Zero-Shot Detection of LLM-Generated Text using Temperature Sensitivity

Shixuan Ma[1], **Jiahao Li**[2], **Zhendong Mao**[2], **Quan Wang**[1]*
[1]MOE Key Laboratory of Trustworthy Distributed Computing and Service,
Beijing University of Posts and Telecommunications, Beijing, China
[2]University of Science and Technology of China, Hefei, China
{shixuanma, wangquan}@bupt.edu.cn, jiahao66@mail.ustc.edu.cn, zdmao@ustc.edu.cn

Abstract

The widespread deployment of Large Language Models (LLMs) has spurred significant progress in the detection of LLM-generated text. However, existing detection methods often rely on statistical features that are insufficient for reliable detection; for example, even though LLM-generated and human-written texts exhibit different probability distributions in surrogate models, they can produce nearly identical entropy values, thereby conflating the two types of text. In this paper, we propose that modulating the decoding temperature and monitoring how the probability distributions respond can better probe the intrinsic discrepancies between two types of text. Building upon this insight, we introduce a new feature termed Temperature Sensitivity (TS) and demonstrate that LLM-generated text tends to exhibit higher TS than human-written text. Finally, we propose NTS, a novel and simple zero-shot detector built upon normalized temperature sensitivity. Extensive experiments across three datasets, multiple domains, and various source models demonstrate the superior effectiveness and robustness of our proposed approach. Code avaliable at: https://github.com/Shixuan-Ma/NTS.

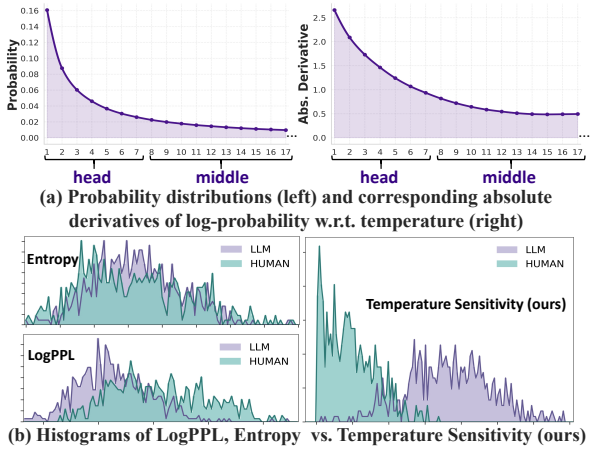

Figure 1: **(a)** Distributions are calculated over the surrogate model's vocabulary for a specific token in each sample (refer to Appendix D.1 for details) and then averaged over 600 human-written samples from XSum and Writing (Yu et al., 2025) and 600 LLM-generated counterparts. The latter are produced by prompting two models with the initial 30 tokens of each human text. Derivative calculations are formulated in Eq. 2. **(b)** Histograms of vanilla LogPPL and Entropy versus Temperature Sensitivity (ours). The dataset used here is identical to that in **(a)**. The calculation of temperature sensitivity is detailed in Section 3.2.

1 Introduction

With the rapid advancement of large language models (LLMs), their powerful generative capabilities have dramatically enhanced productivity across various domains (Peng et al., 2023; Brynjolfsson et al., 2025; Tang et al., 2024). Models such as GPT-4o, Gemini and LLaMA (OpenAI et al., 2024; Gemini Team Google, 2023; Touvron et al., 2023) can now generate coherent and contextually appropriate content in response to human instructions. However, alongside these benefits come emerging risks, including academic misconduct (Eke, 2023), plagiarism (Pudasaini et al., 2024), unauthorized use (Yang et al., 2025), and the spread of misinformation (Pan et al., 2023), etc. As LLMs continue to evolve at a rapid pace, developing more advanced detectors for AI-generated content is crucial to ensuring their responsible and ethical use.

Detecting LLM-generated text is typically formulated as a binary classification task, aimed at determining whether a given text was LLM-generated or human-written (Mitchell et al., 2023; Hashimoto et al., 2019). Existing detection methods are generally categorized into two major classes: supervised detectors and zero-shot detectors. Supervised detectors typically fine-tune a neural-based classifier directly on labeled data (Guo et al., 2024; Verma et al., 2024; Hu et al., 2023). While exhibiting

*Corresponding author: Quan Wang.

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 37664–37679

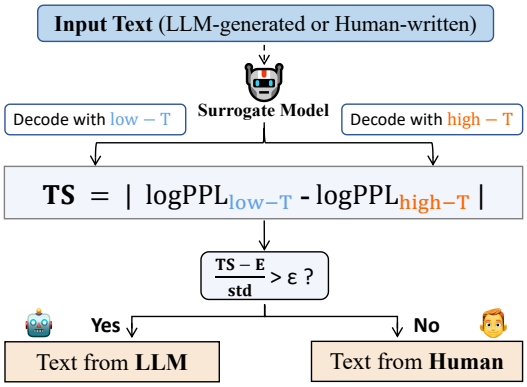

Figure 2: **Overview of the NTS pipeline.**

strong in-domain performance, their generalization capability remains an area for improvement (Tang et al., 2024; Pu et al., 2022). In contrast, zero-shot detectors classify text directly based on statistical features extracted from surrogate models, requiring no training and offering better generalization. Consequently, they have become a primary focus of current research (Su et al., 2023).

However, most existing zero-shot methods primarily rely on statistical features such as vanilla LogPPL and Entropy (Hans et al., 2024; Liu et al., 2025), which tend to assign indistinguishable values to both LLM-generated and human-written text, as illustrated in the left of Figure 1 (b). This is expected, as these statistics are derived from certain reduction operations performed on the probability distributions within surrogate models. Such reduction essentially constitutes an over-compression of high-dimensional information, which inevitably results in significant information loss (Sabour et al., 2017) and thereby limits the discriminative power of these features. Nevertheless, a critical hyperparameter that profoundly influences these statistical features—the decoding temperature—has been largely overlooked, yet we argue it is the key to overcoming this limitation.

In this paper, we propose leveraging decoding temperature as a probe to uncover the intrinsic discrepancies between LLM-generated and human-written text. Our motivation stems from a pivotal observation: while these two types of text tend to yield nearly identical statistics, their probability distributions within the surrogate model remain distinct, with divergent responses when the decoding temperature is varied. This occurs because LLMs typically favor high-probability words during the generation process (Mitchell et al., 2023), causing surrogate model predictions for LLM-generated texts to be concentrated in the head (top) of the probability distribution. In contrast, human writing is not bound by these explicit optimization constraints, with surrogate model predictions for human-written texts typically falling within the middle range of the distribution. As a result, the magnitude of the gradient with respect to temperature is typically larger for the head of the distribution than for the middle portion, as illustrated in Figure 1 (a). Building upon this insight, we introduce a novel feature, **Temperature Sensitivity (TS)**, defined as the absolute difference in a statistical feature (e.g., LogPPL) under high versus low decoding temperatures. Our key assertion is that **LLM-generated text exhibits higher temperature sensitivity than human-written text**. We empirically verify this assertion; as shown in the right of Figure 1 (b), LLM-generated text exhibits significantly higher TS, facilitating a clear separation from human-written text.

Given the discriminative power of the new feature, we present NTS, a novel and simple zero-shot LLM-generated text detector built upon **N**ormalized **T**emperature **S**ensitivity based on LogPPL, as illustrated in Figure 2. As a zero-shot detector, NTS exhibits advantages in both computational efficiency and hyperparameter robustness. Regarding efficiency, NTS requires only a single forward pass of the candidate text through a single surrogate model, coupled with a zero-cost decoding temperature modulation, introducing nearly no additional time or space cost. Concerning hyperparameter, NTS maintains consistently high detection accuracy across a broad range of temperature combinations (low-T and high-T), with no significant fluctuations in performance.

To rigorously evaluate the effectiveness and robustness of our approach, we adopt 12 state-of-the-art baselines, comprising 7 zero-shot detectors (e.g., IRM (Liu et al., 2025)) and 5 supervised detectors, and conduct extensive experiments across three challenging datasets (e.g., HART (Bao et al., 2025)), utilizing LLM-generated passages in five languages produced by over 15 generative models across more than 12 domains. Experimental results demonstrate the overall superior performance of NTS across two mainstream evaluation metrics in diverse scenarios. Furthermore, we also explore training a classifier based on temperature sensitivity. Our results reveal that this TS-based classifier surpasses 5 existing supervised methods, further validating the inherent superiority of temperature

sensitivity.

Our main contributions are as follows: (1) unveiling and validating a new hypothesis that LLM-generated text exhibits higher temperature sensitivity than human-written text, (2) developing a new detection prototype NTS that uses temperature sensitivity to detect LLM-generated text, and (3) achieving new best detection accuracy compared to existing detectors.

