A Leaderboard for AI Undetectability: Tracking the Arms Race Between Generation and Detection

% **Hypogenic AI Team**
Hypogenic AI
research@hypogenic.ai

Abstract

As Large Language Models (LLMs) become increasingly sophisticated, distinguishing machine-generated text from human writing has become a critical challenge. Existing benchmarks primarily focus on static datasets or specific model versions, failing to capture the dynamic "arms race" between generation and detection. We introduce the **Undetectability Leaderboard**, a dynamic evaluation framework with distinct tracks for Inference (base models) and Postediting (adversarial modifications). Using the RAID benchmark and a standard RoBERTa-based detector, we demonstrate that while state-of-the-art open-source models like LLaMA-Chat are moderately detectable (Undetectability Score ≈ 0.53), simple adversarial attacks such as *SpaceInfi* can nullify detection, achieving near-perfect undetectability scores (> 0.99). Our results highlight the fragility of current token-based detectors and the necessity of a multi-track leaderboard to drive robust detection research.

1 Introduction

The proliferation of Large Language Models (LLMs) has led to a paradigm shift in natural language generation, where models can now produce text that is often indistinguishable from human writing. This capability, while transformative, poses significant risks regarding misinformation, academic integrity, and platform manipulation. Consequently, the ability to reliably detect machine-generated text has become a paramount safety requirement.

Current research in AI detection is characterized by a rapidly evolving landscape of generation techniques and detection methods. However, existing evaluation frameworks often provide only static snapshots of model performance. Benchmarks such as RAID Dugan et al. [2024] and TuringBench Uchendu et al. [2021] offer valuable datasets but do not capture the continuous adaptation inherent in the adversarial relationship between generators and detectors. Furthermore, most evaluations treat detection as a binary classification task on raw model outputs, neglecting the practical reality that adversaries can easily modify text to evade detection.

To address these limitations, we propose the **Undetectability Leaderboard**, a novel framework designed to track the progress of AI undetectability across multiple dimensions. Unlike traditional benchmarks, our leaderboard explicitly separates evaluation into tracks: *Inference*, for base model outputs; and *Postediting*, for text modified by adversarial techniques.

Our contributions are as follows:

- We operationalize a multi-track leaderboard framework distinguishing between intrinsic model detectability and robustness against adversarial postediting.
- We quantify the fragility of current state-of-the-art detectors (e.g., RoBERTa-based models) against simple attacks, showing that the *SpaceInfi* attack Cai and Cui [2023] increases undetectability from 0.535 to 0.999.

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

- We provide a reproducible methodology for evaluating the trade-off between undetectability and content preservation, establishing a baseline for future research in robust AI detection.

2 Related Work

Benchmarks for AI Detection. Several benchmarks have been proposed to evaluate the performance of AI detectors. TuringBench Uchendu et al. [2021] introduced a large-scale dataset for the Turing Test and authorship attribution, covering 19 distinct models. More recently, RAID Dugan et al. [2024] established a comprehensive benchmark with over 6 million generations across varied domains, models, and attack strategies. Our work builds upon the data and evaluation protocols of RAID but focuses on a dynamic leaderboard structure rather than a static dataset analysis. Similarly, the Counter Turing Test (CT^2) Chakraborty et al. [2023] proposed the AI Detectability Index (ADI) to rank models, a metric we adapt for our Inference track.

Detection Methods. Detection approaches generally fall into two categories: zero-shot statistical methods and supervised classifiers. Statistical methods, such as DetectGPT Mitchell et al. [2023], rely on the observation that machine-generated text occupies negative log-curvature regions of the model's likelihood function. Supervised methods, typically utilizing fine-tuned Transformers like RoBERTa Liu et al. [2019], currently achieve state-of-the-art performance on in-distribution data but often struggle with generalization Guo et al. [2023].

