Alignment Imprint: Zero-Shot AI-Generated Text Detection via Provable Preference Discrepancy

Junxi Wu[1,2*], **Kailin Huang**[3*], **Dongjian Hu**[1*], **Bin Chen**[4†], **Hao Wu**[2,5], **Shu-Tao Xia**[2], **Changliang Zou**[1]
[1] Nankai University [2] Tsinghua University [3] Wuhan University
[4] Harbin Institute of Technology, Shenzhen [5] Shannon InfoTech
wujunxi@mail.nankai.edu.cn

Abstract

Detecting AI-generated text is an important but challenging problem. Existing likelihood-based detection methods are often sensitive to content complexity and may exhibit unstable performance. In this paper, our key insight is that modern Large Language Models (LLMs) undergo alignment (including fine-tuning and preference tuning), leaving a measurable distributional imprint. We theoretically derive this imprint by abstracting the alignment process as a sequence of constrained optimization steps, showing that the log-likelihood ratio can naturally decompose into implicit instructional biases and preference rewards. We refer to this quantity as the *Alignment Imprint*. Furthermore, to mitigate the instability in high-entropy regions, we introduce *Log-likelihood Alignment Preference Discrepancy (LAPD)*, a standardized information-weighted statistic based on alignment imprint. We provide statistical guarantee that alignment-based statistics dominate Fast-DetectGPT in performance. We also theoretically show that LAPD strictly improves the unweighted alignment scores when the aligned and base models are close in distribution. Extensive experiments show that LAPD achieves an improvement 45.82% relative to the strongest existing baselines, yielding large and consistent gains across all settings.

1 Introduction

The rapid advancement of Large Language Models (LLMs) has significantly expanded the capabilities of automated text generation. However, these capabilities also introduce profound risks, including the spread of disinformation, academic dishonesty, and malicious text attacks [23]. Modern LLMs produce fluent and coherent content, blurring the statistical boundary with human-written text. Consequently, developing more reliable mechanisms to distinguish AI-generated text has become a critical requirement for mitigating misuse.

A popular line of recent work frames detection as a likelihood-based problem, leveraging the observation that AI-generated text tends to receive higher probability under the generative model. However, such approaches remain fundamentally confounded by intrinsic content complexity. A simple human-written sentence may receive a higher likelihood than a long technical paragraph generated by an AI system [37]. As a result, likelihood score alone often collapses in the real world.

In practice, users almost exclusively interact with *aligned models*, rather than raw pre-trained *base models*, which are not designed to reliably follow instructions. Modern LLMs (e.g., ChatGPT [1], Gemini [7], and DeepSeek [13]) are typically first trained on large-scale unsupervised corpora and

*Equal contribution.
†Corresponding author.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

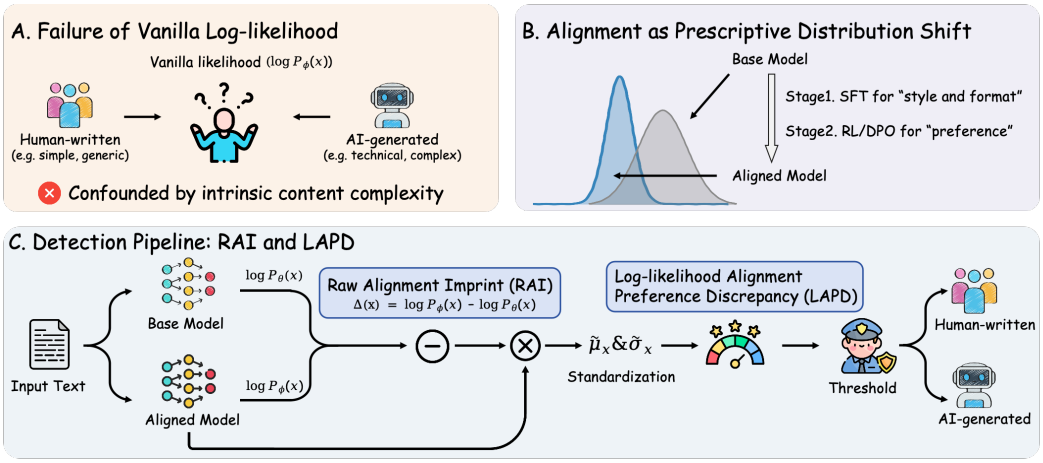


Figure 1: Overview. **(A)** Vanilla log-likelihood is confounded by intrinsic content complexity (e.g. simple human-written text may receive higher likelihood than complex AI-generated text), undermining reliable detection. **(B)** LLM alignment pipelines induce a prescriptive distribution shift, yielding a systematic shift between base and aligned models. **(C)** Detection pipeline.

then transformed through a multi-stage alignment process. This process commonly consists of supervised fine-tuning (SFT), followed by preference tuning methods such as RLHF [24] or DPO [27]. SFT adapts the base model to instruction–response data drawn from curated datasets, while preference tuning further adjusts the output distribution to satisfy explicit reward objectives under KL regularization. Together, these stages systematically modify the model's output distribution by introducing prescriptive constraints that reflect safety norms and human preferences, echoing the conclusions drawn by Sivaprasad et al. [30].

Our key insight is that the alignment process leaves a distributional imprint that is statistically exploitable to distinguish. While human language arises from diverse communicative intents, AI-generated text is optimized to satisfy explicit and implicit alignment objectives. As a consequence, aligned models systematically reallocate the probability mass toward a narrower subset of "preferred" responses, inducing entropy contraction and mode concentration. Kirk et al. [18] have observed this effect and reported reduced lexical and semantic diversity following alignment.

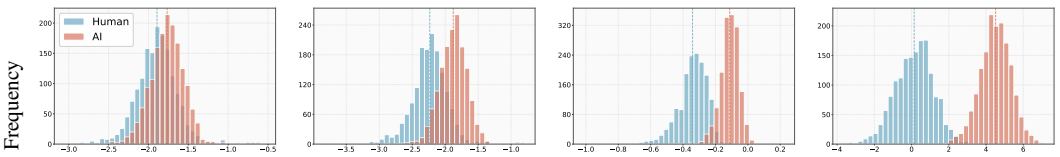
Given a pair of a base model and an aligned model, we characterize the distributional shift induced by alignment and derive that the log-likelihood ratio between them can naturally decompose into implicit instructional biases introduced during SFT and explicit rewards imposed during alignment. We refer to this quantity as the *Alignment Imprint*. Intuitively, it measures how much additional probability a text receives solely due to alignment.

However, directly using the raw log-likelihood ratio as a detection statistic remains suboptimal. In high-entropy regions, small aleatoric variations can induce large fluctuations in log space, masking the alignment signal. To address this issue, we further introduce ***L**og-likelihood **A**lignment **P**reference **D**iscrepancy (LAPD)*, a standardized information-weighted statistic based on alignment imprint. By weighting the alignment imprint according to predictability, LAPD amplifies the imprint and achieves much better performance.

Furthermore, we provide theoretical guarantees for the superiority of the alignment imprint and LAPD. Under mild and interpretable assumptions, we prove that alignment-based statistics dominate Fast-DetectGPT [2] in terms of true negative rate at any fixed false negative rate. Moreover, we show that LAPD strictly improves upon unweighted alignment scores in regimes where the aligned model and base model are close in distribution.

Our contributions are summarized as follows:

- We derive a principled characterization by abstracting alignment pipelines as two constrained optimization steps. We show that the log-likelihood ratio between aligned model and its base model can naturally decompose into implicit instructional biases and preference rewards. This derivation formalizes the alignment imprint as a measurable consequence, without requiring access to reward models or training data.



(a) Base Model (b) Aligned Model (c) RAI (d) LAPD

Figure 2: Distributions of human-written texts and AI-generated texts. Dashed vertical lines represent the mean value of each distribution. (a) Average log-likelihood under the base model; (b) average log-likelihood under the aligned model; (c) raw alignment imprint; and (d) LAPD score.

- Building on alignment imprint, we introduce LAPD, a standardized statistic that combines alignment-induced likelihood ratios with token-level self-information. This enhancement yields a stable and effective detection score that consistently outperforms existing detectors across LLMs, domains, and text lengths.
- We provide two theoretical guarantees on detection performance. Under mild and interpretable assumptions, we prove that the alignment-based statistics dominate Fast-DetectGPT in terms of true negative rate at any fixed false negative rate. Moreover, we also theoretically show that LAPD strictly improves the unweighted alignment scores when the aligned and base models are close in distribution.
- We have conducted extensive experiments on four comprehensive benchmarks. LAPD yields large and consistent gains across all benchmarks, achieving an average relative improvement of 56.99% over Fast-DetectGPT and 45.82% over the strongest baselines. Additional six robustness and ablation studies confirm that the improvements are stable in complex practical scenarios, with each component consistently contributing to effectiveness. In particular, LAPD improves 76.81% under an extremely 0.5% Type-I error constraint.

2 Related Work

2.1 LLM-Generated Text Detection

Existing methods can be categorized into training-based and training-free approaches. Training-based methods [31, 33, 15] involve training a classifier on collected datasets for detection. There are also methods like ImBD [4] that fine-tunes a proxy LLM to favor machine preferences for more sensitive detection. However, these supervised models often suffer from the overfitting problem [3].

Training-free methods leverage the probabilistic features of the text without additional training, exhibiting superior generalization. Early approaches used simple metrics such as Entropy [16], Log-Likelihood [31], and LogRank [11]. More recently, DetectGPT [21] introduced a perturbation paradigm, utilizing separate perturbing and scoring models for detection. Fast-DetectGPT [2] further enhances efficiency by conditional sampling, while Lastde [38] incorporates temporal dynamics. Additionally, Binoculars [14] contrasts perplexity with cross-perplexity, and DNA-DetectLLM [43] employs a repair score based on a mutation-repair mechanism.

Most closely related to our work are that use difference between base and aligned LLMs, including ReMoDetect [19], IRM [20], and dual-network [6]. While these works explore the alignment gap between model pairs, our work provides the first complete theoretical derivation of the alignment imprint across the full SFT and preference tuning pipeline, introduces the noise-robust information-weighted LAPD statistic and offers formal provable performance guarantees, in contrast to ReMoDetect's training-dependent pipeline and the limited theoretical framing of IRM and dual-network method.

2.2 LLM Alignment

The pipeline of modern LLMs training usually progresses through three steps: Pre-training, Supervised Fine-Tuning (SFT), and preference tuning (e.g., RLHF or DPO) [39]. Pre-training establishes the fundamental probability distribution over massive corpora data. Subsequently, SFT adapts the base model to follow user instructions by maximizing the likelihood estimation of high-quality instructions. This process can be conceptualized as reweighting the pretrained distribution to favor coherent and instruction-following sequences [40].

Preference tuning is used to align models with human values such as safety and helpfulness. The traditional approach, RLHF [24], optimizes a policy against a learned reward model with algorithms like PPO [29]. It often constrains the generation diversity to ensure safety [18]. Recently, DPO [27] has emerged as an alternative, which analytically solves the RLHF objective and optimizes the policy directly from preference data, implicitly modeling the reward.

3 Method and Theory

In this section, we derive the *Alignment Imprint* and propose our detection statistic, *Log-likelihood Alignment Preference Discrepancy (LAPD)*. In particular, our objective is not to exactly replicate the training dynamics of specific algorithms, but to characterize the distributional effect induced by alignment. This abstraction allows us to derive a tractable likelihood ratio statistic, even when the true reward functions or training data are unknown.

