VaryBalance: Detecting LLM-generated Text through Variation

Xuecong Li , Xiaohong Li , Qiang Hu* , Yao Zhang , Junjie Wang
College of Intelligence and Computing, Tianjin University
{serenalee, xiaohongli, qianghu, zzyy, junjie.wang}@tju.edu.cn

Abstract

Detecting text generated by large language models (LLMs) is crucial but challenging. Existing detectors depend on impractical assumptions, such as white-box settings, or solely rely on text-level features, leading to imprecise detection ability. In this paper, we propose a simple but effective and practical LLM-generated text detection method, VaryBalance. The core of VaryBalance is that, compared to LLM-generated texts, there is a greater difference between human texts and their rewritten version via LLMs. Leveraging this empirical observation, VaryBalance quantifies this through Mean Squared Deviation and distinguishes human texts and LLM-generated texts. Comprehensive experiments demonstrated that VaryBalance outperforms the state-of-the-art detectors, i.e., Binoculars, by up to 34.3% in terms of AUROC, and maintains robustness against multiple generating models and languages.

1 Introduction

The strong text-generation capability of large language models (LLMs) and ease of access motivate users to rely on LLMs for content creation. However, machine-generated text has increasingly permeated the human information space consequently. According to [Liu *et al.*, 2024], since the release of GPT-3 and ChatGPT, the number of academic papers on arXiv identified as LLM-generated has grown exponentially. Without explicit disclosure of text provenance, LLM-generated content are easily misused across news, academia, and social media [Chen and Shu, 2023; 2024; Gressel *et al.*, 2024], leading to the spread of misinformation, academic misconduct, and phishing. Liu et al. [Liu *et al.*, 2024] also found that even experts perform only marginally better than random guessing on identifying LLM-generated texts, suggesting that humans are not capable enough to discern LLM-generated texts anymore. As a result, developing reliable detectors for LLM-generated text is crucial.

To address this, multiple automatic LLM-generated text detectors have been proposed, which can be roughly divided into white-box and black-box two groups. White-box methods need to know the target LLMs and access the internals of LLMs, such as logits. Representative white-box methods include DetectGPT [Mitchell *et al.*, 2023] and Fast-DetectGPT [Bao *et al.*, 2023]. Black-box approaches make predictions based solely on observable features of the text, or information obtained from surrogate models (not the target model). N-gram statistics [Yang *et al.*, 2023], Levenshtein score [Mao *et al.*, 2024], grammar errors [Wu *et al.*, 2025], and Neural-based methods [Hu *et al.*, 2023; Wu *et al.*, 2024; Chen *et al.*, 2025] are commonly used black-box methods.

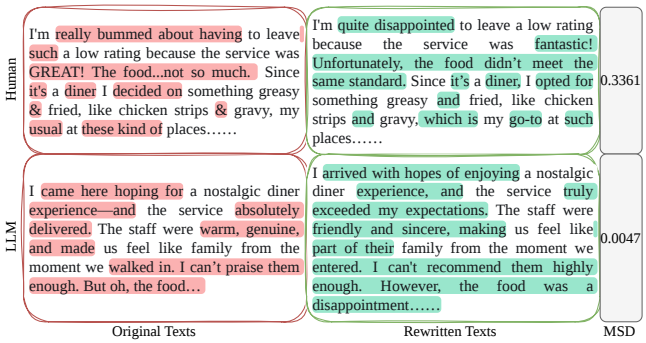

Figure 1: Comparison of Human/LLM texts and their rewritten texts.

However, both methodologies are constrained by their inherent limitation. For white-box detectors, it is generally impractical to precisely identify the underlying text generator in real-world scenarios, and access to the generator's logits is typically unavailable or infeasible. In contrast, as LLMs continue to advance, producing text that highly resembles human writing is no longer challenging, rendering these black-box methods (such as simple text-level methods) fail to detect LLM-generated texts.

To address these limitations, we introduce a simple but effective LLM-generated text detection method, VaryBalance. The core idea of VaryBalance is that, compared to human texts, there is a lighter difference between LLM-generated texts and their rewritten versions LLM. Figure 1 shows texts from humans and LLMs with their corresponding rewritten versions by GPT-4o-mini. We can see that the LLM-generated text and its rewritten version follow similar pat-

*Corresponding author

terns, and thus are more similar after quantified by Mean Squared Deviation(MSD). Based on this observation, VaryBalance employs a rewriter to generate multiple rewrites of an input text, and a scorer to compute the log perplexity of both the original and rewritten texts. After that, it uses the log perplexity and the MSD of the rewritten texts' log perplexities, computed with respect to the original text's log perplexity as the reference, as an indicator for LLM-generated content detection. Moreover, we present an extended scoring variant tailored to shorts and more stylistically diverse social media text.

To evaluate the effectiveness of VaryBalance, we conduct comprehensive experiments on eight datasets and five models. Results demonstrate that in formal writing contexts, VaryBalance outperforms state-of-the-art methods, such as Binocular and Fast-DetectGPT, with an overall improvement of 34.5% in terms of AUROC.

In summary, our contributions are:

- We empirically reveal that, compared to LLM-generated texts, there is a greater difference between human texts and their rewritten version via LLMs.
- Based on our observation, we propose a novel black-box LLM-generated text detection method, VaryBalance.
- We conduct comprehensive experiments to showcase the SOTA performance of VaryBalance and its robustness under different models and languages.

