Securing the Language of Life: Inheritable Watermarks from DNA Language Models to Proteins

Zaixi Zhang[*]
Princeton University
zz8680@princeton.edu

Ruofan Jin
Zhejiang University
ruofanjin@zju.edu.cn

Le Cong[*]
Stanford University
congle@stanford.edu

Mengdi Wang[*]
Princeton University
mengdiw@princeton.edu

Abstract

DNA language models have revolutionized our ability to understand and design DNA sequences—the fundamental language of life—with unprecedented precision, enabling transformative applications in therapeutics, synthetic biology, and gene-editing. However, this capability also poses substantial dual-use risks, including the potential for creating pathogens, viruses, even bioweapons. To address these biosecurity challenges, we introduce two innovative watermarking techniques to reliably track the designed DNA: DNAMark and CentralMark. DNAMark employs synonymous codon substitutions to embed watermarks in DNA sequences while preserving the original function. CentralMark further advances this by creating inheritable watermarks that transfer from DNA to translated proteins, leveraging protein embeddings to ensure detection across the central dogma. Both methods utilize semantic embeddings to generate watermark logits, enhancing robustness against natural mutations, synthesis errors, and adversarial attacks. Evaluated on our therapeutic DNA benchmark, DNAMark and CentralMark achieve F1 detection scores above 0.85 under various conditions, while maintaining over 60% sequence similarity to ground truth and degeneracy scores below 15%. A case study on the CRISPR-Cas9 system underscores CentralMark's utility in real-world settings. This work establishes a vital framework for securing DNA language models, balancing innovation with accountability to mitigate biosecurity risks.

1 Introduction

DNA serves as the cornerstone of the central dogma [13], orchestrating the flow of genetic information from DNA to RNA to proteins. Within this paradigm, DNA encodes the genetic blueprint, RNA acts as a dynamic messenger, and proteins execute a vast array of cellular functions (Figure 1 a). Recent advances in DNA language models have transformed our ability to understand and design DNA sequences with unprecedented precision [46, 8, 74, 47, 71, 42]. These models leverage computational frameworks to decode complex sequence patterns, enabling groundbreaking applications in therapeutics, synthetic biology, gene-editing, and beyond.

However, the remarkable capabilities of DNA language models also introduce significant dual-use risks [7, 52, 5, 76, 23]. For example, these models could lower the barrier to the creation of harmful biological agents, such as pathogens, viruses, or bioweapons. State-of-the-art DNA models excel in predicting and generating sequences with missense mutations or pathogenic properties

[*] Corresponding Authors

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

[46, 8, 71, 42, 18], amplifying biosecurity concerns. The AI and scientific communities have recognized the emerging risks of DNA language models and are advocating robust guardrails and comprehensive oversight mechanisms [70, 63, 5, 52, 49].

Recently, watermarking has emerged as an effective strategy to counter the misuse of large language models (LLMs), enabling the traceability of generated content to ensure accountability and mitigate risks such as misinformation or malicious output [16, 34]. However, the application of watermarking to DNA language models presents unique and underexplored challenges. Unlike LLMs, which operate on expansive vocabularies, DNA language models are constrained by a *small alphabet* of only four nucleotides, complicating the design of robust watermarking strategies, such as green/red list approaches. Moreover, DNA is susceptible to *natural mutations [62], synthesis errors, and sequencing inaccuracies* [58], which can obscure or degrade watermarks. Additional complexities arise from biological constraints to preserve the*functional integrity* of encoded sequences to maintain their utility in applications like protein engineering. These challenges necessitate new watermarking frameworks tailored to the biological and computational intricacies of DNA sequence design.

To tackle these challenges, we propose a function-invariant watermark **DNAMark** using synonymous codon substitutions and **CentralMark** that builds an inheritable watermark transferable from designed DNA to translated protein. DNAMark and CentralMark address the challenges with the following innovations: **(1)** To achieve robust watermark resistant to natural mutations and potential attacks, DNAMark and CentralMark utilize the generated DNA or translated protein embeddings (Evo2 [8] or ESM [37]) to predict watermark logits with trained watermark models. The watermark logits are then added to the original logits from DNA models to bias the next nucleotide selection for watermarking. The intuition is that DNA and protein embeddings are inherently robust to minor mutations, preserving semantic and functional integrity during watermark logit prediction. During training, the watermark model is optimized to prioritize semantic preservation and maintain an unbiased distribution, enhancing watermark robustness and performance. **(2)** To minimize disruption to DNA sequence quality and encoded protein function, **DNAMark** employs a sparse watermarking scheme with synonymous codon substitutions, selectively modifying only the third base of specific codons to ensure the resulting codon encodes the same amino acid as the unmarked sequence (Figure 1 d). **(3)** To ensure inheritable watermark in both DNA and translated protein, **CentralMark** predicts watermark logits from protein embeddings and applies the watermark to the second base of each codon, enabling near non-overlapping separation of amino acids into green/red lists, facilitating reliable watermark detection across the central dogma (Figure 1 e).

Using our curated therapeutic DNA benchmark (Figure 1b), **DNAMark** and **CentralMark** achieve robust F1 detection scores (>0.85) under various attacks, including nucleotide substitution, insertion, and deletion attacks. Meanwhile, DNA sequence qualities are preserved, with over 60% sequence similarity to ground truth and degeneracy scores below 15%. Case studies on watermarking a CRISPR-Cas9 system [11, 12] designed by Evo model [46] (Figure 1c) demonstrate **CentralMark**'s potential for practical applications in real-world synthetic biology and gene-editing.

2 Related Works

2.1 Watermark for Language Models

Driven by the need to identify machine-generated text and mitigate potential misuse, the field of watermarking large language models (LLMs) has seen rapid development. Early and influential approaches, such as the one proposed by Kirchenbauer et al. [34], often referred to as KGW, introduced a method of biasing token generation towards a "green list" determined by a pseudorandom function seeded by preceding tokens. This creates a statistical watermark detectable with high accuracy (More details in Section 3). Subsequent works have aimed to improve detectability [24, 40, 35], text quality [30, 25, 28, 72], capacity [24, 69, 73], robustness [39, 53], and public verifiability [22, 38]. For Example, to enhance watermark detectability, EWD [40] assigns weights to tokens based on their entropy during detection, enhancing sensitivity by emphasizing high-entropy tokens in z-score calculations. To mitigate the logits bias brought by KGW applying a uniform δ to green list tokens, Hu et al. [30] introduced two unbiased reweighting methods to preserve the original text distribution. Aiming at increasing the watermark capacity to convey additional information like timestamps, identifiers, or copyright. Fernandez et al. [24] expand binary vocabulary partition to multi-color partition. To further improve watermark robustness against removal attacks such as

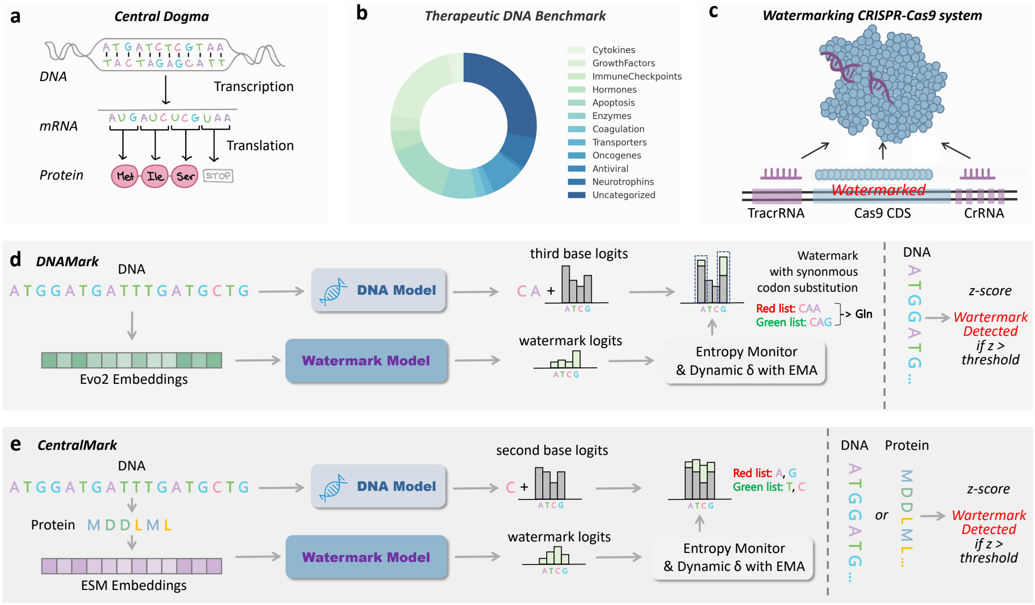


Figure 1: Watemark DNA language models with DNAMark and CentralMark. (a). DNA plays a key role in the central dogma; (b). A therapeutic DNA benchmark is constructed to evaluate DNA watermarks; (c) Our watermark methods successfully watermarks CRISPR-Cas9 generated by Evo; (d) DNAMark leverages watermark models and synonymous codon substitutions for DNA watermark; (e) CentralMark uses ESM-based watermark model to achieve an inheritable watermark. Watermark can be detected in both the DNA and the translated protein sequence generated with CentralMark.

paraphrasing, semantic-invariant watermark methods [39, 53] are proposed to ensure that similar text semantics result in similar partition outcomes, which are robust to attacks. To achieve publicly verifiable watermarks, Fairoze et al. [22] have utilized a digital signature technology from the field of cryptography, involving generating watermarks using a private key and verifying them with a public key. Recently, Zhang et al. [75] and Chen et al. [9] applied watermarks to protein generative models. However, it is unknown whether a watermark scheme can be designed for DNA language models and the central dogma.

2.2 DNA Language Models

Driven by advances in LLMs, DNA Language Models (DNA LMs) have also experienced rapid progress in recent years. Early DNA LMs primarily focused on DNA sequence interpretation and property prediction [33, 78, 55, 4]. For instance, Enformer combined convolutional down-sampling with transformer layers to enable accurate gene-expression prediction [4], while the Nucleotide Transformer, trained on multi-species corpora, markedly improved variant-effect prediction [15]. More recently, DNA LMs with advanced sequence generation capabilities have emerged [59, 74, 47, 71, 42, 46, 8]. For example, HyenaDNA leveraged implicit long-range convolutions to scale context to one million tokens [47]. GENERATOR, a 1.2B transformer decoder trained on 386 billion base pairs of eukaryotic DNA, excels in generating viable protein-coding sequences [71]. Evo, a 7B model trained on billions of prokaryotic and viral base pairs, demonstrated advanced capabilities in designing CRISPR–Cas complexes [46]. Its successor, Evo2, was scaled using 9.3 trillion DNA base pairs with one-million-token context windows, yielding autoregressive models with 7B and 40B parameters. Evo2 enables genome-wide prediction and *de novo* synthesis of DNA sequences across all domains of life [8]. Evo2 excels in generating chromosome-scale sequences, including similar sequences to human mitochondrial, *M. genitalium*, and *S. cerevisiae* genomes.

The advanced capabilities of DNA language models simultaneously raise significant biosafety and biosecurity concerns [70, 63]. Current countermeasures, such as sequence screening [1] and regulatory policies [5], are often suboptimal, as they may fail to detect AI-generated sequences or adapt to evolving model capabilities [49]. Robust watermarking techniques tailored for DNA could enable reliable tracing and detection of AI-generated DNA sequences, addressing these gaps.

