

# Beyond Conditional Computation: Retrieval-Augmented Genomic Foundation Models with Gengram

Huinan Xu<sup>1,2,3,4\*</sup>, Xuyang Feng<sup>3,5,6\*</sup>, Junhong Chen<sup>3,7\*</sup>, Junchen Liu<sup>1,3\*</sup>, Kaiwen Deng<sup>3,5\*</sup>,  
Kai Ding<sup>1,3</sup>, Shengning Long<sup>3,4</sup>, Jiaxue Shuai<sup>3,4</sup>, Zhaorong Li<sup>3</sup>, Shiping Liu<sup>3,4</sup>,  
Guirong Xue<sup>1,3</sup>, Zhan Xiao<sup>1,3\*†</sup>

<sup>1</sup> Zhejiang Lab, Hangzhou, China

<sup>2</sup> Hangzhou Institute for Advanced Study, Hangzhou;  
University of Chinese Academy of Sciences, China

<sup>3</sup> Genos Team    <sup>4</sup> BGI Research, Hangzhou, China    <sup>5</sup> BGI Research, Shenzhen, China

<sup>6</sup> College of Life Sciences, University of Chinese Academy of Sciences, Beijing, China

<sup>7</sup> BGI Research, Wuhan, China

Correspondence: xz@zhejianglab.org

## Abstract

Current genomic foundation models (GFM) rely largely on extensive neural computation to implicitly infer conserved biological motifs from single-nucleotide inputs. We introduce Gengram, a conditional memory module that provides an explicit and efficient lookup primitive for multi-base motifs through a genome-specific hashing scheme. Integrated into state-of-the-art (SOTA) GFM backbones, Gengram achieves substantial improvements of up to 22.6% across functional genomics tasks and generalizes across diverse architectures. Further analysis of Gengram’s latent space reveals biologically meaningful representations aligned with established genomic principles. By introducing structured motif memory as a modeling primitive, Gengram improves empirical performance and mechanistic interpretability, offering a scalable and biology-aligned direction for next-generation GFM. The code is available at <https://github.com/zhejianglab/Gengram>, and the model weights are available at <https://huggingface.co/ZhejiangLab/Gengram>.

## 1 Introduction

DNA sequences constitute the fundamental biological language of life, encoding genetic information that underlies cellular function and organismal development. Understanding the “syntax” of this language is central to deciphering biological processes and their dysfunctions (Kernohan and Boycott, 2024; Yang et al., 2025). Recent advances in genomic foundation models (GFM), including improvements in architectures and tokenization strategies such as k-mer, Byte Pair Encoding, and single-

base tokenization, have substantially accelerated progress in this direction.

While single-base tokenization performs well downstream, especially for evaluating fine-grained single-base effects (Lin et al., 2025), many biologically functional genomic elements are characterized by multi-nucleotide motifs with specific sequence patterns. Under standard Transformer architectures (Vaswani et al., 2017), existing GFM do not explicitly store or retrieve such motifs. Instead, they rely on large-scale pretraining and dense computation to implicitly infer motif-level information from combinations of single-base representations (Tzanakakis et al., 2026). This indirect modeling strategy is inefficient and may limit performance on motif-dominated functional element detection tasks.

To address this limitation, we propose Gengram, a conditional memory module for genomic motif modeling. Inspired by DeepSeek’s Engram, which stores n-gram knowledge in a hash-based memory (Cheng et al., 2026b), Gengram explicitly stores k-mers with  $k = 1, 2, 3, 4, 5, 6$  and their corresponding embeddings. Unlike Engram, which retrieves a single preceding n-gram at each position, Gengram aggregates multiple k-mer embeddings within a fixed local genomic window. This windowed memory retrieval allows the model to inject motif-level contextual signals into hidden representations.

Our contributions are summarized as follows:

**We propose Gengram**, to our knowledge, the first work to integrate a conditional memory mechanism into GFM. It explicitly represents multi-nucleotide motifs and integrates them into Transformer-based GFM through local window aggregation.

**Gengram demonstrates consistent and sub-**

\*Equal contribution

†Corresponding author

**stantial performance improvements** across downstream genomic tasks—especially motif-dominated functional element detection—achieving up to 22.6% improvement over SOTA GFM architectures under identical training protocols.

**Gengram is architecture-agnostic and lightweight.** Unlike Engram, which is primarily formulated as a sparsity axis complementary to MoE models, Gengram is validated across sparse MoE and dense Transformer backbones. Owing to the small genomic vocabulary (ACTGN), it adds negligible parameters (~60M) and computational overhead (Appx. Table 1), while consistently reducing training loss across architectures.

**Gengram is computationally efficient,** achieving benchmark performance comparable to SOTA models while utilizing fewer activated parameters and less training data. The Gengram module exhibits robust capability in modeling long sequences and enables stable load balancing in sparse Mixture of Experts (MoE) models.

**Gengram captures biologically meaningful structure.** Its memory embeddings exhibit reverse-complement symmetry, and its context-dependent gating aligns with functional genomic regions such as promoters and 5' UTRs. These patterns suggest that Gengram learns structured genomic representations rather than merely memorizing static sequence patterns.

## 2 Related Works

### 2.1 Architectures in GFMs

#### 2.1.1 Genomic Foundation Models

Current genomic modeling is dominated by Transformer-derived architectures, primarily including Bidirectional Encoders (e.g., DNABERT) (Ji et al., 2021) for local motif extraction via omnidirectional attention, and Autoregressive Decoders (e.g., Nucleotide Transformer) (Dalla-Torre et al., 2025; Boshar et al., 2025) which offer robust long-range modeling capabilities. Alongside these, non-Transformer alternatives, mainly represented by State Space Models (SSMs) (e.g., Caduceus) (Gu and Dao, 2023; Schiff et al., 2024) based on the Mamba architecture, achieve linear scaling via selective scanning.

#### 2.1.2 Representation Bottlenecks

The efficiency bottlenecks of these architectures often stem from the mismatch between tokenization strategies and biological units. Although single-

base tokenization is effective for high-resolution tasks like single-nucleotide polymorphism (SNP) prediction, genomic functional elements are fundamentally driven by multi-nucleotide **motifs**. Lacking explicit storage and retrieval mechanisms, models must allocate computational resources to reconstruct motif-level patterns from individual bases. This redundant modeling process limits feature extraction efficiency and imposes a performance ceiling on the models.

### 2.2 Memory-Augmented Architectures

DeepSeek Engram performs sparse table lookup for natural language using split-head hashing to handle large vocabularies. By injecting Engram residuals into only a small number of Transformer layers, it decouples static storage from dynamic memory and frees attention capacity for broader context modeling. Its deterministic addressing further supports prefetching and system-level optimization (Cheng et al., 2026b).

We adapt this idea to biological sequences with Gengram. Because the DNA alphabet is small and discrete (A/C/G/T/N) (Jaganathan et al., 2019), the  $N$ -gram space is more manageable and has substantially lower collision risk, enabling a hash-based memory table tailored to genomic models. Gengram further introduces local window retrieval, which retrieves multi-scale  $N$ -gram memories from the preceding window when predicting the current token. This design aligns with biological signals such as promoters, splice sites, and regulatory motifs, which often depend on upstream sequence context (Nguyen et al., 2023). Empirically, the window mechanism improves local motif modeling while supporting long-range dependency capture.

## 3 Method

### 3.1 Notation

Denote the input representation produced by the backbone and the previous-layer hidden states as:

$$X \in \mathbb{R}^{n \times d_{\text{model}}}, X^{(\ell-1)} \in \mathbb{R}^{n \times d_{\text{model}}} \quad (1)$$

where  $n$  is the sequence length, and  $d_{\text{model}}$  is the hidden dimension.

The  $N$ -gram lengths follow chapter 4.1:

$$N \in \mathcal{N} = \{1, 2, 3, 4, 5, 6\} \quad (2)$$

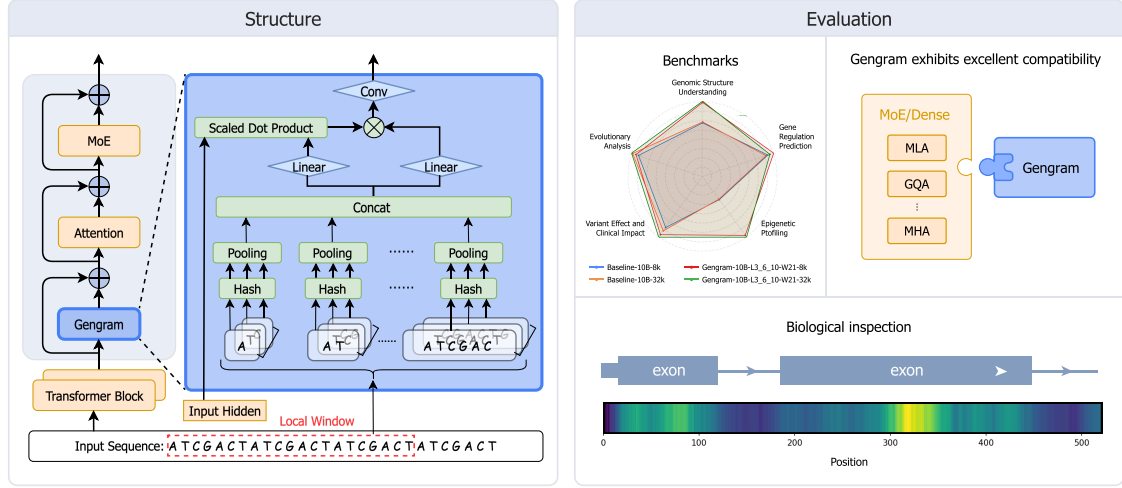


Figure 1: **Overview of the Gengram architecture and evaluation.** (Left) Gengram is inserted into a Transformer block via a residual connection before the attention module. It implements a motif memory with static k-mer keys ( $k=1-6$ ) and learnable embeddings, which are aggregated within a local window and integrated through a gate-controlled module to modulate hidden states. (Right) Evaluation of Gengram across diverse genomic tasks and architectures, demonstrating broad compatibility and biologically aligned activation patterns.

to capture multi-scale motifs. For each  $N$ , we maintain a learnable lookup hash table (memory):

$$\{\mathbf{M}^{(N)}\}_{N \in \mathcal{N}}, \quad \mathbf{M}^{(N)} \in \mathbb{R}^{|\Sigma|^N \times d_m} \quad (3)$$

where  $\Sigma$  is the token alphabet (A,T,C,G,N) and  $d_m$  is the memory vector dimension. Intuitively,  $\mathbf{M}^{(N)}$  stores an embedding for each possible  $N$ -gram key; the key-space size is  $|\Sigma|^N$ . The mapping from an  $N$ -gram  $g \in \Sigma^N$  to a row index is defined by a hash function:

$$h_N(\cdot) : \Sigma^N \rightarrow \{0, 1, \dots, |\Sigma|^N - 1\} \quad (4)$$

Since  $|\Sigma|$  is small,  $h_N(\cdot)$  can use a deterministic base- $|\Sigma|$  encoding, giving efficient collision-free indexing.

### 3.2 Architecture Overview

The Gengram module integrates efficient lookup with a motif prior (Fig. 1): when predicting the current token, it uses a sliding-window mechanism within the Local Window immediately preceding it to retrieve all occurring contiguous substrings, i.e., motif priors. After deduplication, it looks up the corresponding values in a hash table; vectors with the same  $N$  are summed and averaged, vectors across different  $N$  are concatenated, and then, after a linear transformation, dot-product computation, and gating, the resulting signal is added to the backbone network via a residual connection.

### 3.3 Local Window and N-gram Enumeration

For each position  $t$ , we define a causal window index set, where  $W$  denotes the causal window size:

$$\mathcal{W}_t = \{t - W, t - W + 1, \dots, t - 1\} \quad (5)$$

We scan  $N$ -grams over the local window  $\mathcal{W}_t$  to obtain a complete set of substrings, enabling the model to capture multi-scale motif information. The local window also reduces noise and supports efficient retrieval.

Within this window, we enumerate all contiguous  $N$ -grams using a sliding window scan, as shown in Appx. Algorithm 1:

$$\mathcal{G}_t^{(N)} = \{(s_j, \dots, s_{j+N-1}) \mid j \in [t - W, t - N]\} \quad (6)$$

where  $s_j$  is the discrete token at position  $j$ . To make the aggregation focus on distinct motif types rather than their raw frequency, we first deduplicate the retrieved substrings:

$$\tilde{\mathcal{G}}_t^{(N)} = \text{Unique}(\mathcal{G}_t^{(N)}) \quad (7)$$

### 3.4 Hash Retrieval and Aggregation

Each  $N$ -gram  $g \in \Sigma^N$  is retrieved from the corresponding table by

$$v(g) = \mathbf{M}^{(N)}[h_N(g)], \quad g \in \tilde{\mathcal{G}}_t^{(N)} \quad (8)$$

We then average the retrieved vectors to obtain a fixed-size summary for length  $N$ :

$$m_t^{(N)} = \frac{1}{|\tilde{\mathcal{G}}_t^{(N)}|} \sum_{g \in \tilde{\mathcal{G}}_t^{(N)}} \mathbf{M}^{(N)}[h_N(g)] \quad (9)$$

We apply mean pooling to compress a variable number of retrieved  $N$ -gram values into a fixed-dimensional summary  $m_t^{(N)}$ . This choice stabilizes the magnitude of  $m_t^{(N)}$  across positions and sequences. Such scale stability makes the contribution of the memory branch easier to control and more robust during training.

### 3.5 Cross- $N$ Fusion

We fuse multi-scale summaries by concatenation:

$$m_t = \text{Concat}(m_t^{(1)}, m_t^{(2)}, m_t^{(3)}, m_t^{(4)}, m_t^{(5)}, m_t^{(6)}) \quad (10)$$

We then apply two linear projections to the same  $m_t$ :

$$z_t = m_t W_z + b_z, \quad u_t = m_t W_u + b_u \quad (11)$$

Here  $z_t$  parametrizes a token-conditioned gate, while  $u_t$  carries the memory content to be injected.

### 3.6 Scaled Dot-product Gating

We compute a gating scalar via a scaled dot product between normalized vectors:

$$\tilde{u}_t = \sigma \left( \frac{\text{RMSNorm}(X_t)^T \text{RMSNorm}(z_t)}{\sqrt{d_{\text{model}}}} \right) u_t \quad (12)$$

We compute a gating scalar from the scaled dot product of the normalized input  $X_t$  and normalized projection  $z_t$ , where  $\sigma(\cdot)$  is the sigmoid function and RMSNorm is root mean square normalization, to control how strongly the retrieved motif memory affects the backbone at each position; it is then dotted with  $u_t$ .

