Watermark Removal: Preserving Semantics

Abhay Patidar & Gouri Shanker & Shreeman Agrawal
Indian Institute of Science
Bengaluru, KA, India
{abhaypatidar,gourishanker,shreemana}@iisc.ac.in

Abstract

As LLMs have grown to generate more human-like outputs major concerns like widespread misinformation, plagiarism, etc have grown considerable. Text generative models play a crucial role in various applications, but concerns about intellectual property and authenticity have led to the development of watermarking techniques to protect original content. We aim to check the performance of most recent watermarking techniques against common attacks like paraphrasing, re-translation, and also its performance on our proposed hybrid attacks.

1 Introduction

As LLMs have gained popularity and continued to grow it has become a huge concern to discern data produced by these LLMs and data produced by humans, as they lack any perceptible difference. A viable solution for this problem is to use a watermarking scheme which inject the text with an invisible signal without affecting generated text quality.

A watermarking scheme consists of a generation algorithm that is a modified version of the model in which the signal is planted and a detection algorithm that can detect whether a piece of output came from the watermarked model.Watermarks can be useful for applications like preventing AI-generated content from being used for training, making it more expensive or inconvenient to generate misinformation or cheat on assignments, and tracking the provenance of the precise text/image/etc, which the watermark was applied to.

We focus on the watermark introduced in Kirchenbauer et al. (2023), which describes a way to watermark and detect by introducing the concept of red and green lists. Yang et al. (2023) shows us another way to inject a watermark into text generated by black box LLMs. Liu et al. (2024) proposes a watermark technique called as Semantically Invariant Robust Watermark(SIR) which focuses on Semantics Consistent Broad range and Unbiased token preference.

We then check the performance of these aforementioned watermarks under baseline attack, and also on more recent attacks introduced in He et al. (2024) which explores a cross-lingual attack, Sadasivan et al. (2024) explores a recursive paraphrasing attack. We also propose Hybrid attacks which are the combination of the aforementioned attacks.

2 Related Work

2.1 Watermarking Schemes

Previously we have seen techniques which partitions sampling words into 2 lists and favours one of them for generation which however can be attacked using frequency based methods. This also fails to maintain contextual information when the watermark injection follows from strong watermark technique as mentioned in Kirchenbauer et al. (2023) (KGW).

Watermarking technique introduced in Yang et al. (2023) (BBW) which performs strong watermarking without changing the semantics using three metrics , namely sentence embedding similarity (S_{sent}), global word embedding similarity (S_{global}), and contextualized word embedding similarity ($S_{context}$). This performs very well as compared to other watermarking techniques. Although while attacking, random replacement of each word with semantically similar word reduces watermark confidence significantly.

Liu et al. (2024) (SIR) describes a method of watermarking which preserves the semantics by training another model to convert semantic embedding of a sentence into an equivalent logits using similarity of sentences as a loss metric, it then uses these logits to boost the original logits of original generative model. It uses the z-score to perform detection and performs well on baseline attacks.

2.2 Watermarking Attacks

Cross-lingual Watermark Removal Attack (CWRA/PT): As described in He et al. (2024), CWRA is a technique to weaken/remove any watermark by asking the model to generate output in a pivot language that has less or no similarities to our target language. We then take this watermarked text in pivot language and translate it into our desired language. Mandarin to English is the most basic version as both Mandarin and English have robust models and large datasets. We have also used the same set of languages.

Recursive Paraphrasing (RP): Introduced in Sadasivan et al. (2024) recursive paraphrasing as the name suggests takes in a watermarked text input then continues paraphrasing it for n-times, at n = 5 most watermarking schemes fail to recognise the input as watermarked. We employ parrot paraphraser to paraphrase at each iteration.

Re-translation Attack (RT): In this method, we perform a translation of the original watermarked text, initially generated in some language say English by a Language Model, into another pivot language. Subsequently, we revert this translated text back to its original language.