3 Preliminaries

Autoregressive language models, such as transformer-based architectures, generate text by modeling the conditional probability of a token given its preceding context. Formally, for a sequence of tokens $\mathbf{x} = (x_1, x_2, \ldots, x_T)$, an autoregressive model predicts the next token x_t based on the probability distribution $p(x_t|x_{1:t-1}; \theta)$, where θ denotes the model parameters. The joint probability of the sequence is expressed as:

$$p(\mathbf{x}; \theta) = \prod_{t=1}^{T} p(x_t|x_{1:t-1}; \theta). \tag{1}$$

These models excel at producing coherent and contextually relevant text, but their widespread use raises concerns about content authenticity, ownership, and traceability.

To address these challenges, watermarking techniques embed imperceptible identifiers into the outputs of language models. A watermark is a subtle, structured modification to the generated text, designed to be robust against post-processing (e.g., paraphrasing) while remaining inconspicuous to human readers. For example, the KGW watermarking scheme [34] modifies the token probability distribution during generation. Specifically, for a vocabulary $\mathcal{V}$, KGW partitions tokens into a "green" list $\mathcal{G} \subset \mathcal{V}$ and a complementary "red" list $\mathcal{R} = \mathcal{V} \setminus \mathcal{G}$ based on a cryptographic hash of the context. The probability of selecting a token $x_t \in \mathcal{G}$ is boosted by an additive term δ, altering the sampling distribution as:

$$p_{\text{wm}}(x_t|x_{1:t-1}; \theta) \propto p(x_t|x_{1:t-1}; \theta) + \delta \cdot \mathbb{I}(x_t \in \mathcal{G}), \tag{2}$$

where $\mathbb{I}(\cdot)$ is the indicator function, and the modified distribution is normalized. This ensures the watermark is embedded without significantly degrading text quality.

Watermark detection involves identifying the presence of these embedded identifiers in a suspect text. In the KGW scheme, detection leverages a statistical hypothesis test based on the z-score, which quantifies the likelihood that a given text $\mathbf{x}$ was generated by a watermarked model. Specifically, the detector counts the number of tokens in the green list, denoted $r = \sum_{t=1}^{T} \mathbb{I}(x_t \in \mathcal{G})$, over the sequence of length T. Under the null hypothesis (no watermark), tokens are sampled uniformly from $\mathcal{V}$, and the expected proportion of green tokens is $\gamma = |\mathcal{G}|/|\mathcal{V}|$. The z-score is computed as:

$$z = \frac{r - \mathbb{E}[r]}{\sqrt{\text{Var}[r]}} = \frac{r - T \cdot \gamma}{\sqrt{T\gamma\,(1-\gamma)}}, \tag{3}$$

where $\mathbb{E}[r] = T \cdot \frac{|\mathcal{G}|}{|\mathcal{V}|}$ and $\text{Var}[r] = T\gamma\,(1-\gamma)$ assume a binomial distribution for r. A high z-score (e.g., $z \geq \tau$ for a threshold τ) indicates the presence of the watermark, as the observed green token count significantly exceeds the expected count under the null hypothesis.

4 Methods

4.1 DNAMark: Function-invariant Watermark for DNA Models

To achieve resistance to natural mutations and function preservation for synthetic biology, we first build DNAMark (Figure 1 (d)), a robust, and function-invariant watermark scheme for DNA language models in this section. Inspired by previous works on semantic-invariant watermarks for LLMs [39, 53], DNAMark utilizes **a specialized trained watermark model to generate watermark logits** for robustness. For watermarking in the coding region, we use **synonymous codon substitutions** to keep the coded amino acid unchanged. Moreover, **adaptive watermark strength** and **entropy-guided watermark strategy** are applied to balance sequence quality and detection accuracy.

4.1.1 Watermark Model based on Evo2 Embeddings

To embed a robust watermark in generated DNA sequences, DNAMark processes the sequence preceding the current token through the Evo2 [8] model to obtain functional embeddings, which are then transformed into watermark logits and combined with the original token logits. Leveraging DNA's inherent robustness as an information carrier [10, 26, 21], where small mutations typically preserve encoded biological functions, DNAMark is designed to provide a durable watermark for DNA language models, resisting both natural mutations and adversarial modifications. Specifically, the watermark model in DNAMark satisfies two critical properties: *semantic preservation*, ensuring the watermark maintains the sequence's biological semantics (e.g., protein coding or regulatory roles) by aligning logit similarities with Evo2 embedding similarities. Moreover, the logits should be varied sufficiently to enhance complexity and security. Otherwise, if the watermark logits are monotonous, the green list is more static and might be revealed by counting the token frequency. This compromises the watermark protection and leads to the risk of being cracked. The second property, *unbiased distribution*, ensures that watermark logits exhibit no systematic preference for any nucleotide or codon and maintain a balanced distribution of positive and negative values, enhancing security against statistical attacks and ensuring robust, detectable watermarks for DNA sequences.

To realize these properties, we trained the watermark model [39] (Appendix. G), comprising multiple fully connected layers and layer norm, with two main loss functions: an alignment loss and a normalization loss. The alignment loss aligns the watermark logit similarity with the Evo2 embedding similarity: we normalize the embedding similarities by subtracting their mean and applying the hyperbolic tangent function. The alignment loss $\mathcal{L}_a$ is defined as:

$$\mathcal{L}_a = \sum_{i,j} \left| \frac{\mathbf{w}_i \cdot \mathbf{w}_j}{\|\mathbf{w}_i\|_2 \|\mathbf{w}_j\|_2} - \tanh\left(k \left(\frac{\mathbf{e}_i \cdot \mathbf{e}_j}{\|\mathbf{e}_i\|_2 \|\mathbf{e}_j\|_2} - \frac{1}{|N|^2} \sum_{k,l} \frac{\mathbf{e}_k \cdot \mathbf{e}_l}{\|\mathbf{e}_k\|_2 \|\mathbf{e}_l\|_2} \right) \right) \right|, \tag{4}$$

where $\mathbf{e}_i$ is the Evo2 embedding for sequence i, $\mathbf{w}_i$ is the watermark logit vector produced by the watermark model, $|N|$ is the number of sequences, k is a hyperparameter controlling the similarity range, and $\| \cdot \|_2$ denotes the Euclidean norm. This loss ensures watermark logits reflect DNA functional relationships while enhancing separability.

Following [39], the normalization loss enforces unbiased token preference and balanced scores. It constrains the mean of the watermark logits to zero across tokens and sequences and ensures uniform absolute values for stability. The normalization loss $\mathcal{L}_n$ is defined as:

$$\mathcal{L}_n = \sum_{i=1}^{|N|} \left| \sum_{j=1}^{|\mathcal{V}|} \mathbf{w}_i^{(j)} \right| + \sum_{j=1}^{|\mathcal{V}|} \left| \sum_{i=1}^{|N|} \mathbf{w}_i^{(j)} \right| + \lambda \sum_{i=1}^{|N|} \sum_{j=1}^{|\mathcal{V}|} \left| R - \mathbf{w}_i^{(j)} \right|, \tag{5}$$

where $\mathbf{w}_i^{(j)}$ denotes the j-th value in the watermark logit; R is a hyperparameter specifying the target absolute value for each logit component, and λ is a weighting factor. This loss ensures the watermark is statistically neutral and detectable. The total loss combines the above two objectives. During watermarked generation, the watermark logits, scaled by a watermark strength factor δ, are added to the original logits to bias the sampling of the next nucleotide.

4.1.2 Synonymous Codon Substitutions

To design a function-invariant watermark for DNA language models, DNAMark employs *synonymous codon substitution (SCS)* within the coding DNA sequence (CDS), targeting the *third base* of codons to embed identifiers that preserve the encoded amino acid, critical for synthetic biology applications. For a codon with fixed first two bases (e.g., CA) and an intended amino acid (e.g., Histidine for CAT), DNAMark defines green and red lists within the synonymous codon set (e.g., CAC as red list and CAT as green list), to keep the encoded protein unchanged (i.e., no matter red or green list is chosen, the same amino acid type). This approach is motivated by several considerations: **First**, synonymous codons produce identical amino acids, thereby maintaining the protein's structure and function critical for applications in synthetic biology. **Second**, targeting the third base leverages the degeneracy of the genetic code, where mutations at this position are often silent [31], minimizing the influence of watermarking on DNA sequences. **Third**, by watermarking only the third base, DNAMark achieves a sparse watermark that balances robust detectability with high DNA sequence quality, minimizing disruptions to codon usage and sequence optimality. Following previous works [34, 24], we explicitly define the green and red lists for watermark. Considering different cases of synonymous codons (more details in Table B), the green and red lists ($\mathcal{G}, \mathcal{R}$) are constructed as:

$$\mathcal{G}, \mathcal{R} = \begin{cases} \{b_g\}, \mathcal{S} \setminus \{b_g\} & \text{if } |\mathcal{S}| = 2 \text{ (e.g., T, C for CAT, CAC; Histidine)}, \\ \{b_g\}, \mathcal{S} \setminus \{b_g\} & \text{if } |\mathcal{S}| = 3 \text{ (e.g., T, C, A for ATT, ATC, ATA; Isoleucine)}, \\ \{b_g\}, \mathcal{S} \setminus \{b_g\} & \text{if } |\mathcal{S}| = 4 \text{ (e.g., T, C, A, G for GCT/ C/ A/ G; Alanine)}, \\ \emptyset, \emptyset & \text{if } |\mathcal{S}| = 1 \text{ (e.g., G for ATG; Methionine)}, \end{cases} \tag{6}$$

where $\mathcal{S} = \{b_3 \in \{T, C, A, G\} \mid \texttt{translate}(b_1, b_2, b_3) = a\}$ is the set of third bases yielding the same amino acid a, and $|\mathcal{S}|$ is the set size; b_1, b_2 are the first two bases, `translate` maps codons to amino acids; $\{b_g\} \in \mathcal{S}$ is the green base list, selected as the base type with the highest watermark logits in $\mathcal{S}$. For $|\mathcal{S}| = 2$ (e.g., b_1=C, b_2=A, a=Histidine), one base is green (e.g., T for CAT) and one red (e.g., C for CAC); for $|\mathcal{S}| = 3$ (e.g., b_1=A, b_2=T, a=Isoleucine), one is green (e.g., C) and two red (e.g., T, A); for $|\mathcal{S}| = 4$ (e.g., b_1=G, b_2=C, a=Alanine), one is green and three red; and for $|\mathcal{S}| = 1$ (e.g., b_1=A, b_2=T, a=Methionine), watermarking is skipped as no synonymous alternatives exist.

4.1.3 Adaptive Watermark Strength and Entropy-guided Watermark

Given the small vocabulary of DNA sequences (A, C, T, G) and the instability of autoregressive DNA language models, where excessive watermarking may produce invalid sequences such as repeated motifs or model corruption, DNAMark employs optimization strategies to balance detectability and sequence quality. Specifically, we introduce two optimization strategies: *Adaptive Watermark Strength* and *Entropy-guided Watermarking*. The **Adaptive Watermark Strength strategy** dynamically adjusts the watermark logit strength, δ, using an Exponential Moving Average (EMA) [29] based on the current z-score, z_t, which measures the statistical significance of the watermark signal (i.e., green base frequency in green/red lists [34]). The strength is smoothly updated as a weighted average of the current strength within a target range $[z_{\text{min}}, z_{\text{max}}]$. The adjustment is defined as:

$$\texttt{adj}(z_t, z_{\text{min}}, z_{\text{max}}) = \begin{cases} z_{\text{min}} - z_t & \text{if } z_t < z_{\text{min}}, \\ 0 & \text{if } z_{\text{min}} \leq z_t \leq z_{\text{max}}, \\ z_{\text{max}} - z_t & \text{if } z_t > z_{\text{max}}, \end{cases} \tag{7}$$

and δ is smoothly updated as a weighted average of the current strength and a target adjustment:

$$\delta_{t+1} = (1 - \beta)\delta_t + \beta \cdot \max\left(\delta_{\text{min}}, \min\left(\delta_{\text{max}}, \delta_t + \kappa \cdot \texttt{adj}(z_t, z_{\text{min}}, z_{\text{max}})\right)\right), \tag{8}$$

where δ_t is the strength at step t, $\beta \in (0, 1)$ controls the update speed, $\delta_{\text{min}}, \delta_{\text{max}}$ are bounds, and κ scales the adjustment. If $z_t < z_{\text{min}}$, δ increases to enhance detectability; if $z_t > z_{\text{max}}$, δ decreases to preserve sequence quality; and if $z_t \in [z_{\text{min}}, z_{\text{max}}]$, δ remains stable. During generation, watermark logits, scaled by δ_t, are added to the original logits.

