Navigating the Shadows: Unveiling Effective Disturbances for Modern AI Content Detectors

Anonymous ACL submission

Abstract

With the launch of ChatGPT, large language models (LLMs) have attracted global attention. In the realm of article writing, LLMs have witnessed extensive utilization, giving rise to concerns related to intellectual property protection, personal privacy, and academic integrity. In response, AI-text detection has emerged to distinguish between human and machine-generated content. However, recent research indicates that these detection systems often lack robustness and struggle to effectively differentiate perturbed texts. Currently, there is a lack of systematic evaluations regarding detection performance in real-world applications, and a comprehensive examination of perturbation techniques and detector robustness is also absent. To bridge this gap, our work simulates real-world scenarios in both informal and professional writing, exploring the out-of-the-box performance of current detectors. Additionally, we have constructed 12 black-box text perturbation methods to assess the robustness of current detection models across various perturbation granularities. Furthermore, through adversarial learning experiments, we investigate the impact of perturbation data augmentation on the robustness of AI-text detectors. After the review process, we will publicly release all our code and data.

1 Introduction

With the rise of LLMs (OpenAI, 2023; Anil et al., 2023; Touvron et al., 2023), concerns about the misuse of generated content have been growing (McKenna et al., 2023; Bian et al., 2023; Ferrara, 2023), making AI-Text detection a topic of significant attention from the research community. Several methods for detecting AI-generated text have recently been proposed, including fine-tuned classifiers (Uchendu et al., 2020; Liu et al., 2023b), statistical approaches (Lavergne et al., 2008; Mitchell et al., 2023), watermarking (Atallah et al., 2001; Kirchenbauer et al., 2023a), and

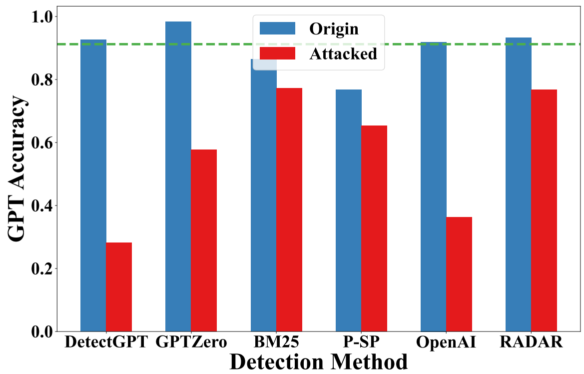

Figure 1: Performance of state-of-the-art AI-text detectors significantly decreases after introducing perturbation attacks. The green dashed threshold line represents the adversarially trained RoBERTa classifier detector, achieving a detection accuracy of 0.912 on the mixed test data of original and perturbed text.

retrieval techniques (Krishna et al., 2023). Additionally, online education service providers such as Copyleak[1] and GPTZero (Tian and Cui, 2023) have introduced AI text detection services. However, criticisms regarding misclassification results from various users have surfaced. Simultaneously, in domains like essay writing, there is a demand from users to bypass AI text detection using perturbation methods. Numerous open-source tools like GPTzzz[2] and GPTZero-Bypasser[3] have emerged to address this need.

Recent efforts have begun to explore the vulnerabilities of current detection models (Sadasivan et al., 2023; Liang et al., 2023; Tripto et al., 2023), utilizing methods such as rewrite and substitution to modify AI-generated content, rendering it indistinguishable from human-authored text. This underscores the importance of investigating and identifying potential weaknesses in current detectors before their deployment, ensuring their robustness and

[1] https://copyleaks.com/ai-content-detector
[2] https://github.com/Declipsonator/GPTZzzs
[3] https://github.com/o2161405/GPTZero-Bypasser

mitigating potential risks. Simultaneously, more comprehensive work has started to summarize the issues with current detection methods and propose corresponding robustness enhancement techniques, such as RADAR (Hu et al., 2023) and retrieval (Krishna et al., 2023). Despite enhancing the models' defense against specific types of text perturbations to some extent, these works still face two major limitations. Firstly, these efforts primarily focus on AI text detection in specific writing scenarios. Secondly, they typically involve only one type of perturbation, i.e., paraphrasing. In practical applications, detectors are likely to encounter a more complex and diverse set of scenarios, involving various application contexts and potential text perturbations.

To this end, our work aims to investigate and analyze the accuracy and robustness of various AI text detection algorithms in simulating real writing scenarios. Specifically, within three categories of AI text detection methods, we evaluate six representative off-the-shelf models on data generated by ChatGPT. To simulate users' writing demands, we categorize AI-generated text into professional and informal writing scenarios and test detection accuracy accordingly. As expected, current text detection models exhibit lower accuracy in professional writing scenarios. Furthermore, following an exploration of current text perturbation methods, we devise 12 types of text perturbations across four granularities. We apply these perturbations to the test data, generating 120,000 adversarial samples to investigate the robustness of current detection systems. The results reveal that, apart from the extensively studied paraphrase methods, word-level perturbations also significantly reduce AI text detection rates. Building on earlier work, we further delve into exploring the minimum budget for adversarial learning to train robust text detectors. Additionally, we conduct preliminary investigations into transfer learning in the context of adversarial text detection.

Our work can be summarized into three parts: 1) We validate the detection accuracy of three types of current detection models in both professional and informal writing scenarios. This analysis identified a lack of generalization performance in current detection systems. 2) We systematically and hierarchically design AI-Text perturbation methods. The results demonstrated that perturbations at various granularities significantly reduced detection performance. Additionally, we observed inconsistent performances of different detection models when faced with perturbations. 3) Budget and transfer experiments provide references and suggestions for future efforts to enhance the robustness of AI-Text detectors.

