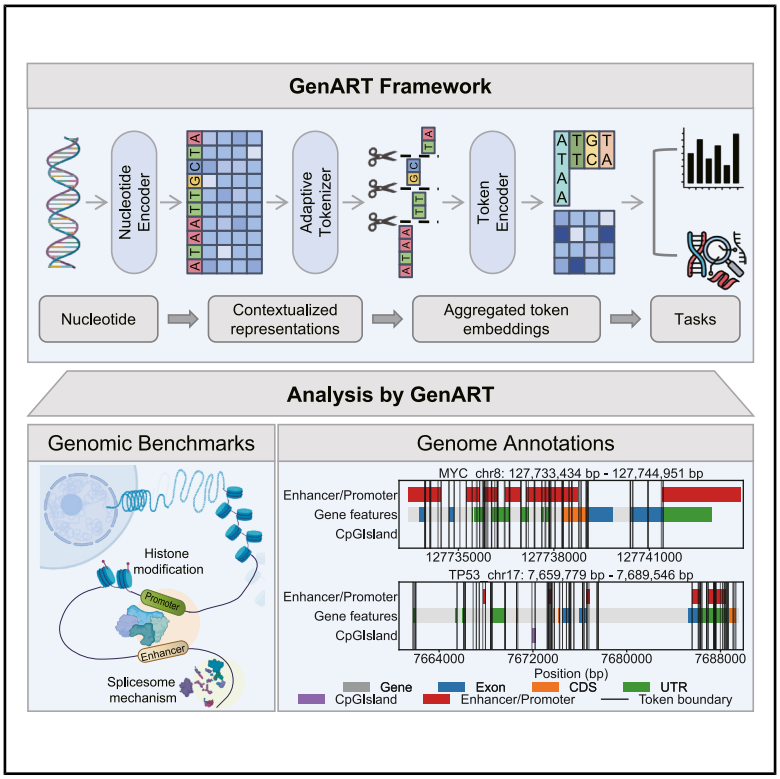
Cell Reports
Methods

Article

GenART: Reading the genome's language with adaptive "words"

Graphical abstract



Highlights

- GenART dynamically segments DNA into variable-length biological words
- GenART achieves competitive performance across 33 genomic tasks
- Unsupervised tokens from GenART naturally align with functional gene boundaries
- Token refinement adapts DNA segmentation to specific biological tasks

Authors

Kehan Chen (陈可涵), Ping Han (韩萍), Zhe Lin (林哲), ..., Yeyun Gong (宫叶云), Zhiliang Ji (纪志梁), Chen Lin (林琛)

Correspondence

cheyenne.lin@foxmail.com

In brief

Chen et al. develop GenART, a genomic language model that dynamically segments raw DNA into biologically meaningful "words." This adaptive approach overcomes arbitrary sequence splitting, boosting predictive performance across diverse tasks while autonomously capturing functional genomic boundaries to enhance interpretability.

Chen et al., 2026, Cell Reports Methods 6, 101516
September 21, 2026
https://doi.org/10.1016/j.crmeth.2026.101516

CellPress

Cell Reports Methods

CellPress
OPEN ACCESS

Article

GenART: Reading the genome's language with adaptive "words"

Kehan Chen (陈可涵),[1,6] Ping Han (韩萍),[1,6] Zhe Lin (林哲),[5,6] Shang Gao (高尚),[2] Xinyu Yang (杨心宇),[3] Xiangxiang Zeng (曾湘祥),[3] Yeyun Gong (宫叶云),[4] Zhiliang Ji (纪志梁),[5] and Chen Lin (林琛)[1,2,7,*]

[1]School of Informatics, National Institute for Data Science in Health and Medicine, Xiamen University, Xiamen 361000, China
[2]Zhongguancun Academy, Beijing 100094, China
[3]College of Computer Science and Electronic Engineering, Hunan University, Changsha 410082, China
[4]Microsoft Research Asia, Microsoft, Beijing 100080, China
[5]State Key Laboratory of Cellular Stress Biology, School of Life Sciences, Faculty of Medicine and Life Sciences, Xiamen University, Xiamen 361000, China
[6]These authors contributed equally
[7]Lead contact
*Correspondence: cheyenne.lin@foxmail.com
https://doi.org/10.1016/j.crmeth.2026.101516

MOTIVATION Genomic language models have advanced DNA sequence analysis by learning general-purpose representations from large-scale genomes. However, DNA lacks natural delimiters, and current models usually segment sequences using single nucleotides, fixed-length k-mers, or statistically derived subwords, which do not explicitly reflect biological context or function. These tokenization strategies limit interpretability and can compromise generalization across diverse genomic tasks. To address this, we developed GenART, an adaptive genomic language model that dynamically discovers variable-length genomic "words" directly from raw DNA sequences.

SUMMARY

Tokenization is both a prerequisite and a central challenge for genomic language models, due to the inherent difficulty of delineating meaningful segments within continuous DNA sequences. We present GenART, a genomic language model framework that dynamically segments DNA into variable-length words directly from raw sequences without manual annotation. Its key adaptive tokenization module infers biologically meaningful boundaries from multi-scale contextual signals during pretraining. GenART demonstrates competitive performance across diverse tasks, outperforming leading approaches. Word boundaries derived from unsupervised tokenization show strong correspondence with base-resolution regulatory signals, such as DNA methylation, and with multi-nucleotide functional elements annotated in GENCODE v.49. These findings highlight adaptive tokenization as a powerful strategy for building interpretable and biologically responsive genomic language models.

INTRODUCTION

Language models such as BERT[1] and GPT[2] have profoundly transformed natural language processing (NLP) by learning general-purpose representations from vast amounts of unlabeled text. This paradigm has inspired a new generation of "genomic language models" that aim to apply similar principles to biology, treating DNA as the fundamental language of life.[3–7]

As computational models cannot process raw sequences directly, *tokenization* is required to split a continuous DNA sequence into discrete, meaningful units. Current genomic language models inherit architectures from NLP, yet their tokenization methods—designed for text with natural boundaries—are inadequate for DNA sequences that lack such delimiters. To illustrate, consider DNA as a vast biological instruction manual. Just as understanding a book requires recognizing words, deciphering DNA's functions depends on identifying its fundamental building blocks. Some previous works split the DNA sequence into single nucleotides (A, T, C, G),[5] akin to reading a book one letter at a time. Other approaches employ k-mers,[3,7,8] which resemble chopping a book into fixed-length chunks (e.g., six letters) regardless of where the actual words end. Byte-pair encoding (BPE),[6,9] MxDNA,[10] and H-Net[11] produce variable-length vocabularies, but the resulting segments are not necessarily biologically informative, analogous to slicing a book by letter frequency instead of semantic boundaries. They have successfully

Please cite this article in press as: Chen et al., GenART: Reading the genome's language with adaptive "words", Cell Reports Methods (2026), https://doi.org/10.1016/j.crmeth.2026.101516

been optimized for specific downstream tasks, but limitations remain for robust generalizability and practical utility across diverse genomic applications.[12]

We propose GenART, a genomic language model that reads genomes as one would read text—progressively segmenting raw DNA sequences word-by-word into biologically meaningful units that also enhance computational learning. This capability is enabled by an *adaptive tokenizer* mechanism that infers word boundaries from multi-scale contextual signals during pretraining, requiring no manual annotation and avoiding reliance on superficial statistics. Moreover, these learned units can be further refined through a dedicated tokenization fine-tuning procedure, allowing GenART to adapt its vocabulary to the semantic and structural demands of diverse biological tasks.

We conducted extensive evaluations. First, we performed comparative assessments against existing genomic language models on two benchmarks covering 33 diverse tasks, demonstrating that GenART not only consistently attains competitive performance across a broad spectrum of genomic prediction tasks but also learns task-specific, biologically interpretable representations. Second, we compared the words segmented by GenART with the latest human genome annotations, showing that GenART effectively captures biologically meaningful words in the genome language.

RESULTS

Overview of the GenART architecture

As shown in Figure 1A, GenART receives a raw DNA sequence as input in self-supervised pretraining. A *nucleotide encoder*, composed of several Transformer blocks, processes the sequence to generate contextualized representations for each nucleotide. This means that the embedding for each base is not static but is informed by its surrounding context, allowing the model to capture local nucleotide-level dependencies. The nucleotide embeddings are then passed to the central component of our model: the *adaptive tokenizer*. This module, detailed in the following paragraph, dynamically partitions the sequence into a series of variable-length, semantically meaningful words (which we refer to as tokens). The resulting sequence of tokens is then fed into a final *token encoder* (i.e., a series of Transformer blocks) to produce token embeddings. These rich, token-level representations serve as the input for our primary pretraining task, masked language modeling (MLM). In this task, we randomly mask certain tokens and train the model to predict them based on the context provided by the surrounding tokens. This process compels the model to learn high-level semantic and syntactic rules inherent in the genome.

As detailed in Figure 1B, the *adaptive tokenizer* begins with *local feature extraction* using multiple parallel 1D convolutional neural network (CNN) layers, each utilizing a different kernel size ranging from 2 to 6. This design allows the model to simultaneously capture multi-scale local motifs, such as dinucleotides, codons, and short transcription factor binding sites. The feature maps from these parallel CNNs are averaged element-wise, and a residual connection[13] adds this composite local information back to the input embeddings, strengthening the representation while stabilizing training. Following *local feature extraction*, the *adaptive segmentation* performs the critical task of dynamic tokenization. It first feeds the enhanced embeddings into a multi-layer perceptron (MLP) to generate a score for each inter-base position (e.g., between A and G or G and C), indicating the likelihood that a segmentation boundary should be placed. The MLP output logits are then passed through a Gumbel-Softmax layer to produce a discrete boundary decision—in this case, to merge or not to merge—while maintaining end-to-end differentiability.

As depicted on the left side of Figure 1C, following pretraining, the GenART model can be fine-tuned for specific downstream tasks by attaching a task-specific head, such as for classification. This allows the model to classify genomic elements into categories like promoters or enhancers, which is crucial for understanding gene regulation and other biological processes.

On the right side of Figure 1C, the radar charts present a comparative analysis of the average performance of 10 distinct genomic language models across two benchmarks. The NT benchmark[7] assesses the models' capabilities in histone modification prediction, enhancer prediction, promoter prediction, and splice site prediction. Meanwhile, the GUE benchmark[6] assesses the models' performance in transcription factor prediction for both humans and mice, core promoter detection, virus variant classification, and splice site prediction. We trained GenART at two scales, 350M and 1B parameters. Despite having far fewer parameters than existing genomic language models such as NT (2.5B) and Evo2 (7B),[14] GenART achieves competitive performance across multiple downstream tasks and, in several cases, surpasses these larger models.

GenART achieves robust performance across diverse downstream benchmarks

To establish the overall effectiveness and generalization capability of GenART, we first conducted a comparative performance evaluation against current state-of-the-art genomic language models. GenART was fine-tuned on the NT benchmark and the GUE benchmark, respectively. For all fine-tuning experiments, we loaded the same pretrained GenART checkpoint and attached a new, randomly initialized classification head on top of the base model. The entire model, including the base and the new head, was then fine-tuned on each specific task.

The NT benchmark consists of 18 datasets (Table S1), including histone modification tasks (Encyclopedia of DNA Elements [ENCODE][15–17]), enhancer prediction tasks (ENCODE), promoter prediction tasks (Eukaryotic Promoter Database[18]), and splice site annotation tasks (GENCODE[19]). It evaluates a model's ability to identify regulatory elements. Following the prior work,[7] we reported the Matthews correlation coefficient (MCC) for tasks related to histone modification and enhancer prediction, accuracy (ACC) for the splice site annotation (both) task, and F1-score for the remaining tasks.

