Can Watermarking Large Language Models Prevent Copyrighted Text Generation and Hide Training Data?

Michael-Andrei Panaitescu-Liess, Zora Che, Bang An, Yuancheng Xu, Pankayaraj Pathmanathan, Souradip Chakraborty, Sicheng Zhu, Tom Goldstein, Furong Huang

University of Maryland, College Park
{mpanaite, zche, bangan, ycxu, pan, schakra3, sczhu, tomg, furongh}@umd.edu

Abstract

Large Language Models (LLMs) have demonstrated impressive capabilities in generating diverse and contextually rich text. However, concerns regarding copyright infringement arise as LLMs may inadvertently produce copyrighted material. In this paper, we first investigate the effectiveness of watermarking LLMs as a deterrent against the generation of copyrighted texts. Through theoretical analysis and empirical evaluation, we demonstrate that incorporating watermarks into LLMs significantly reduces the likelihood of generating copyrighted content, thereby addressing a critical concern in the deployment of LLMs. However, we also find that watermarking can have unintended consequences on Membership Inference Attacks (MIAs), which aim to discern whether a sample was part of the pretraining dataset and may be used to detect copyright violations. Surprisingly, we find that watermarking adversely affects the success rate of MIAs, complicating the task of detecting copyrighted text in the pretraining dataset. These results reveal the complex interplay between different regulatory measures, which may impact each other in unforeseen ways. Finally, we propose an adaptive technique to improve the success rate of a recent MIA under watermarking. Our findings underscore the importance of developing adaptive methods to study critical problems in LLMs with potential legal implications.

Introduction

In recent years, Large Language Models (LLMs) have pushed the frontiers of natural language processing by facilitating sophisticated tasks like text generation, translation, and summarization. With their impressive performance, LLMs are increasingly integrated into various applications, including virtual assistants, chatbots, content generation, and education. However, the widespread usage of LLMs brings forth serious concerns regarding potential copyright infringements. Addressing these challenges is critical for the ethical and legal deployment of LLMs.

Copyright infringement involves unauthorized usage of copyrighted content, which violates the intellectual property rights of copyright owners, potentially undermining content creators' ability to fund their work, and affecting the diversity of creative outputs in society. Additionally, violators can face legal consequences, including lawsuits and financial penalties. For LLMs, copyright infringement can occur through (1) generation of copyrighted content during deployment and (2) illegal usage of copyrighted works during training. Ensuring the absence of copyrighted content in the vast training datasets of LLMs is challenging. Moreover, legal debates around generative AI copyright infringement vary by region, complicating compliance further.

Current lawsuits against AI companies for unauthorized use of copyrighted content (e.g., Andersen v. Stability AI Ltd, NYT v. OpenAI) highlight the urgent need for methods to address these challenges. In this paper, we focus on studying the effects of watermarking LLMs on two critical issues: (1) preventing the generation of copyrighted content, and (2) detecting copyrighted content in training data. We show that watermarking can significantly impact both the generation of copyrighted text and the detection of copyrighted content in training data.

Firstly, we observe that current LLM output watermarking techniques can significantly reduce the probability of LLMs generating copyrighted content, by tens of orders of magnitude. Our empirical results focus on two recent watermarking methods: UMD (Kirchenbauer et al. 2023) and Unigram-Watermark (Zhao et al. 2023). Both methods split the vocabulary into two sets (green and red) and bias the model towards selecting tokens from the green set by altering the logits distribution, thereby embedding a detectable signal. We provide both empirical and theoretical results to support our findings.

Secondly, we demonstrate that watermarking techniques can decrease the success rate of Membership Inference Attacks (MIAs), which aim to detect whether a piece of copyrighted text was part of the training dataset. Since MIAs exploit the model's output, their performance can suffer under watermarking due to changes in the probability distribution of output tokens. Our comprehensive empirical study, including 5 recent MIAs and 5 LLMs, shows that the AUC of detection methods can be reduced by up to 16% in the presence of watermarks.

Finally, we propose an adaptive method designed to

Figure 1: Illustration of the effect of LLM watermarking on generation of copyrighted content. We observe that watermarking can make it more than 10^{20} times less likely for Llama-30B to generate copyrighted content.

enhance the success rate of a recent MIA (Shi et al. 2023) in detecting copyright violations under watermarking. This method applies a correction to the model's output to account for the perturbations introduced by watermarks. By incorporating knowledge about the watermarking scheme, we improve the detection performance for pretraining data, counteracting the obfuscation caused by watermarking. Our contribution underscores the importance of continuously developing adaptive attack methodologies to keep pace with advances in defense mechanisms.