2 Related Works

AI-Text Detection. The current methods for AI-text detection can be categorized into four classes: 1) **Statistical** approaches leverage statistical tools, using metrics such as information entropy, perplexity, and n-gram frequencies to differentiate between human and machine-generated text in a zero-shot manner (Lavergne et al., 2008; Gehrmann et al., 2019; Solaiman et al., 2019; Mitchell et al., 2023; Su et al., 2023). Notable commercial applications include GPTZero (Tian and Cui, 2023), and recent open-source efforts are exemplified by DetectGPT (Mitchell et al., 2023), which defines a curvature-based criterion using a log probability function for AI detection. 2) **Watermark-based** methods (Atallah et al., 2001, 2002; Kirchenbauer et al., 2023a; Liu et al., 2023a) is also evolving with the emergence of LLMs, where Kirchenbauer et al. (2023a) randomly partition the vocabulary into a greenlist and a redlist during generation, based on the hash of previously generated tokens. 3) **Classifier-based** detectors (Uchendu et al., 2020; Deng et al., 2023; Mireshghallah et al., 2023; Guo et al., 2023; Liu et al., 2023b) based on supervised data typically utilize RoBERTa (Liu et al., 2019) to train binary classifiers for text detection. Recent efforts includes the OpenAI's release of detection tools (Solaiman et al., 2019), and the RADAR (Hu et al., 2023), which specifically address the importance of perturbation attacks, and enhance detection robustness through adversarial learning using paraphrasers. 4) **Retrieval-based** method proposed by Krishna et al. (2023) involves collecting historical output data from language models and assessing the AI generation likelihood of the text through semantic matching.

Adversarial Attacks. In addition, some studies (Ren et al., 2023; Tripto et al., 2023; Lu et al., 2023; Liang et al., 2023; Cai and Cui, 2023) have addressed the impact of text perturbations on AI text detection. For instance, both Sadasivan et al. (2023); Krishna et al. (2023) proposed to use paraphraser as the attacker to rewrite AI content, demonstrating effective attacks on many detectors. Kirchenbauer et al. (2023b) validated the detection capabilities of watermarking detectors in scenarios involving a mix of human and machine-generated text. Furthermore, Shi et al. (2023) examined the significant impact of synonym perturbations on text detection performance. Kumarage et al. (2023) designed prompts to generate outputs more similar to human text, evading detection by existing detectors. Notably, a very recent work, Macko et al. (2024) focused on designing perturbations such as paraphrasing, back translation, and substitution in a multilingual environment. They demonstrated the vulnerability of current multilingual AI text detectors and the effectiveness of adversarial training. In comparison to their work, our study concentrates on the detectability of AI-generated text in real-world scenarios. We employ AI-generated text outputs that closely resemble human output, design a more comprehensive set of perturbation attacks, and importantly, extend our focus beyond simple classifier methods. We evaluate the detection performance not only for classifiers but also for retrieval and other detection tools.

3 Experimental Setup

In this section, we initially surveyed the current state-of-the-art AI-text detection frameworks. Subsequently, considering the presence of intentional or unintentional perturbation attacks in real-world applications that can impact the performance of detection models, we synthesized and implemented 12 black-box perturbation methods. Here, "black-box" refers to perturbation algorithms lacking access to accurate internal information, such as gradients or hidden states, of the detection model. Meanwhile, building upon the scoring-based configuration of existing detectors, we further explored the challenges associated with metric selection and threshold determination in the evaluation.

3.1 Off-the-Shelf Detectors

As described in Section 2, the current research in AI detection primarily focuses on four directions. However, watermarking has not been extensively applied to commercial or open-source LLMs, lacking practical application scenarios. Therefore, we consider three readily deployable detection models: 1) statistical models, i.e., DetectGPT (Mitchell et al., 2023) and GPTZero (Tian and Cui, 2023); 2) retrieval-based models (Krishna et al., 2023) including BM25 (Robertson et al., 1995) and P-SP (Wieting et al., 2022); 3) classifier models like OpenAI's text classifier (Solaiman et al., 2019) and RADAR (Hu et al., 2023). Additionally, to accurately assess the impact of training data on classifier detectors, we followed OpenAI's approach to train a RoBERTa-base as a comparative baseline on two datasets we employed.

Furthermore, considering the dependence of retrieval models on corpus data, we also evaluated the influence of documents from four different sources on detection performance. The specific details will be elaborated in Section 4.1. In summary, we assessed a total of 6 off-the-shelf detection models and expanded our evaluation to cover 13 experimental settings.

3.2 Adversarial Attacks

To simulate real-world scenarios where users may modify AI-generated text for cheating purposes and also to account for noise in information transmission, we devised 12 perturbation attack methods across four granularities, i.e., document, sentence, word, and character. Some attack strategies have been validated in prior research (Cai and Cui, 2023; Krishna et al., 2023; Shi et al., 2023), while others were the first time to be proposed and explored by our work.

3.2.1 Document-level Perturbations

Paraphrase. We employ the highly effective DIPPER (Krishna et al., 2023) rewriter with the lex=40, order=40, which is the most intensive settings in their paper.

Back-Translation. Leveraging Neural Machine Translation (NMT) models, we chose French as intermediary language, and utilized the translation models from Helsinki-NLP (Tiedemann and Thottingal, 2020).

3.2.2 Sentence-level Perturbations

Sentence Back-Translation. Similar to full-text Back Translation, but randomly selecting sentences as translation windows. Up to 3 pieces were perturbed within a maximum window of 5 sentences.

MLM Prediction. Randomly masking sentences in the original text and replacing them using the BART-large (Lewis et al., 2020) model. Each document underwent random perturbation of 2-5 sentences.

3.2.3 Word-level Perturbations

MLM Prediction for Words. Similar to the sentence MLM prediction, using the BERT-base (Devlin et al., 2019) model to replace random tokens with synonyms. To control text quality, the maximum word perturbation ratio per article did not exceed 20%. This setting is also applied for all our word perturbations.

Adverb Insertion. Randomly inserting a relevant adverb before verbs in the original text.

Spelling Errors. Simulating situations where users misspell words due to ignorance, implemented through a predefined spelling error dict.

