A Likelihood Based Approach for Watermark Detection

Xingchi Li[*] Texas A&M University

Guanxun Li[*] Beijing Normal University at Zhuhai

Xianyang Zhang[†] Texas A&M University

Abstract

Watermarking techniques embed statistical signals within content generated by large language models to help trace its source. Although existing methods perform well on long texts, their effectiveness significantly decreases for shorter texts. We introduce a statistical detection approach that improves the power of watermark detection, particularly in shorter texts. Our method leverages both the watermark key sequence and the next token probabilities (NTPs) to determine whether a text is generated by a large language model. We demonstrate the optimality of our approach and analyze its power properties. We also investigate an approach to estimating NTPs and extend our method to scenarios where texts face potential attacks such as substitutions, insertions, or deletions. We validate the effectiveness of our technique using texts generated by `Meta-Llama-3-8B` from Meta and `Mistral-7B-v0.1` from Mistral AI, utilizing prompts extracted from Google's C4 dataset. In scenarios without attacks and with short text lengths, our method demonstrates approximately 65% power improvement compared to the baseline method on average. We release all code publicly at `https://github.com/doccstat/llm-watermark-adaptive`.

1 INTRODUCTION

The proliferation of advanced large language models (LLMs) has heightened the need to differentiate between machine-generated and human-authored text. State-of-the-art LLMs like GPT-4 and Claude-3.5 Sonnet produce remarkably human-like content, posing significant challenges for detection even by human readers. Distinguishing between artificial and human-written text is crucial for several reasons: it helps combat the spread of misinformation, maintains academic integrity by preventing misuse of AI tools in educational settings, safeguards against model extraction attacks through distillation techniques, and preserves the quality of training data for future language models by preventing training dataset contamination.

Watermarking provides a way to trace the source of machine-generated content and to ensure accountability without needing direct access to model parameters (Liu et al., 2024; Kirchenbauer et al., 2023a). In the context of LLMs, Watermarking involves embedding signals into texts produced by an LLM, allowing for verifiable detection of the LLM-generated content using the corresponding watermark key sequence. An ideal watermark algorithm should meet three criteria (Kirchenbauer et al., 2023a; Kuditipudi et al., 2024): (i) preserve the original next token prediction distribution; (ii) being capable of detecting watermarked texts with high efficiency; and (iii) withstand perturbations of watermarked texts.

Several watermarking techniques for LLMs have been proposed in the recent literature. Examples include the "red-green list" technique (Kirchenbauer et al., 2023a), the inverse transform sampling (ITS) method (Kuditipudi et al., 2024), and exponential minimum sampling (EMS) (Aaronson, 2023). While these approaches generally perform well for long texts, their effectiveness diminishes notably with shorter texts (Kuditipudi et al., 2024).

In this study, we leverage the watermark key sequence and the next token probabilities (NTPs) to develop an efficient watermark detection method, where NTPs represent the probabilities assigned to generated tokens given their preceding token sequences. Specifically, we focus on the EMS due to its unbiasedness and efficiency compared to other existing approaches and develop a randomization test using the likelihood ratio (LR) test statistic constructed based on the wa-

Proceedings of the 28[th] International Conference on Artificial Intelligence and Statistics (AISTATS) 2025, Mai Khao, Thailand. PMLR: Volume 258.

[*]Equal contribution

[†]Corresponding author: zhangxiany@stat.tamu.edu

termark key sequence and NTPs. In contrast to the original test statistic proposed by Aaronson (2023), we assign unique weights to each token, allowing the proposed method to allocate greater weights to more informative tokens and thereby enhance overall performance. Theoretically, we analyze the power properties of the proposed method and identify scenarios in which it achieves near-perfect power, whereas the previous detection scheme does not.

In real-world applications, the true NTPs are unknown because the true prompts used to generate the watermarked texts are not accessible to the detector. To address this, we propose a prompt estimation method based on optimization over a set of potential prompts related to the text (called an instruction set). We numerically verify that our optimization approach selects the prompt within the instruction set that is closest to the true prompt (in terms of the Levenshtein distance) with a probability exceeding 95%.

