Exons-Detect: Identifying and Amplifying Exonic Tokens via Hidden-State Discrepancy for Robust AI-Generated Text Detection

Anonymous ACL submission

Abstract

The rapid advancement of large language models has increasingly blurred the boundary between human-written and AI-generated text, raising societal risks such as misinformation dissemination, authorship ambiguity, and threats to intellectual property rights. These concerns highlight the urgent need for effective and reliable detection methods. While existing training-free approaches often achieve strong performance by aggregating token-level signals into a global score, they typically assume uniform token contributions, making them less robust under short sequences or localized token modifications. To address these limitations, we propose Exons-Detect, a training-free method for AI-generated text detection based on an exon-aware token reweighting perspective. Exons-Detect identifies and amplifies informative exonic tokens by measuring hidden-state discrepancy under a dual-model setting, and computes an interpretable translation score from the resulting importance-weighted token sequence. Empirical evaluations demonstrate that Exons-Detect achieves state-of-the-art detection performance and exhibits strong robustness to adversarial attacks and varying input lengths. In particular, it attains a 2.2% relative improvement in average AUROC over the strongest prior baseline on DetectRL.

1 Introduction

The rapid advancement of large language models (LLMs) has enabled them to generate highly fluent and coherent text, substantially narrowing the observable gap between AI-generated text and human writing. While such progress has catalyzed significant technological breakthroughs across both industry and academia, it has simultaneously introduced pressing societal risks, including misinformation dissemination, authorship ambiguity, and threats to intellectual property rights (Ahmed et al., 2021; Adelani et al., 2019; Guo et al., 2021). Prior studies (Clark et al., 2021) further reveal that humans perform only marginally above random chance in distinguishing AI-generated from human-written text. This limitation highlights the urgent need for effective and reliable detection methods.

Figure 1: Advantages of Our Method Exons-Detect.

Existing detection approaches can be broadly categorized into training-based and training-free methods. Training-based methods require large-scale labeled data and rely on supervised deep models to learn implicit textual representations, which limits their scalability and cross-domain generalization. In contrast, training-free methods compute token-level statistics, such as Binoculars (Hans et al., 2024), under the generative distributions of proxy LLMs, and typically aggregate these signals by averaging across token positions to form a global detection score. While effective in many settings, this uniform aggregation assumes equal contribution from all tokens, making such methods less robust when token sequences are short, or localized token modifications are introduced. Consequently, truly informative tokens can be overwhelmed by less relevant ones, motivating the need for a mechanism that differentiates token-level functional roles rather than treating all tokens uniformly.

Inspired by molecular biology, we view a text se-

quence as a gene fragment composed of exons and introns. Exons are directly involved in protein translation, while introns play a secondary role. Analogously, tokens with large hidden-state discrepancy are treated as exonic tokens that carry stronger discriminative signals, whereas the remaining tokens are regarded as intronic. During transcription, both exons and introns are preserved in the pre-mRNA, corresponding to uniform initial weighting of all tokens. Splicing then removes introns and emphasizes exons in mature mRNA. Mirroring this process, we identify exonic tokens via hidden-state discrepancy and amplify their contributions through assigning additional weights. The final detection is achieved by aggregating the reweighted token sequence to compute the translation score. This exon-aware reweighting captures intrinsic differences between AI-generated and human-written texts in a fine-grained and interpretable manner.

Building on this intuition, we propose Exons-Detect, a novel training-free method for AI-generated text detection. Given an input sequence, we extract hidden representations at each token position and quantify their discrepancy under a pair of proxy LLMs. Tokens whose representation discrepancy exceeds a predefined discrepancy threshold are identified as exonic tokens. We map these discrepancies through a nonlinear function to obtain additional weights for exonic tokens, which are integrated with the initial weights for computing the translation score. Finally, Exons-Detect determine the detect result by comparing the translation score against a decision threshold, effectively amplifying the discriminative signals carried by exonic tokens.

Exons-Detect achieves state-of-the-art performance across multiple publicly available detection benchmarks. In particular, Exons-Detect achieves a relative improvement of 2.2% in average AUROC over the strongest existing baseline DNA-DetectLLM on DetectRL. Moreover, Exons-Detect exhibits strong robustness against various adversarial attacks and across different input lengths. Efficiency experiments further demonstrate that Exons-Detect offers rapid detection capability, making it well suited for large-scale, real-time detection. Our contributions are summarized as follows:

- Inspired by the distinct roles of exons and introns in gene fragments, we introduce the notions of exonic tokens and intronic tokens in text sequences, emphasizing that different tokens contribute unequally to detection.
- We propose Exons-Detect, a novel training-free method that identifies exonic tokens and amplifies their importance to capture more informative source-specific signals, enabling robust AI-generated text detection.
- Extensive experiments demonstrate that Exons-Detect provides a robust, efficient, and broadly generalizable solution for AI-generated text detection, delivering consistent improvements across 3 public benchmarks, 2 adversarial attacks, and varying input lengths, with inference latency below 0.8 s per sample.

2 Related Work

Training-based methods typically leverage deep learning models to supervisedly learn latent textual features that distinguish AI-generated text from human-written content. Early work by OpenAI (Solaiman et al., 2019) developed a RoBERTa-based classifier. RADAR (Hu et al., 2023) incorporated adversarial training to improve robustness against paraphrased inputs. Biscope (Guo et al., 2024a) introduced a bidirectional cross-entropy loss to optimize classifier performance. DeTeCtive (Guo et al., 2024b) and DETree (He et al., 2025) mapped texts from different sources or constructions into high-dimensional representation spaces, followed by similarity-based detection. Training-based detectors often overfit in-distribution patterns and degrade sharply under distribution shifts (Chakraborty et al., 2023; Uchendu et al., 2020), motivating increasing interest in universal and reliable training-free detection.

Training-free methods distinguish texts by estimating statistical scores from the generative probabilities under proxy LLMs. Traditional approaches including LogRank (Gehrmann et al., 2019), Likelihood (Hashimoto et al., 2019), and Entropy (Ippolito et al., 2020) quantified generative uncertainty by averaging probability rank, log-likelihood, and entropy under a proxy model. DetectGPT (Mitchell et al., 2023) established a new paradigm by introducing phrase-level perturbations to evaluate distributional curvature. Fast-DetectGPT (Bao et al., 2024) proposed an optimized sampling strategy for estimating conditional probability curvature, achieving substantial gains in both speed and accuracy compared to DetectGPT. Binoculars (Hans et al., 2024) mitigated high-perplexity human texts by using the ratio of log-perplexity

