Revisiting the Robustness of Watermarking to Paraphrasing Attacks

Saksham Rastogi
Indian Institute of Science
Bengaluru, India
iitdsaksham@gmail.com

Danish Pruthi
Indian Institute of Science
Bengaluru, India
danishp@iisc.ac.in

Abstract

Amidst rising concerns about the internet being proliferated with content generated from language models (LMs), watermarking is seen as a principled way to certify whether text was generated from a model. Many recent watermarking techniques slightly modify the output probabilities of LMs to embed a signal in the generated output that can later be detected. Since early proposals for text watermarking, questions about their robustness to paraphrasing have been prominently discussed. Lately, some techniques are deliberately designed and claimed to be robust to paraphrasing. However, such watermarking schemes do not adequately account for the ease with which they can be reverse-engineered. We show that with access to only a limited number of generations from a black-box watermarked model, we can drastically increase the effectiveness of paraphrasing attacks to evade watermark detection, thereby rendering the watermark ineffective.[1]

1 Introduction

Given the remarkable fluency and relevance with which language models (LMs) respond to varied queries, it is challenging for humans to distinguish language model outputs from human-written text. Past studies note that human performance in making such a distinction is close to that of random chance (Gehrmann et al., 2019; Brown et al., 2020). In response, watermarking language models is seen to be a principled way to certify whether a piece of text was generated by a model.

A prominent watermarking approach works by implanting a signal during decoding, wherein a certain set of tokens (aka a green list) is boosted (Kirchenbauer et al., 2023a). This signal, albeit imperceptible to an unsuspecting reader, can be verified by running a statistical test. For watermarking to be effective, the implanted signal should be easy to detect and hard to remove. Unsurprisingly, there has been considerable discussion about the robustness of watermarking approaches against paraphrasing attacks (Krishna et al., 2023; Kirchenbauer et al., 2023b).

There exist different ways of choosing tokens in the green list and the extent to which they should be boosted. These approaches offer varying levels of robustness against paraphrasing. The original paper (Kirchenbauer et al., 2023a) recommends pseudo-randomly selecting a different set of green tokens at every timestep based on a hash of the last k tokens. The authors note that higher values of k would render the watermarking scheme ineffective, as any changes to a token would disrupt the green lists for the next k timesteps, and therefore suggest using the last one or two tokens (k = 1 or 2). A recent study (Zhao et al., 2023) argues that "a consistent green list is the most robust choice," as any modifications to the input text have no effect whatsoever on the (fixed) green list. Relatedly, Liu et al. (2024) propose a "semantic-invariant robust" watermarking (SIR) which is designed to produce similar green lists for semantically-similar contexts and is touted to be robust to paraphrasing.

In this ongoing debate, our work highlights just how easy it is to decipher the green list for both the semantic-invariant watermarking scheme (Liu et al., 2024) and watermarking with consistent green list (Zhao et al., 2023). While a recent contemporaneous study (Jovanović et al., 2024) corroborates that watermarking with a fixed green list can be easily reverse-engineered, we show that similar results also hold for semantic-invariant watermarking scheme from Liu et al. (2024). For both these watermarking schemes, with just 200K tokens of watermarked output, we can predict green lists with over 0.8 F1 score. This knowledge of green lists can be exploited while paraphrasing to launch at-

[1] Data and code to replicate our evaluation is available at: https://github.com/codeboy5/revisiting-watermark-robustness

Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, pages 18100–18110
November 12-16, 2024 ©2024 Association for Computational Linguistics

tacks that cause the detection rates to plummet below 10%, rendering the watermark ineffective.

Overall, our findings suggest that one should consider the possibility of reverse-engineering the watermarking scheme, when discussing its robustness to paraphrasing attacks. Our work also raises potential concerns about the generalization of watermarking algorithms that use machine learning models to generate the watermarking signal.

2 Background

A prominent approach to watermarking is to compute watermarking logits that are added to logits generated by a language model at each generation step. Formally, for a language model $\mathcal{M}$ with vocabulary V, and a prefix comprising tokens $\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_n$, the scheme involves first computing the logits $\mathcal{M}(\mathbf{w}_1 \ldots, \mathbf{w}_n) = (l_1, \ldots, l_{|V|})$ of the language model that would ordinarily be used to predict the subsequent token. As per (Kirchenbauer et al., 2023a), the last k tokens, $\mathbf{w}_{n-k+1}$ to $\mathbf{w}_n$, are then fed to a psuedo-random function F to partition V into a green list G and a red list R such that $|G| + |R| = |V|$. Finally, the logits corresponding to the tokens in the green list, G, are boosted by δ ($\delta > 0$). The watermark can then be detected through a one-proportion z-test on the fraction of green tokens in the generated text.

