Temperature Matters: Enhancing Watermark Robustness Against Paraphrasing Attacks

Badr Youbi Idrissi[1], Monica Millunzi[2], Amelia Sorrenti[3], Lorenzo Baraldi[4], Daryna Dementieva[5]
[1]badryoubiidrissi@gmail.com [2]monica.millunzi@unimore.it
[3]lorenzo.baraldi@phd.unipi.it [4]amelia.sorrenti@gmail.com [5]daryna.dementieva@tum.de

Abstract

In the present-day scenario, Large Language Models (LLMs) are establishing their presence as powerful instruments permeating various sectors of society. While their utility offers valuable support to individuals, there are multiple concerns over potential misuse. Consequently, some academic endeavors have sought to introduce watermarking techniques, characterized by the inclusion of markers within machine-generated text, to facilitate algorithmic identification. This research project is focused on the development of a novel methodology for the detection of synthetic text, with the overarching goal of ensuring the ethical application of LLMs in AI-driven text generation. The investigation commences with replicating findings from a previous baseline study [1], thereby underscoring its susceptibility to variations in the underlying generation model. Subsequently, we propose an innovative watermarking approach and subject it to rigorous evaluation, employing paraphrased generated text to asses its robustness. Experimental results highlight the robustness of our proposal compared to the [1] watermarking method.

1. Introduction

In an era dominated by Large Language Models (LLMs), we stand at the intersection of unprecedented linguistic capabilities and profound ethical challenges. While the potential applications of LLMs are vast and promising, their exponential growth also brings to the forefront the serious concerns associated with their potential misuse [4, 9, 12]. From election rigging to social engineering campaigns using automated bots on social media platforms, from spreading fake news to dishonest use in academic and coding assignments, the dark side of LLMs has become increasingly apparent. As a result, the line between genuine human-generated content and machine-generated text is blurring, making accountability and transparency in the area of LLM-generated text urgent [3, 6, 8].

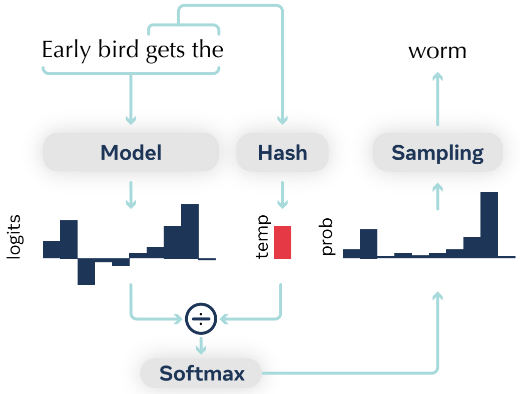

Figure 1. Overview of our temperature-based watermarking technique: we employ h tokens (in this example, 2) as the seed for a hashing process to sample a temperature parameter. This temperature parameter is subsequently applied to adjust the logits, for the prediction of the subsequent token.

To address this critical issue, our project explores the concept of watermarking, which involves the subtle incorporation of imperceptible markers, known as *watermarks*, into machine-generated text. Watermarking enables the algorithmic identification of machine-generated text while remaining largely undetectable to human readers [10, 11, 14]. The aim is not only to facilitate the identification of machine-generated text, but also to minimize false positives, ensuring that authentic human content remains unaltered.

To summarize, the main contributions of our work are the following:

- We replicate the watermarking detection results presented in [1] with open-source model.
- We propose a new approach for watermarking which changes the temperature T of sampling at each token.
- We test our approach on the robustness to the paraphrasing attack, i.e. if some part of the watermarked

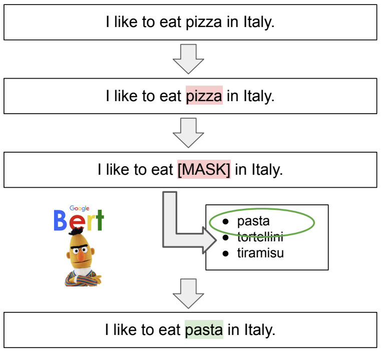
I like to eat pizza in Italy.
I like to eat pizza in Italy.
I like to eat [MASK] in Italy.
Google
Bert
pasta
~~tortellini~~
tiramisu
I like to eat pasta in Italy.