As shown in Figure 2A, GenART delivered the highest average performance on the NT benchmark, exceeding the best baseline by 3.41%. As detailed in Figure 2B and Table S2, GenART consistently demonstrated competitive performance across all 18 tasks. Specifically, GenART-1B achieved state-of-the-art results on 15 out of 18 tasks, while the lightweight GenART-350M outperformed existing baselines on 12 tasks. In the remaining

Please cite this article in press as: Chen et al., GenART: Reading the genome's language with adaptive "words", Cell Reports Methods (2026), https://doi.org/10.1016/j.crmeth.2026.101516

CellPress
OPEN ACCESS

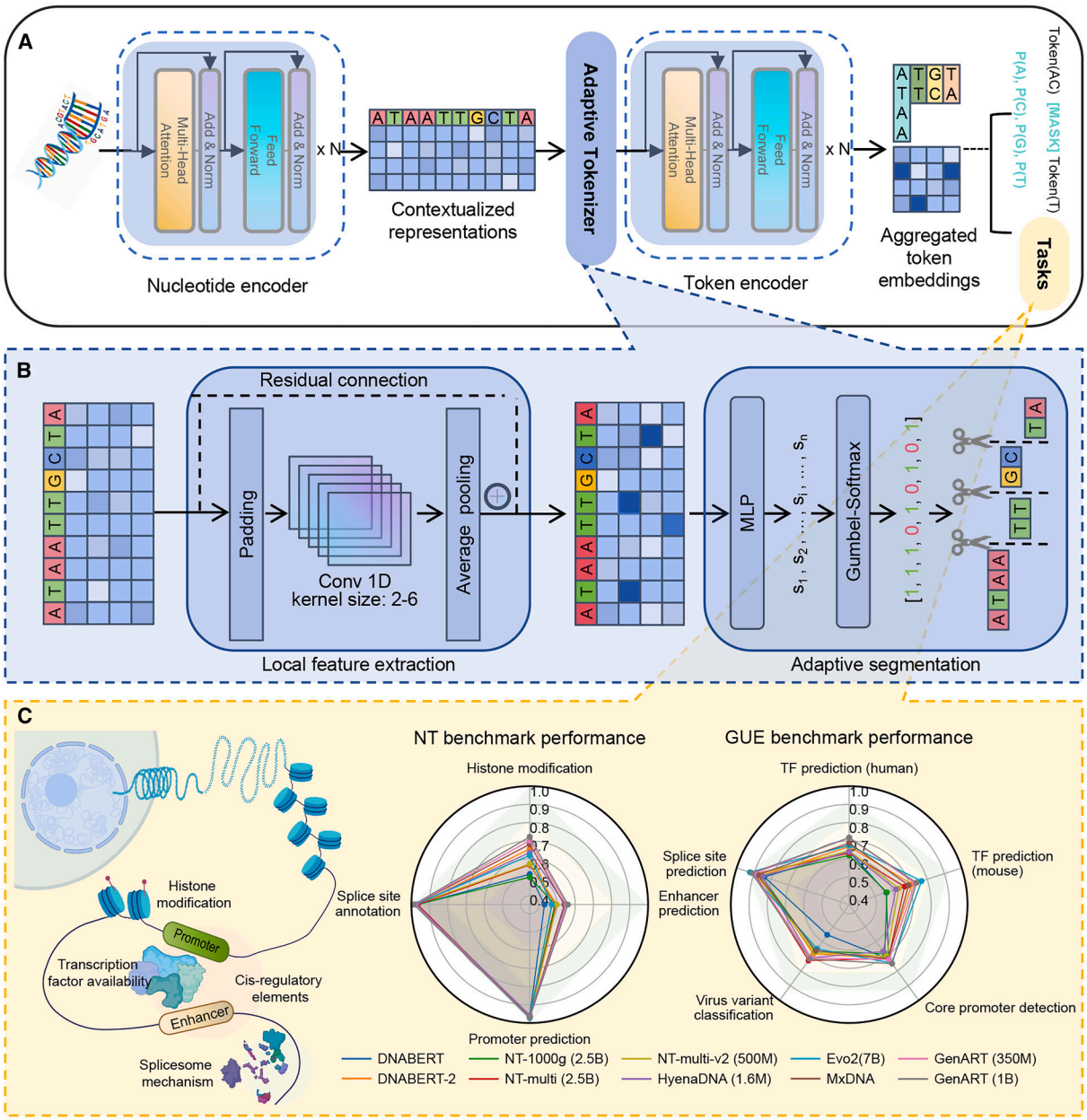


Figure 1. Schematic of the GenART framework

(A) The overall model architecture and pretraining fine-tuning framework of the GenART model.

(B) A detailed schematic of the adaptive tokenization module.

(C) Downstream applications and performance evaluation of genomic language models.

three tasks, GenART-1B still delivered performance nearly on par with that of the significantly larger Evo2-7B. For instance, across three promoter prediction tasks, GenART-350M and GenART-1B both achieved a mean score of 0.971, closely trailing Evo2-7B (0.974). Given that GenART models were 7–20 times smaller than Evo2-7B, these results highlighted our superior efficiency and robustness. Across the three task categories—including histone modification, enhancer prediction, and splice site annotation—GenART improved over the best baseline by an average of up to 6.02%, with the most substantial gain observed in the histone modification category.

The GUE benchmark evaluates a broader spectrum of genomic understanding tasks. We selected a subset of 15 tasks from the GUE benchmark that are not covered by the NT benchmark, including transcription factor prediction, core promoter detection, virus variant classification, and splice site prediction (Table S1). This design allowed for a comprehensive assessment of the model generalizability across diverse biological contexts.

CellPress
OPEN ACCESS

Cell Reports Methods
Article

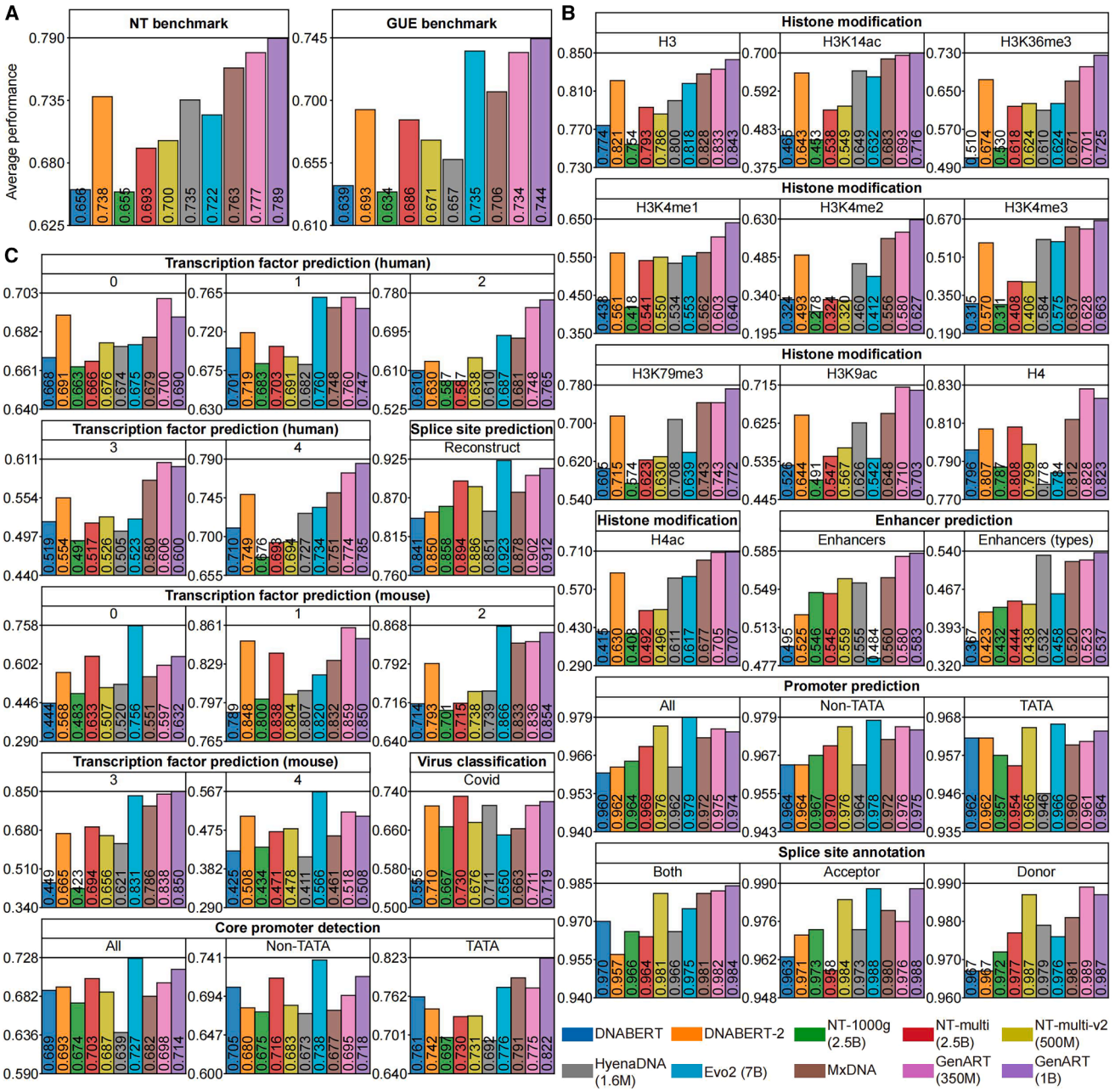


Figure 2. GenART achieves superior performance across diverse downstream tasks on two widely used benchmark datasets

(A) Average performance of the NT benchmark and GUE benchmark.

(B) Performance of 18 downstream tasks on the NT benchmark.

(C) Performance of 15 downstream tasks on the GUE benchmark.

See also Figure S1.

Following the prior work,[6] we reported F1-score for the virus variant classification task and MCC for the remaining tasks.

As shown in Figure 2A, GenART also achieved the highest average performance on the GUE benchmark, surpassing the best baseline by 1.22%. Specifically, as presented in Figure 2C and Table S3, GenART attained the best performance on 8 out of 15 tasks and ranked second on 6 of the remaining tasks. In the scenarios where it ranked second—such as virus variant classification (COVID), splice site prediction, and core promoter detection—GenART trailed the leading models (predominantly the 7-billion-parameter Evo2 and the 2.5-billion-parameter NT-multi) by exceptionally narrow margins, typically ranging from 1.1% to 1.3%. Crucially, GenART delivered this highly competitive performance despite utilizing only 15% to

Please cite this article in press as: Chen et al., GenART: Reading the genome's language with adaptive "words", Cell Reports Methods (2026), https://doi.org/10.1016/j.crmeth.2026.101516

over 20% parameters. Across the diverse task categories, GenART continued to demonstrate robust generalizability and superior parameter efficiency.

To address the potential limitations of standard classification benchmarks (e.g., GUE and NT) in terms of sequence length and task diversity, we evaluated GenART on the challenging task of splice site prediction using the SpliceAI benchmark.[20] Unlike short-read classification, splice site identification requires the model to capture distal regulatory signals across long genomic intervals, providing a rigorous test for both long-context modeling and biological generalizability.

We conducted experiments across three different sequence scales: 400, 2k, and 10k nt. We compared GenART against the original SpliceAI model, which is widely regarded as the state of the art for this specific domain. All models were evaluated using top-k accuracy and the area under the precision-recall curve (PR-AUC). Top-k accuracy measures the model's ability to rank true signals highly, while PR-AUC provides a robust assessment of precision and recall, which is particularly crucial for highly imbalanced genomic datasets.

As illustrated in Figure S1, GenART consistently outperformed SpliceAI across all evaluated context lengths, demonstrating superior robustness in long-range genomic modeling. Notably, at the 10k nt scale—a challenging regime where traditional models often suffer from signal decay or massive computational overhead—GenART achieved a mean top-k accuracy of 0.962, surpassing SpliceAI's 0.950. While SpliceAI exhibited a gradual increase in performance as the context window expanded from 400 to 10k nt, GenART maintained a high and stable performance plateau, reaching near-perfect mean PR-AUC scores across all settings. This stability suggests that GenART effectively captures critical regulatory motifs and preserves their semantic integrity regardless of the sequence length. Furthermore, even at the shorter 400 nt range, GenART's top-k accuracy significantly exceeded that of SpliceAI, indicating that the biologically grounded "units" identified by GenART are highly discriminative for true splice junctions. Collectively, these results confirm that GenART's advantage extends beyond simple classification, successfully capturing the complex, long-distance dependencies essential for precise functional genomic annotation.

Adaptive tokenization drives GenART's performance and discriminative capacity

Having established the superior performance of GenART, we next investigated the underlying mechanisms that account for this advantage. We hypothesized that our key innovation—the learnable adaptive tokenizer—enables more informative sequence decomposition and representation compared with conventional static tokenization schemes. To validate this hypothesis, we systematically dissected the tokenizer's behavior and structural contribution from three complementary perspectives: (1) its contribution to downstream performance across diverse genomic tasks, as evidenced by comprehensive ablation analysis, (2) its capacity to generate highly discriminative, task-specific token representations, and (3) its effectiveness in capturing distinctive token frequency distributions across challenging genomic categories.

To explicitly evaluate the core contribution of the adaptive tokenizer, we conducted an ablation study on the GenART-350M model. For our baseline, we replaced the adaptive tokenizer with a standard single-nucleotide tokenization strategy. Crucially, all other experimental variables, including the pretraining dataset, model hyperparameters, and fine-tuning strategies, were kept strictly identical between the two models.

As illustrated in Figure 3A, integrating the adaptive tokenizer yielded a consistent performance uplift, driving an overall average improvement of 3.27% across both benchmarks. The most substantial gain was observed in histone modification (+9.35%), a task characterized by complex, extended regulatory domains. In a single-nucleotide setup, modeling such extended regions results in excessively long token sequences, which can dilute the model's ability to effectively capture long-range dependencies. The adaptive tokenizer mitigates this by dynamically compressing variable-length motifs, thereby expanding the effective receptive field and enabling the model to better aggregate broader contextual patterns. Conversely, for tasks driven by highly conserved and localized signals (e.g., core promoter prediction), the single-nucleotide baseline already achieved highly saturated performance (exceeding 0.96). In these scenarios, the marginal gains are primarily constrained by the performance ceiling, as base resolution is highly effective at identifying short, specific sequence motifs.