A recent study (Zhao et al., 2023) makes a case for using a fixed green list (where the partitioning function, F, does not depend on the context) to confer robustness to paraphrasing attacks. The underlying intuition is that any changes in the text will not disrupt the constant green list. To counter paraphrasing attacks, another recent proposal (Liu et al., 2024) is to train a model, $\mathcal{W}$, to output watermarking logits using the context: $\mathcal{W}(\mathbf{w}_1 \ldots, \mathbf{w}_n) = (\delta_1, \ldots, \delta_{|V|})$. This model, $\mathcal{W}$, is designed such that similar contexts yield similar watermarking logits. This property is supposed to make models robust to paraphrasing. Further, diverse contexts are supposed to yield different watermarking logits, thus making it hard to reverse-engineer the green list—this is not true in practice, as we show later in our experiments (§4.2).

Paraphrasing Attacks. Krishna et al. (2023) introduce a controllable paraphraser and launch paraphrasing attacks on various text detectors. Their findings indicate that although paraphrasing reduces the effectiveness of most AI-generated text detectors, watermarking is the most resilient method. Another study (Kirchenbauer et al., 2023b) investigates the reliability of watermarks across different paraphrasing models and suggests that the reliability of watermarking should be discussed in terms of the length of the available input. The study concludes that watermarking is extremely reliable for longer texts.

3 Methods

We study the robustness of watermarking approaches against paraphrasing attacks. Unlike prior attacks (Krishna et al., 2023; Kirchenbauer et al., 2023b), we first attempt to decipher the tokens in the green list and then incorporate that knowledge in existing paraphrasing attacks.

3.1 Estimating Green Lists

We assume access to only generations from the watermarked model, with no access to model weights or its tokenizer. To decipher the green list, we use a simple counting-based algorithm similar to the ones used in prior work (Zhao et al., 2023; Sadasivan et al., 2024). Specifically, we compare the relative frequencies of tokens in a corpus generated by the watermarked model against their relative frequencies in a corpus of human-written text. Tokens that exhibit a higher relative frequency in the watermarked corpus compared to the reference corpus are classified as green tokens. We present the detailed algorithm in Appendix A.

This approach is similar to the one proposed in a contemporaneous work (Jovanović et al., 2024), where for each token, they compute two conditional probabilities: probability of a token given its preceding context in a watermarked corpus and the same probability in a base corpus. They investigate two scenarios for obtaining the base corpus: using a non-watermarked version of the same language model or using a different language model as a proxy for the base distribution. In contrast, our approach does not require access to the unwatermarked language model for the base distribution; instead, we derive our base distribution from the OpenWebText corpus.[2] Furthermore, our approach assigns binary scores of 0 and 1 for tokens in the red and green lists, respectively. Please note that the green list once estimated using the algorithm

[2] To demonstrate the robustness of our algorithm across different base distributions, we present additional results in the appendix, where we estimate the green list using the RealNewsLike subset of the C4 dataset (Raffel et al., 2020).

can be used to launch paraphrasing attacks on a variety of downstream datasets (details in §4.1).

A note about metrics: Prior work relies on F1 score to evaluate the correctness of predicting the green list. However, this metric assumes equal importance for all tokens and fails to account for the fact that natural language follows a Zipf's law, wherein the frequency of a word is inversely proportional to its rank in the list (sorted in decreasing order of word frequencies). While it may seem like a minor technicality, we show that the traditional F1 score overestimates the security of watermarking.

To address this limitation, we suggest using a generation-based F1 score that computes the F1 score for classifying tokens into green or red list for each token *in text generated from watermarked models*. This small change incorporates the relative frequency of each token.