Figure 2. An example of text paraphrasing attack utilizing BERT [2], i.e. substituting randomly selected words with model predictions.

text is paraphrased.

2. Method

In this section, we present our novel watermarking technique tailored for Large Language Models. To provide a comprehensive delineation of our approach, we commence by establishing the necessary notational framework and subsequently elaborate on the employed architectural configurations.

2.1. Notation and Preliminaries

Given a word vocabulary of $|V|$ granularity, the Large Language Model (LLM) for next-word prediction is a function f_θ, often parameterized by a neural network, which takes in input a sequence T of tokens denoted as $s^{(t)} \in V^T$ and outputs a vector of V logits, one for each word in the vocabulary. These logits are then passed through a softmax operator to convert them into a discrete probability distribution over the vocabulary. The next token is commonly sampled from this distribution using a sampling strategy of choice. For notational purposes, tokens with negative indices, $s^{(-N_p)}, \cdots, s^{(-1)}$, indicate a prompt characterized by a length of N_p, while $s^{(0)}, \cdots, s^{(T)}$ correspond to tokens produced by an AI system as a response to the given prompt.

2.2. Model

In this project, we employed the Vicuna-7B model [5], which is a LLaMA [13] model fine-tuned on user-shared conversations collected from ShareGPT, a ChatGPT dialogue corpus crawled from sharegpt.com.

2.3. Watermarking Generation

While [10] watermarking method randomly partitioned the vocabulary into two lists of equal size, namely the "green list" and the "red list", we have opted for an alternative methodology by exclusively considering the logits vector obtained as the model's output.

Algorithm 1 Proposed Temperature Watermarking

```
Input: prompt, $s^{(-N_p)}, \cdots, s^{(-1)}$,
n.o. tokens to use for hashing $h$, model $f_\theta$
for $t = 0, 1, \cdots, t$ do
    1. Apply the language model to prior tokens $s^{(-N_p)} \dots s^{(t-1)}$ and retain the logits $l^{(t)}$ over the vocabulary.
    2. Compute a hash of the last $h$ tokens of the prompt $s^{(t-h)} \cdots s^{(t-1)}$ and use it to seed a random number generator.
    3. Using this seed, randomly generate a temperature value $T^t \sim T_0(m + (M - m)\mathcal{U}^t)$
    4. Rescale the logits $l^{(t)} = l^{(t)}/T^t$
    5. Convert the logits to a probability vector using the softmax operator $p_k^{(t)} = \exp(l_k^{(t)})/\sum_i \exp(l_i^{(t)})$.
    6. Sample using this probability distribution
end for
```

Our approach uses a context window of size h to generate a unique hash. This hash is then used as a seed for generating a random temperature T^t for each token. This temperature is sampled as follows :

$$T^t \sim T_0(m + (M - m)\mathcal{U}^t)$$

Where U is a uniform pseudo random variable seeded by the hash of the tokens $[s^{t-h}, \dots, s^{t-1}]$. In other words, U is a deterministic function of the context window. This will allow us to recompute the same temperatures when detecting the watermark later on. The temperature varies around the base value T_0. For some tokens the temperature will be higher and for others it will be lower. We can then compute the probabilities of each token in the vocabulary using the softmax function.

$$P(s^t = k | s^1, \dots, s^{t-1}; T^t) = \frac{\exp(l_k^{(t)}/T^t)}{\sum_i \exp(l_i^{(t)}/T^t)}$$

Since the temperature controls the entropy of the output distribution, this forces the model to take a specific path during generation that allows us to distinguish it from human generated text later on.