Then we align and mix the gated memory signal through a lightweight transformation, and add it residually to the previous-layer hidden state:

$$\hat{u}_t = \tilde{u}_t + \text{SiLU} \left( \text{DWConv1D}(\tilde{U})_t \right) \quad (13)$$

$$Y_t = X_t^{(\ell-1)} + \hat{u}_t \quad (14)$$

DWConv1D denotes a depthwise 1D convolution applied along the sequence ( $\tilde{U}$ ) dimension with groups =  $d_{\text{model}}$ . This short convolution captures local sequential dependencies in the gated

memory signal while keeping the computation lightweight through channel-wise filtering. The resulting  $Y \in \mathbb{R}^{n \times d_{\text{model}}}$  is fed to subsequent network components. This residual formulation stabilizes training while allowing the model to selectively incorporate explicit motif-level information from the windowed context.

### 3.7 Time Complexity (shows as $O(n)$ )

For each position  $t$  and each  $N \in \mathcal{N}$ , we enumerate  $\max(0, W - N + 1)$  contiguous  $N$ -grams within the causal window and perform one table lookup ( $O(1)$ ) per  $N$ -gram. Therefore, the cost over a length- $n$  sequence is:

$$O \left( n \sum_{N \in \mathcal{N}} \max(0, W - N + 1) \right) = O(n|\mathcal{N}|W) \quad (15)$$

With fixed hyperparameters  $W$  and  $\mathcal{N}$ , the retrieval complexity is linear in the sequence length, i.e.,  $O(n)$ .

## 4 Experimental Design

### 4.1 Multi-scale K-mers

The selection of multi-scale k-mers is motivated by both biological and computational considerations. Biologically, different functional genomic elements are characterized by sequence patterns of varying lengths: splice sites are determined by short dinucleotide signals (Mount, 1982), codons encoding amino acids consist of triplets (Crick, 1968), while other regulatory elements—such as promoter motifs (e.g., the TATA box)—span longer sequence segments (Smale and Kadonaga, 2003). Computationally, we restricted k-mers to lengths up to 6 based on the efficiency–precision trade-off validated in GENERator (Wu et al., 2025). As shown in Table 3, the k-mer ablation study confirms that using the full k-mer range ( $k = 1$ –6) achieves the best overall performance across downstream tasks.

### 4.2 Gengram Layer Selection Experiments

Gengram is a conditional memory module that only needs to be inserted into a small subset of Transformer layers (Cheng et al., 2026b). Using our 1.2B model with 0.3B activated parameters (Appx. Table 2), we first varied the insertion layer while keeping all other settings fixed, and selected the best single-layer placement based on validation loss on a unified validation set.

The layer-sweep results show that validation loss generally decreases when Gengram is inserted into deeper layers (Fig. 2, top). Shallow hidden states mainly encode local patterns and short-range statistics (Jawahar et al., 2019), which can help detect motif evidence but may also increase over-retrieval. In contrast, deeper hidden states contain richer contextual and positional information after multiple rounds of attention aggregation, providing more informative signals for the gate and enabling better discrimination between functional motifs and non-functional repeats (Vig and Belinkov, 2019; Avsec et al., 2021).

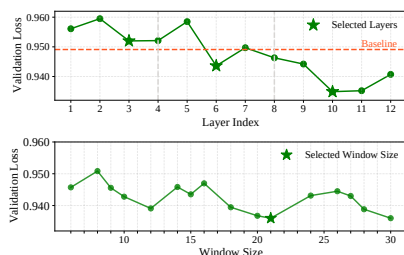


Figure 2: **Parameter Selection Experiments.** **Top:** Validation loss when inserting Gengram into individual Transformer layers in a 1.2B-parameter (0.3B activated) model trained on 50B tokens. The red curve denotes the baseline without Gengram. **Bottom:** Validation loss under different Gengram window sizes.

Because different depths may provide complementary information, we further evaluated multi-layer insertion patterns. Motivated by the hierarchical information-capturing behavior of Transformer layers (Skean et al., 2025) and inspired by cortical hierarchy, which progresses from posterior sensory feature extraction to medial integration and anterior high-level decision-making (Margulies et al., 2016), we selected layers {3, 6, 10} as a shallow/middle/deep combination. Analogously, the {3, 6, 10} configuration is intended to span shallow, middle, and deep stages of the model, capturing the progression from local sequence scanning to high-level functional abstraction.

We also tested competitive alternatives guided by the single-layer rankings, including {10, 11, 12}, {9, 10, 11, 12}, {3, 6}, and {3, 10}. As shown in Appx. Table 4, {3, 6, 10} performed best overall, ranking first in four of the five downstream evaluation categories.

### 4.3 Window Size Selection Experiments

Using the training setup of the 1.2B-parameter model (Appx. Table 2), we empirically selected the

Gengram window length from 17 candidate sizes based on multiples of 2 and 3. We kept the backbone, data, token budget, and all major training hyperparameters fixed, and inserted Gengram at the tenth layer based on preliminary single-layer experiments. Each variant was trained for 50B tokens and evaluated on the same validation set. The window length with the lowest validation loss was then used as the default setting for subsequent experiments.

The validation loss reached its minimum at a window size of approximately 21 bp (Fig. 2, bottom). Importantly, this window size was selected solely based on validation performance rather than according to a predefined biological prior. Matched-parameter local-context controls further show that the improvement is not reproduced to the same extent by either a convolutional local branch or static k-mer pooling (Appx. D), suggesting that the benefit cannot be attributed solely to generic local aggregation.

Interestingly, 21 bp approximately corresponds to two turns of B-form DNA, which has a helical repeat of about 10.4–10.5 base pairs (Wang, 1979). This correspondence provides a potential biological interpretation, as helical phasing can influence the relative spatial orientation of nearby sequence elements. However, our experiments do not establish a causal relationship between DNA helical periodicity and the observed performance gain. We therefore regard this correspondence as a hypothesis that may motivate future mechanistic investigation, rather than as an explanation for why the 21-bp window is optimal.

### 4.4 Component Ablation Study

As shown in Appx. Table 5, the Gengram baseline (mean pooling with gate) achieved the lowest validation loss. In particular, the gating mechanism is crucial, as removing it led to a substantial increase in validation loss. Note that Fig. 2 and Table 5 come from independent experiments on different GPU platforms where distributed training is not bitwise deterministic, only relative loss changes within each batch are comparable, not the absolute values.

### 4.5 Pretraining Setup

To validate the effectiveness of the Gengram module, we trained a 10B-parameter model, Gengram-10B-L3\_6\_10-W21-8k (hereafter, **Gengram-10B-8k**), using the Megatron-LM framework on a dis-

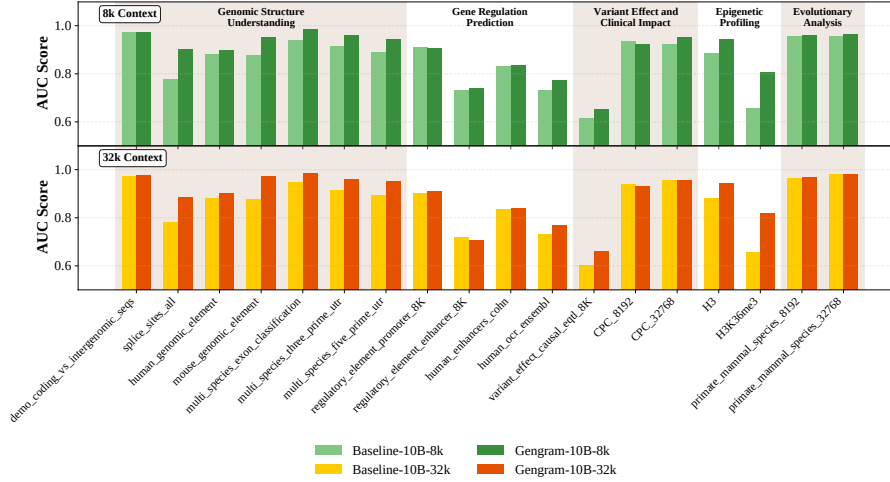


Figure 3: Performance Comparison of Baseline and Gengram Models at 8k and 32k Context Lengths.

tributed system with 8-GPU nodes and pipeline parallelism of 2. Architecture and training details are provided in Appx. B. To ablate the contribution of the local window mechanism, a variant without the window, Gengram-10B-L3\_6\_10-8k, was also trained. Additionally, we trained a baseline model without the Gengram module named Baseline-10B-8k for comparison.

## 5 Results & Analysis

### 5.1 Performance of the Gengram Module

A systematic evaluation on human-centric and multi-species tasks demonstrates the effectiveness of Gengram for genomic sequence modeling. Compared with Baseline-10B, which corresponds to Genos-10B without Gengram, Gengram-10B-8k improves almost all metrics (Fig. 3), indicating a stronger ability to capture complex genomic patterns.

Ablation results show that the local window mechanism is central to these gains. Gengram-10B-8k consistently outperforms its window-free counterpart across downstream tasks (Appx. Table 6), with especially clear improvements on fine-grained local sequence tasks such as splice site detection and histone mark prediction. Notably, Gengram-10B-8k, trained with only an 8k context length, matches or outperforms Baseline-10B-32k, which is adapted from Baseline-10B-8k to 32k sequences through continued pretraining (CPT) (Appx. Table 7). This suggests that Gengram improves long-range sequence modeling while reducing reliance on costly CPT for extended contexts. As an additional long-sequence

regulatory evaluation, we tested Gengram on the eQTL\_Skin\_Not\_Sun\_Exposed\_Suprapubic task from DNALONGBENCH (Cheng et al., 2025), which uses 450-kb genomic sequences. Gengram-10B-8k achieved 0.8325, compared with 0.8282 for the matched Baseline-10B-8k. This targeted comparison was designed to isolate the effect of the Gengram module under the same setup and provides additional evidence that its benefits can extend to long-sequence regulatory settings.

The CPT results further support this interpretation. Extending the window-free Gengram-10B-8k model to 32k sequences yields substantial gains, whereas applying the same procedure to the window-equipped Gengram-10B-8k produces only marginal improvements (Appx. Table 6). This contrast suggests that local window aggregation already captures part of the benefit typically obtained from long-context adaptation. We attribute this effect to an architectural inductive bias: stacked local windows prioritize contiguous sequence patterns while progressively expanding the effective receptive field, allowing Gengram to compose local motif-level features into broader contextual representations under a shorter nominal context length.

Matched-parameter controls further confirm that Gengram’s gains arise from retrieval-gated motif memory rather than merely adding local context (Appx. D).

### 5.2 Benchmarking Results

To contextualize Gengram’s empirical performance, we compare Gengram-10B-8k with existing GFMs, including Genos-10B (Lin et al., 2025) and Evo2-40B (Brix et al., 2026). Despite being

trained on only 200B tokens, compared with 2.2T for Genos-10B and 9.3T for Evo2-40B, Gengram-10B-8k achieves competitive performance across genomic benchmarks (Appx. Table 9). It obtains the highest AUC on Multi-species Exon Classification (0.9832), competitive performance on Splice Site Identification (0.9009), and higher scores than Evo2-40B on Primate-Mammal Species Classification. Because these models differ in training data, token budgets, context lengths, and architectures, we use these cross-model results only to contextualize empirical competitiveness rather than to attribute gains specifically to Gengram.

We therefore rely on controlled comparisons to assess the contribution of Gengram. In the from-scratch setting (Fig. 3), Gengram outperforms the matched baseline on nearly all downstream tasks under identical training and evaluation protocols. In the CPT experiment (Appx. Table 10), both variants start from the same Genos-10B checkpoint and receive the same 200B-token continued-pretraining budget; Gengram improves 11 of 12 tasks, including Human Enhancers Cohn (0.8286→0.8616) and causal eQTL (0.6742→0.7027), while CPC-8K decreases slightly (0.9518→0.9506).

These results suggest that Gengram provides complementary, task-dependent gains on a well-trained backbone, with generally stronger benefits on motif-rich tasks and less consistent improvements on tasks dominated by fine-grained single-nucleotide effects.

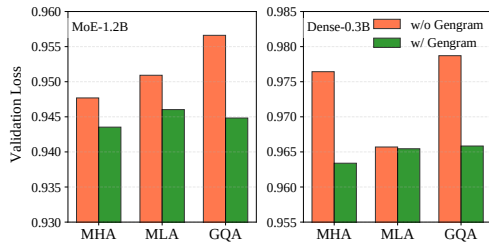


Figure 4: **Validation-Loss Comparison.** Validation-loss bar chart under MoE and Dense architectures across GQA/MLA/MHA.

### 5.3 Generalizability Assessment of Gengram

Adding the Gengram module consistently reduces validation loss across architectures and attention mechanisms (Fig. 4). Gengram is integrated into selected Transformer blocks as a residual branch, allowing it to remain compatible with different network designs. To evaluate its cross-architecture generality, we conducted controlled experiments

across two architectural families: sparse MoE models with 1.2B total and 0.3B activated parameters, and dense models with 0.3B parameters (Liu et al., 2024). Within each family, we further combined Gengram with three mainstream attention mechanisms: Multi-Head Attention (MHA; (Vaswani et al., 2017)), Multi-Head Latent Attention (MLA; (Liu et al., 2024)), and Grouped-Query Attention (GQA; (Ainslie et al., 2023)). All models were trained for 50B tokens, resulting in six experimental groups for systematic comparison. Detailed configurations are provided in Appx. Tables 11, 12, and 13.

These results show that Gengram provides stable gains across diverse architectures. Importantly, the improvements observed on dense backbones indicate that the benefits of conditional motif memory are not tied to expert routing, but extend to standard dense Transformer architectures. As a lightweight residual module with a small parameter footprint, it improves model performance with minimal training overhead, suggesting its potential as a reusable component for future genomic foundation models.

### 5.4 Gengram Stabilizes MoE Load Balancing

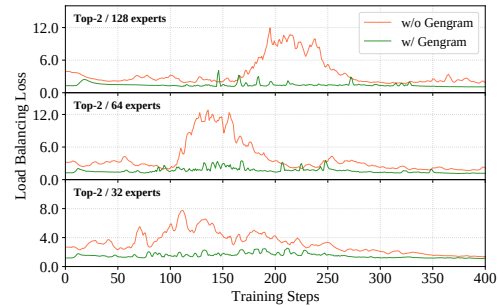


Figure 5: **Cross-Sparsity Load-Balancing.** Load-balancing loss curves w/o the Gengram module under Top-2 / 128, 64, 32 experts, showing consistent stabilization across sparsity settings.