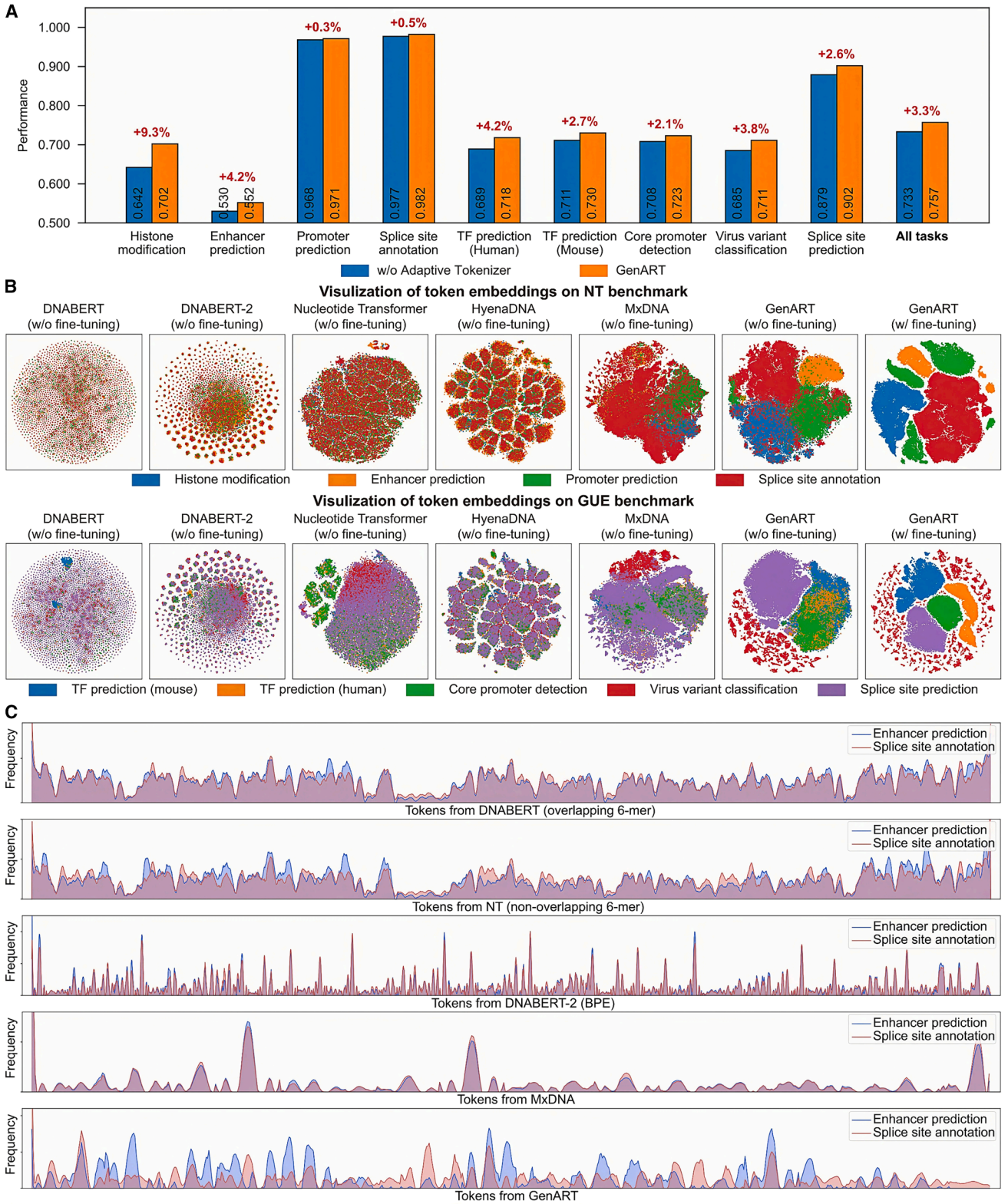
Ultimately, this ablation study indicates that while static tokenization is adequate for localized feature extraction, adaptive tokenization significantly enhances the model's capacity to capture broader contextual dependencies and adapt to complex regulatory syntax, thereby contributing to GenART's robust generalizability.

Current genomic models typically predict downstream tasks by leveraging token representations extracted from input sequences, making the quality of token representations crucial for model performance. To investigate this, we input sequences into our model, GenART, as well as baseline models, and project the embedding vectors for all tokens obtained by these models to a 2D space using t-distributed stochastic neighbor embedding (t-SNE). This allows us to visually compare the token representations produced by GenART and alternative approaches.

Specifically, we classified the sequences in the NT benchmark into four categories and those in the GUE benchmark into five categories according to their different genomic functions. The sequences were first processed by pretrained (but not fine-tuned) models to obtain token embeddings. To ensure a standardized comparison across architectures, we extracted embeddings exclusively from the final hidden layer. As downstream tasks directly utilize these terminal representations, they most accurately reflect a model's capacity to encode functional genomic information. We then applied t-SNE to project these final-layer token embeddings into two dimensions, where tokens from sequences belonging to the same category were labeled with the same color.

As shown in Figure 3B, even without task-specific fine-tuning, GenART effectively separates token representations across different categories, whereas the baseline models fail to do so. This demonstrates that GenART is able to learn and distinguish task-relevant sequence representations solely from unified

Please cite this article in press as: Chen et al., GenART: Reading the genome's language with adaptive "words", Cell Reports Methods (2026), https://doi.org/10.1016/j.crmeth.2026.101516



(legend on next page)

Please cite this article in press as: Chen et al., GenART: Reading the genome's language with adaptive "words", Cell Reports Methods (2026), https://doi.org/10.1016/j.crmeth.2026.101516

pretraining knowledge, which provides a strong foundation for downstream performance improvements.

We further examined the representations obtained from the fine-tuned GenART. After incorporating task-specific supervision, the separation between categories became even more distinct, indicating that the model successfully learns and integrates task-specific knowledge during fine-tuning.

From the results in Figure 3B, we observed that, in the NT benchmark, enhancer prediction and splice site annotation were the most difficult to distinguish for baseline models. This is likely because enhancers and splice sites are both governed by local sequence features, including short functional motifs and combinatorial "regulatory grammars," which are partially conserved and determine their regulatory activity.[21–23] Such intrinsic similarities make it challenging for models with conventional tokenization strategies to capture subtle but critical discriminative features. To further examine these categories, we analyzed the frequency distribution of tokens produced by different models. Specifically, we input the corresponding sequences into pretrained models, collected the resulting tokens, and computed the relative frequency of each token.

As illustrated in Figure 3C, frequency distributions obtained with conventional tokenization methods (e.g., overlapping k-mer, non-overlapping k-mer, and BPE) and the recent MxDNA were highly similar. This indicates that previous approaches fail to provide distinct tokenization patterns for closely related categories, limiting the ability of models to learn discriminative sequence features and ultimately constraining downstream performance. In contrast, the adaptive tokenization introduced by GenART yields clearly differentiated token frequency profiles between the two categories, demonstrating the superiority of our proposed method.

Broad species diversity in training data and expanded model capacity enhance GenART's performance

While pioneering foundational models in genomics, such as Nucleotide Transformer,[7] have explicitly decoupled model capacity scaling from dataset volume scaling, the distinct contributions of pretraining species diversity and pretraining duration frequently remain entangled. In many existing scaling paradigms, the incremental addition of multi-species sequences intrinsically correlates with increased total token exposure and extended training steps, confounding the causal attribution of downstream performance gains. To rigorously address this, we designed highly controlled ablation studies to evaluate the specific impact of phylogenetic diversity on cross-species generalization.

Furthermore, we systematically evaluate the progressive scaling of GenART architectures. While prior foundational models, including Nucleotide Transformer and Evo2,[14] have documented the empirical scaling laws where larger models yield superior downstream performance, we specifically sought to elucidate the underlying mechanisms driving these performance enhancements within the GenART framework.

Previous studies on pretraining dataset diversity often confound the addition of new species sequences with increased token volume or extended training steps. To rigorously isolate the impact of phylogenetic diversity, we conducted a controlled ablation study using identical GenART architectures, fixed training steps, and matched total token exposures. Specifically, we compared a control model pretrained exclusively on the human reference genome (hg38) against an experimental model pretrained on a multi-species corpus sampled to the exact same data volume and token count.

The results in Figure 4A revealed a clear performance trade-off associated with dataset composition. While the multi-species model exhibited a slight performance degradation (1.45%) on human-specific tasks due to a diluted proportion of human sequences during training, this was substantially outweighed by a 2.87% improvement on non-human tasks (e.g., yeast, mouse, and viral genomes). Overall, incorporating a broader phylogenetic spectrum yielded a net average performance increase of 0.98% across all downstream benchmarks.

Beyond average accuracy gains, pretraining with broad species diversity significantly enhanced cross-species generalization. As illustrated in the radar chart (Figure 4B), the multi-species model demonstrated a more balanced performance profile, reducing the standard deviation across diverse task metrics by 3.64%. These findings suggest that phylogenetic diversity acts as a potent regularizer during pretraining, mitigating overfitting to human-specific genomic biases and fostering universally applicable feature representations across evolutionarily distant taxa.

Consistent with established scaling laws, downstream performance improved monotonically across the 100M, 350M, and 1B GenART variants. To mechanistically evaluate this improvement, we quantified attention uniformity using the coefficient of variation (CV), which normalizes variance against the mean to eliminate scale-dependent artifacts. As shown in Figure 4C, the CV consistently decreases as model capacity expands, exhibiting a strong inverse correlation with downstream accuracy. This demonstrates that larger models distribute attention more globally and smoothly, thereby driving superior predictive performance.

This balanced attention mechanism aligns directly with the distributed nature of genomic regulatory landscapes. Within typical 70–1,000 bp windows, regulatory signals—such as clustered transcription factor binding sites (TFBSs) and splicing elements—are rarely localized to a single terminus. We propose that capacity-constrained small models tend to adopt "computational shortcuts," disproportionately fixating on strong local features. Conversely, the expansive representational capacity of the 1B model overcomes this bottleneck, facilitating the simultaneous, holistic integration of spatially dispersed functional elements, which ultimately enhances cross-task generalization.

Figure 3. Adaptive tokenization drives GenART's superior performance and discriminative capacity on task-specific token representations

(A) Ablation analysis comparing the performance of GenART with and without the adaptive tokenizer across various downstream tasks.

(B) t-SNE visualization of token embeddings obtained by different genomic language models for various tasks.

(C) Token frequency distributions obtained by different genomic language models for two tasks.

Please cite this article in press as: Chen et al., GenART: Reading the genome's language with adaptive "words", Cell Reports Methods (2026), https://doi.org/10.1016/j.crmeth.2026.101516