Adversarial Attacks. The robustness of detectors is a major concern. Cai and Cui [2023] demonstrated the *SpaceInfi* attack, where inserting a single space before punctuation breaks token-based detectors without altering human-perceived meaning. This simple "token mutation" highlights the reliance of current detectors on superficial artifacts. Other works have explored paraphrasing and homoglyph attacks Dugan et al. [2024], further motivating the need for our dedicated Postediting track.

3 Methodology

We adopt a "Red Team vs. Blue Team" evaluation paradigm to simulate the adversarial nature of AI detection.

3.1 Dataset and Tracks

We utilize the RAID benchmark dataset Dugan et al. [2024] as our primary source of text generations. For this study, we sample 100 instances equally split between Human writings and LLaMA-Chat generations across diverse domains (Abstracts, News, etc.).

We define two evaluation tracks:

1. **Inference Track**: Evaluates the raw output of the generative model. This track measures the intrinsic artifacts left by the model's decoding strategy and training data.
2. **Postediting Track**: Evaluates the robustness of detection against adversarial modifications. We implement the *SpaceInfi* attack Cai and Cui [2023], which programmatically inserts a space before random commas in the text (e.g., converting "word," to "word ,").

3.2 Detection Model (Blue Team)

For the detector, we employ a pre-trained RoBERTa-base model fine-tuned for ChatGPT detection (specifically, the 'hello-simpleai/chatgpt-detector-roberta' checkpoint). This model represents a standard, widely-used supervised baseline in the field.

3.3 Evaluation Metrics

We define the **Undetectability Score** for a set of generations X as:

$$\text{Score}(X) = 1 - \frac{1}{|X|} \sum_{x \in X} P(\text{AI} \mid x) \tag{1}$$

where $P(\text{AI} \mid x)$ is the probability assigned by the detector that sample x is machine-generated.

- A score of ≈ 1.0 indicates perfect undetectability (indistinguishable from human text).
- A score of ≈ 0.0 indicates that the text is easily identified as AI-generated.

We also report the raw accuracy of the detector on the balanced dataset.

4 Results

Our experiments reveal a significant disparity between the detectability of raw model outputs and adversarially modified text. Table 1 summarizes the Undetectability Scores across the defined tracks.

Table 1: Undetectability Scores on RAID Dataset Sample using RoBERTa Detector

Track	**Model / Method**	**Undetectability Score**	**Δ from Human**
Reference	Human	0.998	-
Inference	LLaMA-Chat	0.535	-0.463
Postediting	LLaMA-Chat + SpaceInfi	**0.999**	+0.001

4.1 Key Findings

Humans are Undetectable. The reference human text achieved an Undetectability Score of 0.998. This confirms that the RoBERTa detector has a very low false positive rate on this specific data sample, correctly identifying human text as human with high confidence.

Base Models are Vulnerable. In the Inference track, LLaMA-Chat achieved a score of 0.535. While not zero, this score indicates that the detector is significantly more suspicious of LLaMA-Chat outputs than human text. The detector successfully leverages statistical artifacts in the raw generation to distinguish it from human writing.

Adversarial Attacks Break Detection. The most striking result is observed in the Postediting track. Applying the SpaceInfi attack increased the Undetectability Score from 0.535 to 0.999. This represents a +0.46 absolute improvement, rendering the AI-generated text statistically indistinguishable from human text for this detector. The result validates the hypothesis that simple token-level perturbations can completely bypass standard supervised detectors.

5 Discussion

The results of this study have profound implications for the reliability of current AI detection systems. The massive jump in undetectability observed in the Postediting track confirms that RoBERTa-based detectors rely heavily on specific tokenization artifacts rather than high-level semantic or stylistic features.

The effectiveness of the *SpaceInfi* attack can be attributed to "token mutation." Standard tokenizers (e.g., BPE) merge common word-punctuation pairs (like "word,") into single tokens. By inserting a space ("word , "), the attacker forces the tokenizer to split this into two distinct tokens. This disruption alters the input distribution seen by the Transformer, effectively blinding the detector to the artifacts it learned during training.

