FP16/BF16 weights, excluding optimizer/activations).

Output. ∼ **6,300 parallel coverage tracks per window** (RNA-seq + CAGE/DNase/ChIP/ATAC; human `targets_human.txt`)—dense epigenomic profiles from sequence.

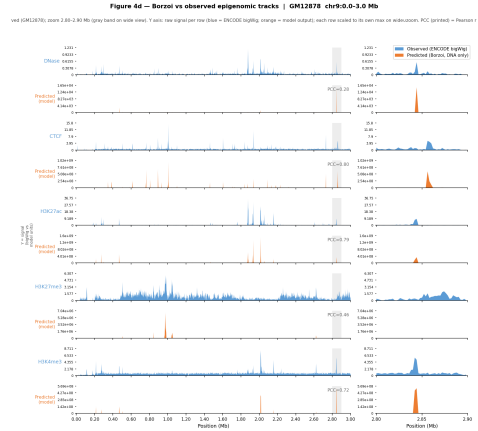
Why a baseline here?

- **Sequence-only** regulatory foundational model: predicts assays (incl. **CAGE**) from DNA without 3D contact maps.
- Same task family as our **CAGE** finetune on Evo2HiC `CDNA1d`, but Borzoi is **trained genome-wide on thousands of tracks**; we compare on held-out GM12878 loci.
- Contrast: Evo2HiC uses a **lighter multimodal encoder** (**Hi-C + DNA**) distilled from Evo2, whereas Borzoi is a larger sequence-only foundational model.

Checkpoint in repo: `johahi/borzoi-replicate-0` (HuggingFace; `datasets/sync_lib.sh -borzoi`).

calico/borzoi.

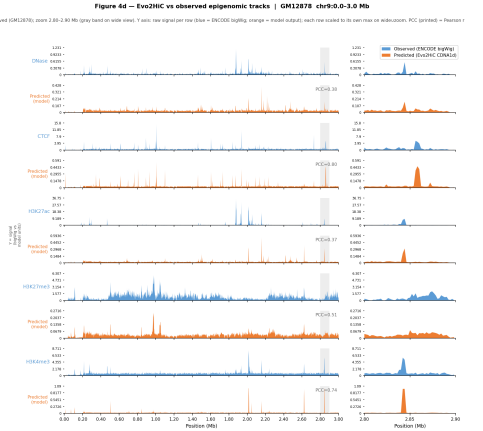
Borzoi vs Evo2HiC (GM12878 chr9) epigenomic predictions



Borzoi — DNA-only; `johahi/borzoi-replicate-0`; 15×524 kb.

	DNase	CTCF	H3K27ac	H3K27me3	H3K4me3
Pearson r	0.284	0.802	0.794	0.461	0.719

PCC = Pearson r; chr9 wide 0–3 Mb, 2 kb bins vs ENCODE bigWig (GT). Chr8 CAGE loci (§3): global pooled CAGE/DNase



Evo2HiC — `epi_prediction`; Hi-C `ENCFF129LMU`; 14 patches.

	DNase	CTCF	H3K27ac	H3K27me3	H3K4me3
Pearson r	0.379	0.801	0.368	0.510	0.740

Same metric/interval as Borzoi (`EVO2HIC_EPI_JSON` / `BOR2OI_FIG4D_JSON`).

Can evo2HiC also handle CAGE prediction? (CDNA1d)

Using the CDNA1d track head (`HiCEncoder + DNAEncoder + Track_Decoder`): same 1D assay branch as DNase/ChIP in `epi_prediction`, extended with a **CAGE** output channel .

Frozen teacher `epi_prediction` (Hi-C + DNA → 5 epigenomic tracks, **no CAGE**) defines a **DNase** readout. Student `CDNA1d` also predicts DNase alongside the new CAGE head; when `distill_weight` > 0, loss adds **MSE(student DNase, teacher DNase)** on top of bigWig supervision (weighted MSE+cosine: CAGE×8, DNase×1)—*preserve teacher epigenomic geometry while learning CAGE*. Smoke run: `distill_weight=0`; K562 ablation **T2d**: 0.3.

Short pipeline.

- **Student** CDNA1d: heads **(CAGE, DNase)**; init from frozen teacher `epi_prediction` (**no CAGE**—DNase row copied).
- **Windows:** GM12878 `hic2track`—**320 kb** (2 kb × 160); train on indexed chromosomes, **held-out chr8**.
- **Inputs / targets:** Hi-C `ENCFF318GOM` + hg38 DNA → predict `CAGE.bw` / `DNase.bw` (2 kb memmap).
- **Smoke finetune** (RCC, 1 GPU, ∼10 min): **6** steps, `distill_weight=0`; MSE+cosine vs bigWigs.

Next slide: comparative readout (three chr8 loci, same windows for Evo2HiC and Borzoi).