2 Related Work

Supervised detectors Supervised approaches typically train a neural-network–based classifier. A particularly practical strategy is to fine-tune a BERT-style model directly on labeled texts (Hello-SimpleAI, 2023; Solaiman et al., 2019), or to employ a PPO-based adversarial training regime to adapt a pre-trained LLM into a detector (Hu et al., 2023). Another approach is to first use a proxy model to extract statistical features, and then train a lightweight classifier (Guo et al., 2024; Verma et al., 2024). Additionally, some work uses the DPO algorithm to align the proxy model with LLM-revised text (Chen et al., 2024), finally calculating a statistical score to classify the text. While these methods achieve excellent performance on their training distributions, they require frequent re-training to accommodate new models and are prone to overfitting (Zhu et al., 2023; Guo et al., 2023).

Zero-shot detectors Zero-shot approaches leverage off-the-shelf proxy models without additional fine-tuning to extract various statistical features of the text. Pioneered by DetectGPT (Mitchell et al., 2023), subsequent work has explored features such as entropy, log-perplexity, and log-rank (Mitchell et al., 2023), likelihood-logrank ratio (LRR) (Su et al., 2023), conditional probability curvature (Fast-DetectGPT) (Bao et al., 2024), normalized log-perplexity (Binoculars) (Hans et al., 2024), and implicit reward score (IRM) (Liu et al., 2025), typically applying a threshold to distinguish between human-written and machine-generated text. However, none of the existing zero-shot methods consider the role of the surrogate model's decoding temperature in detecting the origin of a text, which we identify as a meaningful and informative signal in this paper.

3 Methodology

3.1 Problem Formulation & Motivation

We study zero-shot LLM-generated text detection, which is formulated as a binary classification problem. Given a candidate passage x, the goal is to discern whether x is human-written or generated by an LLM. The problem is zero-shot in the sense that we do not assume access to any labeled samples for detection.

Although existing zero-shot methods, such as Binoculars (Hans et al., 2024), have demonstrated robustness against various source models, we point out that, as shown in the left of Figure 1 (b), the core statistical features they rely on fail to effectively distinguish between LLM-generated and human-written text due to the excessive compression of information incurred by reduction operations (Sabour et al., 2017). Crucially, we recognize that despite the highly similar statistics of both text types, their probability distributions in a surrogate model remain distinct (see Appendix E for case study). This motivates us to exploit the discrepancies in their probability distributions by adjusting the decoding temperature and monitoring how these distributions respond.

3.2 Key Assumption

Definition (Temperature Sensitivity). *Given a candidate text x, the temperature sensitivity of x is then defined as the absolute difference in a statistical feature (SF) under high versus low decoding temperatures, formulated as:*

$$\mathrm{TS}(x) = \left|\mathrm{SF}_{\text{low-T}}(x) - \mathrm{SF}_{\text{high-T}}(x)\right|. \quad (1)$$

Temperature sensitivity quantifies the extent to which a text's probability distribution within a surrogate model responds to modulations of the decoding temperature. The more pronounced the distribution responds, the higher the temperature sensitivity.

We argue that a fundamental gap in temperature sensitivity exists between LLM-generated and human-written text. Specifically, the generation process of LLMs is typically governed by strategies such as greedy, top-p, or top-k decoding, which inherently results in surrogate model predictions for LLM-generated texts being concentrated in the head (top) of the probability distribution. In contrast, human writing is unconstrained by explicit decoding algorithms, causing surrogate model pre-

dictions for these texts to typically fall within the middle range of the distribution.

Based on this premise, we can derive the derivative of the log-probability with respect to the temperature T for a certain word w_j in probability distribution, or specifically, the ground-truth word w_{label}, as follows (see the Appendix A.1 for the detailed derivation):

$$|\frac{\partial \log p(w_{label})}{\partial T}| = |\frac{\hat{\ell} - \ell_{label}}{T^2}|, \tag{2}$$

Here, ℓ_{label} represents the surrogate model's output logits for the ground-truth word, while $\hat{\ell}$ denotes the expected logits over the distribution with a vocabulary of size V.

$$\hat{\ell} = \sum_{j=1}^{V} p(w_j) * \ell_j. \tag{3}$$

Equation 2 shows that, for a fixed temperature T, the absolute value of the derivative of $\log p(w_{label})$ with respect to T depends solely on the quantity $\Delta = \left\| \hat{\ell} - \ell_{label} \right\|$, which measures how much the ℓ_{label} deviates from the expected $\hat{\ell}$ across the entire distribution.

Since surrogate model predictions for LLM-generated text are concentrated in the head—the high-probability region characterized by larger ℓ_{label} that significantly exceed the expected $\hat{\ell}$ value of the distribution—they result in a larger Δ and, consequently, **higher temperature sensitivity**. In contrast, predictions for human-written text primarily fall within the middle range, where ℓ_{label} are smaller and closely align with the distribution's expected $\hat{\ell}$ value. This leads to a smaller Δ and **lower temperature sensitivity**. This correspondence is illustrated from left to right in Figure 1 (a), and we further demonstrate the discrepancy in Δ between LLM-generated text and human-written text in Appendix F.2. Finally, we formalize this assertion as the Temperature Sensitivity Disparity Hypothesis.

Hypothesis (Temperature Sensitivity Disparity). *Let P_{LLM} denote the corpus of LLM-generated text, and P_{Human} that of human-written text. Then, the temperature sensitivity $TS(x)$ tends to be higher for samples $x \sim P_{LLM}$, while lower for $x \sim P_{Human}$.*

We empirically verify the hypothesis in an automated manner. Specifically, we randomly sample 600 LLM-generated texts and 600 human-written texts from the XSum and WritingPrompts (Yu et al., 2025). For each sample, we feed it into a surrogate model and compute the absolute difference in LogPPL under low and high temperatures. Following prior work, we adopt Falcon-7B (Almazrouei et al., 2023) as the surrogate model, with the low temperature set to 0.7 and the high temperature set to 1.4 (see section 4.5 for more details about hyperparameters and surrogate models). Figure 1 (b) shows the results, revealing that the temperature sensitivity does differ significantly between LLM-generated and human-written data. LLM-generated samples typically show higher temperature sensitivity values. Additional figures for temperature sensitivity are provided in the Appendix F.1.

3.3 *NTS*: A LLM-Generated Text Detector

Given the discriminative power of temperature sensitivity, and inspired by prior work (Bao et al., 2024), we propose a novel detector named NTS (Normalized Temperature Sensitivity), as illustrated in Figure 2. To compute this metric, we construct the final decision score step-by-step.

First, we define the text log-perplexity at a specific decoding temperature T. For a given text x of length n, it is calculated as the negative average log-probability of each ground-truth token using a temperature-scaled softmax function:

$$\text{LogPPL}_{\text{T}}(x) = -\frac{1}{n} \sum_{i=1}^{n} \log \left(\text{softmax}(\ell_{i,\text{label}}/T) \right). \tag{4}$$

Building upon this, we compute the raw text-level temperature sensitivity, $\text{TS}(x)$, as the absolute difference in LogPPL under low and high temperatures. This text-level formulation effectively aggregates the temperature sensitivity contributions from all tokens (see Appendix A.2 for the detailed derivation of token-to-text transformation):

$$\text{TS}(x) = \left| \text{LogPPL}_{\text{low-T}}(x) - \text{LogPPL}_{\text{high-T}}(x) \right|. \tag{5}$$

However, the optimal threshold of the raw $\text{TS}(x)$ metric shifts significantly when target domains and source models vary. To minimize these shifts and enhance detection robustness, we formulate the final decision score as a standardized Z-score, which anchors the metric around zero.

To achieve this normalization, we first define the token-level log-probability difference $\delta_i(w_j) = \log \left(P_{\text{low-T}}(w_j) / P_{\text{high-T}}(w_j) \right)$ for any word $w_j \in V$ at the i-th token. Then, $\text{E}(x)$ and $\text{std}(x)$ can be

concisely formulated as:

$$\mathrm{E}(x) = \mathbb{E}_{i,j}\left[\delta_i(w_j)\right], \tag{6}$$

$$\mathrm{std}(x) = \sqrt{\mathbb{E}_{i,j}\left[\delta_i(w_j)^2\right] - \mathrm{E}(x)^2}. \tag{7}$$

Finally, the NTS detection metric is computed using the derived statistics:

$$\mathrm{Score}(x) = \frac{\mathrm{TS}(x) - \mathrm{E}(x)}{\mathrm{std}(x)}. \tag{8}$$

If $\mathrm{Score}(x)$ exceeds a predefined threshold ϵ, the text exhibits high temperature sensitivity and is thus classified as LLM-generated. We explore the stability of the optimal thresholds for NTS in Section 4.4, and detail the improvements in threshold stability achieved through normalization in Appendix C.5.