The rest of the paper is organized as follows. In the "Related Work" section, we review prior research on LLM watermarking and copyright. The "Setup and Notations" section formally introduces the problems we study. We then present our first two contributions and introduce the adaptive version of the Min-K% Prob membership inference attack in the following three sections. Finally, we provide concluding remarks in the last section. Additional experiments, theoretical results, and a discussion on the limitations of our work are included in the appendix [1].

Related Work

Watermarks for LLMs. Language model watermarking techniques embed identifiable markers into output text to detect AI-generated content. Recent strategies incorporate watermarks during the decoding phase of language models (Zhao et al. 2023; Kirchenbauer et al. 2023). Aaronson (2023) develops the Gumbel watermark, which employs traceable pseudo-random sampling for generating subsequent tokens. Kirchenbauer et al. (2023) splits the vocabulary into red and green lists according to preceding tokens, biasing the generation towards green tokens. Zhao et al. (2023) employs a fixed grouping strategy to develop a robust watermark with theoretical guarantees. Liu et al. (2024) proposes to generate watermark logits based on the preceding tokens' semantics rather than their token IDs to boost the robustness. Kuditipudi et al. (2023) and Christ, Gunn, and Zamir (2023) explore watermark methods that do not change the output textual distribution.

Copyright. Copyright protection in the age of AI has gained importance, as discussed by Ren et al. (2024). Vyas, Kakade, and Barak (2023) addresses content protection through near access-freeness (NAF) and developed learning algorithms for generative models to ensure compliance under NAF conditions. Prior works focus on training algorithms to prevent copyrighted text generation (Vyas, Kakade, and Barak 2023; Chu, Song, and Yang 2024), whereas our work emphasizes lightweight, inference-time algorithms. Other works have studied copyright in machine learning from a legal perspective. Hacohen et al. (2024) utilizes a generative model to determine the generic characteristics of works to aid in defining the scope of copyright. Elkin-Koren et al. (2023) demonstrates that copying does not necessarily constitute copyright infringement and argues that existing detection methods may detract from the foundational purposes of copyright law.

Additionally, we include a discussion on memorization and membership inference in the appendix.

Setup and Notations

Definitions

Let D be a training dataset, C be all the copyrighted texts, and C_D be all the copyrighted texts that are part of D. We give definitions for the following setups.

Verbatim Memorization of Copyrighted Content. For a fixed $k \in \mathbb{N}$, Carlini et al. (2022) defines a string s as being memorized by a model if s is extractable with a prompt p of length k using greedy decoding and the concatenation $p \oplus s \in D$. We adopt a similar definition for verbatim memorization of copyrighted content but employ a continuous metric to measure it. Specifically, we measure verbatim memorization of a text $c \in C$ using the perplexity of the model on the copyrighted text c_p when given the prefix p as a prompt (where c_p represents the text c after removing its prefix p). Note that for $c_p = c_p^{(1)} \oplus c_p^{(2)} \oplus \cdots \oplus c_p^{(n)}$ we compute the perplexity using the following formula $\texttt{perplexity}(c_p|p) = \left(\prod_{i=1}^{n} \mathbb{P}(c_p^{(i)}|p \oplus c_p^{(0)} \oplus c_p^{(1)} \oplus \cdots \oplus c_p^{(i-1)})\right)^{-\frac{1}{n}}$, where $c_p^{(0)}$ is the empty string. In our experiments, p is either an empty string or the first 10, 20, or 100 tokens of c. Lower perplexity thereby indicate higher levels of memorization.

MIAs for Copyrighted Training Data Detection. MIAs are privacy attacks aiming to detect whether a sample was part of the training set. We define an MIA for copyrighted data as a binary classifier $A(\cdot)$, which ideally outputs $A(x) = 1, \forall x \in C_D$ and $A(x) = 0, \forall x \in C - C_D$. In practice, $A(\cdot)$ is defined by thresholding a metric (e.g., perplexity), i.e., $A(x) = 1, \forall x$ such that $\texttt{perplexity}(x) < t$ and 0, otherwise. Since the threshold t needs to be set, prior work (Shi et al. 2023) uses AUC (Area Under the ROC

[1]The appendix is available in the arXiv version of the paper (https://arxiv.org/abs/2407.17417).

Curve) as an evaluation metric which is independent of t. Note that we employ the same metric in our experiments.

LLM Watermarking. Watermarking LLMs consists of introducing signals during its training or inference that are difficult to detect by humans without the knowledge of a *watermark key* but can be detected using an algorithm if the key is known. We focus our paper on recent methods that employ logits distribution changes as a way of inserting watermark signals during the decoding process (Kirchenbauer et al. 2023; Zhao et al. 2023).

MIAs

Current MIAs for detecting training data rely on thresholding various heuristics that capture differences in output probabilities for each token between data included in the training set and data that was not. Below, we present an overview of these heuristics.