Load imbalance is a common challenge in sparse MoE training, and prior work such as ST-MoE introduces router  $z$ -loss to improve stability (Zoph et al., 2022). During training, we observed that MoE models equipped with Gengram exhibited more stable load balancing than models without Gengram. To verify this observation, we conducted controlled tests on a 10B model under three sparsity settings: Top-2 routing with 128, 64, and 32 experts, while keeping all other settings unchanged. Detailed configurations are provided in Appx. Table 14.

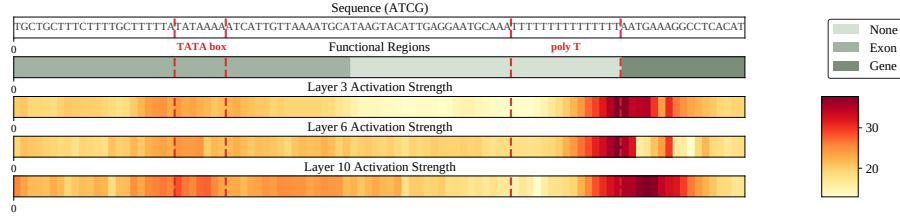


Figure 6: **Gengram residual contribution aligns with functional elements.** Top: Sequence with motif hits (TATA-like, poly-T) and region labels. Bottom: Gengram residual-write strength  $s_\ell(t)$  at L3/6/10. Peaks align with motifs/boundaries and are slightly left-shifted to motif ends, consistent with forward-only windowed matching.

The results show that Gengram maintains stable routing behavior across sparsity levels (Fig. 5). We attribute this stability to its gated memory design. In sparse computation, task-irrelevant high-frequency noise can distort routing or activation scores, causing a small number of experts to be repeatedly selected and reinforced (Cheng et al., 2026a). This issue is especially relevant for genomic data, where repetitive and low-complexity regions are abundant, and many frequent  $k$ -mers are weakly informative for biological function (Haubold and Wiehe, 2006; Li et al., 2014). Gengram absorbs local  $k$ -mer patterns into the memory branch, but injects the aggregated memory output into the backbone only through a gate. This gate can increase sparsity (Qiu et al., 2026) and suppress noisy memory signals unsupported by informative backbone representations. As a result, Gengram reduces noise-driven route reinforcement, suppresses variance in activation and routing scores, and yields a more uniform and stable load distribution even under higher sparsity.

## 5.5 Intrinsic Mechanisms and Interpretability

### 5.5.1 Training Acceleration

Gengram accelerates optimization by front-loading predictive computation: it helps shallow representations align with the model’s final prediction earlier, reducing the effective depth required to reach a prediction-ready state (Cheng et al., 2026b). We quantify this “prediction readiness” using a LogitLens probe (Nostalgebraist, 2020), where smaller KL divergence under a fixed readout indicates that an intermediate representation is closer to the final prediction distribution.

The LogitLens results show that Gengram produces more prediction-aligned intermediate representations (Fig. 8). This suggests that predictive information becomes accessible at earlier layers and may reduce the need for deeper layers to re-

construct motif-related cues from scratch. Adding a local window ( $W = 21$ ) further promotes earlier emergence of prediction-aligned intermediate representations. Overall, the KL-divergence results suggest that Gengram enables earlier prediction-aligned representations, providing a plausible explanation for its optimization benefits.

### 5.5.2 Gengram Residuals Align with Motif-like Sequence Features

We find that Gengram’s intervention along a genomic sequence is highly sparse and high-contrast. We define the activation strength at layer  $\ell$  and position  $t$  as  $s_\ell(t) = \|\text{Gengram}_\ell(\mathbf{h}_\ell(t), x_t)\|_2$ . The per-position residual-write magnitude  $s_\ell(t)$  concentrates into sharp peaks at a small fraction of positions, while most positions remain near baseline (Fig. 6). These peaks occur near motif-like elements and functional boundaries, including promoter-proximal TATA-like substrings and low-complexity poly-T tracts. We next test whether this localized pattern generalizes genome-wide in Sec. 5.5.3.

The spatial distribution of residual writes shifts with depth. In L3 and L6, the strongest peaks are anchored near the poly-T/boundary locus, with both layers reaching maximum activation at position 82, near the boundary between intergenic and gene regions. In contrast, L10 peaks at position 86 within the gene region, and most of its top-10 peaks are located in gene regions (8/10 for L10 vs. 3/10 for L6). This boundary-to-gene shift is consistent with a hierarchical interpretation: early layers may emphasize localized motif and boundary evidence, while deeper layers may integrate this evidence into context-dependent representations.

### 5.5.3 Genome-wide Enrichment and In-silico Perturbation

To determine whether the localized residual-write patterns observed above generalize genome-wide,

we analyzed 1,000 non-overlapping 4,096-bp hg38 windows using chromosome-level discovery and held-out validation splits. High residual-write positions at layers 3, 6, and 10 were compared with matched background positions controlling for genomic annotation, GC content, repeat coverage, and sequence complexity. Of 481 candidates identified exclusively in discovery, 320 replicated in validation (66.5%); among these replicated candidates, 99.3% retained the same effect direction, with highly concordant effect rankings (Spearman  $\rho = 0.909$ ). Under a stricter dinucleotide-preserving empirical null, 130 candidates remained significant at empirical FDR  $q < 0.05$ , supporting reproducible sequence preferences beyond simple composition effects (Appx. E.1).

We further shuffled candidate cores while preserving sequence length and mononucleotide composition. Across layers 3, 6, and 10, core shuffling caused larger residual-write decreases than either composition-matched distal regions or adjacent flanks, with bootstrap confidence intervals consistently above zero (Appx. E.2). Thus, Gengram responses are sensitive to local nucleotide order and context rather than composition alone. After controlling for low-complexity and repeat-associated sequences, no JASPAR/FIMO TF motif remained significant in held-out validation; we therefore interpret these signals as reproducible, local-order-sensitive motif-like sequence memory rather than TF-specific binding memory.

## 6 Conclusion

In this work, we introduced Gengram, a conditional memory module that explicitly encodes and retrieves multi-nucleotide sequence patterns through a structured memory pathway. Through a gate-controlled memory pathway with local window aggregation, Gengram enables direct motif-level access while preserving the flexibility of standard attention-based sequence modeling.

Across extensive experiments, Gengram improves both performance and training efficiency over prior state-of-the-art GFMs. It yields substantial gains on motif-dominated tasks, strengthens long-range sequence modeling, and improves load-balancing behavior in sparse MoE models. Gengram also remains effective across diverse architectures and attention mechanisms while introducing negligible computational overhead.

Beyond empirical improvements, mechanistic

analyses reveal reproducible, context-dependent, and local-order-sensitive sequence responses, supporting Gengram as a conditional motif-like memory rather than a static sequence dictionary. Taken together, these results suggest that Gengram provides a scalable and interpretable direction for future GFMs, with potential extensions to longer sequence contexts and multi-omics modeling.

## 7 Limitations

This work has several limitations that point to important directions for future research. First, our current study primarily validates Gengram as a modular architectural component under the training scales considered here. A systematic investigation of its scaling behavior, including larger training corpora, longer context lengths, and extended training schedules, remains an important next step. Second, Gengram currently employs fixed k-mer levels and a globally fixed window size, with these design choices determined by computational considerations and empirical validation. Although this design proves effective, future studies may explore adaptive mechanisms, such as learnable k-mer granularities and context-dependent window selection, to further improve the flexibility and expressiveness of motif memory. In addition, the benefits of Gengram are task-dependent: its motif-oriented inductive bias appears particularly effective for motif-rich sequence tasks, while some tasks centered on fine-grained single-nucleotide effects show smaller or less consistent gains. Finally, while our results suggest that Gengram makes motif-level sequence information more explicitly accessible and conditionally usable, its role in broader multi-scale biological foundation models remains to be further investigated. In particular, integrating such sequence-level memory mechanisms with models of gene regulation, cell states, and perturbation responses may provide a promising direction toward virtual-cell (Xiao et al., 2026) systems that require interpretable memory links between genotype, regulatory programs, and cellular phenotypes.

## Acknowledgments

This work was supported by the Key Research Project of Zhejiang Lab (2025SSYS0004), the Key Research Project of Zhejiang Lab (2026SSYS0002) and the National Key R&D Program of China (2021YFA0805100).

## References

- Joshua Ainslie, James Lee-Thorp, Michiel De Jong, Yury Zemlyanskiy, Federico Lebrón, and Sumit Sanghai. 2023. [Gqa: Training generalized multi-query transformer models from multi-head checkpoints](#). *arXiv preprint arXiv:2305.13245*.
- Žiga Avsec, Vikram Agarwal, Daniel Visentin, Joseph R Ledsam, Agnieszka Grabska-Barwinska, Kyle R Taylor, Yannis Assael, John Jumper, Pushmeet Kohli, and David R Kelley. 2021. [Effective gene expression prediction from sequence by integrating long-range interactions](#). *Nature methods*, 18(10):1196–1203.
- Sam Boshar, Benjamin Evans, Ziqi Tang, Armand Picard, Yanis Adel, Franziska K Lorbeer, Chandana Rajesh, Tristan Karch, Shawn Sidbon, David Emms, and 1 others. 2025. [A foundational model for joint sequence-function multi-species modeling at scale for long-range genomic prediction](#). *bioRxiv*, pages 2025–12.
- Garyk Brixi, Matthew G Durrant, Jerome Ku, Mohsen Naghipourfar, Michael Poli, Gwanggyu Sun, Greg Brockman, Daniel Chang, Alison Fanton, Gabriel A Gonzalez, and 1 others. 2026. [Genome modelling and design across all domains of life with evo 2](#). *Nature*, 652(8112):1349–1361.
- Anzhe Cheng, Shukai Duan, Shixuan Li, Chenzhong Yin, Mingxi Cheng, Heng Ping, Tamoghna Chattopadhyay, Sophia I Thomopoulos, Shahin Nazarian, Paul Thompson, and 1 others. 2026a. [Ermoe: Eigen-reparameterized mixture-of-experts for stable routing and interpretable specialization](#). In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12997–13006.
- Wenduo Cheng, Zhenqiao Song, Yang Zhang, Shike Wang, Danqing Wang, Muyu Yang, Lei Li, and Jian Ma. 2025. [Dnalongbench: a benchmark suite for long-range dna prediction tasks](#). *Nature communications*, 16(1):10108.
- Xin Cheng, Wangding Zeng, Damai Dai, Qinyu Chen, Bingxuan Wang, Zhenda Xie, Kezhao Huang, Xingkai Yu, Zhewen Hao, Han Zhang, Yu-Kun Li, Huishuai Zhang, Dongyan Zhao, and Wenfeng Liang. 2026b. [Conditional memory via scalable lookup: A new axis of sparsity for large language models](#). In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4968–4990.
- Francis HC Crick. 1968. [The origin of the genetic code](#). *Journal of molecular biology*, 38(3):367–379.
- Hugo Dalla-Torre, Liam Gonzalez, Javier Mendoza-Revilla, Nicolas Lopez Carranza, Adam Henryk Grzywaczewski, Francesco Oteri, Christian Dallago, Evan Trop, Bernardo P de Almeida, Hassan Sirelkhatim, and 1 others. 2025. [Nucleotide transformer: building and evaluating robust foundation models for human genomics](#). *Nature Methods*, 22(2):287–297.
- Tamara Goldfarb, Vamsi K Kodali, Shashikant Pujar, Vyacheslav Brover, Barbara Robbertse, Catherine M Farrell, Dong-Ha Oh, Alexander Astashyn, Olga Ermolaeva, Diana Haddad, and 1 others. 2025. [Ncbi refseq: reference sequence standards through 25 years of curation and annotation](#). *Nucleic Acids Research*, 53(D1):D243–D257.
- Katarína Grešová, Vlastimil Martinek, David Čechák, Petr Šimeček, and Panagiotis Alexiou. 2023. [Genomic benchmarks: a collection of datasets for genomic sequence classification](#). *BMC Genomic Data*, 24(1):25.
- Albert Gu and Tri Dao. 2023. [Mamba: Linear-time sequence modeling with selective state spaces](#). *arXiv preprint arXiv:2312.00752*.
- Bernhard Haubold and Thomas Wiehe. 2006. [How repetitive are genomes?](#) *BMC bioinformatics*, 7(1):541.
- Kishore Jaganathan, Sofia Kyriazopoulou Panagiotopoulou, Jeremy F McRae, Siavash Fazel Darbandi, David Knowles, Yang I Li, Jack A Kosmicki, Juan Arbelaez, Wenwu Cui, Grace B Schwartz, and 1 others. 2019. [Predicting splicing from primary sequence with deep learning](#). *Cell*, 176(3):535–548.
- Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. 2019. [What does bert learn about the structure of language?](#) In *ACL 2019-57th Annual Meeting of the Association for Computational Linguistics*.
- Yanrong Ji, Zhihan Zhou, Han Liu, and Ramana V Davuluri. 2021. [Dnabert: pre-trained bidirectional encoder representations from transformers model for dna-language in genome](#). *Bioinformatics*, 37(15):2112–2120.
- Kristin D Kernohan and Kym M Boycott. 2024. [The expanding diagnostic toolbox for rare genetic diseases](#). *Nature Reviews Genetics*, 25(6):401–415.
- Wentian Li, Jan Freudenberg, and Pedro Miramontes. 2014. [Diminishing return for increased mappability with longer sequencing reads: implications of the k-mer distributions in the human genome](#). *BMC bioinformatics*, 15(1):2.
- Wen-Wei Liao, Mobin Asri, Jana Ebler, Daniel Doerr, Marina Haukness, Glenn Hickey, Shuangjia Lu, Julian K Lucas, Jean Monlong, Haley J Abel, and 1 others. 2023. [A draft human pangenome reference](#). *Nature*, 617(7960):312–324.
- Adi Lin, Bin Xie, Cheng Ye, Cheng Wang, Duoyuan Chen, Ercheng Wang, Fanfeng Lu, Guirong Xue, Haiqiang Zhang, Jiajie Zhan, and 1 others. 2025. [Genos: a human-centric genomic foundation model](#). *GigaScience*, 14:giaf132.
- Aixin Liu, Bei Feng, Bin Wang, Bingxuan Wang, Bo Liu, Chenggang Zhao, Chengqi Deng, Chong Ruan, Damai Dai, Daya Guo, and 1 others. 2024. [Deepseek-v2: A strong, economical, and efficient](#)