Keyboard Typos. Simulating typos during keyboard input, including substitution of nearby characters, swapping adjacent characters, inserting irrelevant characters, and deleting specific characters.

3.2.4 Character-level perturbations.

Word Merging. Simulating scenarios in information transmission contexts where spaces between words are missing. Introducing 3-10 randomly chosen word merging errors per article.

Case of the First Character of a Word. Simulating scenarios where the first character of a word is incorrectly capitalized.

Punctuation Removal. Simulating scenarios where punctuation is lost, removing up to 30% of punctuation marks from the original text.

Space Insertion. Building upon prior work (Cai and Cui, 2023), we control the insertion of spaces to between 5-10 spaces per article.

3.3 Evaluation Metrics

Detection. The prevailing practice in current research is to use the AUC-ROC to comprehensively evaluate the discriminative capability of detectors for AI-generated text (Mitchell et al., 2023; Kirchenbauer et al., 2023a). However, in the real-world deployment of AI text detection, it is essential to select a fixed threshold based on training strategies and internal test data to support subsequent calls. A threshold-independent AUC-ROC metric may no longer accurately reflect the detection performance in practical testings. Therefore, we opted for **F1** and **Accuracy** metrics to assess how accurately input texts are detected as AI-generated content. However, detection rates are heavily influenced by the chosen detection threshold. To address this, we employed the method of maximizing Youden's J statistic to select the optimal threshold for each detection method on a reserved set of 5000 samples. This threshold was then fixed to validate model robustness under perturbations.

	CheckGPT	**HC3**
Train data	720,000*	58,508
Test data	90,000*	25,049
Avg #words	136.68	145.89
Domain	News, Essay, Research	QA

Table 1: Data statistics, where * denotes the data are randomly split with seed 42, and #words denotes the number of words in one sample.

Robustness. In perturbation attack experiments, we considered the **Attack Success Rate (ASR)** as the metric, i.e., the change in AI text detection accuracy after perturbation.

3.4 Benchmarkings

As mentioned earlier, this paper aims to validate the detectability of AI-generated text in real-world scenarios, focusing specifically on the most successful commercial LLMs, the GPT series (Radford et al., 2019; Brown et al., 2020; Ouyang et al., 2022). In contrast to previous work, our attention is solely on data generated by the ChatGPT[4], which was readily accessible to the end users. To simulate two mainstream application scenarios, we selected two datasets: 1) CheckGPT (Liu et al., 2023b) data, which centers around professional writing. The authors generated a dataset of 900k samples encompassing news articles, essays, and scientific research using various prompts. 2) HC3 (Guo et al., 2023), where the authors focused on internet QA scenarios, employing the continuation method to generate ChatGPT response data in fields such as encyclopedia, community, finance, medicine, and open-ended questions. Through these two datasets, we simulate the text detection needs of both professional and ordinary users, with detailed information on the two datasets provided in Table 1.

3.5 Research Questions

Based on off-the-shelf detectors, publicly available data, and black-box perturbations, we propose three research questions to investigate whether current AI-text detectors' development can meet the demands of various real-world application scenarios:

- **RQ1.** What is the detection accuracy when applying current detectors directly to the SoTA LLM-

[4] https://chat.openai.com

Detectors ↓	Professional Writing (CheckGPT)			Informal Writing (HC3)		
	F1	**GPT Acc**	**Human Acc**	**F1**	**GPT Acc**	**Human Acc**
DetectGPT	73.30	71.23	76.81	90.95	92.64	89.16
GPTZero	90.12	86.90	93.95	99.17	98.35	100.0
BM25_{Train}	55.39	45.94	80.02	85.65	86.41	84.97
BM25_{Train^+}	97.78	98.32	97.20	98.49	98.91	98.10
$\text{BM25}_{ShareGPT}$	40.44	29.64	82.98	78.60	77.95	80.06
$\text{BM25}_{ShareGPT^+}$	98.21	98.36	98.04	98.49	98.83	98.18
OpenAI	64.46	55.33	83.62	93.90	91.91	96.24
RADAR	72.23	69.28	77.41	69.36	93.20	26.11
RoBERTa	98.96	98.56	99.36	99.80	99.96	99.64

Table 2: Detection performance of off-the-shelf models on CheckGPT and HC3 datasets. The threshold is determined by maximizing the Youden's J statistic, hence, this detection performance can be considered as the optimal performance of the detector on the current test data.

	OpenAI	**RoBERTa**
GPT-2-Small	97.29	57.85 ↓
GPT-2-Medium	96.96	63.07 ↓
GPT-2-Large	96.74	65.59 ↓
GPT-2-XL	95.35	65.62 ↓
HC3	93.90	99.80
CheckGPT	64.46 ↓	98.96

Table 3: F1 scores for OpenAI detector trained on GPT-2 data and our RoBERTa detector trained on ChatGPT data on both test sets. Lower F1 scores are indicated with a down arrow ↓

generated texts?

- **RQ2.** How does the performance of current detection systems change when facing different perturbations? What are the most effective attack methods?
- **RQ3.** When facing perturbation attacks, can the training strategy or settings of the detection system be adjusted to achieve robust detection?

In the following sections, we will address RQ1 and RQ2 in Section 4 by evaluating the detectors in real-world scenarios. In Section 5, we will explore methods of utilizing perturbation data to provide feasible research directions for future work.

4 Evaluating Detectors in the Wild

4.1 Detectability of the Cutting-Edge AI-Text

We initially validated the performance of three types of AI text detection algorithms on cutting-edge AI text datasets. In our experiments, we considered the HC3 dataset, derived from internet-based QA data, as representative of informal writing scenarios, and the CheckGPT dataset, based on academic paper writing, as representative of professional writing scenarios.