3 Proposed Attacks

Pivot Translation + Paraphrasing ($PT + Para$): Pivot Translation (we will use this for CWRA now onwards), then paraphrase the translated output once, we apply paraphrasing only once as it degrades the perplexity and with this method it does not result in much improvement doing paraphrasing recursively.

Re-translation + Paraphrase ($RT + Para$): It does re-translation on the watermarked text followed by paraphrasing as re-translation alone doesn't decrease watermark confidence by a significant fraction. Paraphrasing is done only once.

Recursive Paraphrase and Re-Translation Attack ($RP_i + RT$): We employ re-translation at the end of recursive paraphrasing, for different $i \in (1, 2, 3, 4, 5)$, this yields better performance than applying either techniques individually as we can see in the Results section.

4 Methodology

To generate outputs from LLM we use use the "xsum" dataset for English Prompts, along with the translated version of this dataset for the Chinese prompts. We use 100 prompts each of size 100 tokens and we used this to generate 300 token outputs.

We re-implemented the watermarking techniques as described in Kirchenbauer et al. (2023), Liu et al. (2024), and Yang et al. (2023), and modified them to use "Llama-2-7b" model (Touvron et al. (2023)) which supports both Mandarin and English, we made this modification so that we could apply the CWRA (Pivot Translation) attack which requires the model to generate output in a pivot language, and also to get a uniform set of results for each watermarking scheme and attack. We also used "Llama-2-7b" model to calculate the perplexities of the generated text, watermarked text and "attacked" text.

We then apply the watermark on both the English and Chinese prompts and test their robustness against the watermark attacks mentioned in section 3 and also the effect of each attack on the text quality using the metrics defined in Section 5.

5 Metrics

We conduct various evaluations on watermarked text both pre and post watermark attacks, employing diverse metrics.

5.1 Watermark Confidence:

Confidence, herein represented as a normalized z-score, offers a value within the range of 0 to 1, inclusively. It signifies the detection scheme's degree of certainty regarding the presence of a watermark within the input text.

5.2 Perplexity:

Perplexity is a measure often used to evaluate the quality of text generated by language models. It is a measure of how well a probability model predicts a sample. In the context of text generation, perplexity measures how well a language model predicts the next word in a sequence of words. A lower perplexity indicates that the language model is better at predicting the next word, and hence, the generated text is of higher quality. Higher perplexity values suggest that the language model is less accurate in its predictions, leading to lower-quality generated text.

5.3 BERTscore:

BERTScore Recall (Zhang et al. (2020)) measures how well the generated text captures the important information present in the reference text (i.e., the ground truth or the expected output). It evaluates the recall of n-grams (typically up to n=4) in the generated text compared to the reference text, using contextual embeddings from BERT to capture the meaning of words and phrases.

5.4 Word Edit Distance:

The word edit distance serves as a metric for assessing the alterations required—such as word deletions, insertions, or substitutions—to restore the original sentence. It quantifies the extent of modifications induced by any attack on the watermarked text, elucidating the magnitude of textual alterations post-attack.

5.5 ROC-AUC Curve:

ROC-AUC (Receiver Operating Characteristic Area Under the Curve) serves as a standard metric in binary classification tasks, assessing a model's ability to discern between positive and negative classes across various thresholds. It quantifies the balance between true positive rate (sensitivity) and false positive rate (1-specificity). The resulting area under the curve reflects the effectiveness of the watermark detection scheme post-watermarking and subsequent application of our proposed attacks.

6 Results

6.1 **Watermark Confidence**

Pivot Translation is one of the most prominent attack reducing the watermark confidence by a significant fraction. Figure 1 shows the average watermark confidence on different texts. Our proposed Pivot translation + Paraphrase attack performs better for SIR and BBW watermarking schemes and almost on par with the best attack for KGW watermarking scheme. For each of the already proposed attack, the corresponding new hybrid attack performed better.

The average watermark confidence on original generated text(without watermark) and on Pivot translation + Paraphrase are close which confirms that we are able to remove watermark almost completely.