Perplexity. This metric distinguishes between data used to train the model (members) and data that was not (non-members), as members are generally expected to have lower perplexity.

Smaller Ref, Lowercase and Zlib (Carlini et al. 2021). Smaller Ref is defined as the ratio of the log-perplexity of the target LLM on a sample to the log-perplexity of a smaller reference LLM on the same sample. Lowercase represents the ratio of the log-perplexity of the target LLM on the original sample to the log-perplexity of the LLM on the lowercase version of the sample. Zlib is defined as the ratio of the log-perplexity of the target LLM on a sample to the zlib entropy of the same sample.

Min-K% Prob (Shi et al. 2023). This heuristic computes the average of the minimum $K\%$ token probabilities outputted by the LLM on the sample. Note that this method requires tuning K, so in all our experiments we chose the best result over $K\% \in \{5\%, 10\%, 20\%, 30\%, 40\%, 50\%, 60\%\}$.

LLM Watermarking Methods

UMD (Kirchenbauer et al. 2023) splits the vocabulary into two sets (green and red) and biases the model towards the green tokens by altering the logit distribution. The hash of the previous token's ID serves as a seed for a pseudo-random number generator used to split the vocabulary into these two groups. For a "hard" watermark, the model is forced not to sample from the red list at all. For a "soft" watermark, a positive bias δ is added to the logits of the green tokens before sampling. We focus our empirical evaluation on "soft" watermarks as they are more suitable for LLM deployment due to their smaller impact on the quality of the generated text.

Unigram-Watermark (Zhao et al. 2023) employs a similar approach of splitting the vocabulary into two sets and biasing the model towards one of the two sets. However, the split remains consistent throughout the generation. This choice is made to provide a provable improvement against paraphrasing attacks (Krishna et al. 2024).

Watermarking LLMs Prevents Copyrighted Text Generation

In this section, we study the effect of LLM watermarking techniques on verbatim memorization. We discuss the their implications for preventing copyrighted text generation.

Datasets. We consider 4 versions of the WikiMIA benchmark (Shi et al. 2023) with 32, 64, 128, and 256 words in each sample and only consider the samples that were very likely part of the training set of all the models we consider (labeled as 1 in Shi et al. (2023)). We consider these subsets as a proxy for text that was used in the training set, and the model may be prone to verbatim memorization. From now on, we refer to this subset as the "training samples" or "training texts". Similarly, we consider BookMIA dataset (Shi et al. 2023), which contains samples from copyrighted books.

Metric. We measure the relative increase in perplexity on the generation of training samples by the watermarked model compared to the original model. We report the increase in both the minimum and average perplexity over the training samples. Note that a large increase in perplexity corresponds to a large decrease in the probability of generating that specific sample, as shown later in this section. When computing the perplexity, we prompt the model with an empty string, the first 10, and the first 20 tokens of the targeted training sample, respectively. In the BookMIA dataset, we designate the initial 100 or 256 tokens as the prompt. This is because each BookMIA sample contains 512 words, which is larger than the sample size in WikiMIA.

Models. We conduct our empirical evaluation on 5 recent LLMs: Llama-30B (Touvron et al. 2023), GPT-NeoX-20B (Black et al. 2022), Llama-13B (Touvron et al. 2023), Pythia-2.8B (Biderman et al. 2023) and OPT-2.7B (Zhang et al. 2022).

Empirical Evaluation

In Table 1, we show the increase in perplexity on the training samples when the model is watermarked relative to the unwatermarked model. We observe that for Llama-30B, Unigram-Watermark induces a relative increase of 4.1 in the minimum and 34.1 in the average perplexity. Note that a relative increase of 4.1 in perplexity for a sample makes it more than 4.3×10^{22} times less likely to be generated. This is based on a sample with only 32 tokens, which is likely a lower bound since the number of tokens is typically larger than the number of words. We observe consistent results over several models and prompt lengths. For all experiments, unless otherwise specified, we use a fixed strength parameter $\delta = 10$ for watermark methods and a fixed percentage of 50% green tokens. All the results are averaged over 5 runs with different seeds for the watermark methods. We include additional results on WikiMIA-64,

		Llama-30B		Llama-13B	
	P.	Min.	Avg.	Min.	Avg.
UMD	0	3.3	31.2	4.9	34.3
	10	2.8	28.7	3.5	31.9
	20	2.4	30.1	3.5	33.4
Unigram	0	4.1	34.1	5.0	36.6
	10	3.0	31.7	4.0	34.3
	20	2.4	31.5	3.4	34.0

Table 1: Measuring the reduction in verbatim memorization of training texts on WikiMIA-32. We report the relative increase in both the minimum and average perplexity between the watermarked and unwatermarked models, where larger values correspond to less memorization. Note that "P." stands for "prompt length".