AI-texts are more easily detected in informal writing scenarios. As shown in Table 2, almost all detectors exhibit a higher false positives in professional writing contexts compared to informal writing contexts. Taking the proprietary commercial detection tool GPTZero as an example, it demonstrates minimal false positives in informal writing scenarios, showcasing strong practical utility. However, in CheckGPT, the performance has significantly declined, where the F1 score of GPTZero dropped from 99.2 to 90.1, markedly lower than the finetuned RoBERTa model's 98.9. Surprisingly, the adversarially trained RADAR model exhibited severe false positives in informal writing scenarios, possibly stemming from partial overlap in training data between RADAR and HC3 datasets. This overlap may lead to overfitting to the paraphraser on which the model relies, making it challenging to distinguish human-generated text in that particular domain.

The retrieval method heavily relies on the test samples within the document corpus. As for the retrieval method proposed by Krishna et al. (2023), we conducted ablation experiments on its corpus data. As seen in Table 2, taking the CheckGPT dataset as an example, when utilizing only the training data of the RoBERTa detector or publicly available ShareGPT data, namely BM25_{Train} and

Perturbations ↓		Statistic		Retrieval	Classifier		
		DetectGPT	GPTZero	BM25_{Train^+}	OpenAI	RADAR	RoBERTa
	Origin F1	73.30	90.12	97.78	64.46	72.23	98.96
Doc	Paraphrase	**29.09**	**41.67**	**67.16**	4.79	3.24	**66.24**
	BackTrans	**38.11**	19.05	**43.67**	8.23	0.76	**25.93**
Sent	BackTrans	**30.04**	14.29	12.98	8.23	1.48	12.62
	MLM	14.70	**39.29**	**22.29**	2.36	2.48	12.66
Word	MLM	**68.88**	**83.73**	4.39	19.30	2.12	**75.59**
	AdvInsert	**64.20**	**71.43**	0.00	**31.56**	**25.93**	**47.26**
	Spelling	**70.48**	**62.70**	0.00	**52.62**	**29.92**	**87.10**
	Typos	**70.95**	**36.51**	0.00	**54.25**	**38.31**	**64.68**
Char	Merge	17.82	**23.81**	0.00	**45.83**	2.60	**27.85**
	Case	**44.39**	**80.16**	0.00	**52.22**	14.38	**39.63**
	Punctuation	**23.13**	**25.00**	0.00	**29.76**	0.28	10.11
	SpaceInsert	**35.36**	11.51	0.00	**52.86**	1.60	**21.45**
	Average ASR	42.26	42.43	12.54	30.17	10.26	40.93

Table 4: Attack Success Rates (ASR) of perturbations on the CheckGPT test set. The Retrieval method utilizes training and test data as retrieval documents, and the threshold for all detection algorithms is set to the optimal result on the original test data to simulate real-world model deployment scenarios. A higher ASR indicates a higher proportion of AI-generated text misclassified as human text after perturbation. All data with ASR exceeding **20%** are highlighted in **bold**.

$\text{BM25}_{ShareGPT}$, the retrieval method exhibits the poorest performance, struggling to distinguish AI-text. However, upon incorporating the test data into the retrieval corpus, i.e., BM25_{Train^+} and $\text{BM25}_{ShareGPT^+}$, the accuracy rapidly improves to over 98%, as every machine-generated text now shares identical retrieval results. This performance poses a significant challenge in practical applications, as providers of retrieval detection services must be capable of acquiring and storing all generated results of target LLMs. Efficiency, security, privacy, and other related concerns may limit the widespread adoption of such retrieval detection.

Classifiers-based detectors exhibit poor generalization performance. OpenAI, RADAR, and our fine-tuned RoBERTa model can be considered as three models with the same architecture, with training data quality continually improving. Specifically, each model is trained on data generated by GPT-2, Vicuna, and ChatGPT, respectively. Excluding RADAR's human accuracy on HC3 data, based on GPT detection performance, it is evident that the quality of training data for classifier-based detectors positively correlates with AI text detection performance on cutting-edge AI-generated content. Furthermore, as shown in Table 3, the OpenAI detector performs poorly on ChatGPT data, and the RoBERTa trained on ChatGPT data exhibits suboptimal detection performance on GPT-2 text. These results indicate that neural network-based AI text detectors have limited generalization performance. When the testing data differs in generation methods, model scale, and other aspects from the training data, the model's detection performance sharply declines.

4.2 Effectiveness of Perturbations

We further delve into perturbation scenarios, examining the impact of intentional or unintentional text perturbations generated by users using AI tools on the performance of detectors. Specifically, we investigate the extent of the decline in detection accuracy for AI-generated text across four levels of perturbation granularity.

All detectors exhibit vulnerability to perturbations, even after defense training. From Table 4, it is evident that all detectors show significant misjudgments in the presence of text perturbations, with an average ASR exceeding 10%. Among them, the retrieval and the RADAR methods, which were

	Sim ↑	Flesch	GPT ↑	PPL ↓
Origin	100.0	26.55	8.85	6.18
Paraphrase	80.51	35.91	7.38	9.75
BackTrans	86.23	16.62	6.93	20.18
BackTrans	92.13	25.87	7.91	9.98
MLM	81.90	36.23	4.73	8.71
MLM	67.16	37.34	3.00	29.81
AdvInsert	97.98	20.38	4.29	12.71
Spelling	87.32	29.08	3.49	24.55
Typos	80.38	29.97	3.95	23.14
Merge	98.77	20.43	8.81	8.04
Case	99.81	26.61	7.10	10.06
Punctuation	99.49	19.31	8.24	7.49
SpaceInsert	97.03	30.55	8.18	8.99

Table 5: Comparative results of the quality between original and perturbed text. An upper arrow indicates that higher values are desirable, and vice versa. A higher Flesch value signifies more easily understandable text.

proposed for robustness issues, demonstrate a certain degree of defensive performance. However, when facing specific perturbation attacks, they still exhibit weaker detection capabilities. For instance, the retrieval method, due to its ability to access the original AI-generated text on the test set, shows high defense capabilities against minor text perturbations such as typos and spaces. Meanwhile, its defense capability sharply declines in scenarios involving substantial deviations from the original text, such as rewriting and back translation. Moreover, RADAR, based on paraphraser for adversarial training, exhibits strong defense against larger granularity perturbations. Nevertheless, it inherits the vulnerability of neural network models and performs poorly on perturbations at the word level.