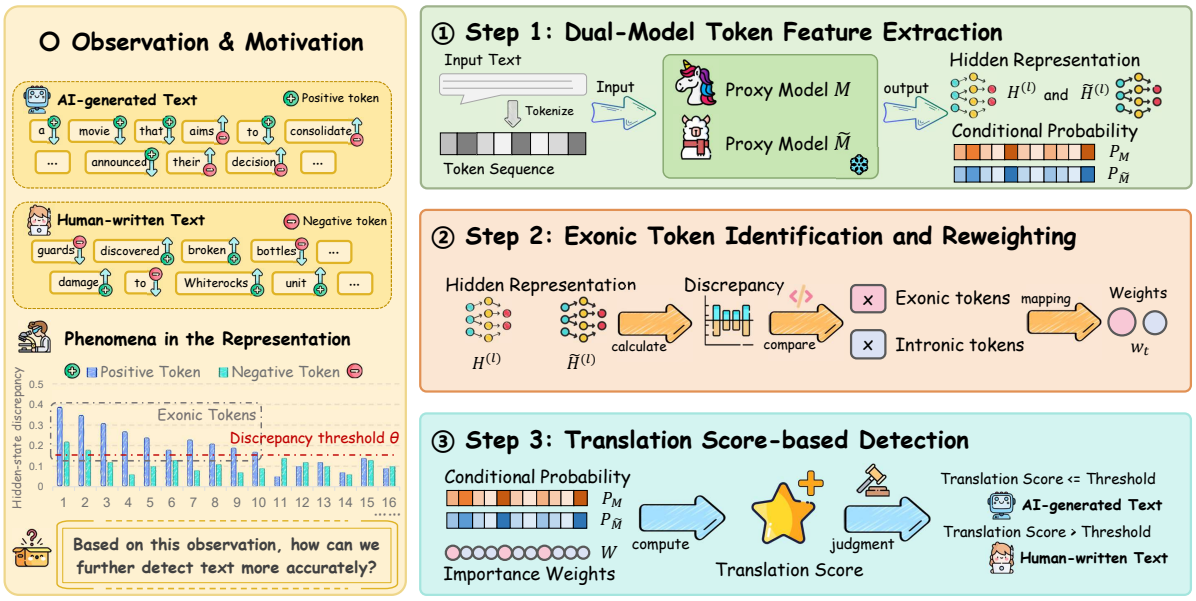

Figure 2: Overview of Exons-Detect.

to cross-perplexity, while DNA-DetectLLM (Zhu et al., 2025b) further enhanced this score with a mutation-repair mechanism, achieving more robust performance. Lastde (Xu et al., 2024) captured local textual characteristics via Diversity Entropy, while IRM (Liu et al., 2025) leveraged discrepancies in generative probabilities before and after reinforcement learning from human feedback (RLHF) to capture the divergence.

3 Methodology

3.1 Preliminaries

Log-perplexity and Cross-perplexity. Log-perplexity quantifies the average token-level negative log-likelihood under a single proxy model, whereas cross-perplexity measures the average per-token cross-entropy computed across two models. To model variation in token-wise importance, we further introduce weighted log-perplexity and weighted cross-perplexity, which incorporate token-specific importance weights into their computation:

$$\log \mathrm{PPL}_M^W(s) = -\sum_{t=1}^{T} w_t \log P_M(x_t),$$
$$\log \text{X-PPL}_{M,\tilde{M}}^W(s) = -\sum_{t=1}^{T} w_t \, P_M(x_t) \, \log P_{\tilde{M}}(x_t), \quad (1)$$

where s denotes an input sequence of length T, w_t denotes the normalized weight, and $P_M(x_t)$ and $P_{\tilde{M}}(x_t)$ denote the conditional generation distributions of the t-th token under models M and $\tilde{M}$.

Observation. In training-free detection, scores such as Binoculars are computed by averaging token-level contributions, where AI-generated texts typically yield lower scores than human-written texts. Accordingly, in AI-generated text, tokens whose individual contributions fall below the decision threshold tend to increase class separability, whereas in human-written text, tokens with contributions above the threshold play an analogous role. We refer to such tokens as positive tokens and, by analyzing their associated hidden-state representations that may encode source-related signals (Chen et al., 2025), as shown in Figure 2. When we examine tokens whose hidden-state discrepancy between model M and $\tilde{M}$ exceeds a discrepancy threshold θ, the number of tokens that increase class separability is markedly larger than the number that decreases it. This asymmetric enrichment suggests that tokens with high-discrepancy more often carry source-relevant signals, whereas other tokens contribute more weakly or inconsistently.

Motivation. Motivated by this observation, we refer to tokens with high hidden-state discrepancies as **exonic tokens**, and the remaining ones as **intronic tokens**, reflecting their different relevance to the text's origin. This naturally suggests a simple and effective strategy: during detection, we identify high-discrepancy exonic tokens and amplify their contributions. By reweighting these tokens, the final detection score is encouraged to move further toward the correct side, resulting in robust and

separable detection. See Appendix A for analysis.

3.2 Overview of Exons-Detect

Figure 2 presents the overall workflow of Exons-Detect, including three steps:

Step 1: Dual-Model Token Feature Extraction. Given an input sequence, we extract token-level hidden representations and generative probability distributions under a reference model M and a paired model $\tilde{M}$.

Step 2: Exonic Token Identification and Reweighting. We measure hidden-state discrepancy between M and $\tilde{M}$ at each token position to identify exonic tokens, and map these discrepancies to additional token-level weights.

Step 3: Translation Score-based Detection. We introduce a translation score by aggregating token contributions according to their weights and probability distributions, and compare it against a decision threshold to determine the detection result.

3.3 Dual-Model Token Feature Extraction

Given an input sequence $s = (x_1, x_2, x_3, \ldots, x_T)$ of length T, we feed it into a proxy LLMs pair: M and $\tilde{M}$. Each model consists of L transformer layers. For model M, we extract the hidden representations at each token position and layer as

$$\mathbf{H}^{(l)} = \left[\mathbf{h}_1^{(l)}, \mathbf{h}_2^{(l)}, \ldots, \mathbf{h}_T^{(l)}\right], \quad l = 1, \ldots, L, \tag{2}$$

where $\mathbf{h}_t^{(l)} \in \mathbb{R}^d$ denotes the hidden representation of the t-th token at layer l. Similarly, model $\tilde{M}$ produces the corresponding hidden representations

$$\tilde{\mathbf{H}}^{(l)} = \left[\tilde{\mathbf{h}}_1^{(l)}, \tilde{\mathbf{h}}_2^{(l)}, \ldots, \tilde{\mathbf{h}}_T^{(l)}\right]. \tag{3}$$

In addition to hidden representations, both models provide token-level generative probabilities. Specifically, at each token position t, model M defines a conditional generation distribution $P_M(x_t) = P_M(\cdot \mid x_{<t})$ over the vocabulary, and $\tilde{M}$ defines $P_{\tilde{M}}(x_t) = P_{\tilde{M}}(\cdot \mid x_{<t})$, where $x_{<t}$ denotes the preceding context.