4 Experiments

4.1 Experimental Setups

Datasets To ensure a fair and comprehensive comparison, we adopt three challenging datasets. The first two serve as our primary benchmarks: RAID (Dugan et al., 2024), comprising 2,000 human-written samples across 8 domains (e.g., reviews) and 2,000 LLM-generated samples from 11 source models (e.g., Llama-chat (Touvron et al., 2023)); and HART (Bao et al., 2025), consisting of 4,000 human-written samples from four domains (e.g., arXiv) and 4,000 LLM-generated samples from 7 source models (e.g., Gemini (Gemini Team Google, 2023)). Notably, the HART benchmark categorizes the detection task into three levels: Level 3 (Relaxed) classifies only fully LLM-generated text as AI, treating all other types (including LLM-polished text) as human-written; Level 2 (Medium) identifies LLM-generated content as AI while still classifying LLM-polished human text as human-written; and Level 1 (Strict) designates only purely human-authored text as human-written, with all remaining content—including AI-polished text—categorized as LLM-generated. Both datasets provide clear text sources, making them well-suited for our main experiments. Additionally, we utilize the Evo dataset (Yu et al., 2025) for further analyses. Detailed settings for these datasets are provided in Appendix B.

Baselines To ensure the validity of our experimental results, we select seven zero-shot detectors as baselines: Entropy, LogPPL, LogRank (Bao et al., 2025), LRR (Su et al., 2023), Fast-DetectGPT (Su et al., 2023), Binoculars (Hans et al., 2024), and the newly proposed IRM (Liu et al., 2025). We choose these seven detectors as they are recently proposed, computationally efficient, and report current state-of-the-art performance. Furthermore, we train a lightweight classifier based on Temperature Sensitivity features and compare it against five existing SOTA supervised classifiers: the RoBERTa-based ChatGPT detector (Hello-SimpleAI, 2023), IMBD (based on DPO preference alignment) (Chen et al., 2024), the AI ghostwriting detector Ghostbuster (Verma et al., 2024), the PPO-based adversarial learning algorithm RADAR (Hu et al., 2023), and Biscope (based on bidirectional cross-entropy) (Guo et al., 2024).

To the best of our knowledge, these baselines represent the most advanced detectors currently available, ensuring the objectivity and fairness of our comparison.

Metric Following prior work, we report AUROC and F1 as the main evaluation metrics.

Implementation All experiments are conducted on NVIDIA A40 GPUs (48GB each). For most of our experiments, we adopt Falcon-7B (Almazrouei et al., 2023) as the surrogate model following the previous work (Hans et al., 2024), with high-T = 1.4 and low-T = 0.7, unless otherwise specified.

4.2 Multi Domains and Multi Source Models

We first evaluate NTS alongside seven state-of-the-art zero-shot baselines on the RAID benchmark, which encompasses a diverse array of domains and multiple source models. We employ AUROC and the F1-score to assess the comprehensive detection reliability and practical accuracy of each method, respectively. The empirical results are presented in Table 1.

In multi-domain scenarios, NTS achieves the highest All AUROC and F1-score across all domains, outperforming Binoculars—the previous leading zero-shot method—by a margin of 5 percentage points in AUROC. These results demonstrate that NTS is currently the most robust zero-shot detection method for cross-domain applications. Regarding the multi-source model setting, NTS consistently surpasses existing methods in AUROC within individual domains containing data from various source models. Notably, NTS exhibits exceptional performance in the "Recipes" domain,

Detector	News	Books	Wiki	Abstr.	Reddit	Recipes	Poetry	Reviews	**ALL**	**F1**
Zero-Shot Detectors										
Entropy (Mitchell et al., 2023)	0.545	0.654	0.624	0.504	0.685	0.582	0.637	0.642	0.585	0.67
LogPPL (Bao et al., 2025)	0.644	0.725	0.701	0.680	0.725	0.627	0.706	0.698	0.663	0.66
Log-Rank (Bao et al., 2025)	0.666	0.745	0.719	0.701	0.735	0.645	0.725	0.716	0.681	0.67
LRR (Su et al., 2023)	0.750	0.816	0.804	0.771	0.779	0.669	0.776	0.773	0.746	0.70
Fast-DetectGPT (Bao et al., 2024) †	0.761	0.845	0.803	0.821	0.794	0.749	0.818	0.810	0.800	0.76
Binoculars (Hans et al., 2024) †	0.768	0.850	0.804	0.826	0.811	0.759	0.826	0.812	0.807	0.77
IRM (Liu et al., 2025) †	0.715	0.712	0.573	0.705	0.647	0.758	0.759	0.660	0.690	0.67
NTS (ours)	**0.844**	**0.888**	**0.844**	**0.897**	**0.841**	**0.844**	**0.862**	**0.831**	**0.856**	**0.78**

Table 1: Detection performance on the RAID benchmark across different domains. We report AUROC for each domain, with the last two columns summarizing ALL AUROC and F1 across all domains, respectively. † denotes methods that require two surrogate models. **Bold** indicates the best result.

Detector	Level-3 Detection Task					Level-2 Detection Task					Level-1 Detection Task				
	ArXiv	Writ.	News	**ALL**	**F1**	ArXiv	Writ.	News	**ALL**	**F1**	ArXiv	Writ.	News	**ALL**	**F1**
Zero-Shot Detectors															
Entropy (Mitchell et al., 2023)	0.660	0.667	0.573	0.656	0.68	0.665	0.529	0.712	0.577	0.67	0.639	0.464	0.687	0.529	0.67
LogPPL (Bao et al., 2025)	0.850	0.810	0.733	0.799	0.75	0.485	0.438	0.596	0.473	0.67	0.530	0.625	0.407	0.576	0.67
Log-Rank (Bao et al., 2025)	0.874	0.813	0.762	0.814	0.77	0.460	0.441	0.571	0.465	0.67	0.542	0.611	0.418	0.573	0.67
LRR (Su et al., 2023)	0.909	0.797	0.841	0.840	0.78	0.616	0.551	0.556	0.573	0.67	0.576	0.558	0.520	0.568	0.67
Fast-DetectGPT (Bao et al., 2024) †	0.877	0.840	0.850	0.862	0.81	0.688	0.692	0.717	0.711	0.68	0.769	0.740	0.711	0.778	0.72
Binoculars (Hans et al., 2024) †	0.882	0.847	**0.866**	0.870	**0.83**	0.715	0.693	0.698	0.711	0.69	0.769	0.740	0.717	0.780	**0.73**
IRM (Liu et al., 2025) †	0.788	0.712	0.718	0.732	0.70	0.778	0.663	0.750	0.697	0.69	0.750	0.636	0.724	0.692	0.67
NTS (ours)	**0.915**	**0.875**	0.853	**0.874**	0.81	**0.840**	**0.817**	**0.780**	**0.794**	**0.72**	**0.818**	**0.802**	**0.726**	**0.801**	**0.73**

Table 2: Detection performance on the HART benchmark under three levels. We report AUROC for different domains in each level, with the last two columns in each level summarizing ALL AUROC and F1 across all domains, respectively. † denotes methods that require two surrogate models. **Bold** indicates the best result.

where it achieves the most significant improvement over existing baselines.

4.3 Different Levels of Task

We then evaluate NTS and seven SOTA baselines on the HART benchmark, which hierarchically defines three detection difficulty levels. The results are presented in Table 2.

Across all three levels, NTS achieves superior overall cross-domain performance. Notably, at Level-2, NTS demonstrates a significant improvement over Binoculars, with an 8 percentage points increase in AUROC and a 3 percentage points increase in the F1-score. Furthermore, NTS maintains its lead at Level-1, consistently outperforming the baselines. These results underscore that NTS is currently the most versatile zero-shot detector for real-world scenarios involving varying degrees of AI-usage restrictions, particularly in contexts where these restrictions are more stringent.

4.4 Robustness in the Wild

Other Languages In the real-world, LLM users are globally distributed, utilizing various languages to generate content. This diversity underscores the critical need for zero-shot detectors to maintain robustness across multilingual contexts. To this end, we leverage the HART benchmark to evaluate NTS against three zero-shot baselines across five languages: English, Chinese, French, Spanish, and Arabic.

As illustrated in the radar plot (Figure 3), NTS consistently achieves a leading position across all three task levels and all tested languages. These results affirm the cross-lingual robustness of NTS, establishing it as the premier zero-shot detector for multilingual scenarios. Interestingly, NTS exhibits a particularly pronounced advantage in Arabic, which is relatively low-resource and represents a small fraction of the pre-training corpus for most surrogate models. This suggests that the temperature sensitivity effectively unlocks the latent potential for detecting generated text even in lower-resource linguistic settings.