3.2 Paraphrasing with Green Lists

One can imagine that incorporating prior knowledge about green lists should be able to improve the efficiency of off-the-shelf paraphrasers to remove the watermark signal and evade detection. Since many paraphrasing models are also autoregressive generative models (Krishna et al., 2023; Lewis et al., 2020; Lin et al., 2021; Witteveen and Andrews, 2019), one can introduce an inverse watermarking signal into the generated text. Specifically, at every generation timestep, we subtract a small positive δ from the logits corresponding to tokens predicted to be in the green list.

4 Results & Discussion

We first share details about our setup and then discuss the results of paraphrasing attacks.

4.1 Experimental Setup

We primarily consider two watermarking schemes that are designed and thought to be robust against paraphrasing, namely, the semantic-invariant robust (SIR) watermarking (Liu et al., 2024) and watermarking with a fixed green list, which is referred as UNIGRAM WATERMARKING in a recent study that analyzes the robustness of this approach (Zhao et al., 2023). We use the LLaMA-7B model (Touvron et al., 2023) and apply the two watermark algorithms with hyperparameters of $\gamma = 0.5$ (fraction of green tokens) and $\delta = 2.0$ (value used to boost the logits for green tokens) for all results presented in the main paper. Additional results using the Pythia model and other watermark hyperparameter choices are presented in Appendix B.

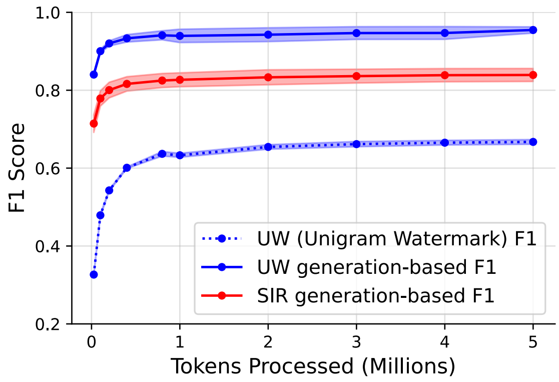
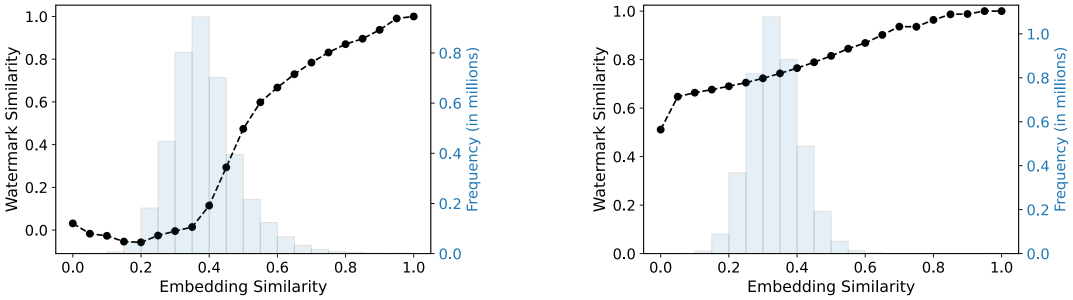


Figure 1: We show that with a limited amount of generated tokens, we can achieve a high F1 score for predicting the green lists of two watermarking schemes.

To evaluate the watermarking schemes and their robustness to paraphrasing, we use 50-token prompts from Wikipedia articles (Foundation, 2022) to generate 200-token completions. (Note that this dataset is different from the one used to estimate the green list.) We consider the subsequent 200 tokens from the Wikipedia articles as human-written text for comparison. Additionally, we present results on prompts from arXiv papers (Cohan et al., 2018) and Booksum (Kryscinski et al., 2022) dataset in the Appendix (§B.3) to demonstrate the effectiveness of our attack on generations using prompts from diverse domains.

The results for each attack setting are aggregated across 500 generations from the LLM. We use the DIPPER paraphrasing model (Krishna et al., 2023) and incorporate the knowledge of (estimated) green list tokens (as described in §3.2). To evaluate the detection accuracy of watermarking algorithms, we follow prior work and measure the True Positive Rate (TPR) at a low False Positive Rate (FPR) of 1% and 10%. The False Positive Rate is set to low values to avoid falsely accusing someone of plagiarism. We use the P-SP metric (Wieting et al., 2022) to assess the semantic similarity of paraphrases, past work considers the semantics of the paraphrase to be preserved if the P-SP value exceeds 0.76 (Krishna et al., 2023). Additionally, we assess the quality of produced text by calculating the perplexity (PPL) using LLaMA-13B, which we consider as an oracle model.



