HLPD: Aligning LLMs to Human Language Preference for Machine-Revised Text Detection

Fangqi Dai[1, 2*], **Xingjian Jiang**[1, 3*], **Zizhuang Deng**[1, 4 †]
[1] School of Cyber Science and Technology, Shandong University, China
[2] Xi'an Jiaotong-Liverpool University
[3] Chalmers University of Technology
[4] State Key Laboratory of Cryptography and Digital Economy Security, Shandong University, China
Daifq7@outlook.com, jiangxingjian@mail.sdu.edu.cn, dengzz@sdu.edu.cn

Abstract

To prevent misinformation and social issues arising from trustworthy-looking content generated by LLMs, it is crucial to develop efficient and reliable methods for identifying the source of texts. Previous approaches have demonstrated exceptional performance in detecting texts fully generated by LLMs. However, these methods struggle when confronting more advanced LLM output or text with adversarial multi-task machine revision, especially in the black-box setting, where the generating model is unknown. To address this challenge, grounded in the hypothesis that human writing possesses distinctive stylistic patterns, we propose *Human Language Preference Detection* (HLPD). HLPD employs a reward-based alignment process, *Human Language Preference Optimization* (HLPO), to shift the scoring model's token distribution toward human-like writing, making the model more sensitive to human writing, therefore enhancing the identification of machine-revised text. We test HLPD in an adversarial multi-task evaluation framework that leverages a five-dimensional prompt generator and multiple advanced LLMs to create diverse revision scenarios. When detecting texts revised by GPT-series models, HLPD achieves a 15.11% relative improvement in AUROC over ImBD, surpassing Fast-DetectGPT by 45.56%. When evaluated on texts generated by advanced LLMs, HLPD achieves the highest average AUROC, exceeding ImBD by 5.53% and Fast-DetectGPT by 34.14%.

1 Introduction

As Large Language Models (LLMs) such as GPT-3.5 (OpenAI 2022) and GPT-4o (Achiam et al. 2023) continue to advance in generating convincing texts across diverse fields (M Alshater 2022; de Lima-Santos and Ceron 2021; Fang et al. 2023), public concern regarding their potential misuse has grown (Kelley et al. 2021). These models can produce persuasive text that is coherent and contextually appropriate but may be incorrect or misleading in practice. The cogent responses generated by LLMs are increasingly indistinguishable from human-written content, which poses significant risks (McKenna et al. 2023), such as the spread of disinformation (Bian et al. 2024) and challenges to social equity (Ferrara 2023). Moreover, certain high-stakes domains such as legal, medical, need rigorous human authorship where every word is deliberately chosen and fully understood to avoid subtle errors or vulnerabilities. AI revisions, even if minor, can introduce subtle errors or vulnerabilities that undermine the integrity of the text.

To mitigate these risks and maintain trust, it is essential to develop efficient and accurate methods for detecting text that LLMs might produce or modify. Existing detection techniques, such as DetectGPT (Mitchell et al. 2023) and Fast-DetectGPT (Bao et al. 2023) have performed exceptionally well in identifying text entirely generated by LLMs in a white-box setting (Gehrmann, Strobelt, and Rush 2019), where the detector evaluates the log probability of texts, relying on the observation that machine-generated content generally exhibits higher log-likelihoods compared to human-written texts. However, such methods struggle when confronted with texts generated by advanced LLMs or machine-revised texts where content initially composed by humans is subsequently polished, rewritten, or expanded by LLMs, particularly under black-box settings where the generating model is unknown.

Recent approaches, like ImBD (Chen et al. 2024), have improved detection under black-box settings on machine-revised texts by using Style Preference Optimization (SPO) to align the scoring model with machine-generated styles. Nevertheless, as each LLM may have its own unique stylistic characteristics (Reinhart et al. 2025), training a scoring model on machine-revised texts from a specific LLM could limit generalization to other models, especially those more advanced ones. Moreover, as illustrated in Figure 1, ImBD faces challenges in a more general situation, where texts are revised by state-of-the-art LLMs under diverse, adversarial (Tufts, Zhao, and Li 2024) prompts.

To overcome these limitations, we first observe that while various LLMs exhibit diverse and unstable linguistic patterns, human writing itself also possesses distinctive stylistic characteristics. (AlAfnan and MohdZuki 2023; Guo et al. 2023; Pu and Demberg 2023), such as the frequent use of modal and epistemic constructions (Herbold et al. 2023). Leveraging these stylistic differences, we can align the scoring model, adjusting the token distribution of the model to the human language preference rather than machine style, thereby making the scoring model capture human-writing features more

*These authors contributed equally.
†Corresponding author.

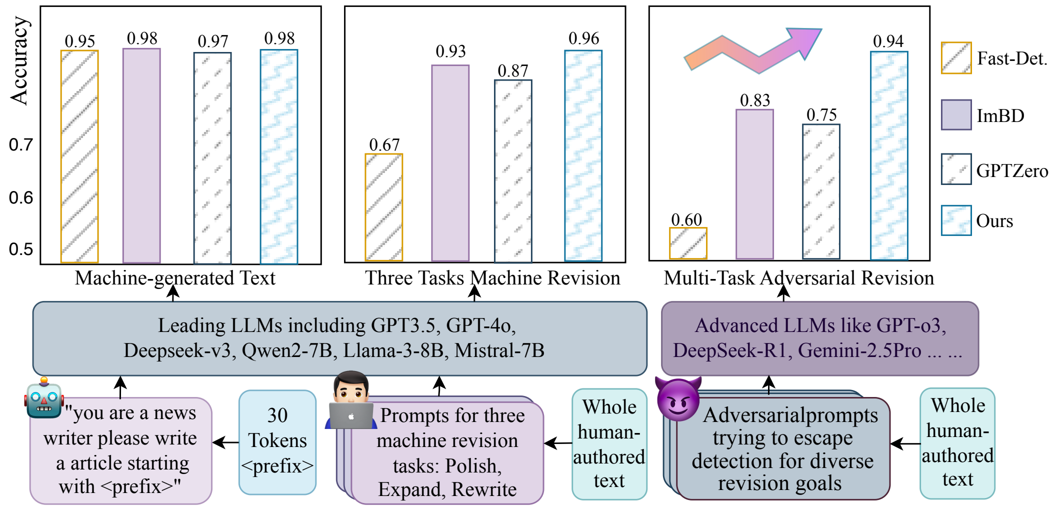

Figure 1: Comparison of detecting methods across different scenarios. Detection accuracy of Fast-DetectGPT, ImBD, GPTZero, and HLPD across machine-generated text (Generate), three tasks revisions (Polish, Expand, Rewrite), and adversarial multi-task revisions. All methods excel when machine-style cues are strong (Generate) but suffer a sharp drop on revising tasks where machine characteristics are attenuated. ImBD amplifies these cues to outperform Fast-DetectGPT, yet still degrades on more general situations, whereas HLPD sustains high accuracy across every scenario.

efficiently and improving detection accuracy. Existing techniques such as RRHF (Yuan et al. 2024) and ORPO (Hong, Lee, and Thorne 2024) have explored the alignment with human preferences, but their primary goal is to enhance the quality of machine-generated text. However, in our work, we adapt the alignment strategy specifically to the scoring task.

