Benchmarking AI Text Detection: Assessing Detectors Against New Datasets, Evasion Tactics, and Enhanced LLMs

Shushanta Pudasaini and Luis Miralles-Pechuán and Marisa Llorens Salvador
School of Computer Science
Technological University Dublin
Dublin, Ireland

David Lillis
University College Dublin
Dublin, Ireland

Abstract

The rapid advancement of Large Language Models (LLMs), such as GPT-4, has sparked concerns regarding academic misconduct, misinformation, and the erosion of originality. Despite the growing number of AI detection tools, their effectiveness is often undermined by sophisticated evasion tactics and the continuous evolution of LLMs. This research benchmarks the performance of leading AI detectors, including OpenAI Detector, RADAR, and ArguGPT, across various text domains, evaded content, and text generated by cutting-edge LLMs. Our experiments reveal that current detection models show considerable unreliability in real-world scenarios, mainly when tested against diverse data domains and novel evasion strategies. The study underscores the need for enhanced robustness in detection systems and provides valuable insights into areas of improvement for these models. Additionally, this work lays the groundwork for future research by offering a comprehensive evaluation of existing detectors under challenging conditions, fostering a deeper understanding of their limitations. The experimental code and datasets are publicly available for further benchmarking on Github.

Keywords: Large Language Models, Evasion Strategies, Cross Domain Testing, AI-Generated text detection.

1 Introduction

LLMs such as GPT-3 have achieved strong performance on several tasks that require on-the-fly reasoning or domain adaptation, such as translation, question answering, and representing text in an intelligent, presentable form (Brown, 2020). In addition, more advanced LLMs such as GPT-4 have achieved human-level performance on academic benchmarks, such as passing a simulated bar examination by scoring around the top 10% test takers (Achiam et al., 2023). On top of that, OpenAI released OpenAI o1-preview, a new series of reasoning models claiming that these models perform on the level of PhD students for challenging benchmark tasks such as Graduate-Level Google-Proof Q&A (GPQA) benchmark in subjects like physics, biology and chemistry (Zhong, 2024). These events highlight the exceptional capabilities of LLMs in academia and their fast-paced evolution.

With exceptional capabilities, LLMs have also brought several threats like academic misconduct (Pudasaini et al., 2024), such as students submitting assignments, passing examinations, misinformation (Liu et al., 2024), lack of creativity (Zhao et al., 2024), and other several ethical concerns (Yan et al., 2024). To overcome these threats, efficient detection of texts generated from LLMs is necessary.

Much research has been done to build highly effective and robust LLM-generated text detection algorithms. A training-based classifier, zero-shot detection, watermarking, and adversarial learning approach are some approaches used to create models that detect LLM-generated text. The training-based classifier approaches treat the problem as a binary classification problem, and labelled datasets are trained on several algorithms to build models such as ArguGPT (Liu et al., 2023), Ghostbuster (Verma et al., 2023) and roberta-base-openai-detector (Solaiman et al., 2019).

A zero-shot learning approach that allows a model to handle tasks it hasn't been explicitly trained for by using its existing general knowledge has also been used to develop detectors such as Binoculars (Hans et al., 2024), Fast-DetectGPT (Bao et al., 2023), DNA-GPT (Yang et al., 2023), and GLTR (Gehrmann et al., 2019).

Similarly, watermarking techniques by which subtle, identifiable patterns are embedded into the generated content, making it easier to recognise as AI-generated, has resulted in models such as POSTMARK (Chang et al., 2024), Waterfall (Lau

Proceedings of the 1st Workshop on GenAI Content Detection (GenAIDetect), pages 68–77
January 19, 2025.

et al., 2024), WaterJudge (Molenda et al., 2024), and WaterMax (Giboulot and Teddy, 2024).

Other approaches, such as adversarial learning, which enhances the robustness of LLM-generated text detection algorithms by exposing the detector to an evasion example while training, have been introduced, resulting in models like RADAR (Hu et al., 2023) and OUTFOX (Koike et al., 2024).

Along with developing LLM detectors, several evasion techniques, such as paraphrasing and synonym replacement, have also been developed. These evasion techniques are applied after generating text from LLMs so that the detector cannot identify the text as AI-generated. These evasion techniques may be as simple as adding a single space randomly before a comma (Cai and Cui, 2023), deleting an article randomly in a sentence, inserting misspellings randomly in a sentence (Antoun et al., 2023), and replacing some random characters with homoglyph characters (Kirchenbauer et al., 2023).

There are also more complex techniques such as paraphrasing(Krishna et al., 2024), word substitution (Peng et al., 2024), sentence substitution (Peng et al., 2024) and prompting (Wang et al., 2024)(using instructions to generate human-written text while generating text) also have been successful in fooling AI detectors. Thus, it becomes crucial to benchmark existing open-source, state-of-the-art AI detectors against these evasion strategies.