3.1 Deriving the Alignment Imprint

Let $\boldsymbol{x} = [x_0, \ldots, x_T]$ denote a token sequence of length $T+1$. We denote the probability distribution of the pre-trained base model as $P_\theta(\boldsymbol{x})$[3], representing the prior distribution learned from large-scale unsupervised corpora. Modern alignment pipelines transform this base distribution into a final aligned distribution, denoted as $P_\phi(\boldsymbol{x})$, through a two-stage process: SFT and preference tuning.

Firstly, we consider the SFT stage, where the base model is adapted to a curated dataset of instruction-response pairs. We conceptualize the construction of this dataset not as random sampling, but as a filtering process governed by a latent quality function $V(\boldsymbol{x})$, which captures implicit attributes such as instruction adherence and formatting. Applying the principle of maximum entropy under the constraint of expected quality, the theoretical SFT distribution P_{SFT} is:

$$P_{\text{SFT}}(\boldsymbol{x}) = \frac{1}{Z_1} P_\theta(\boldsymbol{x}) \exp(V(\boldsymbol{x})), \tag{1}$$

where Z_1 is the partition function required for normalization. This formulation implies that the probability shift in the first stage is directly proportional to the latent instructional quality $V(\boldsymbol{x})$. The full derivation is given in Appendix A.1.

Taking the logarithm and rearranging the terms, we have:

$$\log P_{\text{SFT}}(\boldsymbol{x}) - \log P_\theta(\boldsymbol{x}) = V(\boldsymbol{x}) - \log Z_1. \tag{2}$$

Subsequently, in the preference tuning stage, the model is optimized to align with human preferences. Whether optimized via RLHF or DPO, the objective can be formally defined as maximizing the expected reward $R(\boldsymbol{x})$ while minimizing the KL-divergence from the reference policy:

$$\max_\phi \mathbb{E}_{\boldsymbol{x}\sim P_\phi}[R(\boldsymbol{x})] - \beta D_{KL}(P_\phi || P_{\text{SFT}}), \tag{3}$$

where β is the temperature coefficient controlling the strength of the regularization, and we denote the optimal aligned policy as $P_{\phi^*}(\boldsymbol{x})$.

The closed-form solution to this constrained optimization problem is given by:

$$\log P_{\phi^*}(\boldsymbol{x}) - \log P_{\text{SFT}}(\boldsymbol{x}) = \frac{1}{\beta} R(\boldsymbol{x}) - \log Z_2, \tag{4}$$

where $Z_2 = \sum_{\boldsymbol{x}'} P_{\text{SFT}}(\boldsymbol{x}') \exp(R(\boldsymbol{x}')/\beta)$ is the normalization constant. The full derivation is given in Appendix A.2.

By adding eq. (2) and eq. (4), we can obtain:

$$\log P_{\phi^*}(\boldsymbol{x}) - \log P_\theta(\boldsymbol{x}) = \frac{1}{\beta} R(\boldsymbol{x}) + V(\boldsymbol{x}) - C, \tag{5}$$

[3]For simplicity, we denote $P_\theta(\boldsymbol{x}) = \prod_{t=1}^{T} P_\theta(x_t|x_{<t})$, and $P_\theta(x_t) = P_\theta(x_t|x_{<t})$. The notation for P_ϕ follows similarly.

where $C = \log Z_1 + \log Z_2$ is a constant term and we denote the log-likelihood ratio between the final aligned model and the original base model as *Raw Alignment Imprint (RAI)*:

$$\Delta(\boldsymbol{x}) := \log P_\phi(\boldsymbol{x}) - \log P_\theta(\boldsymbol{x}). \tag{6}$$

This derivation reveals that $\Delta(\boldsymbol{x})$ quantifies the relative likelihood gain (or loss) that a text receives under the aligned model compared to the base. The term $V(\boldsymbol{x})$ accounts for the implicit "style and format" biases injected during SFT, while the term $R(\boldsymbol{x})$ accounts for the explicit "value and preference" biases injected during RLHF. Since LLMs are optimized to satisfy alignment objectives, AI-generated texts tend to exhibit systematically higher values of $\Delta(\boldsymbol{x})$ relative to human-written texts, even though $\Delta(\boldsymbol{x})$ is often negative due to entropy contraction induced by alignment.

3.2 Enhancing the Alignment Imprint

Although $\Delta(\boldsymbol{x})$ theoretically captures the divergence between aligned and base models, using it as a raw detection statistic is suboptimal due to its scale sensitivity in the log-probability space. Specifically, in high-entropy regions where model confidence is low, even minor aleatoric variations in probability can be large fluctuations in log-likelihood ratios.

We observe that AI-generated texts typically converge to a low-entropy region and positive alignment shifts. In contrast, human-written texts often traverse relatively high-entropy regions where the alignment signal is weak or negative. By coupling predictability with alignment imprint, we introduce the information-weighted score:

$$\mathcal{S}(\boldsymbol{x}) = \sum_{t=1}^{T} \mathcal{I}_\phi(x_t) \cdot \Delta(x_t), \tag{7}$$

where $\Delta(x_t) = \log P_\phi(x_t) - \log P_\theta(x_t)$ and $\mathcal{I}_\phi(x_t) = -\log P_\phi(x_t)$ denotes the self-information of the text under the aligned model. By coupling the alignment-induced likelihood ratio with self-information, $\mathcal{S}(\boldsymbol{x})$ captures a structural property: AI-generated text is favored by alignment in confident regions, whereas human-written text tends to exhibit misalignment in regions of low confidence.

To improve robustness across different domains and sequence lengths, we standardize $\mathcal{S}(\boldsymbol{x})$ by perturbing. We define the final LAPD statistic $\mathcal{S}_{\text{LAPD}}(\boldsymbol{x})$:

$$\mathcal{S}_{\text{LAPD}}(\boldsymbol{x}) = \frac{\mathcal{S}(\boldsymbol{x}) - \tilde{\mu}_x}{\tilde{\sigma}_x}, \tag{8}$$

where $\tilde{\mu}_x$ and $\tilde{\sigma}_x$ represent the expectation and standard deviation of the score under the perturbation. To achieve inference efficiency, we use the conditional independence sampling like Bao et al. [2] instead of multiple autoregressive sampling:

$$\tilde{\mu}_x = \sum_{t=1}^{T} \mathbb{E}_{x_t \sim P_\theta(\cdot|x_{<t})}[\mathcal{I}_\phi(x_t) \cdot \Delta(x_t)]$$

$$\tilde{\sigma}_x^2 = \sum_{t=1}^{T} \text{Var}_{x_t \sim P_\theta(\cdot|x_{<t})}[\mathcal{I}_\phi(x_t) \cdot \Delta(x_t)]$$

3.3 Theoretical Guarantees

In this section, we provide theoretical guarantees for the superiority of LAPD. We compare the true negative rate (TNR) of LAPD with fixed FNR, versus $S_{\text{Fast}} = (\log P_\phi(\boldsymbol{x}) - \tilde{\mu}_x)/\tilde{\sigma}_x$, which was proposed by Bao et al. [2].

Note that a consistently higher TNR at fixed FNR levels implies a strictly better AUROC. With a slight abuse of notation, we denote $\tilde{\mu}_x$ and $\tilde{\sigma}_x$ as the expectation and standard deviation of $\log P_\phi$ under perturbation. The TNR for S_{LAPD} with a threshold τ is defined as

$$\text{TNR}_\tau(S_{\text{LAPD}}) = \mathbb{P}_{x \sim p}(S_{\text{LAPD}}(\boldsymbol{x}) \leq \tau),$$

where the probability is taken under the human-written text distribution p. For simplicity, we use p_t and q_t to denote the conditional distributions of human-written text $p(\cdot|x_{<t})$ and AI-generated text $q(\cdot|x_{<t})$, respectively. We start by presenting the following three assumptions:

Assumption 1. *(Indistinguishability of base model).*

$$\sum_{t=1}^{T} \mathbb{E}_{\tilde{x}_t \sim q_t} \log P_\theta(\tilde{x}_t|x_{<t}) - \mathbb{E}_{\tilde{x}_t \sim p_t} \log P_\theta(\tilde{x}_t|x_{<t}) = o(T).$$

Assumption 2. *(Prescriptiveness of AI-generated text). There exists $\epsilon > 0$, such that for any t,*

$$\mathbb{E}_{\tilde{x}_t \sim q_t} \log P_\phi(\tilde{x}_t|x_{<t}) - \mathbb{E}_{\tilde{x}_t \sim p_t} \log P_\phi(\tilde{x}_t|x_{<t}) \geq \epsilon,$$
$$\mathit{Var}_{\tilde{x}_t \sim q_t} \log P_\phi(\tilde{x}_t|x_{<t}) \geq \mathit{Var}_{\tilde{x}_t \sim q_t} \Delta(\tilde{x}_t|x_{<t}).$$

Assumption 3. *(Diversity of human-written text)*

$$\mathit{Cov}_{\tilde{x}_t \sim p_t} \left(\Delta(\tilde{x}_t|x_{<t}), \log P_\phi(\tilde{x}_t|x_{<t})\right) \geq \mathit{Cov}_{\tilde{x}_t \sim q_t} \left(\Delta(\tilde{x}_t|x_{<t}), \log P_\phi(\tilde{x}_t|x_{<t})\right).$$

Assumption 1 formalizes the observation that the pre-trained base model is optimized to model the broad distribution of human language, making it statistically indistinguishable for both sources. Assumption 2 characterizes the phenomenon that the aligned model P_ϕ tends to yield higher likelihoods to AI-generated text. Assumption 3 suggests that due to the lexical and semantic divergence, under human-written text, the variance of the scores tends to be larger. Moreover, there is also a stronger positive correlation between the alignment shift and the aligned model's confidence. We further provide an empirical validation of these assumptions in Appendix B.3.

Based on these assumptions, we first propose theoretical results of S_{Fast} and the alignment imprint $\Delta(\boldsymbol{x})$. To compare at a fixed FNR, we consider the standardized $\Delta(\boldsymbol{x})$, which means $S_{\text{Align}} = (\Delta(\boldsymbol{x}) - \tilde{\mu}_x)/\tilde{\sigma}_x$:

Theorem 1. *Under Assumption 1 and Assumption 2, for any fixed threshold τ, we have*

$$\mathit{TNR}_\tau(S_{Align}) \succeq \mathit{TNR}_\tau(S_{Fast}),$$

where we use the notation $A \succeq B$ to denote that $A \geq B + o_p(1)$ as $T \to \infty$. In other words, the inequality holds asymptotically or with a negligible error term.

Theorem 1 provides the theoretical justification for the alignment imprint, which effectively cancel out the intrinsic content-dependent component shared by the base model and the aligned model.

Furthermore, Theorem 2 below shows that under circumstances where S_{Align} is not distinguishable, LAPD can significantly amplify the alignment signal and more effective.

Theorem 2. *Denote $m_\phi = \sup_{\tilde{x}_t \sim q_t} |\log P_\phi(\tilde{x}_t|x_{<t})|$, $m_\Delta = \sup_{\tilde{x}_t \sim q_t} |\Delta(\tilde{x}_t|x_{<t})|$. under Assumption 2 and Assumption 3, if for some $k > 0$, $k \cdot Var_{\tilde{x}_t \sim q_t} \Delta(\tilde{x}_t|x_{<t}) \geq Var_{\tilde{x}_t \sim q_t} \log P_\phi(\tilde{x}_t|x_{<t})$, and*

$$\left| \frac{\mathbb{E}_{\tilde{x}_t \sim q_t} \Delta(\tilde{x}_t|x_{<t})}{\mathbb{E}_{\tilde{x}_t \sim p_t} \Delta(\tilde{x}_t|x_{<t})} - 1 \right| \leq \frac{\epsilon}{2m_\phi + \sqrt{k} m_\Delta},$$

then we have

$$\mathit{TNR}_\tau(S_{LAPD}) \succeq \mathit{TNR}_\tau(S_{Align}).$$

The proofs of Theorem 1 and Theorem 2 can be seen in Appendix B.1 and Appendix B.2, respectively.