Statistical and classifier-based methods exhibit similar performance when facing perturbations. From the table 4, we observe that, whether it is the commercial GPTZero or other open-source detectors, introducing word-level perturbations to AI-generated articles yields more significant attack results compared to full-text rewriting for these two methods. Simultaneously, the attack performance of word-level perturbation methods seems to be consistent across both groups. For instance, MLM synonym replacements and spelling errors lead to higher perturbation results in both categories of detection methods. This may imply a greater reliance on statistical metrics such as perplexity in the current classifier training. Subsequent work could focus on improving these aspects.

Perturbed texts show significant changes in text quality, readability, or semantic similarity. To assess the changes in semantic similarity and readability introduced by perturbed text, we report four text quality metrics. 1) the semantic similarity between the original and perturbed text, calculated using the P-SP (Wieting et al., 2022) model. 2) the Flesch Reading Ease score, quantifying text readability, with 0 indicating a highly specialized text and 100 representing a fifth-grade level. 3) text quality scores judged by the GPT-3.5-Turbo, ranging from 0 to 10, with 10 being the highest score. The specific prompt will be provided in the Appendix A. 4) perplexity, assessed using the 7B LLaMA-2-base (Touvron et al., 2023) model to evaluate text fluency. From Table 5, it is evident that the success rate of text perturbation is inversely correlated with text quality to a certain extent. Perturbation methods such as Typos can even decrease the GPT score from 8.85 to 3.95.

4.3 Discussions

In summary, for RQ1 and RQ2, we can learn from the results that detection methods based on statistical metrics are generally applicable in informal scenarios. Their zero-shot characteristics endow them with a certain degree of generalization ability. When targeting a certain LLM, training a classifier-based detector, given sufficient training data, proves to be a viable option. However, its generalization capability to other LLMs may be limited. In scenarios with substantial perturbations, retrieval methods exhibit the strongest defense capabilities. Nevertheless, their reliance on the original generated text may constrain their applicability. In future research, proposing more robust detection models or strategies that blend current detection system outcomes would be worthwhile directions.

5 Robustness Enhancement

5.1 Defence Budgets

To further investigate the role of perturbed sample augmentation in enhancing the robustness of AI text detectors, we conducted experiments to evaluate the performance variation of the adversarially trained RoBERTa detector under different perturbation budgets. We define the perturbation budget in two aspects: firstly, the number of augmented

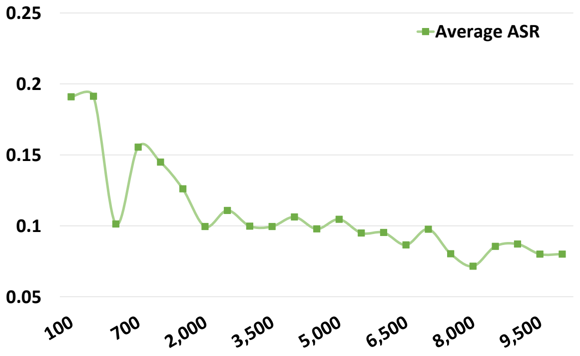

Figure 2: Gradual reduction in average ASR with an increase in the number of perturbed data samples. X-axis represents numbers of each perturbation, while the Y-axis denotes the average ASR on the test set.

	In-domain ASR	OOD ΔASR
Paraphrase	4.82	-29.92
MLM-Sent	8.52	**-65.80**
MLM-Word	7.98	<u>-3.80</u>
Space-Insert	7.90	-11.71

Table 6: Transfer learning results for perturbation attacks. ΔASR represents the reduction in ASR on that target perturbation after training.

samples for each perturbation during adversarial training; secondly, the transferability of different perturbation methods under the same granularity. In this study, we employed the RoBERTa model trained on the CheckGPT dataset as our testing scenario. The results of these two aspects are illustrated in Figure 2 and Table 6.

3,000 Perturbed Samples is All You Need. From Figure 2, we observed the impact of the number of perturbed samples used as augmentation data during the fine-tuning of the RoBERTa model on the average ASR. Our results demonstrate that incorporating a small number of perturbed samples effectively enhances the model's defensive capability against these perturbations. This increasing trend plateaus when the number of perturbed samples reaches around 3000, showing a gradual decline. Ultimately, with the addition of 10,000 perturbed samples (12 perturbation methods, totaling 120,000 augmented data), the average attack success rate decreases from 40.93 to 8.01.

Defense capabilities obtained through transfer learning are not stable. As for transferability, we selected Paraphrase, MLM-Sentence, MLM-Word, and Space Inserting as target perturbations for each of the four granularities. For each experiment, one perturbation was reserved as the target, while the remaining 11 perturbations were used as adversarial training data. We evaluated the detector's defensive capability against the target perturbation post-adversarial training, and the experimental results are presented in Table 6. After fine-tuning, there was a significant decrease in in-domain ASR across the 11 perturbation data, all falling below 9%. However, for out-of-distribution (OOD) target perturbations, notable differences were observed. The MLM-Sentence method, which is more amenable to transfer learning, exhibited a substantial 65.8 decrease in ASR without specific training, with an ASR of only 9.79. In contrast, the more challenging MLM-Word achieved only 3.8 in transfer performance and maintained a high ASR of 43.47 post-training. These results suggest that relying on transfer learning alone to address the robustness of AI text detection is not realistic. Subsequent work should consider a more comprehensive coverage of perturbation attacks.