- mixture-of-experts language model. *arXiv preprint arXiv:2405.04434*.
- Daniel S Margulies, Satrajit S Ghosh, Alexandros Goulas, Marcel Falkiewicz, Julia M Huntentburg, Georg Langs, Gleb Bezgin, Simon B Eickhoff, F Xavier Castellanos, Michael Petrides, and 1 others. 2016. [Situating the default-mode network along a principal gradient of macroscale cortical organization](#). *Proceedings of the National Academy of Sciences*, 113(44):12574–12579.
- Edgar Mariano Marroquin, Yair Schiff, Evan Trop, Chia Hsiang Kao, Aaron Gokaslan, McKinley Polen, Aymen Kallala, Mingyi Shao, Bernardo P de Almeida, Thomas Pierrot, Yang I Li, and Volodymyr Kuleshov. 2026. [The human genomics long-range benchmark: Advancing DNA language models](#).
- Stephen M Mount. 1982. [A catalogue of splice junction sequences](#). *Nucleic acids research*, 10(2):459–472.
- Eric Nguyen, Michael Poli, Marjan Faizi, Armin Thomas, Michael Wornow, Callum Birch-Sykes, Stefano Massaroli, Aman Patel, Clayton Rabideau, Yoshua Bengio, and 1 others. 2023. [Hyenadna: Long-range genomic sequence modeling at single nucleotide resolution](#). *Advances in neural information processing systems*, 36:43177–43201.
- Nostalgebraist. 2020. [Interpreting gpt: The logit lens](#). Accessed: 2026-01-29.
- Zihan Qiu, Zekun Wang, Bo Zheng, Zeyu Huang, Kaiyue Wen, Songlin Yang, Rui Men, Le Yu, Fei Huang, Suozhi Huang, and 1 others. 2026. [Gated attention for large language models: Non-linearity, sparsity, and attention-sink-free](#). *Advances in Neural Information Processing Systems*, 38:100092–100118.
- Yair Schiff, Chia-Hsiang Kao, Aaron Gokaslan, Tri Dao, Albert Gu, and Volodymyr Kuleshov. 2024. [Caduceus: Bi-directional equivariant long-range dna sequence modeling](#). *Proceedings of machine learning research*, 235:43632.
- Oscar Skea, Md Rifat Arefin, Dan Zhao, Niket Patel, Jalal Naghiyev, Yann LeCun, and Ravid Shwartz-Ziv. 2025. [Layer by layer: Uncovering hidden representations in language models](#). *arXiv preprint arXiv:2502.02013*.
- Stephen T Smale and James T Kadonaga. 2003. [The rna polymerase ii core promoter](#). *Annual review of biochemistry*, 72(1):449–479.
- Alexandros Tzanakakis, Ioannis Mouratidis, and Ilias Georgakopoulos-Soares. 2026. [Fundamental limitations of genomic language models for realistic sequence generation](#). *bioRxiv*, pages 2026–01.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. [Attention is all you need](#). *Advances in neural information processing systems*, 30.
- Jesse Vig and Yonatan Belinkov. 2019. [Analyzing the structure of attention in a transformer language model](#). *arXiv preprint arXiv:1906.04284*.
- James C Wang. 1979. [Helical repeat of dna in solution](#). *Proceedings of the National Academy of Sciences*, 76(1):200–203.
- Wei Wu, Qiuyi Li, Yuanyuan Zhang, Zhihao Zhan, Ruipu Chen, Mingyang Li, Kun Fu, Junyan Qi, Yongzhou Bao, Chao Wang, and 1 others. 2025. [Generator: a long-context generative genomic foundation model](#). *arXiv preprint arXiv:2502.07272*.
- Zhan Xiao, Wuke Wang, Xin Long, Wenbo Zhang, Weiqiang Zhang, Duoyuan Chen, Sujie Xu, Qian Yu, Xinpeng Zhang, Shichen Huang, Ning Zhang, Yanbin Yin, Xingxu Huang, Jieping Ye, Jinfang Zheng, and Ling Guo. 2026. [scdiffformer: diffusion-based post-training for virtual cell modeling across large-scale single-cell data](#). *Nucleic Acids Research*, 54(13):gkag706.
- Zhao Yang, Bing Su, Chuan Cao, and Ji-Rong Wen. 2025. [Regulatory dna sequence design with reinforcement learning](#). In *International Conference on Learning Representations*, volume 2025, pages 73499–73524.
- Barret Zoph, Irwan Bello, Sameer Kumar, Nan Du, Yanping Huang, Jeff Dean, Noam Shazeer, and William Fedus. 2022. [St-moe: Designing stable and transferable sparse expert models](#). *arXiv preprint arXiv:2202.08906*.

## Appendix

Table 1: **Computational Efficiency Comparison:** TFLOP/s/GPU and per-iteration time for the baseline model and Gengram variants.

| ARCHITECTURE                 | PARAMETERS | TFLOP/s/GPU | PER-ITERATION TIME (s) |
|------------------------------|------------|-------------|------------------------|
| BASILINE                     | 1.2B       | 166.6       | 1.040                  |
| GENGRAM-LAYER10              | 1.2B       | 164.0       | 1.063                  |
| GENGRAM-LAYER3-6-10          | 1.2B       | 161.2       | 1.090                  |
| GENGRAM-LAYER3-6-10-WINDOW21 | 1.2B       | 159.1       | 1.098                  |

Table 2: **MoE-1.2B Model Configurations For Layer and Window**

| PARAMETER                   | 1.2B-SELECTED-LAYERS | 1.2B-SELECTED-WINDOWS                                                     |
|-----------------------------|----------------------|---------------------------------------------------------------------------|
| TOTAL PARAMETERS            | 1.2B                 |                                                                           |
| ACTIVATED PARAMETERS        | 0.3B                 |                                                                           |
| HIDDEN SIZE                 | 1024                 |                                                                           |
| NUM LAYERS                  | 12                   |                                                                           |
| ATTENTION HEADS             | 16                   |                                                                           |
| GQA QUERY GROUPS            | 8                    |                                                                           |
| NORMALIZATION               | RMSNORM              |                                                                           |
| ACTIVATION                  | SWIGLU               |                                                                           |
| POSITIONAL EMB              | ROPE                 |                                                                           |
| NUM EXPERTS                 | 8                    |                                                                           |
| ROUTER TOP-K                | 2                    |                                                                           |
| MOE OPERATORS               | GROUPED GEMM         |                                                                           |
| LOAD BALANCING              | AUX LOSS             |                                                                           |
| GENGRAM LAYERS              | $L \in [1, 12]$      | 10                                                                        |
| GENGRAM NGRAM-SIZES         | 1, 2, 3, 4, 5, 6     |                                                                           |
| GENGRAM WINDOW-SIZES        | $\backslash$         | $S \in [6, 8, 9, 10, 12, 14, 15, 16, 18, 20, 21, 22, 24, 26, 27, 28, 30]$ |
| GENGRAM CONV-KERNEL-SIZE    | 4                    |                                                                           |
| GENGRAM EMBED-DIM-PER-NGRAM | 1024                 |                                                                           |
| SEQ LENGTH                  | 8192                 |                                                                           |
| GLOBAL BATCH SIZE           | 1024                 |                                                                           |
| LEARNING RATE               | 1E-4 TO 1E-5         |                                                                           |
| LR SCHEDULER                | COSINE DECAY         |                                                                           |

---

**Algorithm 1** Sliding window search

---

Retrieve at Position  $t$ **Input:** position  $t$ **Output:**  $\{m_t^{(N)}\}_{N \in \mathcal{N}}$ **for**  $N \in \mathcal{N}$  **do** $j_{start} \leftarrow \max(1, t - W)$  $j_{end} \leftarrow t - N$  $\mathcal{K} \leftarrow \emptyset$  $sum \leftarrow \mathbf{0}_{d_m}$  $cnt \leftarrow 0$ **for**  $j = j_{start}$  **to**  $j_{end}$  **do** $key \leftarrow h_N(s[j \dots j + N - 1])$ **if**  $key \notin \mathcal{K}$  **then** $\mathcal{K} \leftarrow \mathcal{K} \cup \{key\}$  $sum \leftarrow sum + M^{(N)}[key]$  $cnt \leftarrow cnt + 1$ **end if****end for** $m_t^{(N)} \leftarrow sum / \max(cnt, 1)$ **end for****return**  $\{m_t^{(N)}\}_{N \in \mathcal{N}}$ 

---

*unique N-gram keys  
initialize accumulator  
count valid N-grams**build N-gram key* *$O(1)$  lookup**mean over unique N-gram keys*Table 3: **K-mer Ablation Study of GENGRAM-LAYER3-6-10:** Performance comparison using different k-mer ranges. The best result for each dataset is highlighted in bold.

| Dataset                                    | $k = 1-6$     | $k = 1-3$     | $k = 4-6$ |
|--------------------------------------------|---------------|---------------|-----------|
| demo_coding_vs_intergenomic_seqs           | 0.9780        | <b>0.9788</b> | 0.9776    |
| human_enhancers_cohn                       | <b>0.8366</b> | 0.8365        | 0.8337    |
| human_ocr_ensembl                          | <b>0.7783</b> | 0.7744        | 0.7730    |
| splice_sites_all                           | <b>0.7681</b> | 0.7641        | 0.7480    |
| H3                                         | 0.9106        | <b>0.9150</b> | 0.9064    |
| H3K36me3                                   | <b>0.7609</b> | 0.7605        | 0.7539    |
| CPC_8192                                   | <b>0.9467</b> | 0.9365        | 0.9376    |
| regulatory_element_promoter_8K             | 0.9260        | <b>0.9261</b> | 0.9234    |
| regulatory_element_enhancer_8K             | 0.7473        | <b>0.7481</b> | 0.7471    |
| variant_effect_causal_eqtl_8K              | <b>0.6999</b> | 0.6916        | 0.6959    |
| primate_mammal_species_classification_8192 | 0.9072        | <b>0.9078</b> | 0.9067    |

Table 4: **Downstream Evaluation of Multi-Layer:** Average performance across five downstream categories for single-layer vs. multi-layer insertions. 3, 6, 10 performs best overall, ranking first in four of five categories.

| SELECTED LAYERS                  | 10            | 10,11,12 | 9,10,11,12 | 3,6,10        | 3,6    | 3,10   |
|----------------------------------|---------------|----------|------------|---------------|--------|--------|
| GENOMIC STRUCTURE UNDERSTANDING  | <b>0.8944</b> | 0.8866   | 0.8900     | 0.8937        | 0.8838 | 0.8883 |
| GENE REGULATION PREDICTION       | 0.8206        | 0.8154   | 0.8188     | <b>0.8222</b> | 0.8181 | 0.8196 |
| EPIGENETIC PROFILING             | <b>0.8334</b> | 0.8329   | 0.8333     | <b>0.8334</b> | 0.8276 | 0.8306 |
| VARIANT EFFECT & CLINICAL IMPACT | 0.7577        | 0.7553   | 0.7489     | <b>0.7678</b> | 0.7602 | 0.7649 |
| EVOLUTIONARY ANALYSIS            | 0.9101        | 0.9049   | 0.9059     | <b>0.9117</b> | 0.9093 | 0.9064 |

Table 5: **Component ablation results on the validation set.** The Gengram baseline uses mean pooling with the gating mechanism. All reported values are validation losses, where lower is better. All models were trained using the training setup of the 1.2B-parameter model (Appx. Table 2).

| ABLATION RESULTS                       | VALIDATION LOSS |
|----------------------------------------|-----------------|
| GENGRAM BASELINE (MEAN POOLING + GATE) | 1.0177          |
| MAX POOLING + GATE                     | 1.0217          |
| ATTENTION POOLING + GATE               | 1.0221          |
| MEAN POOLING WITHOUT GATE              | 1.0487          |

Table 6: **CPT Adaptation Results for Models With and Without Local Window.** (GENGRAM-10B-8k\* and GENGRAM-10B-32k\* denote the windowed variants GENGRAM-10B-L3\_6\_10-W21-8k and GENGRAM-10B-L3\_6\_10-W21-32k, respectively; GENGRAM-10B-L3\_6\_10-8k and GENGRAM-10B-L3\_6\_10-32k denote the corresponding non-windowed variants.)

|                                          |                                       | GENGRAM-<br>10B-<br>L3_6_10-8k | GENGRAM-<br>10B-8k* | GENGRAM-<br>10B-<br>L3_6_10-32k | GENGRAM-<br>10B-32k* |
|------------------------------------------|---------------------------------------|--------------------------------|---------------------|---------------------------------|----------------------|
| GENOMIC<br>STRUCTURE<br>UNDERSTANDING    | DEMO_CODING_VS_<br>INTERGENOMIC_SEQS  | 0.9412                         | 0.9733              | 0.9818                          | 0.9776               |
|                                          | SPLICE_SITES_ALL                      | 0.8453                         | 0.9009              | 0.8465                          | 0.8858               |
|                                          | HUMAN_<br>GENOMIC_ELEMENT             | 0.8842                         | 0.8982              | 0.8911                          | 0.8998               |
|                                          | MOUSE_<br>GENOMIC_ELEMENT             | 0.9603                         | 0.9495              | 0.9422                          | 0.9722               |
|                                          | MULTI_SPECIES_<br>EXON_CLASSIFICATION | 0.8903                         | 0.9832              | 0.9794                          | 0.9858               |
|                                          | MULTI_SPECIES_<br>THREE_PRIME_UTR     | 0.8759                         | 0.9601              | 0.9603                          | 0.9598               |
| GENE REGULATION<br>PREDICTION            | MULTI_SPECIES_<br>FIVE_PRIME_UTR      | 0.8397                         | 0.9443              | 0.9521                          | 0.9508               |
|                                          | REGULATORY_ELEMENT_<br>PROMOTER_8K    | 0.8861                         | 0.9074              | 0.9121                          | 0.9097               |
|                                          | REGULATORY_ELEMENT_<br>ENHANCER_8K    | 0.7054                         | 0.7401              | 0.7267                          | 0.7065               |
|                                          | HUMAN_ENHANCERS_COHN                  | 0.7605                         | 0.8358              | 0.8513                          | 0.8375               |
|                                          | HUMAN_OCR_ENSEMBL                     | 0.7216                         | 0.7714              | 0.7785                          | 0.7663               |
| VARIANT EFFECT<br>AND<br>CLINICAL IMPACT | VARIANT_EFFECT_<br>CAUSAL_EQTL_8K     | 0.6078                         | 0.6518              | 0.6189                          | 0.6607               |
|                                          | CPC_8192                              | 0.8875                         | 0.9219              | 0.9269                          | 0.9283               |
|                                          | CPC_32768                             | 0.864                          | 0.952               | 0.9606                          | 0.9572               |
| EPIGENETIC<br>PROFILING                  | H3                                    | 0.8934                         | 0.9434              | 0.941                           | 0.9417               |
|                                          | H3K36ME3                              | 0.7227                         | 0.8039              | 0.8045                          | 0.8161               |
| EVOLUTIONARY<br>ANALYSIS                 | PRIMATE_MAMMAL_<br>SPECIES_8192       | 0.7275                         | 0.9611              | 0.9676                          | 0.9691               |
|                                          | PRIMATE_MAMMAL_<br>SPECIES_32768      | 0.6933                         | 0.963               | 0.9785                          | 0.9793               |

