DNA Foundation Models を読み解く

HyenaDNA → Evo → Evo 2｜7B scratch 学習の作り手目線

tax_free

2026/02/14

今日の 3 本

1. **HyenaDNA**（arXiv 2023）
2. **Evo (Science)**（7B, OpenGenome 300B tokens）
3. **Evo 2 (bioRxiv 2025)**（7B / 40B, OpenGenome2 9.3T tokens）

視点: 「7B を scratch で学習させる作り手目線」で、アーキテクチャ・データ設計・学習戦略・インフラ・意思決定のポイントに絞って整理

① HyenaDNA

🎯 何を解決したいモデルか？

従来の DNA LM の課題：

- Transformer（attention）が **$O(L^2)$** で長文が不可能
- 512–4k token しか扱えない
- k-mer tokenization で単塩基分解能を失う

→ 「**長距離 × 単塩基分解能**」を両立したい。

🧠 アーキテクチャ設計思想

Hyena operator（implicit long convolution）

- FFT ベースの long convolution（O(L log L)）
- Dense + Conv + Gate 構造
- MLP expansion factor = 4x / Order-N = 2
- 0.4M〜6.6M params / 最大 1M context

作る側の意思決定

- Attention を捨てる
- Context 拡張を最優先
- 幅よりも長さに振る設計

学習データ設計

- **単一の human reference genome**
- 次トークン予測（autoregressive）
- 単塩基トークン（4 token）

重要：

- 「巨大データで汎化」ではなく → **長距離構造を掴めるかの実験モデル**

⚙️ 学習設定

- AdamW / LR: 1.5e-4 – 6e-4 / Cosine decay
- 10–20k global steps
- 1M context モデルは **2T tokens / 4 週間**

最大の工夫：Sequence Length Warm-up

- 徐々に context を伸ばす
- 450k fine-tune 時も warm-up 使用

→ **超長文は一気に学習しない**

🧩 HyenaDNA：作る側から学べること

- 長文学習は「スケジューリング問題」
- データ量より「構造の帰納バイアス」
- Warm-up 無しで 1M はほぼ不可能

② Evo (Science, 7B)

🎯 Evo の狙い

HyenaDNA をスケールさせ **Genome-scale foundation model** を構築。

- 7B parameters
- 131k context

🧠 アーキテクチャ：StripedHyena

- 29 hyena layers + 3 attention layers (10%)
- RoPE positional encoding
- 単塩基 tokenization

→ 完全 attention ではない。Hyena 主体＋少量 attention

📊 データ設計：OpenGenome

- 300B tokens
- 80,000+ prokaryotic genomes
- ウイルス（euk 感染）は除外

学習戦略

2 段階 pretraining

- 8k → 131k context
- 合計 340B tokens
- 64 H100 + 128 A100

📈 Scaling law を事前分析

- 300+ models を事前に実験
- compute-optimal 7B は 250B tokens
- 実際は 300B tokens（17% overtrain）

→ 7B なら 250B 前後が理論最適

🧩 Evo：作る側から学べること

- 7B を作る前に scaling law を作れ
- Transformer++ は byte-level で不安定
- Hyena 系は安定

③ Evo 2 (7B / 40B)

🎯 スケールアップ

- 7B: 2.4T tokens
- 40B: 9.3T tokens
- 1M context

🧠 StripedHyena 2

- Multi-hybrid convolution + attention
- 1M context で 3× 高速
- 7B: 32 layers, hidden 4096

📊 データ工夫が最大の進化

OpenGenome2: 8.8T nucleotides

重要な発見：

- 「whole genome をそのまま入れると性能が落ちる」
- → genic windows を重視
- データ構成変更で AUROC 大幅改善

⚙️ Loss 再設計

- repetitive region を 0.1 重み
- reweight なしだと性能悪化

→ DNA 特有の loss engineering

🧩 Evo 2：作る側から学べること

- 「量より構成」
- noncoding を入れすぎると性能低下
- 7B でも 2.4T tokens 必要

🔥 横断まとめ：7B scratch でやるなら

1️⃣ アーキテクチャ

- Transformer 単体は不利（byte-level で不安定）
- Hyena/StripedHyena 系が有利
- Context 拡張は 2 段階以上が必須

2 データ規模

モデル	Tokens
HyenaDNA	2T（1M context 例）
Evo 7B	340B
Evo2 7B	2.4T

- 7B で最低 300B 以上
- 本気なら 1–2T

3 学習設計の鍵

- Sequence length warm-up
- 8k → 131k → 1M の段階拡張
- Loss reweighting

4 Compute

- **Evo:** 64 H100 + 128 A100
- **Evo2:** 9.3T tokens 40B モデル

→ 7B でも数百 GPU-week 規模