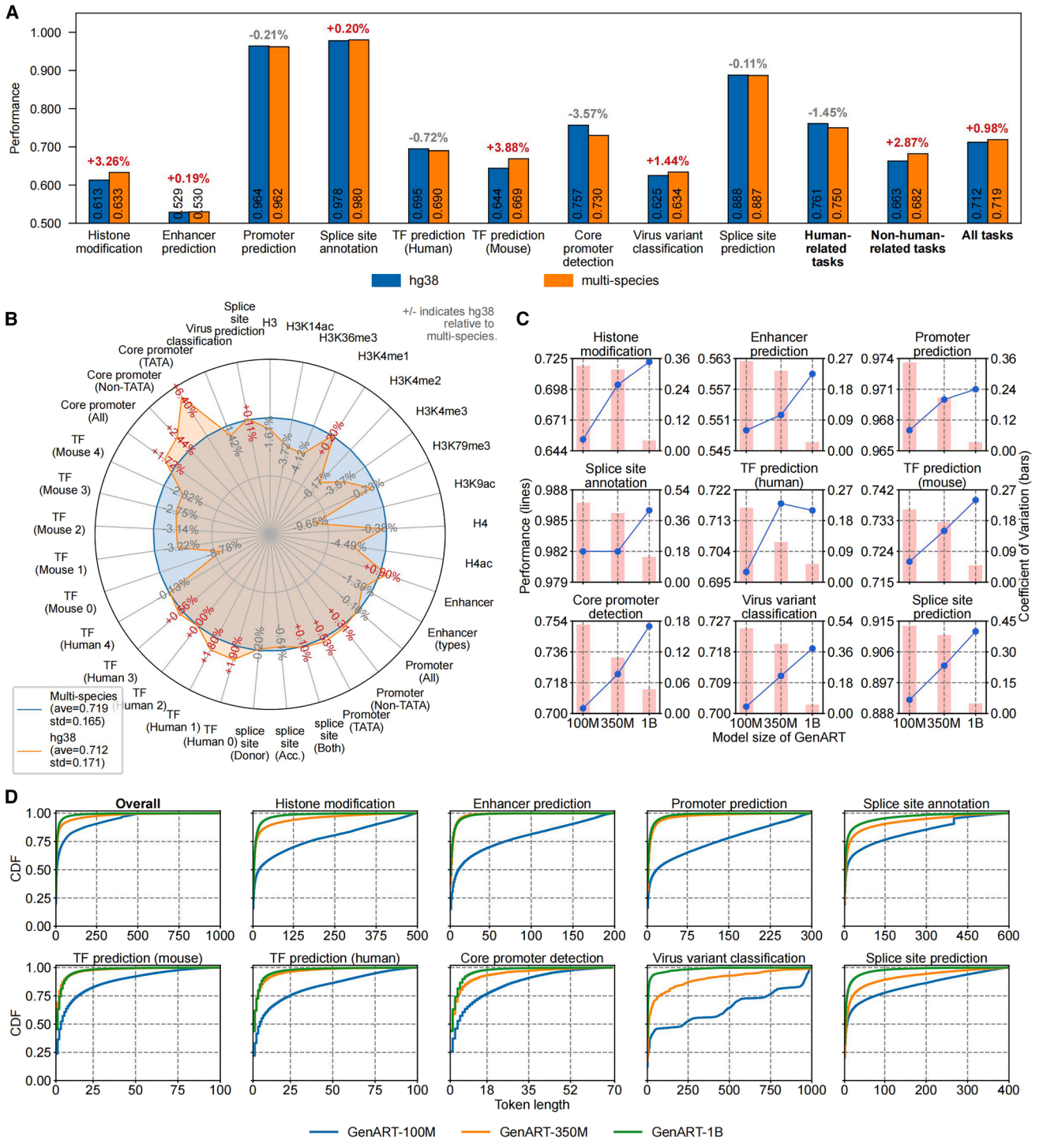


Figure 4. Broad species diversity and expanded model capacity enhance GenART's generalization and downstream performance

(A) Performance comparison of GenART pretrained on human-only (hg38) versus multi-species corpora.

(B) Radar chart detailing relative performance shifts across downstream tasks.

(C) Impact of model scaling on predictive performance and attention uniformity.

(D) Cumulative distribution function (CDF) of token lengths, revealing a shift toward finer granularity in larger models.

Please cite this article in press as: Chen et al., GenART: Reading the genome's language with adaptive "words", Cell Reports Methods (2026), https://doi.org/10.1016/j.crmeth.2026.101516

Building upon the observation that larger models integrate sequence context more holistically, we investigated whether expanded capacity fundamentally alters the adaptive tokenization strategy. Analysis of the cumulative distribution function (CDF) of token lengths across the 100M, 350M, and 1B variants (Figure 4D) reveals a distinct distributional shift. Specifically, the empirical data demonstrate that the average token length decreases monotonically with increasing model scale.

This shift toward finer granularity suggests that enhanced representational capacity reduces the network's reliance on rigidly pre-assembled long tokens to capture local context. Instead, the 1B model opts for a higher resolution, more fundamental vocabulary. By leveraging the globally smoothed attention mechanism, the larger architecture is capable of dynamically modeling complex, long-range dependencies directly from these shorter, granular tokens, ultimately constructing a more flexible and expressive sequence representation.

Fine-tuned token distributions reveal task-specific biological signals

In pretraining, the adaptive tokenization layer in GenART autonomously learned how to segment DNA sequences into variable-length tokens, guided solely by sequence context. To further enhance downstream performance and interpretability, we introduced an optional refinement mechanism as a task-specific optimization step. While the pretrained GenART tokenizer provides a robust representation by default, this mechanism allows users to further bridge the gap between a general-purpose model and specialized biological domains. This is achieved through two tunable parameters, the target token density and the target cluster factor, which jointly control the granularity of the spatial distribution of the token boundaries, allowing GenART to align more closely with biologically meaningful units.

Specifically, the target token density determines the average number of token boundaries per base pair, thereby influencing token length. A lower value of density results in fewer boundaries and thus longer tokens on average, which is suitable for capturing large-scale genomic structures such as chromatin domains or gene clusters. In the context of histone modifications, this is particularly relevant for broad domain-type marks like H3K79me3 and H3K36me3, which span extensive genomic regions and are better modeled with longer, more continuous tokens.

Complementing this, the target cluster factor modulates the degree to which token boundaries are spatially clustered along the sequence. A higher cluster value promotes the formation of boundary-dense regions interspersed with extended token spans, mimicking the uneven distribution of functional elements in the genome—such as promoter-proximal regions with high information density versus long intronic or intergenic regions. This is critical for sharp peak-type marks such as H3K4me3 and H3K9ac, where functional signals are concentrated in narrow, well-defined regions. A higher cluster factor enables the tokenizer to allocate more boundaries around these focal points, thereby resolving fine-grained spatial patterns.

The selection of density and cluster factors is guided by the intrinsic spatial architecture of the target biological signal (e.g., "sharp peak" vs. "broad domain"), a dichotomy firmly established by foundational epigenomic mapping efforts.[24] Following classic computational methodologies that apply specific parameters for divergent signal topologies,[25] this prior-informed alignment ensures that tokenization granularity matches the functional scale of interest, effectively constraining the optimization within biologically plausible ranges and minimizing the risk of overfitting compared to exhaustive, agnostic searches. For instance, a combination of higher density and higher clustering optimally sharpens the tokenization for precise peak localization, whereas lower density with lower clustering better accommodates the diffuse nature of broad epigenetic domains.

We conducted grid search experiments with parameter combinations for eight histone modification tasks and selected two representative cases for detailed analysis. The overall results are shown in Figure 5.

The bar chart on the left side of Figure 5A demonstrated a consistent and significant improvement in the MCC scores following model refinement, as evidenced by the taller bars in the "refinement" category compared with the original "GenART(350M)" baseline. This marked enhancement highlighted the efficacy of tailoring token length and distribution to align with the distinct characteristics of histone modification signals, thereby improving predictive accuracy.

On the right side of Figure 5A, we observed a clear visualization of the distribution of token density and cluster factor values across two distinct categories of histone modification tasks: sharp peak and broad domain. The sharp peak tasks, denoted by red dots, were predominantly situated in the upper-right quadrant of the plot. This positioning indicated a preference for higher token density and cluster factor values, suggesting that these tasks benefit from a model configuration that generates shorter, more densely packed tokens. This aligns with the need to capture the precise and localized signals characteristic of sharp peak modifications.

In contrast, the broad domain tasks, represented by green dots, were clustered in the lower-left quadrant, favoring lower token density and cluster factor values. This distribution was consistent with the broader and more extended signal patterns of broad domain modifications, which necessitate longer tokens to accurately represent the extended regions of modification. The distinct clustering of these two task types in the parameter space further validated the hypothesis that the optimal tokenization strategy is influenced by the inherent width and distribution of the genomic features being modeled. The clear separation between sharp peak and broad domain tasks underscored the refinement process's ability to effectively tailor the tokenization behavior to the specific requirements of each histone modification task, thereby enhancing overall performance.

Furthermore, we performed exhaustive grid searches for the representative tasks in the sharp peak and broad domain categories, respectively.

In Figure 5B and Table S4, we focused on H3K4me3, a histone modification representative of the sharp peak category. The boxplot in the upper-left panel displayed the distribution of MCC scores after model refinement, yielding a mean value of 0.636—slightly higher than the baseline score of 0.628. The maximum MCC value reached 0.650, underscoring a notable performance enhancement over the original model.

CellPress
OPEN ACCESS

Cell Reports Methods
Article

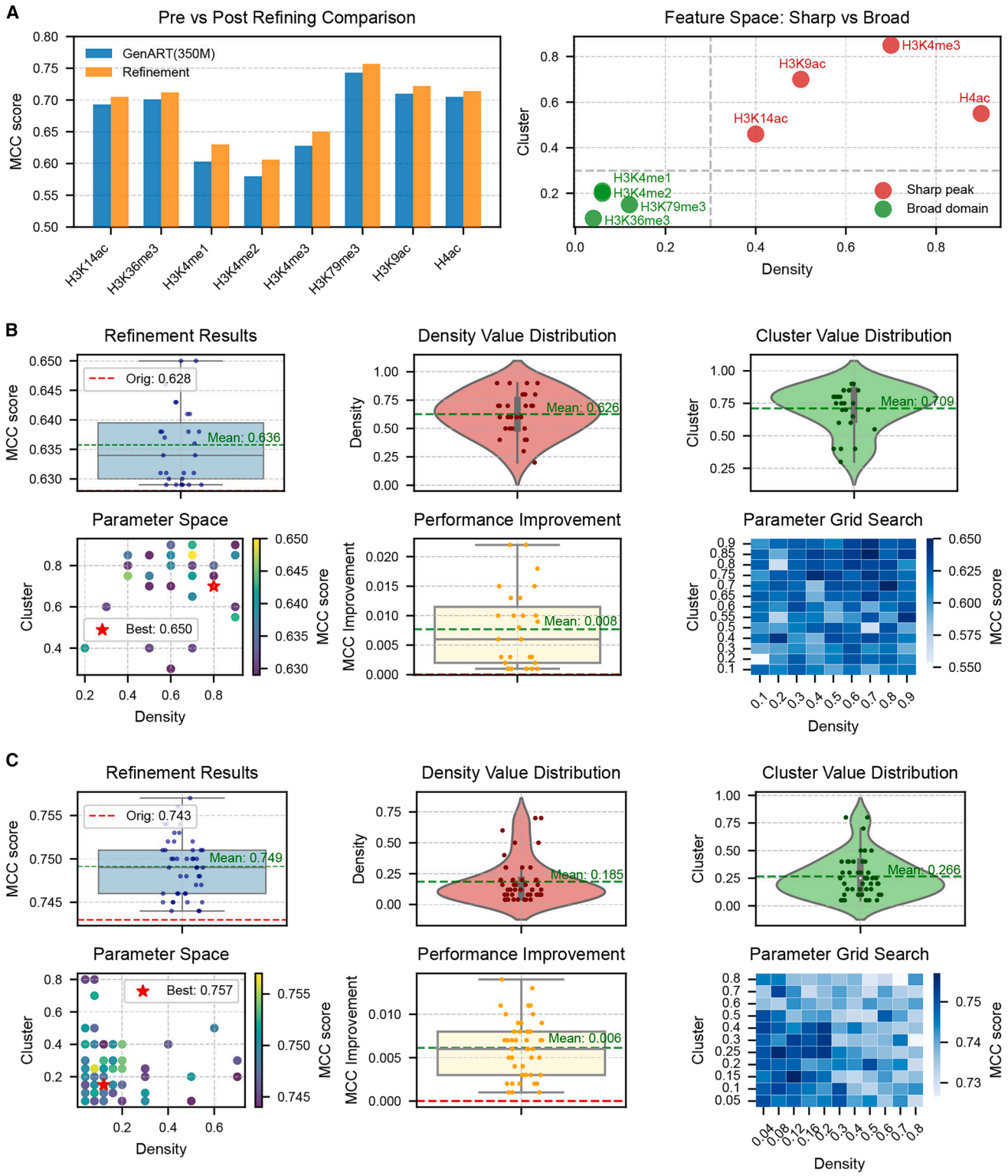


Figure 5. Analysis of token distribution refinement on histone modification prediction tasks