One gap in developing efficient AI detectors is the reliance on subsets of a single dataset for training and testing, compromising the models' robustness (Sadasivan et al., 2023). Such detectors claim high accuracy. However, they typically fail when tested in real settings with very different data from the training and testing set. Out-of-distribution testing of the existing open-source state-of-the-art detectors is crucial (Dugan et al., 2024).

In addition, it is essential to benchmark the existing AI detectors with new generators in the space. The rapid growth in the development of new LLMs with exceptional learning capabilities, along with the increasing number of parameters, has brought up new concepts such as reasoning (Huang and Chang, 2022), coherent and cohesive long-form text generation (Cho et al., 2018), and multilingual and cross-domain capabilities (Chua et al., 2024). This leads to the research question of whether the existing state-of-the-art AI detectors are up to date and capable of detecting text from new LLMs with such capabilities or not. Thus, benchmarking such detectors against recent powerful LLMs is vital.

This paper's main contribution is benchmarking existing AI detectors against different datasets (text from various domains, text created using evasion techniques, and text generated by the latest LLMs such as GPT4 o from OpenAI and Command R+ from Cohere). This benchmarking allows for an in-depth analysis of the different detectors' performance in the context of their general approach to AI-generated text detection and the various types of datasets used.

The rest of the paper is organised as follows. Section 2 discusses the previous benchmarking research done for LLM-generated text detection. Section 3 explains the methodology used to perform the benchmarking experiment. Section 4 highlights the results obtained from the experiment. Section 5 discusses the analysis of the results obtained. Section 7 finally presents the conclusions obtained from the experiment.

2 Literature Review

As the race for the development of robust AI detectors and the development of evasion strategies to fool AI detectors goes on, along with the development of even more powerful LLMs, research has been conducted to test the efficiency of existing LLM-generated text detection algorithms developed so far.

Initially, the benchmarking experiments used human-written and AI-generated text with no further modifications. Chaka (Chaka, 2024) did a comprehensive review of 17 published articles on testing AI detectors. The author found that the machine-generated text used in testing in those research papers was from ChatGPT-3.5 and ChatGPT-4 (Chaka, 2024). Madelyn A. et al. (Flitcroft et al., 2024) tested three AI detector tools, OpenAI's AI Classifer, Content at Scale, and Originality.AI, on human-written scientific and AI-generated articles, which are not modified. The tool Originality.ai achieved 100% accuracies in this testing (Flitcroft et al., 2024). However, people may not just copy-paste the text entirely from LLMs and may try to modify the text.

Weber-Wulff et al. (Weber-Wulff et al., 2023) tested AI detection tools on three types of AI-generated text: AI-generated text, AI-generated text with subsequent human edits, and AI-generated text with subsequent machine paraphrasing and found those detectors were biased in classifying AI-generated text as human-written.

Similarly, Elkhatat et al. (Elkhatat et al., 2023) also tested five AI detection tools on human-written control responses and saw an increase in false positives in the case of human-written control responses. These experiments against types of text observations (human-written, ai-generated and human editing on AI-generated texts) show that AI detectors are easily evaded with few further human edits on AI-generated texts.

When evasion techniques are applied to AI-generated texts, AI detectors fail to perform well. Krishna et al. (Krishna et al., 2023) developed an 11B parameter paraphrase generation model called DIPPER, which successfully evaded detectors such as watermarking, GPTZero, DetectGPT, and OpenAI's text classifier and further proposed a retrieval method which detected 80% to 97% of paraphrased generations across various settings, while only 1% of human-written text was mistakenly flagged as AI-generated. However, after 4 months, Sadasivan et al. (Sadasivan et al., 2023) again introduced the recursive paraphrasing attack, which degraded the accuracy of several watermarking-based, zero-shot-based, neural-network-based and retrieval-based detectors. The adversarial learning approach has been introduced to develop recent AI-generated text detectors such as RADAR (Hu et al., 2023) and OUTFOX (Koike et al., 2024). There seems to be a gap in benchmarking these recently developed models on different scenarios.

The challenge with building AI detectors is these detectors need to be able to generalise on unseen domain text (Wang et al., 2023). For example, existing AI detectors may fail when tested on text generated from recent LLMs released after the AI detector's release. Dugan et al. (Dugan et al., 2024) tested neural and metrics-based AI detectors on AI-generated texts from 11 different LLMs. They found that the performance of these AI detectors varies according to the LLMs used for generating the text. However, with the release of even more powerful LLMs with reasoning capabilities like GPT-4o, it remains to be seen how these detection algorithms perform on the text generated from such new LLMs.

3 Methodology

The benchmarking of existing open-source LLM-generated text detection algorithms was done on three significant aspects: out-of-distribution along with multiple domain testing, evasion applied dataset testing and new LLM-generated text dataset testing.