		Llama-30B		Llama-13B	
	P.	Min.	Avg.	Min.	Avg.
UMD	0	1.5	33.7	2.4	41.2
	10	1.5	33.6	2.3	41.0
	20	1.4	33.5	2.3	40.8
	100	1.3	32.9	1.9	40.3
Unigram	0	1.6	36.4	2.4	44.5
	10	1.6	36.3	2.4	44.3
	20	1.5	36.1	2.3	44.2
	100	1.4	35.5	1.8	43.6

Table 2: Measuring the reduction in verbatim memorization of training texts on BookMIA. We report the relative increase in both the minimum and average perplexity between the watermarked and unwatermarked models, where larger values correspond to less memorization. Note that "P." stands for "prompt length".

WikiMIA-128 and WikiMIA-256 in Tables 7, 8 and 9, respectively, in the appendix. We observe that our findings are consistent across models and splits of WikiMIA. Finally, we include the complete version of Table 1 in the appendix (Table 6), which shows results for additional models and random logit perturbations with the same strength as the watermarking methods. Overall, the additional results are consistent with our previous findings.

In Figure 2, as well as Figure 5 from the appendix, we study the influence of the strength of the watermark δ on the relative increase in both the minimum and average perplexity on the WikiMIA-32 training samples. In this experiment, we also consider a baseline of generating text freely to study the impact of watermarks on the quality of text relative to the impact on training samples' generation (here, perplexity is computed by an unwatermarked model). All the results are averaged over 5 runs with different seeds for the watermark methods. In the case of free generation, we generate 100 samples for 5 different watermarking

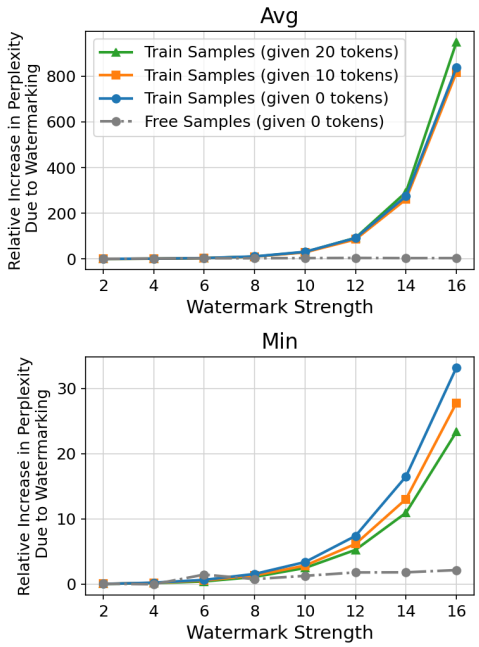

Figure 2: We study how the watermark strength (under the UMD scheme) affects the average and the minimum perplexity of training samples from WikiMIA-32, as well as the quality of generated text.

seeds and average the results. The length of the generated samples is up to 42 tokens, which is approximately 32 words in the benchmark (on a token-to-word ratio of 4 : 3). **The results show an exponential increase in the perplexity of the training samples with the increase in watermark strength, while the generation quality is affected at a slower rate.** This suggests that even if there is a trade-off between protecting the generation of text memorized verbatim and generating high-quality text, **finding a suitable watermark strength for each particular application is possible**. Examples of generated samples at varying watermark strengths are provided in the appendix.

Approximate Memorization. Informally, we consider a training sample approximately memorized by a model if, given its prefix, it is possible to generate a completion that is similar enough to the ground truth completion. In our experiments, we use models fine-tuned on a subset of BookMIA (details provided in the appendix) and we consider Normalized Edit Similarity (referred to as edit similarity from now on) and BLEU score as similarity measures, as in (Ippolito et al. 2023). Note that we consider both word-level and token-level variants for the BLEU score. The range for each metric is between 0 and 1, where values close to 1 represent similar texts. In all experiments, since all the samples are 512 words long, we consider the first 256 words as the prefix and the last 256 words as the ground truth completion. We present the results for edit similarity with the UMD watermark in Figure 3, and

the complete results—using all metrics and including the Unigram watermark—in Figure 7, averaged over 20 runs with different random seeds. Note that the duplication factor (shown on x-axis) represents the number of times the target copyrighted text is duplicated. We observe that for high levels of memorization, a strong watermark significantly reduces the similarity between the generated completion and the ground truth (copyrighted) one.