3.4 Exonic Token Identification and Reweighting

To quantify the representational discrepancy at each token position, we measure the hidden-state discrepancy using the cosine distance. For the t-th token, we aggregate its hidden representations across all L layers from models M and $\tilde{M}$, and define the token-level discrepancy as

$$\delta_t = \frac{1}{L} \sum_{l=1}^{L} \left(1 - \cos\left(\mathbf{h}_t^{(l)},\ \tilde{\mathbf{h}}_t^{(l)}\right)\right), \tag{4}$$

where $\mathbf{h}_t^{(l)}$ and $\tilde{\mathbf{h}}_t^{(l)}$ denote the hidden representations of the t-th token at layer l produced by models M and $\tilde{M}$, respectively.

Based on the magnitude of δ_t, we identify exonic tokens by applying a significance-level criterion. Specifically, a token is classified as an exonic token if its hidden-state discrepancy exceeds a predefined discrepancy threshold θ, and as an intronic token otherwise:

$$x_t = \begin{cases} \text{exonic token}, & \text{if } \delta_t > \theta, \\ \text{intronic token}, & \text{if } \delta_t \leq \theta. \end{cases} \tag{5}$$

To further emphasize the contribution of exonic tokens, we remap their hidden-state discrepancies into importance weights W. Formally, we introduce a nonlinear mapping function $g(\cdot)$ to obtain a token-specific additional weight $\Delta w_t = g(\delta_t)$:

$$g(\delta_t) = 1 - \exp\big(-\alpha\,(\delta_t - \theta)_+\big), \tag{6}$$

where $(\cdot)_+ = \max(\cdot, 0)$ denotes the positive part operator, which ensures that intronic tokens with discrepancies below the discrepancy threshold θ receive zero additional weight.

This nonlinear mapping smoothly amplifies the weights of exonic tokens according to their hidden-state discrepancies, while avoiding excessive emphasis on individual tokens, thereby preserving robustness. The final importance weights is formed by summing the initial uniform weight and the exonic weight increments and normalizing the result, given by:

$$w_t = \frac{1 + \Delta w_t}{\sum_{i=1}^{T} (1 + \Delta w_i)}, \quad t = 1, 2, \ldots, T. \tag{7}$$

3.5 Translation Score-based Detection

We introduce a translation score that integrates the importance weights of both exonic and intronic tokens with their conditional probabilities. Prior work (Hans et al., 2024) has shown that the ratio between log-perplexity and cross-perplexity provides a strong discriminative signal for AI-generated text detection. Following this insight, we define the initial translation score as the ratio of the weighted log-perplexity to the weighted cross-perplexity:

$$R(s) = \frac{\log \mathrm{PPL}_M^W(s)}{\log \text{X-PPL}_{M,\tilde{M}}^W(s)}. \tag{8}$$

To further refine the translation score, we incorporate the mutation-repair mechanism proposed in DNA-DetectLLM (Zhu et al., 2025b) as a complementary component. This mechanism captures the intrinsic discrepancy between an input sequence and the ideal AI-generated sequence by quantifying the difficulty of iteratively repairing mutated tokens. Importantly, the repair process operates under the same exon-aware importance weights. Incorporating this mechanism, the final translation score is formulated as:

$$R(s) = \frac{\log \mathrm{PPL}_M^W(s) + \log \mathrm{PPL}_M^W(\hat{s})}{\log \text{X-PPL}_{M,\tilde{M}}^W(s)}, \quad (9)$$

where $\hat{s}$ denotes the ideal AI sequence, constructed by selecting the token with the maximum generation probability.

For AI-generated text, exonic tokens contribute to shifting the translation score toward smaller values, whereas for human-written text, exonic tokens contribute to increasing the translation score. Accordingly, the detection result for an input sequence is determined as follows:

$$\mathcal{D}(s) = \begin{cases} \text{Human-written Text}, & R(s) > \tau, \\ \text{AI-generated Text}, & R(s) \leq \tau. \end{cases} \quad (10)$$

4 Experiments

4.1 Experimental Setup

Datasets. To evaluate the detection performance of our method under realistic deployment scenarios, we conduct experiments on three diverse and high-quality public benchmarks: M4 (Wang et al., 2024), RealDet (Zhu et al., 2025a), and DetectRL (Wu et al., 2024). In particular, we conduct evaluations on DetectRL using the Multi-LLM and Multi-Domain settings to examine generalization across models and domains.

Baselines. For training-based detectors, we consider OpenAI-D (Solaiman et al., 2019), BiScope (Guo et al., 2024a), and R-Detect (Song et al., 2025). For training-free approaches, we include classical zero-shot detectors such as Likelihood (Hashimoto et al., 2019), LogRank (Gehrmann et al., 2019), and Entropy (Ippolito et al., 2020), as well as more recent representative methods, including DetectGPT (Mitchell et al., 2023), Fast-DetectGPT (Bao et al., 2024), Binoculars (Hans et al., 2024), and Lastde++ (Xu et al., 2024). In addition, we compare against the latest and strongest baselines, IRM (Liu et al., 2025) and DNA-DetectLLM (Zhu et al., 2025b).

Metrics. We evaluate detection performance using the area under the receiver operating characteristic curve (AUROC) and the F1 score.

Implementation details. To ensure a fair comparison, we train all training-based detectors on the HC3 dataset (Guo et al., 2023), which is disjoint from all evaluation benchmarks. Prior work (Bao et al., 2025) has shown that the performance of training-free methods can vary substantially under different combinations of LLMs. To eliminate this factor, we standardize the reference (scoring) model across all methods by using Falcon-7B-Instruct (Penedo et al., 2023) to compute token-level generation probabilities. Moreover, Fast-DetectGPT, Binoculars, Lastde++, IRM, DNA-DetectLLM, and Exons-Detect all employ Falcon-7B (Penedo et al., 2023) as the paired model when computing their respective detection scores. We set the discrepancy threshold $\theta = 0.15$ and the mapping slope $\alpha = 10$ by default. More details see Appendix B.

4.2 Main Results

Table 1 compares the detection performance of Exons-Detect against other baselines across three public benchmarks. Overall, Exons-Detect exhibits strong detection accuracy and robust generalization, delivering consistently competitive results on M4, DetectRL, and RealDet. It achieves an average AUROC of 92.14% and an average F1 score of 87.72%, outperforming the latest baseline DNA-DetectLLM by 1.4% and 0.8%. Notably, Exons-Detect is the only method whose AUROC exceeds 90% on all evaluated datasets, further highlighting its reliability for real-world deployment under different text distributions.