2 Related Work

Detecting LLM-generated texts has become an increasingly challenging task. LLMs achieve superior performance on a wide range of language benchmarks and can produce convincing, contextually appropriate text [Mitchell *et al.*, 2023]. Moreover. Liu et al. [Liu *et al.*, 2024] show that even domain experts perform only slightly better than random guessing when attempting to identify LLM-generated text. Hence, developing reliable detectors for LLM-generated content is both important and urgent.

We introduce approaches of white-box settings or black-box settings. There is also research [Kirchenbauer *et al.*, 2023] on embedding watermarks into LLM or the generated texts for further detection. Here, we do not discuss watermark-based methods as they demand cooperation from commercial or open-source LLM providers based on their inherent requirement for accessing model generation processes or deployment infrastructure, which is often infeasible.

2.1 White-box Methods

What characterizes white-box approaches is the direct access to the logits of the generative model, which evaluates the LLM's confidence over every possible next token. [Hans *et al.*, 2024] proposed Binocular, which measures the ratio between perplexity and cross-perplexity as a detection criterion. [Wu *et al.*, 2023] introduced LLMDet, a multi-model text attribution tool that constructs an n-gram–to–next-token probability mapping for each language model and computes proxy perplexities to classify the source of the text.

Beyond direct logit-based scoring, other white-box works leverage perturbation [Mitchell *et al.*, 2023; Bao *et al.*, 2023; 2025]. These methods perturb text using a language model and measure changes in log-probability or other scores under the target model.

Additional works that amend the existing works include TOCSIN [Ma and Wang, 2024], which combines token cohesiveness with conventional logit-based detection through a dual-channel architecture, and AdaDetectGPT [Zhou *et al.*, 2025], which learns an adaptive witness function to improve log probabilities. Text Fluoroscopy [Yu *et al.*, 2024] explores another methodology that exploits internal hidden representations of LMs for classification.

White-box detectors are typically zero-shot and easy to deploy. However, they rely on a less impractical scenario in which detectors have direct access to model logits. Most commercial LLM APIs do not expose logits, and many users lack the computational resources to deploy open-source models locally.

2.2 Black-box Methods

Black-box detectors do not require access to the model or any proxy logits. Instead, they rely on observable textual signals. The way these methods process these features includes text truncation and continuation [Yang *et al.*, 2023], grammar error correction [Wu *et al.*, 2025] and regeneration [Mao *et al.*, 2024].

Other works employ neural classifiers or fine-tuned language models. [Fagni *et al.*, 2021] demonstrated that fine-tuned RoBERTa achieves state-of-the-art classification performance. [Hu *et al.*, 2023] proposed RADAR, an adversarial learning framework that jointly trains a paraphraser and a detector to achieve robust AI-text detection.[Liu *et al.*, 2024] constructed the GPABench dataset to study the detection of ChatGPT-generated academic content and trained an LSTM-based classifier. [Nguyen-Son *et al.*, 2024] extended Raidar by incorporating multiple rewrites and ranking, followed by fine-tuned RoBERTa classification. [Chen *et al.*, 2025] formulates AI-generated text detection as an inverse prompt reasoning problem, and detects LLM-generated text by inverting the generation prompt and validating its consistency with the input text.

Despite diverse efforts, both white-box and black-box methods face intrinsic limitations. As LLMs continue to evolve, it is close to the world where LLM-generated texts are indistinguishable from human-written text, risking many fundamental theories of existing black-box approaches and highlighting the need for new detection paradigms.

3 VaryBalance

We propose **VaryBalance**, a black-box LLM-generated text detector. As shown in Figure 2, VaryBalance consists of three key components: generation of rewritten texts, scoring texts, and prediction.

Generation of rewritten texts. We leverage the advanced linguistic capabilities of a modern LLM to generate rewritten variants of the input text. We name the LLM Rewriter. To minimize the effect of the prompt, we only instruct the rewriter with a simple prompt: *Revise this text.*

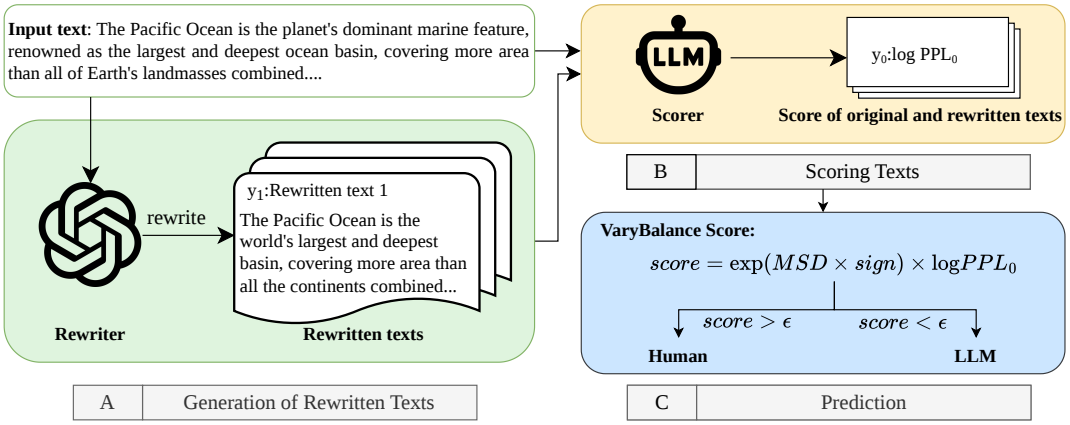

Figure 2: Overall workflow of VaryBalance. We first prompt an LLM to rewrite the text, then utilize a scoring model to calculate logPPL and the VaryBalance score of each text piece. The greater VaryBalance score indicates the more possibility of human text.