The **Entropy-guided Watermarking strategy** skips watermarking in low-entropy subsequences to avoid disrupting critical sequence patterns, such as regulatory motifs in UTRs. The entropy H of a subsequence s (e.g., a window of nucleotides) is computed as:

$$H(s) = -\sum_{b \in \{T, C, A, G\}} p(b) \log p(b), \tag{9}$$

where $p(b)$ is the frequency of base b in s. If $H(s) < H_{\text{threshold}}$, watermarking is skipped for that subsequence, ensuring minimal impact on functional elements like ribosome binding sites or structural motifs. These strategies together enhance DNAMark's watermark, preserving sequence quality while maintaining robust detectability against mutations and adversarial edits.

4.2 CentralMark: Inheritable Watermarks from DNA to Proteins

Recent DNA language models not only learns DNA sequences but also captures the central dogma [13]'s flow of genetic information from DNA to RNA to protein [8, 46]. To extend the traceability of our DNA watermark beyond the nucleotide sequence, we introduce an inheritable watermark (CentralMark) *detectable in both generated DNA and the translated protein sequence*, a critical feature to ensure biosecurity and ownership verification in synthetic biology applications where proteins are the functional output (Figure 1 (e)). Unlike DNAMark introduced above, which uses synonymous codon substitutions to preserve protein function, the inheritable watermark deliberately alters amino acids by targeting the second base of codons in the coding DNA sequence (CDS), leveraging ESM [37] embeddings of the translated protein instead of Evo2 embeddings of DNA for both watermark generation and detection. We target the *second base of each codon* because it predominantly determines the encoded amino acid's identity or chemical properties, facilitating precise amino acid substitutions, and enables near-nonoverlapping green and red lists for amino acids based on second-base patterns (see Table 3). Specifically, for a codon $c = (b_1, b_2, b_3) \in \mathcal{V}_{\text{CDS}}$, where $b_2 \in \{A, C, G, T\}$, we define a green/red list for the protein sequence by indexing the amino acid $a = \texttt{translate}(c)$ to the second base b_2:

$$\mathcal{G}_a = \{a \mid \texttt{translate}(b_1, b_2, b_3) = a, b_2 \in \mathcal{G}_b\},\ \mathcal{R}_a = \{a \mid \texttt{translate}(b_1, b_2, b_3) = a, b_2 \in \mathcal{R}_b\}, \tag{10}$$

where $\mathcal{G}_b$ and $\mathcal{R}_b$ are the green and red sets of second bases (e.g., $\mathcal{G}_b = \{C, G\}$), and `translate` maps codons to amino acids (e.g, $\mathcal{G}_a = \{$Leu, Pro, His, Gln, Arg, Val, Ala, Asp, Glu, Gly$\}$). During watermarking, we bias the selection of codons with $b_2 \in \mathcal{G}_b$ to embed the signature, which propagates to the protein as a biased distribution of amino acids in $\mathcal{G}_a$. In CentralMark, the green sets of second bases are chosen by selecting the bases with the top-2 highest watermark logits. By embedding watermarks in DNA sequences based on their translated protein sequences, we enable subsequent *detection of the protein sequences independently, without requiring additional DNA information*.

4.3 Watermark Detection

The watermark detection of DNAMark and CentralMark follows KGW's calculating z score (Equation. 3). We need to note that the expected proportion of green tokens, γ, may not be 0.5 in DNAMark and CentralMark due to the unique design, such as synonymous codon substitutions. Under the assumption of uniform codon usage, γ is set to 0.3559, 0.5, and 0.55 for DNAMark, CentralMark (DNA), and CentralMark (Protein) respectively. The details are included in the Appendix. E.

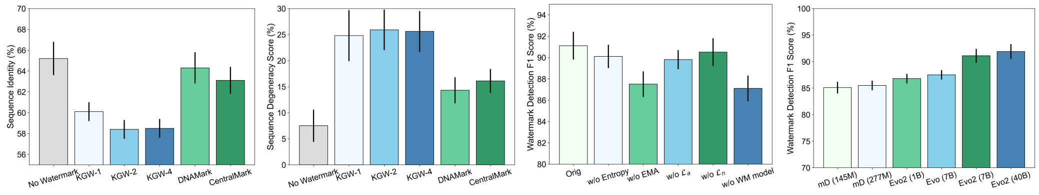
5 Experiments

5.1 Experiment Settings

BenchMark Construction To construct a biologically grounded benchmark for evaluating DNA watermarks, we curated a set of therapeutically important protein-coding genes from *Homo sapiens (Human)* and existing drug modalities. These genes were selected based on their established relevance in clinical and pharmaceutical contexts, encompassing categories such as cytokines (e.g., IL2 [61], TNF [51]), growth factors (e.g., VEGFA [36], EGF [27]), immune checkpoint proteins (e.g., PDCD1 [60], CD274 [19]), apoptosis regulators (e.g., TP53 [41], BCL2 [65]), oncogenes (e.g., KRAS [57], BRAF [17]), antiviral effectors (e.g., IFNA1 [45], TLR3 [3]), coagulation factors (e.g., F8 [64], F2 [68]), and other categories relevant to disease and therapy. For each gene, we queried the NCBI RefSeq database [50] to retrieve validated coding DNA sequences (CDS) with canonical start and stop codons. We integrated secondary structure annotations (helix, β-strand, loop) from UniProt [66] to ensure structural context. Monomeric proteins with varied secondary structures were selected, constructing a benchmark with 400 DNA sequences (More details in Appendix. C). In Case Study, we explored watermarking CRISPR-Cas9 with both coding and non-coding regions.

Attacks To evaluate the robustness of our proposed watermarking scheme, we subjected the watermarked DNA sequences to a series of simulated genetic alterations, mimicking common evolutionary and mutational processes. These *in silico* attacks comprised three distinct types of modifications: (1) **Synonymous Codon Substitutions** replace codons with alternatives that encode the same amino acid [14, 48] (2) **Nucleotide Substitutions** means changing randomly seleted nucleotides to other types in DNA [54, 67], which can lead to either synonymous or non-synonymous codon changes; and (3) **Insertions and deletions (Indels)**, are structural variants that add or remove nucleotides. Here we consider add or remove codons [43, 44]. These attacks are performed at a frequency of 5% across the sequence to simulate a harsh test for the watermark's detectability and robustness (natural mutation frequency $10^{-3} - 10^{-8}$ [56, 20]).

Evaluations For each DNA sequence, we use the first half as a prompt to the DNA language models and generate the rest for 5 times. Inspired by previous works on LLM watermark [39, 77], we report the detection True positive rates at different false positive rates (1% and 10%) to avoid the



(a) (b) (c) (d)

Figure 2: (a) & (b) Generated DNA sequence quality measured by Sequence Identity (the higher the better) and Degeneracy Score (the lower the better). (c) Ablation studies of Entropy Guidance, Adaptive δ with EMA, Alignment and Normalization loss, and the watermark model. We perform 3-time generations for each model and show the standard deviation. (d) Applying DNAMark to different DNA language models and measuring the watermark detection F1 score. mD: megaDNA.

impact of detection thresholds (τ). To assess the quality of generated DNA sequences, we compute the **Sequence Identity** to the ground truth, where higher values indicate better alignment, and the **Degeneracy Score**, defined as the percentage of a sequence covered by repetitive substrings longer than four nucleotides, where lower values are preferable, following Evo [46].

DNA Language Models and Baselines We evaluate DNAMark and CentralMark on the latest and largest DNA language models, Evo [46] and Evo2 (7B, and 40B) [8]. Our methods can also be applied to other DNA models. Hyperparameters are set to $k = 20, \lambda = 10, \kappa = 0.1, \delta_{\text{min}} = 0.5, \delta_{\text{max}} = 3.5, z_{\text{min}} = 2.5, z_{\text{max}} = 4.0, H_{\text{threshold}} = 2.0$, and the Adam optimizer (lr=1e-3) is used for training (Selected Hyperparamter analysis in Figure 4). We adapt KGW with 1, 2, and 4 codon window sizes to DNA as a baseline. All experiments are conducted on 4 Tesla H100 GPUs.

5.2 Results and Robustness Analysis

In Table 1, we compare the performance of DNAMark and CentralMark, detected using DNA and protein sequences, against KGW-1, KGW-2, and KGW-4 under various attack scenarios. Watermarking and detecting DNA sequences is notably more challenging than in natural language models, where methods like KGW achieve near-100% TPR for texts [34], compared to only 70–80% TPR for DNA. We identify two primary reasons for this disparity: (1) DNA's limited vocabulary of four nucleotides (A, C, G, T), versus tens of thousands of tokens in natural language models, severely constrains green/red list assignments, reducing the watermark's statistical distinctiveness. (2) DNA language models exhibit greater brittleness than large language models (LLMs), showing high sensitivity to perturbations in their output distributions. When the watermark strength δ is excessive, it overly biases nucleotide selection, leading to model collapse (e.g., generating repetitive motifs like AAAAA), which compromises both sequence quality and watermark detectability.

Across all attack conditions, DNAMark and CentralMark consistently outperform KGW baselines in TPR and F1 scores at both 1% and 10% FPR. The detection F1 of DNAMark and CentralMark are all above 0.85. CentralMark (DNA) achieves the highest performance in most cases, followed closely by CentralMark (Protein) and DNAMark. The unique design of CentralMark makes the watermark detectable in both the generated DNA and the translated protein. The robustness of DNAMark and CentralMark is due to their use of embeddings (Evo2, ESM), which capture functional/semantic similarity, making watermarks robust even with attacks. For instance, DNAMark and CentralMark achieve high TPR and F1 scores under synonymous codon substitutions, as these changes preserve amino acid sequences and minimally affect the embeddings.

Comparing different attacks, we observe that Nucleotide Substitutions and Indels are the most strong attacks: Substitutions can lead to non-synonymous codons, and Indels can disrupt sequence patterns critical for watermark integrity. For example, the TRP of CentralMark with Indels drops to around 76%, highlighting the severity of these attacks. Nevertheless, DNAMark and CentralMark outperform all baselines. Future work will focus on enhancing robustness to such challenging attacks.

Table 1: We compared the performance of our watermarking methods, DNAMark and CentralMark (DNA/Protein), with baselines, including KGW-k [34], with DNA language model Evo2-7B [8]. Tests evaluated watermark detection accuracy under no attack, synonymous codon substitution, Nucleotide Substitutions, and insertion-deletion (Indels) attacks.