A closer inspection reveals that most baselines exhibit substantial performance variance across datasets, indicating sensitivity to changes in text source and distribution. For training-based detectors, the mismatch between the training corpus and the evaluation benchmarks leads to limited OOD generalization, with AUROC typically remaining below 80%. Among training-free approaches, representative methods such as Fast-DetectGPT, Binoculars, and Lastde++ perform strongly on M4 and RealDet, yet their performance

Detectors	M4		DetectRL Multi-LLM		DetectRL Multi-Domain		RealDet		Avg.	
	AUROC	F_1	AUROC	F_1	AUROC	F_1	AUROC	F_1	AUROC	F_1
Training-based Methods										
OpenAI-D	77.51	71.18	78.15	71.90	74.60	70.03	84.75	77.47	78.75	72.65
Biscope	79.74	73.08	79.97	73.20	76.52	71.64	92.88	86.90	82.28	76.21
R-Detect	61.91	67.14	67.40	66.56	79.19	73.38	65.93	67.72	68.61	68.70
Training-free Methods										
Entropy	83.72	79.10	64.30	71.92	47.82	69.24	75.42	74.72	67.82	73.75
Likelihood	85.77	78.38	66.82	66.71	48.96	66.69	85.35	79.75	71.73	72.88
LogRank	87.50	80.70	67.30	66.71	50.55	66.69	86.28	80.69	72.91	73.70
DetectGPT	73.13	70.11	49.57	66.67	34.67	66.67	78.69	73.80	59.02	69.31
Fast-DetectGPT	89.77	84.12	82.26	75.93	74.98	68.91	93.25	90.00	85.07	79.74
Binoculars	90.00	87.40	83.21	82.87	77.45	80.20	93.64	90.51	86.08	85.25
Lastde++	91.43	84.97	75.36	69.24	67.30	66.67	93.90	89.41	82.00	77.57
IRM	71.85	70.75	83.02	76.46	**91.51**	84.05	77.70	76.62	81.02	76.97
DNA-DetectLLM	91.74	87.72	88.97	84.85	88.23	84.94	94.48	90.58	90.86	87.02
Exons-Detect	**92.43**	**88.05**	**90.67**	**84.95**	90.46	**86.59**	**94.98**	**91.30**	**92.14**	**87.72**

Table 1: Detection performance (AUROC and F1 score) on public benchmark datasets.

degrades sharply on the more challenging DetectRL setting. IRM excels on DetectRL, reaching an AUROC of 91.51% under the Multi-Domain setting, but fails to maintain comparable performance on M4 and RealDet. We conjecture that this behavior arises from IRM's reliance on probability discrepancies induced by RLHF, which become weaker and harder to exploit when texts are generated by less strongly aligned open-source LLMs. While DNA-DetectLLM partially alleviates this issue, it still falls noticeably behind Exons-Detect on DetectRL. In particular, Exons-Detect achieves AUROC gains of 1.9% under Multi-LLM and 2.5% under Multi-Domain. We attribute this consistent advantage to Exons-Detect's ability to reliably identify exonic tokens from hidden-state discrepancies across diverse text distributions, enabling it to extract more precise source-relevant signals and thereby counteract the cross-dataset bias.

4.3 Robustness

4.3.1 Robustness against Various Attacks

In realistic scenarios, input texts are often non-pristine and may be subject to adversarial attacks. AI-generated texts can undergo paraphrasing attacks to evade detection, while human-written texts are frequently refined using advanced LLMs through polishing attacks. Paraphrasing attacks employ DIPPER (Krishna et al., 2023) to rephrase AI-generated texts, while polish attacks apply GPT-4o-based polishing to human-written texts. Figure 3 shows the AUROC of Exons-Detect and baselines on DetectRL under various attack settings.

Experimental results demonstrate that Exons-Detect exhibits strong robustness across these attack scenarios. Specially, under both paraphrasing and polishing attacks on DetectRL, Exons-Detect consistently outperforms all competing baselines, maintaining a clear performance advantage. A notable observation is that training-based detectors are substantially more vulnerable to adversarial attacks, as evidenced by BiScope's AUROC degrading to near-random performance under paraphrasing and polishing attacks. In contrast, strong training-free baselines exhibit substantially higher robustness to paraphrasing attacks, likely because DIPPER-based paraphrasing mainly introduces lexical and syntactic variations without fundamentally altering underlying statistical features.

However, polishing attacks pose a greater challenge by injecting advanced LLM alignment and generation signals into human-written texts, blurring the boundary between human-written and AI-generated content. This leads to noticeable performance degradation for several baselines, including Fast-DetectGPT and IRM. In contrast, Exons-Detect preserves a high level of detection accuracy even under polishing attacks, highlighting its superior robustness. We attribute this robustness to its ability to exploit hidden-state discrepancies to identify critical exonic tokens, thereby suppressing the adverse impact of localized textual modifications on the global detection score.

4.3.2 Robustness on Different Lengths

Prior studies (Bao et al., 2024; Tian et al., 2024) have demonstrated that detection performance is highly sensitive to input length, with shorter texts being substantially more difficult to identify. To systematically examine this effect, we truncate in-

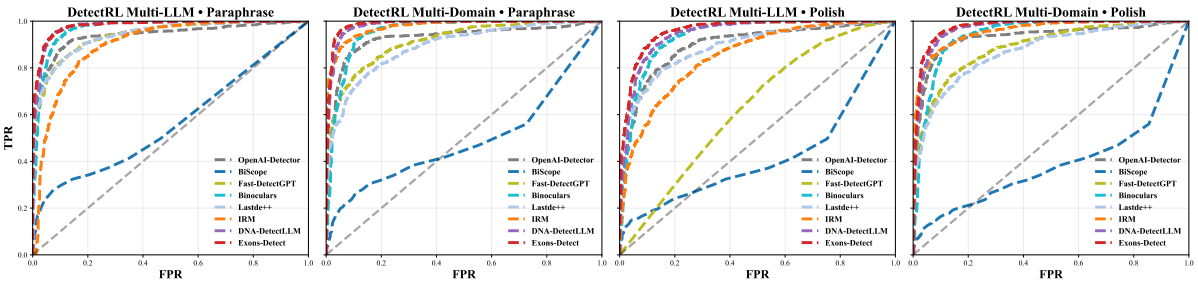

Figure 3: Detection performance (AUROC curves) against various attacks.

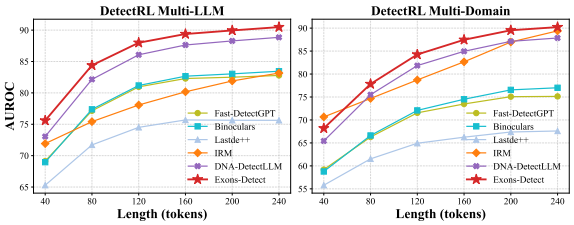

Figure 4: Detection performance on different length.