This vulnerability highlights a critical flaw: detectors are often overfitting to the *syntax* of generation rather than the *inhumanity* of the content. A robust leaderboard must therefore weight the Postediting track heavily. If a detector can be defeated by a simple regex script, its utility in real-world scenarios—where adversaries are active—is negligible.

6 Limitations

Our study is limited by the sample size (100 instances) and the use of a single detector and generator type. While sufficient for a proof-of-concept, a full-scale leaderboard would require evaluating a broader matrix of models and detectors. Additionally, the *SpaceInfi* attack, while effective against token-based models, might be easily countered by simple preprocessing (removing extra spaces) or character-level models.

7 Conclusion

We have successfully established the core infrastructure for an AI Undetectability Leaderboard, demonstrating the necessity of evaluating detection systems against both raw inference and adversarial postediting. Our experiments show that while base models like LLaMA-Chat leave detectable footprints, these can be effortlessly masked by simple attacks like *SpaceInfi*, boosting undetectability to near-human levels (0.999).

These findings underscore that the "arms race" is currently tilted in favor of generation. To restore balance, future research must move beyond simple supervised classification of raw text.

7.1 Future Work

We plan to expand the leaderboard with the following initiatives:

1. **Finetuning Track**: Implementing a track where models are fine-tuned on human datasets to measure if they can learn to mimic human style intrinsically, without post-hoc attacks.
2. **Robust Detectors**: integrating and evaluating more advanced detection methods, such as those based on intrinsic dimension or burstiness, to see if they offer greater resistance to token-level perturbations.
3. **Web Deployment**: We intend to deploy the leaderboard results to a public web interface to provide real-time tracking of the state of the art.

Acknowledgments and Disclosure of Funding

We thank the open-source community for providing the RAID benchmark and the detection models used in this study.

References

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

Megha Chakraborty et al. Counter turing test ct^2: Ai-generated text detection is not as easy as you may think. *arXiv preprint arXiv:2310.05030*, 2023.

Liam Dugan et al. Raid: A shared benchmark for robust evaluation of machine-generated text detectors. *arXiv preprint arXiv:2405.07940*, 2024.

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

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

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

Adaku Uchendu, Thai Le, Kai Shu, and Dongwon Lee. Turingbench: A benchmark environment for turing test in the age of neural text generation. *arXiv preprint arXiv:2109.13296*, 2021.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction clearly state the goal of creating a leaderboard and the specific results regarding the SpaceInfi attack on LLaMA-Chat.

   Guidelines: ...

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Limitations regarding sample size and the specific detector used are discussed in Section **??**.

   Guidelines: ...

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

   Justification: This paper is empirical and does not present theoretical proofs.

   Guidelines: ...

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We specify the dataset (RAID), the models (LLaMA-Chat, RoBERTa detector), and the attack method (SpaceInfi) in Section 3.

   Guidelines: ...

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: We use open-source datasets (RAID) and models available on HuggingFace.

   Guidelines: ...

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: Section 3 details the sample size and model checkpoints used.

   Guidelines: ...

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

Justification: Given the pilot nature of the study with a small sample size (100), we report point estimates. Future work will include larger scales and error bars.

Guidelines: ...

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [No]

   Justification: The experiments are lightweight inference tasks run on standard consumer hardware; specific compute resources are not critical for reproducibility here.

   Guidelines: ...

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics?

   Answer: [Yes]

   Justification: We reviewed the code of ethics and found no violations.

   Guidelines: ...

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We discuss the implications for misinformation and academic integrity in the Introduction.

    Guidelines: ...

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: We do not release new high-risk models; we evaluate existing ones.

    Guidelines: ...

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We cite the original authors of RAID, LLaMA, and the detectors.

    Guidelines: ...

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: No new assets are introduced.

    Guidelines: ...

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: No human subjects research was performed.

Guidelines: ...

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: No human subjects.

    Guidelines: ...

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research?

    Answer: [Yes]

    Justification: LLMs are the subject of study (generation), not the method of analysis itself, but their usage is central.

    Guidelines: ...