Method	**No attack**				**Synonymous Codon Substitution**			
	1% FPR		10% FPR		1% FPR		10% FPR	
	TPR	F1	TPR	F1	TPR	F1	TPR	F1
KGW-1	0.765	0.862	0.805	0.845	0.580	0.729	0.756	0.815
KGW-2	0.770	0.865	0.820	0.854	0.545	0.701	0.740	0.805
KGW-4	0.774	0.868	0.817	0.852	0.371	0.537	0.520	0.642
DNAMark	0.845	0.911	0.915	0.908	0.820	0.896	0.896	0.898
CentralMark (DNA)	**0.875**	**0.928**	0.920	0.911	0.854	0.916	**0.910**	**0.905**
CentralMark (Protein)	0.868	0.924	**0.922**	**0.912**	**0.860**	**0.920**	0.904	0.902

Method	**Nucleotide Substitutions**				**Indels**			
	1% FPR		10% FPR		1% FPR		10% FPR	
	TPR	F1	TPR	F1	TPR	F1	TPR	F1
KGW-1	0.520	0.680	0.710	0.785	0.515	0.675	0.723	0.794
KGW-2	0.505	0.667	0.658	0.749	0.477	0.642	0.645	0.739
KGW-4	0.330	0.493	0.551	0.668	0.339	0.503	0.497	0.623
DNAMark	0.808	0.902	0.886	0.892	**0.795**	**0.878**	**0.860**	**0.877**
CentralMark (DNA)	**0.840**	**0.908**	**0.890**	**0.894**	0.765	0.862	0.850	0.872
CentralMark (Protein)	0.825	0.900	0.885	0.892	0.759	0.858	0.832	0.861

5.3 Generation Quality and Ablation Studies

It is important to keep the sequence quality when watermarking DNA for practical use. In Figure 2 (a) & (b), we show the Sequence Identity to the ground truth and the Degeneracy Score of the generated DNA sequences by different watermark methods. Compared with KGW, DNAMark and CentralMark shows more alignment with no watermark, indicating higher generation quality. This can be attributed to the sparse watermark adapted to DNA and unique methods such as synonymous codon substitution of DNAMark, minimizing the side-effects on sequence quality. In Figure 2 (c), we did ablation studies of various components in DNAMark. Generally, Adaptive watermark strength with EMA and the watermark model are most critical to the successful watermark detection.

5.4 Generalization to Different DNA Models and Time Complexity

In Figure 2 (d), we observe that DNAMark demonstrates robust watermark detection across a range of DNA language models. Using models of varying sizes—megaDNA (145M and 277M parameters), Evo2 (1B, 7B, and 40B), and Evo1 (7B)—DNAMark achieves F1 scores from 0.851 to 0.919. Smaller models, such as megaDNA-145M (F1=0.851) and 277M (F1=0.855), deliver respectable detection accuracy, but are limited by reduced generation capability. Larger models like Evo2-7B (F1=0.911) and Evo2-40B (F1=0.919) excel, leveraging high-capacity embeddings to enhance generation quality and watermark detection. We further measure the generation time cost of DNAMark and CentralMark, comparing them to a baseline with no watermark generation. The time complexity increases by approximately 30% (Table 7), attributable to the compact size of the watermark model.

5.5 Case Study of Watermarking CRISPR-Cas9 System

To show the practical application in gene editing, we utilized the Evo model (evo-1-8k-crispr) to generate the CRISPR-Cas9 [11, 12] DNA sequences, embedding a watermark during generation using CentralMark. Following [46], we use Prodigal [32] to extract Cas9 CDS, MinCED [6] to detect CRISPR arrays, and AlphaFold3 (AF3) [2] to predict the structure. Figure 3 visualizes the generated watermarked Cas9 aligned with the wild-type SpCas9 crystal structure (PDB ID: 4OO8). The generated sequence achieves a TM-score of 0.6802, indicating high structural alignment, and a

Z-score of 5.41, confirming strong watermark detectability. These results demonstrate the efficacy of watermarking Evo-generated CRISPR-Cas9 DNA sequences with minimal impact on biological quality.

6 Conclusions

In this paper, we tackle the pressing biosecurity challenges arising from DNA language models, which hold immense potential for genetic engineering but also pose dual-use risks by enabling the creation of harmful biological agents. To counter these risks, we propose DNAMark, a watermarking method that uses synonymous codon substitutions to embed robust, function-preserving watermarks in DNA sequences, and CentralMark, an advanced technique that generates inheritable watermarks detectable in both DNA and translated proteins. Future work should explore watermark schemes independent of green/red lists to enhance adaptability, investigate their effects on UTRs for regulatory insights, and validate DNAMark and CentralMark through wet lab experiments. These steps are vital to responsibly balance genetic technology innovation with biosecurity.

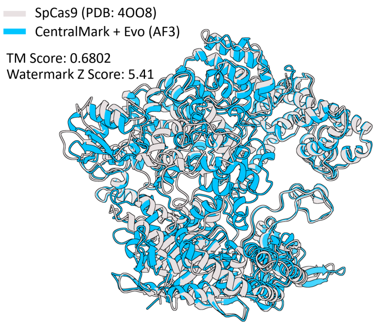


Figure 3: Predicted structure of Evo-desinged Cas9 with CentralMark.

References

[1] Common mechanism - ibbis. `https://ibbis.bio/our-work/common-mechanism/`. Accessed: 2025-04-27.

[2] Josh Abramson, Jonas Adler, Jack Dunger, Richard Evans, Tim Green, Alexander Pritzel, Olaf Ronneberger, Lindsay Willmore, Andrew J Ballard, Joshua Bambrick, et al. Accurate structure prediction of biomolecular interactions with alphafold 3. *Nature*, pages 1–3, 2024.

[3] L. Alexopoulou, A. C. Holt, R. Medzhitov, and R. A. Flavell. Recognition of double-stranded RNA and activation of NF-κb by Toll-like receptor 3. *Nature*, 413(6857):732–738, 2001.

[4] Žiga Avsec, Vikram Agarwal, Daniel Visentin, Joseph R Ledsam, Agnieszka Grabska-Barwinska, Kyle R Taylor, Yannis Assael, John Jumper, Pushmeet Kohli, and David R Kelley. Effective gene expression prediction from sequence by integrating long-range interactions. *Nature methods*, 18(10):1196–1203, 2021.

[5] David Baker and George Church. Protein design meets biosecurity. *Science*, 383(6681):349–349, 2024.

[6] Charles Bland, Teresa L Ramsey, Fareedah Sabree, Micheal Lowe, Kyndall Brown, Nikos C Kyrpides, and Philip Hugenholtz. Crispr recognition tool (crt): a tool for automatic detection of clustered regularly interspaced palindromic repeats. *BMC bioinformatics*, 8:1–8, 2007.

[7] Doni Bloomfield, Jaspreet Pannu, Alex W Zhu, Madelena Y Ng, Ashley Lewis, Eran Bendavid, Steven M Asch, Tina Hernandez-Boussard, Anita Cicero, and Tom Inglesby. Ai and biosecurity: The need for governance. *Science*, 385(6711):831–833, 2024.

[8] Garyk Brixi, Matthew G Durrant, Jerome Ku, Michael Poli, Greg Brockman, Daniel Chang, Gabriel A Gonzalez, Samuel H King, David B Li, Aditi T Merchant, et al. Genome modeling and design across all domains of life with evo 2. *bioRxiv*, pages 2025–02, 2025.

[9] Yanshuo Chen, Zhengmian Hu, Yihan Wu, Ruibo Chen, Yongrui Jin, Marcus Zhan, Chengjin Xie, Wei Chen, and Heng Huang. Enhancing privacy in biosecurity with watermarked protein design. *Bioinformatics*, page btaf141, 2025.

[10] George M Church, Yuan Gao, and Sriram Kosuri. Next-generation digital information storage in dna. *Science*, 337(6102):1628–1628, 2012.

[11] Le Cong, F Ann Ran, David Cox, Shuailiang Lin, Robert Barretto, Naomi Habib, Patrick D Hsu, Xuebing Wu, Wenyan Jiang, Luciano A Marraffini, et al. Multiplex genome engineering using crispr/cas systems. *Science*, 339(6121):819–823, 2013.

[12] Le Cong and Feng Zhang. Genome engineering using crispr-cas9 system. In *Chromosomal mutagenesis*, pages 197–217. Springer, 2014.

[13] Francis Crick. Central dogma of molecular biology. *Nature*, 227(5258):561–563, 1970.

[14] Francis H. C. Crick. Codon–anticodon pairing: the wobble hypothesis. *Journal of Molecular Biology*, 19(2):548–555, 1966.

[15] Hugo Dalla-Torre, Liam Gonzalez, Javier Mendoza-Revilla, Nicolas Lopez Carranza, Adam Henryk Grzywaczewski, Francesco Oteri, Christian Dallago, Evan Trop, Bernardo P de Almeida, Hassan Sirelkhatim, et al. Nucleotide transformer: building and evaluating robust foundation models for human genomics. *Nature Methods*, 22(2):287–297, 2025.

[16] Sumanth Dathathri, Abigail See, Sumedh Ghaisas, Po-Sen Huang, Rob McAdam, Johannes Welbl, Vandana Bachani, Alex Kaskasoli, Robert Stanforth, Tatiana Matejovicova, et al. Scalable watermarking for identifying large language model outputs. *Nature*, 634(8035):818–823, 2024.

[17] H. Davies, G. R. Bignell, C. Cox, P. Stephens, S. Edkins, S. Clegg, J. Teague, H. Woffendin, M. J. Garnett, W. Bottomley, N. Davis, E. Dicks, R. Ewing, Y. Floyd, K. Gray, S. Hall, R. Hawes, J. Hughes, V. Kosmidou, A. Menzies, C. Mould, A. Parker, C. Stevens, S. Watt, S. Hooper, R. Wilson, H. Jayatilake, B. A. Gusterson, C. Cooper, J. Shipley, D. Hargrave, K. Pritchard-Jones, N. Maitland, G. Chenevix-Trench, G. J. Riggins, D. D. Bigner, G. Palmieri, A. Cossu, A. Flanagan, A. Nicholson, J. W. C. Ho, S. Y. Leung, S. T. Yuen, B. L. Weber, H. F. Seigler, T. L. Darrow, H. Paterson, R. Marais, C. J. Marshall, R. Wooster, M. R. Stratton, and P. A. Futreal. Mutations of the BRAF gene in human cancer. *Nature*, 417(6892):949–954, 2002.

[18] Sajib Acharjee Dip, Uddip Acharjee Shuvo, Tran Chau, Haoqiu Song, Petra Choi, Xuan Wang, and Liqing Zhang. Patholm: Identifying pathogenicity from the dna sequence through the genome foundation model. *arXiv preprint arXiv:2406.13133*, 2024.

[19] H. Dong, G. Zhu, K. Tamada, and L. Chen. B7-h1, a third member of the b7 family, co-stimulates t-cell proliferation and interleukin-10 secretion. *Nature Medicine*, 5(12):1365–1369, 1999.

[20] John W. Drake, Brian Charlesworth, Deborah Charlesworth, and James F. Crow. Rates of spontaneous mutation. *Genetics*, 148(4):1667–1686, 1998.

[21] Andy Extance. How dna could store all the world's data. *Nature*, 537(7618), 2016.

[22] Jaiden Fairoze, Sanjam Garg, Somesh Jha, Saeed Mahloujifar, Mohammad Mahmoody, and Mingyuan Wang. Publicly-detectable watermarking for language models. *arXiv preprint arXiv:2310.18491*, 2023.

[23] Jigang Fan, Zhenghong Zhou, Ruofan Jin, Le Cong, Mengdi Wang, and Zaixi Zhang. Safeprotein: Red-teaming framework and benchmark for protein foundation models. *arXiv preprint arXiv:2509.03487*, 2025.