The resulting vector of probabilities is then used for sampling with any desired method. A detailed version of the algorithm is presented in Alg. 1.

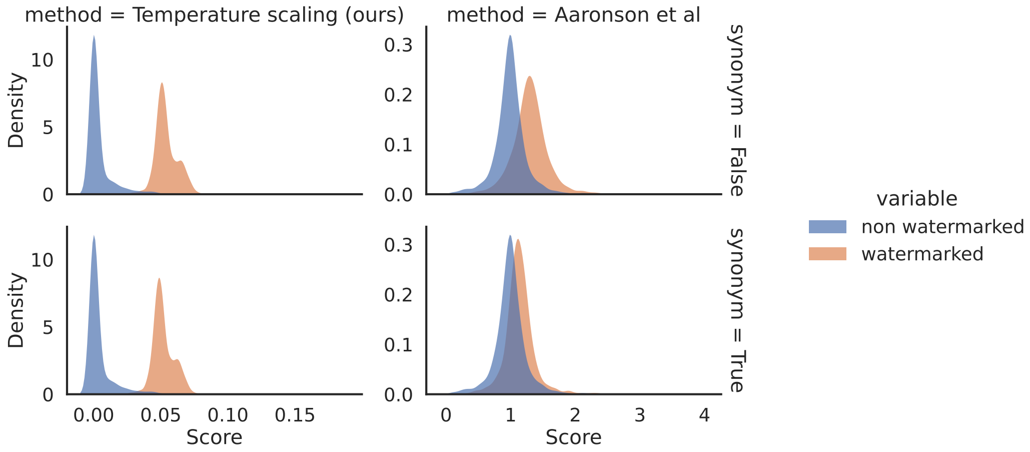

Figure 3. Score of our temperature-based watermarking (left column) technique compared to the baseline method (right column). In the first row both methods are compared without paraphrasing attacks, on the second row BERT [2] is applied to 30% of the generated input tokens.

2.4. Watermarking Detection

The outcome of the process described above is that the model's confidence in its predictions varies in a known way throughout generation. This variation in confidence across a specific sequence of tokens is what ultimately constitutes the watermark.

To detect the watermark, it is sufficient to recompute the same probabilities as in the watermarking phase, by doing a forward pass on the text, then select the probabilities of the tokens, and average them. Eq. 1.

$$\text{Score} = \frac{1}{N}\sum_{t=1}^{N} P(s^t|s^1,\ldots,s^{t-1};T^t) \tag{1}$$

This score can be then used to determine whether the sequence is watermarked or not. Indeed, sequences with no watermarking are likely to have a smaller score than watermarked sequences.

3. Evaluation Setup

3.1. Baseline Reproducibility

Firstly, we decide to reprocude the machine-generated papers detection with already published SOTA approach. For this, as for baseline, we refer to the method presented in [1]. There, the experiments were conducted on LLaMa [13] which is available upon the request. We substitute the model with the opensource model—Vicuna[1]—with has the same amount of hyper-parameters and fairly compared performance on the benchmarks.

Thus, with this experiment, we want to test the following ***Hypothesis H1***: if the published results of machine-generated texts detection are reproducible with the open-source model. While in [10] the theoretical proof of watermarking model-agnosticism was provided, it still should be verified empirically.

3.2. Paraphrasing Attack

One of the main attacks on the watermarking can be the paraphrasing of some parts of machine-generated watermarked text [11]. In [7], the significant drop of their detection method performance on the paraphrased texts was observed. Thus, we formulate the next ***Hypothesis H2***: if our proposed watermarking method is robust to the paraphrasing attacks.

To imitate the paraphrasing process, we utilize BERT[2] [2] for masked language modeling (Figure 2). For each text sample, we replace 30% of the tokens with `[MASK]` and ask the model for the prediction. Each `[MASK]` prediction was done one-by-one to save the content.