(a) in-domain samples

(b) out-of-domain samples

Figure 2: For SIR watermarking, we depict the cosine similarity of the context embeddings (x-axis) vs the cosine similarity of the watermarking logits (y-axis). For in-domain samples, similar contexts produce similar watermarking logits and dissimilar ones produce different logits, however, this is not the case for out-of-domain samples.

Attack	UNIGRAM-WATERMARK				SIR			
	TPR @ 1% FPR	TPR @ 10% FPR	P-SP	PPL	TPR @ 1% FPR	TPR @ 10% FPR	P-SP	PPL
No Attack	99.3±0.7	100.0	1.00	14.5±1.0	93.3±0.0	98.8±0.1	1.00	12.8±0.8
DIPPER (L20)	88.7±2.4	98.0±0.5	0.95	11.3±0.7	45.5±0.9	82.7±4.1	0.94	10.1±0.6
DIPPER (L60)	62.8±2.2	92.1±0.7	0.90	10.5±0.7	24.0±0.5	62.3±2.4	0.90	9.6±0.4
Ours (L20)	3.2±0.8 ↓	13.4±2.5 ↓	0.87	11.6±1.0	7.3±2.1 ↓	20.3±5.5 ↓	0.88	10.6±0.5
Ours (L60)	0.2±0.2 ↓	1.9±0.7 ↓	0.78	11.1±1.1	3.8±0.8 ↓	10.2±3.1 ↓	0.81	10.2±0.8

Table 1: We compare the detection rates of UNIGRAM-WATERMARK and SIR against paraphrasing attacks. We use two settings of the paraphrasing model, DIPPER, with lexical diversities (LD) of 20 and 60; higher LD implies stronger attack. Our attack involves modifying DIPPER with the estimated knowledge of the green list (details in §3.2). We report the median P-SP & PPL values.

4.2 Results

We show that with just as few as 200K tokens, **we can accurately predict whether a token belongs to green list** (Figure 1). It may be unsurprising that one can decipher the fixed green list used in the UNIGRAM WATERMARKING, as also documented by Jovanović et al. (2024). However, is noteworthy and surprising that even semantic-invariant watermarking (SIR) scheme, which dynamically produces a green list (based on the embeddings of the context) is just as vulnerable.

While the SIR approach aspires to produce similar watermarking logits for similar contexts and dissimilar ones for dissimilar contexts, we discover that this is not the case in practice. In Figure 2, we plot the cosine similarity of the context embeddings vs the cosine similarity of the watermarking logits. Interestingly, we notice that the aspired notion of producing similar watermarking logits for only similar contexts holds true only for in-domain samples and breaks for out-of-domain (OOD) samples. For OOD samples, the produced watermarking logits are highly similar regardless of the similarity in contexts (Figure 2b), suggesting that the green lists in SIR are not as dynamic as previously believed and are susceptible to be deciphered. Our findings suggest that other (future) watermarking algorithms that use machine learning to generate the watermarking logits might suffer from similar generalization concerns.

From Figure 1, we can also observe that the vanilla F1 scores present an overly optimistic picture about the security of watermarking approaches. As discussed in §3.2, the vanilla F1 metric weighs in all tokens uniformly. This approach fails to account for the long tail of rare tokens—whose presence in green or red list is hard to predict—which (by definition) occur infrequently in practical application. However, tokens that are actually generated can be predicted far more accurately, as can be clearly seen through about a 50% higher generation-based F1 score in Figure 1.

Finally, we present results showing how the two watermarking schemes hold up against paraphrasing attacks (Table 1). We notice that the default DIPPER attack reduces the performance of both watermarking schemes. For FPR of 1%, it brings down the TPR to 88% (from 99.3%) for UNIGRAM-WATERMARK and to 45.5% (from 93.3%) for SIR. When we empower the attack with the knowledge of (estimated) green lists, the TPR values plummet to below 10%, rendering the watermarking schemes unusable. Across all setups, we confirm that the quality of LMs (measured through PPL) and the semantic meaning of paraphrases (evaluated via P-SP scores) is largely preserved. Interestingly, our attack is slightly less effective for SIR than UNIGRAM-WATERMARK as our estimates for green lists are less accurate for SIR.