4 Experiments

4.1 Settings

Datasets. To evaluate the detectors in practical scenarios, we conduct our primary evaluations on four benchmarks: M4 [35], DetectRL [36], RAID [8] and RealDet [42], randomly select 2,000 balanced samples from each. Additionally, we conduct on other three datasets: XSum [22], WritingPrompts [9], and Arxiv [25]. For each dataset, we collect 1,600 human-written and AI-generated text pairs, which are generated by three advanced LLMs: GPT-4-Turbo, Gemini-2.0-Flash, and Claude-3.7-Sonnet.

Baselines. We compare our *Raw Alignment Imprint (RAI)* and *Log-likelihood Alignment Preference Discrepancy (LAPD)* with 8 representative training-free detection methods, including Entropy [16], Log-Likelihood [31], LogRank [11], DetectGPT [21], Fast-DetectGPT [2], Lastde++ [38], Binoculars [14], and DNA-DetectLLM [43].

Evaluation Metrics. We adopt the commonly used area under the receiver operating characteristic (AUROC [17]) curve as the main evaluation metric. Since the most harms often arise from false positives, we also adopt True Positive Rate (TPR) controlling False Positive Rate (FPR) at 0.5%.

Implementation. For RAI and LAPD, we use Llama2-7B [32] as base and perturbation model, and Llama2-7B-Instruct [32] as aligned model. More detailed information about implementation, benchmarks, datasets, and used LLMs can be seen in Appendix C.

4.2 Main Results

Table 1: Detection performance (AUROC %) on four comprehensive benchmarks. DetectRL Multi-LLM and DetectRL Multi-Domain are abbreviated as **Detect LLM** and **Detect Domain**, respectively. The best results are highlighted in **bold**. Methods with clubs (♣) standardize the scores by perturbing or generating auxiliary sequences.

Methods	M4	DetectRL LLM	DetectRL Domain	RAID	RealDet	Avg.
Entropy	71.24	64.37	47.91	68.56	90.08	68.43
Likelihood	77.63	66.82	48.95	71.02	89.17	70.72
LogRank	78.69	67.29	50.52	72.03	89.93	71.69
DetectGPT♣	46.54	43.13	28.00	49.43	61.33	45.69
Fast-DetectGPT♣	81.10	82.13	74.81	80.59	92.66	82.26
Lastde++♣	79.44	75.18	66.87	82.03	92.39	79.18
Binoculars	83.42	83.26	77.50	82.30	93.67	84.03
DNA-DetectLLM♣	77.38	89.01	88.29	81.42	94.48	86.12
RAI	70.46	93.13	**96.19**	81.28	86.92	85.60
LAPD♣	**88.02**	**97.17**	96.11	**85.21**	**95.32**	**92.37**

Table 2: Robustness across source LLMs and domains (AUROC %). Source LLMs are abbreviated as: **GPT** (GPT-4 Turbo), **Gemini** (Gemini-2.0 Flash), and **Claude** (Claude-3.7 Sonnet). The best results are highlighted in **bold**. Methods with clubs (♣) standardize the scores by perturbing or generating auxiliary sequences.

Methods	XSum			WritingPrompts			Arxiv			Avg.
	GPT	Gemini	Claude	GPT	Gemini	Claude	GPT	Gemini	Claude	
Entropy	72.76	54.93	74.97	87.88	90.39	91.21	45.96	80.05	80.49	75.41
Likelihood	70.02	69.48	70.37	80.85	95.23	85.84	57.91	93.60	86.22	78.84
LogRank	69.58	69.25	69.76	78.92	94.53	84.12	58.19	94.15	85.80	78.26
DetectGPT♣	64.77	44.71	44.62	59.75	60.60	47.32	28.94	53.92	60.67	51.70
Fast-DetectGPT♣	98.17	91.41	93.32	95.89	98.01	91.83	91.57	99.80	98.21	95.36
Lastde++♣	96.59	88.13	89.98	91.85	96.45	84.41	88.97	99.54	97.47	92.60
Binoculars	98.13	95.68	96.94	97.79	99.55	97.36	93.74	99.87	98.05	97.46
DNA-DetectLLM♣	99.35	96.77	98.51	98.89	99.73	98.55	95.08	99.88	98.42	98.36
RAI	98.94	94.77	95.94	94.50	96.25	87.85	95.94	99.18	97.85	95.69
LAPD♣	**99.84**	**98.64**	**99.76**	**99.73**	**99.77**	**99.83**	**97.87**	**99.99**	**99.99**	**99.49**

Table 1 shows the performance across four comprehensive benchmarks. RAI achieves a competitive average AUROC of 85.60%. LAPD further yields 92.37%, outperforming Fast-DetectGPT and

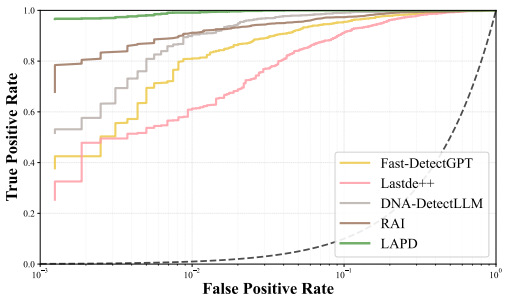


Figure 3: ROC curve in log scale. The methods are evaluated on XSum dataset with AI text generated by GPT-4-Turbo. The dash line denotes the random classifier.

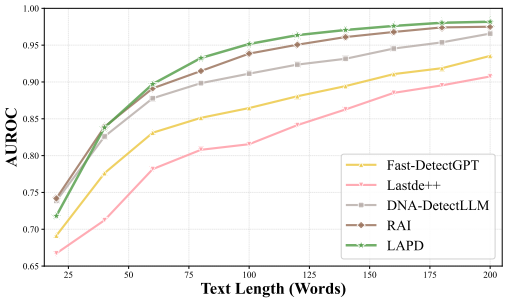


Figure 4: Detection performance (AUROC %) across varying text lengths (20–200 words). The methods are evaluated on XSum dataset with AI text generated by GPT-4-Turbo.

the best baselines relatively by 56.99% and 45.82%, where the "relative" is calculated by $(new - old)/(1.0 - old)$, representing the improvement made relative to the maximum possible.

Notably, in the DetectRL Multi-Domain benchmark, RAI achieves the highest AUROC of 96.19%. This observation indicates that our alignment imprint can effectively avoid interference from intrinsic content complexity, empirically validating the failure illustrated in Figure 1.A.

4.3 Robustness Analysis

4.3.1 Robustness across LLMs and domains

Table 2 presents the detection results across various LLMs and domains. RAI achieves competitive performance without extra perturbations, while LAPD further improves the average AUROC by 89.01% relative to Fast-DetectGPT. Even compared to the strongest baseline, LAPD achieves 68.90% relative enhancement. We further examine the transferability of a single detection threshold across domains in Appendix D.3.

4.3.2 Robustness on Low-FPR Area

In real-world scenarios, minimizing the rate of Type-I errors ($i.e.$ False Positive Rate, FPR) is often prioritized over aggregate metrics like AUROC. Figure 3 visually highlights this ability of LAPD, which maintains a high TPR (True Positive Rate) even when the FPR approaches zero, much better than all the baselines. Notably, the longer initial vertical segment indicates that TPR is highly sensitive to marginal changes in FPR, indicating a lack of robustness. While the baselines present significant vertical jumps, LAPD demonstrates superior stability in this low-FPR area.

To quantify this robustness across diverse domains, Table 9 shows the TPR at a 0.5% FPR threshold. Under this constraint, existing methods such as Fast-DetectGPT and DNA-DetectLLM exhibit significant performance drop. In contrast, our simple RAI already surpasses these baselines, demonstrating the intrinsic separability of the alignment signal. Furthermore, LAPD achieves an average TPR of 92.27%, representing a relative improvement of 76.81%, which highlights the robustness of LAPD under minimal false-positive tolerance.

4.3.3 Robustness on Different Text Lengths

Detecting shorter texts is significantly challenging due to limited statistical signals [33]. To evaluate the robustness across varying text lengths, we truncate the input texts to target word counts. Specifically, we conduct experiments on the XSum dataset with AI text generated by GPT-4-Turbo, varying the text length from 20 to 200 words.

Figure 4 visualizes the performance curves of five representative methods. While all detectors show improved AUROC as the length increases, LAPD demonstrates the most rapid AUROC growth in the short-text interval (20-80 words). Moreover, LAPD consistently outperforms all baselines once the length exceeds 40 words, while RAI ranking second. This observation presents the intrinsic robustness of our methods across various text lengths.

Table 3: Ablation results (AUROC %) of different components of LAPD. Best results in **bold**, second underlined.

Statistic	XSum	Writing	Arxiv
$\log P_\theta$	69.57	89.23	82.07
$\log P_\phi$	69.96	87.31	79.24
Δ	<u>96.55</u>	92.87	97.66
$\mathcal{S}$	96.40	<u>95.84</u>	<u>98.04</u>
$\mathcal{S}_{\text{LAPD}}$	**99.42**	**99.78**	**99.28**

Table 4: Ablation results (AUROC %) of LAPD with different model pairs across three datasets. Llama2-7B pair serves as the default.

Model Pairs	XSum	Writing	Arxiv
Falcon-7B Pair	87.35	93.37	86.29
GPT-J-6B Pair	94.62	91.56	92.75
Llama3.1-8B Pair	97.09	94.26	98.78
Llama2-7B Pair	**99.42**	**99.54**	**99.61**

4.3.4 Robustness under Attacks

We apply three types of token-level attacks: insertion, deletion, and replacement to 1% of the tokens and further extend to four more realistic attacks: paraphrase, synonym substitution, homoglyph, and zero-width space insertion from the RAID benchmark. Overall, our LAPD method outperforms all baselines, achieving the highest average attack AUROC and the highest clean performance, as illustrated in Appendix E.

4.4 Ablation Studies

4.4.1 Component Analysis

We evaluate the individual contribution of each component of LAPD, as summarized in Table 3. The results of log-likelihood alone are poor, regardless of base model or aligned model. This can be attributed to their high sensitivity to intrinsic content complexity. Δ achieves superior results by capturing the alignment imprint. Compared with Δ, $\mathcal{S}$ exhibits comparable performance on XSum and attains relative gains of 41.65% and 16.24% on WritingPrompts and Arxiv, respectively. These improvements confirm the suboptimality of Δ and the utility of the information-weighted score. Finally, LAPD outperforms all baselines across the three datasets, demonstrating its effectiveness.

4.4.2 Other Base-Aligned Model Pairs

Table 4 evaluates the sensitivity of LAPD to various model pairs. While the Llama2-7B pair has the best performance, the Falcon-7B, GPT-J-6B, and Llama3.1-8B pairs also demonstrate robust performance, achieving average AUROCs of 89.01%, 92.98%, and 96.71%, respectively.

4.5 Time Efficiency

Beyond detection accuracy, computational efficiency is a critical factor for AI-generated text detection, especially in real-time scenarios. Table 5 reports the time cost per sample for each method. To ensure a rigorous comparison, we detected 300 instances from the benchmarks and measured the average processing time with a batch size of 1.

While classic methods like Entropy, Likelihood, and LogRank are efficient ($\approx$ 0.3s), their practical utility is limited by bad performance. The other methods process one sample range from 0.35s to 0.80s, among which RAI is the fastest and LAPD achieves the best performance.

Table 5: Per-text time costs (s) of all the methods. Methods with clubs (♣) standardize the scores by perturbing or generating auxiliary sequences.