Against Perturbation Attacks To verify robustness against common evasion techniques—such as adversarial attacks, decoding strategy modifications, and high-temperature paraphrasing (Bao et al., 2025)—we evaluate NTS on the RAID bench-

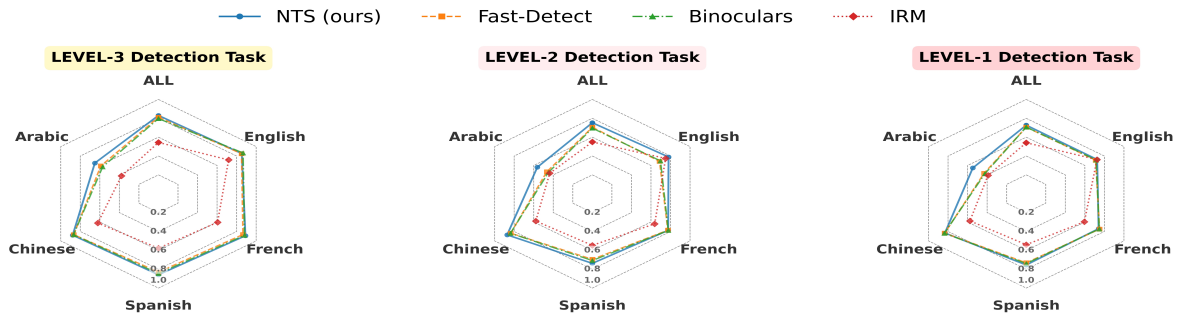

Figure 3: **Multilingual radar chart.** For each language, results represent the ALL AUROC across four domains (essay writing arxiv news); **ALL** denotes the ALL AUROC across all five languages.

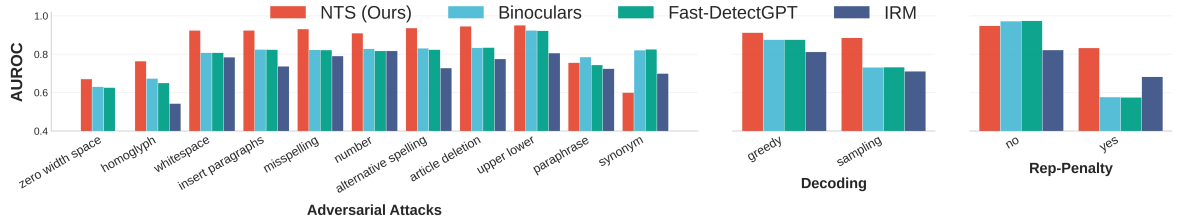

Figure 4: Comparison on detector's ability to handle adversarial attacks, decoding strategies, and repetition penalty.

mark across a broad spectrum of perturbation scenarios, and we employ TempParaphraser (Huang et al., 2025) to perturb the original RAID texts under high-temperature conditions. The results are presented in Figure 4, with the exception of the high-temperature paraphrasing evaluation, which is detailed in Appendix C.4.

The overall results demonstrate that NTS achieves the highest AUROC in the majority of perturbation scenarios, including various decoding strategies and high-temperature perturbations. This suggests that the Temperature Sensitivity underlying NTS effectively captures the intrinsic differences between LLM-generated and human-written text. Consequently, NTS exhibits significantly greater robustness against evasion-based attacks.

Robustness of Optimal Thresholds In practical applications, zero-shot detectors rely on a decision threshold to categorize text as either human-written or LLM-generated. Consequently, the stability of the optimal classification threshold across diverse domains and varying source models serves as a critical metric for evaluating a detector's robustness. To investigate this, we conducted experiments using the Xsum and Writing datasets, featuring content generated by Claude, Gemini, and GPT-4o (Yu et al., 2025), as shown in Figure 5. Our results demonstrate that NTS maintains a remarkably consistent optimal classification threshold across all domain-model combinations. This stability further validates the reliability and generalizability of NTS.

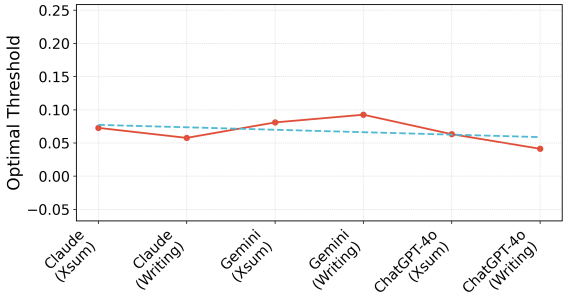

Figure 5: **Optimal classification thresholds of NTS across two domains and three generative models.** Dashed lines denote the fitted curves derived from these six data points.

4.5 Additional Analyses

Time & Space Efficiency As a lightweight zero-shot detector, NTS introduces only a near-zero cost temperature modulation operation. Consequently, the additional overhead in terms of per-text inference time and overall memory is negligible. As illustrated in Table 3, when tested on an NVIDIA A40 GPU, NTS incurs only a 0.01% increase in processing time for each text and a 3.00% increase in GPU memory usage over LogPPL. In contrast, existing methods, such as IRM, require the simultaneous deployment of two surrogate models, leading

Detector	Time (s)	Space (GB)
LogPPL (Bao et al., 2025)	0.105	18.25
IRM (Liu et al., 2025) †	0.210	36.14
NTS (ours)	0.106	18.81
(*Absolute* ↑)	**0.01%**	**3.00%**

Table 3: **Runtime per instance and GPU memory usage.** Falcon-7b is used as the surrogate model for NTS and LogPPL, while Falcon-7b and Falcon-7b-Instruct serve as surrogate models for IRM. "(Absolute ↑)" denotes the additional time/space cost brought by NTS relative to vanilla LogPPL.

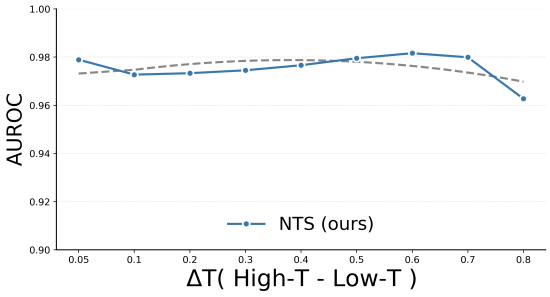

Figure 6: **NTS performance (AUROC) across different ΔT values.** Here, $\Delta T \in \{0.05, 0.1, \ldots, 0.8\}$ represents combinations of (low-T, high-T) ranging from $(1.05, 1.1)$ to $(0.65, 1.45)$, with low-T decreasing and high-T increasing in steps of 0.05 for $\Delta T \geq 0.1$.

to a substantial increase in both memory consumption and computational latency.

Impact of Hyperparameters NTS involves two primary hyperparameters: the low temperature (low-T) and high temperature (high-T). We evaluate the robustness of NTS to the temperature difference (ΔT = high-T − low-T) across a range of values: $\{0.05, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8\}$, corresponding to various combinations of low-T and high-T. This analysis was conducted using the Writing dataset, with Gemini serving as the source model (Yu et al., 2025).

As illustrated in Figure 6, the detection performance of NTS is remarkably robust to hyperparameter selection, maintaining consistent stability across a wide range of temperature differences Δ. Specifically, we observe that NTS achieves optimal performance when the temperature difference is approximately 0.7 (where low-T = 0.7 and high-T = 1.4). Based on these empirical findings, we adopt this specific configuration ($\Delta T = 0.7$) for the majority of our experimental evaluations.

Impact of Different Surrogate Models We individually evaluate various surrogate models

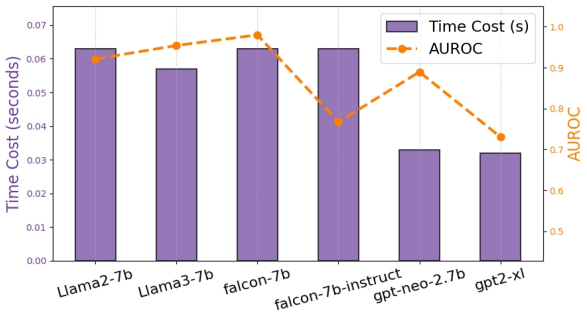

Figure 7: **NTS performance across different surrogate models.** The left y-axis denotes the processing time per text instance for each surrogate model, while the right y-axis represents the corresponding AUROC achieved by NTS.

for NTS—including Llama-2-7b, Llama-3.1-8B, Falcon-7b, Falcon-7b-instruct, GPT-Neo-2.7b, and GPT2-XL (Touvron et al., 2023; Almazrouei et al., 2023; OpenAI et al., 2024)—using the same dataset as in Section Impact of Hyperparameters 4.5.

As Figure 7 illustrates, NTS excels with large base models (consistently yielding AUROC > 0.9), whereas small-parameter models perform poorly because their limited capacity results in inherently high perplexity for both LLM-generated and human-written text. Furthermore, we observe a distinct performance degradation with instruction-tuned models. This occurs because, unlike general pre-training, instruction-tuning is an alignment process that skews the model's probability distribution toward specific stylistic expectations. Since our dataset features highly diverse prompts, they rarely align with the narrow instruction paradigm of the instruction-tuned surrogates. Consequently, the probability advantage typically held by LLM-generated text is flattened, diminishing detection accuracy.