Table 7: **Comparison with a 32k-context adapted baseline.** GENGRAM-10B-8k\* is compared with Baseline-10B-32k, which is obtained by continued pretraining of Baseline-10B-8k at 32k context length. This comparison evaluates whether the 200B-token, 8k-context Gengram model remains competitive with a baseline adapted to longer context, rather than serving as a fully matched architectural ablation. (GENGRAM-10B-8k\* denotes the GENGRAM-10B-L3\_6\_10-W21-8k model for short.)

| TASK                                        | BASELINE-10B-32K | GENGRAM-10B-8K* |
|---------------------------------------------|------------------|-----------------|
| DEMO_CODING_VS_INTERGENOMIC_SEQS            | 0.9717           | <b>0.9733</b>   |
| HUMAN_ENHANCERS_COHN                        | 0.8351           | <b>0.8358</b>   |
| HUMAN_OCR_ENSEMBL                           | 0.7301           | <b>0.7714</b>   |
| SPLICE_SITES_ALL                            | 0.7809           | <b>0.9009</b>   |
| H3                                          | 0.8820           | <b>0.9434</b>   |
| H3K36ME3                                    | 0.6556           | <b>0.8039</b>   |
| CPC_8192                                    | <b>0.9397</b>    | 0.9219          |
| CPC_32768                                   | <b>0.9533</b>    | 0.9520          |
| REGULATORY_ELEMENT_PROMOTER_8K              | 0.9020           | <b>0.9074</b>   |
| REGULATORY_ELEMENT_ENHANCER_8K              | 0.7177           | <b>0.7401</b>   |
| VARIANT_EFFECT_CAUSAL_EQTL_8K               | 0.6011           | <b>0.6518</b>   |
| PRIMATE_MAMMAL_SPECIES_CLASSIFICATION_8192  | <b>0.9640</b>    | 0.9611          |
| PRIMATE_MAMMAL_SPECIES_CLASSIFICATION_32768 | <b>0.9783</b>    | 0.9630          |
| HUMAN_GENOMIC_ELEMENT                       | 0.8822           | <b>0.8982</b>   |
| MOUSE_GENOMIC_ELEMENT                       | 0.8756           | <b>0.9495</b>   |
| MULTI_SPECIES_EXON_CLASSIFICATION           | 0.9469           | <b>0.9832</b>   |
| MULTI_SPECIES_THREE_PRIME_UTR               | 0.9151           | <b>0.9601</b>   |
| MULTI_SPECIES_FIVE_PRIME_UTR                | 0.8933           | <b>0.9443</b>   |

Table 8: **Comparison of various GFMs:** (GENGRAM-10B-8k\* and GENGRAM-10B-32k\* denote GENGRAM-10B-L3\_6\_10-W21-8k and GENGRAM-10B-L3\_6\_10-W21-32k for short.)

| MODEL             | TOTAL<br>PARAMETERS | ACTIVATED<br>PARAMETERS | TRAINED<br>TOKENS | SEQ.<br>LENGTH | TOKENIZER   | DATASET                              |
|-------------------|---------------------|-------------------------|-------------------|----------------|-------------|--------------------------------------|
| GENGRAM-10B-8K*   | 10B                 | 2.87B                   | 200B              | 8K             | SINGLE BASE | HPRC,<br>REFSEQ DATABASE             |
| GENGRAM-10B-32K*  | 10B                 | 2.87B                   | 200B+100B         | 32K            | SINGLE BASE | HPRC,<br>REFSEQ DATABASE             |
| GENOS-10B         | 10B                 | 2.87B                   | 2.2T              | 1M             | SINGLE BASE | HPRC,HGSVC,<br>CEPH,GRCH38,<br>CHM13 |
| Evo2-40B          | 40.3B               | 40.3B                   | 9.3T              | 1M             | SINGLE BASE | OPENGENOME2                          |
| NTv3-<br>PRETRAIN | 650M                | 650M                    | 11T               | 1M             | SINGLE BASE | OPENGENOME2                          |
| GENERATOR-<br>3B  | 3B                  | 3B                      | 386B              | 98K            | 6-MER       | REFSEQ EUKARYOTIC<br>GENOMES         |

Table 9: **Cross-model benchmarking results.** Comparison of GENGRAM-10B-8k\* with externally trained GFMs. Models differ in training data, token budget, context length, architecture, and parameterization; therefore, this comparison is intended to contextualize empirical competitiveness rather than provide a controlled estimate of the effect of Gengram. The first place is in bold and the second place is in italics. (GENGRAM-10B-8k\* denotes GENGRAM-10B-L3\_6\_10-W21-8k.)

|                                          | MODEL                                       | GENGRAM-10B-8k*     | GENOS-10B     | Evo2-40B        | GENERATOR-3B    | NT-v3-PRETRAIN (650M) |
|------------------------------------------|---------------------------------------------|---------------------|---------------|-----------------|-----------------|-----------------------|
|                                          | TRAINED TOKENS                              | <b>200B</b><br>(1×) | 2.2T<br>(11×) | 9.3T<br>(46.5×) | 386B<br>(1.93×) | 11T<br>(55×)          |
| GENOMIC<br>STRUCTURE<br>UNDERSTANDING    | DEMO_CODING_VS_INTERGENOMIC_SEQS            | 0.9733              | <b>0.9914</b> | <i>0.9886</i>   | 0.9855          | 0.983                 |
|                                          | SPLICE_SITES_ALL_HUMAN_GENOMIC_ELEMENT      | <i>0.9009</i>       | 0.7990        | <b>0.9138</b>   | 0.8071          | 0.8644                |
|                                          | MOUSE_GENOMIC_ELEMENT                       | 0.8982              | 0.8890        | 0.9103          | <i>0.9173</i>   | <b>0.9189</b>         |
|                                          | MULTI_SPECIES_EXON_CLASSIFICATION           | 0.9495              | <i>0.9521</i> | <b>0.9888</b>   | 0.9015          | 0.9092                |
|                                          | MULTI_SPECIES_THREE_PRIME_UTR               | <b>0.9832</b>       | <i>0.9755</i> | 0.9332          | 0.9455          | 0.9615                |
|                                          | MULTI_SPECIES_FIVE_PRIME_UTR                | <i>0.9601</i>       | 0.9511        | <b>0.9634</b>   | 0.9400          | 0.9408                |
|                                          |                                             | <i>0.9443</i>       | 0.9317        | <b>0.9895</b>   | 0.8841          | 0.9084                |
|                                          | REGULATORY_ELEMENT_PROMOTER_8K              | 0.9074              | <b>0.9249</b> | <i>0.9227</i>   | 0.9195          | 0.922                 |
|                                          | REGULATORY_ELEMENT_ENHANCER_8K              | 0.7401              | <b>0.7532</b> | <i>0.7527</i>   | 0.7390          | 0.7415                |
| GENE REGULATION<br>PREDICTION            | HUMAN_ENHANCERS_COHN                        | <i>0.8358</i>       | <b>0.8552</b> | 0.7756          | 0.8181          | 0.8184                |
|                                          | HUMAN_OCR_ENSEMBL                           | <b>0.7714</b>       | 0.7623        | <i>0.7635</i>   | 0.7270          | 0.7426                |
|                                          | VARIANT_EFFECT_CAUSAL_EQTL_8K               | 0.6518              | 0.6773        | <b>0.7054</b>   | <i>0.6920</i>   | 0.687                 |
|                                          | CPC_8192                                    | 0.9219              | <b>0.9522</b> | 0.9401          | 0.9315          | <i>0.9468</i>         |
| VARIANT EFFECT<br>AND<br>CLINICAL IMPACT | CPC_32768                                   | <i>0.952</i>        | <b>0.9625</b> | <i>0.9611</i>   | 0.9237          | 0.9426                |
|                                          |                                             |                     |               |                 |                 |                       |
| EPIGENETIC<br>PROFILING                  | H3                                          | <i>0.9434</i>       | 0.9400        | 0.9311          | 0.9163          | <b>0.9505</b>         |
|                                          | H3K36ME3                                    | 0.8039              | 0.7658        | <b>0.8823</b>   | 0.8247          | <i>0.8535</i>         |
| EVOLUTIONARY<br>ANALYSIS                 | PRIMATE_MAMMAL_SPECIES_CLASSIFICATION_8192  | <i>0.9611</i>       | <b>0.9739</b> | 0.8867          | 0.8424          | 0.8652                |
|                                          | PRIMATE_MAMMAL_SPECIES_CLASSIFICATION_32768 | <i>0.9630</i>       | <b>0.9804</b> | 0.9101          | 0.8709          | 0.8928                |

Table 10: **Controlled CPT comparison on the well-trained Genos-10B backbone.** The Genos-10B-CPT and Genos-10B-CPT-Gengram variants are initialized from the same Genos-10B checkpoint and undergo the same 200B-token continued pretraining; their comparison therefore isolates the effect of introducing Gengram under a matched CPT setting. Evo2-7B results are included only as external reference. The first place is in bold and the second place is in italics.

|                                    | MODEL                                | GENOS-10B-CPT-GENGRAM | EVO2-7B BASE 8K | EVO2-7B 1M    | GENOS-10B-CPT |
|------------------------------------|--------------------------------------|-----------------------|-----------------|---------------|---------------|
|                                    | TRAINED TOKENS                       | <b>2.4T</b><br>(1×)   | 2.4T<br>(1×)    | 2.4T<br>(1×)  | 2.4T<br>(1×)  |
| GENOMIC STRUCTURE UNDERSTANDING    | DEMO_CODING_VS_INTERGENOMIC_SEQS     | <b>0.9930</b>         | 0.9807          | 0.9824        | <i>0.9928</i> |
|                                    | SPLICE_SITES_ALL                     | 0.8080                | <i>0.8677</i>   | <b>0.8747</b> | 0.7977        |
| GENE REGULATION PREDICTION         | REGULATORY_ELEMENT_PROMOTER_8K       | <b>0.9279</b>         | 0.9251          | <i>0.9255</i> | 0.9245        |
|                                    | REGULATORY_ELEMENT_ENHANCER_8K       | <b>0.7534</b>         | 0.7465          | 0.7454        | <i>0.7515</i> |
|                                    | HUMAN_ENHANCERS_COHN                 | <b>0.8616</b>         | 0.7684          | 0.7733        | <i>0.8286</i> |
|                                    | HUMAN_OCR_ENSEMBL                    | <b>0.7593</b>         | <i>0.7588</i>   | 0.7505        | 0.7507        |
| VARIANT EFFECT AND CLINICAL IMPACT | VARIANT_EFFECT_CAUSAL_EQTL_8K        | 0.7027                | <i>0.7031</i>   | <b>0.7039</b> | 0.6742        |
|                                    | VARIANT_EFFECT_PATHOGENIC_CLINVAR_8K | <b>0.9369</b>         | 0.7234          | 0.7308        | <i>0.9353</i> |
|                                    | CPC_8192                             | <i>0.9506</i>         | 0.9400          | 0.9425        | <b>0.9518</b> |
|                                    | CPC_32768                            | <b>0.9702</b>         | /               | 0.9504        | <i>0.9679</i> |
| EPIGENETIC PROFILING               | H3                                   | <b>0.9342</b>         | 0.9167          | 0.9140        | <i>0.9275</i> |
|                                    | H3K36ME3                             | 0.7895                | <i>0.8603</i>   | <b>0.8615</b> | 0.7815        |

Table 11: **MHA-based Gengram Configurations (Dense vs. MoE)**

| PARAMETER                   | 0.3B DENSE (BASE/GENGRAM) | 1.2B MoE (BASE/GENGRAM) |
|-----------------------------|---------------------------|-------------------------|
| TOTAL PARAMETERS            | 0.3B                      | 1.2B                    |
| ACTIVATED PARAMETERS        | 0.3B                      |                         |
| HIDDEN SIZE                 | 1280                      | 1024                    |
| NUM LAYERS                  | 12                        |                         |
| ATTENTION HEADS             | 16                        |                         |
| NORMALIZATION               | RMSNORM                   |                         |
| ACTIVATION                  | SWIGLU                    |                         |
| POSITIONAL EMB              | ROPE                      |                         |
| NUM EXPERTS                 | \                         | 8                       |
| ROUTER TOP-K                | \                         | 2                       |
| MoE OPERATORS               | \                         | GROUPED GEMM            |
| LOAD BALANCING              | \                         | AUX LOSS                |
| GENGRAM LAYERS              | OPTIONAL {3, 6, 10}       |                         |
| GENGRAM NGRAM-SIZES         | OPTIONAL (1 TO 6)         |                         |
| GENGRAM WINDOW-SIZES        | OPTIONAL (21)             |                         |
| GENGRAM CONV-KERNEL-SIZE    | OPTIONAL (4)              |                         |
| GENGRAM EMBED-DIM-PER-NGRAM | OPTIONAL (1024)           |                         |
| SEQ LENGTH                  | 8192                      |                         |
| GLOBAL BATCH SIZE           | 1024                      |                         |
| LEARNING RATE               | 1E-4 TO 1E-5              |                         |
| LR SCHEDULER                | COSINE DECAY              |                         |