[24] Pierre Fernandez, Antoine Chaffin, Karim Tit, Vivien Chappelier, and Teddy Furon. Three bricks to consolidate watermarks for large language models. In *2023 IEEE International Workshop on Information Forensics and Security (WIFS)*, pages 1–6. IEEE, 2023.

[25] Yu Fu, Deyi Xiong, and Yue Dong. Watermarking conditional text generation for ai detection: Unveiling challenges and a semantic-aware watermark remedy. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 18003–18011, 2024.

[26] Nick Goldman, Paul Bertone, Siyuan Chen, Christophe Dessimoz, Emily M LeProust, Botond Sipos, and Ewan Birney. Towards practical, high-capacity, low-maintenance information storage in synthesized dna. *nature*, 494(7435):77–80, 2013.

[27] H. Gregory. Isolation and structure of urogastrone and its relationship to epidermal growth factor. *Nature*, 257(5524):325–327, 1975.

[28] Batu Guan, Yao Wan, Zhangqian Bi, Zheng Wang, Hongyu Zhang, Pan Zhou, and Lichao Sun. Codeip: A grammar-guided multi-bit watermark for large language models of code. *arXiv preprint arXiv:2404.15639*, 2024.

[29] David Haynes, Steven Corns, and Ganesh Kumar Venayagamoorthy. An exponential moving average algorithm. In *2012 IEEE Congress on Evolutionary Computation*, pages 1–8. IEEE, 2012.

[30] Zhengmian Hu, Lichang Chen, Xidong Wu, Yihan Wu, Hongyang Zhang, and Heng Huang. Unbiased watermark for large language models. *arXiv preprint arXiv:2310.10669*, 2023.

[31] Ryan C Hunt, Vijaya L Simhadri, Matthew Iandoli, Zuben E Sauna, and Chava Kimchi-Sarfaty. Exposing synonymous mutations. *Trends in Genetics*, 30(7):308–321, 2014.

[32] Doug Hyatt, Gwo-Liang Chen, Philip F LoCascio, Miriam L Land, Frank W Larimer, and Loren J Hauser. Prodigal: prokaryotic gene recognition and translation initiation site identification. *BMC bioinformatics*, 11:1–11, 2010.

[33] Yanrong Ji, Zhihan Zhou, Han Liu, and Ramana V Davuluri. Dnabert: pre-trained bidirectional encoder representations from transformers model for dna-language in genome. *Bioinformatics*, 37(15):2112–2120, 2021.

[34] John Kirchenbauer, Jonas Geiping, Yuxin Wen, Jonathan Katz, Ian Miers, and Tom Goldstein. A watermark for large language models. In *International Conference on Machine Learning*, pages 17061–17084. PMLR, 2023.

[35] John Kirchenbauer, Jonas Geiping, Yuxin Wen, Manli Shu, Khalid Saifullah, Kezhi Kong, Kasun Fernando, Aniruddha Saha, Micah Goldblum, and Tom Goldstein. On the reliability of watermarks for large language models. *arXiv preprint arXiv:2306.04634*, 2023.

[36] D. W. Leung, G. Cachianes, W. J. Kuang, D. V. Goeddel, and N. Ferrara. Vascular endothelial growth factor is a secreted angiogenic mitogen. *Science*, 246(4935):1306–1039, 1989.

[37] Zeming Lin, Halil Akin, Roshan Rao, Brian Hie, Zhongkai Zhu, Wenting Lu, Nikita Smetanin, Robert Verkuil, Ori Kabeli, Yaniv Shmueli, et al. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*, 379(6637):1123–1130, 2023.

[38] Aiwei Liu, Leyi Pan, Xuming Hu, Shuang Li, Lijie Wen, Irwin King, and S Yu Philip. An unforgeable publicly verifiable watermark for large language models. In *The Twelfth International Conference on Learning Representations*, 2023.

[39] Aiwei Liu, Leyi Pan, Xuming Hu, Shiao Meng, and Lijie Wen. A semantic invariant robust watermark for large language models. In *The Twelfth International Conference on Learning Representations*, 2024.

[40] Yijian Lu, Aiwei Liu, Dianzhi Yu, Jingjing Li, and Irwin King. An entropy-based text watermarking detection method. *arXiv preprint arXiv:2403.13485*, 2024.

[41] G. Matlashewski, P. Lamb, D. Pim, J. Peacock, L. Crawford, and S. Benchimol. Isolation and characterization of a human p53 cdna clone: expression of the human p53 gene. *The EMBO Journal*, 3(13):3257–3262, 1984.

[42] Aditi T Merchant, Samuel H King, Eric Nguyen, and Brian L Hie. Semantic mining of functional de novo genes from a genomic language model. *bioRxiv*, pages 2024–12, 2024.

[43] Ryan E. Mills, Charles T. Luttig, Christine E. Larkins, Ashley Beauchamp, Cissy Tsui, W. Stephen Pittard, and Scott E. Devine. An initial map of insertion and deletion (indel) variation in the human genome. *Genome Research*, 16(9):1182–1190, 2006.

[44] J. M. Mullaney, R. E. Mills, W. S. Pittard, and S. E. Devine. Small insertions and deletions (indels) in human genomes. *Human Molecular Genetics*, 19(R2):R131–R136, 2010.

[45] S. Nagata, H. Taira, A. Hall, L. Johnsrud, M. Streuli, J. Ecsödi, W. Boll, K. Cantell, and C. Weissmann. Synthesis in e. coli of a polypeptide with human leukocyte interferon activity. *Nature*, 284(5754):316–320, 1980.

[46] Eric Nguyen, Michael Poli, Matthew G Durrant, Brian Kang, Dhruva Katrekar, David B Li, Liam J Bartie, Armin W Thomas, Samuel H King, Garyk Brixi, et al. Sequence modeling and design from molecular to genome scale with evo. *Science*, 386(6723):eado9336, 2024.

[47] Eric Nguyen, Michael Poli, Marjan Faizi, Armin Thomas, Michael Wornow, Callum Birch-Sykes, Stefano Massaroli, Aman Patel, Clayton Rabideau, Yoshua Bengio, et al. Hyenadna: Long-range genomic sequence modeling at single nucleotide resolution. *Advances in neural information processing systems*, 36:43177–43201, 2023.

[48] M. W. Nirenberg, P. Leder, M. Bernfield, R. Brimacombe, J. Trupin, F. Rottman, and C. O'Neal. RNA Codewords and Protein Synthesis, VII. On the General Nature of the RNA Code. *Proceedings of the National Academy of Sciences of the United States of America*, 53(5):1161–1168, 1964.

[49] Nuclear Threat Initiative. Developing guardrails for ai biodesign tools. Online report, November 2024. Accessed: 2025-05-12.

[50] N. A. O'Leary, M. W. Wright, J. R. Brister, S. Ciufo, D. Haddad, R. McVeigh, B. Rajput, B. Robbertse, B. Smith-White, D. Ako-Adjei, A. Astashyn, A. Badretdin, Y. Bao, O. Blinkova, B. Brover, K. Chetvernin, J. Choi, E. Cox, O. Ermolaeva, C. M. Farrell, T. Goldfarb, T. Gupta, D. Haft, E. Hatcher, K. Hlavina, V. S. Joardar, V. K. Kodali, W. Li, D. Maglott, P. Masterson, K. M. McGarvey, M. R. Murphy, K. O'Neill, S. Pujar, L. O. Rangwala, D. Rausch, L. D. Riddick, C. Schoch, K. Shkeda, S. S. Storz, H. Sun, F. Thibaud-Nissen, I. Tolstoy, R. E. Tully, A. R. Vatsan, C. Wallin, D. Webb, W. Wu, M. J. Landrum, A. Kimchi, T. Tatusova, M. DiCuccio, P. Kitts, K. D. Pruitt, and J. Ostell. Reference sequence (RefSeq) database at NCBI: current status, taxonomic expansion, and functional annotation. *Nucleic Acids Research*, 44(D1):D733–D745, January 2016.

[51] D. Pennica, G. E. Nedwin, J. S. Hayflick, P. H. Seeburg, R. Derynck, M. A. Palladino, W. J. Kohr, B. B. Aggarwal, and D. V. Goeddel. Human tumour necrosis factor: precursor structure, expression and homology to lymphotoxin. *Nature*, 312(5996):724–729, 1984.

[52] Rami Puzis, Dor Farbiash, Oleg Brodt, Yuval Elovici, and Dov Greenbaum. Increased cyber-biosecurity for dna synthesis. *Nature Biotechnology*, 38(12):1379–1381, 2020.

[53] Jie Ren, Han Xu, Yiding Liu, Yingqian Cui, Shuaiqiang Wang, Dawei Yin, and Jiliang Tang. A robust semantics-based watermark for large language model against paraphrasing. *arXiv preprint arXiv:2311.08721*, 2023.

[54] Ravi Sachidanandam, Dror Weissman, S. C. Schmidt, J. M. Kakol, L. D. Stein, G. Marth, S. Sherry, J. C. Mullikin, B. J. Mortimore, D. L. Willey, S. E. Hunt, C. G. Cole, P. C. Coggill, C. M. Rice, Z. Ning, J. Rogers, D. R. Bentley, P.-Y. Kwok, E. R. Mardis, R.-F. Yeh, B. Schultz, L. Cook, R. Davenport, M. Dante, L. Fulton, L. Hillier, R. H. Waterston, J. D. McPherson, B. Gilman, S. Schaffner, W. J. Van Etten, D. Reich, J. Higgins, M. J. Daly, S. Gnerre, E. S. Lander, and D. for The International SNP Map Working Group Altshuler. A map of human genome sequence variation containing 1.42 million single nucleotide polymorphisms. *Nature*, 409(6822):928–933, 2001.

[55] Melissa Sanabria, Jonas Hirsch, Pierre M Joubert, and Anna R Poetsch. Dna language model grover learns sequence context in the human genome. *Nature Machine Intelligence*, 6(8):911–923, 2024.

[56] Rafael Sanjuán, Miguel R. Nebot, Nicola Chirico, Louis M. Mansky, and Robert Belshaw. Viral mutation rates. *Journal of Virology*, 84(19):9733–9748, 2010.

[57] E. Santos, S. R. Tronick, S. A. Aaronson, S. Pulciani, and M. Barbacid. T24 human bladder carcinoma oncogene is an activated form of the normal human homologue of balb- and harvey-msv transforming genes. *Nature*, 298(5872):343–347, 1982.

[58] Michael Schwarz, Marius Welzel, Tolganay Kabdullayeva, Anke Becker, Bernd Freisleben, and Dominik Heider. Mesa: automated assessment of synthetic dna fragments and simulation of dna synthesis, storage, sequencing and pcr errors. *Bioinformatics*, 36(11):3322–3326, 2020.

[59] Bin Shao and Jiawei Yan. A long-context language model for deciphering and generating bacteriophage genomes. *Nature Communications*, 15(1):9392, 2024.

[60] T. Shinohara, M. Taniwaki, Y. Ishida, M. Kawaichi, and T. Honjo. Structure and chromosomal localization of the human pd-1 gene (pdcd1). *Genomics*, 23(3):704–706, 1994.

[61] T. Taniguchi, H. Matsui, T. Fujita, C. Takaoka, N. Kashima, R. Yoshimoto, and J. Hamuro. Structure and expression of a cloned cdna for human interleukin-2. *Nature*, 302(5906):305–310, 1983.

[62] Robert W Taylor and Doug M Turnbull. Mitochondrial dna mutations in human disease. *Nature Reviews Genetics*, 6(5):389–402, 2005.

[63] Kristel Tjandra. Built-in safeguards might stop ai from designing bioweapons, April 2025. Accessed: 2025-05-05.