	KGW	SIR	BBW
Original Text	16.70	28.60	50.02
Watermarked Text	93.64	83.48	93.00
Pivot Translation Attack	39.03	**25.92**	**52.59**
Re-translation Attack	82.45	61.81	82.97
Recursive Attack 0	60.20	61.10	91.25
Recursive Attack 1	36.85	51.58	78.19
Recursive Attack 2	26.89	44.34	68.12
Recursive Attack 3	22.78	41.08	66.84
Recursive Attack 4	**16.86**	36.24	62.53
Pivot Translation + Paraphrase Attack	**19.70**	**12.39**	**52.01**
Re-translation + Paraphrase Attack	41.73	11.26	64.99
Paraphrase 0 + Re-translation Attack	41.50	13.36	75.05
Paraphrase 1 + Re-translation Attack	31.71	12.95	66.75
Paraphrase 2 + Re-translation Attack	24.20	11.71	57.30
Paraphrase 3 + Re-translation Attack	21.47	13.56	57.71
Paraphrase 4 + Re-translation Attack	21.25	13.58	55.25

Figure 1: Watermark Confidence Against different Watermarking schemes and Attacks, each value is the mean confidence over 100 samples (each of length 300) in percentage. The light gray cells are the already existing attacks and the dark grey cells are our proposed hybrid attacks (lower is better).

6.2 **Data Perplexity**

6.2.1 On Watermarked Text

From Figure 2, it is clear that all three watermarks preserve text quality to some extent, we only see a small increase in perplexity for the KGW, SIR watermark although for BBW watermark Yang et al. (2023) there is a significant increase in perplexity as it watermarks the text after generation while the other 2 watermarks the text while generation which affects the generation maintaining the data quality to a more extent. Still all of them were able to preserves semantic and contextual meaning of the text. We analysed the Recursive paraphraser attack by paraphrasing the watermarked text upto 5 times and checked the text quality at each iteration, we can see in Figure 3, 4, 5 that the text quality (in terms of perplexity) degrades by a small amount but the watermark confidence decreases significantly as we continue paraphrasing. The translation and re-translation attack on the other hand doesn't degrade text quality yet is able to remove watermark which can be seen in Figure 1.