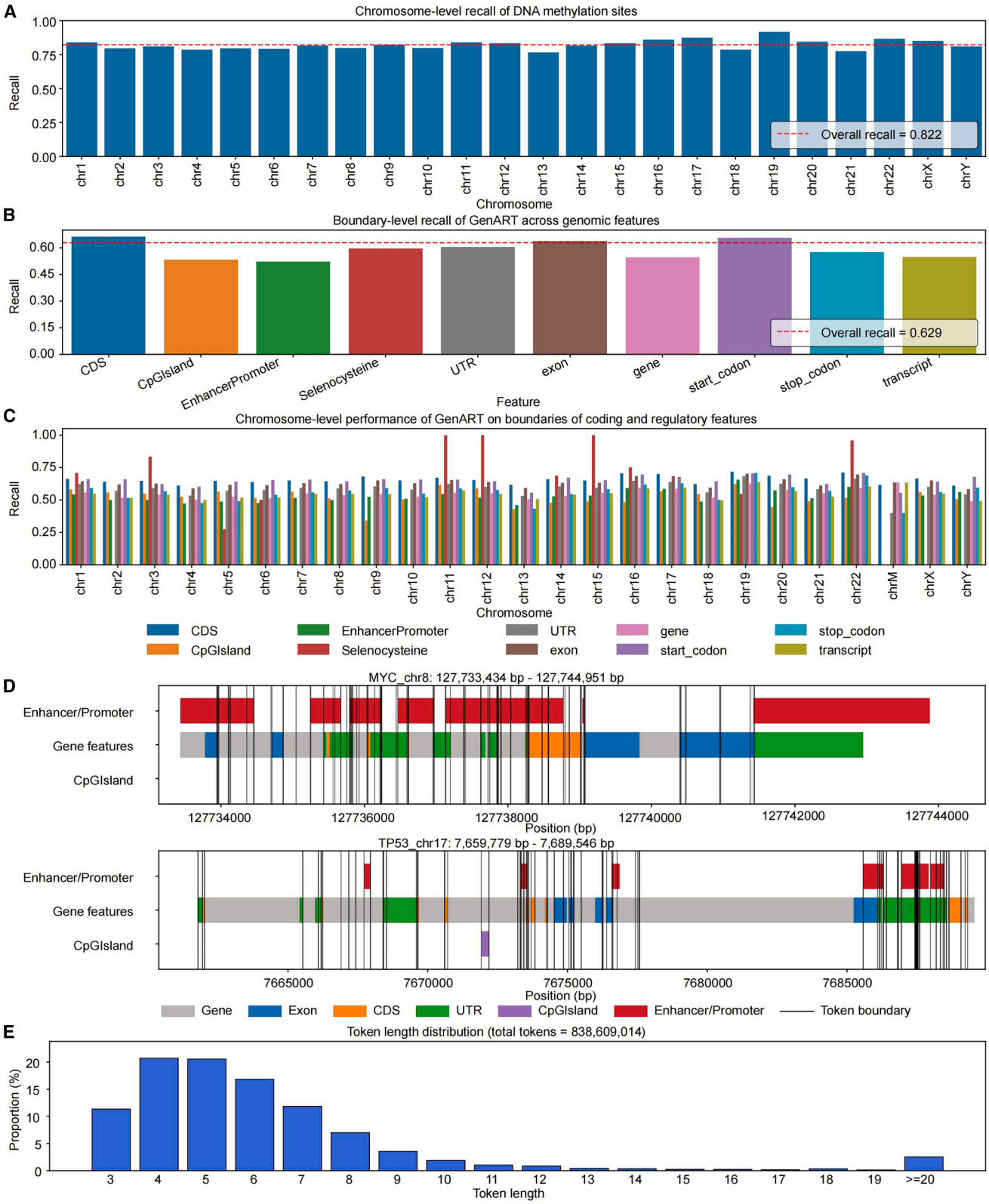
(A) Comprehensive evaluation of model refinement across eight histone modification tasks. The left image presents a comparative analysis of model performance before and after refinement. The right image illustrates the distribution of optimal token density and cluster factor values for both sharp peak and broad domain tasks.

(B) Parameter grid search analysis for the H3K4me3 task (sharp peak).

(C) Parameter grid search analysis for the H3K79me3 task (broad domain).

Please cite this article in press as: Chen et al., GenART: Reading the genome's language with adaptive "words", Cell Reports Methods (2026), https://doi.org/10.1016/j.crmeth.2026.101516

CellPress
OPEN ACCESS



(legend on next page)

The violin plot in the top-middle panel illustrated the distribution of token density values, with a mean of 0.626, while the top-right plot showed the distribution of cluster factors, averaging 0.709. These results suggested that optimal performance for H3K4me3 is associated with higher density and cluster factor values, aligning with the biological characteristic that sharp histone marks require fine-grained, densely clustered tokenization to capture localized signal patterns.

The scatterplot in the lower-left quadrant revealed a positive correlation between parameter settings and MCC scores, with the highest performance occurring in the upper-right region of the parameter space, where the density and clustering factors were in the higher range. The middle-bottom plot presented the distribution of performance improvements, averaging 0.008. Finally, the heatmap in the bottom-right panel confirmed that MCC scores generally improved as parameters shifted toward the upper-right quadrant.

Figure 5C and Table S5 presented the analysis for H3K79me3, a representative broad-domain mark. The boxplot in the top-left panel showed a refined mean MCC of 0.749, compared to the baseline of 0.743, with a maximum value reaching 0.757. These results confirmed the efficacy of the refinement strategy for broad-domain-type modifications.

The middle-top violin plot indicated a mean density value of 0.185, while the top-right plot showed a mean cluster factor of 0.266. These relatively low values suggested that optimal performance for H3K79me3 is achieved with sparser tokenization and wider token boundaries, aligning with the diffuse nature of broad histone modifications.

The scatterplot in the middle-left panel demonstrated that peak performance occurred at a lower parameter space. The bottom-middle plot illustrated the performance improvement distribution, with a mean gain of 0.006, reflecting consistent, albeit modest, enhancement. Finally, the heatmap in the bottom-right panel confirmed that higher MCC scores were attained under lower density and cluster factor settings—a trend opposite to that observed for sharp-peak marks. This divergent behavior across two representative case studies underscored the task-specific adaptability of our refinement approach.

In conclusion, our findings have confirmed that the optimal tokenization strategy is correlated with the biological characteristics of the downstream task. Specifically, sharp peak and broad domain tasks exhibit distinct preferences in terms of token density and token clustering tendencies. This prior-informed optimization demonstrates strong generalizability, providing a principled approach to model diverse functional elements—whether they are narrowly localized motifs like core promoters or extended epigenetic marks. This alignment not only yields consistent performance gains but also transforms the tokenization process into a specialized, biologically interpretable tool for genomic discovery.

Automated genome annotation: Biological applications and insights

GenART's adaptive words have demonstrated effectiveness for AI learning, as evidenced by robust generalization across tasks. We next examined the flexibility of the segmentation and the biological relevance of these words through two representative classes of biological applications that place different constraints on segmentation granularity. Specifically, we applied GenART to segment the reference genome GRCh38.p14 and compared the segmentations with the latest annotation.

For genomic features that do not require a specific word-level pattern—such as DNA methylation and other base-resolution chemical modifications—the main requirement is accurate localization of individual positions rather than structured multi-base motifs. In this setting, we directly used the length-1 tokens output by GenART and compared their genomic coordinates to a curated collection of experimentally profiled methylation sites. 218 cross-platform methylation datasets of human tissues were primarily obtained from the ENCODE,[16,24,26] the Roadmap Epigenomics Project,[24] the EWAS Data Hub,[27,28] and the Gene Expression Omnibus (GEO).[29,30] Across chromosomes, GenART achieved a high recall of ~0.8 for known DNA methylated sites, with per-chromosome values ranging from ~0.77 to ~0.92 (Figure 6A). This indicates that, even before any post-processing, the shortest GenART tokens already provide an efficient and accurate representation for base-resolution functional annotations.

In contrast, many gene- and regulation-related events—such as alternative splicing, coding sequence organization, and *cis*-/*trans*-regulatory elements—are governed by characteristic sequence patterns that span multiple nucleotides. For these applications, we simply merged adjacent bases based on the output logits of the adaptive tokenizer and obtained words of length $\geq$3 (details provided in STAR Methods). We compared them with annotations from GENCODE v.49,[31] which provides comprehensive gene annotation for the GRCh38.p14 reference genome, including protein-coding genes, long non-coding RNAs, pseudogenes, small RNAs, and other functional elements. Additional regulatory annotations—including CpG islands and enhancer/promoter regions—were obtained from the UCSC CpG Island track[32] and the Ensembl Regulatory Build.[33] GenART achieved a genome-wide boundary-level recall of 0.629, accurately recovering 10.27 million boundaries out of 16.32 million annotated positions (Table S6). Recall was highest for coding sequence (CDS) (0.662), start codon (0.657), and exon (0.640) boundaries, demonstrating strong

Figure 6. GenART effectively captures biologically meaningful boundaries across diverse genomic contexts

(A) Chromosome-level recall of known DNA methylation sites.

(B) Boundary-level recall across key genomic and regulatory features.

(C) Chromosome-level performance of GenART in recalling boundaries of coding (CDS, exon, UTR, etc.) and regulatory (CpG island, enhancer/promoter) features.

(D) Visualization of GenART segmentation boundaries aligned with annotated genomic tracks for representative loci (MYC and TP53).

(E) Distribution of effective token lengths after post-processing segmentation logits.

See also Figure S2.

Please cite this article in press as: Chen et al., GenART: Reading the genome's language with adaptive "words", Cell Reports Methods (2026), https://doi.org/10.1016/j.crmeth.2026.101516

alignment between GenART's segmentation patterns and canonical gene structure (Figure 6B). UTR (0.605) and selenocysteine-encoding site (0.596) boundaries were also well captured. Regulatory elements showed moderate but consistent performance—CpG islands (0.534) and enhancer/promoter regions (0.523)—reflecting the greater sequence diversity characteristic of non-coding regulatory landscapes. A chromosome-resolved analysis further revealed stable performance across the genome, with most chromosomes exhibiting recall values between 0.5 and 0.7 across feature categories (Figure 6C). These results indicate that GenART produces biologically coherent segmentation patterns that generalize robustly across genomic contexts. Representative examples of correctly recovered gene structural boundaries and regulatory regions are shown in Figure 6D.

We further compared GenART with both general-purpose genomic language models (DNABERT2, MxDNA, and a fixed-length Nucleotide Transformer, 6-mer) and a task-specific segmentation model (SegmentNT[34]) using boundary-level MCC as a unified metric (Figure S2). On both the curated BUSCO eukaryotic gene set and GRCh38-based human genomic annotations, SegmentNT achieved the highest MCC across most feature categories. GenART consistently outperformed the general-purpose language models DNABERT2, MxDNA, and fixed-length Nucleotide Transformer and showed stable performance across gene structural features as well as regulatory elements. Performance differences across feature types were also observed: while SegmentNT achieved higher MCC on gene structural elements, its advantage was less pronounced on regulatory features such as CpG islands and enhancers, where GenART showed more consistent performance. In contrast to SegmentNT, which inherits the representation capacity of the fixed Nucleotide Transformer with an added segmentation head, GenART recovers biologically meaningful boundaries without explicit supervision and simultaneously achieves robust downstream task performance.

Consistent with these two application regimes, the token length distribution after post-processing the segmentation logits shows a clear peak between 3 and 8 nucleotides (Figure 6E), which closely matches the typical length scale of many biological sequence patterns (e.g., splice motifs, transcription factor binding sites, and short regulatory elements). Together, these results indicate that GenART naturally adapts to both base-resolution and pattern-dependent contexts: length-1 tokens provide precise anchors for chemical modifications such as methylation, while tokens of length $\geq$3 capture higher order genomic patterns underlying gene structure and regulation.

DISCUSSION

In this study, we present GenART, a genomic language model whose core innovation lies in replacing static, fixed-length tokenization with a learnable, adaptive tokenizer. This approach directly addresses a fundamental bottleneck in genomic understanding: the lack of inherent grammatical units in DNA sequence. Our comprehensive evaluation demonstrates that this adaptive capability translates into state-of-the-art performance across a diverse array of genomic prediction tasks, underscoring the critical importance of biologically informed sequence segmentation.

The power of GenART is not merely a result of increased model scale or dataset size, though our scaling experiments confirm that these factors contribute to performance. Rather, the key driver is the ability of the model to learn how to parse the genome. Our analyses reveal that the learned tokenizer segments sequences in close alignment with reference gene architecture, despite the absence of explicit supervision. Enrichment of predicted boundaries around CDS, exon, and start codon regions suggests that the model internalizes conserved genomic grammar directly from DNA sequences. Regulatory features, while inherently less positionally constrained, also exhibit consistent boundary coverage, indicating that GenART detects meaningful sequence cues across broad regulatory regions. Moreover, GenART generates token representations and distributions that are broadly discriminative for downstream biological tasks without task-specific fine-tuning. Together, these results demonstrate that GenART learns a computationally efficient and biologically grounded "genomic grammar," enabling downstream genomic interpretation tasks. Comparison with existing models further highlights the distinction between adaptive tokenization and task-specific modeling strategies. SegmentNT, which is explicitly trained for genomic segmentation, produces fewer but higher confidence boundaries and achieves higher MCC due to direct supervision on boundary prediction. In contrast, DNABERT2 and MxDNA are general-purpose DNA language models that learn sequence representations without explicit optimization for boundary detection. GenART differs from both paradigms. Its segmentation is not directly supervised by annotated genomic features but instead emerges from learning an adaptive representation. Consistent with this design, GenART shows stable performance across feature types, including both gene structural and regulatory elements. These observations position GenART between task-specific segmentation models and general-purpose language models, providing a complementary perspective that emphasizes comprehensive sequence organization rather than optimized boundary classification.