5.2 Discussions

In summary, for RQ3, concerning text perturbations, augmenting the training data with perturbed samples can enhance the robustness of the detector to some extent. However, there is an upper limit to this enhancement, and the trend levels off after 3,000 perturbed samples. Meanwhile, vanilla transfer learning for defense brings about unstable improvements, contingent on whether the perturbation patterns can be learned from in-domain data.

6 Conclusions

In this paper, we propose two real-world application scenarios for AI text detection: professional writing and informal writing. We evaluate the current SoTA detection performance on these scenarios using three categories of detection methods and six representative models. Furthermore, we introduce and design a novel set of 12 text perturbation methods, demonstrating the vulnerability of current detection models at different granularities. Finally, we apply adversarial learning in the context of perturbed data augmentation, validating the minimum budget and transferability of enhancing classifier models. In future work, we plan to extend our evaluations to include more LLM-generated data, such as Vicuna (Chiang et al., 2023) and Mistral (Jiang et al., 2023).

Limitations

This paper aspires to provide a comprehensive evaluation and analysis of the overall performance of state-of-the-art AI detectors. However, given the challenges posed by multilingual and multimodal applications, our study may not fully cover all aspects. Additionally, it is acknowledged that we cannot encompass all existing text perturbation methods, and the 4 granularities and 12 perturbation tools we constructed might not entirely cover real-world scenarios. Thus, the definition and evaluation of real-world application scenarios in this paper may lack more comprehensive coverage and consideration. Furthermore, this work focuses on adversarial learning for improving the robustness of classifier-based detectors and does not delve into designing more complex and effective defense algorithms. Considering the rapid development of bypass methods for AI-text detectors in reality, more in-depth research on the robustness of AI detection may be a direction for future work.

Ethics Statement

In this paper, we explore the detectability of AI-text in professional and informal writing scenarios and validate the vulnerabilities in current detection systems through perturbation experiments. Our aim is to provide insights and recommendations for the design and training of robust AI detection frameworks in subsequent research. Additionally, we offer robustness validation methods to facilitate the reliable deployment of detection systems for commercial use.

References

Rohan Anil, Andrew M. Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, and et al. 2023. Palm 2 technical report. *CoRR*, abs/2305.10403.

Mikhail J. Atallah, Victor Raskin, Michael Crogan, Christian Hempelmann, Florian Kerschbaum, Dina Mohamed, and Sanket Naik. 2001. Natural language watermarking: Design, analysis, and a proof-of-concept implementation. In *Information Hiding, 4th International Workshop, IHW 2001, Pittsburgh, PA, USA, April 25-27, 2001, Proceedings*, volume 2137 of *Lecture Notes in Computer Science*, pages 185–199. Springer.

Mikhail J. Atallah, Victor Raskin, Christian Hempelmann, Mercan Karahan, Radu Sion, Umut Topkara, and Katrina E. Triezenberg. 2002. Natural language watermarking and tamperproofing. In *Information Hiding, 5th International Workshop, IH 2002, Noordwijkerhout, The Netherlands, October 7-9, 2002, Revised Papers*, volume 2578 of *Lecture Notes in Computer Science*, pages 196–212. Springer.

Ning Bian, Peilin Liu, Xianpei Han, Hongyu Lin, Yaojie Lu, Ben He, and Le Sun. 2023. A drop of ink makes a million think: The spread of false information in large language models. *CoRR*, abs/2305.04812.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, and et al. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Shuyang Cai and Wanyun Cui. 2023. Evade chatgpt detectors via A single space. *CoRR*, abs/2307.02599.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, and et al. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality.

Zhijie Deng, Hongcheng Gao, Yibo Miao, and Hao Zhang. 2023. Efficient detection of llm-generated texts with a bayesian surrogate model. *CoRR*, abs/2305.16617.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

Emilio Ferrara. 2023. Should chatgpt be biased? challenges and risks of bias in large language models. *CoRR*, abs/2304.03738.

Sebastian Gehrmann, Hendrik Strobelt, and Alexander M. Rush. 2019. GLTR: statistical detection and visualization of generated text. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28 - August 2, 2019, Volume 3: System Demonstrations*, pages 111–116. Association for Computational Linguistics.

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

Xiaomeng Hu, Pin-Yu Chen, and Tsung-Yi Ho. 2023. RADAR: robust ai-text detection via adversarial learning. *CoRR*, abs/2307.03838.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de Las Casas, and et al. 2023. Mistral 7b. *CoRR*, abs/2310.06825.

John Kirchenbauer, Jonas Geiping, Yuxin Wen, Jonathan Katz, Ian Miers, and Tom Goldstein. 2023a. A watermark for large language models. In *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pages 17061–17084. PMLR.

John Kirchenbauer, Jonas Geiping, Yuxin Wen, Manli Shu, Khalid Saifullah, Kezhi Kong, Kasun Fernando, Aniruddha Saha, Micah Goldblum, and Tom Goldstein. 2023b. On the reliability of watermarks for large language models. *CoRR*, abs/2306.04634.

Kalpesh Krishna, Yixiao Song, Marzena Karpinska, John Wieting, and Mohit Iyyer. 2023. Paraphrasing evades detectors of ai-generated text, but retrieval is an effective defense. *CoRR*, abs/2303.13408.

Tharindu Kumarage, Paras Sheth, Raha Moraffah, Joshua Garland, and Huan Liu. 2023. How reliable are ai-generated-text detectors? an assessment framework using evasive soft prompts. In *Findings of the Association for Computational Linguistics: EMNLP 2023, Singapore, December 6-10, 2023*, pages 1337–1349. Association for Computational Linguistics.

Thomas Lavergne, Tanguy Urvoy, and François Yvon. 2008. Detecting fake content with relative entropy scoring. In *Proceedings of the ECAI'08 Workshop on Uncovering Plagiarism, Authorship and Social Software Misuse, Patras, Greece, July 22, 2008*, volume 377 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 7871–7880. Association for Computational Linguistics.