Setting ↓	M4	Multi-L	Multi-D	RealDet	Avg.
Exons-Detect	**92.43**	**90.67**	**90.46**	**94.98**	**92.14**
w/o $\log \mathrm{PPL}_M^W(\hat{s})$	91.28	85.32	80.46	93.97	87.76
w/o $g(\cdot)$	91.92	89.63	88.66	94.64	91.21
Model Family ↓					
Falcon-7B	**92.43**	90.67	90.46	**94.98**	**92.14**
LLaMA-7B	88.64	92.34	90.31	94.47	91.44
Mistral-v0.1-7B	90.41	91.42	85.77	93.00	90.15
LLaMA-3.2-1B	90.50	**92.87**	**90.82**	92.58	91.69

Table 2: Ablation study results under different settings.

put texts to several predefined lengths and evaluate method robustness under varying length constraints. Figure 4 compares the robustness of Exons-Detect against five strong baselines on DetectRL across different input lengths. The results show that Exons-Detect consistently outperforms all competing baselines across predefined lengths, achieving an average improvement of 2.7% over DNA-DetectLLM and 6.4% over IRM. While all methods benefit from increased text length, Exons-Detect exhibits markedly stronger performance in the short-text regime. These results highlight that Exons-Detect captures precise source-related signals from limited text, leading to superior robustness under short-length conditions.

4.4 Ablation Studies

Impact of the $g(\cdot)$ and $\log \mathrm{PPL}_M^W(\hat{s})$. Table 2 evaluates the impact of removing the nonlinear mapping function $g(\cdot)$ or the computation of $\log \mathrm{PPL}_M^W(\hat{s})$ (i.e., the mutation-repair mechanism). Removing $g(\cdot)$ corresponds to assigning additional weights as $\Delta w_t = \delta_t$, while removing $\log \mathrm{PPL}_M^W(\hat{s})$ refers to performing detection using the initial translation score. Overall, both $g(\cdot)$ and $\log \mathrm{PPL}_M^W(\hat{s})$ make essential contributions to detection performance and are indispensable components of Exons-Detect. Specifically, removing $g(\cdot)$ and $\log \mathrm{PPL}_M^W(\hat{s})$ results in average AUROC drops of 1.0% and 4.4%. These degradations indicate that properly mapping hidden-state discrepancies to token importance weights, as well as leveraging the mutation-repair mechanism to further capture class-discriminative differences, are effective and necessary for achieving strong performance.

Impact of the proxy LLM pair. Table 2 also reports Exons-Detect's performance across different LLM pairings, including Falcon-7B-Instruct with Falcon-7B, LLaMA-2-7B with LLaMA-7B, Mistral-v0.1-7B-Instruct with Mistral-v0.1-7B, and LLaMA-3.2-1B-Instruct with LLaMA-3.2-1B. Overall, Exons-Detect achieves consistently strong performance across all model combinations, with only modest variation and an average AUROC exceeding 90% in every setting. Notably, the "LLaMA-3.2-1B-Instruct + LLaMA-3.2-1B" pairing slightly outperforms "Falcon-7B-Instruct + Falcon-7B" on DetectRL, attaining AUROC scores of 92.87% and 90.82%. These results indicate that while certain LLM combinations can offer incremental gains, the effectiveness of Exons-Detect does not hinge on a specific model pairing. Instead, the method remains robust across diverse LLM families, and can be further enhanced by selecting better LLM pairings.

4.5 Hyperparameter Sensitivity

This subsection analyzes the impact of both hyperparameters (discrepancy threshold θ and mapping slope α) and a structural parameter (hidden layers L) on detection performance. Figure 5 reports the detection performance of Exons-Detect under different parameter settings across multiple datasets.

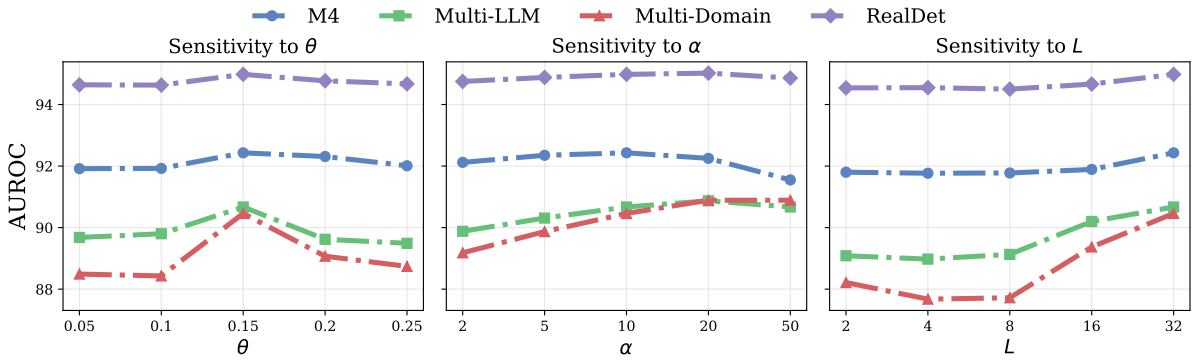

Figure 5: Detection performance of Exons-Detect under different parameter settings.

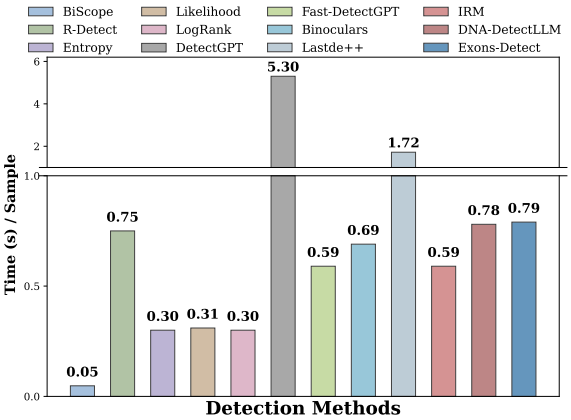

Figure 6: Time costs for processing a single sample.

Overall, Exons-Detect exhibits low sensitivity to hyperparameter choices, maintaining stable detection performance across a broad range of settings. Specifically, varying either θ or α results in performance fluctuations typically within 1.0%, while consistently outperforming existing baselines under all configurations. For the threshold θ, we observe that extreme values lead to inferior performance compared to values around $\theta = 0.15$. This behavior is intuitive: overly small thresholds tend to treat most tokens as exonic tokens, excessively amplifying noise, whereas overly large thresholds fail to emphasize informative tokens, diminishing the benefit of reweighting. For the mapping slope α, performance improves noticeably when $\alpha \geq 10$, indicating that sufficiently steep mappings are necessary to translate moderate hidden-state discrepancies into effective importance weights.

Reducing L leads to a clear degradation in detection performance. For instance, on DetectRL under the Multi-LLM and Multi-Domain settings, reducing L from 32 to 4 results in relative AUROC drops of 1.9% and 3.1%. Across all datasets, Exons-Detect consistently achieves its best performance when utilizing 32 hidden layers. These results highlight that fully exploiting representational discrepancies across the entire depth of the model is crucial for robust detection.