Finally, we validate the effectiveness of our techniques through a series of numerical examples using texts generated by two language models, namely `Meta-Llama-3-8B` from Meta and `Mistral-7B-v0.1` from Mistral AI, with prompts extracted from Google's C4 dataset. In the ideal scenario where no modifications are made to the watermarked text, our method achieves up to 65% higher detection power than the baseline method on average when the text length is short. In more realistic scenarios, where the text has been partially deleted, inserted, or substituted, and without knowledge of the true prompt, the proposed method still outperforms the baseline method by a noticeable margin.

1.1 Related Works and Contributions

A seminal watermarking scheme was introduced by Kirchenbauer et al. (2023a). Their approach, known as the "red-green list" technique, divides the vocabulary into two distinct categories: green and red. This method enhances the likelihood of selecting tokens from the "green" list during the next token prediction. Additionally, the authors developed a statistical test to identify the presence of such watermarks in the generated text. Cai et al. (2024) optimized watermarking in LLMs by addressing the trade-off between model distortion and detection capability. They employed a dual gradient ascent algorithm to enhance the "red-green list" watermarking scheme. Building on the "red-green list" framework, Nemecek et al. (2024) develop a topic-based watermarking algorithm that leverages topics extracted from input prompts or non-watermarked outputs to embed watermarks sensitive to the specific topic of the text. Other studies discussing this type of watermarking scheme include (Kirchenbauer et al., 2023b), (Liu and Bu, 2024), and (Zhao et al., 2023). However, this watermarking technique introduces bias by modifying the next token prediction distributions, which could degrade the performance of the LLM.

Along a different line, several unbiased watermarking techniques have been proposed in the recent literature to overcome the drawbacks of the "red-green list" scheme. Aaronson (2023) proposed the EMS technique. When generating the next tokens, EMS involves multiplying the probabilities of all tokens by an exponential factor, embedding a detectable watermark signal into the generated text without altering the original text distribution. This approach is closely related to the Gumbel trick (Papandreou and Yuille, 2011) in machine learning. Kuditipudi et al. (2024) introduced an ITS watermarking method that is robust against perturbations while maintaining the original text distribution, ensuring no distortion in the generated text. Other unbiased watermarking methods include the works of (Zhao et al., 2024), (Hu et al., 2023), and (Wu et al., 2023).

However, limited attention has been given to understanding the statistical properties of watermark generation and detection schemes. The paper by Huang et al. (2023) is the first to develop a statistical framework for watermark detection by formulating it as a hypothesis testing problem. The authors employ pseudo-random generators and hypothesis testing to link output tokens with rejection regions, thereby characterizing the Uniformly Most Powerful (UMP) tests. However, Huang et al. (2023) focused on the i.i.d. token setting, assuming that the next token prediction distributions remain unchanged, which is unrealistic. In Li et al. (2024c), the authors introduced a flexible statistical framework for embedding watermarks to enhance the efficiency of watermark detection and provided optimal detection rules. They proposed using a pivotal statistic and a secret key to control false positive rates and evaluated the power of detection rules by calculating the asymptotic false negative rate. However, this framework operates under the idealized assumption that tokens are either entirely human-written or fully generated by an LLM. Li et al. (2024b) extended this framework by considering the detection of watermarked text that may have been edited by humans.

To sum up, compared to the existing literature, we make the following contributions:

- We propose the first method to leverage NTPs in watermarked text detection and demonstrate its optimality; see Theorem 2. Additionally, we rigorously analyze its power properties in scenarios

with and without attacks.

- We study the NTPs estimation problem and propose a greedy method to find the best prompt within an instruction set.

The rest of the paper is organized as follows. In Section 2, we introduce the watermark detection problem and propose a randomization test for its detection. We further present the theory to control Type I and Type II errors. In Section 3, we present our likelihood-based test statistic and prove its optimality. We introduce a method to estimate the NTPs in Section 4 and present the numerical results in Section 5. Section 6 discusses a few open problems.

2 WATERMARK DETECTION

2.1 Problem Setups

Let $\mathcal{V}$ denote the vocabulary (which is a discrete set) and let $V = |\mathcal{V}|$ represent its size. Each token in the vocabulary is indexed uniquely by an element from $[V] \coloneqq \{1, 2, \ldots, V\}$. Define P as an autoregressive LLM that maps a string $y_{-n_0:i-1}$ to a distribution over the vocabulary, where $y_{-n_0:0}$ represents the user-provided prompt. The distribution of the next token y_i is denoted as $\mu_i(\cdot) \coloneqq p(\cdot|y_{-n_0:i-1})$. Suppose a string $y_{1:n}$ is generated from an LLM. We denote by $p_i \coloneqq \mu_i(y_i \mid y_{-n_0:i-1})$ the probability of generating y_i given the previous tokens $y_{-n_0:i-1}$ (a.k.a NTP).