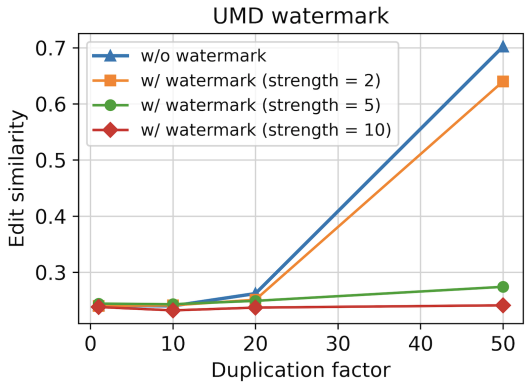

Figure 3: Edit similarity between the generated completion and the ground truth when considering different watermark strengths and memorization levels.

Takeaways. Watermarking significantly increases the perplexity of generating training texts, reducing verbatim memorization likelihood. This is achieved with only a moderate impact on the overall quality of generated text. This suggests that watermark strength can be effectively tailored to balance verbatim memorization and text quality for specific applications. Finally, we believe that our findings on WikiMIA—which does not necessarily contain copyrighted data—directly extend to the generation of copyrighted text verbatim, as this constitutes a form of verbatim memorization of the training data. To confirm, we run similar experiments on a dataset containing copyrighted data (BookMIA) and include the results in the Table 2. Additionally, we consider finetuning Llama-7B (Touvron et al. 2023) on BookMIA while controlling memorization by duplicating training samples. Detailed information about this experiment is provided in the appendix.

Impact of Watermarking on Pretraining Data Detection

Datasets. We revisit the WikiMIA benchmark as discussed in the previous section. We consider the full datasets, rather than the subset of samples that were part of the training for models we study. Additionally, we consider the BookMIA benchmark, which contains copyrighted texts.

Metrics. We follow the prior work (Shi et al. 2023; Duarte et al. 2024) and report the AUC and AUC drop to study the detection performance of the MIAs. Note that this metric has the advantage of not having to tune the threshold for the detection classifier.

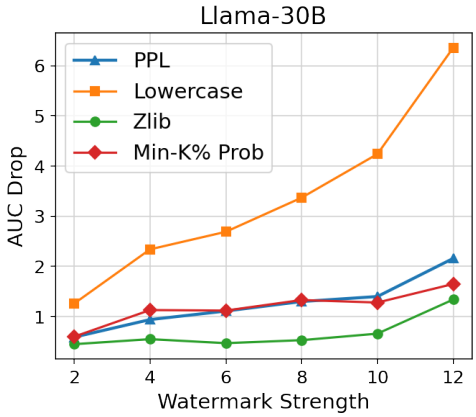

Figure 4: AUC drop due to watermarking for each MIA when varying the strength of the watermark.

Models. We conduct experiments on the same LLMs as in the previous section. Additionally, for the Smaller Ref method that requires a smaller reference model along with the target LLM, we consider Llama-7B, Neo-125M, Pythia-70M, and OPT-350M as references.

Empirical Evaluation

In Table 4, we show the AUC for the unwatermarked and watermarked models using the UMD scheme, as well as the drop between the two. We observe that watermarking reduces the AUC (drop shown in bold in the table) by up to 14.2% across 4 detection methods and 5 LLMs. All the experiments on watermarked models are run with 5 different seeds and we report the mean and standard deviation of the results. We also report the AUC drop, which is computed by the difference between the AUC for the unwatermarked model and the mean AUC over the 5 runs for the watermarked model. Additionally, while the experiments from Table 4 are conducted on WikiMIA-256, we observe similar trends for WikiMIA-32, WikiMIA-64, and WikiMIA-128 in the appendix. We also study the impact of the watermark's strength on the AUC drop for Llama-30B in Figure 4 and for the other models in Figure 6 from the appendix. Note that we considered WikiMIA-256 for these experiments. We observe that higher watermark strengths generally induce larger AUC drops.

In addition to the 4 detection methods, we also consider Smaller Ref attack, which we include in Table 13 of the appendix. We consider different variations, including an unwatermarked reference model and a watermarked one with a similar strength but a different seed or with both strength and seed changed in comparison to the watermarked target model. The baseline is an unwatermarked model with an unwatermarked reference model. We observe the AUC drops in all scenarios (up to 16.4%), which is consistent with our previous findings.

	Llama-30B	Llama-13B
PPL	85.4% 84.7 ± 1.4% **0.7%**	68.2% 67.6 ± 2.5% **0.6%**
Lowercase	87.9% 80.9 ± 3.1% **7.0%**	77.6% 67.2 ± 4.0% **10.4%**
Zlib	82.5% 77.8 ± 1.2% **4.7%**	62.5% 57.1 ± 2.0% **5.4%**
Min-K% Prob	85.1% 85.0 ± 1.0% **0.1%**	70.2% 68.5 ± 0.1% **1.7%**

Table 3: AUC of each MIA for the unwatermarked (*top* of each cell), watermarked models (*middle* of each cell), and the drop between the two (*bottom* of each cell) on BookMIA using UMD scheme.