Methods	Time (s)	Methods (♣)	Time (s)
Entropy	0.2784	DetectGPT♣	17.8047
Likelihood	0.2777	Fast-DetectGPT♣	0.5578
LogRank	0.2851	Lastde++♣	0.6169
Binoculars	0.6549	DNA-DetectLLM♣	0.7696
RAI	0.3506	LAPD♣	0.5792

5 Conclusion

In this paper, we introduce a novel perspective for AI-generated text detection by characterizing the distribution shift induced by LLM alignment. We theoretically formalize it as the *Alignment Imprint*, demonstrating that the log-likelihood ratio can naturally disentangle alignment-induced biases from intrinsic content complexity. Furthermore, we propose the ***L**og-likelihood **A**lignment **P**reference **D**iscrepancy (LAPD)*, a standardized information-weighted statistic. We provide statistical guarantee that LAPD dominates Fast-DetectGPT. We also theoretically show that LAPD improves further when the aligned and base models are close in distribution. Extensive experiments show that LAPD achieves an improvement 45.82% relative to the strongest existing baselines and exhibits exceptional robustness in the low-FPR area.

ACKNOWLEDGEMENT

This work is supported in part by the National Natural Science Foundation of China under grant 62301189, 62576122,62571298, Guangdong Basic and Applied Basic Research Foundation under grant 2026A1515011139. We also thank all anonymous reviewers and the area chair for their helpful comments.

References

[1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

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

[3] Souradip Chakraborty, Amrit Bedi, Sicheng Zhu, Bang An, Dinesh Manocha, and Furong Huang. Position: On the possibilities of ai-generated text detection. In *Forty-first International Conference on Machine Learning*, 2024.

[4] Jiaqi Chen, Xiaoye Zhu, Tianyang Liu, Ying Chen, Chen Xinhui, Yiwen Yuan, Chak Tou Leong, Zuchao Li, Long Tang, Lei Zhang, et al. Imitate before detect: Aligning machine stylistic preference for machine-revised text detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 23559–23567, 2025.

[5] Jiaqi Chen, Xiaoye Zhu, Tianyang Liu, Ying Chen, Chen Xinhui, Yiwen Yuan, Chak Tou Leong, Zuchao Li, Long Tang, Lei Zhang, et al. Imitate before detect: Aligning machine stylistic preference for machine-revised text detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 23559–23567, 2025.

[6] Zheng Chen, Huming Liu, and Jianxiao Tang. Zero-shot detection of llm-generated text via dual-network preference divergence. *Expert Systems with Applications*, page 131212, 2026.

[7] Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025.

[8] Liam Dugan, Alyssa Hwang, Filip Trhlík, Andrew Zhu, Josh Magnus Ludan, Hainiu Xu, Daphne Ippolito, and Chris Callison-Burch. Raid: A shared benchmark for robust evaluation of machine-generated text detectors. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12463–12492, 2024.

[9] Angela Fan, Mike Lewis, and Yann Dauphin. Hierarchical neural story generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 889–898, 2018.

[10] Hao Fang, Jiawei Kong, Tianqu Zhuang, Yixiang Qiu, Kuofeng Gao, Bin Chen, Shu-Tao Xia, Yaowei Wang, and Min Zhang. Your language model can secretly write like humans: Contrastive paraphrase attacks on llm-generated text detectors. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 8596–8613, 2025.

[11] Sebastian Gehrmann, Hendrik Strobelt, and Alexander M Rush. 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, 2019.

[12] Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

[13] Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

[14] Abhimanyu Hans, Avi Schwarzschild, Valeriia Cherepanova, Hamid Kazemi, Aniruddha Saha, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Spotting llms with binoculars: Zero-shot detection of machine-generated text. In *International Conference on Machine Learning*, pages 17519–17537. PMLR, 2024.

[15] Xiaomeng Hu, Pin-Yu Chen, and Tsung-Yi Ho. Radar: Robust ai-text detection via adversarial learning. *Advances in neural information processing systems*, 36:15077–15095, 2023.

[16] Daphne Ippolito, Daniel Duckworth, Chris Callison-Burch, and Douglas Eck. 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, 2020.

[17] Alberto Jiménez-Valverde. Insights into the area under the receiver operating characteristic curve (auc) as a discrimination measure in species distribution modelling. *Global Ecology and Biogeography*, 21(4):498–507, 2012.

[18] Robert Kirk, Ishita Mediratta, Christoforos Nalmpantis, Jelena Luketina, Eric Hambro, Edward Grefenstette, and Roberta Raileanu. Understanding the effects of rlhf on llm generalisation and diversity. In *The Twelfth International Conference on Learning Representations*, 2024.

[19] Hyunseok Lee, Jihoon Tack, and Jinwoo Shin. Remodetect: Reward models recognize aligned llm's generations. *Advances in Neural Information Processing Systems*, 37:2886–2913, 2024.

[20] Runheng Liu, Heyan Huang, Xingchen Xiao, and Zhijing Wu. Zero-shot detection of llm-generated text via implicit reward model. *Advances in Neural Information Processing Systems*, 2025.

[21] Eric Mitchell, Yoonho Lee, Alexander Khazatsky, Christopher D Manning, and Chelsea Finn. Detectgpt: Zero-shot machine-generated text detection using probability curvature. In *International conference on machine learning*, pages 24950–24962. PMLR, 2023.

[22] Shashi Narayan, Shay B Cohen, and Mirella Lapata. Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1797–1807, 2018.

[23] Andreas L Opdahl, Bjørnar Tessem, Duc-Tien Dang-Nguyen, Enrico Motta, Vinay Setty, Eivind Throndsen, Are Tverberg, and Christoph Trattner. Trustworthy journalism through ai. *Data & Knowledge Engineering*, 146:102182, 2023.

[24] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022.

[25] Sayak Paul and Soumik Rakshit. arxiv paper abstracts, 2021. URL `https://www.kaggle.com/datasets/spsayakpaul/arxiv-paper-abstracts`.

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

[27] Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems*, 36:53728–53741, 2023.

[28] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67, 2020.

[29] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

[30] Sarath Sivaprasad, Pramod Kaushik, Sahar Abdelnabi, and Mario Fritz. A theory of response sampling in llms: Part descriptive and part prescriptive. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 30091–30135, 2025.

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

[32] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

[33] Vivek Verma, Eve Fleisig, Nicholas Tomlin, and Dan Klein. Ghostbuster: Detecting text ghostwritten by large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 1702–1717, 2024.

[34] Ben Wang and Aran Komatsuzaki. Gpt-j-6b: A 6 billion parameter autoregressive language model, 2021.

[35] Yuxia Wang, Jonibek Mansurov, Petar Ivanov, Jinyan Su, Artem Shelmanov, Akim Tsvigun, Chenxi Whitehouse, Osama Mohammed Afzal, Tarek Mahmoud, Toru Sasaki, et al. M4: Multi-generator, multi-domain, and multi-lingual 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, 2024.

[36] Junchao Wu, Runzhe Zhan, Derek Wong, Shu Yang, Xinyi Yang, Yulin Yuan, and Lidia Chao. Detectrl: Benchmarking llm-generated text detection in real-world scenarios. *Advances in Neural Information Processing Systems*, 37:100369–100401, 2024.

[37] Junxi Wu, Jinpeng Wang, Zheng Liu, Bin Chen, Dongjian Hu, Hao Wu, and Shu-Tao Xia. Moses: Uncertainty-aware ai-generated text detection via mixture of stylistics experts with conditional thresholds. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 5797–5816, 2025.

[38] Yihuai Xu, Yongwei Wang, Yifei Bi, Huangsen Cao, Zhouhan Lin, Yu Zhao, and Fei Wu. Training-free llm-generated text detection by mining token probability sequences. In *The Thirteenth International Conference on Learning Representations*, 2025.

[39] Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. A survey of large language models. *arXiv preprint arXiv:2303.18223*, 2023.

[40] Chunting Zhou, Pengfei Liu, Puxin Xu, Srinivasan Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, et al. Lima: Less is more for alignment. *Advances in Neural Information Processing Systems*, 36:55006–55021, 2023.

[41] Hongyi Zhou, Jin Zhu, Pingfan Su, Kai Ye, Ying Yang, Shakeel AOB Gavioli-Akilagun, and Chengchun Shi. Adadetectgpt: Adaptive detection of llm-generated text with statistical guarantees. *arXiv preprint arXiv:2510.01268*, 2025.

[42] Xiaowei Zhu, Yubing Ren, Yanan Cao, Xixun Lin, Fang Fang, and Yangxi Li. Reliably bounding false positives: A zero-shot machine-generated text detection framework via multiscaled conformal prediction. *arXiv preprint arXiv:2505.05084*, 2025.

[43] Xiaowei Zhu, Yubing Ren, Fang Fang, Qingfeng Tan, Shi Wang, and Yanan Cao. DNA-DetectLLM: Unveiling AI-generated text via a DNA-inspired mutation-repair paradigm. In *Advances in Neural Information Processing Systems 38 (NeurIPS)*. Curran Associates, Inc., 2025.

A Derivations of the Alignment Distributions

A.1 Derivation in the SFT stage

In this section, we derive the theoretical form of the SFT distribution under the maximum entropy framework. Without loss of generality, the goal is to find a data distribution $P(\boldsymbol{x})$ that maximizes the expected latent quality $V(\boldsymbol{x})$ (representing adherence to instructions and format) while remaining close to the pre-trained base distribution $P_\theta(\boldsymbol{x})$ to preserve linguistic capabilities and knowledge. Formally, we frame this as a constrained optimization problem:

$$\min_P D_{\text{KL}}(P \parallel P_\theta) \quad \text{s.t.} \quad \mathbb{E}_{\boldsymbol{x}\sim P}[V(\boldsymbol{x})] \geq \mu, \tag{9}$$

where μ is a target quality threshold. The Lagrangian for this optimization problem is:

$$\mathcal{L}(P, \lambda, \gamma) = \sum_{\boldsymbol{x}} P(\boldsymbol{x}) \log \frac{P(\boldsymbol{x})}{P_\theta(\boldsymbol{x})} + \gamma \left(\sum_{\boldsymbol{x}} P(\boldsymbol{x}) - 1 \right) - \lambda \left(\sum_{\boldsymbol{x}} P(\boldsymbol{x}) V(\boldsymbol{x}) - \mu \right), \tag{10}$$

where λ is the Lagrange multiplier associated with the quality constraint (related to the inverse temperature), and γ enforces the normalization constraint $\sum P(\boldsymbol{x}) = 1$.

To find the optimal distribution P^*, we take the functional derivative with respect to $P(\boldsymbol{x})$ and set it to zero:

$$\log P^*(\boldsymbol{x}) + 1 - \log P_\theta(\boldsymbol{x}) - \lambda V(\boldsymbol{x}) + \gamma = 0. \tag{11}$$

Rearranging the equation and exponentiating both sides, we can obtain as follow:

$$P^*(\boldsymbol{x}) = P_\theta(\boldsymbol{x}) \exp(\lambda V(\boldsymbol{x})) \exp(-(1+\gamma)). \tag{12}$$

To satisfy the normalization constraint $\sum_{\boldsymbol{x}} P(\boldsymbol{x}) = 1$, the term $\exp(-(1+\gamma))$ must equal the inverse of the partition function. Let us define the partition function Z_1 as:

$$Z_1 = \sum_{\boldsymbol{x}'} P_\theta(\boldsymbol{x}') \exp(\lambda V(\boldsymbol{x}')). \tag{13}$$

Absorbing the constants λ (by redefining the scale of V) and $\exp(-(1+\gamma))$ into a normalization factor $1/Z_1$, we arrive at the Boltzmann distribution:

$$P^*(\boldsymbol{x}) = \frac{1}{Z_1} P_\theta(\boldsymbol{x}) \exp(V(\boldsymbol{x})). \tag{14}$$

The training objective of SFT is maximum likelihood estimation. Although practical models P_{SFT} may not reach optimality, the probability mass of $P_{\text{SFT}}(\boldsymbol{x})$ will be concentrated in regions where $P^*_{\text{SFT}}(\boldsymbol{x})$ is high. Therefore, P_{SFT} is monotonically correlated with the latent quality function $V(\boldsymbol{x})$.