3.3. Dataset and Metric

For the test set, we use random 1k samples fro Alpaca dataset.[3] This dataset consists of the prompts, human answers, and machine-generated texts. We generate our own texts with Vicuna and the proposed watermarking method inputting the prompts and comparing them with human-written ones from the dataset.

We report the F1 score, True Positive Rate (TPR), and False Positive Rate (FPR) as a main evaluation metric as

[1] https://huggingface.co/lmsys/vicuna-7b-v1.3

[2] https://huggingface.co/bert-base-uncased

[3] https://huggingface.co/datasets/tatsu-lab/alpaca

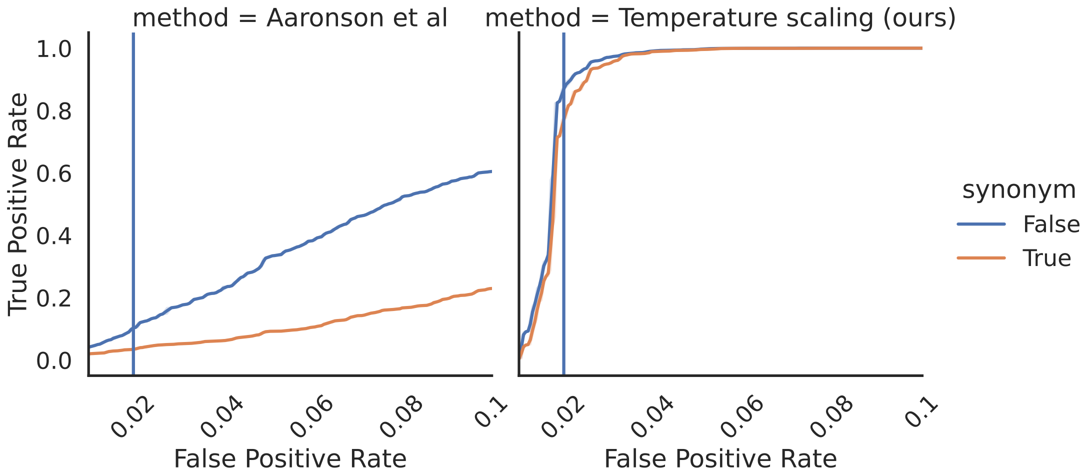

Figure 4. Receiver Operating Characteristic (ROC) curve for the baseline [1] approach (left) and our proposed approach (right). *synonym* indicate if the paraphrasing attack was applied or not.

with our work we are aiming to increase the probability of machine-generated texts detection in comparison to the previous approach.

4. Results

Firstly, we illustrate the comparison with the baseline approach on the first row of Figure 3. We can observe that in the baseline approach (right column) it is difficult to differentiate between distributions for human-written and machine-generated text. On the other hand in the proposed approach (left column), the differences in the distributions are clear making it easy to separate machine-generated textual descriptions. Thus, previously stated ***H1*** is not confirmed. The reasons for this are the grounds for further investigation.

For **H2**, we are investigating the robustness to the paraphrasing attack of the proposed technique. The intuition in the proposed watermark technique lies in the idea that the change in temperature will, at some point, allow the model to take into account not only the most probable word on the step but also its synonyms. In the second row of Figure 3, the analysis with attacked generated texts is represented. In this case, the previous observation on the distinction of human and machine-generate texts is even more accentuated. Indeed our model seems invariant to the attack while baseline-processed text is nearly overlapped with human input making distinction difficult.

Furthermore, a quantitative assessment has been carried out using the Receiver Operating Characteristic (ROC) curve. As depicted in Figure 4, it is evident that the temperature watermarking method we propose outperforms the baseline approach at all stages of evaluation. To illustrate, when the False Positive Rate (FPR) is held constant at 2%, the baseline method achieves a True Positive Rate (TPR) of 15%, whereas our proposed method attains a TPR of 90%. Moreover, this performance advantage holds true even when subjected to a paraphrasing attack. This proves **H2**—our proposed temperature watermark is more robust to the paraphrasing attack—making further analysis possible in future works.