We also experiment with several percentages of green tokens for a fixed watermark strength of $\delta = 10$. We show the results in Table 14 of the appendix. We observe that for all models, in at least 80% of the cases all of the attacks' AUCs are negatively affected (positive drop value), suggesting that, in general, **finding a watermarking scheme that reduces the success rates of the current MIAs is not a difficult task**. Note that the experiments are run on WikiMIA for UMD scheme and the results are averaged over 5 watermark seeds.

Takeaways. Watermarking can significantly reduce the success of membership inference attacks (MIAs), with AUC drops up to 16.4%. By varying the percentage of green tokens as well as the watermark's strength, we observe that watermarking schemes can be easily tuned to negatively impact the detection success rates of MIAs. Finally, we conduct experiments on the BookMIA dataset and observe results consistent with our previous findings. These results are included in Table 3.

Improving Detection Performance with Adaptive Min-K% Prob

This section demonstrates how an informed, adaptive attacker can improve the success rate of a recent MIA, Min-K% Prob. Our main idea is that an attacker with knowledge of the watermarking technique (including green-red token lists and watermark's strength δ) can readjust token probabilities. This is possible even without additional information about the logit distribution, relying solely on the probability of each token from the target sample given the preceding tokens. Our approach relies on two key assumptions. First, knowledge of the watermarking scheme, which aligns with assumptions made in prior work on public watermark detection (Kirchenbauer et al. 2023). Second, access to the probability of each token in a sample, given the previous tokens—an assumption also made by the Min-K% Prob method (Shi et al. 2023).

Threat model. (1) The attacker's goal is to infer whether specific samples are part of the training set or not. In our setting, the attacker is not malicious, as the goal is to detect copyright violations. (2) Regarding the attacker's knowledge, we assume the attacker knows the watermarking method and its parameters (green and red lists, and the watermark strength), which aligns with the assumption made in prior work by Kirchenbauer et al. (2023) for public watermark detection. (3) As for the attacker's capabilities, we assume they can access the probabilities for each token in the given samples, similar to what a copyright auditor may have access to. This also mirrors the assumption made by Shi et al. (2023) in the context of training data detection.

Our method described in Algorithm 1 is based on the observation that if the denominator of softmax function (i.e., $\sum_i e^{z_i}$, where z_i is the logit for the i-th vocabulary) does not vary significantly when generating samples with the watermarked model (and similarly for the unwatermarked model), then we can readjust the probabilities of the green tokens by "removing" the bias δ. More precisely, assuming the approximation for the denominator of softmax is good, then the probability for each token t_i in an unwatermarked model will be around $\frac{e^{L_i}}{c}$, where L_i is the logit corresponding to the token t_i and c is a constant. However, for a watermarked model, if the token t_i is green, then the probability would be approximated by $\frac{e^{L_i+\delta}}{d}$, where d is again a constant, while in the case t_i is red the probability will be around $\frac{e^{L_i}}{d}$. To compensate for the bias introduced by watermarking, we divide the probability of green tokens by e^{δ} and this way we end up with probabilities that are just a scaled (by $\frac{c}{d}$) version of the probabilities from the unwatermarked model. The scaling factor will not affect the orders between the samples when computing the average of the minimum K% log-probabilities as long as the tested sentences are approximately the same length, which is an assumption made by Shi et al. (2023) as well.

Despite the strong assumption we assumed regarding the approximation of the denominator, empirical results show that our method effectively improves the success rate of Min-K% under watermarking. We show results for two LLMs in Table 5 and include the complete results for 5 LLMs in Table 17 from the appendix. We observe that our method improves over the baseline in 95% of the cases, and the increase is as high as 4.8% (averaged over 5 runs).

Finally, we also consider adaptive versions of the Lowercase and Zlib methods. Our findings show that these adaptive methods outperform the baselines in at least 80% of cases. Detailed results are provided in the appendix.

Takeaways. We demonstrate that an adaptive attacker can leverage the knowledge of a watermarking scheme to increase the success rates of recent MIAs.