- **Setup:** top-3 held-out windows by CAGE variance; top panel = **CAGE**, bottom = **DNase**.
- **Ground truth (green):** ENCODE bigWigs → memmap (`tracks.r2000.values`, clip/scale ∼ [0, 1]; CAGE y-rescale for display only).
- **Evo2HiC** (`CDNA1d`, Hi-C + DNA): orange = prediction; box = PCC/Spearman vs memmap.
- **Borzoi** (`johahi/borzoi-replicate-0`, DNA only): orange = prediction; box = PCC vs bigWig and vs memmap.
- **PCC (smoke checkpoint):** Evo2HiC per-locus **CAGE 0.12–0.37**, **DNase 0.76–0.92**.

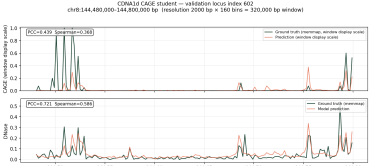
Comparative CAGE (+ DNase): Evo2HiC vs Borzoi — three chr8 loci

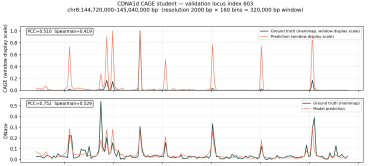
Locus 602
chr8:144.48–144.80 Mb

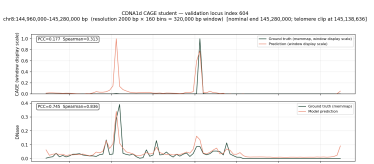
Locus 603
chr8:144.72–145.04 Mb

Locus 604
chr8:144.96–145.28 Mb

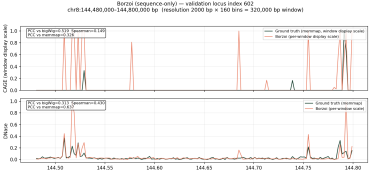
Evo2HiC CDNA1d

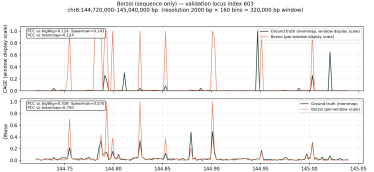


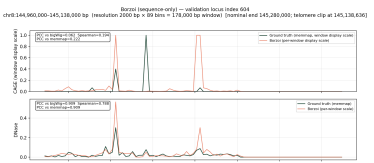




Borzoi (seq.)







CAGE by Evo2HiC: K562 finetune — head vs DNA encoder

CAGE is sparse on K562; we compare four finetune settings to see which `CDNA1d` blocks must be trained —decoder head only vs `DNA_encoder`, with or without joint DNase supervision.

Setup. K562 Hi-C `ENCFF616PUW` + hg38 DNA; 320 kb @ 2 kb; train chr1–7,10–22; valid chr8 (605 windows); 2000 steps; init from GM12878 `epi_prediction` teacher. Loss on supervised bigWigs = MSE + cosine (when CAGE+DNase: weights CAGE×8, DNase×1).

Four runs (T0–T2d). T0 (`head_only`, teacher embed, supervise CAGE): frozen teacher Hi-C+DNA embeddings → train `decoder.mlp` only.

T1 (`dna_encoder`, student embed, supervise CAGE): frozen `HiC_encoder`; retrain `DNA_encoder`+decoder.

T2 (idem T1, supervise CAGE+DNase): joint DNase bigWig loss (no distill).

T2d (idem T2, distill λ=0.3): T2 + teacher DNase distillation (preserve teacher epigenomic geometry).

chr8 validation

global = Pearson on **all** 2 kb bins (zeros included); *active* = bins with CAGE GT>0

Borzoi = zero-shot GM12878, 8 chr8 loci, pooled PCC @ 2 kb (`borzoi_presentation` §3, 20260608_active_pcc_v1)

metric	T0	T1	T2	T2d	Borzoi
CAGE global PCC	0.201	0.229	**0.230**	0.218	0.257
CAGE active PCC	0.213	**0.256**	0.255	0.242	0.242
DNase global PCC	0.534	0.348 | 0.494		**0.568**	0.338

Why PCC ≪ 0.7–0.8 (paper headlines)? Our setup (stricter): bin-wise Pearson on **raw** CAGE bigWig **@ 2 kb**, **all** bins pooled (sparse zeros dominate); whole chr8 valid (∼96k bins). **Typical paper setup (easier):** per-promoter / per-gene summaries, log-smoothed or coarser bins, often **active regions only**—not comparable to global bin PCC.

Conclusions. (1) Head-only (T0) weakest; `DNA_encoder` finetune (T1–T2) raises global CAGE ∼15%; T2 best global.
(2) Distillation (T2d) improves DNase (+0.07 vs T2) but lowers CAGE; active CAGE peaks at T1 (0.256), above Borzoi GM12878 baseline (0.242).

EPInformer: distal regulation baseline

EPInformer (Zhang et al., Nat. Commun. 2026; code):

predicts gene expression and ranks enhancer–promoter links from promoter sequence, candidate enhancer sequences, and multimodal epigenomics.