5. Conclusion

In this work, we investigated the robustness of the watermarking approach for machine-generated texts. Firstly, we found out that reproducibility of the detection results based on one model is challenging on the other model while the method is claimed to be model agnostic.

Then, we proposed a new watermark which generates a unique temperature at each step of generation. This approach made the watermark detection easier and more robust on the Vicuna model. We also considered paraphrasing attack as one of the main attacks on the machine-generated texts detector. In our proposed watermarking method, this attack is addressed outperforming the baseline. The method allows to choose at some steps within more options thus take into account more suitable generation options. We understand that the thorough investigation of the proposed method is still to be addressed in the future work.

References

[1] S. Aaronson and H. Kirchner. Watermarking gpt outputs. `https://www.scottaaronson.com/talks/watermark.ppt`, 2023.

[2] Shivaji Alaparthi and Manit Mishra. Bidirectional encoder representations from transformers (bert): A sentiment analysis odyssey. *arXiv preprint arXiv:2007.01127*, 2020.

[3] Emily M Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. On the dangers of stochastic parrots: Can language models be too big? In *Proceedings of the 2021 ACM conference on fairness, accountability, and transparency*, pages 610–623, 2021.

[4] A Stevie Bergman, Gavin Abercrombie, Shannon Spruit, Dirk Hovy, Emily Dinan, Y-Lan Boureau, Verena Rieser, et al. Guiding the release of safer e2e conversational ai through value sensitive design. In *Proceedings of the 23rd Annual Meeting of the Special Interest Group on Discourse and Dialogue*. Association for Computational Linguistics, 2022.

[5] Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E Gonzalez, et al. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality. *See https://vicuna. lmsys. org (accessed 14 April 2023)*, 2023.

[6] Evan Crothers, Nathalie Japkowicz, and Herna L Viktor. Machine-generated text: A comprehensive survey of threat models and detection methods. *IEEE Access*, 2023.

[7] Pierre Fernandez, Antoine Chaffin, Karim Tit, Vivien Chappelier, and Teddy Furon. Three bricks to consolidate watermarks for large language models. *CoRR*, abs/2308.00113, 2023.

[8] Alexei Grinbaum and Laurynas Adomaitis. The ethical need for watermarks in machine-generated language. *arXiv preprint arXiv:2209.03118*, 2022.

[9] Katarina Kertysova. Artificial intelligence and disinformation: How ai changes the way disinformation is produced, disseminated, and can be countered. *Security and Human Rights*, 29(1-4):55–81, 2018.

[10] John Kirchenbauer, Jonas Geiping, Yuxin Wen, Jonathan Katz, Ian Miers, and Tom Goldstein. A watermark for large language models. *arXiv preprint arXiv:2301.10226*, 2023.

[11] John Kirchenbauer, Jonas Geiping, Yuxin Wen, Manli Shu, Khalid Saifullah, Kezhi Kong, Kasun Fernando, Aniruddha Saha, Micah Goldblum, and Tom Goldstein. On the reliability of watermarks for large language models. *arXiv preprint arXiv:2306.04634*, 2023.

[12] Yisroel Mirsky, Ambra Demontis, Jaidip Kotak, Ram Shankar, Deng Gelei, Liu Yang, Xiangyu Zhang, Maura Pintor, Wenke Lee, Yuval Elovici, et al. The threat of offensive ai to organizations. *Computers & Security*, page 103006, 2022.

[13] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. Llama: Open and efficient foundation language models, 2023.

[14] Yuxin Wen, John Kirchenbauer, Jonas Geiping, and Tom Goldstein. Tree-ring watermarks: Fingerprints for diffusion images that are invisible and robust. *arXiv preprint arXiv:2305.20030*, 2023.