The data flow in the benchmarking experiment has been represented in Fig 1. Initially, sampling was performed from two datasets, i.e., the M4 dataset and the HC3 dataset, resulting in a data subset of 3,000 AI-generated text observations and 3,000 AI-generated text observations. The 3,000 AI-generated observations were further edited using the six evasion strategies, and 3,000 evasion-applied AI-generated text observations were created for each evasion strategy. Additionally, 1,000 new AI-generated text observations were generated from each of the recent LLMs, i.e. GPT-4 o and Command R plus, using the same prompt used to create them previously.

Because of computational constraints, the number of observations was limited. The complete testing dataset representing multiple datasets, multiple evasion strategies, and multiple generators was passed to the AI-generated text detection algorithms, and the benchmarking results were obtained.

3.1 Datasets Used

Data samples from different datasets and domains were required for out-of-distribution and multidomain testing of the AI detectors. Two different datasets were used; details of the datasets are explained below.

- **M4 Dataset**: M4 is a large-scale dataset for **M**achine-generated text detection which included data samples from **M**ultiple languages, **M**ultiple domains and **M**ultiple LLMs (Wang et al., 2023). Sampling was applied concerning the domain of the data for multidomain analysis. The subset considered 1,000 observations representing text from multiple domains such as Arxiv, Wikipedia, and Reddit. Thus, combining these data from multiple sources resulted in 3,000 human-written observations and respective 3,000 AI-generated observations.

- **HC3 Dataset**: The Human ChatGPT Comparision Corpus(HC3) is built from tens of thousands of comparison responses from ChatGPT and human experts in financial, medical, legal, and open-domain (Guo et al., 2023). Sampling was done randomly from the dataset, resulting in 3,000 human-written and 3,000 AI-generated texts.

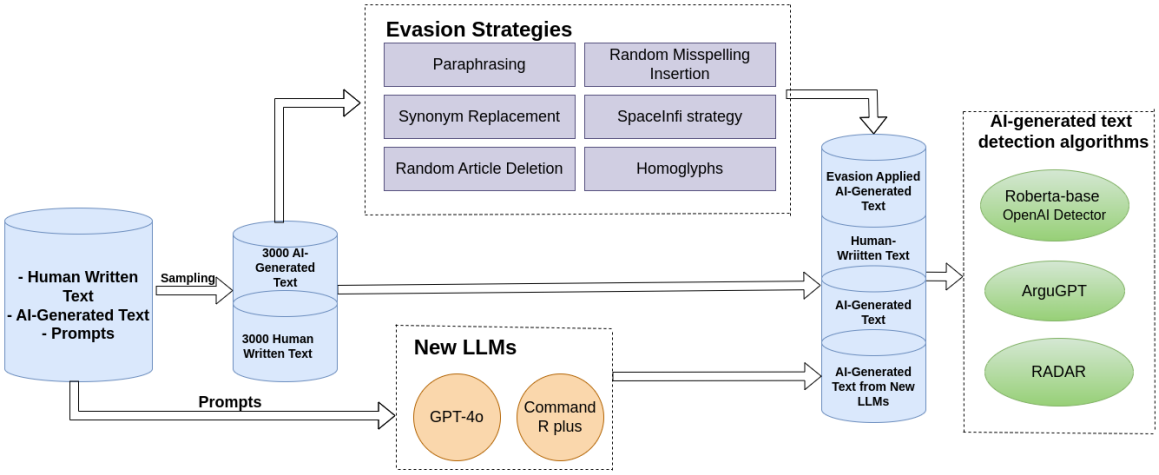

Figure 1: Diagram representing the process, the components used, and the data flow of the benchmarking experiment.

3.2 Evasion Strategies

Several evasion techniques have been used to try to fool AI detectors. This paper used six evasion techniques to modify the selected AI-generated text subsets. The evasion strategies applied are discussed below.

- **Paraphrasing**: Paraphrasing is the most commonly used technique to fool AI detectors (Sadasivan et al., 2023; Krishna et al., 2024). Parrot paraphrase, which uses LLMs to paraphrase a given text, was used to paraphrase each text and generate the paraphrased AI-generated samples (Damodaran, 2021).
- **Synonym Replacement**: A random word from each sentence in the given text was replaced with the word's synonym retrieved from the Natural Language Toolkit (NLTK) (Bird et al., 2009) Synset and replaced in the sentence.
- **Misspelling**: A random word from a sentence is replaced with a misspelt version of the word. Further insertion of random misspellings is also an effective evasion strategy for BERT-based AI detectors (Antoun et al., 2023).
- **Article Deletion**: This technique removes a random article from a sentence from the AI-generated text (Odri and Yoon, 2023).
- **SpaceInfi Strategy**: Cai et al. (Cai and Cui, 2023) introduced the SpaceInfi strategy in which a space is inserted before a random comma in the AI-generated text to fool the detectors.
- **Homoglyph Attack**: Unicode characters that look very similar to the existing characters in the sentence are referred to as homoglyphs. Kirchenbauer et al. (Kirchenbauer et al., 2023) mentioned that homoglyph changing the tokenization process affects the prediction of AI detectors. We applied homoglyphs to 50% characters in the AI-generated text, as suggested by Antoun et al. (Antoun et al., 2023).