Table 12: **MLA-based Gengram Configurations (Dense vs. MoE)**

| PARAMETER                   | 0.3B DENSE (BASE/GENGRAM) | 1.2B MoE (BASE/GENGRAM) |
|-----------------------------|---------------------------|-------------------------|
| TOTAL PARAMETERS            | 0.3B                      | 1.2B                    |
| ACTIVATED PARAMETERS        | 0.3B                      |                         |
| HIDDEN SIZE                 | 1280                      | 1024                    |
| NUM LAYERS                  | 12                        |                         |
| ATTENTION HEADS             | 16                        |                         |
| KV-LoRA RANK                | 512                       | 256                     |
| QK HEAD DIM                 | 128                       | 64                      |
| QK POS-EMB DIM              | 128                       | 64                      |
| V HEAD DIM                  | 256                       | 128                     |
| NORMALIZATION               | RMSNORM                   |                         |
| ACTIVATION                  | SwiGLU                    |                         |
| POSITIONAL EMB              | RoPE                      |                         |
| NUM EXPERTS                 | \                         | 8                       |
| ROUTER TOP-K                | \                         | 2                       |
| MoE OPERATORS               | \                         | GROUPED GEMM            |
| LOAD BALANCING              | \                         | AUX LOSS                |
| GENGRAM LAYERS              | OPTIONAL {3, 6, 10}       |                         |
| GENGRAM NGRAM-SIZES         | OPTIONAL (1 TO 6)         |                         |
| GENGRAM WINDOW-SIZES        | OPTIONAL (21)             |                         |
| GENGRAM CONV-KERNEL-SIZE    | OPTIONAL (4)              |                         |
| GENGRAM EMBED-DIM-PER-NGRAM | OPTIONAL (1024)           |                         |
| SEQ LENGTH                  | 8192                      |                         |
| GLOBAL BATCH SIZE           | 1024                      |                         |
| LEARNING RATE               | 1E-4 TO 1E-5              |                         |
| LR SCHEDULER                | COSINE DECAY              |                         |

Table 13: **GQA-based Gengram Configurations (Dense vs. MoE)**

| PARAMETER                   | 0.3B DENSE (BASE/GENGRAM) | 1.2B MoE (BASE/GENGRAM) |
|-----------------------------|---------------------------|-------------------------|
| TOTAL PARAMETERS            | 0.3B                      | 1.2B                    |
| ACTIVATED PARAMETERS        | 0.3B                      |                         |
| HIDDEN SIZE                 | 1280                      | 1024                    |
| NUM LAYERS                  | 12                        |                         |
| ATTENTION HEADS             | 16                        |                         |
| GQA QUERY GROUPS            | 8                         |                         |
| NORMALIZATION               | RMSNORM                   |                         |
| ACTIVATION                  | SwiGLU                    |                         |
| POSITIONAL EMB              | RoPE                      |                         |
| NUM EXPERTS                 | \                         | 8                       |
| ROUTER TOP-K                | \                         | 2                       |
| MoE OPERATORS               | \                         | GROUPED GEMM            |
| LOAD BALANCING              | \                         | AUX LOSS                |
| GENGRAM LAYERS              | OPTIONAL {3, 6, 10}       |                         |
| GENGRAM NGRAM-SIZES         | OPTIONAL (1 TO 6)         |                         |
| GENGRAM WINDOW-SIZES        | OPTIONAL (21)             |                         |
| GENGRAM CONV-KERNEL-SIZE    | OPTIONAL (4)              |                         |
| GENGRAM EMBED-DIM-PER-NGRAM | OPTIONAL (1024)           |                         |
| SEQ LENGTH                  | 8192                      |                         |
| GLOBAL BATCH SIZE           | 1024                      |                         |
| LEARNING RATE               | 1E-4 TO 1E-5              |                         |
| LR SCHEDULER                | COSINE DECAY              |                         |

Table 14: Load Balancing Test

| PARAMETER                   | 10B MoE (GENGRAM) |         |        |
|-----------------------------|-------------------|---------|--------|
| TOTAL PARAMETERS            | 10B               |         |        |
| ACTIVATED PARAMETERS        | 0.15625B          | 0.3125B | 0.625B |
| HIDDEN SIZE                 | 1280              |         |        |
| NUM LAYERS                  | 12                |         |        |
| ATTENTION HEADS             | 16                |         |        |
| GQA QUERY GROUPS            | 8                 |         |        |
| NORMALIZATION               | RMSNORM           |         |        |
| ACTIVATION                  | SWIGLU            |         |        |
| POSITIONAL EMB              | RoPE              |         |        |
| NUM EXPERTS                 | 128               | 64      | 32     |
| ROUTER TOP-K                | 2                 |         |        |
| LOAD BALANCING              | AUX LOSS          |         |        |
| GENGRAM LAYERS              | {3, 6, 10}        |         |        |
| GENGRAM NGRAM-SIZES         | (1 TO 6)          |         |        |
| GENGRAM WINDOW-SIZES        | /                 |         |        |
| GENGRAM CONV-KERNEL-SIZE    | 4                 |         |        |
| GENGRAM EMBED-DIM-PER-NGRAM | 1024              |         |        |
| SEQ LENGTH                  | 8192              |         |        |
| GLOBAL BATCH SIZE           | 1024              |         |        |
| LEARNING RATE               | 1E-4 TO 1E-5      |         |        |
| LR SCHEDULER                | COSINE DECAY      |         |        |

## A Data Collection and Processing

### A.1 Pre-training Dataset

The pretrain dataset was curated from the Human Pangenome Reference Consortium (HPRC, release 2) (Liao et al., 2023) and NCBI RefSeq databases (Goldfarb et al., 2025), encompassing DNA sequences from humans and various primate species. The inclusion of non-human primate genomes provides evolutionarily informative signals while maintaining sufficient similarity to the human genome, thereby helping preserve the core sequence logic relevant to human genomic modeling. More distantly related species were not included in the pretraining dataset. Their greater sequence divergence may introduce patterns that are less relevant to human-centered genomic modeling and potentially dilute human-specific representations.

To support different stages of model development, we systematically constructed three datasets: (1) a 50B-token dataset with a sequence length of 8,192, designed for the ablation studies; (2) a 200B-token dataset with an 8k context length; and (3) a 100B-token dataset with a 32k context length, both designated for the formal pre-training of the 10B model. Across all datasets, a balanced 1:1 ratio between human and non-human sequences was strictly maintained to ensure taxonomic parity. Detailed lists of the datasets are provided in the Appendix Table 15,16 and 17.

### A.2 Benchmark Datasets

We evaluated Gengram on standard human-centric and multi-species benchmarks, including the Genomic Benchmarks (GB), the Nucleotide Transformer Benchmarks (NTB), the Long-Range Benchmarks (LRB), and the Genos Benchmarks (GeB). (Grešová et al., 2023; Dalla-Torre et al., 2025; Marroquin et al., 2026; Lin et al., 2025) From these sources, we selected 18 datasets spanning five major categories: Genomic Structure Understanding, Gene Regulation Prediction, Epigenetic Profiling, Variant Effect and Clinical Impact, and Evolutionary Analysis. For downstream evaluation, the backbone was frozen, and mean-pooled embeddings were used to train task-specific classifiers, including MLPs and XGBoost models. This frozen-backbone protocol was used to evaluate the quality of pretrained representations while minimizing the influence of task-specific optimization and hyperparameter choices. We followed the original benchmark construction, data splits, and evaluation protocols, without additional locus-level, chromosome-level, or homology-level decontamination between the pretraining corpus and downstream benchmark sequences. No downstream labels or split information were used during pretraining. Detailed descriptions of each dataset are provided in Appendix Table 18.

### A.3 Tokenization and Encoding

All sequences were processed using one-hot encoding. The vocabulary includes the four canonical bases (A, T, C, G), the ambiguous nucleotide N, and the end-of-document token <EOD>.

### A.4 Two-Stage Data Processing Pipeline

#### A.4.1 Stage I: Base Pre-training (8k Context)

In the first stage, Datasets 1 and 2 were processed into 8,192 bp fragments 15,16. This yielded a total of 250B tokens, with a 50B subset allocated for architectural ablation studies and the remaining 200B for formal pre-training. Sequences in this stage were restricted to the forward-strand orientation.

#### A.4.2 Stage II: Continued Pre-training (CPT, 32k Context)

In the CPT stage, Dataset 3 was formatted into 32,768 bp fragments 17. This stage introduced a dual-strand randomized assignment mechanism (56% negative strand and 44% positive strand) and removed the gene-distance filtering. Integrating distal intergenic regions and repetitive elements exposed the model to the full genomic landscape, enabling robust feature extraction from functionally 'negative' background sequences.

Table 15: **Datasets for 1.2B model, 50B token total, 8k seq length.**

| CURRENT_ACCESSION | ORGANISM_NAME                    | CURRENT_ACCESSION | ORGANISM_NAME                 |
|-------------------|----------------------------------|-------------------|-------------------------------|
| HUMAN (HPRC)      | <i>Homo sapiens</i>              | GCA_021498455.1   | <i>Colobus guereza</i>        |
| GCA_047047975.1   | <i>Trachypithecus germaini</i>   | GCA_026956025.1   | <i>Macaca cyclopis</i>        |
| GCA_049640505.1   | <i>Gorilla beringei beringei</i> | GCA_046862485.1   | <i>Callithrix penicillata</i> |
| GCA_030222135.1   | <i>Aotus nancymae</i>            | GCA_965153135.1   | <i>Papio papio</i>            |
| GCA_047655295.1   | <i>Semnopithecus entellus</i>    | GCA_023783065.1   | <i>Nomascus siki</i>          |

Table 16: **Datasets for 10B model, 200B token total, 8k seq length.**

| CURRENT_ACCESSION | ORGANISM_NAME                          | CURRENT_ACCESSION | ORGANISM_NAME                       |
|-------------------|----------------------------------------|-------------------|-------------------------------------|
| HUMAN (HPRC)      | <i>Homo sapiens</i>                    | GCA_014849445.1   | <i>Cercopithecus mona</i>           |
| GCA_047047975.1   | <i>Trachypithecus germaini</i>         | GCA_031835075.1   | <i>Saguinus oedipus</i>             |
| GCF_008728515.1   | <i>Papio anubis</i>                    | GCA_023783475.1   | <i>Daubentonia madagascariensis</i> |
| GCF_009663435.1   | <i>Callithrix jacchus</i>              | GCF_009828535.3   | <i>Hylobates moloch</i>             |
| GCA_030128845.1   | <i>Pan troglodytes</i>                 | GCA_965153135.1   | <i>Papio papio</i>                  |
| GCA_000235385.1   | <i>Saimiri boliviensis boliviensis</i> | GCF_008122165.1   | <i>Gorilla gorilla gorilla</i>      |
| GCA_947095605.1   | <i>Pongo pygmaeus</i>                  | GCA_026956025.1   | <i>Macaca cyclopis</i>              |
| GCA_046862485.1   | <i>Callithrix penicillata</i>          | GCA_049640505.1   | <i>Gorilla beringei beringei</i>    |
| GCF_003339765.1   | <i>Macaca mulatta</i>                  | GCA_023764695.1   | <i>Pygathrix nigripes</i>           |
| GCA_008086735.1   | <i>Cheirogaleus medius</i>             | GCA_048565385.1   | <i>Saimiri boliviensis</i>          |
| GCA_028878085.3   | <i>Symphalangus syndactylus</i>        | GCF_041146395.1   | <i>Eulemur rufifrons</i>            |
| GCF_024542745.1   | <i>Macaca thibetana thibetana</i>      | GCA_023807365.1   | <i>Macaca silenus</i>               |
| GCF_009764315.1   | <i>Trachypithecus francoisi</i>        | GCA_047655295.1   | <i>Semnopithecus entellus</i>       |
| GCA_023783065.1   | <i>Nomascus siki</i>                   | GCF_000165445.2   | <i>Microcebus murinus</i>           |
| GCA_040869165.1   | <i>Theropithecus gelada</i>            | GCA_030222135.1   | <i>Aotus nancymae</i>               |
| GCA_023783095.1   | <i>Macaca assamensis</i>               | GCA_040437455.1   | <i>Plecturocebus cupreus</i>        |
| GCA_047496115.1   | <i>Brachyteles arachnoides</i>         | GCA_023783575.1   | <i>Cebus albifrons</i>              |
| GCF_012559485.2   | <i>Macaca fascicularis</i>             | GCA_047834645.1   | <i>Nasalis larvatus</i>             |
| GCA_021498455.1   | <i>Colobus guereza</i>                 |                   |                                     |

## B Training Setup

### B.1 Pretraining Setup

To validate the effectiveness of the Gengram module, we trained a 10B-parameter model named Gengram-10B-L3\_6\_10-W21-8k(Gengram-10B-8k) using the Megatron-LM framework on a distributed system with 8-GPU nodes and pipeline parallelism of 2. The model consisted of 12 transformer layers with a hidden size of 4096 and 16 attention heads, incorporating RMSNorm, Rotary Position Embedding (RoPE), Switched Gated Linear Unit (SwiGLU) activation, and group query attention with 8 query groups (see Appendix Table 19). The Gengram module was enabled at layers 3, 6, and 10, configured with n-gram sizes from 1 to 6, an embedding dimension of 1024 per n-gram, token IDs (8, 5, 6, 7, 9) representing nucleotides A, C, G, T, and N respectively, a short convolution of kernel size 4, and a local window size of 21. The design also employed MoE architecture with 8 experts, top- $k$  routing ( $k=2$ ), auxiliary loss-based load balancing, and Grouped GEMM (General Matrix Multiplication). The model was trained on a mixed human and non-human genomic dataset comprising approximately 25 million samples (200B tokens), with a sequence length of 8192 and maximum position embeddings of 8192 (see Appendix A.1). Key hyperparameters included a global batch size of 1024, an initial learning rate of  $1e-4$  decaying cosine to  $1e-5$ , weight decay of 0.1, gradient clipping at 1.0, and BF16 precision with flash attention. To ablate the contribution of the local window mechanism, a variant without the window, Gengram-10B-L3\_6\_10-8k, was also trained. Additionally, a baseline model without the Gengram module, Baseline-10B-8k, was trained for comparison.

### B.2 Evaluation of Computational and Training Efficiency

We conducted systematic profiling experiments on the 1.2B model (architecture as in Appx. Table2) across configurations with varying numbers of Gengram layers and window settings, measuring average TFLOPS and per-iteration wall-clock time throughout training (Appx. Table1).