Let $\xi_{1:t} = \xi_1\xi_2\cdots\xi_t$ represent a watermark key sequence, where $\xi_i \in \Xi$ with Ξ being the space to which the key belongs. Given a prompt from a third-party user, the LLM provider generates text autoregressively using a decoder function Γ, which maps ξ_t and the distribution μ_t to a value in $\mathcal{V}$. A watermarking scheme is unbiased if it preserves the original text distribution, i.e.,

$$P(\Gamma(\xi_t, \mu_t) = y) = \mu_t(y).$$

A watermarked text generation algorithm recursively produces a string $y_{1:n}$ as follows:

$$y_i = \Gamma(\xi_i, p(\cdot|y_{-n_0:i-1})), \quad 1 \le i \le n,$$

where n is the total number of tokens in the generated text $y_{1:n}$. The values ξ_is are assumed to be independently generated from some distribution ν over Ξ. In other words, given $p(\cdot|y_{-n_0:i-1})$, distribution of y_i is fully determined by ξ_i and $y_{-n_0:i-1}$.

In this research, we concentrate on the EMS watermarking technique proposed by Aaronson (2023), which is inspired by the Gumbel softmax rule. OpenAI has already implemented a prototype of this scheme. One of the reasons for our focus on EMS is that its associated detection method is generally more powerful than detection procedures based on ITS, as shown in Figure 1. To generate each token of a text using EMS, we independently sample $\xi_{ik} \sim \text{Unif}[0,1]$ for $1 \le k \le V$, where Unif$[0,1]$ denotes the uniform distribution on $[0,1]$. Then, we define

$$\begin{aligned} y_i &= \arg\max_{1\le k\le V} \frac{\log(\xi_{ik})}{p(k|y_{-n_0:i-1})} \\ &= \arg\min_{1\le k\le V} \frac{-\log(\xi_{ik})}{p(k|y_{-n_0:i-1})} = \arg\min_{1\le k\le V} E_{ik}, \end{aligned} \tag{1}$$

where

$$E_{ik} \coloneqq -\log(\xi_{ik})/p(k|y_{-n_0:i-1}) \sim \text{Exp}(p(k|y_{-n_0:i-1})),$$

with Exp(a) denoting the exponential distribution with the rate parameter a. For two exponential random variables $X \sim \text{Exp}(a)$ and $Y \sim \text{Exp}(b)$, the following properties hold: (i) $\min(X, Y) \sim \text{Exp}(a+b)$, and (ii) $P(X < Y) = \mathbb{E}[1 - \exp(-aY)] = a/(a+b)$. Using these properties, we can verify that

$$P(y_i = k) = P\left(E_{ik} < \min_{j\neq k} E_{ij}\right) = p(k|y_{-n_0:i-1}).$$

Thus, EMS preserves the original text distribution. In addition, EMS has low computational overhead on top of the LLM generation and is robust to local perturbations (Aaronson, 2023; Kuditipudi et al., 2024).

2.2 Watermarked Text Detection via Randomization Tests

In this section, we address the detection problem, which involves determining whether a given text is watermarked. Suppose a string $\widetilde{y}_{1:n}$ is published by a third-party user, and a key sequence $\xi_{1:n}$ is provided to a detector. The detector performs a hypothesis test of the form:

$$H_0 : \widetilde{y}_{1:n} \text{ is not watermarked},$$

against

$$H_a : \widetilde{y}_{1:n} \text{ is watermarked},$$

using a p-value computed from a test statistic $\phi(\xi_{1:n}, \widetilde{y}_{1:n})$. The test statistic ϕ quantifies the dependence between the text $\widetilde{y}_{1:n}$ and the key sequence $\xi_{1:n}$. Throughout our discussion, we assume that larger values of ϕ provide stronger evidence against the null hypothesis, indicating greater dependence between $\widetilde{y}_{1:n}$ and $\xi_{1:n}$. For instance, Aaronson (2023) proposed a metric to measure the dependence between a string $\widetilde{y}_{1:n}$ and a key sequence $\xi_{1:n}$, defined as

$$\phi(\xi_{1:n}, \widetilde{y}_{1:n}) = \frac{1}{n}\sum_{i=1}^{n} \left\{\log(\xi_{i,\widetilde{y}_i}) + 1\right\}. \tag{2}$$