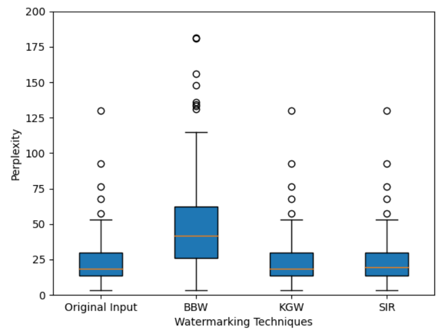

Figure 2: Perplexity Analysis Post Watermarking, as Calculated by Llama-2-7b

6.2.2 On Attacked Text

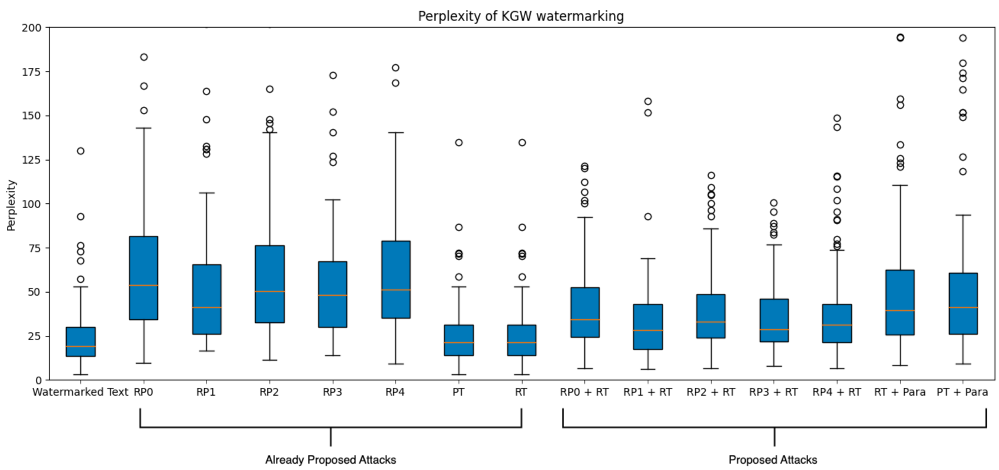

Figure 3: Perplexity Comparison for KGW Watermarking

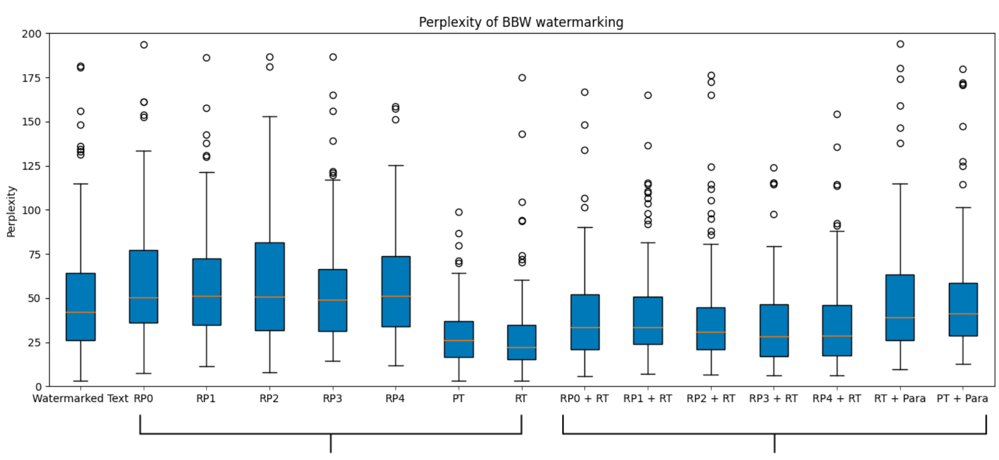

Figure 4: Perplexity Comparison for BBW Watermarking

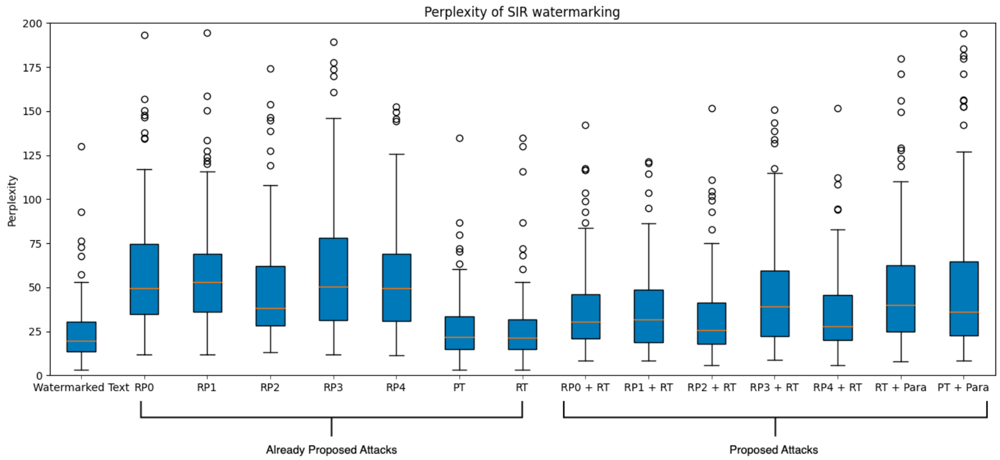

Figure 5: Perplexity Comparison for SIR Watermarking

Recursive Paraphrasing + Re-translation attacks have lower perplexities as compared to the Recursive paraphrasing attack. Although for Pivot translation + Paraphrase and Re-translation + Paraphrase, perplexity increases which indicates that paraphrasing reduces the data quality while translation has less impact on the data quality.