Similarly, text generated from recently released LLMs, such as OpenAI's GPT-4o model (Islam and Moushi, 2024) and the Command R plus model from the open-source LLM community Cohere, was included in the whole testing set for benchmarking the detectors against recent LLMs. The test data size was limited to 1,000 observations for each model due to the limitations of OpenAI API and Cohere API credits.

3.3 LLM-generated text detection algorithms

Several algorithms have been developed to solve LLM-generated text detection following different approaches such as watermarking, zero-shot, training-based classifiers, fine-tuning LLMs, adversarial learning methods, and treating another LLM as a detector (Wu et al., 2023). The detection algorithms used are described below.

- **ArguGPT**: ArguGPT is a RoBERTa base classifier trained on a corpus of 4,038 argumentative essays generated by 7 GPT models. It achieved 90% accuracy in document, paragraph, and sentence level classification (Liu et al., 2023). This model was chosen for the

experiment because the dataset used to train was composed of multiple sources (in-class or homework exercises, TOEFL writing tasks, GRE writing tasks), multiple generators and on multiple levels.

- **RADAR**: Robust AI-text Detector via Adversarial Learning (RADAR) is built by joint adversarial training of a paraphraser model and a detector model (Hu et al., 2023). RADAR model claims to outperform existing AI-detection methods, mainly when paraphrasing is applied as an evasion strategy (Hu et al., 2023). This model was chosen because it is trained by using an adversarial approach.

- **OpenAI Detector**: Openai-detector is an open-source language model based on transformers 125 billion parameters released by OpenAI (Solaiman et al., 2019). This model was chosen because it was trained using the outputs of the 1.5B GPT-2 model.

4 Experiments and Results

This section presents the results of the benchmarking experiment on different edge cases such as multiple domain data, multiple evasion applied data and multiple LLMs generated data.

The testing of a model developed on another dataset rather than testing on the test set of its dataset is referred to as out-of-distribution testing. The benchmarking dataset used in this experiment was not used to train and test these models. Such out-of-distribution testing of the AI detectors was done by passing the text sample observations to the models. Prediction probability was obtained for each of the predictions. We set the threshold value to 0.5 to classify it as human-written or AI-generated. Out-of-distribution set testing was performed with text observations from the M4 and HC3 datasets. We calculated the accuracy of the binary classification evaluation metrics, F1-score, false negatives, false positives, precision, and recall. The results obtained are shown in Table 1.

The detectors were benchmarked on different evasion strategies. The results from the benchmarking against the evasion strategies are shown in Table 2. The results from benchmarking the detectors against recent LLMs are shown in Table 3. The baseline dataset was created using the GPT 3.5 Turbo model.

5 Discussion and Analysis

This section discusses and analyses the experiment's results to report key insights. The discussion is organised according to the edge cases, i.e., out-of-distribution and Multi-Domain, Evasion Strategies, and recent LLMs. We mainly analyse the False Negatives (FN), which gives the number of AI-generated text observations that were misclassified as human-written, representing the inefficiency of the AI-detector tools.

5.1 Analysis of out-of-distribution and Multi Domain Benchmarking

Existing AI detectors suffer from data drift while testing on text observations from another dataset or another domain. From the results in Table 1, we can see that the evaluation metrics of these AI detector models are very different when tested on various datasets with the same number of testing observations. Differences of 8.9 % accuracy in the OpenAI detector model, 4.95 % in the RADAR model and 5.21 % in the ArguGPT model were reported while tested on the M4 and HC3 datasets. This indicates that these models are not resistant to data drift over new domains and settings, i.e. sentence, paragraph or document-level text.

The detectors cannot be relied upon to be used in real settings. From Table 1, we can see a big difference in terms of False Negatives (AI-generated samples misclassified as human-written) and False Positives (human-written samples that were misclassified as AI-generated) when the detectors are tested on text observation they are not familiar with as the models tested were tested on entirely different data. We can see an increase in false positives: 1 to 43 in OpenAI detector, 175 to 553 in RADAR and 155 to 472 in ArguGPT. This is critical because, with such an increase in FPs, it will be challenging to use the detectors in real settings as many submitting authentic human-written text will be flagged as AI-generated. Similarly, the increase in false negatives (FN) in the OpenAI detector from 113 to 607 suggests that the detector fails to detect AI-generated text when data observations are taken from different domains.