	Llama-30B	NeoX-20B	Llama-13B	Pythia-2.8B	OPT-2.7B
PPL	72.0% 70.6 ± 1.9% **1.4%**	71.3% 64.7 ± 2.3% **6.6%**	71.2% 70.0 ± 2.6% **1.2%**	67.8% 64.4 ± 1.9% **3.4%**	60.5% 54.9 ± 2.2% **5.6%**
Lowercase	68.1% 63.8 ± 4.5% **4.3%**	68.2% 55.4 ± 5.5% **14.2%**	65.5% 61.6 ± 3.8% **3.9%**	62.9% 58.7 ± 3.2% **4.2%**	58.9% 49.7 ± 2.9% **9.2%**
Zlib	72.7% 72.0 ± 1.6% **0.7%**	73.2% 66.6 ± 2.0% **6.6%**	73.1% 71.6 ± 2.3% **1.5%**	69.2% 66.1 ± 1.2% **3.1%**	62.7% 58.1 ± 1.8% **4.6%**
Min-K% Prob	71.8% 70.5 ± 1.8% **1.3%**	78.0% 76.2 ± 2.1% **1.8%**	72.9% 70.4 ± 3.2% **2.5%**	71.0% 69.5 ± 1.6% **1.5%**	65.5% 63.1 ± 3.4% **2.4%**

Table 4: AUC of each MIA for the unwatermarked (*top* of each cell), watermarked models (*middle* of each cell), and the drop between the two (*bottom* of each cell) on WikiMIA-256 using UMD scheme.

Algorithm 1: Adaptive Min-K% Prob

Require : Tokenized target sample $t = t_1 \oplus t_2 \oplus ... \oplus t_n$, access to the probability of the target (watermarked) LLM f to generate t_i given the $i-1$ previous tokens and t_0 (empty string) $f(t_i|t_0 \oplus t_1 \oplus ... \oplus t_{i-1})$ (similar assumption as Min-K% Prob algorithm), K, we assume we know the watermarking scheme (e.g., for public watermark detection purposes), i.e. we know the green and red lists as well as δ.

Output : Adjusted average of the minimum $K\%$ token probabilities when generating $t_1 \oplus t_2 \oplus ... \oplus t_n$

adj_prob $\leftarrow \{\}$ ▷ The set of adjusted probabilities

for $i \in 1, 2, \ldots, n$ **do**

  $p_f(t_i) \leftarrow f(t_i|t_0 \oplus t_1 \oplus ... \oplus t_{i-1})$ ▷ Probability of t_i when the model is watermarked

  if t_i *is green* **then**

    adj_prob $\leftarrow$ adj_prob $\cup \{\frac{p_f(t_i)}{e^\delta}\}$ ▷ Adjust the probability if the token is green

  else

    adj_prob $\leftarrow$ adj_prob $\cup \{p_f(t_i)\}$

  end

end

$k = floor(n \cdot K\%)$ ▷ Find the number of token probabilities to keep

adj_k_prob $\leftarrow min\_k$(adj_prob) ▷ Select the minimum k probabilities

return mean(log(adj_k_prob)) ▷ Return the mean of the minimum k log-probabilities

		Llama-30B	Llama-13B
WikiMIA 32	Not adapt.	66.2%	64.5%
	Adapt.	**68.5%**	**66.3%**
WikiMIA 64	Not adapt.	64.4%	62.8%
	Adapt.	**67.3%**	**64.9%**
WikiMIA 128	Not adapt.	70.0%	68.9%
	Adapt.	**73.1%**	**71.0%**
WikiMIA 256	Not adapt.	70.5%	70.4%
	Adapt.	**71.3%**	**72.4%**

Table 5: We show the AUC of Min-%K Prob (referred as "Not adapt.") and our method (referred as "Adapt.") when using UMD watermarking scheme. We highlight the cases when our method improves over the baseline.

Conclusion and Discussion

Watermarking LLMs has unintended consequences on methods towards copyright protection. Our experiments demonstrate that while watermarking may be a promising solution to prevent copyrighted text generation, watermarking also complicates membership inference attacks that may be employed to detect copyright abuses. Watermarking can be a double-edged sword for copyright regulators since it promotes compliance during generation time, while making training time copyright violations harder to detect. We hope our work furthers the discussion around watermarking and copyright issues for LLMs.

Acknowledgements

Panaitescu-Liess, Che, An, Xu, Pathmanathan, Chakraborty, Zhu, and Huang are supported by DARPA Transfer from Imprecise and Abstract Models to Autonomous Technologies

(TIAMAT) 80321, National Science Foundation NSF-IIS-2147276 FAI, DOD-AFOSR-Air Force Office of Scientific Research under award number FA9550-23-1-0048, Adobe, Capital One and JP Morgan faculty fellowships.

References

Aaronson, S. 2023. Simons institute talk on watermarking of large language models.

Biderman, S.; Schoelkopf, H.; Anthony, Q. G.; Bradley, H.; O'Brien, K.; Hallahan, E.; Khan, M. A.; Purohit, S.; Prashanth, U. S.; Raff, E.; et al. 2023. Pythia: A suite for analyzing large language models across training and scaling. In *International Conference on Machine Learning*, 2397–2430. PMLR.