Crucially, NTS requires only a single optimal surrogate. Practitioners can entirely bypass these limitations and ensure state-of-the-art performance simply by deploying NTS with a widely accessible, large-scale base model (such as Falcon-7B).

As a Feature for Training a Classifier Furthermore, we leveraged Temperature Sensitivity (TS) as input features to train a Random Forest classifier. Remarkably, this TS-based supervised detector outperforms five state-of-the-art supervised detection baselines on both the HART and RAID benchmarks, as shown in Table 4. This further validates the discriminative power of temperature sensitivity, proving it to be a highly effective indicator for dis-

Detector	RAID		HART	
	ROC	F1	ROC	F1
Supervised Detectors				
RoBERTa (Hello-SimpleAI, 2023)	0.614	0.67	0.502	0.69
RADAR (Hu et al., 2023)	0.828	0.77	0.687	0.71
IMBD (Chen et al., 2024)	0.804	0.75	0.778	0.70
Ghostbuster (Verma et al., 2024)	0.845	0.76	0.878	0.79
Biscope (Guo et al., 2024)	0.903	0.80	0.907	0.82
RF (ours)	**0.928**	**0.83**	**0.924**	**0.83**

Table 4: Comparison of our TS-based Random Forest classifier with 5 SOTA supervised detectors. The detailed training strategy for our classifier is provided in Appendix D.2.

tinguishing between human and machine-authored text. See Appendix D.2 for implementation details.

Reliability of NTS Metric To quantify the reliability of the detection output, we introduce a confidence metric that evaluates both the correctness and the margin of the prediction relative to the optimal classification threshold (THR).

First, we compute a standardized signed distance, z_i, for each text i. We define a label-aligned signed distance, d_i, between the detection score and THR, orienting the sign such that $d_i > 0$ strictly indicates a correct classification and $d_i < 0$ indicates a misclassification. This distance is then normalized by σ_{dev}, the standard deviation of the raw signed distances calculated on the development set:

$$z_i = \frac{d_i}{\sigma_{\text{dev}}} \tag{9}$$

Next, we map this standardized distance z_i to a confidence level $C_i \in [-1, 1]$ using a scaled sigmoid function:

$$C_i = \frac{2}{1 + e^{-k \cdot z_i}} - 1 \tag{10}$$

where k is a scaling hyperparameter. Under this formulation, $C_i > 0$ signifies a correct prediction on the proper side of the threshold, $C_i < 0$ denotes an incorrect prediction, and $C_i = 0$ represents a score falling exactly on the threshold.

Finally, to evaluate the overall reliability of the NTS metric, we calculate the Mean Confidence across all N samples in the RAID dataset:

$$\text{Mean Confidence} = \frac{1}{N} \sum_{i=1}^{N} C_i \tag{11}$$

A higher Mean Confidence demonstrates a more trustworthy and robust detection method. We compare this metric against established baselines under $k \in \{0.5, 2, 5\}$, where a larger k represents a more relaxed confidence condition. As illustrated in Table 5, NTS achieves the highest confidence scores across all three k settings. This demonstrates that NTS not only delivers the best overall detection performance but is also the most reliable.

Detector	k=0.5	k=2.0	k=5.0	Avg.
Fast-detectGPT (Bao et al., 2024)	0.112	0.340	0.474	0.308
Binoculars (Hans et al., 2024)†	0.117	0.326	0.460	0.301
NTS (ours)	**0.144**	**0.380**	**0.497**	**0.340**

Table 5: **Comparison of confidence metric across different detectors.** A higher value indicates greater reliability.

5 Conclusion

This paper introduces the concept of temperature sensitivity and demonstrates its effectiveness as a powerful feature for distinguishing between LLM-generated and human-written text. Building upon this insight, we propose NTS, a novel zero-shot detector that utilizes normalized temperature sensitivity. We evaluate NTS on a variety of challenging datasets. Thanks to the potent discriminative power of temperature sensitivity, our method achieves superior performance across diverse domains and source models, underscoring the robustness and effectiveness of our proposed approach.

Limitations

This work has two primary limitations. First, NTS optimally pairs with large-parameter surrogate models and exhibits reduced compatibility with instruction-tuned models as surrogate (Section 4.5). Second, its performance degrades under high-temperature perturbations. Consequently, developing a more advanced NTS implementation robust against such perturbations remains for future research.

Acknowledgements

We thank the Area Chair and reviewers for their valuable suggestions that significantly improved this work. This research is supported by the National Natural Science Foundation of China (No. 62376033 and 62232006) and the Fundamental and Interdisciplinary Disciplines Breakthrough Plan of the Ministry of Education of China (No. JYB2025XDXM103).

References

Ebtesam Almazrouei, Hamza Alobeidli, Abdulaziz Alshamsi, Alessandro Cappelli, Ruxandra Cojocaru, Mérouane Debbah, Étienne Goffinet, Daniel Hesslow, Julien Launay, Quentin Malartic, Daniele Mazzotta, Badreddine Noune, Baptiste Pannier, and Guilherme Penedo. 2023. The falcon series of open language models. *Preprint*, arXiv:2311.16867.

Guangsheng Bao, Lihua Rong, Yanbin Zhao, Qiji Zhou, and Yue Zhang. 2025. Decoupling content and expression: Two-dimensional detection of ai-generated text. *Preprint*, arXiv:2503.00258.

Guangsheng Bao, Yanbin Zhao, Zhiyang Teng, Linyi Yang, and Yue Zhang. 2024. Fast-detectgpt: Efficient zero-shot detection of machine-generated text via conditional probability curvature. *Preprint*, arXiv:2310.05130.

Erik Brynjolfsson, Danielle Li, and Lindsey Raymond. 2025. Generative ai at work*. *The Quarterly Journal of Economics*, 140(2):889–942.

Jiaqi Chen, Xiaoye Zhu, Tianyang Liu, Ying Chen, Xinhui Chen, Yiwen Yuan, Chak Tou Leong, Zuchao Li, Tang Long, Lei Zhang, Chenyu Yan, Guanghao Mei, Jie Zhang, and Lefei Zhang. 2024. Imitate before detect: Aligning machine stylistic preference for machine-revised text detection. *Preprint*, arXiv:2412.10432.

Liam Dugan, Alyssa Hwang, Filip Trhlík, Andrew Zhu, Josh Magnus Ludan, Hainiu Xu, Daphne Ippolito, and Chris Callison-Burch. 2024. 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, Bangkok, Thailand. Association for Computational Linguistics.

Damian Okaibedi Eke. 2023. Chatgpt and the rise of generative ai: Threat to academic integrity? *Journal of Responsible Technology*, 13:100060.

Gemini Team Google. 2023. Gemini: A family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.

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

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

Abhimanyu Hans, Avi Schwarzschild, Valeriia Cherepanova, Hamid Kazemi, Aniruddha Saha, Micah Goldblum, Jonas Geiping, and Tom Goldstein. 2024. Spotting llms with binoculars: Zero-shot detection of machine-generated text. *Preprint*, arXiv:2401.12070.

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

Hello-SimpleAI. 2023. chatgpt-detector-roberta (revision d2b342c).

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

Junjie Huang, Ruiquan Zhang, Jinsong Su, and Yidong Chen. 2025. Tempparaphraser:"heating up" text to evade ai-text detection through paraphrasing. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 31542–31561.

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

Eric Mitchell, Yoonho Lee, Alexander Khazatsky, Christopher D. Manning, and Chelsea Finn. 2023. Detectgpt: Zero-shot machine-generated text detection using probability curvature. *Preprint*, arXiv:2301.11305.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, and 262 others. 2024. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Yikang Pan, Liangming Pan, Wenhu Chen, Preslav Nakov, Min-Yen Kan, and William Yang Wang. 2023. On the risk of misinformation pollution with large language models. *Preprint*, arXiv:2305.13661.

Sida Peng, Eirini Kalliamvakou, Peter Cihon, and Mert Demirer. 2023. The impact of ai on developer productivity: Evidence from github copilot. *Preprint*, arXiv:2302.06590.

Jiameng Pu, Zain Sarwar, Sifat Muhammad Abdullah, Abdullah Rehman, Yoonjin Kim, Parantapa Bhattacharya, Mobin Javed, and Bimal Viswanath. 2022. Deepfake text detection: Limitations and opportunities. *Preprint*, arXiv:2210.09421.

Shushanta Pudasaini, Luis Miralles-Pechuán, David Lillis, and Marisa Llorens Salvador. 2024. Survey on plagiarism detection in large language models: The impact of chatgpt and gemini on academic integrity. *Preprint*, arXiv:2407.13105.