A particularly powerful extension of this adaptive capability is our proposed refining methodology. By guiding the tokenizer with two intuitive parameters—token density and cluster factor—we can explicitly tailor the model's inductive bias to the spatial scale of the target biological signal. The clear separation we observed between the optimal parameter settings for sharp-peak marks (e.g., H3K4me3) and broad-domain marks (e.g., H3K79me3) provides strong evidence that task-aware tokenization is not only possible but highly beneficial. This refining process offers a paradigm for model specialization, bridging the gap between a general-purpose genomic language model and highly accurate task-specific predictors controlled by domain experts, without architectural changes.

Limitations of the study

While GenART demonstrates robust performance and adaptive flexibility across diverse genomic tasks, our study has several limitations. First, although the adaptive tokenizer successfully delineates known genomic boundaries (e.g., CDS, exons, and DNA methylation sites), the specific biological functions of

many *de novo* "words" dynamically identified by the model remain uncharacterized and require further experimental validation. Second, while we evaluated GenART on sequence contexts up to 10 kb for splice site prediction, modeling higher order genomic topologies—such as megabase-scale 3D chromatin interactions or large-scale structural variants—remains constrained by the current maximum sequence length of the underlying Transformer architecture. Third, in our automated genome annotation application, we employed a straightforward heuristic to merge adjacent bases based on tokenizer logits to extract multi-base structural features; future iterations could explore more sophisticated decoding strategies to better capture highly complex or discontinuous regulatory motifs. Finally, GenART prioritizes a biologically contextualized representation space optimized for downstream fine-tuning, while sacrificing naive zero-shot capabilities (e.g., near random) on zero-shot variant effect prediction (VEP). The limitation arises from GenART's dynamic tokenization mechanism. For example, fixed token boundaries maintain perfect sequence alignment, enabling direct spatial distance comparisons between the reference and mutated embeddings. Conversely, in GenART, a single point mutation triggers a localized frameshift in the dynamic chunk boundaries. Consequently, calculating distance metrics between these structurally misaligned global embeddings introduces significant computational noise, thereby obscuring the true biological mutation signal in a zero-shot setting.

RESOURCE AVAILABILITY

Lead contact

Requests for further information and resources should be directed to and will be fulfilled by the lead contact, Chen Lin (cheyenne.lin@foxmail.com).

Materials availability

This study did not generate new unique reagents.

Data and code availability

- This paper analyzes existing, publicly available data. These accession numbers for the datasets are listed in the key resources table.
- The original code of GenART has been deposited at https://github.com/XMUDM/GenART, https://doi.org/10.5281/zenodo.20529925.
- The checkpoints of the pretrained GenART models implemented in PyTorch can be found at https://github.com/XMUDM/GenART/tree/main/pretrained_model, https://doi.org/10.5281/zenodo.20530072.
- Any additional information required to reanalyze the data reported in this paper is available from the lead contact upon request.

ACKNOWLEDGMENTS

This research is supported by the Natural Science Foundation of China (no. 62432011), the Fujian Provincial Natural Science Foundation of China (no. 2025J010001), and the Zhongguancun Academy Project (no. 02012410).

AUTHOR CONTRIBUTIONS

C.L. conceptualized the project. C.L. and K.C. designed the GenART model. C.L. and P.H. designed the refining algorithm, and K.C. and Z.L. designed the automated genome annotation experiment. K.C., P.H., Z.L., S.G., and X.Y. collected and processed the data. K.C., P.H., and Z.L. validated the experimental results. C.L., X.Z., Y.G., and Z.J. provided guidance on algorithm implementation, data collection, model training, and result analysis. K.C., P.H., and Z.L. drafted the initial manuscript. C.L. and Z.J. revised and edited the manuscript. C.L. acquired the funding.

DECLARATION OF INTERESTS

The authors declare no competing interests.

DECLARATION OF GENERATIVE AI AND AI-ASSISTED TECHNOLOGIES IN THE WRITING PROCESS

During the preparation of this work, the authors used Gemini 3.1 and ChatGPT-5 to refine the language and improve the readability of the manuscript. After using these tools, the authors reviewed and edited the content as needed and take full responsibility for the content of the publication.

STAR★METHODS

Detailed methods are provided in the online version of this paper and include the following:

SUPPLEMENTAL INFORMATION

Supplemental information can be found online at https://doi.org/10.1016/j.crmeth.2026.101516.

Received: December 12, 2025
Revised: April 23, 2026
Accepted: June 10, 2026

REFERENCES

1. Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), J. Burstein, C. Doran, and T. Solorio, eds. (Association for Computational Linguistics), pp. 4171–4186. https://doi.org/10.18653/v1/N19-1423.
2. Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J.D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. (2020). Language Models are Few-Shot Learners. Adv. Neural Inf. Process. Syst. *33*, 1877–1901.
3. Ji, Y., Zhou, Z., Liu, H., and Davuluri, R.V. (2021). DNABERT: pre-trained Bidirectional Encoder Representations from Transformers model for DNA-language in genome. Bioinformatics *37*, 2112–2120. https://doi.org/10.1093/bioinformatics/btab083.
4. Avsec, Ž., Agarwal, V., Visentin, D., Ledsam, J.R., Grabska-Barwinska, A., Taylor, K.R., Assael, Y., Jumper, J., Kohli, P., and Kelley, D.R. (2021). Effective gene expression prediction from sequence by integrating long-range

interactions. Nat. Methods *18*, 1196–1203. https://doi.org/10.1038/s41592-021-01252-x.

5. Nguyen, E., Poli, M., Faizi, M., Thomas, A., Wornow, M., Birch-Sykes, C., Massaroli, S., Patel, A., Rabideau, C., Bengio, Y., et al. (2023). HyenaDNA: Long-Range Genomic Sequence Modeling at Single Nucleotide Resolution. Adv. Neural Inf. Process. Syst. *36*, 43177–43201.
6. Zhou, Z., Ji, Y., Li, W., Dutta, P., Davuluri, R., and Liu, H. (2024). DNABERT-2: Efficient Foundation Model and Benchmark For Multi-Species Genomes. Int. Conf. Learn. Represent. *2024*, 41642–41665.
7. Dalla-Torre, H., Gonzalez, L., Mendoza-Revilla, J., Lopez Carranza, N., Grzywaczewski, A.H., Oteri, F., Dallago, C., Trop, E., de Almeida, B.P., Sirelkhatim, H., et al. (2025). Nucleotide Transformer: building and evaluating robust foundation models for human genomics. Nat. Methods *22*, 287–297. https://doi.org/10.1038/s41592-024-02523-z.
8. Quang, D., and Xie, X. (2016). DanQ: a hybrid convolutional and recurrent deep neural network for quantifying the function of DNA sequences. Nucleic Acids Res. *44*, e107. https://doi.org/10.1093/nar/gkw226.
9. Sennrich, R., Haddow, B., and Birch, A. (2016). Neural Machine Translation of Rare Words with Subword Units. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), K. Erk and N.A. Smith, eds. (Association for Computational Linguistics), pp. 1715–1725. https://doi.org/10.18653/v1/P16-1162.
10. Bai, W., Dong, N., Liang, C., Ma, X., Ouyang, W., Qiao, L., Ren, Y., and Ye, P. (2024). Model Decides How to Tokenize: Adaptive DNA Sequence Tokenization with MxDNA. Adv. Neural Inf. Process. Syst. *37*, 66080–66107. https://doi.org/10.52202/079017-2112.
11. Hwang, S., Wang, B., and Gu, A. (2025). Dynamic Chunking for End-to-End Hierarchical Sequence Modeling. Preprint at arXiv. https://doi.org/10.48550/arXiv.2507.07955.
12. Kelley, D.R., Snoek, J., and Rinn, J.L. (2016). Basset: learning the regulatory code of the accessible genome with deep convolutional neural networks. Genome Res. *26*, 990–999. https://doi.org/10.1101/gr.200535.115.
13. Lee, D., Kim, C., Kim, S., Cho, M., and Han, W.-S. (2022). Autoregressive Image Generation using Residual Quantization. In 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (IEEE)), pp. 11513–11522. https://doi.org/10.1109/CVPR52688.2022.01123.
14. Brixi, G., Durrant, M.G., Ku, J., Naghipourfar, M., Poli, M., Sun, G., Brockman, G., Chang, D., Fanton, A., Gonzalez, G.A., et al. (2026). Genome modelling and design across all domains of life with Evo 2. Nature *652*, 1349–1361. https://doi.org/10.1038/s41586-026-10176-5.
15. Dunham, I., Kundaje, A., Aldred, S.F., Collins, P.J., Davis, C.A., Doyle, F., Epstein, C.B., Frietze, S., Harrow, J., Kaul, R., et al. (2012). An integrated encyclopedia of DNA elements in the human genome. Nature *489*, 57–74. https://doi.org/10.1038/nature11247.
16. Moore, J.E., Purcaro, M.J., Pratt, H.E., Epstein, C.B., Shoresh, N., Adrian, J., Kawli, T., Davis, C.A., Dobin, A., Kaul, R., et al. (2020). Expanded encyclopaedias of DNA elements in the human and mouse genomes. Nature *583*, 699–710. https://doi.org/10.1038/s41586-020-2493-4.
17. Meuleman, W., Muratov, A., Rynes, E., Halow, J., Lee, K., Bates, D., Diegel, M., Dunn, D., Neri, F., Teodosiadis, A., et al. (2020). Index and biological spectrum of human DNase I hypersensitive sites. Nature *584*, 244–251. https://doi.org/10.1038/s41586-020-2559-3.
18. Meylan, P., Dreos, R., Ambrosini, G., Groux, R., and Bucher, P. (2019). EPD in 2020: enhanced data visualization and extension to ncRNA promoters. Nucleic Acids Res. *48*, gkz1014. https://doi.org/10.1093/nar/gkz1014.
19. Harrow, J., Frankish, A., Gonzalez, J.M., Tapanari, E., Diekhans, M., Kokocinski, F., Aken, B.L., Barrell, D., Zadissa, A., Searle, S., et al. (2012). GENCODE: The reference human genome annotation for The ENCODE Project. Genome Res. *22*, 1760–1774. https://doi.org/10.1101/gr.135350.111.
20. Jaganathan, K., Kyriazopoulou Panagiotopoulou, S., McRae, J.F., Darbandi, S.F., Knowles, D., Li, Y.I., Kosmicki, J.A., Arbelaez, J., Cui, W., Schwartz, G.B., et al. (2019). Predicting Splicing from Primary Sequence with Deep Learning. Cell *176*, 535–548.e24. https://doi.org/10.1016/j.cell.2018.12.015.
21. Chen, L., and Capra, J.A. (2020). Learning and interpreting the gene regulatory grammar in a deep learning framework. PLoS Comput. Biol. *16*, e1008334. https://doi.org/10.1371/journal.pcbi.1008334.
22. Sullivan, P.J., Quinn, J.M.W., Ajuyah, P., Pinese, M., Davis, R.L., and Cowley, M.J. (2025). Data-driven insights to inform splice-altering variant assessment. Am. J. Hum. Genet. *112*, 764–778. https://doi.org/10.1016/j.ajhg.2025.02.012.
23. Boumpas, P., Merabet, S., and Carnesecchi, J. (2023). Integrating transcription and splicing into cell fate: Transcription factors on the block. WIREs RNA *14*, e1752. https://doi.org/10.1002/wrna.1752.
24. Kundaje, A., Meuleman, W., Ernst, J., Bilenky, M., Yen, A., Heravi-Moussavi, A., Kheradpour, P., Zhang, Z., Wang, J., Ziller, M.J., et al. (2015). Integrative analysis of 111 reference human epigenomes. Nature *518*, 317–330. https://doi.org/10.1038/nature14248.
25. Zhang, Y., Liu, T., Meyer, C.A., Eeckhoute, J., Johnson, D.S., Bernstein, B.E., Nusbaum, C., Myers, R.M., Brown, M., Li, W., and Liu, X.S. (2008). Model-based Analysis of ChIP-Seq (MACS). Genome Biol. *9*, R137. https://doi.org/10.1186/gb-2008-9-9-r137.
26. Luo, Y., Hitz, B.C., Gabdank, I., Hilton, J.A., Kagda, M.S., Lam, B., Myers, Z., Sud, P., Jou, J., Lin, K., et al. (2020). New developments on the Encyclopedia of DNA Elements (ENCODE) data portal. Nucleic Acids Res. *48*, D882–D889. https://doi.org/10.1093/nar/gkz1062.
27. Xiong, Z., Li, M., Yang, F., Ma, Y., Sang, J., Li, R., Li, Z., Zhang, Z., and Bao, Y. (2020). EWAS Data Hub: a resource of DNA methylation array data and metadata. Nucleic Acids Res. *48*, D890–D895. https://doi.org/10.1093/nar/gkz840.
28. Xiong, Z., Yang, F., Li, M., Ma, Y., Zhao, W., Wang, G., Li, Z., Zheng, X., Zou, D., Zong, W., et al. (2022). EWAS Open Platform: integrated data, knowledge and toolkit for epigenome-wide association study. Nucleic Acids Res. *50*, D1004–D1009. https://doi.org/10.1093/nar/gkab972.
29. Barrett, T., Wilhite, S.E., Ledoux, P., Evangelista, C., Kim, I.F., Tomashevsky, M., Marshall, K.A., Phillippy, K.H., Sherman, P.M., Holko, M., et al. (2013). NCBI GEO: archive for functional genomics data sets—update. Nucleic Acids Res. *41*, D991–D995. https://doi.org/10.1093/nar/gks1193.
30. Clough, E., Barrett, T., Wilhite, S.E., Ledoux, P., Evangelista, C., Kim, I.F., Tomashevsky, M., Marshall, K.A., Phillippy, K.H., Sherman, P.M., et al. (2024). NCBI GEO: archive for gene expression and epigenomics data sets: 23-year update. Nucleic Acids Res. *52*, D138–D144. https://doi.org/10.1093/nar/gkad965.
31. Frankish, A., Carbonell-Sala, S., Diekhans, M., Jungreis, I., Loveland, J.E., Mudge, J.M., Sisu, C., Wright, J.C., Arnan, C., Barnes, I., et al. (2023). GENCODE: reference annotation for the human and mouse genomes in 2023. Nucleic Acids Res. *51*, D942–D949. https://doi.org/10.1093/nar/gkac1071.
32. Kent, W.J., Sugnet, C.W., Furey, T.S., Roskin, K.M., Pringle, T.H., Zahler, A.M., and Haussler, a.D. (2002). The Human Genome Browser at UCSC. Genome Res. *12*, 996–1006. https://doi.org/10.1101/gr.229102.
33. Zerbino, D.R., Wilder, S.P., Johnson, N., Juettemann, T., and Flicek, P.R. (2015). The Ensembl Regulatory Build. Genome Biol. *16*, 56. https://doi.org/10.1186/s13059-015-0621-5.
34. de Almeida, B.P., Dalla-Torre, H., Richard, G., Blum, C., Hexemer, L., Gélard, M., Mendoza-Revilla, J., Tang, Z., Marin, F.I., Emms, D.M., et al. (2025). Annotating the genome at single-nucleotide resolution with DNA foundation models. Nat. Methods *22*, 2301–2315. https://doi.org/10.1038/s41592-025-02881-2.
35. Dao, T., Fu, D., Ermon, S., Rudra, A., and Ré, C. (2022). FlashAttention: Fast and Memory-Efficient Exact Attention with IO-Awareness. Adv. Neural Inf. Process. Syst. *35*, 16344–16359.