A.2 Derivation in the preference tuning stage

Here we derive the closed-form solution for the preference tuning stage. The objective is to maximize the expected reward $R(\boldsymbol{x})$ subject to a KL-divergence constraint relative to the reference policy (which is usually P_{SFT} in the LLM alignment pipeline):

$$\max_{\phi} \mathcal{J}(\phi) = \mathbb{E}_{\boldsymbol{x}\sim P_\phi}[R(\boldsymbol{x})] - \beta D_{\text{KL}}(P_\phi \parallel P_{\text{ref}}). \tag{15}$$

Expanding the KL divergence term:

$$\begin{aligned} \mathcal{J}(\phi) &= \sum_{\boldsymbol{x}} P_\phi(\boldsymbol{x}) R(\boldsymbol{x}) - \beta \sum_{\boldsymbol{x}} P_\phi(\boldsymbol{x}) \log \frac{P_\phi(\boldsymbol{x})}{P_{\text{ref}}(\boldsymbol{x})} \\ &= \sum_{\boldsymbol{x}} P_\phi(\boldsymbol{x}) \left(R(\boldsymbol{x}) - \beta \log P_\phi(\boldsymbol{x}) + \beta \log P_{\text{ref}}(\boldsymbol{x}) \right). \end{aligned} \tag{16}$$

To find the optimal policy P_{ϕ^*}, we introduce a Lagrange multiplier γ for the normalization constraint $\sum P_\phi(\boldsymbol{x}) = 1$:

$$\mathcal{L}(P_\phi, \gamma) = \mathcal{J}(\phi) - \gamma \left(\sum_{\boldsymbol{x}} P_\phi(\boldsymbol{x}) - 1 \right). \tag{17}$$

Taking the derivative with respect to $P_\phi(\boldsymbol{x})$ and setting to zero:

$$R(\boldsymbol{x}) - \beta(\log P_{\phi^*}(\boldsymbol{x}) + 1) + \beta \log P_{\text{ref}}(\boldsymbol{x}) - \gamma = 0. \tag{18}$$

Rearranging the equation and dividing by β:

$$\log P_{\phi^*}(\boldsymbol{x}) - \log P_{\text{ref}}(\boldsymbol{x}) = \frac{1}{\beta} R(\boldsymbol{x}) - \log Z_2, \tag{19}$$

where $\log Z_2 = \frac{\gamma+\beta}{\beta}$ is a constant with respect to $\boldsymbol{x}$. Let us solve for it using the constraint $\sum P_{\phi^*} = 1$. From this equation, we can write the optimal policy $P_{\phi^*}(\boldsymbol{x})$ as:

$$P_{\phi^*}(\boldsymbol{x}) = \frac{1}{Z_2} P_{\text{ref}}(\boldsymbol{x}) \exp\left(\frac{R(\boldsymbol{x})}{\beta} \right). \tag{20}$$

Summing over all $\boldsymbol{x}'$:

$$1 = \sum_{\boldsymbol{x}'} P^*_\phi(\boldsymbol{x}') = \frac{1}{Z_2} \sum_{\boldsymbol{x}'} P_{\text{ref}}(\boldsymbol{x}') \exp\left(\frac{R(\boldsymbol{x}')}{\beta} \right). \tag{21}$$

Then we obtain the constant

$$Z_2 = \sum_{\boldsymbol{x}'} P_{\text{ref}}(\boldsymbol{x}') \exp\left(\frac{R(\boldsymbol{x}')}{\beta} \right). \tag{22}$$

B Proofs

To begin, we first present two technical assumptions, which is analogous to the setting introduced in [41].

Assumption 4. *Define*

$$\sigma_{T,S_{LAPD}} = \frac{\sqrt{\sum_{t=1}^{T} Var_{\tilde{x}_t \sim q_t} S(\tilde{x}_t | x_{<t})}}{\sqrt{\sum_{t=1}^{T} Var_{\tilde{x}_t \sim p_t} S(\tilde{x}_t | x_{<t})}}.$$

Similarly, we define $\sigma_{T,S_{Align}}$ and $\sigma_{T,S_{Align}}$. Then $\sigma_{T,S_{LAPD}} - \sigma_{T,S_{Align}} \to 0$, $\sigma_{T,S_{Align}} - \sigma_{T,S_{Align}} \to 0$, in probability.

Assumption 5. *Define*

$$\bar{\sigma}_{T,S_{LAPD}} = \frac{\sqrt{\sum_{t=1}^{T} Var_{\tilde{x}_t \sim q_t} S(\tilde{x}_t|x_{<t})}}{\sqrt{\sum_{t=1}^{T} Var_{\tilde{\boldsymbol{x}} \sim \boldsymbol{q}} S(\tilde{x}_t|\tilde{x}_{<t})}}.$$

Similarly, we define $\bar{\sigma}_{T,S_{Align}}$ and $\bar{\sigma}_{T,S_{Align}}$. If $\boldsymbol{x} \sim \boldsymbol{q}$, then $\bar{\sigma}_{T,S_{LAPD}} \to 1,\ \bar{\sigma}_{T,S_{Align}} \to 1,\ \bar{\sigma}_{T,S_{Align}} \to 1$, in probability.

It is worth noting that, Assumption 3 in [41] basically requires the conditional variance of logits be asymptotically equivalent for human-written text and AI-generated text. However, this may not hold in real-world scenarios since human-written text tends to exhibit higher variance due to lexical and semantic diversity. In contrast, our Assumption 4 only requires the ratio of the sentence-level variance under AI-generated and human-written text to be asymptotically equivalent across the three statistics.

The following Lemma 1 parallels the Lemma S1 in [41], which can be viewed as a non-asymptotic version of martingale central limit theorem. Lemma 1 is essential for proving our main theorems.

Lemma 1. *Let $\boldsymbol{x} = (x_1, \dots x_n)$ be sequences of real valued random variables satisfying for all $1 \le t \le n$,*

$$\mathbb{E}(x_t|x_{<t}) = 0 \quad \textit{almost surely.}$$

Let $\sigma_t^2 = \mathbb{E}(x_t^2|x_{<t})$, $\bar{\sigma}_t^2 = \mathbb{E}(x_t^2)$, $s_n^2 = \sum_{t=1}^n \bar{\sigma}_t^2$ and $v_n^2 = \sum_{t=1}^n \sigma_t^2/s_n^2$. Suppose $|x_n|$ is bounded by some constant almost surely for all n and $s_n/\sqrt{n}$ is bounded away from zero. Then

$$\sup_{z\in\mathbb{R}} \left| \mathbb{P}\left(\frac{\sum_{t=1}^n x_t}{\sqrt{\sum_{t=1}^n \sigma_t^2}} \le z \right) - \Phi(z) \right| = O\left(\frac{\log n}{\sqrt{n}} + (\mathbb{E}|v_n^2 - 1|)^{1/3} \right),$$

where $\Phi(\bullet)$ is the cumulative distribution function of standard normal distribution.

B.1 Proof of Theorem 1

By Lemma 1, we obtain that

$$\begin{aligned}
\text{TNR}_\tau(S_{\text{Align}}) &= \mathbb{P}_{x\sim p}(S_{\text{Align}}(x) \le \tau) \\
&= \mathbb{P}_{x\sim p}\left(\frac{\sum_{t=1}^T \Delta(\tilde{x}_t|x_{<t}) - \sum_{t=1}^T \mathbb{E}_{\tilde{x}_t\sim q_t}\Delta(\tilde{x}_t|x_{<t})}{\sqrt{\sum_{t=1}^T \text{Var}_{\tilde{x}_t\sim q_t}\Delta(\tilde{x}_t|x_{<t})}} \le \tau \right) \\
&= \mathbb{P}_{x\sim p}\left(\frac{\sum_{t=1}^T \Delta(\tilde{x}_t|x_{<t}) - \sum_{t=1}^T \mathbb{E}_{\tilde{x}_t\sim q_t}\Delta(\tilde{x}_t|x_{<t})}{\sqrt{\sum_{t=1}^T \text{Var}_{\tilde{x}_t\sim q_t}\Delta(\tilde{x}_t|x_{<t})}} \cdot \sigma_{T,S_{\text{Align}}} \le \tau \cdot \sigma_{T,S_{\text{Align}}} \right) \\
&= \mathbb{P}_{x\sim p}\left(\frac{\sum_{t=1}^T \Delta(\tilde{x}_t|x_{<t}) - \sum_{t=1}^T \mathbb{E}_{\tilde{x}_t\sim p_t}\Delta(\tilde{x}_t|x_{<t})}{\sqrt{\sum_{t=1}^T \text{Var}_{\tilde{x}_t\sim p_t}\Delta(\tilde{x}_t|x_{<t})}} \le \tau \cdot \sigma_{T,S_{\text{Align}}} + \right. \\
&\qquad \left. \frac{\sum_{t=1}^T \mathbb{E}_{\tilde{x}_t\sim q_t}\Delta(\tilde{x}_t|x_{<t}) - \sum_{t=1}^T \mathbb{E}_{\tilde{x}_t\sim p_t}\Delta(\tilde{x}_t|x_{<t})}{\sqrt{\sum_{t=1}^T \text{Var}_{\tilde{x}_t\sim q_t}\Delta(\tilde{x}_t|x_{<t})}} \right) \\
&= \mathbb{E}_{x\sim p}\Phi\left(\tau \cdot \sigma_{T,S_{\text{Align}}} + \frac{\sum_{t=1}^T \mathbb{E}_{\tilde{x}_t\sim q_t}\Delta(\tilde{x}_t|x_{<t}) - \sum_{t=1}^T \mathbb{E}_{\tilde{x}_t\sim p_t}\Delta(\tilde{x}_t|x_{<t})}{\sqrt{\sum_{t=1}^T \text{Var}_{\tilde{x}_t\sim q_t}\Delta(\tilde{x}_t|x_{<t})}} \right) + o_p(1)
\end{aligned} \tag{23}$$

The last equality is obtained by Lemma 1 and Assumption 5. Similarly, we have

$$\text{TNR}_\tau(S_{Align}) = \mathbb{E}_{x\sim p}\Phi\left(\tau \cdot \sigma_{T,S_{\text{Align}}} + \frac{\sum_{t=1}^T \mathbb{E}_{\tilde{x}_t\sim q_t}\log P_\phi(\tilde{x}_t|x_{<t}) - \sum_{t=1}^T \mathbb{E}_{\tilde{x}_t\sim p_t}\log P_\phi(\tilde{x}_t|x_{<t})}{\sqrt{\sum_{t=1}^T \text{Var}_{\tilde{x}_t\sim q_t}\log P_\phi(\tilde{x}_t|x_{<t})}} \right) + o_p(1).$$