Gongbo Liang, Jesus Guerrero, and Izzat Alsmadi. 2023. Mutation-based adversarial attacks on neural text detectors. *CoRR*, abs/2302.05794.

Aiwei Liu, Leyi Pan, Xuming Hu, Shuang Li, Lijie Wen, Irwin King, and Philip S. Yu. 2023a. A private watermark for large language models. *CoRR*, abs/2307.16230.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Zeyan Liu, Zijun Yao, Fengjun Li, and Bo Luo. 2023b. Check me if you can: Detecting chatgpt-generated academic writing using checkgpt. *CoRR*, abs/2306.05524.

Ning Lu, Shengcai Liu, Rui He, Qi Wang, and Ke Tang. 2023. Large language models can be guided to evade ai-generated text detection. *CoRR*, abs/2305.10847.

Dominik Macko, Róbert Móro, Adaku Uchendu, Ivan Srba, Jason Samuel Lucas, Michiharu Yamashita, Nafis Irtiza Tripto, Dongwon Lee, Jakub Simko, and Mária Bieliková. 2024. Authorship obfuscation in multilingual machine-generated text detection. *CoRR*, abs/2401.07867.

Nick McKenna, Tianyi Li, Liang Cheng, Mohammad Javad Hosseini, Mark Johnson, and Mark Steedman. 2023. Sources of hallucination by large language models on inference tasks. *CoRR*, abs/2305.14552.

Fatemehsadat Mireshghallah, Justus Mattern, Sicun Gao, Reza Shokri, and Taylor Berg-Kirkpatrick. 2023. Smaller language models are better black-box machine-generated text detectors. *CoRR*, abs/2305.09859.

Eric Mitchell, Yoonho Lee, Alexander Khazatsky, Christopher D. Manning, and Chelsea Finn. 2023. Detectgpt: Zero-shot machine-generated text detection using probability curvature. In *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pages 24950–24962. PMLR.

OpenAI. 2023. GPT-4 technical report. *CoRR*, abs/2303.08774.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, and et al. 2022. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Jie Ren, Han Xu, Yiding Liu, Yingqian Cui, Shuaiqiang Wang, Dawei Yin, and Jiliang Tang. 2023. A robust semantics-based watermark for large language model against paraphrasing. *CoRR*, abs/2311.08721.

Stephen E. Robertson, Steve Walker, Micheline Hancock-Beaulieu, Mike Gatford, and A. Payne. 1995. Okapi at TREC-4. In *Proceedings of The Fourth Text REtrieval Conference, TREC 1995, Gaithersburg, Maryland, USA, November 1-3, 1995*, volume 500-236 of *NIST Special Publication*. National Institute of Standards and Technology (NIST).

Vinu Sankar Sadasivan, Aounon Kumar, Sriram Balasubramanian, Wenxiao Wang, and Soheil Feizi. 2023. Can ai-generated text be reliably detected? *CoRR*, abs/2303.11156.

Zhouxing Shi, Yihan Wang, Fan Yin, Xiangning Chen, Kai-Wei Chang, and Cho-Jui Hsieh. 2023. Red teaming language model detectors with language models. *CoRR*, abs/2305.19713.

Irene Solaiman, Miles Brundage, Jack Clark, Amanda Askell, Ariel Herbert-Voss, Jeff Wu, Alec Radford, and Jasmine Wang. 2019. Release strategies and the social impacts of language models. *CoRR*, abs/1908.09203.

Jinyan Su, Terry Yue Zhuo, Di Wang, and Preslav Nakov. 2023. Detectllm: Leveraging log rank information for zero-shot detection of machine-generated text. *CoRR*, abs/2306.05540.

Edward Tian and Alexander Cui. 2023. Gptzero: Towards detection of ai-generated text using zero-shot and supervised methods".

Jörg Tiedemann and Santhosh Thottingal. 2020. OPUS-MT - building open translation services for the world. In *Proceedings of the 22nd Annual Conference of the European Association for Machine Translation, EAMT 2020, Lisboa, Portugal, November 3-5, 2020*, pages 479–480. European Association for Machine Translation.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, and et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *CoRR*, abs/2307.09288.

Nafis Irtiza Tripto, Saranya Venkatraman, Dominik Macko, Róbert Móro, Ivan Srba, Adaku Uchendu, Thai Le, and Dongwon Lee. 2023. A ship of theseus: Curious cases of paraphrasing in llm-generated texts. *CoRR*, abs/2311.08374.

Adaku Uchendu, Thai Le, Kai Shu, and Dongwon Lee. 2020. Authorship attribution for neural text generation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 8384–8395. Association for Computational Linguistics.

John Wieting, Kevin Gimpel, Graham Neubig, and Taylor Berg-Kirkpatrick. 2022. Paraphrastic representations at scale. In *Proceedings of the The 2022 Conference on Empirical Methods in Natural Language Processing, EMNLP 2022 - System Demonstrations, Abu Dhabi, UAE, December 7-11, 2022*, pages 379–388. Association for Computational Linguistics.

A GPT Judgement Prompt

Following the GPT judgement method proposed by Hu et al. (2023), we conducted scoring experiments on 2,503 AI-generated texts from the Check-GPT dataset using the GPT-3.5-Turbo API. The prompts for both original and perturbed texts were as follows: *You are given an array of 13 sentences. Please rate these sentences and reply with an array of scores assigned to these sentences. Each score is on a scale from 1 to 10, the higher the score, the sentence is written more like a human. Your reply example: [2,2,2,2,2,2,2,2,2,2,2,2,2].*

B Perturbation Samples

In this section, we show the original AI-generated sample and all perturbed texts for a random sample.