4.6 Efficiency Analysis

Faster detection is critical for real-world deployment and monitoring. Figure 6 compares the per-text runtime of all methods. To control for the effect of text length, we sample 1,000 long samples from RealDet and truncate each to 300 tokens, reporting the average processing time per text. Training-based detectors (e.g., BiScope) achieve the lowest inference latency, but at the cost of substantial training overhead. Among training-free methods, classical detectors such as Likelihood are relatively fast (around 0.3 s per text) since they require only a single forward pass, but their detection accuracy did not meet our requirements. Exons-Detect and representative baselines (e.g. Binoculars) incur two forward passes but still run within 0.8 s. Within this efficiency regime, Exons-Detect delivers better detection performance, offering a favorable accuracy-latency trade-off.

5 Conclusion

This paper proposes Exons-Detect, a novel training-free method for AI-generated text detection that operates by identifying and reweighting exonic tokens. Extensive experiments demonstrate that Exons-Detect consistently achieves SOTA performance across diverse evaluation settings, while exhibiting strong robustness to adversarial attacks and varying input lengths. We hope our work offers new insights for AI-generated text detection and plan to further explore token-level contribution modeling to enhance detection performance.

Limitations

Prior studies (Chen et al., 2025) have shown that hidden-state representations may carry signals related to text provenance. In Exons-Detect, we employ cosine distance to efficiently measure token-level hidden-state discrepancies for assessing token importance. We believe that more fine-grained and more specific discrepancy evaluations could better exploit source-related information and lead to more accurate detection results, which represents a potential direction for further improvement from a token-level perspective.

References

David Ifeoluwa Adelani, Hao Thi Mai, Fuming Fang, Huy Hoang Nguyen, Junichi Yamagishi, and Isao Echizen. 2019. Generating sentiment-preserving fake online reviews using neural language models and their human- and machine-based detection. In *International Conference on Advanced Information Networking and Applications*.

Alim Al Ayub Ahmed, Ayman Aljabouh, Praveen Kumar Donepudi, and Myung Suh Choi. 2021. Detecting fake news using machine learning : A systematic literature review. *Preprint*, arXiv:2102.04458.

Guangsheng Bao, Yanbin Zhao, Juncai He, and Yue Zhang. 2025. Glimpse: Enabling white-box methods to use proprietary models for zero-shot LLM-generated text detection. In *The Thirteenth International Conference on Learning Representations*.

Guangsheng Bao, Yanbin Zhao, Zhiyang Teng, Linyi Yang, and Yue Zhang. 2024. Fast-detectGPT: Efficient zero-shot detection of machine-generated text via conditional probability curvature. In *The Twelfth International Conference on Learning Representations*.

Souradip Chakraborty, Amrit Singh Bedi, Sicheng Zhu, Bang An, Dinesh Manocha, and Furong Huang. 2023. On the possibilities of ai-generated text detection. *Preprint*, arXiv:2304.04736.

Xin Chen, Junchao Wu, Shu Yang, Runzhe Zhan, Zeyu Wu, Ziyang Luo, Di Wang, Min Yang, Lidia S. Chao, and Derek F. Wong. 2025. Repreguard: Detecting llm-generated text by revealing hidden representation patterns. *Preprint*, arXiv:2508.13152.

Elizabeth Clark, Tal August, Sofia Serrano, Nikita Haduong, Suchin Gururangan, and Noah A. Smith. 2021. All that's 'human' is not gold: Evaluating human evaluation of generated text. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 7282–7296, Online. Association for Computational Linguistics.

Sebastian Gehrmann, Hendrik Strobelt, and Alexander Rush. 2019. GLTR: Statistical detection and visualization of generated text. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 111–116, Florence, Italy. Association for Computational Linguistics.

Biyang Guo, Xin Zhang, Ziyuan Wang, Minqi Jiang, Jinran Nie, Yuxuan Ding, Jianwei Yue, and Yupeng Wu. 2023. How close is chatgpt to human experts? comparison corpus, evaluation, and detection. *Preprint*, arXiv:2301.07597.

Hanxi Guo, Siyuan Cheng, Xiaolong Jin, Zhuo Zhang, Kaiyuan Zhang, Guanhong Tao, Guangyu Shen, and Xiangyu Zhang. 2024a. Biscope: Ai-generated text detection by checking memorization of preceding tokens. In *Advances in Neural Information Processing Systems*, volume 37, pages 104065–104090. Curran Associates, Inc.

Xun Guo, Shan Zhang, Yongxin He, Ting Zhang, Wanquan Feng, Haibin Huang, and Chongyang Ma. 2024b. Detective: Detecting ai-generated text via multi-level contrastive learning. In *Advances in Neural Information Processing Systems*, volume 37, pages 88320–88347. Curran Associates, Inc.

Zhiwei Guo, Yu Shen, Ali Kashif Bashir, Muhammad Imran, Neeraj Kumar, Di Zhang, and Keping Yu. 2021. Robust spammer detection using collaborative neural network in internet-of-things applications. *IEEE Internet of Things Journal*, 8(12):9549–9558.

Abhimanyu Hans, Avi Schwarzschild, Valeriia Cherepanova, Hamid Kazemi, Aniruddha Saha, Micah Goldblum, Jonas Geiping, and Tom Goldstein. 2024. Spotting LLMs with binoculars: Zero-shot detection of machine-generated text.

Tatsunori B. Hashimoto, Hugh Zhang, and Percy Liang. 2019. Unifying human and statistical evaluation for natural language generation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1689–1701, Minneapolis, Minnesota. Association for Computational Linguistics.

Yongxin He, Shan Zhang, Yixuan Cao, Lei Ma, and Ping Luo. 2025. DETree: DEtecting human-AI collaborative texts via tree-structured hierarchical representation learning. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

Xiaomeng Hu, Pin-Yu Chen, and Tsung-Yi Ho. 2023. Radar: Robust ai-text detection via adversarial learning. In *Advances in Neural Information Processing Systems*, volume 36, pages 15077–15095. Curran Associates, Inc.

Daphne Ippolito, Daniel Duckworth, Chris Callison-Burch, and Douglas Eck. 2020. Automatic detection of generated text is easiest when humans are fooled. In *Proceedings of the 58th Annual Meeting of*

the Association for Computational Linguistics, pages 1808–1822, Online. Association for Computational Linguistics.

Kalpesh Krishna, Yixiao Song, Marzena Karpinska, John Wieting, and Mohit Iyyer. 2023. Paraphrasing evades detectors of ai-generated text, but retrieval is an effective defense. In *Advances in Neural Information Processing Systems*, volume 36, pages 27469–27500. Curran Associates, Inc.

Runheng Liu, Heyan Huang, Xingchen Xiao, and Zhijing Wu. 2025. Zero-shot detection of LLM-generated text via implicit reward model. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

Eric Mitchell, Yoonho Lee, Alexander Khazatsky, Christopher D Manning, and Chelsea Finn. 2023. DetectGPT: Zero-shot machine-generated text detection using probability curvature. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 24950–24962. PMLR.