By Assumptions 1 and 2, we obtain that:

$$\begin{aligned}
&\mathbb{E}_{x\sim p}\Phi\left(\tau\cdot\sigma_{T,S_{\text{Align}}}+\frac{\sum_{t=1}^{T}\mathbb{E}_{\tilde{x}_t\sim q_t}\Delta(\tilde{x}_t|x_{<t})-\sum_{t=1}^{T}\mathbb{E}_{\tilde{x}_t\sim p_t}\Delta(\tilde{x}_t|x_{<t})}{\sqrt{\sum_{t=1}^{T}\text{Var}_{\tilde{x}_t\sim q_t}\Delta(\tilde{x}_t|x_{<t})}}\right)\\
&\geq\mathbb{E}_{x\sim p}\Phi\left(\tau\cdot\sigma_{T,S_{\text{Align}}}+\frac{\sum_{t=1}^{T}\mathbb{E}_{\tilde{x}_t\sim q_t}\Delta(\tilde{x}_t|x_{<t})-\sum_{t=1}^{T}\mathbb{E}_{\tilde{x}_t\sim p_t}\Delta(\tilde{x}_t|x_{<t})}{\sqrt{\sum_{t=1}^{T}\text{Var}_{\tilde{x}_t\sim q_t}\log P_\phi(\tilde{x}_t|x_{<t})}}\right)\\
&=\mathbb{E}_{x\sim p}\Phi\left(\tau\cdot\sigma_{T,S_{\text{Align}}}+\frac{\sum_{t=1}^{T}\mathbb{E}_{\tilde{x}_t\sim q_t}\log P_\phi(\tilde{x}_t|x_{<t})-\sum_{t=1}^{T}\mathbb{E}_{\tilde{x}_t\sim q_t}\log P_\phi(\tilde{x}_t|x_{<t})}{\sqrt{\sum_{t=1}^{T}\text{Var}_{\tilde{x}_t\sim p_t}\log P_\phi(\tilde{x}_t|x_{<t})}}\right)+o_p(1),
\end{aligned}\tag{24}$$

where the last equality arises from $\sigma_{T,S_{\text{Align}}}-\sigma_{T,\Delta}=o_p(1)$ in Assumption 4 and $\sum_{t=1}^{T}\mathbb{E}_{\tilde{x}_t\sim q_t}\log P_\theta(\tilde{x}_t|x_{<t})-\mathbb{E}_{\tilde{x}_t\sim p_t}\log P_\theta(\tilde{x}_t|x_{<t})=o(T)$ in Assumption 1. This completes the proof.

B.2 Proof of Theorem 2

Following the same proof strategy as in Theorem 1, we have

$$\text{TNR}_\tau(S_{\text{LAPD}})=\mathbb{E}_{x\sim p}\Phi\left(\tau\cdot\sigma_{T,S_{\text{LAPD}}}+\frac{\sum_{t=1}^{T}\mathbb{E}_{\tilde{x}_t\sim p_t}S(\tilde{x}_t|x_{<t})-\sum_{t=1}^{T}\mathbb{E}_{\tilde{x}_t\sim q_t}S(\tilde{x}_t|x_{<t})}{\sqrt{\sum_{t=1}^{T}\text{Var}_{\tilde{x}_t\sim q_t}S_\phi(\tilde{x}_t|x_{<t})}}\right)+o_p(1).$$

We first prove that

$$\text{Var}_{\tilde{x}_t\sim q_t}S_\phi(\tilde{x}_t|x_{<t})\leq(2m_\phi+\sqrt{k}m_\Delta)^2\cdot\text{Var}_{\tilde{x}_t\sim q_t}\Delta(\tilde{x}_t|x_{<t}).$$

For simplicity, let $p_t=-\log P_\phi(\tilde{x}_t|x_{<t}),\Delta_t=\Delta(\tilde{x}_t|x_{<t})$. Note that for random variables X,Y,Z,

$$\begin{aligned}
\text{Var}(X+Y+Z)&=\text{Var}(X)+\text{Var}(Y)+\text{Var}(Z)+2\text{Cov}(X,Y)+2\text{Cov}(X,Z)+2\text{Cov}(Y,Z)\\
&\leq\text{Var}(X)+\text{Var}(Y)+\text{Var}(Z)+2\sqrt{\text{Var}(X)\text{Var}(Y)}\\
&\quad+2\sqrt{\text{Var}(X)\text{Var}(Z)}+2\sqrt{\text{Var}(Y)\text{Var}(Z)}\\
&=(\sqrt{\text{Var}(X)}+\sqrt{\text{Var}(Y)}+\sqrt{\text{Var}(Z)})^2.
\end{aligned}$$

Hence,

$$\begin{aligned}
\sqrt{\text{Var}_{\tilde{x}_t\sim q_t}S_\phi(\tilde{x}_t|x_{<t})}&=\sqrt{\text{Var}\left[p_t(\Delta_t-\mathbb{E}\Delta_t)+\Delta_t(p_t-\mathbb{E}p_t)-(\Delta_t-\mathbb{E}\Delta_t)(p_t-\mathbb{E}p_t)\right]}\\
&\leq\sqrt{\text{Var}\left[p_t(\Delta_t-\mathbb{E}\Delta_t)\right]}+\sqrt{\text{Var}\left[\Delta_t(p_t-\mathbb{E}p_t)\right]}\\
&\quad+\sqrt{\text{Var}\left[(\Delta_t-\mathbb{E}\Delta_t)(p_t-\mathbb{E}p_t)\right]}\\
&\leq m_\phi\sqrt{\text{Var}\Delta_t}+m_\Delta\sqrt{\text{Var}\,p_t}+\sqrt{\mathbb{E}(\Delta_t-\mathbb{E}\Delta_t)^2(p_t-\mathbb{E}p_t)^2}\\
&\leq(2m_\phi+\sqrt{k}m_\Delta)\text{Var}\Delta_t=(2m_\phi+\sqrt{k}m_\Delta)\sqrt{\text{Var}_{\tilde{x}_t\sim q_t}\Delta(\tilde{x}_t|x_{<t})},
\end{aligned}\tag{25}$$

where the last inequality follows from $0\leq p_t\leq m_\phi$, hence $|p_t-\mathbb{E}p_t|^2\leq m_\phi^2$, almost surely.

Assumption 3 implies that

$$\begin{aligned}
&\mathbb{E}_{\tilde{x}_t\sim p_t}S(\tilde{x}_t|x_{<t})-\mathbb{E}_{\tilde{x}_t\sim p_t}\log P_\phi(\tilde{x}_t|x_{<t})\cdot\mathbb{E}_{\tilde{x}_t\sim p_t}\Delta(\tilde{x}_t|x_{<t})\\
&\geq\mathbb{E}_{\tilde{x}_t\sim q_t}S(\tilde{x}_t|x_{<t})-\mathbb{E}_{\tilde{x}_t\sim q_t}\log P_\phi(\tilde{x}_t|x_{<t})\cdot\mathbb{E}_{\tilde{x}_t\sim q_t}\Delta(\tilde{x}_t|x_{<t}),
\end{aligned}$$

then we have

$$\begin{aligned}
&\frac{\sum_{t=1}^{T}\mathbb{E}_{\tilde{x}_t\sim p_t}S(\tilde{x}_t|x_{<t})-\sum_{t=1}^{T}\mathbb{E}_{\tilde{x}_t\sim q_t}S(\tilde{x}_t|x_{<t})}{\sqrt{\sum_{t=1}^{T}\mathrm{Var}_{\tilde{x}_t\sim q_t}S_\phi(\tilde{x}_t|x_{<t})}}\\
&\geq\frac{\sum_{t=1}^{T}\mathbb{E}_{\tilde{x}_t\sim p_t}\log P_\phi(\tilde{x}_t|x_{<t})\cdot\mathbb{E}_{\tilde{x}_t\sim p_t}\Delta(\tilde{x}_t|x_{<t})-\sum_{t=1}^{T}\mathbb{E}_{\tilde{x}_t\sim q_t}\log P_\phi(\tilde{x}_t|x_{<t})\cdot\mathbb{E}_{\tilde{x}_t\sim q_t}\Delta(\tilde{x}_t|x_{<t})}{(2m_\phi+\sqrt{k}m_\Delta)\sqrt{\sum_{t=1}^{T}\mathrm{Var}_{\tilde{x}_t\sim q_t}\Delta(\tilde{x}_t|x_{<t})}}\\
&\geq\frac{\sum_{t=1}^{T}\mathbb{E}_{\tilde{x}_t\sim p_t}\Delta(\tilde{x}_t|x_{<t})\cdot(\mathbb{E}_{\tilde{x}_t\sim p_t}\log P_\phi(\tilde{x}_t|x_{<t})-\mathbb{E}_{\tilde{x}_t\sim q_t}\log P_\phi(\tilde{x}_t|x_{<t}))}{(2m_\phi+\sqrt{k}m_\Delta)\sqrt{\sum_{t=1}^{T}\mathrm{Var}_{\tilde{x}_t\sim q_t}\Delta(\tilde{x}_t|x_{<t})}}\\
&\geq\frac{\sum_{t=1}^{T}\mathbb{E}_{\tilde{x}_t\sim q_t}\Delta(\tilde{x}_t|x_{<t})-\sum_{t=1}^{T}\mathbb{E}_{\tilde{x}_t\sim p_t}\Delta(\tilde{x}_t|x_{<t})}{\sqrt{\sum_{t=1}^{T}\mathrm{Var}_{\tilde{x}_t\sim q_t}\Delta(\tilde{x}_t|x_{<t})}},
\end{aligned}\tag{26}$$

where the last inequality follows from the assumption that

$$\begin{aligned}
\frac{\mathbb{E}_{\tilde{x}_t\sim p_t}\Delta(\tilde{x}_t|x_{<t})-\mathbb{E}_{\tilde{x}_t\sim q_t}\Delta(\tilde{x}_t|x_{<t})}{\mathbb{E}_{\tilde{x}_t\sim p_t}\Delta(\tilde{x}_t|x_{<t})}&\leq\frac{\epsilon}{2m_\phi+\sqrt{k}m_\Delta}\\
&\leq\frac{\mathbb{E}_{\tilde{x}_t\sim q_t}\log P_\phi(\tilde{x}_t|x_{<t})-\mathbb{E}_{\tilde{x}_t\sim p_t}\log P_\phi(\tilde{x}_t|x_{<t})}{(2m_\phi+\sqrt{k}m_\Delta)},
\end{aligned}$$

Hence, $\mathrm{TNR}_\tau(S_{LAPD})\succeq\mathrm{TNR}_\tau(S_{Align})$.

B.3 Empirical Validation of Assumptions

To verify that the distributional relations underlying these assumptions are already visible in finite samples, we extract the logits of all tokens in each benchmark and compute the corresponding token-level averages. Table 6 reports the results.

Assumption 1 states that the base model is approximately insensitive to the text source in the aggregate. Empirically, the per-token base-model gap $\mathbb{E}_q[\log P_\theta]-\mathbb{E}_p[\log P_\theta]$ ranges from 0.0853 to 0.6446. The aligned-model gap is consistently positive and ranges from 0.3369 to 0.8780, with an average of 0.6248. This average exceeds that of the base model by 50.7%, supporting the alignment-induced shift formalized by Assumption 2.

The variance rows provide a further check of Assumption 2. Across all five benchmarks, $\mathrm{Var}_q[\log P_\phi]$ is greater than $\mathrm{Var}_q[\Delta]$, with ratios ranging from 3.9802 to 7.3495. A positive ϵ is only required for Theorem 2. Theorem 1 also holds with $\epsilon=0$. Finally, Assumption 3 requires the covariance under human text to be no smaller than that under AI text, and this relation holds on every benchmark. Assumption 3 is used only in the proof of Theorem 2.

These statistics are provided to demonstrate the empirical plausibility of the assumptions rather than as a formal replacement for them.