Following this principle, we propose a new detection method, *Human language Preference Detection (HLPD)*. First, we train the scoring model on paired samples of original human-written text and its corresponding machine-revised version through a reward-based alignment process called *Human Language Preference Optimization (HLPO)*, guiding the model to prioritize features of human-like language and to be more sensitive to human writing. Second, after training, we use the resulting scoring model to compute the metric of *Human Language Preference Conditional Probability Curvature (HLP-CPC)*, capturing the log-probability difference between a candidate text and its perturbed versions, thus facilitating more accurate detection.

Experimental results demonstrate that HLPD achieves robust performance not only on multi-task adversarial machine revisions, but also excels in detecting texts fully generated by advanced, state-of-the-art LLMs under black-box conditions. In single task revision detection on the GPT series, HLPD achieves a 15.11% relative improvement in AUROC over ImBD, surpasses the logit-based method Fast-DetectGPT by 45.56% and outperforms the supervised method RoBERTa-large by 38.40%. Meanwhile, when detecting adversarial multi-task Revisions, HLPD attains the highest average AUROC, outperforming ImBD by 13.52% and GPTZero by 24.54%. Notably, HLPD also delivers a 5.53% improvement over ImBD on texts generated by advanced LLMs, showing its effectiveness.

Contribution: ❶ We propose a novel strategy, HLPD, aligning the scoring model to human language style, significantly enhancing detection accuracy across fully machine-generated and multi-task revised texts under black-box settings. Furthermore, we apply the trained scoring model in an adaptive attack on GPTZero to highlight its broader potential. ❷ We design and implement a comprehensive adversarial multi-task evaluation framework that better simulates real user scenarios. Employing a five-dimensional adversarial prompt generator together with multiple state-of-the-art LLMs to systematically construct diverse machine-revision scenarios for assessing the effectiveness and robustness of the detection method, i.e., HLPD. ❸ HLPD achieves significant accuracy improvements across various machine revision tasks and multiple languages in the black-box setting. Showcasing its robustness and adaptability to various scenarios and different advanced models.

2 Methodology

We present a novel method, *Human Language Preference Detection (HLPD)*, for detecting both machine-revised and fully generated texts by aligning the scoring model to human writing styles. This alignment improves robustness in the black-box setting, where the specific LLM is unknown, and improves sensitivity to outputs from state-of-the-art LLMs in diverse revision and generation scenarios. For more details of those scenarios, please refer to Section 3.1.

2.1 Foundation

Style alignment involves fine-tuning a language model to prefer specific stylistic characteristics. In the context of text detection, aligning a scoring model to a particular style, whether machine or human, enables the model to become more sensitive to deviations from that style. This sensitivity widens the gap of log-probabilities difference between the target text x and its perturbed versions, $\tilde{x}$. This gap is crucial for accurately identifying texts altered or generated by LLMs.

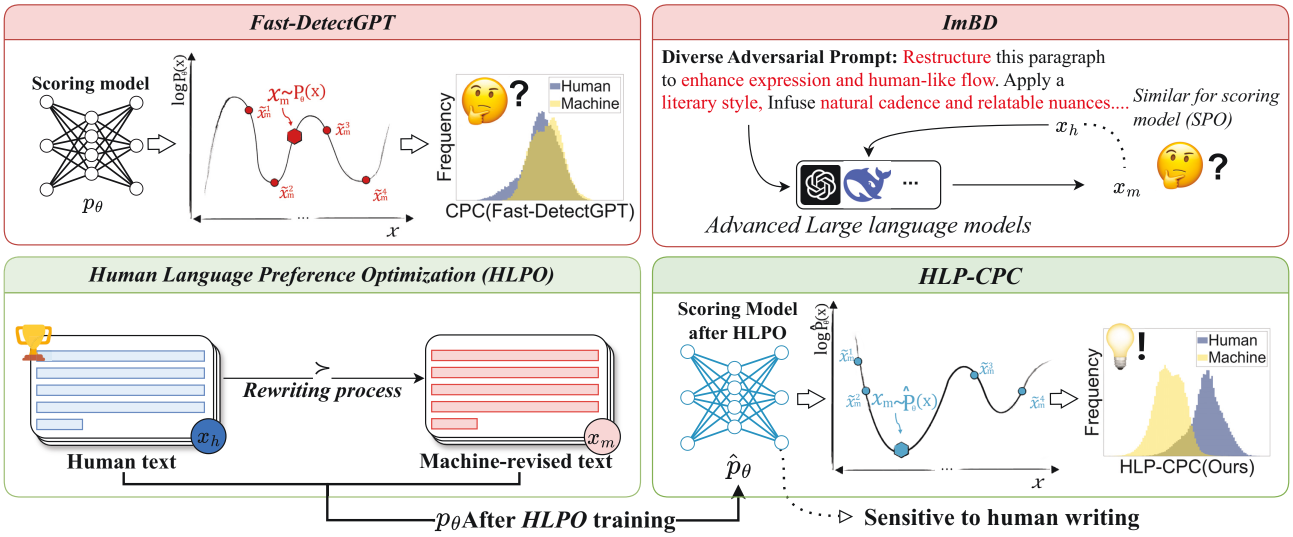

Figure 2: Challenges of Fast-DetectGPT and ImBD under adversarial revision tasks and overview of HLPD. *Top row:* On the left, logit-based detectors such as Fast-DetectGPT struggle when texts x_h are only revised, due to weak machine signals. On the right, in ImBD, scoring model after SPO still favors machine-like signals. This preference can limit generalization, especially when faced with revisions from advanced models under diverse prompts. *Bottom row:* HLPO forms paired human-written and machine-revised texts to train model p_θ, aligning it directly to human writing style. With the trained $\hat{p}_\theta$, HLPD can reliably detect subtle deviations across various scenarios, including minimal revisions and adversarially generated texts by state-of-the-art LLMs, thereby significantly improving robustness in black-box settings.

2.2 Human Language Preference Optimization

As illustrated in Figure 2, the essence of this method lies in the pre-training part. Before using the scoring model in calculating Conditional Probability Curvatures (CPC), we need to create preference relations between pairs of texts with identical content: one written by a human (x_h) and one revised by a machine (x_m). To align the scoring model with human stylistic preferences, we introduce an HLPO process. This process involves training the scoring model to assign higher log probabilities to human-written texts compared to their machine-revised counterparts.