A Note about Adaptive Text Watermark. Just recently, an approach called Adaptive Text Watermark (ATW) was proposed, aiming to generate high-quality text while maintaining robustness, security, and detectability (Liu and Bu, 2024). Conceptually similar to SIR, Adaptive Text Watermark generates a logit scaling vector (v) based on the semantic embeddings of previously generated text. The watermark is added to the LLM logits by proportionally scaling the original logits by a factor of $(1 + \delta.v)$, where $\delta > 0$ controls the watermark strength. We find that while our attack strategy can significantly reduce the detection rate of ATW by about 10% (Table 5), it is considerably more robust than the other two approaches. Further analysis reveals that unlike SIR, the semantic mapping module of ATW (that converts embeddings of prefixes to logits) generalizes better for out-of-domain distribituions. This result suggests that semantics-based watermarking may be a viable alternate to defend against paraphrasing attacks, however, we suggest practitioners to confirm whether such learning-based approaches generalize to OOD domains.

5 Conclusion

We analyze watermarking schemes believed to be specifically robust to paraphrasing and show that it is easy to reverse engineer these algorithms and launch severe paraphrasing attacks. The effectiveness of our attacks underscores the need to account for the ease of reverse-engineering watermarking schemes, when discussing its robustness to paraphrasing attacks. Additionally, we highlight that existing metrics concerning the security of watermarking are overly optimistic.

6 Limitations

Our work focuses specifically on watermarking schemes proposed to be robust against paraphrasing attacks. Future work can focus on other schemes, such as (Kuditipudi et al., 2024; Aaronson, 2023), which implant the watermark signal during sampling and claim to preserve the original distributions up to certain generation budgets. Another limitation of our work is that we do not address how effectively (ill-intentioned) humans can remove the watermark signal once they are aware of the estimated green lists. Additionally, paraphrasing attacks require significant compute as it uses a large language model for generating paraphrases.

Acknowledgements

We thank the anonymous reviewers for their contructive feedback. We are also grateful to Anirudh Ajith for his feedback. DP acknowledges Adobe Inc., Google Research, Schmidt Sciences, National Payments Corporations of India (NPCI) and Pratiksha Trust for supporting his group's research.

References

Scott Aaronson. 2023. 'reform' ai alignment with scott aaronson. AXRP - the AI X-risk Research Podcast.

Stella Biderman, Hailey Schoelkopf, Quentin Gregory Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, Aviya Skowron, Lintang Sutawika, and Oskar van der Wal. 2023. Pythia: A suite for analyzing large language models across training and scaling. In *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pages 2397–2430. PMLR.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 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*.

Arman Cohan, Franck Dernoncourt, Doo Soon Kim, Trung Bui, Seokhwan Kim, Walter Chang, and Nazli

Goharian. 2018. A discourse-aware attention model for abstractive summarization of long documents. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 615–621, New Orleans, Louisiana. Association for Computational Linguistics.

Wikimedia Foundation. 2022. Wikimedia downloads. https://dumps.wikimedia.org.

Sebastian Gehrmann, Hendrik Strobelt, and Alexander Rush. 2019. GLTR: Statistical detection and visualization of generated text. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 111–116, Florence, Italy. Association for Computational Linguistics.

Aaron Gokaslan, Vanya Cohen, Ellie Pavlick, and Stefanie Tellex. 2019. Openwebtext corpus. http://Skylion007.github.io/OpenWebTextCorpus.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b.

Nikola Jovanović, Robin Staab, and Martin Vechev. 2024. Watermark stealing in large language models. *ArXiv preprint*, abs/2402.19361.

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. *ArXiv preprint*, 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. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Wojciech Kryscinski, Nazneen Rajani, Divyansh Agarwal, Caiming Xiong, and Dragomir Radev. 2022. BOOKSUM: A collection of datasets for long-form narrative summarization. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 6536–6558, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Rohith Kuditipudi, John Thickstun, Tatsunori Hashimoto, and Percy Liang. 2024. Robust distortion-free watermarks for language models. *Preprint*, arXiv:2307.15593.