C Details of Experimental Settings

C.1 Additional Implementation Details

Consistent with the settings in Zhu et al. [43], we use Falcon-7B-Instruct [26] as the scoring (reference) model for all baselines. For the perturbation (observer) model, Fast-DetectGPT, Lastde++, Binoculars, and DNA-DetectLLM utilize Falcon-7B [26], while DetectGPT employs T5-3B [28] and is restricted to 10 perturbations per sample due to computational overhead. We set the sample size to 10,000 for both Fast-DetectGPT and LAPD. For Lastde++, we follow the recommended configuration of $s=4,\epsilon=8,\tau'=15$, with 100 generated samples per detection. During testing, input texts length are truncated to a maximum length of 1024 tokens, while DetectGPT is capped at 512 tokens due to the position limit of T5-3B. All experiments are conducted on two NVIDIA RTX 3090 GPUs (24GB VRAM each). Default temperature, top-k and other generation parameters are used unless stated otherwise.

Table 6: Empirical validation of the distributional statistics used in Assumptions 1–3. Entries are averages over token-level statistics on each benchmark.

Statistic	M4	DetectRL Domain	DetectRL LLM	RAID	RealDet
Base model					
$\mathbb{E}_q[\log P_\theta]$	-1.4206	-1.8007	-1.8478	-1.4551	-1.8934
$\mathbb{E}_p[\log P_\theta]$	-1.8513	-1.8860	-2.2089	-2.0065	-2.5380
Gap $(q-p)$	**0.4307**	**0.0853**	**0.3611**	**0.5514**	**0.6446**
Aligned model					
$\mathbb{E}_q[\log P_\phi]$	-1.6693	-1.9270	-1.9310	-1.5876	-2.0486
$\mathbb{E}_p[\log P_\phi]$	-2.1796	-2.2639	-2.5737	-2.3435	-2.9266
Gap $(q-p)$	**0.5103**	**0.3369**	**0.6427**	**0.7559**	**0.8780**
$\mathrm{Var}_q[\log P_\phi]$	6.6297	8.2167	8.4087	6.2520	8.0638
$\mathrm{Var}_q[\Delta]$	1.3169	1.1180	1.3139	1.4552	2.0260
Variance ratio	**5.0343**	**7.3495**	**6.3998**	**4.2963**	**3.9802**
Covariance					
$\mathrm{Cov}_p(\Delta, \log P_\phi)$	2.1782	2.5682	2.3666	2.2080	2.7890
$\mathrm{Cov}_q(\Delta, \log P_\phi)$	2.0641	1.7991	1.9692	1.8423	2.7293
$\mathrm{Cov}_p \geq \mathrm{Cov}_q$	**Yes**	**Yes**	**Yes**	**Yes**	**Yes**

C.2 Details of Used LLMs

A complete list of the LLMs used in this study is provided in Table 7. All these models are open-sourced and can be downloaded from HuggingFace. We primarily select models with approximately 7B parameters. This consistency in model scale ensures that the observed performance variations are attributable to the detection methods themselves rather than differences in model capacity.

Table 7: Details of the LLMs used.

Model	Model File	Organization
Llama-2-7B [32]	`meta-llama/Llama-2-7b-hf`	Meta
Llama-2-7B-Instruct [32]	`meta-llama/Llama-2-7b-chat-hf`	Meta
Falcon-7B [26]	`tiiuae/falcon-7b`	TII
Falcon-7B-Instruct [26]	`tiiuae/falcon-7b-instruct`	TII
T5-3B [28]	`google-t5/t5-3b`	Google AI
GPT-J-6B [34]	`EleutherAI/gpt-j-6b`	EleutherAI
GPT-J-6B-Instruct [34]	`Provectus/gpt-j-6b-instruct`	EleutherAI
Llama-3.1-8B [12]	`meta-llama/Llama-3.1-8B`	Meta
Llama-3.1-8B-Instruct [12]	`meta-llama/Llama-3.1-8B-Instruct`	Meta

C.3 Details of Benchmarks

The following sections give a brief introduction about the characteristics of the four benchmarks we used in this study. The detailed domains and source LLMs each benchmark contains are listed in Table 8.

M4 [35]. The Multi-generator, Multi-domain, and Multi-lingual dataset is a large-scale benchmark for AI-generated text detection in black-box scenarios. It contains human and AI texts across 7 domains and 7 languages. AI texts are generated by 6 model families, including GPT-4, ChatGPT and BLOOMz.

DetectRL [36]. This benchmark dataset evaluates the reliability of detectors in realistic scenarios by simulating human revisions, prompts, and writing noises. The **Multi-LLM** setting uses generations

from models like GPT-3.5-Turbo, Claude-Instant, and Llama-2-70B to assess the model-agnosticism, while **Multi-Domain** spans academic abstracts, news, creative stories, and social reviews.

RAID [8]. This is a large-scale benchmark for assessing the robustness of detectors across 11 models and 8 domains. It contains 11 types of adversarial attacks, along with 4 different decoding strategies, providing a rigorous environment to test how well detection methods handle realistic modifications.

RealDet [42]. RealDet is a high-reliability bilingual benchmark that emphasizes realistic detection settings and threshold calibration. It spans 15 distinct textual domains and 22 popular LLMs, notably including recent reasoning models such as DeepSeek-R1.

Table 8: The comprehensive landscape of evaluation benchmarks, detailing the domains and generative source models of each benchmark.

Benchmarks	Domains	Source LLMs
M4	Wikipedia, WikiHow, Reddit, arXiv, PeerRead, Baike/Web QA, News, Arabic Wikipedia.	GPT-4, ChatGPT, GPT-3.5, Cohere, Dolly-v2, BLOOMz-176B.
DetectRL	Academic Abstracts, News Articles, Creative Stories, Social Reviews.	GPT-3.5-turbo, PaLM-2-bison, Claude-instant, Llama-2-70b.
RAID	Abstracts, Books, News, Poetry, Recipes, Reddit, Reviews, Wikipedia, Python Code, Czech News, German News.	GPT-2 XL, GPT-3, ChatGPT, GPT-4, Cohere, Mistral-7B, MPT-30B, LLaMA 2 70B.
RealDet	ELI5, WikiQA, Wikipedia, Medical Dialog, FiQA, XSum, TLDR, BBC News, WritingPrompt, ROC Stories, Yelp, IMDB, CMV, Abstracts, SQuAD, WebTextQA, BaikeQA, BaiduBaike, NLPCC-DBQA, FinanceZhidao, Chinese Psychological QA, LegalQA.	DeepSeek-R1, GPT-4, ChatGPT, PaLM 2, Ernie Bot, Spark Desk, Qwen turbo, 360GPT S2 V9, Minimax abab 5.5, LLaMA2-13B, ChatGLM2-6B, MOSS-moon-003, MPT-7B, InternLM-7B, Alpaca-7B, Guanaco-7B, Vicuna-13B, BLOOMz-7B, Falcon-7B, OPT-6.7B, Baichuan-13B, Flan-T5-XXL.

D Additional Results

D.1 Details of TPR at low FPR

The complete results of TPR at 0.5% FPR across different domains are presented in Table 9.

Existing methods generally exhibit poor performance under this strict metric: Fast-DetectGPT, Binoculars, and DNA-DetectLLM achieve average TPRs of 51.79%, 66.67%, and 59.56%, respectively.

On the contrary, RAI demonstrates performance comparable to the best baseline, while LAPD attains an average TPR of 92.27%, obtaining a substantial relative improvement of 76.81%. Notably, on the XSum dataset, most baselines suffer significant performance degradation. However, both RAI and LAPD maintain robust detection abilities, consistently outperforming other methods.

D.2 ROC Curves on various models and datasets

Figure 5 presents the log-scale ROC curves of five representative methods across three datasets (columns) and three source models (rows). To emphasize the high-confidence detection performance, the FPR is visualized on a logarithmic scale starting from 0.1% to 100%.

Across all experimental configurations, RAI shows competitive results, while LAPD consistently maintains a significantly higher TPR compared to all the baselines, particularly in the minimal FPR from 0.1% to 1%. These observations validate the cross-domain and cross-model robustness of LAPD in practical scenarios requiring minimal FPRs.

Table 9: Detailed detection results of TPR (%) at 0.5% FPR across three datasets. Source models are abbreviated as: **GPT** (GPT-4 Turbo), **Gemini** (Gemini-2.0 Flash), and **Claude** (Claude-3.7 Sonnet). The best results are highlighted in **bold**. Methods with clubs (♣) standardize the scores by perturbing or generating auxiliary sequences.

Methods	**XSum**			**WritingPrompts**			**Arxiv**			**Avg.**
	GPT	Gemini	Claude	GPT	Gemini	Claude	GPT	Gemini	Claude	
Entropy	0.30	3.90	0.30	0.10	0.20	0.00	0.60	0.00	0.10	0.61
Likelihood	0.00	0.10	0.00	8.80	51.40	16.20	1.90	27.10	7.00	12.50
LogRank	0.10	0.70	0.10	10.40	50.00	16.80	2.40	31.90	7.20	13.29
DetectGPT♣	0.30	0.30	0.30	0.70	1.00	0.40	0.00	0.30	0.10	0.38
Fast-DetectGPT♣	63.50	57.60	27.70	41.10	87.70	16.40	23.20	90.40	58.50	51.79
Lastde++♣	51.70	51.20	19.00	22.00	78.70	5.20	23.10	85.90	54.90	43.52
Binoculars	57.30	77.60	56.30	52.30	97.40	66.20	42.70	94.30	55.90	66.67
DNA-DetectLLM♣	76.00	1.30	48.30	64.80	85.60	56.40	32.10	96.90	74.60	59.56
RAI	86.50	76.70	69.90	62.40	79.90	50.90	57.60	81.80	63.60	69.92
LAPD♣	**97.90**	**85.00**	**97.20**	**92.60**	**94.70**	**95.40**	**68.20**	**99.60**	**99.80**	**92.27**

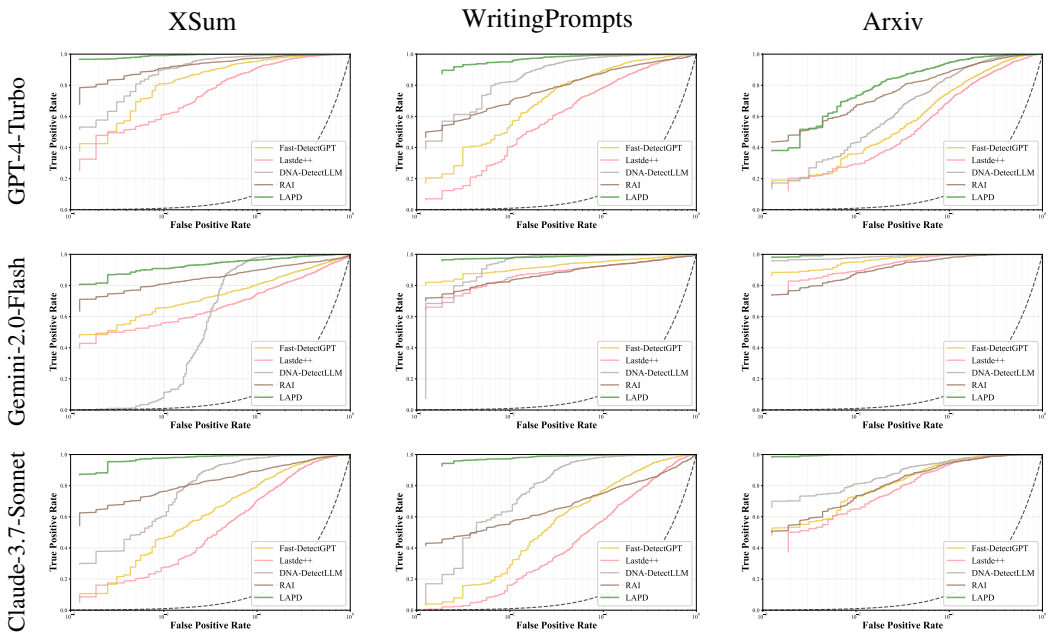


Figure 5: Log-scale ROC curves evaluated across three datasets (columns) and three source models (rows). The dashed lines denote the random classifiers.