[64] J. J. Toole, J. L. Knopf, J. M. Wozney, L. A. Sultzman, J. L. Buecker, D. D. Pittman, R. J. Kaufman, E. Brown, C. Shoemaker, E. C. Orr, G. W. Amphlett, W. B. Foster, M. L. Coe, G. J. Knutson, D. N. Fass, and R. M. Hewick. Molecular cloning of a cDNA encoding human antihaemophilic factor. *Nature*, 312(5992):342–347, 1984.

[65] Y. Tsujimoto, J. Cossman, E. Jaffe, and C. M. Croce. Involvement of the bcl-2 gene in human follicular lymphoma. *Science*, 228(4706):1440–1443, 1985.

[66] UniProt Consortium. Uniprot: the universal protein knowledgebase in 2023. *Nucleic Acids Research*, 51(D1):D523–D531, January 2023.

[67] Alain Vignal, Denis Milan, Magali SanCristobal, and André Eggen. A review on snp and other types of molecular markers for animal genetics. *Genetics Selection Evolution*, 34(3):275–305, 2002.

[68] D. A. Walz, D. Hewett-Emmett, and W. H. Seegers. Amino acid sequence of human prothrombin fragments 1 and 2. *Proceedings of the National Academy of Sciences of the United States of America*, 74(5):1969–1972, 1977.

[69] Lean Wang, Wenkai Yang, Deli Chen, Hao Zhou, Yankai Lin, Fandong Meng, Jie Zhou, and Xu Sun. Towards codable watermarking for injecting multi-bits information to llms. *arXiv preprint arXiv:2307.15992*, 2023.

[70] Mengdi Wang, Zaixi Zhang, Amrit Singh Bedi, Alvaro Velasquez, Stephanie Guerra, Sheng Lin-Gibson, Le Cong, Yuanhao Qu, Souradip Chakraborty, Megan Blewett, et al. A call for built-in biosecurity safeguards for generative ai tools. *Nature Biotechnology*, pages 1–3, 2025.

[71] Wei Wu, Qiuyi Li, Mingyang Li, Kun Fu, Fuli Feng, Jieping Ye, Hui Xiong, and Zheng Wang. Generator: A long-context generative genomic foundation model. *arXiv preprint arXiv:2502.07272*, 2025.

[72] Yihan Wu, Zhengmian Hu, Hongyang Zhang, and Heng Huang. Dipmark: A stealthy, efficient and resilient watermark for large language models. 2023.

[73] KiYoon Yoo, Wonhyuk Ahn, and Nojun Kwak. Advancing beyond identification: Multi-bit watermark for language models. *arXiv preprint arXiv:2308.00221*, 2023.

[74] Daoan Zhang, Weitong Zhang, Yu Zhao, Jianguo Zhang, Bing He, Chenchen Qin, and Jianhua Yao. Dnagpt: a generalized pre-trained tool for versatile dna sequence analysis tasks. *arXiv preprint arXiv:2307.05628*, 2023.

[75] Zaixi Zhang, Ruofan Jin, Guangxue Xu, Xiaotong Wang, Marinka Zitnik, Le Cong, and Mengdi Wang. Foldmark: Safeguarding protein structure generative models with distributional and evolutionary watermarking. *bioRxiv*, pages 2024–10, 2025.

[76] Zaixi Zhang, Zhenghong Zhou, Ruofan Jin, Le Cong, and Mengdi Wang. Genebreaker: Jailbreak attacks against dna language models with pathogenicity guidance. *arXiv preprint arXiv:2505.23839*, 2025.

[77] Xuandong Zhao, Prabhanjan Ananth, Lei Li, and Yu-Xiang Wang. Provable robust watermarking for ai-generated text. *arXiv preprint arXiv:2306.17439*, 2023.

[78] Zhihan Zhou, Yanrong Ji, Weijian Li, Pratik Dutta, Ramana Davuluri, and Han Liu. Dnabert-2: Efficient foundation model and benchmark for multi-species genome. *arXiv preprint arXiv:2306.15006*, 2023.

A Broad Impacts

The societal implications of DNAMark and CentralMark are profound and multifaceted. On the positive side, these watermarking techniques mitigate biosecurity threats by enabling researchers, regulators, and biosafety organizations to track and verify the origins of synthetic DNA, deterring malicious applications such as the engineering of pathogens. This traceability fosters trust in synthetic biology, supporting advancements in therapeutics, agriculture, and environmental solutions. Moreover, by establishing a framework for responsible innovation, these methods could encourage international collaboration on biosecurity standards, strengthening global oversight of genetic technologies. However, negative consequences must also be considered. The watermarking methods may not be entirely impervious to circumvention by sophisticated adversaries who could exploit vulnerabilities, such as reverse-engineering watermarks or introducing mutations to obscure them. This limitation risks fostering a false sense of security among stakeholders, potentially undermining trust in regulatory frameworks if breaches occur. Additionally, the computational and expertise barriers to implementing these watermarks could disproportionately burden smaller research institutions or developing nations, exacerbating inequities in access to cutting-edge genetic technologies. In the future, we will further refine our watermark methods and establish a community to advance watermarking research and reduce the potential negative impacts.

B Codon-to-Amino-Acid Table

1st/2nd	U	C	A	G
U	UUU Phe UUC Phe UUA Leu UUG Leu	UCU Ser UCC Ser UCA Ser UCG Ser	UAU Tyr UAC Tyr UAA Stop UAG Stop	UGU Cys UGC Cys UGA Stop UGG Trp
C	CUU Leu CUC Leu CUA Leu CUG Leu	CCU Pro CCC Pro CCA Pro CCG Pro	CAU His CAC His CAA Gln CAG Gln	CGU Arg CGC Arg CGA Arg CGG Arg
A	AUU Ile AUC Ile AUA Ile AUG Met	ACU Thr ACC Thr ACA Thr ACG Thr	AAU Asn AAC Asn AAA Lys AAG Lys	AGU Ser AGC Ser AGA Arg AGG Arg
G	GUU Val GUC Val GUA Val GUG Val	GCU Ala GCC Ala GCA Ala GCG Ala	GAU Asp GAC Asp GAA Glu GAG Glu	GGU Gly GGC Gly GGA Gly GGG Gly

Table 2: Standard RNA codon table organized by the first two nucleotides. Each cell shows four codons sharing the same first two bases.

Second Base	Amino Acids
A	Isoleucine (Ile), Methionine (Met), Threonine (Thr), Asparagine (Asn), Lysine (Lys), Serine (Ser), Arginine (Arg)
C	Leucine (Leu), Proline (Pro), Histidine (His), Glutamine (Gln), Arginine (Arg)
G	Valine (Val), Alanine (Ala), Aspartic Acid (Asp), Glutamic Acid (Glu), Glycine (Gly)
T	Phenylalanine (Phe), Leucine (Leu), Serine (Ser), Tyrosine (Tyr), Cysteine (Cys), Tryptophan (Trp), Stop

Table 3: Second base to amino acid mapping for the standard genetic code. This table lists the amino acids corresponding to each possible second base (A, C, G, T) in codons of the coding DNA sequence (CDS), used for CentralMark's inheritable watermark, where the second base is modified to embed a detectable signature in the translated protein.

3-Letter	1-Letter	3-Letter	1-Letter	3-Letter	1-Letter
Ala	A	Gly	G	Pro	P
Arg	R	His	H	Ser	S
Asn	N	Ile	I	Thr	T
Asp	D	Leu	L	Trp	W
Cys	C	Lys	K	Tyr	Y
Glu	E	Met	M	Val	V
Gln	Q	Phe	F		

Table 4: Amino Acid Three-Letter to One-Letter Code Mapping

C Therapeutic DNA Benchmark

Table 5: Statistics of CDS sequences in each therapeutic category.

Category	Count	Avg Length	Min Length	Max Length
Cytokines	15	556.20	282	759
GrowthFactors	77	665.88	180	3501
ImmuneCheckpoints	14	816.21	525	1578
Hormones	18	431.67	333	654
Apoptosis	58	848.90	471	1182
Enzymes	31	4101.10	912	7650
Coagulation	9	2630.33	651	7056
Transporters	7	3706.71	1479	4443
Oncogenes	31	1735.35	567	2424
Antiviral	3	2000.00	570	2715
Neurotrophins	28	1052.68	726	2391
Uncategorized	112	1784.22	255	5028

Table 6: Representative therapeutic genes by category.

Category	Genes
Cytokines	IL2, IL6, IL10, TNF, IFNG
GrowthFactors	EGF, FGF1, VEGFA, PDGFA, TGFB1
ImmuneCheckpoints	PDCD1, CD274, CTLA4, LAG3
Hormones	INS, LEP, GH1, PTH
Apoptosis	BCL2, CASP3, TP53
Enzymes	JAK1, CDK4, MAPK1, MTOR
Coagulation	F8, F9, F2
Transporters	ABCB1, CFTR, SLC2A1
Oncogenes	KRAS, BRAF, MYC
Antiviral	IFNA1, IFNB1, TLR3
Neurotrophins	NGF, BDNF, NTRK1

D More Results of DNAMark and CentralMark

Watermarking Method	Evo(7B)	Evo(40B)
No Watermark	9.7	30.5
DNAMark	12.4	37.2
CentralMark	13.2	40.5

Table 7: Generation times (in seconds) for producing a 128-nucleotide DNA sequence using No Watermark, DNAMark, and CentralMark on Evo(7B) and Evo(40B) models. DNAMark and CentralMark incur computational overhead due to obtaining embedding and watermark model computations.

E Calculation Details of γ

E.1 Calculation of Expected Green Token Proportion (γ) for DNAMark

In the DNAMark watermarking scheme, γ represents the expected proportion of green tokens (third bases in the green list $\mathcal{G}$) under the null hypothesis of no watermark, where the first two bases of codons are uniformly distributed. This calculation is performed for watermarkable positions, i.e., codons with synonymous third bases $|\mathcal{S}| \geq 2$, as defined in Equation (6) and detailed in Appendix B. The process is summarized as follows:

1. **Identify watermarkable codons**: For each codon prefix (b_1, b_2), uniformly distributed over 16 possibilities (probability $\frac{1}{16}$), the synonymous set $\mathcal{S} = \{b_3 \in \{T, C, A, G\} \mid \texttt{translate}(b_1, b_2, b_3) = a\}$ determines the number of third bases encoding the intended amino acid a. Excluding stop codons, the 61 sense codons yield 59 watermarkable codons: 32 with $|\mathcal{S}| = 4$ (e.g., Alanine: GCT, GCC, GCA, GCG), 3 with $|\mathcal{S}| = 3$ (e.g., Isoleucine: ATT, ATC, ATA), and 24 with $|\mathcal{S}| = 2$ (e.g., Histidine: CAT, CAC).
2. **Assign green list probability**: For each watermarkable codon, the green list $\mathcal{G} = \{b_g\}$ contains one base from $\mathcal{S}$, selected as the base with the highest watermark logits. Under the null hypothesis, the third base is chosen uniformly from $\mathcal{S}$, so the probability of selecting the green base is $\frac{1}{|\mathcal{S}|}$.
3. **Compute** γ: The expected proportion γ is the weighted average of $\frac{1}{|\mathcal{S}|}$ over all watermarkable codons, weighted by their counts:

$$\gamma = \frac{\sum_{k=2}^{4} (\text{number of codons with } |\mathcal{S}| = k) \times \frac{1}{k}}{\text{total watermarkable codons}}.$$

   Calculating contributions: $32 \times \frac{1}{4} = 8$ for $|\mathcal{S}| = 4$, $3 \times \frac{1}{3} = 1$ for $|\mathcal{S}| = 3$, and $24 \times \frac{1}{2} = 12$ for $|\mathcal{S}| = 2$. Total = $8 + 1 + 12 = 21$. With 59 watermarkable codons, $\gamma = \frac{21}{59} \approx 0.3559$.