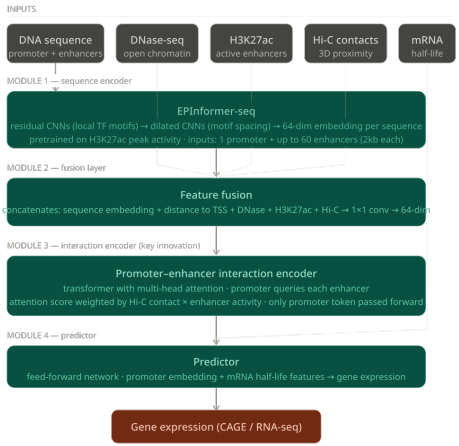
Task (per gene). One promoter (**2 kb** around TSS) + up to **60 candidate enhancers** (**2 kb** each) → predicted expression and attention-based enhancer scores.

Baseline candidate set (paper default).

- Search window: **100 kb** around TSS (best performance reported at **100–250 kb**; weaker beyond **500 kb**).
- Candidates: **DNase** hypersensitive (DHS) peaks in open chromatin, capped at **60** nearest sites (∼95% of elements in a 200 kb neighborhood).
- Activity priors: **H3K27ac** (and DNase) ChIP-seq signal at enhancer summits to prioritize/rank candidates.
- 3D context: promoter–enhancer **Hi-C** contacts (KR-normalized; ABC-style nomination in training data).

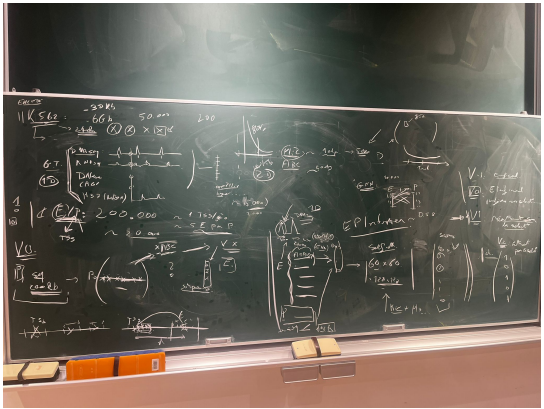
Architecture (4 blocks). (1) CNN sequence encoder on one-hot DNA ($61 \times 2000 \times 4$: 1 promoter + 60 enhancers); (2) fusion of sequence embeddings with distance, enhancer activity, and contact features; (3) transformer interaction encoder (3 layers, 4-head self-attention); (4) MLP head for expression (attention weights used for E–P prioritization).

Baseline scale / training. ∼**0.4M parameters** (∼447k); ∼**1 h** on one GPU to train one cell type (∼18k protein-coding genes in the paper). Trained per cell line (e.g. K562, GM12878); compact vs. Enformer-scale models.



Pipelines for distal regulation prediction

Goal: given an enhancer (easy to spot/well known, partly thanks to CAGE) and a context window where we want to spot enhancers involved in the regulation of a gene, we want the pipeline to come up with a list of candidate enhancers and a score for each of them.
First basic idea



Conclusions

- **Sequence-only GLMs (Evo2) hit context limits for distal E–P.** On GH200, Evo2-7B forward passes cap at ∼147–150 kb; chr22 embedding probes (CRISPRi / cCRE) show regulatory geometry depends on context and binning. Without 3D contacts, pure sequence models lack explicit chromatin-structure supervision.

- **Evo2HiC: lightweight Hi-C + DNA encoder—first quantitative readouts.** `CDNA1d` epigenomic tracks are competitive with DNA-only Borzoi on held-out loci (e.g. GM12878 chr9 DNase Pearson r: 0.379 vs 0.284). K562 CAGE finetune: head-only weakest (0.201 global PCC); `DNA_encoder` finetune slightly better (0.230 best global, 0.256 active at T1 vs Borzoi GM12878 0.242 active / 0.257 global @ 2 kb). Distilled embeddings retain ∼93% of Evo2 linear variance (∼7% lost) but differ in geometry (CKA ∼0.2–0.3).

- **Baselines clarify the distal-regulation stack.** Borzoi (524 kb DNA-only, ∼6.3k tracks) benchmarks sequence-first epigenome prediction; EPInformer (∼0.4M params) ranks E–G links from promoter + **candidate enhancer sequences** + DNase/H3K27ac/Hi-C. CAGE localizes promoters; **nominating and scoring distal enhancers** in 3D context remains the core difficulty.

- **Practical recipe from this work.** Combine multimodal inputs (Hi-C, accessibility, histone marks), feasible long context, explicit E–P benchmarks (CRISPRi, SCREEN), and penalties against TSS-proximal shortcuts. Evo2HiC is a strong **compressed proxy** for Evo2 **information** (linear probes, track heads), not a drop-in substitute for Evo2 **embedding geometry**.