Guilherme Penedo, Quentin Malartic, Daniel Hesslow, Ruxandra Cojocaru, Alessandro Cappelli, Hamza Alobeidli, Baptiste Pannier, Ebtesam Almazrouei, and Julien Launay. 2023. The RefinedWeb dataset for Falcon LLM: outperforming curated corpora with web data, and web data only. *arXiv preprint arXiv:2306.01116*.

Irene Solaiman, Miles Brundage, Jack Clark, Amanda Askell, Ariel Herbert-Voss, Jeff Wu, Alec Radford, Gretchen Krueger, Jong Wook Kim, Sarah Kreps, and 1 others. 2019. Release strategies and the social impacts of language models. *arXiv preprint arXiv:1908.09203*.

Yiliao Song, Zhenqiao Yuan, Shuhai Zhang, Zhen Fang, Jun Yu, and Feng Liu. 2025. Deep kernel relative test for machine-generated text detection. In *The Thirteenth International Conference on Learning Representations*.

Yuchuan Tian, Hanting Chen, Xutao Wang, Zheyuan Bai, QINGHUA ZHANG, Ruifeng Li, Chao Xu, and Yunhe Wang. 2024. Multiscale positive-unlabeled detection of AI-generated texts. In *The Twelfth International Conference on Learning Representations*.

Adaku Uchendu, Thai Le, Kai Shu, and Dongwon Lee. 2020. Authorship attribution for neural text generation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8384–8395, Online. Association for Computational Linguistics.

Yuxia Wang, Jonibek Mansurov, Petar Ivanov, Jinyan Su, Artem Shelmanov, Akim Tsvigun, Chenxi Whitehouse, Osama Mohammed Afzal, Tarek Mahmoud, Toru Sasaki, Thomas Arnold, Alham Fikri Aji, Nizar Habash, Iryna Gurevych, and Preslav Nakov. 2024. M4: Multi-generator, multi-domain, and multilingual black-box machine-generated text detection. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1369–1407, St. Julian's, Malta. Association for Computational Linguistics.

Junchao Wu, Runzhe Zhan, Derek F. Wong, Shu Yang, Xinyi Yang, Yulin Yuan, and Lidia S. Chao. 2024. DetectRL: Benchmarking LLM-generated text detection in real-world scenarios. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Yihuai Xu, Yongwei Wang, Yifei Bi, Huangsen Cao, Zhouhan Lin, Yu Zhao, and Fei Wu. 2024. Training-free llm-generated text detection by mining token probability sequences. *CoRR*, abs/2410.06072.

Xiaowei Zhu, Yubing Ren, Yanan Cao, Xixun Lin, Fang Fang, and Yangxi Li. 2025a. Reliably bounding false positives: A zero-shot machine-generated text detection framework via multiscaled conformal prediction. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12298–12319, Vienna, Austria. Association for Computational Linguistics.

Xiaowei Zhu, Yubing Ren, Fang Fang, Qingfeng Tan, Shi Wang, and Yanan Cao. 2025b. DNA-detectLLM: Unveiling AI-generated text via a DNA-inspired mutation-repair paradigm. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

A Effect Analysis of Exonic Reweighting

Notation. Given a sequence $s = \{x_1, \ldots, x_T\}$, we define the token-level quantities

$$a_i \triangleq -\log P_M(x_i), \tag{11}$$

$$b_i \triangleq -P_M(x_i) \log P_{\tilde{M}}(x_i), \tag{12}$$

and the unweighted global score

$$R_0 \triangleq \frac{A_0}{B_0}, \qquad A_0 = \sum_{i=1}^{T} a_i, \quad B_0 = \sum_{i=1}^{T} b_i. \tag{13}$$

Let $S = \{i : \delta_i > \theta\}$ denote the set of exonic tokens. For each $i \in S$, we assign an additional, token-specific weight $\Delta w_i > 0$ mapped from the hidden-state discrepancy δ_i; tokens outside S retain unit weight. We define the corresponding weighted sums over exonic tokens as

$$A_S \triangleq \sum_{i \in S} \Delta w_i a_i, \qquad B_S \triangleq \sum_{i \in S} \Delta w_i b_i. \tag{14}$$

The resulting exon-aware translation score is then given by

$$R^W \triangleq \frac{A_0 + A_S}{B_0 + B_S}. \tag{15}$$

Score shift under exon-aware reweighting. Since $b_i > 0$ for all tokens, it follows directly that $B_0 > 0$ and $B_S > 0$. We analyze the difference between the reweighted and unweighted scores:

$$\begin{aligned} R^W - R_0 &= \frac{A_0 + A_S}{B_0 + B_S} - \frac{A_0}{B_0} \\ &= \frac{B_0(A_0 + A_S) - A_0(B_0 + B_S)}{B_0(B_0 + B_S)} \\ &= \frac{B_0 A_S - A_0 B_S}{B_0(B_0 + B_S)} \\ &= \frac{B_0 B_S}{B_0(B_0 + B_S)} \left(\frac{A_S}{B_S} - \frac{A_0}{B_0} \right). \end{aligned} \tag{16}$$

As the denominator is strictly positive, we obtain the following sign equivalence:

$$\operatorname{sign}(R^W - R_0) = \operatorname{sign}\left(\frac{A_S}{B_S} - R_0 \right). \tag{17}$$

Connection to empirical observations. Figure 2 shows that among tokens with large hidden-state discrepancy ($\delta_i > \theta$), the number of tokens that increase class separability substantially exceeds the number that decrease it. In addition, the magnitudes of the corresponding token-level quantities a_i and b_i are empirically observed to be of comparable scale, rather than differing by orders of magnitude. Taken together, these observations indicate that the aggregated exonic ratio

$$\frac{A_S}{B_S} = \frac{\sum_{i \in S} \Delta w_i a_i}{\sum_{i \in S} \Delta w_i b_i} \tag{18}$$

is predominantly influenced by tokens that contribute in a label-consistent direction. Accordingly, for near-boundary (hard) samples with $R_0 \approx \tau$, we expect $\frac{A_S}{B_S} < \tau$ for AI-generated texts and $\frac{A_S}{B_S} > \tau$ for human-written texts. By (17), this implies $R^W - R_0 < 0$ for AI-generated texts and $R^W - R_0 > 0$ for human-written texts, indicating that exon-aware reweighting pushes the score toward the correct side of the decision boundary and improves separability.

B Additional Implementation Details

Regarding the construction of the evaluation datasets, we randomly and uniformly sample 2,000 text samples from each public benchmark, including M4, DetectRL (Multi-LLM and Multi-Domain settings), and RealDet, ensuring balanced class distributions for experimental evaluation.