This γ value serves as the baseline for watermark detection, enabling the z-score calculation to identify the presence of a watermark by comparing observed green base frequencies against this expected proportion.

E.2 Calculation of Expected Green Amino Acid Proportion (γ) for CentralMark

In the CentralMark watermarking scheme, γ represents the expected proportion of green amino acids in the translated protein sequence under the null hypothesis, assuming a uniform distribution over the 20 standard amino acids. The watermark targets the second base of codons, with the green set $\mathcal{G}_b$ comprising the two bases with the top-2 watermark logits from $\{A, C, G, T\}$, as defined in Equation (10) and detailed in Table 3. Since $\mathcal{G}_b$ is not fixed, we average γ over all possible pairs $\mathcal{G}_b \in \{\{A, C\}, \{A, G\}, \{A, T\}, \{C, G\}, \{C, T\}, \{G, T\}\}$. The process is summarized as follows:

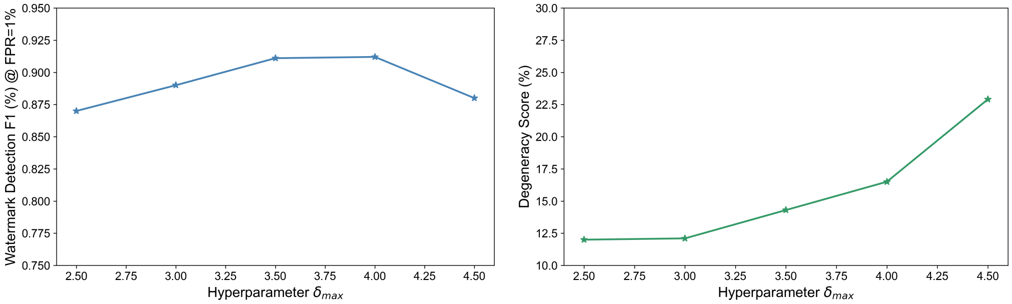
1. **Identify green amino acids**: For each $\mathcal{G}_b$, the green amino acids $\mathcal{G}_a = \{a \mid \texttt{translate}(b_1, b_2, b_3) = a, b_2 \in \mathcal{G}_b\}$ are the union of amino acids associated with the two second bases, per Table 3 (e.g., $\mathcal{G}_b = \{A, C\}$ yields 11 amino acids).
2. **Compute per-pair** γ: For each $\mathcal{G}_b$, $\gamma = \frac{\text{Number of unique amino acids in } \mathcal{G}_a}{20}$, reflecting the uniform probability $\frac{1}{20}$ per amino acid. Values range from $\frac{10}{20}$ (e.g., $\{C, G\}$) to $\frac{12}{20}$ (e.g., $\{A, G\}$).
3. **Average** γ: Assuming equal likelihood for each $\mathcal{G}_b$, the average is:

$$\gamma = \frac{1}{6}\left(\frac{11}{20} + \frac{12}{20} + \frac{12}{20} + \frac{10}{20} + \frac{10}{20} + \frac{11}{20}\right) = \frac{11}{20} = 0.55.$$

This γ serves as the baseline for detecting the CentralMark watermark in protein sequences, enabling z-score calculations to identify biased amino acid distributions.

F Influence of Hyperparameter Selection

In Figure. 4, we show the influence of δ_{max} on the watermark detection F1 and sequence degeneracy score of DNAMark. We observe that too large δ_{max} may lead to worse sequence quality measured by degeneracy, and δ_{max} in a suitable range maximizes detection F1. In experiments, we choose $\delta_{max} = 3.5$ as the default setting.



(a)

(b)

Figure 4: Hyperparamter analysis of δ_{max}

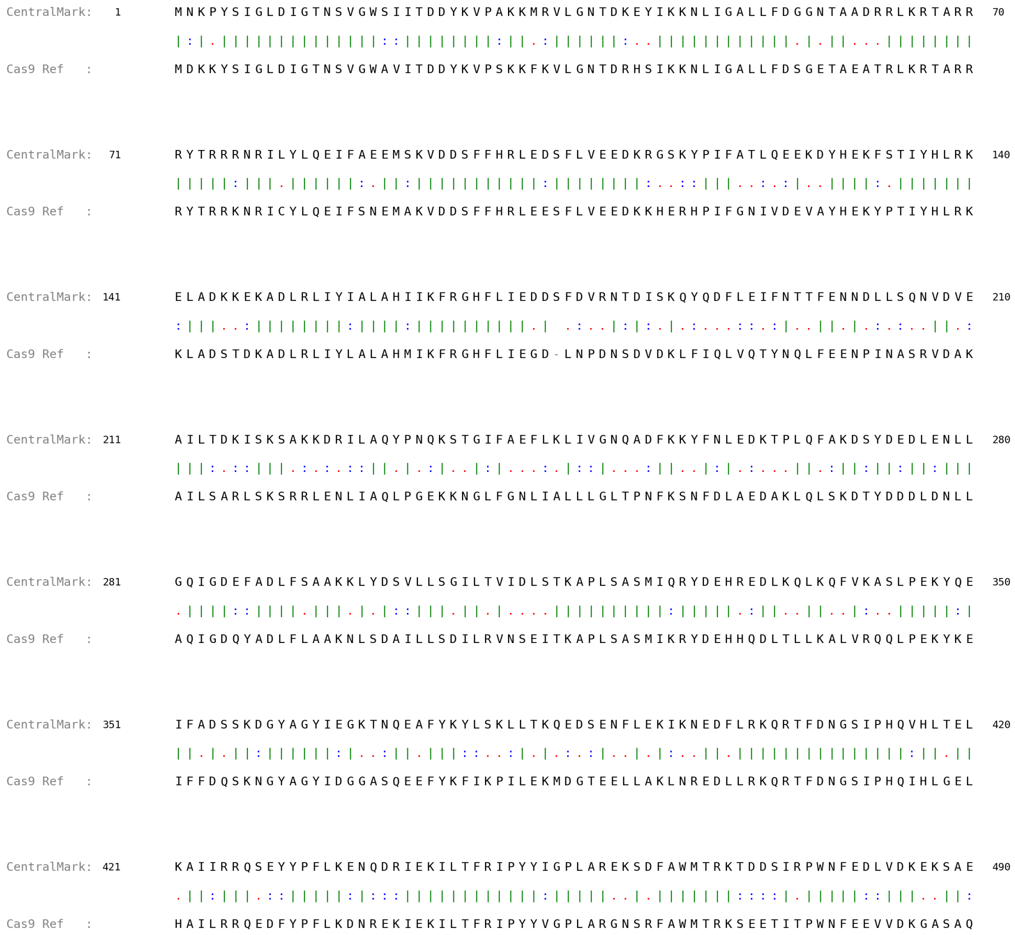
G Details of Watermark Model

Our watermark model adopts an architecture similar to SIR [39], consisting of a series of residual blocks with ReLU activation, as detailed in the code. However, our implementation incorporates additional LayerNorm layers after each residual block to stabilize training and improve convergence. Notably, the input embeddings for our model are derived from Evo2 (7B) and ESM2 (35M), leveraging their robust representations to enhance the model's ability to capture the biological semantics of DNA/protein sequences. To train the watermark model, we crawl 1000 random human coding sequences (CDS) from RefGen, subsample them to extract 20-length codons/amino acid embeddings with the Evo/Evo2 and ESM2 as input, and fine-tune the model for 200 epochs using the combination of alignment and normalization loss (Equation. 4 and 5). More details of code are included at `https://anonymous.4open.science/r/DNA_Watermark-1687/README.md`.

```
class ResidualBlock(nn.Module):
    def __init__(self, dim):
        super(ResidualBlock, self).__init__()
        self.fc = nn.Linear(dim, dim)
        self.relu = nn.ReLU()

    def forward(self, x):
        out = self.fc(x)
        out = self.relu(out)
        out = out + x
        return out

class WatermarkModel(nn.Module):
    def __init__(self, num_layers=4, input_dim=1024, hidden_dim=500,
       output_dim=4):
        super(TransformModel, self).__init__()
        self.layers = nn.ModuleList()
        self.norms = nn.ModuleList()
        self.layers.append(nn.Linear(input_dim, hidden_dim))
        self.norms.append(nn.LayerNorm(hidden_dim))
        for _ in range(num_layers - 2):
            self.layers.append(ResidualBlock(hidden_dim))
            self.norms.append(nn.LayerNorm(hidden_dim))
        self.layers.append(nn.Linear(hidden_dim, output_dim))
        self.norms.append(nn.LayerNorm(output_dim))

    def forward(self, x):
        for i in range(len(self.layers)):
            x = self.layers[i](x)
            x = self.norms[i](x)
        return x
```

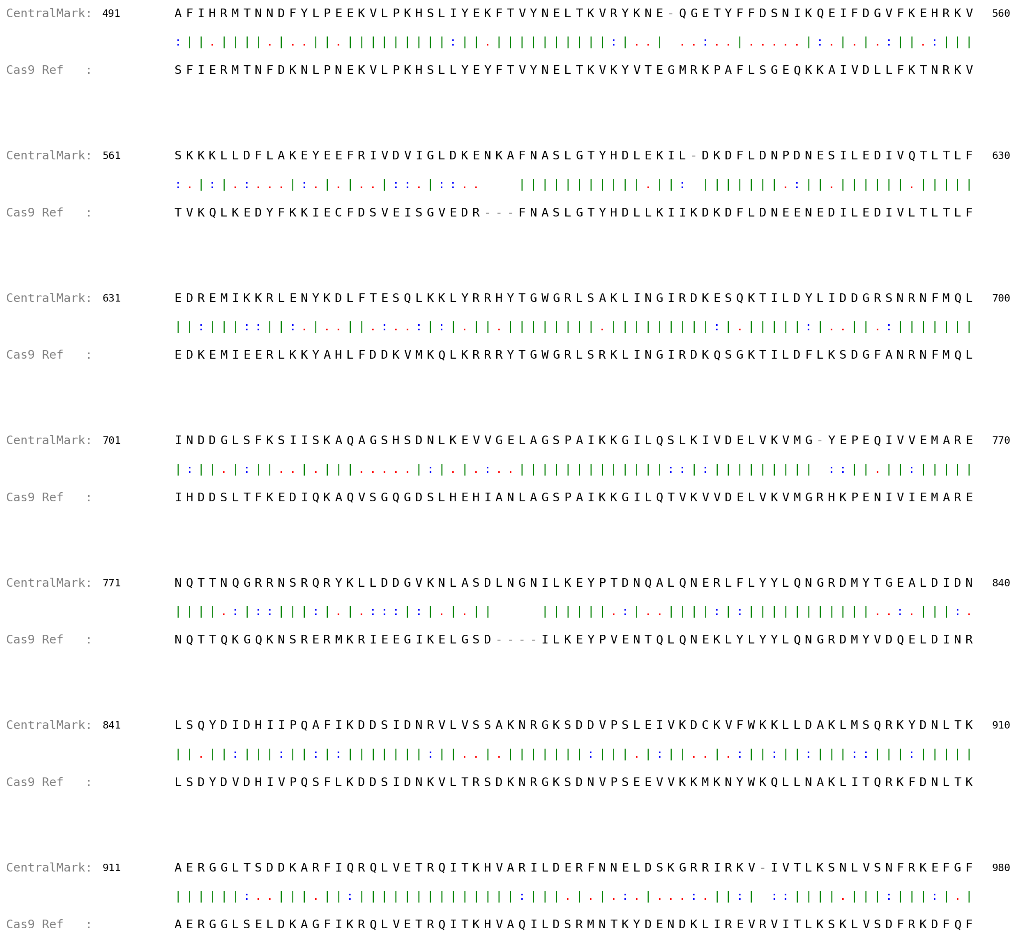
H Case Study of CRISPR-Cas9 Design with CentralMark