Sara Sabour, Nicholas Frosst, and Geoffrey E Hinton. 2017. Dynamic routing between capsules. *Preprint*, arXiv:1710.09829.

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

Jinyan Su, Terry Zhuo, Di Wang, and Preslav Nakov. 2023. DetectLLM: Leveraging log rank information for zero-shot detection of machine-generated text. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 12395–12412, Singapore. Association for Computational Linguistics.

Ruixiang Tang, Yu-Neng Chuang, and Xia Hu. 2024. The science of detecting llm-generated text. *Commun. ACM*, 67(4):50–59.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models. *Preprint*, arXiv:2302.13971.

Vivek Verma, Eve Fleisig, Nicholas Tomlin, and Dan Klein. 2024. 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, Mexico City, Mexico. Association for Computational Linguistics.

Ziqing Yang, Yixin Wu, Yun Shen, Wei Dai, Michael Backes, and Yang Zhang. 2025. The challenge of identifying the origin of black-box large language models. *Preprint*, arXiv:2503.04332.

Xiao Yu, Yi Yu, Dongrui Liu, Kejiang Chen, Weiming Zhang, Nenghai Yu, and Jing Shao. 2025. Evobench: Towards real-world llm-generated text detection benchmarking for evolving large language models. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 14605–14620.

Biru Zhu, Lifan Yuan, Ganqu Cui, Yangyi Chen, Chong Fu, Bingxiang He, Yangdong Deng, Zhiyuan Liu, Maosong Sun, and Ming Gu. 2023. Beat LLMs at their own game: Zero-shot LLM-generated text detection via querying ChatGPT. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 7470–7483, Singapore. Association for Computational Linguistics.

A Derivation

A.1 Derivation of the Derivative

The derivative of the probability with respect to the temperature T for the ground-truth word w_{label} in the distribution is given by:

$$\frac{\partial p(w_{label})}{\partial T} = \frac{\partial}{\partial T}\left(\frac{e^{\ell_{label}/T}}{\sum_{j=1}^{V} e^{\ell_j/T}}\right). \tag{12}$$

Here, V denotes the vocabulary size. The index j iterates over all possible words in the vocabulary, and ℓ_j denotes the output logit for the word w_j.

We first rewrite $p(w_{label})$ as:

$$p(w_{label}) = \frac{e^{\ell_{label}/T}}{Z}, \quad Z = \sum_{j=1}^{V} e^{\ell_j/T}. \tag{13}$$

Taking the derivative, we apply the quotient rule. To fit within the column, we factor out $\frac{1}{Z^2}$:

$$\frac{\partial p(w_{label})}{\partial T} = \frac{1}{Z^2}\Bigg[e^{\ell_{label}/T}\left(-\frac{\ell_{label}}{T^2}\right)Z - e^{\ell_{label}/T}\left(-\frac{1}{T^2}\sum_{j=1}^{V}\ell_j e^{\ell_j/T}\right)\Bigg]. \tag{14}$$

Factoring out $\frac{e^{\ell_{label}/T}}{Z^2T^2}$, we obtain:

$$\frac{\partial p(w_{label})}{\partial T} = \frac{e^{\ell_{label}/T}}{Z^2T^2}\left(-\ell_{label}Z + \sum_{j=1}^{V}\ell_j e^{\ell_j/T}\right). \tag{15}$$

Recognizing that $p(w_j) = \frac{e^{\ell_j/T}}{Z}$ represents the probability of the j-th word, and noting that Z is constant with respect to the summation index j, we can rewrite the second term in parentheses as the expectation of the logits:

$$\sum_{j=1}^{V}\ell_j e^{\ell_j/T} = Z\sum_{j=1}^{V}p(w_j)\ell_j = Z\cdot\hat{\ell}, \tag{16}$$

where $\hat{\ell} = \sum_{j=1}^{V} p(w_j)\ell_j$ denotes the expected logits under the current distribution.

Simplifying, we obtain:

$$\frac{\partial p(w_{label})}{\partial T} = \frac{p(w_{label})}{T^2}\left(\hat{\ell} - \ell_{label}\right). \tag{17}$$

Finally, for the log-probability:

$$\begin{aligned}\frac{\partial \log p(w_{label})}{\partial T} &= \frac{1}{p(w_{label})} \cdot \frac{\partial p(w_{label})}{\partial T} \\ &= \frac{1}{p(w_{label})} \cdot \frac{p(w_{label})}{T^2}\left(\hat{\ell} - \ell_{label}\right) \\ &= \frac{\hat{\ell} - \ell_{label}}{T^2}.\end{aligned} \tag{18}$$

A.2 Derivation of transformation

To extend the single-token TS difference to a text-level Temperature Sensitivity (TS) metric, for the i-th token of a text, the token-level TS is defined as $|\log P_{T_{low}}(x_i) - \log P_{T_{high}}(x_i)| = |\Delta_i|$. A straightforward approach would be to define the text-level TS as the average of these token-level values:

$$\text{TS}' = \frac{1}{N}\sum_{i=1}^{N}|\Delta_i| \tag{19}$$

However, we can enhance the discriminative power of this text-level metric by moving the absolute value operation outside the summation:

$$\text{TS} = \left|\frac{1}{N}\sum_{i=1}^{N}\Delta_i\right| \tag{20}$$

We analyze the effects of this modification across two distinct scenarios:

Case A: LLM-Generated Text
For LLM-generated text, predictions concentrate in the distribution's head ($\ell_{i,label} \gg \hat{\ell}_i$), driving the derivative $\frac{\partial \log P^{(i)}}{\partial T} < 0$. Consequently, decreasing temperature consistently increases log-probabilities ($\Delta_i > 0$). With overwhelmingly positive signs, the triangle inequality effectively becomes an equality:

$$\text{TS} \approx \text{TS}' \tag{21}$$

Case B: Human-written Text
Conversely, human-written text frequently falls in the middle of the distribution, where $\ell_{i,label}$ closely aligns with the expected logit $\hat{\ell}_i$. The derivative $\frac{\partial \log P^{(i)}}{\partial T}$ fluctuates around zero, causing the discrete differences Δ_i to have mixed signs (both positive and negative). When summing these mixed-sign values, positive and negative Δ_i terms cancel each other out:

$$\text{TS} \ll \text{TS}' \tag{22}$$

As a result, the computed TS value for human-written text is lower than TS′, which yields a stronger discriminative capability between human and LLM text. Therefore, **we adopt TS rather than TS′ as our text-level metric in this paper**, which is precisely equivalent to the absolute difference in LogPPL under high versus low temperatures.

B Datasets settings

Tables 6, 7, and 8 summarize the detailed specifications of the RAID, HART, and Evo datasets used in this paper, respectively.

Domain	Language	Dev	Test
News	English	500	500
Books	English	500	500
Wiki	English	500	500
Abstracts	English	500	500
Reddit	English	500	500
Recipes	English	500	500
Poetry	English	500	500
Reviews	English	500	500

Table 6: Domains and languages covered by RAID.

Domain	Language	Length	Dev	Test
Student essay	English	241 words	2K	2K
Arxiv Intro	English	410 words	2K	2K
Creative Wrting	English	345 words	2K	2K
CC News	English	148 words	2K	2K
CC News	Chinese	590 chars	2K	2K
CC News	French	258 words	2K	2K
CC News	Spanish	285 words	2K	2K
CC News	Arabic	152 words	2K	2K

Table 7: Domains and languages covered by HART.

Domain	Language	Test
Xsum	English	1200
Writing	English	1200

Table 8: Domains and languages covered by EVO. We sample a subset of the EvoBench dataset.

C Additional Experiments Results

C.1 F1 results on RAID

The F1 results of zero-shot detectors on RAID are shown in Table 9. The F1-score of NTS consistently outperforms other baselines across all domains with the sole exception of the 'Review' cate-

gory. This underscores the superior practical performance and cross-domain robustness of NTS in diverse real-world scenarios.

C.2 F1 results on HART

The F1 results of zero-shot detectors on RAID are shown in Table 10. NTS maintains its leading F1-scores across both LEVEL-1 and LEVEL-2 scenarios, demonstrating that it is the most effective LLM-generated text detector for environments with stringent AI-usage restrictions.

C.3 *Other Languages*

See Table 11 for detailed results on the multilingual dataset under three level tasks. NTS significantly outperforms existing zero-shot methods, particularly in low-resource languages. This suggests that temperature sensitivity captures an intrinsic distinction between LLM-generated and human-written text that transcends specific linguistic boundaries.