Table 17: **Datasets for 10B model, 100B token total, 32k seq length.**

| CURRENT_ACCESSION | ORGANISM_NAME                   | CURRENT_ACCESSION | ORGANISM_NAME                   |
|-------------------|---------------------------------|-------------------|---------------------------------|
| HUMAN(HPRC)       | <i>Homo sapiens</i>             | GCA_003339765.3   | <i>Macaca mulatta</i>           |
| GRCh38            | <i>Homo sapiens</i>             | GCA_013052645.3   | <i>Pan paniscus</i>             |
| GCA_002880755.3   | <i>Pan troglodytes</i>          | GCA_023764695.1   | <i>Pygathrix nigripes</i>       |
| GCF_028878055.3   | <i>Symphalangus syndactylus</i> | GCA_007565055.1   | <i>Rhinopithecus roxellana</i>  |
| GCF_030222135.1   | <i>Aotus nancymae</i>           | GCA_023767775.1   | <i>Pongo pygmaeus</i>           |
| GCA_028878055.3   | <i>Symphalangus syndactylus</i> | GCA_965153045.1   | <i>Macaca nemestrina</i>        |
| GCF_003255815.1   | <i>Theropithecus gelada</i>     | GCF_041146395.1   | <i>Lemur catta</i>              |
| GCA_021498475.1   | <i>Saguinus midas</i>           | GCA_023761135.1   | <i>Symphalangus syndactylus</i> |
| GCA_903645265.1   | <i>Macaca fascicularis</i>      | GCA_012559485.3   | <i>Macaca fascicularis</i>      |

## C Biological interpretability of the hash table

We further probed the interpretability of the 3-gram hash table learned by Gengram. Because DNA is double-stranded, signals tied to structure/accessibility are expected to be approximately invariant under reverse complementation (RC). Meanwhile, if the table behaved like a motif dictionary, codon representations would tend to collapse by amino-acid semantics (i.e., synonymous codons cluster together). We therefore examined the hash table from two complementary angles: RC symmetry and synonymous-codon clustering consistency, to determine whether it functions as a reusable motif memory rather than a static motif dictionary.

For each layer, we enumerated all 64 codons and constructed the reverse complement  $rc(c)$  for each codon  $c$ . We computed the cosine distance  $d(c, rc(c))$  in representation space. Using a cluster map obtained by clustering codon embeddings, we categorized each RC pair as:

**within:**  $c$  and  $rc(c)$  fall into the same cluster, so  $d(c, rc(c))$  contributes to the within distribution;

**between:** otherwise, it contributes to the between distribution.

We compared within vs. between using a Mann–Whitney U test (and visualized them with violin plots). If RC equivalence is learned, RC pairs should be more “cluster-aligned,” i.e., within distances become significantly smaller than between.

The RCwithin vs RCbetween comparison reveals a clear layer-wise pattern. In Layer3 and Layer10, within distances are generally smaller than between distances, with statistically significant separation (marked as \*\* and \* in the Fig. 7) left, while in Layer6 the difference is not significant (ns). This suggests that the 3-gram memory does not enforce RC equivalence uniformly across layers; instead, it exhibits a division of labor where early or deep layers are more RC-aligned, whereas the middle layer is more task-sensitive. This is biologically plausible: intermediate representations may emphasize strand-aware cues such as transcription directionality and splice donor/acceptor patterns (which are inherently not RC-invariant), while deeper layers integrate broader context and favor features tied to double-stranded structure/accessibility, leading to stronger RC pairing alignment (more “same-cluster” RC pairs and smaller within distances).

We also find that synonymous codons do not systematically collapse into the same cluster. For many amino acids, their synonymous codons are distributed across multiple clusters; quantitatively, the agreement between cluster labels and amino-acid labels is weak (e.g., near-zero ARI and only moderate NMI). This behavior argues against the hash table being a static motif dictionary that compresses codons purely by amino-acid semantics. Instead, it is more consistent with a motif memory: representations jointly reflect codon usage bias, local sequence context, coupling with regulatory constraints (e.g., splice-related short motifs), and mixed coding/non-coding pressures, hence they are not fully governed by the “synonymous = equivalent” rule (Fig. 7 right).

### C.1 Window Selection

We analyzed the norm of the block output hidden states: for each position  $t$ , we took the final hidden state vector after self-attention and Gengram residual injection, and computed its  $L2$  norm

$$\|h_t\|_2 \leftarrow h + \text{Gengram}(h, \text{input\_ids}) \quad (16)$$

Table 18: **Benchmarks Datasets.**

|                                       | TASK                                            | SOURCE | LENGTH (BP) | CLASSIFIER |
|---------------------------------------|-------------------------------------------------|--------|-------------|------------|
| GENOMIC<br>STRUCTURE<br>UNDERSTANDING | DEMO_CODING_VS<br>_INTERGENOMIC_SEQS            | GB     | 200         | MLP        |
|                                       | SPLICE_SITES_ALL                                | NTB    | 400         | MLP        |
|                                       | HUMAN_GENOMIC_ELEMENT                           | GeB    | UP TO 1024  | MLP        |
|                                       | MOUSE_GENOMIC_ELEMENT                           | GeB    | UP TO 1024  | MLP        |
|                                       | MULTI_SPECIES_<br>EXON_CLASSIFICATION           | GeB    | 1024        | MLP        |
|                                       | MULTI_SPECIES_<br>THREE_PRIME_UTR               | GeB    | 1024        | MLP        |
|                                       | MULTI_SPECIES_<br>FIVE_PRIME_UTR                | GeB    | 1024        | MLP        |
| GENE<br>REGULATION<br>PREDICTION      | REGULATORY_ELEMENT_<br>PROMOTER_8K              | LRB    | 8192        | MLP        |
|                                       | REGULATORY_ELEMENT_<br>ENHANCER_8K              | LRB    | 8192        | MLP        |
|                                       | HUMAN_ENHANCERS_COHN                            | GB     | 500         | MLP        |
|                                       | HUMAN_OCR_ENSEMBL                               | GB     | UP TO 593   | MLP        |
| VARIANT EFFECT AND<br>CLINICAL IMPACT | VARIANT_EFFECT_<br>CAUSAL_EQTL_8K               | LRB    | 8192        | MLP        |
|                                       | CPC_8192                                        | GeB    | 8192        | MLP        |
|                                       | CPC_32768                                       | GeB    | 32768       | MLP        |
| EPIGENETIC PROFILING                  | H3                                              | NTB    | UP TO 500   | MLP        |
|                                       | H3K36ME3                                        | NTB    | UP TO 500   | MLP        |
| EVOLUTIONARY ANALYSIS                 | PRIMATE_MAMMAL_SPECIES_<br>CLASSIFICATION_8192  | GeB    | 8192        | XGBOOST    |
|                                       | PRIMATE_MAMMAL_SPECIES_<br>CLASSIFICATION_32768 | GeB    | 32768       | XGBOOST    |

This metric captures the position-wise magnitude of representations and serves as an interpretable signal closely related to numerical stability and the strength of residual updates. Since the hidden state already includes the attention contribution, the observed changes reflect how Gengram modulates the entire block representation via its residual update, rather than the Engram branch alone. We took a checkpoint trained with the non-windowed Gengram configuration and ran two inference-time forward passes:

- (i) the original non-windowed implementation to obtain the position-wise  $\|h_t\|_2$  curve;
- (ii) keeping the checkpoint weights unchanged, we replaced the Engram readout/aggregation with a windowed variant (e.g., window size 21) and recomputed  $\|h_t\|_2$ .

As shown in Figure 9, this is a “structure intervention with frozen weights”: if the windowed variant substantially changes the magnitude fluctuation pattern of hidden states, it indicates that the window’s effect is not due to additional training, but due to a structural constraint on memory retrieval/aggregation, which alters the statistics of the residual update and thus the block’s output dynamics.

In the non-windowed implementation,  $\|h_t\|_2$  exhibits more pronounced high-frequency jitter and spiky fluctuations across positions, suggesting that the Gengram residual injection introduces more uneven magnitude perturbations along the sequence. After introducing a window (e.g., 21 bp), the  $\|h_t\|_2$  curve becomes smoother, local spikes are suppressed, and the magnitude variation is more bounded/regulated. This indicates that windowed retrieval converts highly position-sensitive “point-like” memory triggering into a more consistent local aggregation response, reducing abrupt local changes in representation magnitude—an effect of structural stabilization.

Importantly, since we measure the full block output, this stability implies not only a smoother Engram branch, but also reduced uncertainty in how the Engram residual composes with the attention residual,

Table 19: Configuration comparison of 10B-parameter models with and without the Gengram module

| PARAMETER                   | BASELINE-10B | GENGRAM-10B-L3_6_10 | GENGRAM-10B-L3_6_10-W21 |
|-----------------------------|--------------|---------------------|-------------------------|
| TOTAL PARAMETERS            | 10B          |                     |                         |
| ACTIVATED PARAMETERS        | 2.87B        |                     |                         |
| HIDDEN SIZE                 | 4096         |                     |                         |
| NUM LAYERS                  | 12           |                     |                         |
| ATTENTION HEADS             | 16           |                     |                         |
| GQA QUERY GROUPS            | 8            |                     |                         |
| NORMALIZATION               | RMSNORM      |                     |                         |
| ACTIVATION                  | SwIGLU       |                     |                         |
| POSITIONAL EMB              | RoPE         |                     |                         |
| NUM EXPERTS                 | 8            |                     |                         |
| ROUTER TOP-K                | 2            |                     |                         |
| MoE OPERATORS               | GROUPED GEMM |                     |                         |
| LOAD BALANCING              | AUX LOSS     |                     |                         |
| GENGRAM LAYERS              | \            | 3, 6, 10            |                         |
| GENGRAM NGRAM-SIZES         | \            | 1, 2, 3, 4, 5, 6    |                         |
| GENGRAM WINDOW-SIZES        | \            | \                   | 21                      |
| GENGRAM CONV-KERNEL-SIZE    | \            | 4                   |                         |
| GENGRAM EMBED-DIM-PER-NGRAM | \            | 1024                |                         |
| SEQ LENGTH                  | 8192 / 32768 |                     |                         |
| GLOBAL BATCH SIZE           | 1024         |                     |                         |
| LEARNING RATE               | 1E-4 TO 1E-5 |                     |                         |
| LR SCHEDULER                | COSINE DECAY |                     |                         |

yielding more stable hidden states overall.

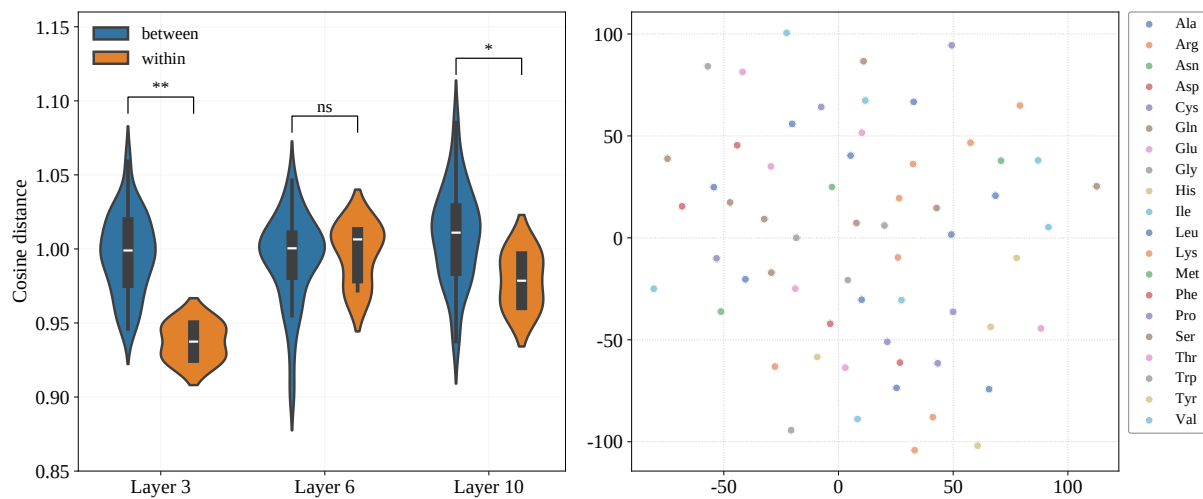


Figure 7: **RC violin (within vs between) at left:** demonstrates whether RC pairs are aligned in the learned memory space. The key takeaway is: smaller within  $\Rightarrow$  RC pairs tend to co-cluster and be more similar  $\Rightarrow$  memory captures strand/double-helix equivalence, with a meaningful layer-wise specialization (significant at L3/L10 but not at L6). **t-SNE by AA at right:** serves as a counter-evidence showing the representations do not trivially cluster by amino-acid identity, supporting “memory rather than dictionary.”

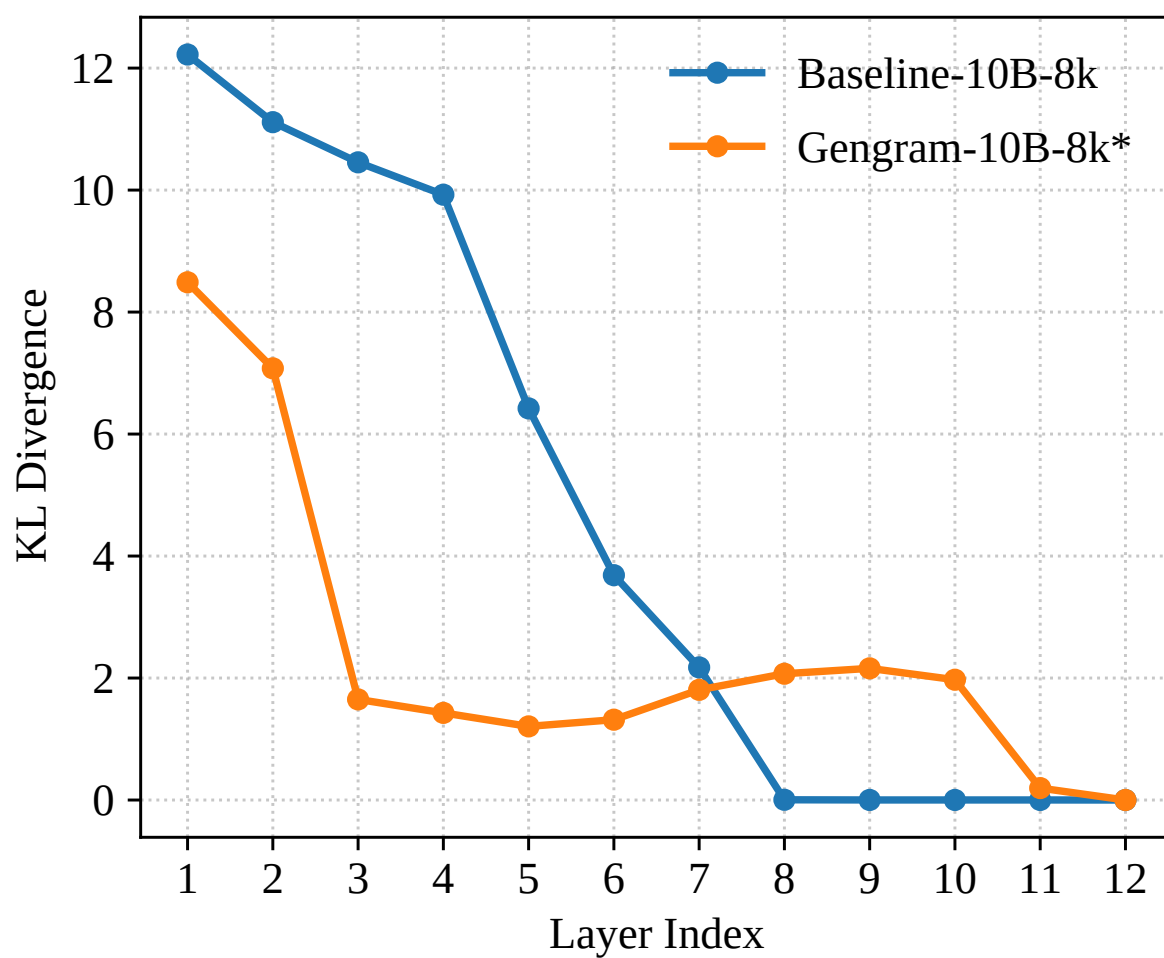


Figure 8: **Layer-wise prediction readiness via LogitLens-KL.** Compared to baseline-10B-8k, \*Gengram-10B-8k exhibits a much faster KL drop in early layers. (GENGRAM-10B-8k\* denotes the GENGRAM-10B-L3\_6\_10-W21-8k model for short.)