Scoring Texts. After the preparation of texts and their rewritten versions, a scorer is needed to transfer these texts to computable values for further comparison. VaryBalance employs Perplexity (PPL) as the quantification score. PPL measures how *confused* a model is in predicting the next token: lower perplexity indicates higher confidence and better predictive performance, making it a useful indicator for model-generated content, and has been widely used in quantifying texts [Solaiman *et al.*, 2019; Mitchell *et al.*, 2023; Bao *et al.*, 2023; Yang *et al.*, 2023; Hans *et al.*, 2024]

Given a token sequence $W = (w_1, w_2, \ldots, w_n)$, the perplexity of a model M is defined as:

$$\mathrm{PPL}(W) = \exp\left(-\frac{1}{n}\sum_{i=1}^{n}\log P_M(w_i \mid w_{<i})\right), \quad (1)$$

where $P_M(w_i \mid w_{<i})$ denotes the conditional probability of token w_i given its preceding context $w_{<i}$, and n is the length of the token sequence. However, direct use of PPL is not ideal for a scoring-based detector for the instability introduced by the exponential operation. Instead, we adopt the logarithmic form $\log(\mathrm{PPL})$ as a more stable and interpretable component in our framework. VaryBalance employs a small LLM to compute the $\log(\mathrm{PPL})$ of the original texts and rewritten variants, according to Eq. (1).

Prediction. The next step is to quantify the difference between texts and their rewritten variants. VaryBalance integrates Mean Squared Deviation (MSD) for this quantification. Here, $\log(\mathrm{PPL})$ is used as the indicator, and the log perplexity of the original text $\log(\mathrm{PPL}_0)$ is selected as the central reference.

$$\mathrm{MSD}(\log(\mathrm{PPL}_0)) = \frac{1}{n}\sum_{i=1}^{n}(\log(\mathrm{PPL}_i) - \log(\mathrm{PPL}_0))^2 \quad (2)$$

where $\log(\mathrm{PPL}_i)$ represents the log perplexities of rewritten texts, and n is the total number of rewritten texts.

We found that some machine-generated texts exhibit lower log perplexity scores than the rewritten texts, as they align more closely with the preferences of the relatively smaller scoring model. This results in inflated MSD values for some of LLM-generated texts. To address this issue, we introduce Eq. (3) to adjust the influence of MSD on the total score. With a negative sign, a larger MSD corresponds to a stronger indication of the LLM-generated text.

$$\mathrm{Sign} = \mathrm{Sign}(\log(\mathrm{PPL}_0) - \frac{1}{n}\sum_{i=1}^{n}\log(\mathrm{PPL}_i)) \quad (3)$$

Finally, VaryBalance calculates the final score as:

$$\mathrm{Score} = \exp(sign \cdot \mathrm{MSD}(\log(\mathrm{PPL}_0)) \cdot \log(\mathrm{PPL}_0) \quad (4)$$

We describe the influence of MSD on the original text's log perplexity using an exponential function, which allows human-written text to fully benefit from the amplification effect of MSD, thereby creating a clear separation from machine-generated text with similar log perplexity values. The score is then compared with a chosen threshold to classify LLM-generated text.

We also offer an expansion of the original VaryBalance score. We observe that for social media–style text, the gap in MSD between human-written text and the variance of its rewritten texts is substantially larger than that of machine-generated text. This discrepancy can be leveraged to refine the original Score for social media texts for boosting the detection performance.

$$\mathrm{Score_e} = \exp(sign \cdot \rho \cdot \mathrm{MSD}(\log(\mathrm{PPL}_0))) \cdot \log(\mathrm{PPL}_0)$$
$$\rho = \frac{\mathrm{MSD}(\log(\mathrm{PPL}_0)}{\mathrm{Var}(\log(\mathrm{PPL}_i)) \wedge 1 \le i \le n)} \quad (5)$$

where Var is the variance, and ρ as the expansion coefficient that is applied to texts with a more informal style.

4 Preliminary Study

The proposed new method is motivated by the observation that, compared to LLM-generated texts, there is a greater difference between human texts and their rewritten version via

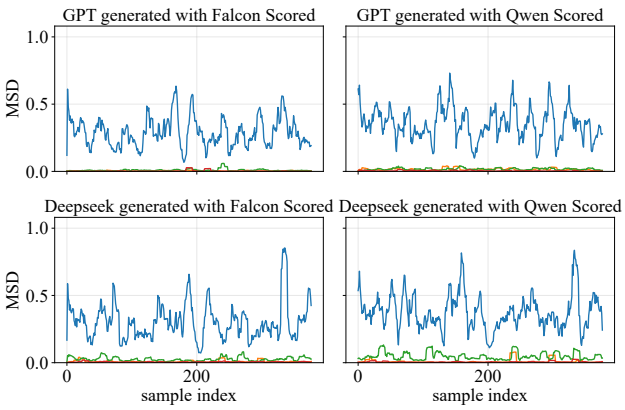

Figure 3: The MSD of human text, machine text, human text rewritten by LLM ,and machine text rewritten by LLM, Human text indicates texts are written by humans, the logic suits for machine text etc. The blue line represents human text.

LLMs. To systematically validate this assumption, we conduct a preliminary study.