Given a dataset $\mathcal{D} = \{(x_h^{(i)}, x_m^{(i)})\}_{i=1}^N$, where each pair consists of a human-written text $x_h^{(i)}$ and its machine-revised version $x_m^{(i)}$, our goal is to train the scoring model to prefer human style. Specifically, we establish preference relations where the human-written text is preferred over the machine-revised text:

$$x_h^{(i)} \succ x_m^{(i)}, \quad \forall i \in \{1, \dots, N\}. \tag{1}$$

We adopt a preference-based optimization framework inspired by the Direct Preference Optimization (DPO) approach (Rafailov et al. 2024). The probability of preferring one text over another is modeled using the Bradley-Terry model (Bradley and Terry 1952):

$$P(x_h \succ x_m) = \sigma\left(r(x_h) - r(x_m)\right), \tag{2}$$

where $r(x)$ is a reward function representing the model's preference for text x, and σ is the sigmoid function. To make the scoring model prefer the human language style, we need to increase the value of $r(x_h) - r(x_m)$.

Following DPO, we relate the reward function to the log probabilities assigned by the scoring model p_θ we are training and a reference model $p_{\theta_{\text{ref}}}$, which can simply be the original scoring model before training:

$$r(x) = \beta\left(\log p_\theta(x) - \log p_{\theta_{\text{ref}}}(x)\right), \tag{3}$$

where β is a temperature parameter controlling the sharpness of the preference. At this point, The optimization $\max_\theta \mathbb{E}_{(x_h,x_m)\sim\mathcal{D}}\left[\log \sigma\left(r(x_h) - r(x_m)\right)\right]$ becomes:

$$\begin{aligned}\max_\theta\ \mathbb{E}_{(x_h,x_m)\sim\mathcal{D}}\Big[\log \sigma\Big(&\beta(\log p_\theta(x_h) - \log p_\theta(x_m)) \\ &- \beta(\log p_{\theta_{\text{ref}}}(x_h) - \log p_{\theta_{\text{ref}}}(x_m))\Big)\Big].\end{aligned} \tag{4}$$

Since $p_{\theta_{\text{ref}}}$ is fixed during training, we can simplify the optimization to focus on maximizing the difference $\log p_\theta(x_h^{(i)}) - \log p_\theta(x_m^{(i)})$, thus adjusting the scoring model p_θ to assign higher probabilities to human-written texts. And to acquire $\max_\theta \mathbb{E}_{(x_h,x_m)\sim\mathcal{D}}\left[\log \sigma\left(r(x_h) - r(x_m)\right)\right]$, equal to acquire $\min_\theta -\mathbb{E}_{(x_h,x_m)\sim\mathcal{D}}\left[r(x_h) - r(x_m)\right]$. With limiting training data, to directly enlarge the gap of $\log p_\theta(x_h) - \log p_\theta(x_m)$, address potential problems like saturation and avoid overfitting in small-sample preference tuning, based on our experiment, we remove the sigmoid function and introduce an adaptive coefficient β_t, yielding the linear contrastive loss:

$$\min_\theta\ \mathcal{L}(\theta) = -\,\mathbb{E}_{(x_h,x_m)\sim\mathcal{D}}\left[\beta_t \cdot r(x_h, x_m)\right]. \tag{5}$$

where

$$\begin{aligned}r(x_h, x_m) = &\left[\log p_\theta(x_h) - \log p_\theta(x_m)\right] \\ &- \left[\log p_{\theta_{\text{ref}}}(x_h) - \log p_{\theta_{\text{ref}}}(x_m)\right].\end{aligned} \tag{6}$$

Here, β_t is adjusted on-the-fly via a lightweight variance-aware scheduler (Dynamic-β):

$$\beta_t \;=\; \text{Dynamic-}\beta\left(\mathrm{Var}_{\text{window}}[r(x_h, x_m)]\right) \quad (7)$$

At each step, we compute the variance of the margin $r(x_h, x_m)$ over a sliding window, increasing β when the variance is low (model confident) and decreasing it when the variance is high (noisy signal). This stabilizes early training, prevents overfitting, and sharpens the margin later, without extra theoretical assumptions. This supervised fine-tuning process ensures that the scoring model $\hat{p}_\theta$ becomes attuned to the stylistic nuances of human writing, thereby enhancing its ability to detect deviations introduced by machine revisions. We present more detailed comparisons of different loss functions in ablation studies in Section 3.4.

2.3 Detection via HLP-Conditional Probability Curvature

As shown in Figure 2, after aligning the scoring model with human writing styles, we employ a modified probability curvature metric to detect machine-revised text. This metric, termed *Human Language Preference Conditional Probability Curvature (HLP-CPC)*, builds upon the probability curvature introduced in Fast-DetectGPT and adapts it to our human style-aligned scoring model.

In Fast-DetectGPT, the key observation is that machine-generated texts often occupy regions of negative curvature in the log probability landscape, as they tend to have higher log probabilities under the language model compared to human-written texts. However, since our scoring model $\hat{p}_\theta$ is aligned to prefer human writing styles, the situation is reversed: human-written texts receive higher log probabilities, and machine-revised texts receive lower log probabilities.

To quantify this, given a passage x, the aligned scoring model $\hat{p}_\theta$, and the perturbation model q_ϕ, we define the conditional probability function as:

$$\hat{p}_\theta(\tilde{x}|x) = \prod_{j=1}^{n} \hat{p}_\theta(\tilde{x}_j|x_{<j}), \quad (8)$$

where $\tilde{x}_j$ is the j-th token in the perturbed passage $\tilde{x}$, generated by sampling from the conditional distribution $\hat{p}_\theta(\tilde{x}_j|x_{<j})$, and $x_{<j}$ denotes the sequence of tokens preceding the j-th token in passage x without conditioning on other sampled tokens.