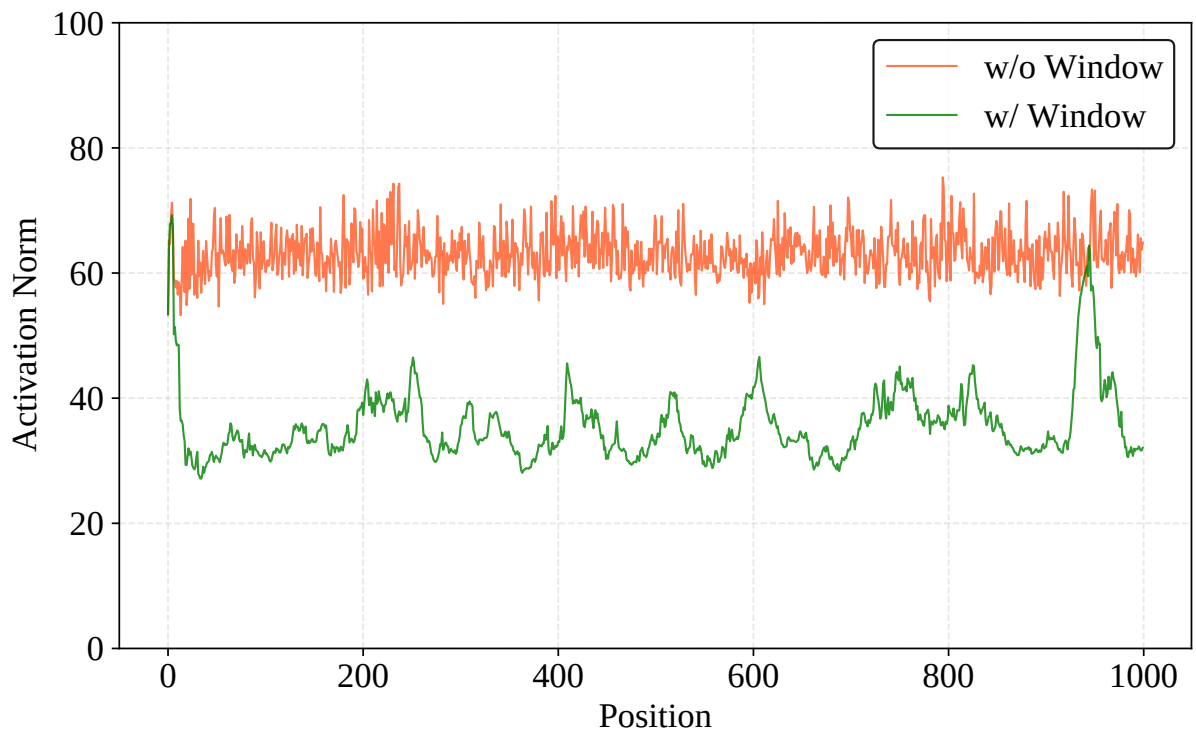


Figure 9: **Windowed Gengram stabilizes block output representation magnitude.** We plot the position-wise L2 norm of the Transformer block output hidden states  $\|h_t\|_2$ , which includes self-attention and the Gengram residual injection. Using the same checkpoint trained without windowing, we compare inference-time forward passes with (green) vs. without (orange) windowed Gengram retrieval. Windowing produces smoother and more bounded magnitude dynamics, suggesting that local aggregation in memory readout suppresses spiky residual perturbations and stabilizes the composition between Engram and attention updates.

## D Matched-Parameter Local-Context Controls

We conducted experiments comparing Gengram against (i) a strong matched-parameter conv-local branch and (ii) a k-mer tokenization + pooling branch, with identical training conditions. Gengram achieved lower loss (loss=1.01) than conv-local (loss=1.02) and k-mer (loss=1.07). All models were trained using the training setup of the 1.2B-parameter model. The loss curves (Appx. Fig 10) show that the conv-local branch has huge spikes, while the k-mer branch has no large spikes but its loss is higher than others. Therefore Gengram is an excellent choice: smaller spikes and lower loss. This indicates that the gain is not simply from adding a local branch, but from Gengram’s retrieval-and-gating mechanism, which is more effective than static local filtering (conv-local) or static k-mer pooled features at same insertion locations.

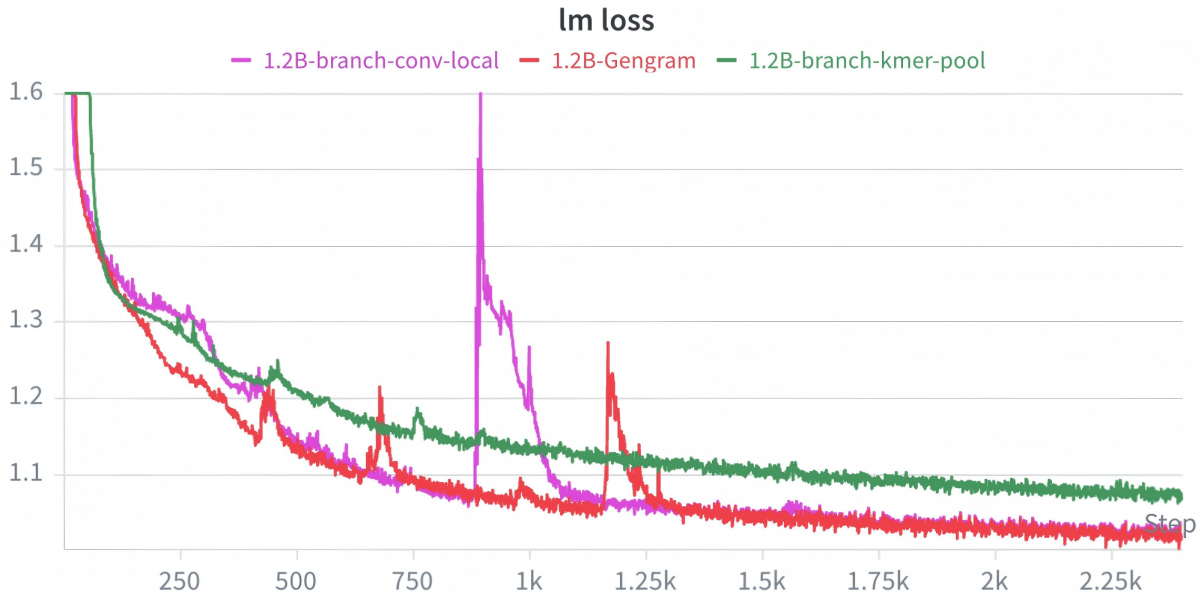


Figure 10: Matched-Parameter Local-Context controls.

## E Genome-wide Residual-Write Analysis

To assess whether the localized residual-write patterns observed in Sec. 5.5.2 generalize beyond individual sequence examples, we performed genome-wide enrichment and in-silico perturbation analyses using held-out chromosomes. These analyses were designed to distinguish reproducible sequence-order-dependent responses from effects attributable to simple nucleotide composition, repetitive sequence, or low-complexity regions.

### E.1 Genome-wide Enrichment and Held-out Validation

We sampled 1,000 non-overlapping 4,096-bp windows from the hg38 reference genome and partitioned them by chromosome into discovery and validation sets. For Gengram layers 3, 6, and 10, positions with high residual-write magnitude were treated as foreground positions. Each foreground position was matched to background positions from the same source sequence and the same Gengram layer, while controlling for genomic annotation, GC content, repeat coverage, and sequence complexity. Candidate sequence patterns were defined exclusively using the discovery set and were subsequently evaluated on held-out chromosomes without redefining the candidate set.

Across the three Gengram layers, 481 candidate sequence patterns were enriched in the discovery set, of which 320 replicated in the held-out validation set, corresponding to a replication rate of 66.5%. Under 1:1 foreground-background matching, among the 320 replicated candidates, 99.3% of candidates retained the

Table 20: **Representative sequence patterns with replicated residual-write enrichment on held-out chromosomes.** OR denotes the validation enrichment odds ratio. Paired effects are estimated at the source-sequence level, with 95% bootstrap confidence intervals. Reported  $q$ -values correspond to multiple-testing-adjusted significance measures for the respective tests.

| Layer | $k$ -mer (reverse complement) | Discovery empirical $q$ | Validation OR; Fisher $q$        | Validation paired effect [95% CI]; Wilcoxon $q$ |
|-------|-------------------------------|-------------------------|----------------------------------|-------------------------------------------------|
| 3     | AGCAGAA (TTCTGCT)             | 0.0308                  | 7.36; 0.00033                    | 0.0198 [0.0047, 0.0390]; 0.0248                 |
| 6     | ATGTGCA (TGCACAT)             | 0.0347                  | $\infty$ ; $1.49 \times 10^{-4}$ | 0.0046 [0.0011, 0.0089]; 0.0272                 |
| 10    | GAGAC (GTCTC)                 | 0.0495                  | 2.78; 0.0241                     | 0.0226 [0.0016, 0.0465]; 0.0350                 |

Table 21: **In-silico perturbation effects across Gengram layers.** Positive differences indicate a larger decrease in residual-write magnitude following perturbation of the candidate core. Confidence intervals were obtained by bootstrap resampling, and  $q$ -values were adjusted using the Benjamini–Hochberg procedure.

| Layer | vs. matched random: difference [95% CI]; BH $q$ | vs. flank: difference [95% CI]; BH $q$      |
|-------|-------------------------------------------------|---------------------------------------------|
| 3     | 0.910 [0.584, 1.266]; $3.47 \times 10^{-6}$     | 0.676 [0.355, 1.015]; $3.67 \times 10^{-4}$ |
| 6     | 0.529 [0.315, 0.768]; $5.38 \times 10^{-5}$     | 0.436 [0.196, 0.689]; $1.40 \times 10^{-3}$ |
| 10    | 0.383 [0.130, 0.642]; $5.98 \times 10^{-3}$     | 0.299 [0.018, 0.577]; $1.34 \times 10^{-2}$ |

same effect direction between discovery and validation, and their effect rankings were highly concordant (Spearman  $\rho = 0.909$ ). Replication was defined as a concordant enrichment effect with validation Fisher FDR  $q < 0.05$ . These results indicate that the residual-write preferences identified in discovery are largely reproducible across chromosomes rather than being specific to the sampled loci.

Because simple sequence composition can generate apparent  $k$ -mer enrichment, we further constructed a stricter empirical null that preserves dinucleotide composition. Under this control, 130 candidates remained significant at empirical FDR  $q < 0.05$ , including 70 candidates not classified as low-complexity sequences. Among these, ten candidates additionally showed positive validation effects together with significant Fisher and sequence-level Wilcoxon FDRs and bootstrap confidence intervals above zero. Representative validated candidates are shown in Table 20.

## E.2 In-silico Perturbation Analysis

We next tested whether the residual-write responses associated with the enriched sequence patterns depend on nucleotide order rather than nucleotide composition alone. For each candidate, we shuffled the nucleotide order within the candidate core while preserving its length and mononucleotide composition. We compared the resulting change in residual-write magnitude with two controls: perturbation of distal regions matched for sequence composition and perturbation of adjacent flanking regions.

The source sequence, rather than individual nucleotide positions, was used as the statistical unit. Across layers 3, 6, and 10, perturbing the candidate core produced a consistently larger decrease in residual-write magnitude than perturbing either composition-matched distal regions or adjacent flanks. All estimated differences had bootstrap confidence intervals above zero, and the corresponding comparisons remained significant after Benjamini–Hochberg correction (Table 21).

The effect was strongest at layer 3 and decreased progressively at deeper layers. Relative to composition-matched distal perturbations, the estimated differences were 0.910, 0.529, and 0.383 for layers 3, 6, and 10, respectively. Relative to adjacent-flank perturbations, the corresponding differences were 0.676, 0.436, and 0.299. These results indicate that Gengram residual writes are sensitive to the local nucleotide ordering of candidate sequence patterns and cannot be explained solely by their overall nucleotide composition.

Finally, we compared the enriched sequence patterns with known transcription-factor motifs using JASPAR/FIMO. After accounting for low-complexity and repeat-associated sequences, no JASPAR/FIMO TF motif remained significant in held-out validation. We therefore do not interpret the observed residual-write enrichment as evidence of TF-specific binding memory. Instead, together with the perturbation results, these analyses support a more constrained interpretation in which Gengram exhibits reproducible, input-conditioned, and local-order-sensitive motif-like sequence memory.

## **F GenAI Disclosure**

AI assistants were used for limited language editing and writing assistance. No AI-generated scientific claims, experimental results, or analyses were incorporated. All datasets, tables, figures, and the original manuscript content were produced by the authors.