C.4 *Against High-Temp Perturbation Attacks*

See Table 12 for the detailed results of detectors on the RAID benchmark attacked by TempParaphraser (Huang et al., 2025) under the high-temperature settings. As presented, NTS still achieves the highest AUROC of 0.800, demonstrating its superior detection capability. However, we acknowledge a performance degradation across all three detectors compared to their results on the original dataset. We attribute this universal performance drop to the adversarial nature of TempParaphraser. By injecting high-temperature perturbations, it intentionally introduces lower-probability tokens into the LLM-generated text. This directly compromises its inherent high-probability advantage, thereby making it significantly harder to differentiate from human text. Nevertheless, the fact that NTS maintains the highest AUROC under such targeted attacks further underscores its exceptional robustness.

C.5 *Optimal Classification Thresholds*

See Table 13 for detailed optimal thresholds results across two domains and three generative models. The optimal classification thresholds for the normalized NTS remain more consistent than those of the raw TS as calculated by equation 5, exhibiting lower std across six distinct datasets. This stability underscores the practical feasibility and robustness of NTS for real-world deployment, as it minimizes the need for per-scenario recalibration.

C.6 *Impact of Hyperparameters*

See Table 14 for detailed results on the Impact of Hyperparameters. NTS exhibits minimal sensitivity to hyperparameter variations, demonstrating that temperature sensitivity is a plug-and-play and highly versatile approach for diverse detection tasks.

C.7 *Impact of Different Surrogate Models*

As detailed in Table 15, NTS demonstrates superior compatibility with larger surrogate models, whereas its efficacy is comparatively attenuated when applied to instruction-tuned variants. This attenuation occurs because instruction-tuned models diminish the high-probability advantage typically assigned to LLM-generated text that deviates from specific instructional formats. Nevertheless, this does not compromise the practical utility of NTS; practitioners can fully leverage its detection capabilities simply by selecting an existing, open-source, large-scale base model.

C.8 *As a Feature for Training a Classifier*

See Table 16 and Table 17 for detailed results of supervised detectors on the RAID and HART benchmark. Classifiers trained on temperature-sensitivity-derived features achieve state-of-the-art (SOTA) detection performance, providing strong empirical validation for the inherent disparity in temperature sensitivity between LLM-generated and human-written text.

D Implementation Details

D.1 Token Selection Strategy

To illustrate the probability distributions in Figure 1 (a), we isolate token at the 90th percentile of entropy for each text to ensure comparability between human-written and LLM-generated content. We then sort the probability distributions of these selected tokens and calculate the mean value at each rank across a dataset of 600 LLM-generated and 600 human-written texts.

D.2 Training Strategy

To train the classifier, we first partition each text into four equal segments. For each segment, we calculate three token-level features: (i) the log-probability ($\log p$), (ii) the first-order derivative of $\log p$ with respect to temperature T, and (iii)

Detector	News	Books	Wiki	Abstr.	Reddit	Recipes	Poetry	Reviews	AUROC	F1
Zero-Shot Detectors										
Entropy (Mitchell et al., 2023)	0.67	0.66	0.67	0.67	0.67	0.66	0.67	0.67	0.585	0.67
LogPPL (Bao et al., 2025)	0.62	0.72	0.65	0.66	0.72	0.58	0.67	0.70	0.663	0.66
Log-Rank (Bao et al., 2025)	0.63	0.73	0.67	0.67	0.73	0.59	0.67	0.72	0.681	0.67
LRR (Su et al., 2023)	0.65	0.76	0.70	0.73	0.71	0.63	0.72	0.73	0.745	0.70
Fast-DetectGPT (Bao et al., 2024) †	0.73	0.80	**0.76**	0.77	0.74	0.71	**0.78**	0.77	0.799	0.76
Binoculars (Hans et al., 2024) †	0.75	**0.81**	**0.76**	0.77	**0.78**	0.72	**0.78**	**0.78**	0.807	0.77
IRM (Liu et al., 2025) †	0.67	0.67	0.67	0.67	0.67	0.67	0.67	0.67	0.690	0.67
NTS (ours)	**0.77**	**0.81**	**0.76**	**0.82**	**0.78**	**0.78**	**0.78**	0.77	**0.856**	**0.78**

Table 9: Detection performance on the RAID benchmark across different domains. We report F1 for each domain, with the last two columns summarizing ALL AUROC and F1 across all domains, respectively. † denotes methods that require two surrogate models. **Bold** indicates the best result.

Detector	Level-3 Detection Task					Level-2 Detection Task					Level-1 Detection Task				
	ArXiv	Writ.	News	AUROC	F1	ArXiv	Writ.	News	AUROC	F1	ArXiv	Writ.	News	AUROC	F1
Zero-Shot Detectors															
Entropy (Mitchell et al., 2023)	0.69	0.65	0.67	0.656	0.68	0.67	0.67	0.68	0.577	0.67	0.67	0.67	0.69	0.529	0.67
LogPPL (Bao et al., 2025)	0.78	0.73	0.71	0.799	0.75	0.67	0.67	0.68	0.473	0.67	0.67	0.67	0.67	0.576	0.67
Log-Rank (Bao et al., 2025)	0.81	0.73	0.72	0.814	0.77	0.67	0.67	0.67	0.465	0.67	0.67	0.67	0.67	0.572	0.67
LRR (Su et al., 2023)	**0.86**	0.72	0.77	0.839	0.78	0.67	0.67	0.67	0.572	0.67	0.67	0.67	0.67	0.597	0.67
Fast-DetectGPT (Bao et al., 2024) †	0.83	**0.82**	0.80	0.862	0.81	0.70	0.68	0.67	0.711	0.68	0.72	0.69	0.66	0.778	0.72
Binoculars (Hans et al., 2024) †	0.85	**0.82**	**0.82**	0.870	**0.83**	0.70	0.68	0.67	0.711	0.69	0.72	0.70	0.68	0.779	**0.73**
IRM (Liu et al., 2025) †	0.73	0.69	0.72	0.732	0.70	0.73	0.66	**0.72**	0.697	0.69	0.69	0.65	**0.70**	0.692	0.67
NTS (ours)	0.85	0.81	0.79	**0.874**	0.81	**0.76**	**0.73**	0.70	**0.794**	**0.72**	**0.73**	**0.73**	0.66	**0.801**	**0.73**

Table 10: Detection performance on the HART benchmark under three levels. We report F1 for different domains in each level, with the last two columns in each level summarizing ALL AUROC and F1 across all domains, respectively. † denotes methods that require two surrogate models. **Bold** indicates the best result.

Detector	Level-3 Detection Task					Level-2 Detection Task					Level-1 Detection Task				
	English	French	Spanish	Chinese	Arabic	English	French	Spanish	Chinese	Arabic	English	French	Spanish	Chinese	Arabic
Zero-Shot Detectors															
Fast-DetectGPT (Bao et al., 2024) †	0.850	0.865	0.830	0.867	0.590	0.688	0.773	0.695	0.836	0.466	0.711	**0.750**	0.732	0.835	0.432
Binoculars (Hans et al., 2024) †	**0.866**	0.881	0.846	0.869	0.571	0.697	**0.777**	0.708	0.838	0.450	0.716	0.747	0.740	**0.838**	0.424
IRM (Liu et al., 2025) †	0.718	0.605	0.577	0.621	0.378	0.750	0.638	0.547	0.579	0.436	0.724	0.595	0.540	0.576	0.391
NTS (ours)	0.853	**0.890**	**0.859**	**0.878**	**0.651**	**0.780**	0.776	**0.738**	**0.872**	**0.563**	**0.726**	0.741	**0.755**	0.832	**0.547**

Table 11: Detection performance on the multilingual dataset under three levels. We report AUROC for different languages in each level. † denotes methods that require two surrogate models. **Bold** indicates the best result.

Detector	News	Books	Wiki	Abstr.	Reddit	Recipes	Poetry	Reviews	AUROC
Zero-Shot Detectors									
Fast-DetectGPT (Bao et al., 2024) †	0.818	0.753	0.729	0.779	**0.819**	0.685	0.777	**0.821**	0.772
Binoculars (Hans et al., 2024) †	0.791	0.758	0.700	0.737	0.795	0.675	**0.833**	0.796	0.762
NTS (ours)	**0.820**	**0.808**	**0.810**	**0.819**	**0.819**	**0.837**	0.679	0.817	**0.800**

Table 12: Detection performance on the RAID benchmark attacked by TempParaphraser (Huang et al., 2025). We report AUROC for each domain, with the last column summarizing ALL AUROC across all domains. † denotes methods that require two surrogate models. **Bold** indicates the best result.

Detector	cluade-xsum	cluade-writ	gemini-xsum	gemini-writ	gpt4o-xsum	gpt4o-writ	STD
TS (raw)	0.170	0.140	0.176	0.225	0.153	0.152	0.030
NTS	0.073	0.057	0.081	0.092	0.063	0.041	**0.018**

Table 13: Optimal classification thresholds across two domains and three generative models. The standard deviation (STD) of the optimal threshold for NTS is lower than that of the raw TS.