Specifically, we first randomly sample 400 text pairs from the HC3 dataset [Guo *et al.*, 2023]. Each pair in the dataset consists of a question with both a human-written answer and a machine-generated answer. Then, we use these questions as prompts to produce new machine-generated responses, and rewrite machine-generated responses and original human texts using LLMs. As a result, we have two groups of data, human-generated texts with their rewritten versions, and machine-generated texts with their rewritten versions. After that, we employ MSD to quantify the difference between texts and their rewritten versions.

We consider two types of LLMs for the rewritten, DeepSeek-chat and GPT-4o-mini, and two types of LLMs as scorers, Falcon and Qwen. Fig. 3 depicts the results. We can see that it is clear that the MSD of human text and its rewritten texts is greater than that of the counterpart of LLM-generated texts and their rewritten texts. The average MSD value of human texts is around 0.34, while that of machine texts is around 0.009. Moreover, we observe that 96% of total texts in the dataset follow this observation, which demonstrates that our assumption stands.

To assess the effectiveness of VaryBalance, we conduct comparative experiments across multiple languages, generation models, generation strategies, and application scenarios. It is worth noting that the part of the dataset used to design the VaryBalance scoring function is not involved in the performance evaluation of VaryBalance.

4.1 Datasets

Benchmarks. We evaluate our method on several datasets that are widely used in prior literature. Among them, a commonly adopted benchmark introduced by [Verma *et al.*, 2024] consists of three datasets: Creative Writing, News, and Student Essay. For each dataset, the authors generate machine-written counterparts of the human texts using gpt-3.5-turbo as the training set, while additional texts generated by Claude are provided as a held-out test set.

Datasets for Robustness Experiment We select three datasets to represent different genres of texts with different lengths. Drawing samples from dataset CNN [Hermann *et al.*, 2015], reddit posts dataset RedditClean [1] and yelp review dataset YelpReviewFull [Zhang *et al.*, 2015], we generate corresponding LLM-generated texts using Qwen3-Next-80B-A3B-Instruct [Yang *et al.*, 2025; Team, 2025] and grok-4-fast [2]. All human texts were collected before the release of ChatGPT, which ensures that they are not contaminated by LLM-generated content.

Multi-language. We evaluate the multilingual capability of our detector using the M4 dataset[Wang *et al.*, 2024], which contains texts in Chinese, Arabic, Urdu, and Bulgarian collected from multiple sources. We regenerate the machine-generated texts with qwen3-235b-a22b-instruct-2507[Yang *et al.*, 2025; Team, 2025] by following the prompts provided in the dataset, thereby better simulating outputs from more recent and capable large language models. To reduce bias towards non-native speaker, we additionally employ the EssayForum[3] dataset, which contains both original student-written essays and their grammar-corrected counterparts.

Rewrite. To evaluate the detector's ability to identify rewritten text, we construct a rewritten-text dataset based on the HC3 dataset[Guo *et al.*, 2023] using two large language models, deepseek-chat and qwen3-235b-a22b-instruct-2507.

4.2 Baseline

Among the white-box detectors, we compare our detector with Fast-DetectGPT[Bao *et al.*, 2023], Binoculars[Hans *et al.*, 2024] and AdaDetectGPT [Zhou *et al.*, 2025]. For black-box detectors, we choose RADAR[Hu *et al.*, 2023] and DNA-GPT[Yang *et al.*, 2023] as baselines. RADAR presents a neural-based detector trained through adversarial learning for robust AI-text detection, while DNA-GPT is mainly based on word-level features caused by text continuation through an LLM. All detectors are implemented in the setting of their original papers to avoid confounders.

4.3 Metric

Following previous works [Mitchell *et al.*, 2023; Ma and Wang, 2024; Zhou *et al.*, 2025], we evaluate the detection effectiveness using the area under the receiver operating characteristic curve (AUROC).AUROC indicates the probability of a detector to rank a randomly-selected positive sample higher than a negative counterpart, capturing the overall performance of the subject.

5 Experiment

In large-scale experiments, we employed Qwen3-14B as the scoring model with bfloat16 precision. All text rewriting and generation were performed under consistent settings: temperature=0.7, max_tokens=1000, with each input text rewritten 10 times.

[1] https://huggingface.co/datasets/SophieTr/reddit_clean
[2] https://x.ai/news/grok-4-fast
[3] https://huggingface.co/datasets/nid989/EssayFroum-Dataset