5.2 Analysis of Benchmarking on Evasion Strategies

OpenAI detector performs poorly when evasion techniques are applied. Table 2 shows that the OpenAI detector is very poor at detecting AI-generated

Model	Dataset	Acc.(%)	F1	FN	FP	Prec	Rec
OpenAI Detector	M4	89.17	0.8804	607	43	0.982	0.797
	HC3	98.09	0.9806	113	1	0.999	0.962
RADAR	M4	94.13	0.9413	177	175	0.943	0.941
	HC3	89.18	0.8994	96	553	0.84	0.968
ArguGPT	M4	92	0.9257	8	472	0.863	0.997
	HC3	97.41	0.9748	0	155	0.951	1

Table 1: Benchmarking results of AI Detectors tested on two different subsets of M4 and HC3 with 6,000 samples each. All observations are texts in which no evasion strategies have been applied. Evaluation Accuracy (Acc.), F1-Score (F1), False Negatives (FN), False Positives (FP), Prec (Precision), and Rec (Recall) were obtained from out-of-distribution testing of models.

texts where evasion techniques have been applied. The baseline(non-evasive) experiment was conducted on the observation where no evasion techniques were applied, which resulted in 89.17% accuracy in the M4 dataset and 98.1% accuracy in the HC3 dataset. Using that as a reference point, we can see the decrease in the model's accuracy under different evasion techniques. The decreasing accuracy was found to be similar to both datasets. This is mainly because the OpenAI detector was trained using the texts generated from the GPT-2 model on which no modifications were applied to the generated text, and the training samples do not cover multiple domains and multiple generation settings on the GPT-2 model (Solaiman et al., 2019).

RADAR model could effectively identify the AI-generated text on which further paraphrasing and synonym replacement evasion techniques were applied. According to the results in Table 2, we can see that the RADAR model performs even better in evasion techniques such as paraphrasing and synonym replacement. The RADAR model is trained jointly with a detector and paraphrased with an adversarial approach. We observed a decrease in false negatives even after evasion had been applied. However, the RADAR model still behaves poorly under other evasion techniques, such as article deletion and homoglyphs. Thus, we can conclude that adversarial learning methods incorporating several evasion strategies rather than a single evasion strategy (paraphrasing in RADAR) could lead to developing an AI detector resistant to any evasion strategies.

The performance of the ArguGPT model could be better, with some evasion strategies such as homoglyphs and misspellings. The results of the ArguGPT model from Table 2 The ArguGPT model worked well when no evasive techniques were applied (8 false negatives among 3,000 observations in the M4 data set and 0 false negatives among 3,000 observations in the HC3 dataset). However, false negatives started to increase when evasion techniques were applied. This is also mainly because the data in the model's training did not contain such observations where evasion techniques have been applied further.

5.3 Analysis of Benchmarking on Recent LLMs

During the test of observations from the latest LLMs, the performance of existing AI detectors was degraded. However, ArguGPT performed better than other models. The results in Table 3 show that models such as OpenAI Detector and RADAR fail faster than the ArguGPT model while testing text generated from recent LLMs: GPT-4o and Command R plus. The baseline represents the result when the text was generated from the GPT-3.5 model. The number of false negatives increased from 197 to 509 while testing text generated from Command R plus and 985 while testing text generated from the GPT-4o model, indicating that the OpenAI detector cannot perform well on text detection from recent LLMs.

Similarly, in the case of the RADAR model, false negatives were increased from 52 to 299 while testing on the text generated from the Command R plus model and to 777 while testing on the text generated from the GPT-4o model, indicating the RADAR model also does not perform well on text detection from recent LLMs. However, the ArguGPT model saw only a slight increase in false negatives (3 in GPT-3.5, 23 in Command R plus and 70 in GPT-4o). This behaviour can be attributed to the ArguGPT model being trained using the text generated from 7 GPT models (Liu et al., 2023). The GPT4o model could also fool the AI detectors more than other LLMs. We believe the