(T_{low}, T_{high})	(1.05,1.10)	(1.00,1.1)	(0.95,1.15)	(0.90,1.2)	(0.85,1.25)	(0.80,1.30)	(0.75,1.35)	(0.70,1.40)	(0.65,1.45)
NTS (ours)	0.979	0.973	0.973	0.975	0.977	0.980	0.982	0.980	0.963

Table 14: AUROC of NTS under different combinations of (low-T, high-T).

(Surroage Model)	Llama2-7b	Llama3.1-8b	Falcon-7b	Falcon-7b-ins	GPT-Neo-2.7b	GPT2-xl
NTS (ours)	0.921	0.954	0.980	0.767	0.890	0.731

Table 15: AUROC of NTS under different surrogate models.

Detector	News	Books	Wiki	Abstr.	Reddit	Recipes	Poetry	Reviews	**ALL**	**F1**
Supervised Detectors										
Roberta (Mitchell et al., 2023)	0.591	0.621	0.584	0.643	0.674	0.501	0.640	0.709	0.614	0.67
RADAR (Bao et al., 2025)	0.885	0.911	0.842	0.842	0.870	0.818	0.780	0.783	0.828	0.78
IMBD (Bao et al., 2025)	0.783	0.875	0.803	0.806	0.826	0.699	0.823	0.841	0.804	0.75
Ghostbuster (Su et al., 2023)	0.853	0.896	0.833	0.859	0.864	0.849	0.748	0.874	0.845	0.75
Biscope (Bao et al., 2024)	0.884	**0.937**	0.904	0.910	**0.929**	0.844	0.905	0.905	0.903	0.80
RF (ours)	**0.918**	0.935	**0.911**	**0.938**	0.912	**0.945**	**0.939**	**0.926**	**0.928**	**0.83**

Table 16: Detection performance of supervised detectors on the RAID benchmark across different domains. We report AUROC for each domain, with the last two columns summarizing ALL AUROC and F1 across all domains, respectively. **Bold** indicates the best result.

Detector	**Level-2 Detection Task**				
	ArXiv	Writ.	News	**ALL**	**F1**
Supervised Detectors					
Roberta (Mitchell et al., 2023)	0.686	0.498	0.473	0.502	0.69
RADAR (Bao et al., 2025)	0.814	0.629	0.818	0.687	0.71
IMBD (Bao et al., 2025)	0.786	0.762	0.787	0.778	0.70
Ghostbuster (Su et al., 2023)	0.933	0.857	0.870	0.878	0.79
Biscope (Bao et al., 2024)	0.935	0.909	0.889	0.907	0.82
RF (ours)	**0.958**	**0.935**	**0.907**	**0.924**	**0.83**

Table 17: Detection performance of supervised detectors on the HART benchmark under level-2. We report AUROC for different domains, with the last two columns summarizing ALL AUROC and F1 across all domains, respectively. **Bold** indicates the best result.

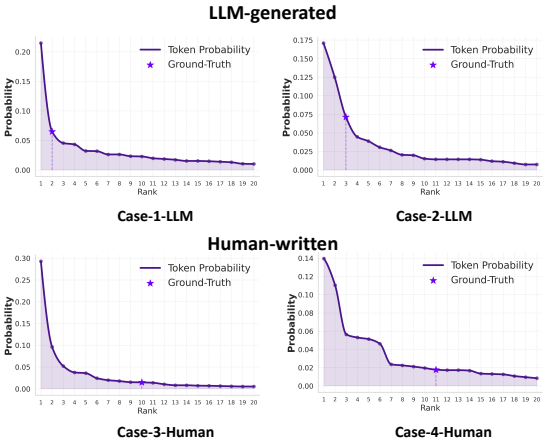

Figure 8: **Case Study:** Probability distributions of two LLM-generated tokens and two human-written tokens.

the curvature (defined as the absolute value of the second-order derivative) of $\log p$ with respect to T. These metrics are evaluated across five temperature settings ($T \in \{0.5, 0.75, 1.0, 1.25, 1.5\}$).

Within each segment, we aggregate these token-level values using seven statistical measures: mean, standard deviation, minimum, maximum, median, and the first and third quartiles. This process results in a 420-dimensional feature vector (3 metrics $\times$ 5 temperatures $\times$ 7 statistics $\times$ 4 segments). Finally, we employ this vector to train a Random Forest classifier, allowing the model to automatically capture the relationships inherent in temperature sensitivity.

E Case Study

We select two representative LLM-generated tokens and two human-written tokens to illustrate their probability distributions. As shown in Figure 8, the distributions of these two types of text within the surrogate model exhibit distinct patterns. For LLM-generated tokens, the ground-truth words reside in the 'head' of the distribution with higher probabilities and top-tier rankings, making them highly sensitive to temperature variations. In contrast, the ground-truth words of human-written tokens often fall into the middle section of the distribution. These tokens manifest lower probabilities

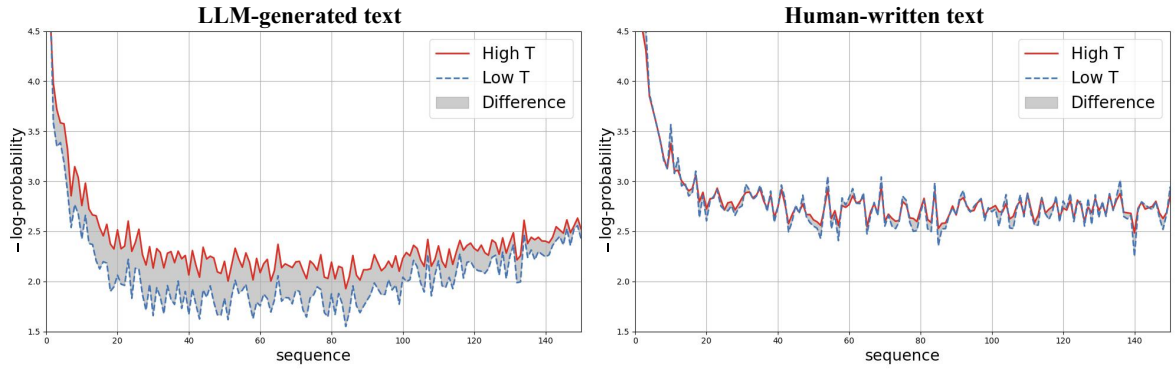

Figure 9: **The log-probability differences at each token position in the sequence under high and low decoding temperatures of surrogate model**, averaged over 500 mixed-source LLM-generated samples and 500 human-written samples from RAID benchmark (Dugan et al., 2024). The surrogate model shows significant log-probability differences between high and low temperatures on LLM-generated samples (indicated by the larger shaded area), whereas the differences on human-written samples are minimal (with the shaded area nearly zero). For visualization purposes, we truncate the maximum sequence length to 150 tokens.

and relatively lower ranks, resulting in a diminished sensitivity to temperature shifts.

F Supplementary Corroborative Figures

F.1 Temperature Sensitivity Disparity

We calculate the log-probabilities of each token under both high and low temperature settings to verify temperature sensitivity at the token level. As illustrated in Figure 9, LLM-generated tokens exhibit significantly higher sensitivity to temperature variations, with the log-probabilities of the tokens (plotted on the x-axis) increasing at lower temperatures and decreasing at higher temperatures. In contrast, human-written tokens remain relatively stable, maintaining consistent log-probabilities across varying temperatures.

F.2 Discrepancy in derivative

The histogram of the derivative of log-probability with respect to temperature at $T = 1$ is illustrated in Figure 10. For LLM-generated text, the derivatives are primarily concentrated in the highly negative region (< 0). This corroborates that the logits of the ground-truth words, l_{label}, in LLM-generated text are significantly larger than the expected logits of the overall distribution, $\hat{l}$. Conversely, for human-written text, the distribution clusters around zero, indicating that the logits of ground-truth words (l_{label}) align closely with the expected logits ($\hat{l}$). This disparity indicates that

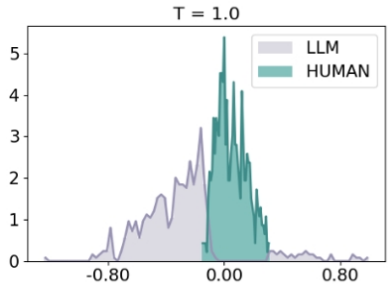

Figure 10: **Histogram of the derivative of log-probability with respect to temperature ($\frac{\partial \log p}{\partial T}$) at $T = 1$.** For each text, the value is calculated as the mean derivative across all ground-truth tokens in text. The histogram is plotted for 500 mixed-source LLM-generated samples and 500 human-written samples from the RAID benchmark (Dugan et al., 2024).

human-written text exhibits significantly lower temperature sensitivity compared to LLM-generated content.