Mike Lewis, Marjan Ghazvininejad, Gargi Ghosh, Armen Aghajanyan, Sida I. Wang, and Luke Zettlemoyer. 2020. Pre-training via paraphrasing. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Zhe Lin, Yitao Cai, and Xiaojun Wan. 2021. Towards document-level paraphrase generation with sentence rewriting and reordering. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 1033–1044, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Aiwei Liu, Leyi Pan, Xuming Hu, Shiao Meng, and Lijie Wen. 2024. A semantic invariant robust watermark for large language models. *Preprint*, arXiv:2310.06356.

Yepeng Liu and Yuheng Bu. 2024. Adaptive text watermark for large language models.

Yijian Lu, Aiwei Liu, Dianzhi Yu, Jingjing Li, and Irwin King. 2024. An entropy-based text watermarking detection method.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21:140:1–140:67.

Vinu Sankar Sadasivan, Aounon Kumar, Sriram Balasubramanian, Wenxiao Wang, and Soheil Feizi. 2024. Can ai-generated text be reliably detected? *Preprint*, arXiv:2303.11156.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models. *Preprint*, arXiv:2307.09288.

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

Sam Witteveen and Martin Andrews. 2019. Paraphrasing with large language models. In *Proceedings of the 3rd Workshop on Neural Generation and Translation*, pages 215–220, Hong Kong. Association for Computational Linguistics.

Xuandong Zhao, Prabhanjan Ananth, Lei Li, and Yu-Xiang Wang. 2023. Provable robust watermarking for ai-generated text. *Preprint*, arXiv:2306.17439.

A Additional Experiment Details

A.1 Algorithm for estimating the green list

To estimate the green list, we compare the distribution of tokens between watermarked text and text from OpenWebText (Gokaslan et al., 2019) dataset (to simulate the distribution of non-watermarked text). We query the LLaMA-7B watermark model with 50 token prompts from (Gokaslan et al., 2019) to generate 256 token completions. We calculate the relative token frequencies for the watermarked text and text from the OpenWebText dataset. We use a minor modification of the algorithm used in (Zhao et al., 2023), with the difference being using relative frequencies instead of absolute and using a small positive threshold τ. We use a constant τ of 1×10^{-6} across all our experiments.

D_{wtm} and D_{human} refer to the distribution of tokens in watermarked and non-watermarked text.

Algorithm 1 Estimating the Green List tokens

```
for every token $v$ in the vocabulary $\mathcal{V}$ do
    $\Delta(v) \leftarrow D_{\text{wtm}}(v) - D_{\text{human}}(v)$
    if $\Delta(v) \geq \tau$ then
        $v$ is in the Green List.
    else
        $v$ is in the Red List.
    end if
end for
```

A.2 Estimating the green list using a different base distribution

To evaluate the robustness of Algorithm 1, we present additional results on estimating the green list using the RealNewsLike subset of the c4 dataset (Raffel et al., 2020) as the base distribution (D_{human} in Algorithm 1). The results of these experiments, summarized in Table 2, span evaluations on two distinct base models. Our findings demonstrate that the algorithm's performance remains consistent across different choices of base distribution, thus confirming its robustness.

B Additional Results

We present additional results for other choices of γ (0.1, 0.25) in §B.1 and present results on Pythia 1.4B and Mistral 7B in §B.2. These additional analyses serve to underscore the generalizability of our findings.

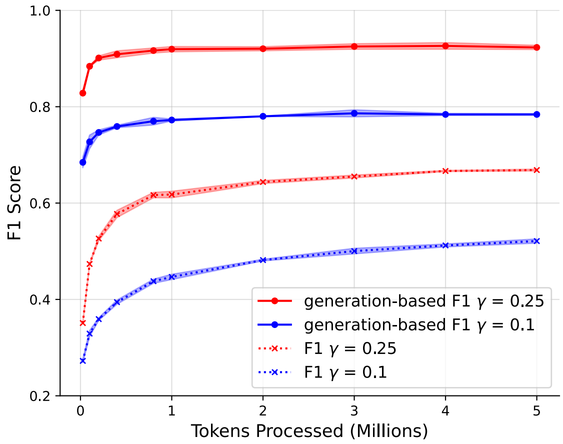


Figure 3: Comparision between the traditional F1 score and generated-based F1 score. We can observe that across all choices of γ, the traditional F1 metric can understate the security robustness.