Model	Dataset	Experiment Type	Acc. (%)	F1	FN (Out of 3,000)
OpenAI Detector	M4 Dataset	non-evasive	89.17	0.8804	607
		evasion whitespace	79.63	0.7488	1,179
		evasion removed articles	51.95	0.0999	2,840
		evasion misspell text	51.77	0.0934	2,851
		evasion homoglyph	61.53	0.3891	2,265
		evasion synonym replaced	74.55	0.6651	1,484
		evasion paraphrase	68.67	0.553	1,837
	HC3 Dataset	non-evasive	98.1	0.9806	113
		evasion whitespace	92.02	0.9133	478
		evasion removed articles	60.33	0.3425	2379
		evasion misspell text	51.12	0.0443	2932
		evasion homoglyph	50.6	0.6693	1983
		evasion synonym replaced	83.43	0.8015	993
		evasion paraphrase	55.76	0.6932	1840
RADAR	M4 Dataset	non-evasive	94.13	0.9413	177
		evasion whitespace	95.10	0.9515	119
		evasion removed articles	71.47	0.6309	1,537
		evasion misspell text	47.10	0.0006	2,999
		evasion homoglyph	47.15	0.0025	2,996
		evasion synonym replaced	94.27	0.9427	169
		evasion paraphrase	95.70	0.9576	83
	HC3 Dataset	non-evasive	89.18	0.8995	96
		evasion whitespace	89.82	0.9059	58
		evasion removed articles	82.06	0.8215	523
		evasion misspell text	41.70	0.0305	2,945
		evasion homoglyph	40.92	0.0045	2,991
		evasion synonym replaced	88.98	0.8974	108
		evasion paraphrase	90.22	0.9100	34
ArguGPT	M4 Dataset	non-evasive	92	0.9257	8
		evasion whitespace	91.93	0.9251	12
		evasion removed articles	89.20	0.8971	176
		evasion misspell text	42.13	0.0000	3,000
		evasion homoglyph	42.13	0.0000	3,000
		evasion synonym replaced	91.87	0.9244	16
		evasion paraphrase	90.55	0.9111	95
	HC3 Dataset	non-evasive	97.42	0.9748	0
		evasion whitespace	97.40	0.9747	1
		evasion removed articles	97.23	0.9730	11
		evasion misspell text	47.42	0.0000	3,000
		evasion homoglyph	47.42	0.0000	2,999
		evasion synonym replaced	97.37	0.9743	3
		evasion paraphrase	96.58	0.9664	50

Table 2: Benchmarking on different evasion strategies across models on 6000 samples for each experiment. Evaluation Accuracy (Acc.), F1-Score (F1), and False Negatives (FN) were obtained under different evasion strategies for different datasets while testing on OpenAI Detector, RADAR, and ArguGPT models.

high reasoning capabilities of the GPT-4o model can explain these results (Chen et al., 2024).

Models	LLM tested against	Acc. (%)	F1-Score	FN (Out of 1000)
OpenAI Detector	baseline	89.9	0.8883	197
	Command R plus	71.94	0.5569	509
	GPT-4o	50.5	0.0294	985
RADAR	baseline	95.65	0.9561	52
	Command R plus	81.77	0.7614	299
	GPT-4o	59.4	0.3545	777
ArguGPT	baseline	95.5	0.9568	3
	Command R plus	94	0.9363	23
	GPT-4o	92.15	0.9222	70

Table 3: Benchmarking results of the AI detectors tested against the 1,000 text generated from the GPT-4o and Command R plus models. Evaluation Accuracy (Acc.), F1-Score (F1), and False Negatives (FN) reported for baseline, GPT-4 o and Command R plus model.

6 Limitations

The benchmarking experiment is done on three popular open-source AI text detection models. However, the framework and the datasets can be used for testing other AI text detection models also. Six of the basic evasion strategies have been applied to generate data samples representing evasion applied to AI-generated text. This can be further enhanced by employing other additional evasion strategies such as adversarial prompting (Wang et al., 2024), Substitution-based In-Context example Optimization method (SICO) (Lu et al., 2023), Self-color testing-based substitution (Wu and Chandrasekaran, 2024), and Reinforcement learning (Nicks et al., 2023).

7 Conclusion

The research highlighted significant critical challenges in detecting LLM-generated text. The existing state-of-the-art algorithms for detecting text generated from LLMs could perform better when tested on text generated from other domains, LLM-generated text on which evasion techniques have been applied and text generated from recent LLMs. This leads to the conclusion that these algorithms cannot be fully relied upon and used in university assignment checkers and research publications checkers.

From the results and analysis of the benchmarking against the evasion techniques, we can observe that even simple techniques, such as deleting a random article or misspelling a random word on AI-generated text, can bypass existing AI detectors. Similarly, from the results and analysis from multiple benchmarking experiments, it can be concluded that training on diverse AI-generated text, including evasion techniques, domains, and outputs from various LLMs, improves detector robustness.

Furthermore, the knowledge extracted from the critical analysis of the models serves as the baseline for future researchers trying to build robust AI-generated text detection algorithms. Training models representing a wide variety of data (multiple domains, multiple evasion techniques being applied, and multiple generators) may lead to the development of more efficient detectors. Likewise, training models with an adversarial learning approach that aims to train the model in different adversarial attacks and scenarios also seems promising.