Please cite this article in press as: Chen et al., GenART: Reading the genome's language with adaptive "words", Cell Reports Methods (2026), https://doi.org/10.1016/j.crmeth.2026.101516

STAR★METHODS

KEY RESOURCES TABLE

REAGENT or RESOURCE	SOURCE	IDENTIFIER
Deposited data		
The GUE benchmark	Zhou et al.[6]	https://huggingface.co/datasets/leannmlindsey/GUE
The NT benchmark	Dalla-Torre et al.[7]	https://huggingface.co/datasets/InstaDeepAI/nucleotide_transformer_downstream_tasks
BUSCO eukaryotic lineage dataset	BUSCO	busco:eukaryota_odb12.2025-07-01; https://busco.ezlab.org
GENCODE v49 annotation	GENCODE	https://www.gencodegenes.org
CpG island annotations	UCSC Genome Browser	https://genome.ucsc.edu
Regulatory annotations	Ensembl Regulatory Build	https://www.ensembl.org
DNA methylation resources	ENCODE/GEO/Roadmap/EWAS	Table S7
Software and algorithms		
H-Net	Hwang et al.[11]	https://github.com/goombalab/hnet
DNABERT	Ji et al.[3]	https://github.com/jerryji1993/DNABERT
DNABERT-2	Zhou et al.[6]	https://github.com/MAGICS-LAB/DNABERT_2
nucleotide-transformer	Dalla-Torre et al.[7]	https://github.com/instadeepai/nucleotide-transformer
HyenaDNA	Nguyen et al.[5]	https://github.com/HazyResearch/hyena-dna
Evo 2	Brixi et al.[14]	https://github.com/arcinstitute/evo2
MxDNA	Qiao et al.[10]	https://github.com/qiaoqiaoLF/MxDNA
SpliceAI	Jaganathan et al.[20]	https://github.com/Illumina/SpliceAI
SegmentNT	de Almeida et al.[34]	https://github.com/instadeepai/nucleotide-transformer
GenART	This paper	https://github.com/XMUDM/GenART, https://doi.org/10.5281/zenodo.20529925

METHOD DETAILS

Models

Our proposed model, GenART, is designed to process DNA sequences through a hierarchical architecture that transitions from nucleotide-level representations to dynamically learned, variable-length token-level representations. GenART is composed of an encoder-only transformer architecture interspersed with our proposed adaptive tokenization.

The overall architecture of GenART begins by processing an input DNA sequence at the individual nucleotide level. Raw DNA sequences are first processed by a pre-trained 1-mer tokenizer. This tokenizer inserts spaces between each character, effectively treating each nucleotide as a distinct unit. It then prepends a [CLS] token and appends a [SEP] token to frame the sequence. The vocabulary includes standard nucleotides, these special tokens, and a [MASK] token for the pre-training task. Each processed sequence is padded or truncated to a fixed maximum length, as specified in our experimental setup. For each sequence, this process generates three key tensors.

- Input IDs: A tensor of integer indices representing the tokenized sequence.
- Attention Mask: A binary mask that distinguishes real tokens from padding tokens, preventing the model from attending to padded positions.
- Special Tokens Mask: A binary mask that identifies the [CLS] and [SEP] tokens, used to exclude them from the masking procedure during pre-training.

Please cite this article in press as: Chen et al., GenART: Reading the genome's language with adaptive "words", Cell Reports Methods (2026), https://doi.org/10.1016/j.crmeth.2026.101516

These tensors are organized into batches using a data loader with a random sampler for training to ensure data is shuffled between epochs.

The input IDs are passed to an embedding layer, which maps each token index to a dense vector representation. The dimensionality of these vectors is referred to as the hidden size of the model. The padding index is configured to be mapped to a zero vector.

After tokenization and embedding, the sequence is fed into a stack of Transformer-based nucleotide encoder layers. These layers are responsible for capturing local and long-range contextual dependencies among individual bases, producing a context-aware representation for each nucleotide in the sequence. Each encoder layer is a standard Transformer block composed of.

- A pre-layer normalization step.
- A multi-head self-attention mechanism utilizing Flash Attention[35] for computational efficiency and implementing Grouped-Query Attention (GQA).
- A residual connection followed by a second pre-layer normalization step.
- A position-wise feedforward network with a GELU activation function.
- A final residual connection.

Rotary Position Embeddings (RoPE) are applied to the query and key projections within the attention mechanism to inject relative positional information.

Following this initial encoding, the sequence of nucleotide embeddings is passed to the core innovation of our model: the learned tokenizer layer. This layer dynamically segments the nucleotide sequence into a shorter sequence of variable-length tokens. The resulting aggregated token embeddings represent higher-level, more abstract features of the sequence.

Finally, this new sequence of token embeddings is processed by another stack of Transformer-based token encoder layers. Operating at this higher level of abstraction allows the model to reason about the relationships and interactions between these learned functional units, producing the final, deeply contextualized representation of the input DNA sequence. This hierarchical design enables the model to first understand the "letters" of the genome and then learn to compose them into meaningful "words".

The adaptive tokenizer layer

This layer transforms the nucleotide-level hidden states into a shorter sequence of token-level hidden states. Its implementation and hyperparameters are detailed in Table S8.

The process begins by further enriching the contextual information of each nucleotide hidden state, denoted as $H_{nuc} \in R^{B \times N \times D}$, where B is the batch size, N is the sequence length, and D is the hidden dimension. This is achieved using a set of five parallel one-dimensional Convolutional Neural Networks (CNNs). Each CNN has a kernel size k ranging from 2 to 6, with input and output channels equal to the hidden dimension D. To handle sequence boundaries, we apply asymmetric padding manually before the convolution, with $(\lfloor (k\text{-}1)/2 \rfloor, \lceil (k\text{-}1)/2 \rceil)$ padding on the left and right, respectively. The outputs from these five CNNs, $6^{k=2}_{\{C_k(H_{nuc})\}}$, are then averaged to produce a multi-scale local context representation, H_{cnn}:

$$H_{cnn} = \frac{1}{5} \sum_{k=2}^{6} C_k(H_{nuc}). \quad \text{(Equation 2)}$$

To ensure stable training and preserve original information, we apply a residual connection, adding the original hidden states back to the CNN output: $H_{context} = H_{nuc} + H_{cnn}$. This final contextualized representation, $H_{context}$, is then passed through a LayerNorm operation.

This normalized representation is fed into a two-layer Multi-Layer Perceptron (MLP) to compute merge scores. Specifically, the MLP first projects the input from dimension D to $2D$ with a GELU activation, followed by a second projection from $2D$ back to D, also with a GELU activation. A final linear layer then maps the representation to a single scalar value, yielding a score s_i for each nucleotide i. To determine whether to merge the boundary between nucleotides i and $i+1$, we sum their scores: $s'_i = s_i + s_{i+1}$. These boundary scores are subsequently converted into two-dimensional logits, $[-s'_i, s'_i]$, corresponding to the decision between "split" (start a new token) and "merge".

To enable discrete yet differentiable decisions, we adopt the Gumbel–Softmax method with a temperature parameter $\tau = 1.0$. This facilitates sampling a one-hot vector from the categorical distribution defined by the logits, thereby producing a binary sequence of "merge" (1) or "split" (0) decisions. These decisions specify the segmentation of the nucleotide sequence.

Next, the hidden states of all nucleotides within each segment are aggregated. In our implementation, we apply scatter-averaging: the hidden states within a segment are summed and normalized by the segment length (i.e., the number of nucleotides). This produces a new sequence of variable-length token embeddings, $H_{token} \in R^{B \times M \times D}$, where $M < N$ denotes the reduced sequence length.

The resulting token embeddings are then processed by a second stack of Transformer encoder layers. The number of these layers equals the total number of hidden layers minus those used in the nucleotide encoder and the tokenization layer. This encoder follows the same architecture as the nucleotide encoder but operates on the aggregated, variable-length token representations. Finally, a terminal layer normalization step is applied to yield the final hidden states of the model.

Please cite this article in press as: Chen et al., GenART: Reading the genome's language with adaptive "words", Cell Reports Methods (2026), https://doi.org/10.1016/j.crmeth.2026.101516

Pre-training

To learn general-purpose representations of the genomic language, we pre-train GenART on a large corpus of unlabeled DNA sequences using a Masked Language Modeling (MLM) objective. In this self-supervised task, a certain fraction of the nucleotides is randomly masked, and the model is trained to predict the original nucleotides based on the surrounding context. We follow the standard BERT strategy, where masked positions are replaced with a [MASK] token 80% of the time, a random nucleotide 10% of the time, and left unchanged for the remaining 10%.

A crucial architectural component for this task is a decoder module, implemented as a cross-attention layer. Since the final model output is at the token level, while the prediction target is at the nucleotide level, this decoder is essential for bridging the two representations. It takes the nucleotide-level hidden states (from before the tokenization layer) as the query and the final token-level hidden states as the key and value. The output of the decoder is then passed to a prediction head. This head consists of a dense layer, a GELU activation, a layer normalization step, and a final linear layer that projects to the vocabulary size, producing logits for each position in the sequence. This allows the model to project the rich, high-level context from the learned tokens back onto each nucleotide position to make an accurate prediction.

The MLM loss is calculated using a cross-entropy loss function, which is configured to ignore non-masked positions. The total pre-training loss is a weighted sum of the MLM cross-entropy loss and the auxiliary length loss from the tokenization layer. This joint optimization ensures that the model simultaneously learns to understand genomic context and to develop an effective tokenization strategy for representing that context.

The model is trained using the AdamW optimizer with mixed-precision training. A cosine learning rate scheduler with a linear warmup phase is employed. The specific pre-training parameters are listed in Table S9.

Fine-tuning

After pre-training, the model is adapted for specific downstream biological tasks through fine-tuning on labeled datasets. For sequence classification tasks, the pre-training MLM head is replaced with a classification head. This new head typically takes the final hidden state of the special [CLS] token from the token encoder, passes it through a small MLP, and outputs logits for the target classes.

Critically, during fine-tuning, the entire model, including the learned tokenization layer, remains trainable. The total loss for fine-tuning is composed of the task-specific classification loss (e.g., cross-entropy) and the auxiliary length loss. This continued optimization of the tokenization mechanism allows the model to adapt its segmentation strategy to what is most discriminative for the specific downstream task. This dynamic, task-aware tokenization is a key advantage of our approach, enabling the model to refine its understanding of genomic "words" to best suit the context of the biological problem at hand.