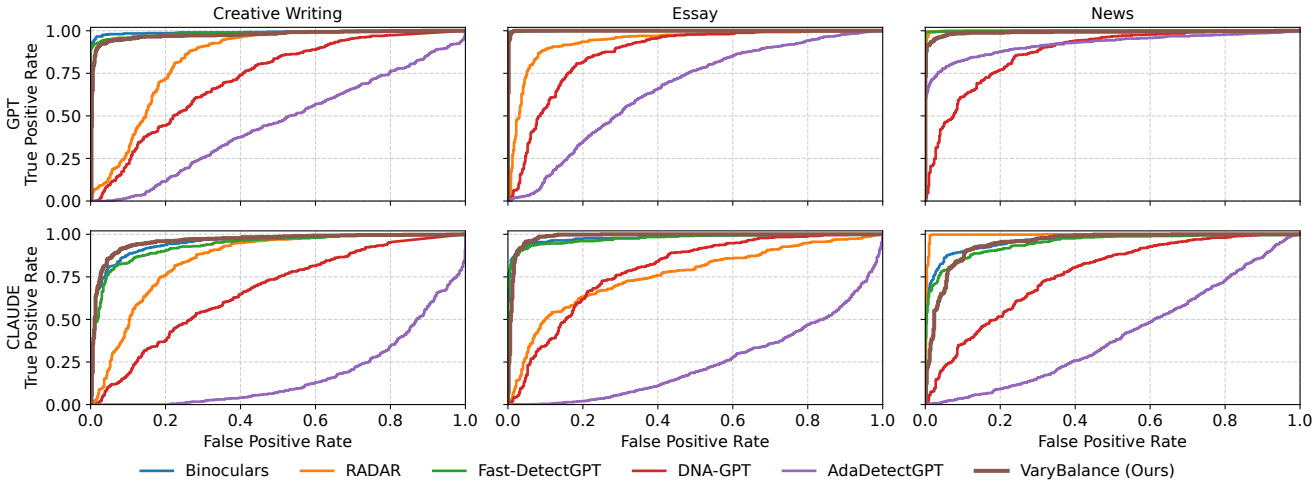

Figure 4: The ROC curve of VaryBalance and the other four baselines on the benchmark datasets. Upper: GPT as the source model; Lower: Claude as the source model.

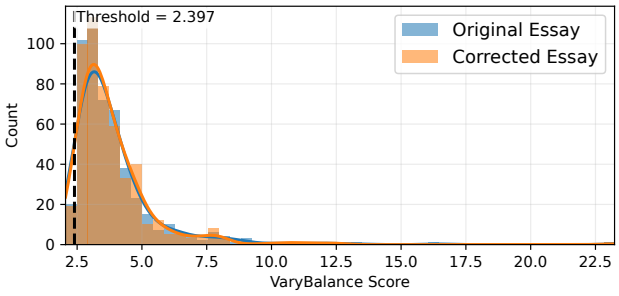

Figure 5: The comparison of score distribution of VaryBalance on Essay Forum dataset. The threshold is Youden's J statistic.

Detectors	Grok	Qwen	GPT
DNA-GPT	0.6216	0.6406	0.7035
RADAR	0.8413	0.7817	0.9842
Binoculars	0.6763	0.9009	0.9846
Fast-DetectGPT	0.6683	0.8876	0.9623
AdaDetectGPT	0.2693	0.6352	0.3285
VaryBalance	0.9083	0.9491	0.9870
VaryBalance with expansion	**0.9342**	**0.9862**	**0.9936**

Table 1: AUROC score of different detectors on Reddit under Grok, Qwen and GPT generations.

5.1 Results

Benchmark Performance

As shown in Fig. 4, VaryBalance consistently outperforms DNA-GPT, RADAR, and AdaDetectGPT across most experimental settings, while achieving performance comparable to Fast-DetectGPT and Binoculars. When the generation model shifts to Claude, most detectors exhibit a slight performance degradation while VaryBalance maintains a high level of effectiveness and continues to surpass the baseline methods.

It is worth noting that the machine-generated portions of the Creative Writing, Essay, and News datasets are produced using gpt-3.5-turbo, which is relatively less capable and slower compared to more recent models. Evaluating performance solely on these datasets is insufficient.

Performance on Various Text Sources and Genres

In real-world applications, LLM-generated text exhibits substantial variation in length, style, domain, and generating models. Therefore, we evaluate our detector on the Reddit, Yelp, and CNN datasets, which represent different genres of texts with different lengths from different source models, enabling us to assess the detector's robustness to LLM-generated content beyond the GPT and Claude model families. The result shown in Fig. 6 demonstrates that VaryBalance surpasses baselines by a substantial performance gap on Yelp and Reddit datasets while maintaining comparable effectiveness with baselines on CNN datasets. Combining the benchmark results, we observe that the baselines perform well on relatively formal and longer texts (e.g., mainstream news) but exhibit pronounced performance degradation on social media-style text. Our detector remains robust to the change in text genres with an AUROC of 0.949 on Reddit, 0.963 on Yelp, and 0.997 on CNN, surpassing other detectors by margins of 0.30.

In terms of the impact of generating models, we perform comparative evaluations on Reddit texts generated by three models, Grok, Qwen, and gpt-4o-mini(represented as GPT), with the results summarized in Table 1. VaryBalance outperforms baseline methods, with the expansion further enhancing its advantages, demonstrating robustness across generating models and text genres.

Multi-language Scenarios

Our detector nearly achieves the best performance across all evaluated languages. As shown in Table 2, on ChatGPT-generated texts, VaryBalance attains state-of-the-art performance, with an AUROC of up to 0.986. On text generated