D.3 Cross-Domain Threshold Transfer

The experiments above calibrate the detection threshold within each dataset. In practical deployments, repeating this calibration for every target domain may not be feasible. We therefore test whether a single threshold can be transferred across domains. Specifically, for each detector, we determine the threshold that yields an FPR of 0.5% on XSum and keep this threshold fixed when evaluating WritingPrompts and Arxiv.

Table 10 reports the resulting FPR and TPR. LAPD is the only method that transfers a single threshold to the other domains while maintaining an FPR below 2% and a TPR above 90%. These results demonstrate the robustness of LAPD to threshold selection and support its practical viability in real-world deployment.

Table 10: Cross-domain threshold transfer results (%). For each method, the threshold is calibrated on XSum to achieve an FPR of 0.5% and is then kept fixed on WritingPrompts and Arxiv.

Methods	Threshold	XSum		WritingPrompts		Arxiv	
		FPR	TPR	FPR	TPR	FPR	TPR
Fast-DetectGPT	3.0593	0.5	53.8	1.5	64.0	0.0	39.8
Binoculars	-0.8594	0.5	64.2	0.4	65.5	0.3	61.8
DNA-DetectLLM	-0.5119	0.5	67.8	0.0	72.0	4.3	92.7
LAPD	2.3692	0.5	93.3	**0.6**	**96.3**	**1.2**	**91.5**

Table 11: Robustness against various editing attacks (AUROC %). We evaluate three types of random editing: Insert, Delete, and Replace at a 1% token ratio. The best results are highlighted in **bold**. Methods with clubs (♣) standardize the scores by perturbing or generating auxiliary sequences.

Methods	GPT-4 Turbo			Gemini-2.0 Flash			Claude-3.7 Sonnet		
	Insert	Delete	Replace	Insert	Delete	Replace	Insert	Delete	Replace
Entropy	57.66	63.51	55.59	65.70	71.20	63.58	73.42	78.36	70.98
Likelihood	51.54	60.73	49.91	76.39	82.85	74.74	64.13	73.03	61.94
LogRank	52.21	60.69	50.47	76.79	82.95	74.96	64.24	72.78	61.99
DetectGPT♣	45.71	46.65	46.15	47.77	51.63	47.43	47.18	49.83	46.16
Fast-DetectGPT♣	84.82	90.85	84.57	94.37	95.82	94.16	84.32	91.29	83.23
Lastde++♣	68.89	85.13	68.36	86.36	92.51	85.88	66.78	84.77	66.09
Binoculars	87.69	93.43	87.46	97.43	98.07	97.37	89.93	95.41	89.00
DNA-DetectLLM♣	93.48	96.34	93.16	97.73	98.17	97.70	93.84	96.93	93.32
RAI	92.83	91.81	86.92	95.45	94.58	95.54	92.99	91.44	93.60
LAPD♣	**99.16**	**98.99**	**99.05**	**99.34**	**99.37**	**99.36**	**99.64**	**99.68**	**99.70**

E Experimental Results under Attacks

Table 11 illustrates the performance under attacks. We apply three types of attacks: insertion, deletion, and replacement to 1% of the tokens in AI-generated texts from GPT-4 Turbo, Gemini-2.0 Flash, and Claude-3.7 Sonnet. Experimental results demonstrate that while the baselines suffer from varying degrees of performance degradation, LAPD exhibits negligible performance drop.

Currently, there are many different attack methods [10]. So we further evaluated the robustness of the methods against more realistic attacks from the RAID benchmark. Specifically, we consider four representative attack strategies: semantic-level attacks (paraphrase and synonym substitution) and character-level attacks (homoglyph and zero-width space insertion), together with a no-attack control condition. For each type of attack, we construct a balanced evaluation set of 1,000 samples.

As shown in Table 12, our LAPD method outperforms all baselines, achieving the highest average attack AUROC of 83.36% and the highest clean performance of 89.96%. This experiment further strengthens the rigor and fairness of our evaluation.

F Experimental Results under Cross-family Model Pairs

To analyze the performance of the detectors when the aligned model comes from a different family from the base model, we conducted an ablation study using cross-family model pairs. Note that the methods using a sampling model for perturbation-based standardization such as Fast-DetectGPT, Lastde++, and LAPD, require the sampling model and the scoring model to share a same tokenizer, which prevents them from being applied in this cross-model setting. We therefore report only the results of RAI, which does not rely on a sampling model.

Table 12: Detection performance (AUROC %) under 4 representative attack types. The best results are highlighted in **bold**. Methods with clubs (♣) standardize the scores by perturbing or generating auxiliary sequences.

Method	Homoglyph	Paraphrase	Synonym	Zero-width Space	Attack Avg.	None
Fast-DetectGPT♣	67.21	72.82	80.49	63.36	70.97	84.95
Lastde++♣	65.25	72.65	80.15	61.73	69.95	89.20
Binoculars	71.45	80.25	80.70	64.03	74.11	85.59
DNA-DetectLLM♣	68.29	81.10	82.99	59.83	73.05	87.32
RAI	48.09	68.90	68.50	36.93	55.61	78.06
LAPD♣	**82.71**	**82.53**	**87.09**	**81.10**	**83.36**	**89.96**

Table 13 demonstrates that using a cross-family pair leads to an average AUROC drop ranging from 8.73% to 29.80%. This validates our derivation that the aligned model should come from the base model by the alignment process, so that the detection statistic $\Delta(\boldsymbol{x})$ captures the alignment imprint without interference from pretraining distribution shifts.

Table 13: Detection performance (AUROC %) when the base and instruct models come from different model families. The last row corresponds to the original RAI configuration with a matched Llama2-7B pair. The best results are highlighted in **bold**.

Base/Instruct Model	M4	DetectRL LLM	DetectRL Domain	RAID	RealDet	Avg.
Falcon-7B/Llama2-7B-Instruct	55.87	90.27	92.35	67.22	78.61	76.86
GPT-J-6B/Llama2-7B-Instruct	55.22	84.41	89.38	66.85	67.87	72.75
Llama2-7B/Falcon-7B-Instruct	63.33	53.65	45.33	64.69	65.95	58.59
Llama2-7B/GPT-J-6B-Instruct	53.36	49.66	43.49	60.23	72.24	55.80
Llama2-7B/Llama2-7B-Instruct	**70.46**	**93.13**	**96.19**	**81.28**	**86.92**	**85.60**

G Experimental Results under Same Proxy Model

In the main experiments, we follow the official implementations of the baselines, which use Falcon-7B-Instruct as the scoring model, while LAPD uses the Llama-2-7B model pair for detection. To eliminate the potential confounding factor introduced by the proxy model, we evaluated the baselines using the Llama-2-7B model pair. As summarized in Table 14, LAPD still consistently achieves SOTA performance across all benchmarks, confirming that the performance gains stem from the method itself rather than from the choice of scoring model.

H Comparison with supervised methods

We also compare LAPD with supervised (or hybrid) detection methods. Specifically, we tested three representative baselines, OpenAI Detector [31], ReMoDetect [19], and ImBD [5], and evaluated in four benchmarks.

As demonstrated in Table 15, LAPD achieves the best average AUROC of 92.37%, outperforming all the baselines. This shows that LAPD maintains a significant advantage over strong supervised and hybrid detection methods.

Table 14: Matched comparison of detection performance (AUROC %) across baselines and LAPD. All methods use the Llama2-7B base/instruct pair for detection. The best results are highlighted in **bold**. Methods with clubs (♣) standardize the scores by perturbing or generating auxiliary sequences.

Methods	M4	DetectRL LLM	DetectRL Domain	RAID	RealDet	Avg.
Fast-DetectGPT♣	86.14	92.42	86.42	84.77	94.16	88.78
Lastde++♣	86.69	86.59	78.57	86.88	94.02	86.55
Binoculars	87.27	93.11	88.33	85.30	94.56	89.72
DNA-DetectLLM♣	85.79	92.98	87.72	84.55	94.63	89.13
RAI	70.46	93.13	**96.19**	81.28	86.92	85.60
LAPD♣	**88.02**	**97.17**	96.11	**85.21**	**95.32**	**92.37**

Table 15: Detection performance (AUROC, %) of LAPD compared with representative supervised and hybrid baselines. The best results are highlighted in **bold**.

Methods	M4	DetectRL LLM	DetectRL Domain	RAID	RealDet	Avg.
OpenAI Detector	72.24	78.15	74.60	73.37	84.96	76.66
ReMoDetect	74.98	89.11	80.93	66.89	92.18	80.82
ImBD	73.41	83.25	89.81	77.77	92.12	83.27
LAPD	**88.02**	**97.17**	**96.11**	**85.21**	**95.32**	**92.37**

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 introduction clearly state the main contributions: the alignment imprint, the LAPD detector, and the theoretical guarantees. Experimental results on multiple benchmarks support these claims.

   Guidelines:

   - The answer [N/A] 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 [N/A] 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 paper discuss the limitations of the work in Section 3 and Section 4.

   Guidelines:

   - The answer [N/A] 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: [Yes]

   Justification: The paper includes formal assumptions with corresponding theoretical results. Proofs are provided in the appendix with cross-references from the main text.

   Guidelines:

   - The answer [N/A] 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: The paper specifies all datasets, source models, evaluation metrics, and the base, aligned model pair used. Sufficient detail is provided to reproduce the main results.

   Guidelines:

   - The answer [N/A] 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: Code for reproducing the main experimental results will be released upon acceptance. The repository includes evaluation scripts for all baselines and the proposed LAPD method.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.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://neurips.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) necessary to understand the results?

   Answer: [Yes]

   Justification: Experimental details including datasets, source models, evaluation protocols, attack configurations, and text length ranges are described in Section 4 and Appendix.

   Guidelines:

   - The answer [N/A] 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: [No]

   Justification: Given the deterministic nature of the detection pipeline and the scale of evaluation across multiple benchmarks, there is no variance estimation.

   Guidelines:

   - The answer [N/A] 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 paper report experimental compute resources in Section 4 and Appendix C.

   Guidelines:

   - The answer [N/A] 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 focuses on AI-generated text detection, which contributes to content integrity and combating misinformation. No human subjects or sensitive data collection is involved.

   Guidelines:

   - The answer [N/A] 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 discuss the impacts in the Section 1 and Section 5.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] 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., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

Justification: The paper does not release new models or datasets with high misuse risk. The proposed method is a detection tool rather than a generative model.

Guidelines:

- The answer [N/A] 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 datasets and models used (XSum, WritingPrompts, Arxiv, Llama2, etc.) are cited with their original papers. License information for publicly available assets is noted where applicable.

Guidelines:

- The answer [N/A] 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: [N/A]

Justification: This paper does not release new assets.

Guidelines:

- The answer [N/A] 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: [N/A]

Justification: The paper does not involve crowdsourcing or human subject experiments. All evaluations are conducted on existing benchmark datasets.

Guidelines:

- The answer [N/A] 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: [N/A]

Justification: No human subjects research is conducted in this paper.

Guidelines:

- The answer [N/A] 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 rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: LLMs (GPT-4 Turbo, Gemini-2.0 Flash, Claude-3.7 Sonnet) are used solely as the source of AI-generated text in evaluation benchmarks, not as components of the proposed methodology.

Guidelines:

- The answer [N/A] 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 in the NeurIPS handbook for what should or should not be described.