B.1 Results for other choices of γ

We compare the F1 and generation-based F1 score across other choices of γ (Figure 3). We consistently observe a gap between the two metrics. We also note that we can reverse engineer the watermark across all choices of γ. Additionally, we present the impact of paraphrasing on the watermarking scheme in Table 6. Our results indicate that our attack remains highly effective regardless of the value of γ.

B.2 Results on Additional Models

We present the performance of watermarking schemes against paraphrasing attacks, using Pythia-1.4b (Biderman et al., 2023) and Mistral 7B (Jiang et al., 2023) as the base language models in Table 7 and Table 8. For all experiments, we set the watermark hyperparameters to $\gamma = 0.5$ and $\delta = 2.0$. Our results demonstrate that the proposed paraphrasing attack significantly degrades the performance of both watermarking schemes evaluated. These results indicate that our findings are applicable across different model classes and sizes.

B.3 Efficacy of our paraphrasing attack on prompts from diverse datasets

To demonstrate the effectiveness of our proposed paraphrasing attack on generations from diverse domains, we evaluate our attack on prompts from arXiv papers (Cohan et al., 2018) and Booksum (Kryscinski et al., 2022). We follow a similar setup as explained in §4.1. These results also serve as evidence that the green list estimated once using a

	LLaMA 7B				Mistral 7B			
	Precision	Recall	FPR	F1	Precision	Recall	FPR	F1
UW	0.89	0.46	0.05	0.61	0.91	0.51	0.05	0.65
UW generation-based	0.96	0.88	0.08	0.92	0.98	0.92	0.05	0.95
SIR generation-based	0.82	0.77	0.27	0.79	0.88	0.72	0.22	0.79

Table 2: We evaluate the performance of our watermark reverse engineering approach using a corpus of 1 million tokens, with the RealNewsLike subset of the C4 dataset serving as the base distribution. Our assessment metrics include precision, recall, false positive rate (FPR), and F1 score. These results indicate that our proposed paraphrasing attack is robust to the choice of the base distribution we use to reverse engineer the watermark scheme.

	Precision	Recall	FPR	F1
UW	0.89	0.48	0.05	0.62
UW generation-based	0.96	0.92	0.09	0.93
SIR generation-based	0.88	0.80	0.24	0.83

Table 3: We report the precision, recall, FPR and F1 for reverse engineering the watermarking using 1 million tokens. This table complements the results reported in Figure 1 and provides additional insight that the difference in F1 score is primarily driven by the difference in recall. This aligns with our intuition that we fail to correctly classify tokens that are less frequent.

particular dataset (OpenWebText (Gokaslan et al., 2019) in this case) can be used to launch paraphrasing attacks on a variety of downstream datasets. The results are summarized in Table 9 and Table 10 for arXiv papers and Booksum dataset respectively.

C Paraphrasing attacks against EWD

Entropy-based Text Watermarking Detection (EWD) (Lu et al., 2024) introduces a novel approach to watermark detection by incorporating token entropy. This method assigns higher importance to high-entropy tokens during the detection process, thereby enhancing detection performance in low-entropy contexts. We conducted an empirical investigation into the robustness of EWD against paraphrasing attacks. The results of our analysis are presented in Table 4, providing insights into the method's resilience to paraphrasing attacks. From the results we can observe that incorporating the estimated green list can significantly improve the effectiveness of the paraphrasing attack.

Attack	TPR @ 1% FPR	TPR @ 10% FPR	P-SP	PPL
No Attack	100.0	100.0	1.0	15.3
DIPPER (L20)	99.2	99.8	0.95	11.8
DIPPER (L60)	93.2	98.0	0.90	11.0
Ours (L20)	9.0 ↓	19.2 ↓	0.88	12.3
Ours (L60)	0.2 ↓	0.6 ↓	0.81	11.9

Table 4: Paraphrasing attacks against EWD algorithm.

Attack	TPR @ 1% FPR	TPR @ 10% FPR	P-SP	PPL
No Attack	96.5	99.7	1.0	11.6
DIPPER (L20)	67.2	91.7	0.94	7.9
DIPPER (L60)	43.5	82.0	0.90	7.8
Ours (L20)	57.2 ↓	89.5 ↓	0.92	8.4
Ours (L60)	33.7 ↓	76.0 ↓	0.88	8.4

Table 5: Paraphrasing attacks against Adaptive Text Watermark.