6.3 BERTScore - Recall

Figure 6 shows average BERTscore recall for different attacks on different watermarking schemes. Classical attacks demonstrate a good BERTScore signifying minimal decrease in text semantics, our proposed hybrid techniques perform similarly with a slight decrease in this score, while decreasing the confidence more than the classical attacks.

	KGW	SIR	BBW
Pivot Translation Attack	0.909	**0.895**	0.888
Re-translation Attack	**0.965**	0.869	**0.939**
Recursive Attack 0	0.927	0.847	0.926
Recursive Attack 1	0.908	0.845	0.907
Recursive Attack 2	0.896	0.844	0.896
Recursive Attack 3	0.889	0.842	0.890
Recursive Attack 4	0.883	0.842	0.885
Pivot Translation + Paraphrase Attack	0.878	**0.871**	0.871
Re-translation + Paraphrase Attack	0.918	0.843	0.911
Paraphrase 0 + Re-translation Attack	**0.934**	0.863	**0.923**
Paraphrase 1 + Re-translation Attack	0.922	0.861	0.911
Paraphrase 2 + Re-translation Attack	0.911	0.860	0.905
Paraphrase 3 + Re-translation Attack	0.904	0.858	0.900
Paraphrase 4 + Re-translation Attack	0.899	0.857	0.896

Figure 6: BERTscore Recall calculated w.r.t. the watermarked (Higher is better), range - [0,1]

6.4 Word Edit Distance

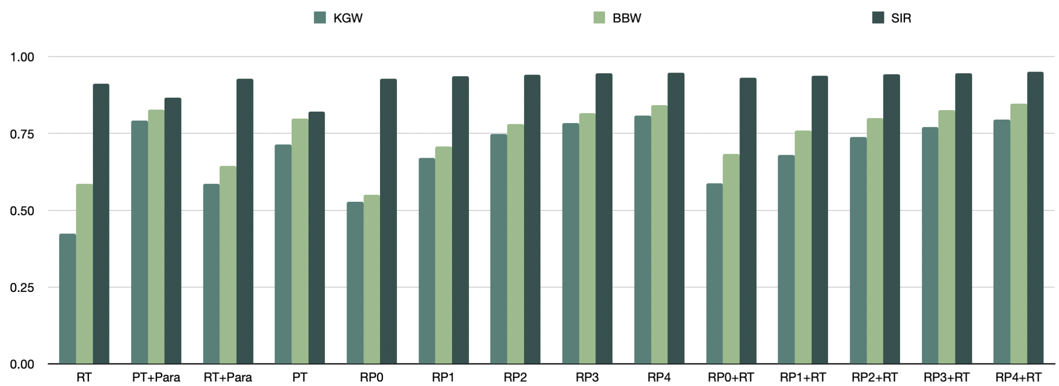

Figure 7: Average word edit distance, normalized by the length of the text for each sample.

In the context of word edit distance, the Re-Translation approach maintains a higher percentage of original words compared to Paraphrasing, as illustrated in Figure 7. This outcome aligns with expectations, as Paraphrasing involves substituting words and phrases to convey similar meanings while Re-Translation generally retains a larger portion of the original words while potentially swapping some.

6.5 ROC-AUC Curve

Following watermarking the ROC-AUC curves of all three watermarking schemes that we have considered demonstrate promising results with respective area under the curve of 0.88 for SIR, 0.92 for BBW, and 0.98 for KGW. We can see that already existing attacks (red, blue, brown line in Fig. 8, 9, 10) decrease this area by a significant amount. Pivot translation gives the best performance in general out of the already existing attacks we have explored.

Our proposed attacks perform better in terms of reducing the area under the curve than all the already existing attacks (pink, green and black line in Fig. 8, 9, 10). Out of the proposed attacks Pivot Translation + paraphrasing reduces the area the most and hence is the most effective.