During evaluation, the maximum input length is capped at 1024 tokens. All experiments are conducted on a single NVIDIA A100 GPU with 80GB memory. All models are executed using 32-bit floating-point precision (FP32).

C Data Construction in the Robustness Experiment

In the Polish Attack, we employ GPT-4o to refine texts originally written by humans. The specific model version and decoding parameters are as follows:

- **GPT-4o Turbo**: `gpt-4o-2024-11-20`, Temperature = 1.0, Top-p = 1.0.

To ensure that the semantic content and overall structure of the original human-written texts remain largely unchanged, the model is instructed to perform light polishing only, focusing on improving fluency and expression rather than rewriting or altering meaning. The input prompt is carefully constructed to enforce this constraint and is specified as follows:

- `Polish the following human-written text by correcting grammar and improving fluency, while ensuring that the semantic content, author intent, and discourse structure remain unchanged. The result should read more natural but convey exactly the same meaning as the original text: \n + original human-written text`

D Further Exploration of Mapping Functions

This section further investigates the impact of different mapping functions on detection performance, focusing on a comparison between a linear mapping and the default nonlinear mapping. The nonlinear mapping is computed as defined in Eq 6, while the linear mapping is formalized as follows:

$$g(\delta_t) = \alpha \left(\delta_t - \theta \right)_+. \tag{19}$$

Under the linear mapping, the additional weight Δw_t is constrained within the range $(0, \alpha)$, which results in a steeper scaling behavior compared to the $(0, 1)$ range adopted by the nonlinear mapping. As reported in Table 3, the experimental results

Detectors	M4		DetectRL Multi-LLM		DetectRL Multi-Domain		RealDet		Avg.	
	AUROC	F_1	AUROC	F_1	AUROC	F_1	AUROC	F_1	AUROC	F_1
Nonlinear mapping										
$\alpha = 10$	92.43	88.05	90.67	84.95	90.46	86.59	94.98	91.30	92.14	87.72
$\alpha = 20$	92.25	87.55	90.87	85.20	90.89	87.05	95.02	91.29	92.26	87.77
Linear mapping										
$\alpha = 10$	92.24	87.92	90.90	84.93	91.18	87.24	95.03	91.02	92.34	87.78
$\alpha = 20$	91.65	86.63	90.66	84.40	91.68	86.85	95.04	90.66	92.26	87.14

Table 3: Detection performance (AUROC and F1 score) with different mapping functions ($\theta = 0.15$).

Parameter Setting	DetectRL Multi-LLM		DetectRL Multi-Domain		Avg.	
	AUROC	F_1	AUROC	F_1	AUROC	F_1
Reverse						
$L = 2 \quad (\theta = 0.15, \alpha = 10)$	89.09	84.39	88.24	84.53	88.67	84.46
$L = 4 \quad (\theta = 0.15, \alpha = 10)$	89.41	85.07	88.38	84.65	88.90	84.86
$L = 8 \quad (\theta = 0.15, \alpha = 10)$	89.13	84.36	88.56	84.70	88.85	84.53
$L = 16 \quad (\theta = 0.15, \alpha = 10)$	90.63	85.38	90.55	86.79	90.59	86.09
Forward						
$L = 2 \quad (\theta = 0.15, \alpha = 10)$	89.09	84.39	88.21	84.41	88.65	84.40
$L = 4 \quad (\theta = 0.15, \alpha = 10)$	88.98	84.38	87.68	84.12	88.33	84.25
$L = 8 \quad (\theta = 0.15, \alpha = 10)$	89.13	84.36	87.72	84.21	88.43	84.29
$L = 16 \quad (\theta = 0.15, \alpha = 10)$	90.20	85.08	89.37	85.52	89.79	85.30

Table 4: Detection performance with different hidden layers.

show that the linear mapping can still achieve competitive performance. In some cases, it even attains a higher average performance than the nonlinear counterpart. However, its performance tends to be less stable across different datasets.

Overall, these results consistently demonstrate the effectiveness of mapping hidden-state discrepancies to additional token-level weights. Importantly, this effectiveness remains robust to the specific choice of mapping function, indicating that the proposed reweighting mechanism is not sensitive to the particular form of the mapping employed.

E Effect of Hidden-Layer Discrepancies

In the hyperparameter sensitivity analysis, we compared the impact of extracting hidden-layer discrepancies across varying numbers of layers, ranging from 2 to 32, on detection performance. Building upon this analysis, this section further investigates the effect of reverse extraction of hidden-layer discrepancies, specifically considering differences computed from the last layers backward, spanning from 2 to 16 layers.

As reported in Table 4, the experimental results show that, for the same number of layers, reverse extraction consistently outperforms forward extraction. This observation suggests that discrepancies derived from higher hidden layers are more informative, as they more effectively capture token-level importance at the current position. Moreover, these findings indicate the potential benefits of employing more fine-grained and structurally richer strategies for modeling hidden-layer discrepancies, which may further enhance detection performance.

Setting ($L = 16$) ↓	$\theta = 0.05$	$\theta = 0.10$	$\theta = 0.15$	$\theta = 0.20$
M4				
$\alpha = 2$	92.03	92.09	92.23	92.09
$\alpha = 6$	91.99	92.06	92.57	92.38
$\alpha = 10$	91.89	91.88	**92.60**	92.50
Multi-LLM				
$\alpha = 2$	89.58	89.64	89.72	89.47
$\alpha = 6$	89.59	89.67	90.07	89.54
$\alpha = 10$	89.52	89.56	**90.20**	89.60
Multi-Domain				
$\alpha = 2$	88.57	88.66	88.89	88.64
$\alpha = 6$	88.42	88.49	89.27	88.87
$\alpha = 10$	88.26	88.18	**89.37**	88.97
RealDet				
$\alpha = 2$	94.71	94.73	94.78	94.68
$\alpha = 6$	94.72	94.79	95.01	94.81
$\alpha = 10$	94.68	94.75	**95.10**	94.88

Table 5: AUROC under different hyperparameters.

F Additional Hyperparameter Experiments

We conduct additional experiments to further examine the effects of the hyperparameters α and θ. Specifically, on public benchmark datasets, we evaluate detection performance under $L = 16$ hidden layers, with $\alpha \in \{2, 6, 10\}$ and $\theta \in \{0.05, 0.10, 0.15, 0.20\}$, as summarized in Table 5.

It is evident that, regardless of the hyperparameter configuration, utilizing hidden-layer discrepancies from only 16 layers consistently underperforms the setting with 32 layers. Nevertheless, under the 16-layer configuration, EXONS-DETECT remains largely insensitive to variations in both α and θ, exhibiting only marginal performance fluctuations. Furthermore, an analysis of AUROC score variations indicates that, for the LLM pair (Falcon-7B Instruct + Falcon-7B), the optimal hyperparameter combination is $\alpha = 10$ and $\theta = 0.15$.