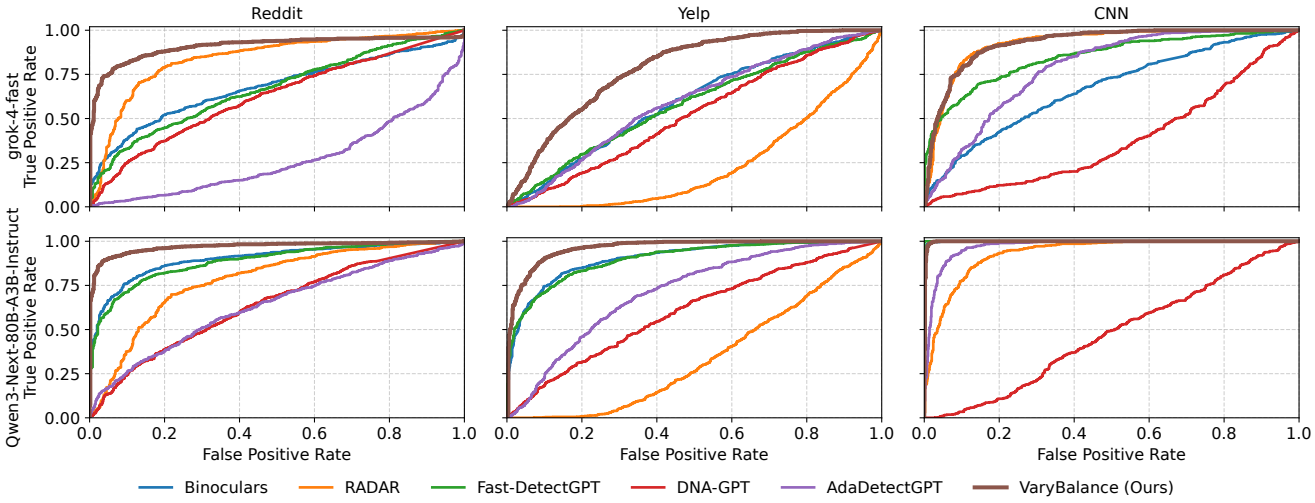

Figure 6: The ROC curve of VaryBalance and the other five baselines on the CNN, Reddit and Yelp. Upper: grok as the source model; Lower: qwen as the source model.

	ChatGPT				Qwen3-235b-a22b-instruct			
Detectors	Arabic	Bulgarian	Urdu	Chinese	Arabic	Bulgarian	Urdu	Chinese
DNA-GPT	0.2117	0.3145	0.4571	0.6310	0.6780	0.5904	0.4571	0.6310
RADAR	0.3570	0.6969	0.4462	0.3631	0.4725	0.7842	0.4462	0.3631
binoculars	0.8355	0.8658	0.9420	0.7749	0.7680	0.6280	0.9421	0.7749
fast-detectgpt	0.6146	0.8692	0.7846	0.4416	0.6170	0.4466	0.7847	0.4416
AdaDetectGPT	0.5632	**0.8847**	0.6784	0.4728	0.9489	0.8851	0.6527	0.9148
VaryBalance	**0.8357**	0.8821	**0.9867**	**0.8581**	**0.9708**	**0.9444**	**0.8917**	**0.9752**

Table 2: The AUROC scores on detecting the four languages where ChatGPT and Qwen3-235b-a22b-instruct are generating models. The AUROC scores are rounded to four decimal places.

by Qwen, VaryBalance consistently achieves AUROC values above 0.892, with the highest reaching 0.975, significantly surpassing the performance of other methods.

An important challenge in machine-generated text detection is that some detectors exhibit a bias toward classifying texts written by non-native English speakers as machine-generated [Liang *et al.*, 2023]. On the EssayForum dataset, VaryBalance achieves an accuracy of 0.98. Meanwhile, Figure. 5 shows that the score distributions of the original essay and grammar-corrected essay are highly similar, indicating that our detector does not exhibit bias against text written by non-native English speakers.

Performance on Rewritten Texts

Prior studies have shown that machine-rewritten text is particularly difficult to detect[Wahle *et al.*, 2022; Pudasaini *et al.*, 2024]. As shown in Table 3, VaryBalance significantly outperforms baselines: across three different models, it consistently surpasses other detectors, achieving AUROC scores above 0.82 in all cases. Moreover, VaryBalance with expansion further increases AUROC by 0.0256 on average across three settings.

Overall, the experimental results show that VaryBalance is robust to variations in text type, length, generation model, and language. It effectively identifies LLM-rewritten text and, through the incorporation of a single parameter, further enhances performance on shorter, social media–style texts, surpassing current state-of-the-art detection models.

Detectors	GPT	Qwen	DS
DNA-GPT	0.5151	0.4963	0.5061
RADAR	0.3570	0.3641	0.3642
Binoculars	0.8225	0.6272	0.5882
Fast-DetectGPT	0.8103	0.6004	0.5671
AdaDetectGPT	0.7673	0.5980	0.6288
VaryBalance	0.9098	0.8919	0.8200
VaryBalance with expansion	**0.9299**	**0.9249**	**0.8438**

Table 3: The AUROC on the Rewrite dataset. GPT: GPT-4o-mini, Qwen: Qwen3-Next-80B-A3B-Instruct, DS: deepseek-chat. The AUROC scores are rounded to four decimal places.

6 Discussion

We investigate the contribution of each component in VaryBalance to its overall performance, including the scoring function, the number of rewrites, the text length, and the choice of the scoring model.

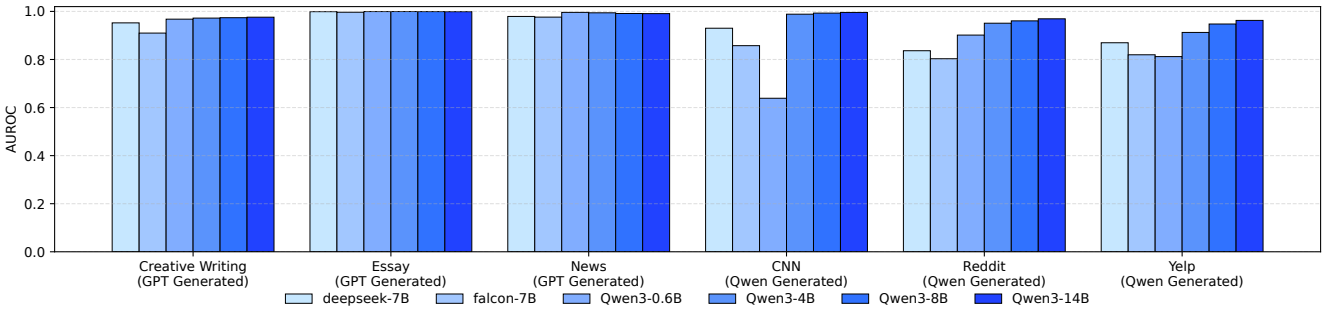

Figure 7: Comparison on performance of different series and size of scoring model.