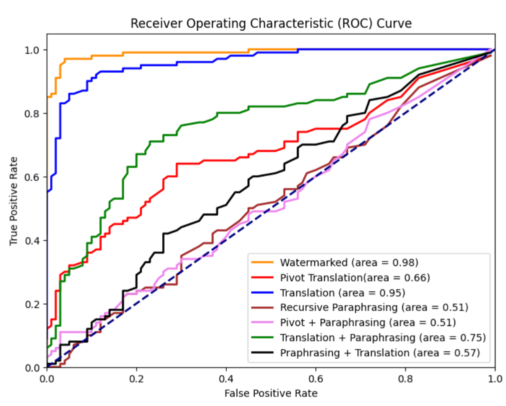

Figure 8: KGW Watermark

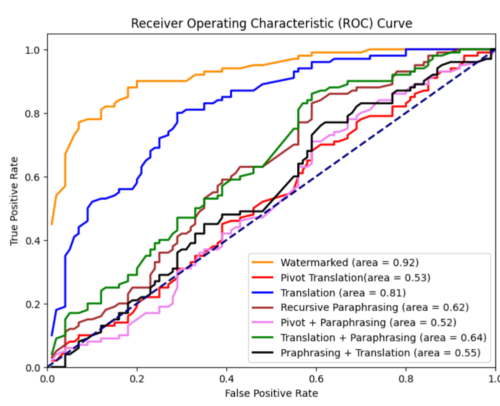

Figure 9: BBW Watermark

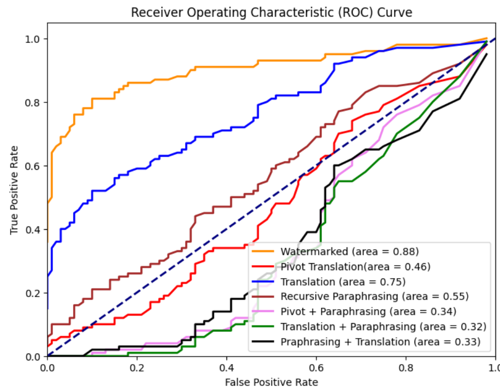

Figure 10: SIR Watermark

7 Conclusion

When comparing existing attacks with the proposed hybrid attacks, the Pivot Translation followed by Paraphrasing emerges as the most effective approach in maintaining data perplexity and text semantics while simultaneously reducing watermark confidence.

Performing Recursive Paraphrasing followed by Re-translation yields superior results compared to Recursive Paraphrasing alone. Re-translation proves effective in reducing the watermark while causing only a minimal increase in perplexity. Additionally, the ROC-AUC analysis indicates that post-attack, predictions align closely with random predictions for KGW and BBW, but deteriorate notably for SIR.

Our research has revealed the lack of robustness and reliability in existing watermarking techniques. Black box attacks have demonstrated the ability to substantially decrease watermark confidence, resulting in increased false negatives during watermark detection. There is a pressing need for more resilient watermarking schemes capable of withstanding cross-linguistic attacks.

Github Repo: https://github.com/gourishankerJK/WaterMark.git

References

Zhiwei He, Binglin Zhou, Hongkun Hao, Aiwei Liu, Xing Wang, Zhaopeng Tu, Zhuosheng Zhang, and Rui Wang. Can watermarks survive translation? on the cross-lingual consistency of text watermark for large language models, 2024.

John Kirchenbauer, Jonas Geiping, Yuxin Wen, Jonathan Katz, Ian Miers, and Tom Goldstein. A watermark for large language models, 2023. URL https://arxiv.org/abs/2301.10226.

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

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

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. Llama 2: Open foundation and fine-tuned chat models, 2023.

Xi Yang, Kejiang Chen, Weiming Zhang, Chang Liu, Yuang Qi, Jie Zhang, Han Fang, and Nenghai Yu. Watermarking text generated by black-box language models, 2023. URL https://arxiv.org/abs/2305.08883.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. Bertscore: Evaluating text generation with bert, 2020.