We estimate the curvature at the point x by comparing the value of $\hat{p}_\theta(x|x)$ with the values of perturbed texts $\hat{p}_\theta(\tilde{x}|x)$. If $\hat{p}_\theta(x|x)$ has a higher or equal value compared with the average of $\hat{p}_\theta(\tilde{x}|x)$, the function has a positive curvature at x, indicating that x is more likely human-written. Conversely, if $\hat{p}_\theta(x|x)$ is lower, the curvature is negative, suggesting that x is more likely machine-revised. Formally, following Fast-DetectGPT, we quantify the HLP-CPC as:

$$d(x, \hat{p}_\theta, q_\phi) = \frac{\log \hat{p}_\theta(x|x) - \mu}{\sigma}, \quad (9)$$

where

$$\mu = \mathbb{E}_{\tilde{x}\sim q_\phi(\tilde{x}|x)}\left[\log \hat{p}_\theta(\tilde{x}|x)\right], \quad (10)$$

$$\sigma^2 = \mathbb{E}_{\tilde{x}\sim q_\phi(\tilde{x}|x)}\left[\left(\log \hat{p}_\theta(\tilde{x}|x) - \mu\right)^2\right], \quad (11)$$

We establish a detection criterion based on the HLP-CPC metric and a pre-defined threshold ϵ:

$$f(x) = \begin{cases} 1, & \text{if } -d(x, \hat{p}_\theta, q_\phi) > \epsilon, \\ 0, & \text{otherwise.} \end{cases} \quad (12)$$

Here, $f(x) = 1$ indicates that the text is machine-revised, and $f(x) = 0$ indicates that the text is human-written. Note that we use the negative sign before $d(x, \hat{p}_\theta, q_\phi)$ because our scoring model tends to assign higher scores to human-like text and lower scores to machine-revised text, making $d(x)$ negative. By multiplying by -1, we can more directly apply the threshold ϵ to detect revised text.

3 Experiments

To evaluate the efficacy and the potential application of our proposed HLPD method, we conduct experiments and comprehensive analysis of the results to answer the following research questions:

RQ1. How effective and efficient of HLPD compare to other state-of-the-art (SOTA) methods?
RQ2. How robust of HLPD in detecting multi-task adversarial machine-revised text?
RQ3. How effective of our strategy HLPO compare to other baselines?
RQ4. How are the performance and generalizability of our designed Loss function?

3.1 Experiment Settings

Constructing the Single-Task Dataset. Following ImBD's methodology for revision tasks and generation task, datasets were constructed based on three distinct tasks: rewriting, expand, polish and generation. The process involved a two-stage pipeline: (1) Revision instructions generation (2) Paragraph revision under generated instructions.

Constructing the Adversarial Multi-Task Dataset. To evaluate model robustness under realistic conditions, we constructed a challenging adversarial dataset inspired by Zhao et al. (2024). We developed a five-dimensional prompt generator that creates complex instructions by combining: (1) a core revision goal, (2) a target style, (3) instructions to add human-like qualities for detection evasion, (4) operational constraints, and (5) auxiliary user requests. Our generation pipeline consists of two stages: first, these five components are used to prompt DeepSeek-R1, which generates a consolidated instruction. Second, this instruction, along with a human-written text, is fed to a target LLM to produce the final adversarial sample. This methodology expanded our prompt pool to 750 unique prompts.

Source Models. To ensure a comprehensive evaluation that reflects realistic scenarios, we extended the benchmark's original models (GPT-3.5-Turbo, Qwen2-7B (Yang, Chen et al. 2024), Llama-3-8B (Touvron et al. 2023), Mixtral-7B (Jiang, Wang et al. 2024), Deepseek-7B (DeepSeek-AI et al. 2024),) with more recent, advanced LLMs like GPT-o3, GPT-4o, Deepseek-R1 (Guo et al. 2025), Gemini-2.5Pro (Deepmind 2024), Grok-3 (x.ai 2024), Claude-3.5 (Anthropic 2025).

Training datasets. To ensure a fair comparison, we constructed our training dataset using the same human-written source texts as the ImBD training set. Each text was then revised by GPT-3.5-Turbo, which randomly performed one of four revision tasks: expansion, polishing, rewriting, or generation, as explained above. The final dataset consists of pairs, each containing an original human-written text and its corresponding machine-revised version.

Testing datasets. We adopt the experimental setup from prior work (Bao et al. 2023; Howard and Ruder 2018) and source our human-written text from five diverse datasets: *XSum* (Narayan, Cohen, and Lapata 2018) (news articles), *SQuAD* (Rajpurkar et al. 2016) (question answering), *WritingPrompts* ('Writing') (Fan, Lewis, and Dauphin 2018) (creative stories), *PubMedQA* (Jin et al. 2019) (biomedical QA), and *WikiText* ('Wiki') (Merity et al. 2016) (encyclopedic text).

Baselines. We compare our method with two categories of approaches: five training-based models (ReMoDetect (Lee, Tack, and Shin 2024), Ghostbuster (Verma et al. 2024), RoBERTa-base, RoBERTa-large (Liu 2019) and the commercial detector GPTZero (Tian and Cui 2023) and nine logit-based models (ImBD (Chen et al. 2024), Likelihood (Ippolito et al. 2019), LogRank (Solaiman et al. 2019), Entropy (Gehrmann, Strobelt, and Rush 2019), LRR (Su et al. 2023), NPR (Su et al. 2023), DNA-GPT (Yang et al. 2023), DetectGPT and Fast-DetectGPT).

To ensure the robustness of our findings, all key experiments were repeated five times with different random seeds. Figure 3 presents the ROC curves for detection performance. The HLPD† refers to a baseline version implemented without any optimizations. Further implementation and training details are provided in Appendix B.

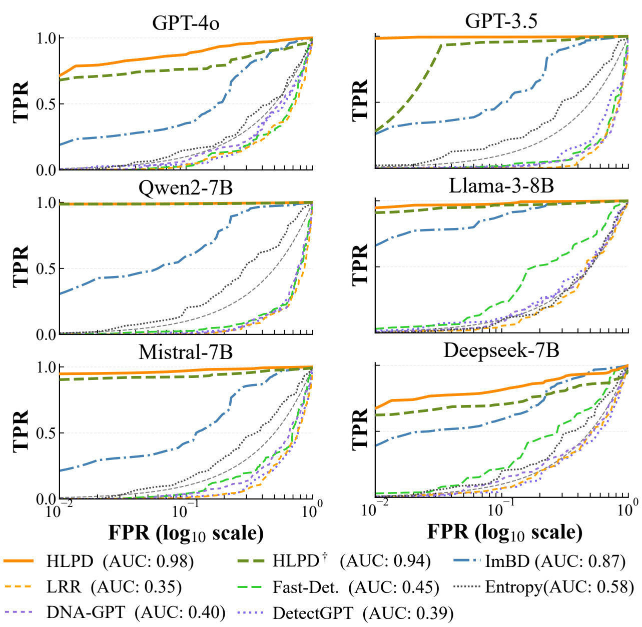

Figure 3: ROC Curve for Detection on XSum dataset.