The rationale behind this metric is that if $\widetilde{y}_i$ is generated using the key ξ_i, then $\xi_{i,\widetilde{y}_i}$ is likely to have a higher value than the other components of ξ_i. Consequently, a larger value of $\phi(\xi_{1:n}, \widetilde{y}_{1:n})$ suggests that the string $\widetilde{y}_{1:n}$ is more likely to be watermarked.

To calculate the p-value, we employ a randomization test (Edgington and Onghena, 2007; Good, 2013). Specifically, we generate $\xi_i^{(t)} \sim \nu$ independently for each $1 \le i \le n$ and $1 \le t \le T$, ensuring that $\xi_i^{(t)}$s are independent of $\widetilde{y}_{1:n}$. The randomization-based p-value is then defined as

$$\mathsf{p}_T = \frac{1}{T+1}\left(1 + \sum_{t=1}^{T} \mathbf{1}\left\{\phi(\xi_{1:n}, \widetilde{y}_{1:n}) \le \phi(\xi_{1:n}^{(t)}, \widetilde{y}_{1:n})\right\}\right).$$

Given a pre-specified level $\alpha \in (0,1)$, we reject the null hypothesis H_0 whenever $\mathsf{p}_T \le \alpha$. To study the type I and type II errors of this testing procedure, we let $\mathcal{F}_n = [y_{-n_0:0}, \widetilde{y}_{1:n}]$. Given the test statistic $\phi(\xi_{1:n}, \widetilde{y}_{1:n})$, define $\mathrm{Sd}_\xi := \mathrm{Sd}(\phi(\xi_{1:n}, \widetilde{y}_{1:n}) \mid \mathcal{F}_n)$ and $\mathsf{E}_\xi := \mathbb{E}[\phi(\xi_{1:n}, \widetilde{y}_{1:n}) \mid \mathcal{F}_n]$. Similarly, let $\mathrm{Sd}_{\xi'}$ and $\mathsf{E}_{\xi'}$ be defined with $\xi_{1:n}$ replaced by $\xi'_{1:n}$, where $\xi'_{1:n}$ is a key sequence generated independently of $\widetilde{y}_{1:n}$ but in the same manner as $\xi_{1:n}$.

The following theorem states that the proposed randomization test effectively controls both the Type I and Type II errors under appropriate conditions.

Theorem 1. For the randomization-based test, the following results hold:

(i) Under the null hypothesis,

$$P(\mathsf{p}_T \le \alpha) = \lfloor (T+1)\alpha \rfloor / (T+1) \le \alpha,$$

where $\lfloor a \rfloor$ denotes the greatest integer less than or equal to a;

(ii) For any $\epsilon > 0$, if $T > 2/\epsilon - 1$, $\mathsf{E}_{\xi'} = 0$, $\mathrm{Sd}_{\xi'} = o(\mathsf{E}_\xi)$, and $\mathrm{Sd}_\xi = o(\mathsf{E}_\xi)$, then

$$P(\mathsf{p}_T \le \alpha \mid \mathcal{F}_n) \ge 1 - C_1 \exp(-2T\epsilon^2) + o(1), \quad (3)$$

as $n \to \infty$, where $C_1 > 0$.

Remark 1. Theorem 1 part (i) was first proved by Li et al. (2024a). In view of the proof of Theorem 1 part (ii), the condition $\mathrm{Sd}_{\xi'} = o(\mathsf{E}_\xi)$ can be relaxed as follows: there exists a sufficiently large constant C such that $\mathsf{E}_\xi > C\mathrm{Sd}_{\xi'}$ and $\mathrm{Sd}_\xi = o(\mathsf{E}_\xi - C\mathrm{Sd}_{\xi'})$, where C depends on the permutation-based critical value. In other words, if E_ξ is always greater than $C\mathrm{Sd}_{\xi'}$, and the difference $\mathsf{E}_\xi - C\mathrm{Sd}_{\xi'}$ is sufficiently large, then Theorem 1 (ii) remains valid even when E_ξ and $\mathrm{Sd}_{\xi'}$ are of the same order.