Here, we show the designed Cas9 sequence with CentralMark + Evo, aligned with the wild type. The total DNA similarity is 67.3%.



```
CentralMark:   1   MNKPYSIGLDIGTNSVGWSIITDDYKVPAKKMRVLGNTDKEYIKKNLIGALLFDGGNTAADRRLKRTARR  70
                   |:|.||||||||||||||::||||||||:||.:||||||:..||||||||||||.|.||...||||||||
Cas9 Ref   :       MDKKYSIGLDIGTNSVGWAVITDDYKVPSKKFKVLGNTDRHSIKKNLIGALLFDSGETAEATRLKRTARR

CentralMark:  71   RYTRRRNRILYLQEIFAEEMSKVDDSFFHRLEDSFLVEEDKRGSKYPIFATLQEEKDYHEKFSTIYHLRK  140
                   |||||:|||.||||||:.||:|||||||||||:||||||||:..::|||..:.:|..||||:.|||||||
Cas9 Ref   :       RYTRRKNRICYLQEIFSNEMAKVDDSFFHRLEESFLVEEDKKHERHPIFGNIVDEVAYHEKYPTIYHLRK

CentralMark: 141   ELADKKEKADLRLIYIALAHIIKFRGHFLIEDDSFDVRNTDISKQYQDFLEIFNTTFENNDLLSQNVDVE  210
                   :|||..:||||||||:||||:||||||||||.| .:..|:|:.|.:...::.:|..||.|.:.:..||.:
Cas9 Ref   :       KLADSTDKADLRLIYLALAHMIKFRGHFLIEGD-LNPDNSDVDKLFIQLVQTYNQLFEENPINASRVDAK

CentralMark: 211   AILTDKISKSAKKDRILAQYPNQKSTGIFAEFLKLIVGNQADFKKYFNLEDKTPLQFAKDSYDEDLENLL  280
                   |||:.::|||.:.:.::||.|.:|..|:|...:.|::|...:||..|:|.:...||.:||:||:||:|||
Cas9 Ref   :       AILSARLSKSRRLENLIAQLPGEKKNGLFGNLIALLLGLTPNFKSNFDLAEDAKLQLSKDTYDDDLDNLL

CentralMark: 281   GQIGDEFADLFSAAKKLYDSVLLSGILTVIDLSTKAPLSASMIQRYDEHREDLKQLKQFVKASLPEKYQE  350
                   .||||::||||.|||.|.|::|||.||.|....||||||||||:|||||.:||..||..|:..|||||:|
Cas9 Ref   :       AQIGDQYADLFLAAKNLSDAILLSDILRVNSEITKAPLSASMIKRYDEHHQDLTLLKALVRQQLPEKYKE

CentralMark: 351   IFADSSKDGYAGYIEGKTNQEAFYKYLSKLLTKQEDSENFLEKIKNEDFLRKQRTFDNGSIPHQVHLTEL  420
                   ||.|.||:||||||:|..:||.|||::..:|.|.:.:|..|.|:..||.|||||||||||||||:||.||
Cas9 Ref   :       IFFDQSKNGYAGYIDGGASQEEFYKFIKPILEKMDGTEELLAKLNREDLLRKQRTFDNGSIPHQIHLGEL

CentralMark: 421   KAIIRRQSEYYPFLKENQDRIEKILTFRIPYYIGPLAREKSDFAWMTRKTDDSIRPWNFEDLVDKEKSAE  490
                   .||:|||.::|||||:|:::||||||||||||:|||||..|.|||||||::::|.|||||::|||..||:
Cas9 Ref   :       HAILRRQEDFYPFLKDNREKIEKILTFRIPYYVGPLARGNSRFAWMTRKSEETITPWNFEEVVDKGASAQ
```

Figure 5: Aligning CentralMark + Evo designed Cas9 to the wild type Cas9 protein sequence.



```
CentralMark: 491   AFIHRMTNNDFYLPEEKVLPKHSLIYEKFTVYNELTKVRYKNE-QGETYFFDSNIKQEIFDGVFKEHRKV   560
                   :||.||||.|..||.|||||||||:||.||||||||||:|..| ..:..|.....|:.|.|.:||.:|||
Cas9 Ref   :        SFIERMTNFDKNLPNEKVLPKHSLLYEYFTVYNELTKVKYVTEGMRKPAFLSGEQKKAIVDLLFKTNRKV

CentralMark: 561   SKKKLLDFLAKEYEEFRIVDVIGLDKENKAFNASLGTYHDLEKIL-DKDFLDNPDNESILEDIVQTLTLF   630
                   :.|:|.:...|:.|.|..|::.|::..   |||||||||||.||: ||||||.:||.|||||.|||||
Cas9 Ref   :        TVKQLKEDYFKKIECFDSVEISGVEDR---FNASLGTYHDLLKIIKDKDFLDNEENEDILEDIVLTLTLF

CentralMark: 631   EDREMIKKRLENYKDLFTESQLKKLYRRHYTGWGRLSAKLINGIRDKESQKTILDYLIDDGRSNRNFMQL   700
                   ||:|||::||:.|..||.:..:|:|.||.||||||||.|||||||||:|.|||||:|..||.:|||||||
Cas9 Ref   :        EDKEMIEERLKKYAHLFDDKVMKQLKRRRYTGWGRLSRKLINGIRDKQSGKTILDFLKSDGFANRNFMQL

CentralMark: 701   INDDGLSFKSIISKAQAGSHSDNLKEVVGELAGSPAIKKGILQSLKIVDELVKVMG-YEPEQIVVEMARE   770
                   |:||.|:||..|.|||.....|:|.|.:..||||||||||||::|:|||||||||| ::||.||:|||||
Cas9 Ref   :        IHDDSLTFKEDIQKAQVSGQGDSLHEHIANLAGSPAIKKGILQTVKVVDELVKVMGRHKPENIVIEMARE

CentralMark: 771   NQTTNQGRRNSRQRYKLLDDGVKNLASDLNGNILKEYPTDNQALQNERLFLYYLQNGRDMYTGEALDIDN   840
                   ||||.:|::|||:|.|.:::|:|.|.||    ||||||.:|..||||:|:|||||||||||..:.|||:.
Cas9 Ref   :        NQTTQKGQKNSRERMKRIEEGIKELGSD----ILKEYPVENTQLQNEKLYLYYLQNGRDMYVDQELDINR

CentralMark: 841   LSQYDIDHIIPQAFIKDDSIDNRVLVSSAKNRGKSDDVPSLEIVKDCKVFWKKLLDAKLMSQRKYDNLTK   910
                   ||.||:|||:||:|:||||||||:||..|.||||||:|||.|:||..|.:||:||:|||::|||:|||||
Cas9 Ref   :        LSDYDVDHIVPQSFLKDDSIDNKVLTRSDKNRGKSDNVPSEEVVKKMKNYWKQLLNAKLITQRKFDNLTK

CentralMark: 911   AERGGLTSDDKARFIQRQLVETRQITKHVARILDERFNNELDSKGRRIRKV-IVTLKSNLVSNFRKEFGF   980
                   ||||||:..|||.||:||||||||||||||:|||.|.|.:.|...:.||:| ::||||.||:|||:|.|
Cas9 Ref   :        AERGGLSELDKAGFIKRQLVETRQITKHVAQILDSRMNTKYDENDKLIREVRVITLKSKLVSDFRKDFQF
```

Figure 6: Aligning CentralMark + Evo designed Cas9 to the wild type Cas9 protein sequence.

```
CentralMark: 981    YKIREVNNYHHAHDAYLNAVVAKAILTKYPQLEPEFVYGDYPKYNSYKT-RKS------ATEKLFFYSNI  1050
                    ||:||:|||||||||||||||..|::.|||:||.|||||||..|:..|. .||      ||.|.||||||
Cas9 Ref   :        YKVREINNYHHAHDAYLNAVVGTALIKKYPKLESEFVYGDYKVYDVRKMIAKSEQEIGKATAKYFFYSNI

CentralMark: 1051   MNFFKTKVTLADGTVVVKDDIEVNNDTGEIVWDKKKHFATVRKVLSYPQNNIVKKTEIQTGGFSKESILA  1120
                    ||||||::|||:|.:..:..||.|.:||||||||.:.|||||||||.||.|||||||:|||||||||||.
Cas9 Ref   :        MNFFKTEITLANGEIRKRPLIETNGETGEIVWDKGRDFATVRKVLSMPQVNIVKKTEVQTGGFSKESILP

CentralMark: 1121   HGNSDKLIPRKTKDIYLDPKKYGGFDSPIVA-YSVLVVADIKKGKAQKLKTVTELLGITIMERSRFEKNP  1190
                    ..||||||.|| ||  .|||||||||||.|| |||||||.::|||::|||:|.|||||||||||.|||||
Cas9 Ref   :        KRNSDKLIARK-KD--WDPKKYGGFDSPTVALYSVLVVAKVEKGKSKKLKSVKELLGITIMERSSFEKNP

CentralMark: 1191   SAFLESKGYLNIRADKLIILPKYSLFELENGRRRLLASAGELQKGNELALPTQFMKFLYLASRYNESKGK  1260
                    ..|||:|||..:|.|.:|.|||||||||||||:|:||||||||||||||||::::.||||||.|.:.||.
Cas9 Ref   :        IDFLEAKGYKEVRKDLIIKLPKYSLFELENGRKRMLASAGELQKGNELALPSKYVNFLYLASHYEKLKGS

CentralMark: 1261   PEEIEKKQEFVNQHVSYFDDILQLINDFSKRVILADANLEKINKLYQDNKENISVDELANNIINLFTFTS  1330
                    ||:.|:||.||.||..|.|:|::.|::||||||||||||:|:...|..:::. .:.|.|.|||:|||.|:
Cas9 Ref   :        PEDNEQKQLFVEQHKHYLDEIIEQISEFSKRVILADANLDKVLSAYNKHRDK-PIREQAENIIHLFTLTN

CentralMark: 1331   LGAPAA-FKFFDKIVDRKRYTSTKEVLNSTLIHQSITGLYETRIDLGKLGED  1382
                    |||||| ||:||..:||||||||||||::|||||||||||||||||.:||.|
Cas9 Ref   :        LGAPAAAFKYFDTTIDRKRYTSTKEVLDATLIHQSITGLYETRIDLSQLGGD
```

Figure 7: Aligning CentralMark + Evo designed Cas9 to the wild type Cas9 protein sequence.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introductions clearly summaries the contributions of this paper.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The authors discussed the limitations and potential future works in experiments and discussions.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [NA]

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: All the experimental settings are clearly described in the paper.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: The paper provides data, code, and sufficient instructions to reproduce the main results (https://anonymous.4open.science/r/DNA_Watermark-1687/README.md).

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: The training and test details are specified in the experiments.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: Yes, the error bars are provided in figures.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).

- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The information on computer resources is provided in experiments.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research conducted in the paper conform the NeurIPS code of ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: The paper discusses the broad impacts in the introduction and conclusions.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: All the related assets used in the paper are well credited.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The benchmark datasets will be release after further screening and the related documentation with be provided.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: The paper does not use LLMs as a component of the core method.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.