3.2 Effectiveness & Efficiency (RQ1)

Detection Performance for Single-task Revision. As shown in Table 1, in ImBD's rewrite task, our method outperforms ImBD by 15.11% relative improvement in detecting GPT-series outputs. Compared to the ReMoDetect and Ghostbuster, our method achieves average improvements of 9.27% and 3.21%, respectively. Additionally, our method achieves a 10.57% relative improvement over GPTZero. This indicates that our approach is highly training-efficient, attaining superior performance with a limited amount of data compared to models trained on significantly large datasets. See Appendix E Table 1 for the performance on **polish** task. Furthermore, our method gets the highest score on four *open-source* models. See Appendix E Table 2, 5 and 6 for detailed results.

Detecting Advanced LLM Generations. As shown in Table 2, HLPD demonstrates superior performance on text from advanced LLMs, outperforming ImBD by an average of 5.53%. This suggests that HLPD's strategy of aligning with stable human language patterns, rather than shifting machine-generated ones, provides greater resilience to the architectural and stylistic variations inherent in newer models.

Multilingual Detection Results. To evaluate the performance of HLPD in multilingual text scenarios, we generated Chinese, Portuguese, and Spanish dataset variants. As shown in Appendix E Figure 5, HLPD exhibits superior detection capabilities: achieving a 10% relative advantage over ImBD and gaining an average improvement of 33% against RoBERTa-large across all language versions.

Inference Efficiency. As shown in Table 1, HLPD achieves an average inference time of 0.72 seconds per 1,000 words, matching the efficiency of Fast-DetectGPT while delivering superior detection performance on an L20 GPU card.

3.3 Robustness Studies(RQ2)

Robustness in Adversarial Multi-Task Revisions. As detailed in Table 3, the efficacy of HLPD in detecting adversarial multi-task revised texts from four prominent LLMs was evaluated against leading baselines. HLPD achieved an overall average AUROC of 0.9463, surpassing ImBD by 13.52% and outperforming ReMoDetect by 6.69%. This indicated that HLPD's robustness and enhanced ability to identify multi-task machine-revised text even when it employs sophisticated adversarial prompts.

Robustness in Diverse Revision Goals. HLPD's robustness against diverse revision goals is evident in Appendix E Table 3. Evaluated on the XSum dataset revised by six LLMs, HLPD achieved the highest average AUROC across all three revision tasks. This performance represents a significant 32.38% improvement over Fast-DetectGPT and a 4.11% relative gain over ImBD, demonstrating HLPD's strong capability in multi-task revision and multi-context detection.

Robustness in Multiple Revision Iterations. To assess HLPD's robustness to multiple revision iterations, we evaluated its performance on texts undergoing one to five revisions from two LLMs: GPT-4o and Gemini-2.5-Pro. As shown in Appendix Table 4, HLPD consistently maintained highest average AUROC scores, indicating HLPD's detection efficacy even as texts undergo extensive iterative machine revisions.

3.4 Ablation Studies (RQ3 & RQ4)

Ablation on Strategy. As detailed in Table 4, our method, HLPD, achieves a remarkable average AUROC improvement

Method	Time cost (s/1k words)	GPT-3.5-turbo XSum	Writing	PubMed	Squad	Wiki	GPT-4o XSum	Writing	PubMed	Squad	Wiki	Overall Avg.
RoBERTa-base	0.07	0.5806	0.7225	0.4370	0.4588	0.5930	0.4921	0.4774	0.2496	0.4898	0.6512	0.5152
RoBERTa-large	0.11	0.6391	0.7236	0.4848	0.4863	0.6028	0.4782	0.4708	0.3089	0.4716	0.6378	0.5304
Entropy	0.35	0.6236	0.4564	0.5160	0.4942	0.3288	0.5351	0.3281	0.4923	0.4852	0.3074	0.4567
Likelihood	0.38	0.2774	0.5448	0.4481	0.4814	0.6250	0.4290	0.6834	0.4955	0.5206	0.6950	0.5200
Logrank	0.36	0.2528	0.4847	0.4454	0.4534	0.6097	0.4064	0.6581	0.4936	0.5008	0.6716	0.4976
LRR	0.41	0.2185	0.3208	0.4505	0.3760	0.5312	0.3647	0.5528	0.4820	0.4422	0.5641	0.4303
NPR	111.99	0.2873	0.5753	0.4118	0.4018	0.4822	0.4066	0.7067	0.4811	0.4980	0.5131	0.4764
DNA-GPT	35.92	0.2381	0.5284	0.4139	0.4611	0.6252	0.4492	0.6204	0.4933	0.4759	0.6720	0.4978
Detect-GPT	111.33	0.3118	0.6023	0.4320	0.4357	0.4629	0.4350	0.7270	0.4949	0.5195	0.5025	0.4924
Fast-DetectGPT	0.72	0.2684	0.5520	0.4404	0.4399	0.4192	0.3963	0.6212	0.4842	0.4718	0.4944	0.4588
ImBD	0.72	0.8651	0.8828	0.6218	0.9223	0.7930	0.7995	0.8136	0.6178	0.8396	0.7881	0.7870
GPTZero	1.64	0.9204	0.8484	0.6453	0.8490	0.6794	0.8023	0.8081	0.7640	0.7548	0.7986	0.7944
ReMoDetect	0.28	0.9013	0.8643	0.8343	0.8693	0.8293	0.7973	**0.8553**	0.7547	0.8133	0.8493	0.8368
Ghostbuster	0.23	0.9350	0.9728	0.8087	0.9033	0.9602	0.9295	0.8306	**0.7853**	0.8663	0.8681	0.8860
HLPD	0.72	**0.9998**	**0.9813**	**0.8125**	**0.9598**	**0.9796**	**0.9619**	0.8435	0.7393	**0.9125**	**0.9549**	**0.9144**

Table 1: Detection Performance of GPT-3.5 and GPT-4o rewritten text. Metric: AUROC.

Method	Source Model GPT-o3	DeepSeek-R1	Gemini-2.5	Grok-3	Avg.
Fast-DetectGPT	0.8164	0.3765	0.6716	0.6649	0.6324
ImBD	0.9238	0.9114	0.8688	0.9873	0.9228
GPTZero	0.8266	0.6787	0.8061	0.6882	0.7499
HLPD	**0.9780**	**0.9850**	**0.9453**	**0.9869**	**0.9738**

Table 2: Performance of Advanced Methods Across Generation Tasks on Xsum Dataset.

of 56% when benchmarked against Fast-DetectGPT, a baseline established without alignment techniques. HLPD further demonstrates consistent superiority by outperforming SFT, RLHF, ORPO and IPO by average margins of 30%, 26%, 23% and 23%, respectively. Notably, our optimized HLPO‡⋆ variant secures a 14.5% relative advantage over ImBD, strongly affirming the efficacy of detection paradigms aligned with human writing styles.