Black, S.; Biderman, S.; Hallahan, E.; Anthony, Q.; Gao, L.; Golding, L.; He, H.; Leahy, C.; McDonell, K.; Phang, J.; et al. 2022. Gpt-neox-20b: An open-source autoregressive language model. *arXiv preprint arXiv:2204.06745*.

Carlini, N.; Ippolito, D.; Jagielski, M.; Lee, K.; Tramer, F.; and Zhang, C. 2022. Quantifying memorization across neural language models. *arXiv preprint arXiv:2202.07646*.

Carlini, N.; Tramer, F.; Wallace, E.; Jagielski, M.; Herbert-Voss, A.; Lee, K.; Roberts, A.; Brown, T.; Song, D.; Erlingsson, U.; et al. 2021. Extracting training data from large language models. In *30th USENIX Security Symposium (USENIX Security 21)*, 2633–2650.

Christ, M.; Gunn, S.; and Zamir, O. 2023. Undetectable watermarks for language models. *arXiv preprint arXiv:2306.09194*.

Chu, T.; Song, Z.; and Yang, C. 2024. How to Protect Copyright Data in Optimization of Large Language Models? In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 17871–17879.

Duarte, A. V.; Zhao, X.; Oliveira, A. L.; and Li, L. 2024. DE-COP: Detecting Copyrighted Content in Language Models Training Data. *arXiv preprint arXiv:2402.09910*.

Elkin-Koren, N.; Hacohen, U.; Livni, R.; and Moran, S. 2023. Can Copyright be Reduced to Privacy? *arXiv preprint arXiv:2305.14822*.

Hacohen, U.; Haviv, A.; Sarfaty, S.; Friedman, B.; Elkin-Koren, N.; Livni, R.; and Bermano, A. H. 2024. Not All Similarities Are Created Equal: Leveraging Data-Driven Biases to Inform GenAI Copyright Disputes. *arXiv preprint arXiv:2403.17691*.

Ippolito, D.; Tramèr, F.; Nasr, M.; Zhang, C.; Jagielski, M.; Lee, K.; Choquette-Choo, C. A.; and Carlini, N. 2023. Preventing generation of verbatim memorization in language models gives a false sense of privacy. In *Proceedings of the 16th International Natural Language Generation Conference*, 28–53. Association for Computational Linguistics.

Kirchenbauer, J.; Geiping, J.; Wen, Y.; Katz, J.; Miers, I.; and Goldstein, T. 2023. A watermark for large language models. In *International Conference on Machine Learning*, 17061–17084. PMLR.

Krishna, K.; Song, Y.; Karpinska, M.; Wieting, J.; and Iyyer, M. 2024. Paraphrasing evades detectors of ai-generated text, but retrieval is an effective defense. *Advances in Neural Information Processing Systems*, 36.

Kuditipudi, R.; Thickstun, J.; Hashimoto, T.; and Liang, P. 2023. Robust distortion-free watermarks for language models. *arXiv preprint arXiv:2307.15593*.

Liu, A.; Pan, L.; Hu, X.; Meng, S.; and Wen, L. 2024. A Semantic Invariant Robust Watermark for Large Language Models. In *The Twelfth International Conference on Learning Representations*.

Ren, J.; Xu, H.; He, P.; Cui, Y.; Zeng, S.; Zhang, J.; Wen, H.; Ding, J.; Liu, H.; Chang, Y.; et al. 2024. Copyright Protection in Generative AI: A Technical Perspective. *arXiv preprint arXiv:2402.02333*.

Shi, W.; Ajith, A.; Xia, M.; Huang, Y.; Liu, D.; Blevins, T.; Chen, D.; and Zettlemoyer, L. 2023. Detecting pretraining data from large language models. *arXiv preprint arXiv:2310.16789*.

Touvron, H.; Lavril, T.; Izacard, G.; Martinet, X.; Lachaux, M.-A.; Lacroix, T.; Rozière, B.; Goyal, N.; Hambro, E.; Azhar, F.; Rodriguez, A.; Joulin, A.; Grave, E.; and Lample, G. 2023. LLaMA: Open and Efficient Foundation Language Models. *arXiv preprint arXiv:2302.13971*.

Vyas, N.; Kakade, S. M.; and Barak, B. 2023. On provable copyright protection for generative models. In *International Conference on Machine Learning*, 35277–35299. PMLR.

Zhang, S.; Roller, S.; Goyal, N.; Artetxe, M.; Chen, M.; Chen, S.; Dewan, C.; Diab, M.; Li, X.; Lin, X. V.; et al. 2022. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*.

Zhao, X.; Ananth, P.; Li, L.; and Wang, Y.-X. 2023. Provable robust watermarking for ai-generated text. *arXiv preprint arXiv:2306.17439*.