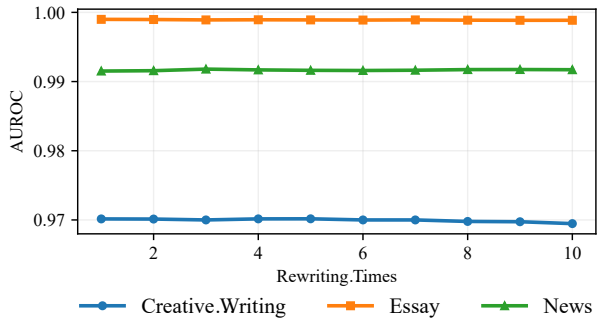

Figure 8: The AUROC scores are stable regardless of regeneration times.

Scores	Reddit	Essay	Rewrite
log PPL	0.9421	**0.9989**	0.8119
VaryBalance	0.9692	**0.9989**	0.9097
VaryBalance with expansion	**0.9861**	0.9964	**0.9299**

Table 4: The AUROC score of log PPL, VaryBalance and VaryBalance with expansion.

6.1 Size and Series of Scorers

We evaluate the impact of the scoring model on VaryBalance by selecting Qwen3 models with parameter sizes ranging from 0.6B to 14B, as well as several 7B models from other families, and applying the base VaryBalance framework across six datasets. As shown in the Fig. 7, on benchmarks, the choice of the scoring model has only a limited effect on performance, while a more pronounced performance gap shows on other datasets that were constructed using more recent models. On these datasets, we observe a clear trend of performance improvement with increasing model size. Among the 7B-scale models, DeepSeek-7B consistently outperforms Falcon-7B, while the slightly larger Qwen3-8B consistently surpasses both DeepSeek-7B and Falcon-7B. Notably, on the Reddit and benchmark datasets, even the smallest Qwen3-0.6B model outperforms DeepSeek-7B and Falcon-7B.

These results indicate that the choice of the scoring model has a significant impact on the performance of VaryBalance, encompassing both the selection of the model family and the model size. Due to computational constraints, we do not evaluate larger models; however, the experimental findings suggest that, when computational resources permit, larger Qwen models can further enhance the effectiveness of VaryBalance. The impact of model selection for the scorer is more pronounced when evaluated on texts generated by recent large language models.

6.2 Regeneration Times

The experiment shows that the number of regeneration times has only a marginal effect on overall performance, see Fig.8. We hypothesize that multiple rewritten outputs produced by the same LLM tend to have similar log perplexity values, resulting in low variance among the rewritten texts and consequently a stable MSD. Therefore, in practical applications, the number of rewrites can be selected flexibly according to API constraints without significantly affecting the performance of VaryBalance.

6.3 The influence of MSD

To verify the effectiveness of our modification to log PPL, we conduct comparative experiments on the Reddit, Essay, and Rewrite datasets, which respectively represent social media text, academic text, and machine-rewritten text. As shown in Table. 4, both variants of the VaryBalance score achieve better performance than log PPL score, and VaryBalance with expansion yields noticeable improvements on the Reddit and Rewrite dataset, while exhibiting a slight performance degradation on the Essay dataset. These results confirm the effectiveness of our modification to log PPL, and indicate that the impact of MSD is less pronounced for longer and more formal texts, and therefore its contribution should not be overly amplified in such settings.

7 Conclusion

We proposed VaryBalance, a black-box LLM-generated text detector based on the observation that the difference between the MSD of human text and machine text is significant. Extensive experiments show that VaryBalance surpasses SOTA detectors by 0.3 in terms of the AUROC score while keeping robustness against multiple genres, multiple generating models, and multiple languages.

Acknowledgments

This work was partly supported by the National Key Research and Development Program of China (2023YFB3107100).

References

[Bao *et al.*, 2023] Guangsheng Bao, Yanbin Zhao, Zhiyang Teng, Linyi Yang, and Yue Zhang. Fast-detectgpt: Efficient zero-shot detection of machine-generated text via conditional probability curvature. *arXiv preprint arXiv:2310.05130*, 2023.

[Bao *et al.*, 2025] Guangsheng Bao, Yanbin Zhao, Juncai He, and Yue Zhang. Glimpse: Enabling White-Box Methods to Use Proprietary Models for Zero-Shot LLM-Generated Text Detection, February 2025. arXiv:2412.11506 [cs].

[Chen and Shu, 2023] Canyu Chen and Kai Shu. Can llm-generated misinformation be detected? *arXiv preprint arXiv:2309.13788*, 2023.

[Chen and Shu, 2024] Canyu Chen and Kai Shu. Combating misinformation in the age of llms: Opportunities and challenges. *AI Magazine*, 45(3):354–368, 2024.

[Chen *et al.*, 2025] Zheng Chen, Yushi Feng, Changyang He, Yue Deng, Hongxi Pu, and Bo Li. Ipad: Inverse prompt for ai detection–a robust and explainable llm-generated text detector. *arXiv preprint arXiv:2502.15902*, 2025.

[Fagni *et al.*, 2021] Tiziano Fagni, Fabrizio Falchi, Margherita Gambini, Antonio Martella, and Maurizio Tesconi. TweepFake: About detecting deepfake tweets. *PLOS ONE*, 16(5):e0251415, May 2021. Publisher: Public Library of Science.