Ablation on Optimization. To demonstrate the efficacy of our innovations, as explained in Section 2.2, we ablated the core components of our loss function: the switch to a linear loss (from sigmoid activation) and the introduction of a dynamically adjusted β (compared to a fixed beta from DPO/ImBD). As detailed in Table 4, isolating the dynamic β (HLPO⋆) resulted in a 1% relative accuracy improvement. The linear loss function alone (HLPO‡) contributed a more substantial 5% relative improvement. The integration of these two strategies in HLPO (‡⋆) yielded a relative improvement of 8%, indicating a synergistic effect. This benefit extended to the ImBD baseline, where ImBD‡⋆ outperformed ImBD by 6%. These results confirm that our modifications are robustly beneficial, offering enhanced training performance, particularly when data or training epochs are limited.

3.5 Humanization Adaptive Attack Experiments

After HLPO, we get a model that prefers the human language style. This scoring model shows its potential in a fair amount

Model	Method	Dataset Xsum	Writing	SQuAD	Wiki	Avg.
GPT-4o	Fast-DetectGPT	0.3621	0.6280	0.4685	0.4896	0.4870
	ImBD	0.8601	0.9144	0.7735	0.8285	0.8441
	GPTZero	0.9133	0.7277	0.8630	0.7323	0.8091
	ReMoDetect	0.8916	0.9016	0.8116	0.9016	0.8766
	HLPD	**0.9612**	**0.9564**	**0.9108**	**0.9572**	**0.9464**
DeepSeek-R1	Fast-DetectGPT	0.2061	0.3612	0.3168	0.2924	0.2941
	ImBD	0.8745	0.8476	0.7520	0.7513	0.8064
	GPTZero	0.7997	0.6741	0.8336	0.7407	0.7620
	ReMoDetect	0.9201	0.9041	0.9281	0.8961	0.9121
	HLPD	**0.9730**	**0.9281**	**0.9287**	**0.9708**	**0.9501**
Claude-3.5	Fast-DetectGPT	0.2474	0.4572	0.4103	0.3722	0.3718
	ImBD	0.9105	0.9098	0.8110	0.8504	0.8704
	GPTZero	0.8266	0.6787	0.8061	0.6882	0.7499
	ReMoDetect	0.9086	0.8866	0.9196	0.8756	0.8976
	HLPD	**0.9988**	**0.9535**	**0.9383**	**0.9512**	**0.9604**
Gemini-2.5-Pro	Fast-DetectGPT	0.3148	0.5413	0.4525	0.3412	0.4125
	ImBD	0.8544	0.9120	0.7599	0.7284	0.8137
	GPTZero	0.8020	0.7060	0.7915	0.5745	0.7185
	ReMoDetect	0.9416	0.7816	**0.9016**	0.8216	0.8616
	HLPD	**0.9433**	**0.9328**	0.8980	**0.9394**	**0.9284**

Table 3: Performance on Adversarial Multi-Task Revision.

of downstream tasks besides detection. Here, we present a case that uses an additional strategy to make machine-revised texts less detectable. We apply the perturbation model making minor edits to introduce variability of human writing, and the scoring model after HLPO to pick up the most human-looked edited text. The process involves: **1) Perturbation Generation**: For each machine-revised text x_m, we generate perturbed versions $\tilde{x}_m$ by sampling from $q_\phi(\tilde{x}|x_m)$, introducing human-style variations. **2) Selection of Humanized Texts**: Among the perturbed texts, we select those with higher log-probabilities under the scoring model $\hat{p}_\theta$, as they are more likely to align with human writing styles. **3) Replacement**: Replace the original machine-revised text x_m with the selected $\tilde{x}_m$ to obtain a humanized version.

Strategy	Deepseek-R1				Gemini-2.5-Pro				Avg.
	XSum	Writ.	SQu.	Wiki	XSum	Writ.	SQu.	Wiki	
W/O A.	0.21	0.36	0.32	0.29	0.31	0.54	0.45	0.34	0.35
SFT	0.68	0.70	0.66	0.78	0.67	0.73	0.65	0.81	0.71
ORPO	0.75	0.70	0.67	0.79	0.77	0.74	0.69	0.80	0.74
IPO	0.75	0.70	0.67	0.79	0.71	0.78	0.61	0.67	0.71
RLHF	0.70	0.82	0.78	0.54	0.64	0.79	0.72	0.68	0.71
ImBD	0.87	0.84	0.75	0.75	0.85	0.91	0.76	0.73	0.81
ImBD‡⋆	0.81	0.92	0.79	0.85	0.87	0.92	0.87	0.85	0.86
HLPO	0.93	0.85	0.89	0.87	0.95	0.87	0.81	0.89	0.88
HLPO⋆	0.93	0.89	0.90	0.91	0.92	0.86	0.85	0.86	0.89
HLPO‡	0.92	0.90	0.89	0.86	**0.95**	0.87	**0.93**	0.94	0.91
HLPO‡⋆	**0.97**	**0.93**	**0.93**	**0.97**	0.94	**0.93**	0.90	**0.94**	**0.94**

[1] SFT (Ziegler et al. 2019), RLHF(Christiano et al. 2017), and ORPO(Hong, Lee, and Thorne 2024), IPO(Gheshlaghi Azar et al. 2024) are different preference strategies. "W/O A."denotes training with non-alignment or preference optimization.

[2] ‡ denotes the loss function incorporates our proposed linearization of HLPO; ⋆ denotes the loss function is scaled by a dynamically adjusted β to improve robustness.

Table 4: Ablation on Strategy & Optimization for Adversarial Multi-Task Revisions.

The selected $\tilde{x}_m$ can be applied to additional iterations for further reductions in detection probability. However, it is important to note that repeating this process might introduce more substantial semantic deviations from the original text.

Experiment Settings. We generate our text samples via a two-stage process. First, to create the base machine-revised text, we use GPT-3.5-Turbo to polish 100 paragraphs sourced from the XSum dataset. Second, for each revised paragraph, we employ a perturbation-and-selection method: T3-B5 generates 100 candidate perturbations, and our scoring model selects the one with the highest preference score as the final "humanized" output for each humanization.

Effectiveness Against Commercial Detectors. As shown in Appendix D Figure 4, our method reduces the detection AUROC by an average of 5% per iteration, culminating in a 20% total reduction after four iterations compared to the original machine-revised text. On the commercial GPTZero platform, this corresponds to a significant 74% decrease in the predicted AI probability. These results highlight the efficacy of our iterative humanization technique in evading commercial detection systems.

Effectiveness Against Open-Source Detectors. The method demonstrates similar effectiveness against open-source detectors. On the ImBD demo, our 4-iteration humanized text reduced the AI probability score by 44.5% (from 81.8% to 37.3%). Similarly, on the Ghostbuster demo, the probability dropped by 61% (from 87% to 26%). See detailed examples of the original and humanized texts in Appendix D.3.