Origin. *In this paper, we explore grand unified theories that utilize an SU(5)xSU(5) gauge group. Our focus is on preventing fast proton decay through a combination of small triplet couplings and a large triplet mass, achieved through discrete symmetries. We demonstrate that in many of our models, the GUT scale (M_{GUT}) occurs naturally due to a balance of higher dimension terms and soft supersymmetry breaking masses. Our findings include intriguing patterns in quark and lepton masses, and we examine the differences between grand unified theories and string unification.*

Paraphrase. *Here we look at Grand Unified Theories which make use of the SU(5)xSU(5) gauge group, concentrating on avoiding fast proton decay by the use of small triplet couplings and large triplet masses, obtained through discrete symmetries. We show that in many of our models, the GUT scale (M_{GUT}) arises naturally from a balance between higher dimension terms and the soft breaking of supersymmetry. We find some unusual patterns in the quark and leptoon masses, and we also discuss the differences between the GUT approach and the string approach to unified theories.*

Back Translation. *In this paper, we examine the main unified theories that use a SU(5)xSU(5) gauge group. We focus on preventing the rapid decomposition of protons by a combination of small triplet couplings and large triplet mass obtained by discrete symmetries. We show that in many of our models, the GUT (M_{GUT}) scale occurs naturally due to a balance of upper dimensional terms and soft supersymmetry break masses.*

Back Translation Sentence. *In this paper, we examine the main unified theories that use a SU(5)xSU(5) gauge group. We focus on preventing the rapid decomposition of protons by a combination of small triplet couplings and large triplet mass obtained by discrete symmetries. We show that in many of our models, the GUT scale (M_{GUT}) occurs naturally due to a balance of the upper dimension terms and the soft supersymmetry break masses.*

MLM Prediction for Sentence. *Abstract We demonstrate that in many of our models, the GUT scale (M_{GUT}) occurs naturally due to a balance of higher dimension terms and soft supersymmetry breaking masses. In this paper, we discuss the role of string unification in the Evolution of the Proton. Abstract Our focus is on string unification and its role in proton evolution. Our findings include the following: String Unification in Proton Evolution and its Role in the Universe*

MLM Prediction for Word. *In this paper, we read most unified theories that utilize an SU(5)xSU(5) conclusion conclusion. Our focus is on read fast proton decay as a combination of small triplet couplings and a most triplet mass, achieved as discrete symmetries. their demonstrate that in many of our models, the GUT scale (conclusion }) occurs naturally due to a conclusion of higher dimension terms and soft conclusion breaking conclusion. their conclusion include intriguing patterns in conclusion and lepton conclusion, and we examine the conclusion between grand unified theories and conclusion unification.*

Adverb Insertion. *In this paper, we rarely explore grand emily unified theories that utilize an SU(5)xSU(5) gauge group. Our focus overseas is on preventing fast proton decay through a combination of small triplet couplings and a large triplet mass, less achieved through discrete symmetries. We gradually demonstrate that in many of our models, the GUT scale (M_{GUT}) occurs naturally due to a balance of higher dimension terms and soft supersymmetry breaking masses. Our findings probably include intriguing patterns in quark and lepton masses, and we examine the differences between grand unified theories and string unification.*

Spelling Errors. *In this paperl, we explove grand unified theories that utilize an SU(5)xSU(5) gauge groop. Our foccus is on preventing fast proton decay through a combination of sall triplet couplings and a larg triplet mess, achieved through discrete symmetries. Why demonstatrate thate in mary of ours models, the GUT scale (M_{GUT}) occurs naturally dur take a balance of hight dimension terms and soft supersymmetry breking masses. Our findinds include intriguing patterns in quark and lepton masses, and wie examine the differeces between grand unified theories and string unification.*

Keyboard Typos. *In this papetr, we explore grand unifeid theroies that utlilize an SU(5xSU(5) gage group. Our focus is on prventing fast proton deacy through a combination of small triplet couplings and a laege triplet mass, achieved through discrete sybmetries. We demonstrate thaft in many of our models, the GUT scale (M_{GUT}) occurs naturally due to a balance of higehr dimension tearms and sot supersymmetry breakinvg masses. Our findings include intriguing patterns in quark and lepton masses, and we eamine the differences between grand unified theories and string unification.*

Word Merging. *In this paper, we exploregrand unified theories that utilize an SU(5)xSU(5) gauge group.Our focus is on preventing fast proton decay through a combination of small triplet couplings and a large triplet mass, achieved throughdiscrete symmetries. We demonstrate that in many of our models, the GUT scale (M_{GUT}) occurs naturally due to a balance of higher dimension terms and soft supersymmetry breaking masses. Our findings include intriguing patterns in quark and lepton masses, and we examine the differences between grand unified theories and string unification.*

Case of the First Character of a Word. *In this paper, we explore grand Unified theories That Utilize an SU(5)xSU(5) gauge group. Our focus is on Preventing fast proton decay Through a combination of small Triplet couplings and a large triplet mass, achieved through discrete symmetries. we demonstrate That in Many of our Models, the gUT scale (m_{GUT}) occurs naturally Due To a balance of higher dimension Terms and Soft supersymmetry breaking masses. Our Findings include intriguing patterns in quark and lepton masses, and we examine the differences between grand unified theories and String Unification.*

Punctuation Removal. *In this paper, we explore grand unified theories that utilize an SU(5)xSU(5 gauge group. Our focus is on preventing fast proton decay through a combination of small triplet couplings and a large triplet mass, achieved through discrete symmetries. We demonstrate that in many of our models, the GUT scale (M_{GUT} occurs natu-*

rally due to a balance of higher dimension terms and soft supersymmetry breaking masses. Our findings include intriguing patterns in quark and lepton masses, and we examine the differences between grand unified theories and string unification

Space Insertion. *In this paper, we explore grand unified theories that utilize an SU(5)xSU(5) gauge group. Our focus is on preventing fast proton decay through a combination of small triplet couplings and a large triplet mass, achieved through discrete symmetries. We demonstrate that in many of our models, the GUT scale (M_{GUT}) occurs naturally due to a balance of higher dimension terms and soft supersymmetry breaking masses. Our findings in clude intriguing patterns in q uark and lepton masses, and we examine the differences between grand unified theories and string un ification.*