[Gressel *et al.*, 2024] Gilad Gressel, Rahul Pankajakshan, and Yisroel Mirsky. Discussion paper: Exploiting llms for scam automation: A looming threat. In *Proceedings of the 3rd ACM Workshop on the Security Implications of Deepfakes and Cheapfakes*, pages 20–24, 2024.

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

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

[Hermann *et al.*, 2015] Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. Teaching machines to read and comprehend. *Advances in neural information processing systems*, 28, 2015.

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

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

[Liang *et al.*, 2023] Weixin Liang, Mert Yuksekgonul, Yining Mao, Eric Wu, and James Zou. Gpt detectors are biased against non-native english writers. *Patterns*, 4(7), 2023.

[Liu *et al.*, 2024] Zeyan Liu, Zijun Yao, Fengjun Li, and Bo Luo. On the Detectability of ChatGPT Content: Benchmarking, Methodology, and Evaluation through the Lens of Academic Writing. In *Proceedings of the 2024 on ACM SIGSAC Conference on Computer and Communications Security*, CCS '24, pages 2236–2250, New York, NY, USA, December 2024. Association for Computing Machinery.

[Ma and Wang, 2024] Shixuan Ma and Quan Wang. Zero-Shot Detection of LLM-Generated Text using Token Cohesiveness. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen, editors, *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 17538–17553, Miami, Florida, USA, November 2024. Association for Computational Linguistics.

[Mao *et al.*, 2024] Chengzhi Mao, Carl Vondrick, Hao Wang, and Junfeng Yang. Raidar: geneRative AI Detection viA Rewriting, April 2024. arXiv:2401.12970 [cs].

[Mitchell *et al.*, 2023] Eric Mitchell, Yoonho Lee, Alexander Khazatsky, Christopher D. Manning, and Chelsea Finn. DetectGPT: Zero-Shot Machine-Generated Text Detection using Probability Curvature. In *Proceedings of the 40th International Conference on Machine Learning*, pages 24950–24962. PMLR, July 2023. ISSN: 2640-3498.

[Nguyen-Son *et al.*, 2024] Hoang-Quoc Nguyen-Son, Minh-Son Dao, and Koji Zettsu. Simllm: Detecting sentences generated by large language models using similarity between the generation and its re-generation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 22340–22352, 2024.

[Pudasaini *et al.*, 2024] Shushanta Pudasaini, Luis Miralles-Pechuán, David Lillis, and Marisa Llorens Salvador. Survey on ai-generated plagiarism detection: The impact of large language models on academic integrity. *Journal of Academic Ethics*, pages 1–34, 2024.

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

[Team, 2025] Qwen Team. Qwen3 technical report, 2025.

[Verma *et al.*, 2024] Vivek Verma, Eve Fleisig, Nicholas Tomlin, and Dan Klein. Ghostbuster: Detecting Text Ghostwritten by Large Language Models, April 2024. arXiv:2305.15047 [cs].

[Wahle *et al.*, 2022] Jan Philip Wahle, Terry Ruas, Frederic Kirstein, and Bela Gipp. How large language models are transforming machine-paraphrase plagiarism. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang, editors, *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 952–963, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics.

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

[Wu *et al.*, 2023] Kangxi Wu, Liang Pang, Huawei Shen, Xueqi Cheng, and Tat-Seng Chua. Llmdet: A third party large language models generated text detection tool. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 2113–2133, 2023.

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

[Wu *et al.*, 2025] Junchao Wu, Runzhe Zhan, Derek F. Wong, Shu Yang, Xuebo Liu, Lidia S. Chao, and Min Zhang. Who wrote this? the key to zero-shot LLM-generated text detection is GECScore. In Owen Rambow, Leo Wanner, Marianna Apidianaki, Hend Al-Khalifa, Barbara Di Eugenio, and Steven Schockaert, editors, *Proceedings of the 31st International Conference on Computational Linguistics*, pages 10275–10292, Abu Dhabi, UAE, January 2025. Association for Computational Linguistics.

[Yang *et al.*, 2023] Xianjun Yang, Wei Cheng, Yue Wu, Linda Petzold, William Yang Wang, and Haifeng Chen. DNA-GPT: Divergent N-Gram Analysis for Training-Free Detection of GPT-Generated Text, October 2023.

[Yang *et al.*, 2025] An Yang, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoyan Huang, Jiandong Jiang, Jianhong Tu, Jianwei Zhang, Jingren Zhou, Junyang Lin, Kai Dang, Kexin Yang, Le Yu, Mei Li, Minmin Sun, Qin Zhu, Rui Men, Tao He, Weijia Xu, Wenbiao Yin, Wenyuan Yu, Xiafei Qiu, Xingzhang Ren, Xinlong Yang, Yong Li, Zhiying Xu, and Zipeng Zhang. Qwen2.5-1m technical report. *arXiv preprint arXiv:2501.15383*, 2025.

[Yu *et al.*, 2024] Xiao Yu, Kejiang Chen, Qi Yang, Weiming Zhang, and Nenghai Yu. Text Fluoroscopy: Detecting LLM-Generated Text through Intrinsic Features. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen, editors, *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 15838–15846, Miami, Florida, USA, November 2024. Association for Computational Linguistics.

[Zhang *et al.*, 2015] Xiang Zhang, Junbo Zhao, and Yann LeCun. Character-level convolutional networks for text classification. *Advances in neural information processing systems*, 28, 2015.

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