4 Related Work

We classify related works into three parts, each with distinct limitations in discerning human-machine hybrid content.

Non-Alignment Detection. These approaches use either classifier training or logit-based analysis. Supervised models (Liu 2019) achieve domain-specific accuracy but suffer from distributional overfitting. Probability-based metrics (Mitchell et al. 2023; Liu 2019) analyze token-level characteristics through log-likelihood and curvature estimation, While methods like DetectGPT (Mitchell et al. 2023) enable zero-shot detection, they remain insensitive to style-preserving machine edits.

Machine-Aligned Detection. These new methods like ImBD employ style preference optimization on the scoring model to capture machine-generated patterns. Though effective for overt AI features, their imitation-based paradigm demonstrates limited discriminative power when confronting more advanced LLMs under diverse, adversarial prompts.

Human-Aligned Models. In NLP, the goal of aligning models with human text is a major focus, with efforts centered on making generated texts more human-like or better matching human intent. InstructGPT (Ouyang et al. 2022) and RLHF (Christiano et al. 2017) use human feedback to guide models toward producing more natural, human-understandable texts. Additionally, Chain-of-Thought (CoT) (Wei et al. 2022) prompting improves model reasoning by guiding the model to break down tasks into logical steps, allowing it to generate more interpretable and human-like output. However, these techniques still focus mainly on improving generation rather than capturing the difference between human-style and machine-style texts. Our HLPD (Human-Language-Preference-Detection) framework enlarges this gap by enabling human language style alignment. Unlike ImBD, HLPD trains the model towards a completely opposite direction for scoring, shows a 13.52% improvement in multi-task adversarial revisions and achieves a 5.53% higher AUROC in machine-generated text across four advanced LLMs compared to ImBD. Also, the trained scoring model shows some extra potential in downstream tasks like adaptive attack.

5 Conclusion and Limitations

We presented HLPD, which aligns scoring models with human language style, achieving significant accuracy improvements in detecting both fully machine-generated and multi-task revised texts under black-box conditions. HLPD's robustness and adaptability were validated through our novel adversarial multi-task evaluation framework, featuring a five-dimensional prompt generator and diverse LLMs. This framework confirmed HLPD's superiority over existing methods across various tasks and languages. The importance of preference optimization was underscored by ablation studies. This work advances text detection through style alignment, paving the way for more resilient systems, and demonstrated HLPD's wider potential in an adaptive attack on GPTZero. As for limitations, despite extensive evaluation across multiple state-of-the-art LLMs and diverse datasets, it remains unclear how the proposed detection approach will generalize to other models and domains not covered in our experiments. Additionally, our method may exhibit degraded performance when applied to very short sentences, where limited contextual information constrains reliable style and coherence based detection. Code will be made available at `https://github.com/dfq2021/HLPD`.

Acknowledgments

We thank all the anonymous reviewers for their constructive feedback. The authors are supported by NSFC (62502281), Shandong Provincial Natural Science Foundation (ZR2025QC1560), Jiangsu Provincial Natural Science Foundation (BK20250411), and Taishan Scholars Program.

References

Achiam, J.; Adler, S.; Agarwal, S.; Ahmad, L.; Akkaya, I.; Aleman, F. L.; Almeida, D.; Altenschmidt, J.; Altman, S.; Anadkat, S.; et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

AlAfnan, M. A.; and MohdZuki, S. F. 2023. Do artificial intelligence chatbots have a writing style? An investigation into the stylistic features of ChatGPT-4. *Journal of Artificial intelligence and technology*, 3(3): 85–94.

Anthropic. 2025. Cluade. https://claude.ai.

Bao, G.; Zhao, Y.; Teng, Z.; Yang, L.; and Zhang, Y. 2023. Fast-detectgpt: Efficient zero-shot detection of machine-generated text via conditional probability curvature. *arXiv preprint arXiv:2310.05130*.

Bian, N.; Lin, H.; Liu, P.; Lu, Y.; Zhang, C.; He, B.; Han, X.; and Sun, L. 2024. Influence of external information on large language models mirrors social cognitive patterns. *IEEE Transactions on Computational Social Systems*.

Bradley, R. A.; and Terry, M. E. 1952. Rank analysis of incomplete block designs: I. The method of paired comparisons. *Biometrika*, 39(3/4): 324–345.

Chen, J.; Zhu, X.; Liu, T.; Chen, Y.; Chen, X.; Yuan, Y.; Leong, C. T.; Li, Z.; Long, T.; Zhang, L.; Yan, C.; Mei, G.; Zhang, J.; and Zhang, L. 2024. Imitate Before Detect: Aligning Machine Stylistic Preference for Machine-Revised Text Detection.

Christiano, P. F.; Leike, J.; Brown, T.; Martic, M.; Legg, S.; and Amodei, D. 2017. Deep reinforcement learning from human preferences. *Advances in neural information processing systems*, 30.

de Lima-Santos, M.-F.; and Ceron, W. 2021. Artificial intelligence in news media: current perceptions and future outlook. *Journalism and media*, 3(1): 13–26.

Deepmind, G. 2024. Gemini - Google DeepMind. https://deepmind.google/technologies/gemini/.

DeepSeek-AI; et al. 2024. Deepseek-7B: A High-Performance Autoregressive Language Model for Natural Language Processing. *arXiv preprint arXiv:2403.01987*.

Fan, A.; Lewis, M.; and Dauphin, Y. N. 2018. Hierarchical Neural Story Generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, 889–898.

Fang, X.; Ng, D. T. K.; Leung, J. K. L.; and Chu, S. K. W. 2023. A systematic review of artificial intelligence technologies used for story writing. *Education and Information Technologies*, 28(11): 14361–14397.

Ferrara, E. 2023. Should chatgpt be biased? challenges and risks of bias in large language models. *arXiv preprint arXiv:2304.03738*.

Gehrmann, S.; Strobelt, H.; and Rush, A. M. 2019. Gltr: Statistical detection and visualization of generated text. *arXiv preprint arXiv:1906.04043*.

Gheshlaghi Azar, M.; Daniel Guo, Z.; Piot, B.; Munos, R.; Rowland, M.; Valko, M.; and Calandriello, D. 2024. A General Theoretical Paradigm to Understand Learning from Human Preferences. In Dasgupta, S.; Mandt, S.; and Li, Y., eds., *Proceedings of The 27th International Conference on Artificial Intelligence and Statistics*, volume 238 of *Proceedings of Machine Learning Research*, 4447–4455. PMLR.