The benchmarking in this research validates that the problem is far from solved. The knowledge gained from the critical analysis of the results concerning different approaches will help shape the further development of algorithms that can solve the problem more robustly. With the contribution of knowledge extracted from the experiment and thorough analysis of the results obtained, we aim to develop more robust AI detectors.

Acknowledgments

This publication has emanated from research conducted with the financial support of/supported in part by a grant from Science Foundation Ireland under Grant number 18/CRT/6183. For Open Access, the author has applied a CC by public copyright licence to any Author Accepted Manuscript version arising from this submission.

References

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

Wissam Antoun, Virginie Mouilleron, Benoît Sagot, and Djamé Seddah. 2023. Towards a robust detection of language model generated text: Is chatgpt that easy to detect? *arXiv preprint arXiv:2306.05871*.

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

Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural language processing with Python: analyzing text with the natural language toolkit*. " O'Reilly Media, Inc.".

Tom B Brown. 2020. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*.

Shuyang Cai and Wanyun Cui. 2023. Evade chatgpt detectors via a single space. *arXiv preprint arXiv:2307.02599*.

Chaka Chaka. 2024. Reviewing the performance of ai detection tools in differentiating between ai-generated and human-written texts: A literature and integrative hybrid review. *Journal of Applied Learning and Teaching*, 7(1).

Yapei Chang, Kalpesh Krishna, Amir Houmansadr, John Wieting, and Mohit Iyyer. 2024. Postmark: A robust blackbox watermark for large language models. *arXiv preprint arXiv:2406.14517*.

Zhiyuan Chen, Yaning Li, and Kairui Wang. 2024. Optimizing reasoning abilities in large language models: A step-by-step approach.

Woon Sang Cho, Pengchuan Zhang, Yizhe Zhang, Xiujun Li, Michel Galley, Chris Brockett, Mengdi Wang, and Jianfeng Gao. 2018. Towards coherent and cohesive long-form text generation. *arXiv preprint arXiv:1811.00511*.

Lynn Chua, Badih Ghazi, Yangsibo Huang, Pritish Kamath, Ravi Kumar, Pasin Manurangsi, Amer Sinha, Chulin Xie, and Chiyuan Zhang. 2024. Crosslingual capabilities and knowledge barriers in multilingual large language models. *arXiv preprint arXiv:2406.16135*.

Prithiviraj Damodaran. 2021. Parrot: Paraphrase generation for nlu.

Liam Dugan, Alyssa Hwang, Filip Trhlik, Josh Magnus Ludan, Andrew Zhu, Hainiu Xu, Daphne Ippolito, and Chris Callison-Burch. 2024. Raid: A shared benchmark for robust evaluation of machine-generated text detectors. *arXiv preprint arXiv:2405.07940*.

Ahmed M Elkhatat, Khaled Elsaid, and Saeed Almeer. 2023. Evaluating the efficacy of ai content detection tools in differentiating between human and ai-generated text. *International Journal for Educational Integrity*, 19(1):17.

Madelyn A Flitcroft, Salma A Sheriff, Nathan Wolfrath, Ragasnehith Maddula, Laura McConnell, Yun Xing, Krista L Haines, Sandra L Wong, and Anai N Kothari. 2024. Performance of artificial intelligence content detectors using human and artificial intelligence-generated scientific writing. *Annals of Surgical Oncology*, pages 1–7.

Sebastian Gehrmann, Hendrik Strobelt, and Alexander M Rush. 2019. Gltr: Statistical detection and visualization of generated text. *arXiv preprint arXiv:1906.04043*.

Eva Giboulot and Furon Teddy. 2024. Watermax: breaking the llm watermark detectability-robustness-quality trade-off. *arXiv preprint arXiv:2403.04808*.

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. *arXiv preprint arXiv:2301.07597*.

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

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

Jie Huang and Kevin Chen-Chuan Chang. 2022. Towards reasoning in large language models: A survey. *arXiv preprint arXiv:2212.10403*.

Raisa Islam and Owana Marzia Moushi. 2024. Gpt-4o: The cutting-edge advancement in multimodal llm. *Authorea Preprints*.

John Kirchenbauer, Jonas Geiping, Yuxin Wen, Jonathan Katz, Ian Miers, and Tom Goldstein. 2023. A watermark for large language models. In *International Conference on Machine Learning*, pages 17061–17084. PMLR.

Ryuto Koike, Masahiro Kaneko, and Naoaki Okazaki. 2024. Outfox: Llm-generated essay detection through in-context learning with adversarially generated examples. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 21258–21266.

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. *Preprint*, arXiv:2303.13408.

Kalpesh Krishna, Yixiao Song, Marzena Karpinska, John Wieting, and Mohit Iyyer. 2024. Paraphrasing evades detectors of ai-generated text, but retrieval is an effective defense. *Advances in Neural Information Processing Systems*, 36.