	UNIGRAM-WATERMARK with $\gamma = 0.1$				UNIGRAM-WATERMARK with $\gamma = 0.25$			
Attack	TPR @ 1% FPR	TPR @ 10% FPR	P-SP	PPL	TPR @ 1% FPR	TPR @ 10% FPR	P-SP	PPL
No Attack	97.0	99.8	1.00	13.03	99.0	100.0	1.0	15.27
DIPPER-*L20*	76.6	94.8	0.94	10.33	87.2	98.6	0.95	12.33
DIPPER-*L60*	49.2	83.6	0.90	9.85	65.2	89.8	0.90	11.05
Ours-*L20*	14.0	45.6	0.90	10.68	10.6	31.6	0.87	12.2
Ours-*L60*	2.8	15.8	0.85	10.20	1.6	7.6	0.81	11.4

Table 6: Performance of UNIGRAM-WATERMARK (Zhao et al., 2023) across different fractions of green list γ. We can observe that our attack is highly effective irrespective of the value of γ.

	UNIGRAM-WATERMARK				SIR			
Attack	TPR @ 1% FPR	TPR @ 10% FPR	P-SP	PPL	TPR @ 1% FPR	TPR @ 10% FPR	P-SP	PPL
No Attack	99.4	100.0	1.0	18.7	100.0	100.0	1.0	21.1
DIPPER-*L20*	96.2	99.0	94.6	13.7	87.5	98.4	0.95	15.3
DIPPER-*L60*	81.8	97.0	90.7	12.1	69.3	93.3	0.91	13.2
Ours -*L20*	4.2	13.6	0.88	14.4	11.1	43.1	0.90	15.6
Ours -*L60*	1.6	2.8	0.79	12.9	4.2	14.2	0.85	13.6

Table 7: Performance of UNIGRAM-WATERMARK and SIR against paraphrasing attacks. Pythia-1.4B is used as the base language model for all the experiments.

	UNIGRAM-WATERMARK				SIR			
Attack	TPR @ 1% FPR	TPR @ 10% FPR	P-SP	PPL	TPR @ 1% FPR	TPR @ 10% FPR	P-SP	PPL
No Attack	98.4	99.8	1.0	15.8	83.3	93.5	1.0	11.9
DIPPER-*L20*	87.4	98.0	0.95	11.8	48.0	72.2	0.95	9.6
DIPPER-*L60*	64.6	91.4	0.91	10.7	25.5	52.8	0.92	9.2
Ours -*L20*	5.4	30.9	0.89	12.5	1.3	6.6	0.9	10.4
Ours -*L60*	0.4	4.0	0.82	11.7	0.2	1.1	0.82	10.4

Table 8: Performance of UNIGRAM-WATERMARK and SIR against paraphrasing attacks. Mistral 7B is used as the base language model for all the experiments.

	UNIGRAM-WATERMARK				SIR			
Attack	TPR @ 1% FPR	TPR @ 10% FPR	P-SP	PPL	TPR @ 1% FPR	TPR @ 10% FPR	P-SP	PPL
No Attack	100	100	1	27.9	98.6	100.0	1.0	29.6
DIPPER (L20)	93.4	99.2	0.93	19.87	65.3	87.7	0.94	22.0
DIPPER (L60)	57.2	88.8	0.88	16.3	23.1	59.3	0.88	16.80
Ours (L20)	2.4 $\downarrow$	11.2 $\downarrow$	0.85	19.1	2.6 $\downarrow$	11.7 $\downarrow$	0.86	17.87

Table 9: Result demonstraing the efficacy of paraphrasing attacks on prompts from the arXiv papers dataset.

Attack	UNIGRAM-WATERMARK				SIR			
	TPR @ 1% FPR	TPR @ 10% FPR	P-SP	PPL	TPR @ 1% FPR	TPR @ 10% FPR	P-SP	PPL
No Attack	99.8	100.0	1.0	28.1	99.3	99.7	1.0	25.0
DIPPER (L20)	99.6	100.0	0.94	24.4	91.7	74	0.94	23.8
DIPPER (L60)	92.0	98.4	0.87	22.5	39.1	68.2	0.88	20.6
Ours (L20)	9.8 ↓	26.4 ↓	0.83	24.61	16.8 ↓	32.4 ↓	0.87	22.9

Table 10: Result demonstraing the efficacy of paraphrasing attacks on prompts from the Booksum dataset.