Guo, B.; Zhang, X.; Wang, Z.; Jiang, M.; Nie, J.; Ding, Y.; Yue, J.; and Wu, Y. 2023. How close is chatgpt to human experts? comparison corpus, evaluation, and detection. *arXiv preprint arXiv:2301.07597*.

Guo, D.; Yang, D.; Zhang, H.; Song, J.; Zhang, R.; Xu, R.; Zhu, Q.; Ma, S.; Wang, P.; Bi, X.; et al. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Herbold, S.; Hautli-Janisz, A.; Heuer, U.; Kikteva, Z.; and Trautsch, A. 2023. A large-scale comparison of human-written versus ChatGPT-generated essays. *Scientific reports*, 13(1): 18617.

Hong, J.; Lee, N.; and Thorne, J. 2024. Orpo: Monolithic preference optimization without reference model. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, 11170–11189.

Howard, J.; and Ruder, S. 2018. Universal language model fine-tuning for text classification. *arXiv preprint arXiv:1801.06146*.

Ippolito, D.; Duckworth, D.; Callison-Burch, C.; and Eck, D. 2019. Automatic detection of generated text is easiest when humans are fooled. *arXiv preprint arXiv:1911.00650*.

Jiang, H.; Wang, Y.; et al. 2024. Mixtral: Combining Multiple Training Paradigms for Enhanced Autoregressive Models. *arXiv preprint arXiv:2402.11234*.

Jin, Q.; Dhingra, B.; Liu, Z.; Cohen, W. W.; and Lu, X. 2019. PubMedQA: A Dataset for Biomedical Research Question Answering. *arXiv preprint arXiv:1909.06146*.

Kelley, P. G.; Yang, Y.; Heldreth, C.; Moessner, C.; Sedley, A.; Kramm, A.; Newman, D. T.; and Woodruff, A. 2021. Exciting, useful, worrying, futuristic: Public perception of artificial intelligence in 8 countries. In *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*, 627–637.

Lee, H.; Tack, J.; and Shin, J. 2024. ReMoDetect: Reward Models Recognize Aligned LLM's Generations. *arXiv preprint arXiv:2405.17382*.

Liu, Y. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

M Alshater, M. 2022. Exploring the role of artificial intelligence in enhancing academic performance: A case study of ChatGPT. *Available at SSRN 4312358*.

McKenna, N.; Li, T.; Cheng, L.; Hosseini, M. J.; Johnson, M.; and Steedman, M. 2023. Sources of hallucination by large language models on inference tasks. *arXiv preprint arXiv:2305.14552*.

Merity, S.; Xiong, C.; Bradbury, J.; and Socher, R. 2016. Pointer Sentinel Mixture Models. *arXiv preprint arXiv:1609.07843*.

Mitchell, E.; Lee, Y.; Khazatsky, A.; Manning, C. D.; and Finn, C. 2023. Detectgpt: Zero-shot machine-generated text detection using probability curvature. In *International Conference on Machine Learning*, 24950–24962. PMLR.

Narayan, S.; Cohen, S. B.; and Lapata, M. 2018. 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*, 1797–1807.

OpenAI, T. 2022. Chatgpt: Optimizing language models for dialogue. OpenAI.

Ouyang, L.; Wu, J.; Jiang, X.; Almeida, D.; Wainwright, C.; Mishkin, P.; Zhang, C.; Agarwal, S.; Slama, K.; Ray, A.; et al. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35: 27730–27744.

Pu, D.; and Demberg, V. 2023. ChatGPT vs human-authored text: Insights into controllable text summarization and sentence style transfer. *arXiv preprint arXiv:2306.07799*.

Rafailov, R.; Sharma, A.; Mitchell, E.; Manning, C. D.; Ermon, S.; and Finn, C. 2024. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36.

Rajpurkar, P.; Zhang, J.; Lopyrev, K.; and Liang, P. 2016. SQuAD: 100,000+ Questions for Machine Comprehension of Text. *arXiv preprint arXiv:1606.05250*.

Reinhart, A.; Markey, B.; Laudenbach, M.; Pantusen, K.; Yurko, R.; Weinberg, G.; and Brown, D. W. 2025. Do LLMs write like humans? Variation in grammatical and rhetorical styles. *Proceedings of the National Academy of Sciences*, 122(8): e2422455122.

Solaiman, I.; Brundage, M.; Clark, J.; Askell, A.; Herbert-Voss, A.; Wu, J.; Radford, A.; Krueger, G.; Kim, J. W.; Kreps, S.; et al. 2019. Release strategies and the social impacts of language models. *arXiv preprint arXiv:1908.09203*.

Su, J.; Zhuo, T. Y.; Wang, D.; and Nakov, P. 2023. Detectllm: Leveraging log rank information for zero-shot detection of machine-generated text. *arXiv preprint arXiv:2306.05540*.

Tian, E.; and Cui, A. 2023. GPTZero: Towards detection of AI-generated text using zero-shot and supervised methods. *GPTZero*.

Touvron, H.; Martin, L.; Stone, K.; Albert, P.; Almahairi, A.; Babaei, Y.; Bashlykov, N.; Batra, S.; Bhargava, P.; Bhosale, S.; et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Tufts, B.; Zhao, X.; and Li, L. 2024. A practical examination of AI-generated text detectors for large language models. *arXiv preprint arXiv:2412.05139*.

Verma, V.; Fleisig, E.; Tomlin, N.; and Klein, D. 2024. Ghostbuster: Detecting Text Ghostwritten by Large Language Models. In Duh, K.; Gomez, H.; and Bethard, S., eds., *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, 1702–1717. Mexico City, Mexico: Association for Computational Linguistics.

Wei, J.; Wang, X.; Schuurmans, D.; Bosma, M.; Xia, F.; Chi, E.; Le, Q. V.; Zhou, D.; et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35: 24824–24837.

x.ai. 2024. Grok. https://x.ai/grok. Accessed: 2024-12-29.

Yang, L.; Chen, F.; et al. 2024. Qwen-2: Advanced Autoregressive Models with Efficient Training Techniques. *arXiv preprint arXiv:2401.09876*.

Yang, X.; Cheng, W.; Wu, Y.; Petzold, L.; Wang, W. Y.; and Chen, H. 2023. Dna-gpt: Divergent n-gram analysis for training-free detection of gpt-generated text. *arXiv preprint arXiv:2305.17359*.

Yuan, H.; Yuan, Z.; Tan, C.; Wang, W.; Huang, S.; and Huang, F. 2024. RRHF: Rank responses to align language models with human feedback. *Advances in Neural Information Processing Systems*, 36.

Ziegler, D. M.; Stiennon, N.; Wu, J.; Brown, T. B.; Radford, A.; Amodei, D.; Christiano, P.; and Irving, G. 2019. Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593*.