Gregory Kang Ruey Lau, Xinyuan Niu, Hieu Dao, Jiangwei Chen, Chuan-Sheng Foo, and Bryan Kian Hsiang Low. 2024. Waterfall: Framework for robust and scalable text watermarking. *arXiv preprint arXiv:2407.04411*.

Aiwei Liu, Qiang Sheng, and Xuming Hu. 2024. Preventing and detecting misinformation generated by large language models. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 3001–3004.

Yikang Liu, Ziyin Zhang, Wanyang Zhang, Shisen Yue, Xiaojing Zhao, Xinyuan Cheng, Yiwen Zhang, and Hai Hu. 2023. Argugpt: evaluating, understanding and identifying argumentative essays generated by gpt models. *arXiv preprint arXiv:2304.07666*.

Ning Lu, Shengcai Liu, Rui He, Qi Wang, Yew-Soon Ong, and Ke Tang. 2023. Large language models can be guided to evade ai-generated text detection. *arXiv preprint arXiv:2305.10847*.

Piotr Molenda, Adian Liusie, and Mark JF Gales. 2024. Waterjudge: Quality-detection trade-off when watermarking large language models. *arXiv preprint arXiv:2403.19548*.

Charlotte Nicks, Eric Mitchell, Rafael Rafailov, Archit Sharma, Christopher D Manning, Chelsea Finn, and Stefano Ermon. 2023. Language model detectors are easily optimized against. In *The Twelfth International Conference on Learning Representations*.

Guillaume-Anthony Odri and Diane Ji Yun Yoon. 2023. Detecting generative artificial intelligence in scientific articles: evasion techniques and implications for scientific integrity. *Orthopaedics & Traumatology: Surgery & Research*, 109(8):103706.

Xinlin Peng, Ying Zhou, Ben He, Le Sun, and Yingfei Sun. 2024. Hidding the ghostwriters: An adversarial evaluation of ai-generated student essay detection. *arXiv preprint arXiv:2402.00412*.

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

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

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

Vivek Verma, Eve Fleisig, Nicholas Tomlin, and Dan Klein. 2023. Ghostbuster: Detecting text ghostwritten by large language models. *arXiv preprint arXiv:2305.15047*.

Yichen Wang, Shangbin Feng, Abe Bohan Hou, Xiao Pu, Chao Shen, Xiaoming Liu, Yulia Tsvetkov, and Tianxing He. 2024. Stumbling blocks: Stress testing the robustness of machine-generated text detectors under attacks. *arXiv preprint arXiv:2402.11638*.

Yuxia Wang, Jonibek Mansurov, Petar Ivanov, Jinyan Su, Artem Shelmanov, Akim Tsvigun, Chenxi Whitehouse, Osama Mohammed Afzal, Tarek Mahmoud, Toru Sasaki, et al. 2023. M4: Multi-generator, multi-domain, and multi-lingual black-box machine-generated text detection. *arXiv preprint arXiv:2305.14902*.

Debora Weber-Wulff, Alla Anohina-Naumeca, Sonja Bjelobaba, Tomáš Foltnek, Jean Guerrero-Dib, Olumide Popoola, Petr Šigut, and Lorna Waddington. 2023. Testing of detection tools for ai-generated text. *International Journal for Educational Integrity*, 19(1):26.

Junchao Wu, Shu Yang, Runzhe Zhan, Yulin Yuan, Derek F. Wong, and Lidia S. Chao. 2023. A survey on llm-gernerated text detection: Necessity, methods, and future directions. *CoRR*, abs/2310.14724.

Qilong Wu and Varun Chandrasekaran. 2024. Bypassing llm watermarks with color-aware substitutions. *arXiv preprint arXiv:2403.14719*.

Lixiang Yan, Lele Sha, Linxuan Zhao, Yuheng Li, Roberto Martinez-Maldonado, Guanliang Chen, Xinyu Li, Yueqiao Jin, and Dragan Gašević. 2024. Practical and ethical challenges of large language models in education: A systematic scoping review. *British Journal of Educational Technology*, 55(1):90–112.

Xianjun Yang, Wei Cheng, Yue Wu, Linda Petzold, William Yang Wang, and Haifeng Chen. 2023. Dna-gpt: Divergent n-gram analysis for training-free detection of gpt-generated text. *arXiv preprint arXiv:2305.17359*.

Yunpu Zhao, Rui Zhang, Wenyi Li, Di Huang, Jiaming Guo, Shaohui Peng, Yifan Hao, Yuanbo Wen, Xing Hu, Zidong Du, Qi Guo, Ling Li, and Yunji Chen. 2024. Assessing and understanding creativity in large language models. *Preprint*, arXiv:2401.12491.

Tianyang Zhong. 2024. Evaluation of openai o1: Opportunities and challenges of agi. *Preprint*, arXiv:2409.18486.