The optimizer and training setup are similar to pre-training, but with hyperparameters adjusted for the fine-tuning regime. For both benchmarks, we keep the learning rate and the number of training epochs consistent with previous works.[6,7] Detailed parameters are provided in Table S10.

Refining

Our approach introduces a specialized regularization term, the length loss L_{len}, into the total loss function during refining. This term penalizes the deviation of the model's learned token boundary distribution from a given ideal pattern for a specific task. The total loss is formulated as:

$$L_{total} = L_{task} + \lambda \cdot L_{len}, \quad \text{(Equation 3)}$$

where L_{task} is the primary task loss, and λ is a balancing coefficient (set to 0.1 in our experiments). The computation of L_{len} is the core of our adaptive mechanism and is controlled by two key hyperparameters.

The first level of control is achieved through a target token density parameter (ρ). This hyperparameter defines the desired average number of token boundaries per nucleotide. An ideal average token length (L_{avg}) is inversely related to the density: $L_{avg} = \frac{1}{\rho}$. For instance, a target token density of 0.25 guides the model toward an average token length of 4 bp.

In its basic form, L_{len} calculates the Mean Squared Error (MSE) between the predicted boundary distribution of the model and an ideal uniform distribution. The ideal pattern, P_{ideal}, is a vector where boundaries are placed at regular intervals determined by the target density. For a single sample, the loss is:

$$L_{len} = MSE\left(\frac{P_{model}}{\|P_{model}\|_1}, \frac{P_{ideal}}{\|P_{ideal}\|_1}\right), \quad \text{(Equation 4)}$$

where P_{model} is the binary vector of predicted token boundaries, and the division by the L1 norm normalizes the distributions. This loss encourages the model to match the overall frequency of token boundaries specified by the user.

While controlling density is powerful, many biological signals, such as transcription factor binding sites (TFBSs) in enhancers, are not uniformly distributed but appear in clusters. To model this, we introduced a more sophisticated mechanism controlled by a target cluster factor parameter (c_f). This enhancement replaces the uniform ideal pattern with a stochastically generated clustered pattern.

The algorithm to generate this pattern is as follows.

Please cite this article in press as: Chen et al., GenART: Reading the genome's language with adaptive "words", Cell Reports Methods (2026), https://doi.org/10.1016/j.crmeth.2026.101516

- A vector of random noise is generated from a standard normal distribution.
- This noise is then smoothed by convolving it with a 1D Gaussian kernel. The standard deviation (σ) of this kernel is controlled by the target cluster factor ($c_f \in (0,1]$). A smaller c_f results in a larger σ, leading to stronger smoothing and more pronounced clustering of high-value regions in the noise vector. A c_f of 1.0 corresponds to minimal smoothing, approximating a uniform random distribution.
- Finally, a threshold is determined based on the target token density (ρ) to binarize the smoothed noise, producing an ideal boundary pattern $P_{ideals,\ clustered}$ that exhibits the desired density and spatial clustering.

The L_{len} is then calculated as the MSE between the predicted boundary distribution of the model and this new, more biologically plausible, clustered ideal pattern.

Nucleotide Transformer (NT) benchmark

The NT benchmark is a rigorously curated suite of 18 human-centric classification tasks that quantify the ability of a model to predict sequence-based molecular phenotypes.[7] The compendium encompasses four functional themes: (i) histone-modification detection (10 ENCODE ChIP-chip assays for H3, H4 and 8 acetylation/methylation marks), (ii) enhancer prediction (strong/weak enhancer versus non-enhancer), (iii) promoter classification (TATA-box and non-TATA-box promoters versus distal intergenic sequence) and (iv) splice-site annotation (donor, acceptor and non-splice 400-bp windows from GENCODE). Input sequences range from 200 to 600 bp and are experimentally labeled by ChIP-seq, DNase-seq or manual curation.

To ensure a fair and rigorous comparison with literature baselines, we strictly adhered to the original NT evaluation protocol. Specifically, model hyperparameter optimization was conducted exclusively via a 10-fold cross-validation strategy (90% training, 10% validation) within the training sets, thereby preventing any data leakage. Final model performance was subsequently evaluated on the fixed, non-overlapping, and standardized held-out test sets. Furthermore, we aligned our evaluation with the original benchmark's hierarchical metric preference, reporting Matthews Correlation Coefficient (MCC) for histone-modification and enhancer tasks, accuracy (Acc) for the multi-class splice-site task, and F1-score for the remaining promoter predictions. By demanding base-level discrimination of regulatory motifs, the benchmark directly probes a model's capacity to learn the *cis*-regulatory syntax of the non-coding genome.

Genome Understanding Evaluation (GUE) benchmark

Genome Understanding Evaluation (GUE) is a rigorously standardized benchmark that assesses the generalization capacity of genome-scale language models across 36 binary- or multi-class datasets spanning nine core regulatory tasks and four species (human, mouse, yeast, virus).[6] Sequences range from 70 bp (core-promoter detection) to 1000 bp (virus variant classification), forcing models to capture both local motifs and kilobase-long dependencies. Tasks include splice-site, transcription-factor-binding-site, promoter and enhancer prediction, epigenetic-mark classification, and virus variant classification. Each dataset is balanced for moderate difficulty, supplied with fixed train/dev/test splits, and evaluated with Matthews correlation coefficient (MCC) or F1-score.

In this study, we selected 15 tasks from the GUE benchmark that are non-overlapping with the NT benchmark—covering transcription-factor prediction, core-promoter detection, virus variant classification, and splice-site prediction—to extend the biological scope of evaluation. To ensure a strictly fair and meaningful comparison with existing literature, we adhered entirely to the standardized GUE evaluation protocol. Specifically, we utilized the predefined, fixed train/development/test splits to guarantee that our model and all baselines were trained and evaluated on identical data instances. To prevent dataset-specific overfitting and assess true generalization, core hyperparameters for the baselines were kept constant across all tasks, with only maximum sequence length and training steps adjusted per task requirements. Furthermore, we utilized a rigorous early-stopping validation strategy: models were evaluated on the development set every 200 steps, and the checkpoint yielding the lowest validation loss was subsequently applied to the hold-out test set. To mitigate stochastic variance, all reported performance values (F1-score for virus variant classification and MCC for all remaining tasks) represent the average across three independent runs initialized with different random seeds. By requiring simultaneous nucleotide-level accuracy and long-range contextual reasoning, this GUE subset directly quantifies how well the model encodes the *cis*-regulatory grammar of complex genomes.

Mapping to the reference genome

To avoid dense segmentation and inflated recall, we merged shorter tokens using the output logits of the adaptive segmentation layer. For each boundary between adjacent bases, we compared the logits on the left and right sides and merged the base with its neighboring base according to the higher logit. This process was repeated until no token shorter than 3 bp remained, consistent with the fixed 3 bp length constraint of start and stop codons. In contrast, for base-resolution biological signals such as DNA methylation, no merging was applied and length-1 tokens from the raw GenART segmentation were used directly. The derived boundaries were then evaluated against reference annotations from GENCODE v49 (GRCh38). Gene and transcript structures were obtained from the comprehensive GENCODE GTF, while CpG islands and enhancer/promoter regions were sourced from the UCSC CpG Island track and the Ensembl Regulatory Build, respectively. DNA methylated sites were compiled from multiple large-scale methylome resources, including ENCODE, the Roadmap Epigenomics Project, the EWAS Data Hub, and GEO (Table S7). Raw datasets from these sources (450K/850K arrays, RRBS, and WGBS) were harmonized by converting probe or read-level methylation calls

Please cite this article in press as: Chen et al., GenART: Reading the genome's language with adaptive ''words'', Cell Reports Methods (2026), https://doi.org/10.1016/j.crmeth.2026.101516

to GRCh38 genomic coordinates, removing duplicated and malformed entries, normalizing chromosome naming, and unifying all coordinates into a standardized 1-based, end-inclusive BED format. The final curated, non-redundant genome-wide set of DNA methylated sites used for evaluation is provided as Data S1. All external annotations were converted to a unified BED representation and standardized to consistent chromosome naming and 1-based, end-inclusive coordinates. Annotation intervals were parsed into independent start and end boundary points using a uniform Python-based preprocessing pipeline. Chromosome labels were normalized, malformed intervals were removed, and BED/GTF end-position conventions were reconciled. GenART-predicted token boundaries were extracted from model outputs and mapped to the same coordinate system. A predicted boundary was considered correct if it fell within ±1 bp of a reference boundary. Boundary-level recall was defined as the proportion of annotated boundary points matched by at least one GenART boundary.

Baseline model evaluation on automated genome annotation

To enable a fair comparison across models, we evaluated GenART together with SegmentNT, DNABERT2, MxDNA, and a fixed-length Nucleotide Transformer (6-mer) using a unified evaluation framework. Experiments were conducted on two complementary datasets: a curated gene set derived from the BUSCO eukaryotic lineage and genome-wide annotations from GRCh38.

BUSCO genes were retrieved using the eukaryota lineage dataset (version 2025-07-01). A total of 42,711 gene IDs were initially collected. Gene sequences and corresponding annotations were obtained from NCBI using ID mapping. Entries with unsuccessful ID conversion or missing annotation information were removed, resulting in a final curated set of 33,416 genes. The scripts utilized to generate the processed BUSCO sequences and annotation files are available in Data S2.

For inference, input genomic sequences were segmented into non-overlapping windows of 1 kb to accommodate model input constraints. Each segment was tokenized independently using the respective model, and tokenized outputs were subsequently concatenated to reconstruct the full-length sequence representation.

To ensure consistent evaluation, only the sequence region corresponding to the annotated gene interval (query_start to query_end) was retained for downstream analysis. Predicted token boundaries within this region were extracted and mapped back to genomic coordinates.

All models were evaluated using a consistent boundary definition and matching criterion: a predicted boundary was considered correct if it fell within ±1 bp of a reference boundary. Boundary-level performance was quantified using Matthews correlation coefficient (MCC) as the primary metric, providing a balanced evaluation under class imbalance.

Hardware

All experiments were conducted on NVIDIA A800 GPUs. During pre-training, we employed a cluster of eight A800 GPUs. GenART (350M) required approximately 50 GPU-hours, whereas GenART (1B) demanded about 70 GPU-hours. Fine-tuning was performed with two A800 GPUs. Although downstream task-specific fine-tuning times varied, the averages were as follows: GenART (350M) required ∼90 min on the NT benchmark and ∼25 min on the GUE benchmark; GenART (1B) required ∼150 min on NT and ∼40 min on GUE.

QUANTIFICATION AND STATISTICAL ANALYSIS

Performance metrics and evaluation

To rigorously evaluate model performance, we strictly adhered to the original evaluation protocols of the Nucleotide Transformer (NT) and Genome Understanding Evaluation (GUE) benchmarks. Specifically, for the NT benchmark, we reported the Matthews Correlation Coefficient (MCC) for histone modification and enhancer prediction tasks, Accuracy (ACC) for the multi-class splice-site annotation task, and F1-score for the remaining tasks including promoter predictions. For the GUE benchmark, we reported the F1-score for the virus variant classification task and MCC for all remaining tasks. Additionally, for long-context splice-site identification (SpliceAI benchmark[20]), models were evaluated using Top-k accuracy to assess true signal ranking, alongside the Area Under the Precision-Recall Curve (PR-AUC) to provide a robust assessment of precision and recall.

Cross-validation and replicability

For the Nucleotide Transformer (NT) benchmark evaluations, model hyperparameter optimization was conducted exclusively via a 10-fold cross-validation strategy (90% training, 10% validation) to prevent data leakage. Final model performance was subsequently evaluated on fixed, non-overlapping, held-out test sets. For the Genome Understanding Evaluation (GUE) benchmark, predefined and fixed train/development/test splits were utilized to ensure identical data instances across baselines. To mitigate stochastic variance in GUE evaluations, all reported performance values represent the mean across three independent experimental runs ($n = 3$) initialized with different random seeds.

Genome annotation analysis and exclusion criteria

Boundary-level recall was calculated as the proportion of annotated boundary points matched by at least one model-predicted boundary. A predicted boundary was defined as correct only if it fell within ±1 bp of a reference genomic boundary. During data

Please cite this article in press as: Chen et al., GenART: Reading the genome's language with adaptive "words", Cell Reports Methods (2026), https://doi.org/10.1016/j.crmeth.2026.101516

preprocessing for automated genome annotation, raw datasets were standardized, and malformed intervals, duplicated entries, and entries with unsuccessful ID conversions or missing annotation information were excluded from downstream analysis.

Data distribution and statistical measures

Statistical dispersion of attention uniformity across different model capacities was quantified using the coefficient of variation (CV), which normalizes variance against the mean to eliminate scale-dependent artifacts. Changes in tokenization granularity were analyzed via the empirical cumulative distribution function (CDF) of token lengths. Parameter combinations for refining target token density and cluster factors were assessed using grid search experiments.