3 METHODOLOGY

In this section, we introduce a likelihood based approach for watermark detection.

3.1 Likelihood Ratio Test

Given a string $\widetilde{y}_{1:n}$ and a watermark key sequence $\xi_{1:n}$, the watermark detection problem can be reformulated as a hypothesis test of the following form:

$$\mathcal{H}_0 \colon \widetilde{y}_i \text{ is independent of } \xi_i \text{ for all } i = 1, \ldots, n,$$

against

$$\mathcal{H}_a \colon \widetilde{y}_i \text{ is generated from } \xi_i \text{ for all } i = 1, \ldots, n.$$

To implement the likelihood ratio (LR) test, we require the following lemma, which demonstrates that under the EMS framework, the watermark key $-\log(\xi_{i,\widetilde{y}_i})$ follows an exponential distribution.

Lemma 1. We have

$$-\log(\xi_{i,\widetilde{y}_i}) \mid y_{-n_0:i-1}, \widetilde{y}_{1:i-1} \sim \begin{cases} \mathrm{Exp}(1) & \text{if } \widetilde{y}_i \text{ is not generated from } \xi_i, \\ \mathrm{Exp}\,(1/p_i) & \text{if } \widetilde{y}_i \text{ is generated from } \xi_i. \end{cases}$$

By Lemma 1, the log-likelihood ratio test statistic for testing whether $\widetilde{y}_i$ is generated from ξ_i is defined as

$$L(\xi_{1:n}, \widetilde{y}_{1:n}) = \frac{1}{n}\sum_{i=1}^{n} \frac{1-p_i}{p_i}\left(\log(\xi_{i,\widetilde{y}_i}) + 1\right), \quad (4)$$

where we have centered the statistic to ensure that its mean is zero under $\mathcal{H}_0$. Consider the test function ψ defined as

$$\psi(\xi_{1:n}, \widetilde{y}_{1:n}) = \mathbf{1}\left\{L(\xi_{1:n}, \widetilde{y}_{1:n}) > q_{1-\alpha}\right\},$$

where $q_{1-\alpha}$ is the $1-\alpha$ quantile of the random variable $n^{-1}\sum_{i=1}^{n}(1-p_i)(1-e_i)/p_i$, with e_i being a sequence of independent Exp(1) random variables. Clearly, the size of this test is exactly α. The Neyman-Pearson Lemma ensures that it is the most powerful test among all tests controlling the size at level α. In other words, any other test with size at most α has lower power than the LR test.

Theorem 2. The test $\psi(\xi_{1:n}, \widetilde{y}_{1:n})$ is most powerful among all tests controlling the size at level α.

3.2 A General Class of Statistics

Motivated by the form of the log-likelihood ratio test statistic (4), we consider a general class of test statistics:

$$\phi(\xi_{1:n}, \widetilde{y}_{1:n}; w_{1:n}) = \frac{1}{n}\sum_{i=1}^{n} w_i\left\{\log(\xi_{i,\widetilde{y}_i}) + 1\right\}, \quad (5)$$

for some non-negative weights $w_{1:n} = (w_1, \ldots, w_n)$. Note that this class of statistics includes the original statistic (2) considered in Aaronson (2023) and the LR test statistic (4) as special cases. Specifically, when $w_i = 1$ for all i, test statistic (5) reduces to the original statistic (2); when $w_i = (1 - p_i)/p_i$, test statistic (5) becomes the LR test statistic (4).

The weights $w_{1:n}$ allow us to treat each token differently. By assigning distinct weights to each token, we can prioritize certain tokens over others based on their importance within the LLM. For instance, tokens with smaller p_i contain more information about whether the text is watermarked than those with higher p_is. This flexibility enables the proposed method to assign greater weights to more informative tokens, enhancing the overall performance.

By Theorem 1, the randomization test based on the test statistic in (5) has power approaching one under the following condition.

Corollary 1. If

$$\frac{\sum_{i=1}^n w_i(1-p_i)}{\sqrt{\sum_{i=1}^n w_i^2}} \to +\infty, \tag{6}$$

then the conditions in Theorem 1 are fulfilled. Consequently, the power of the randomization-based test using the statistic in (5) converges to one as $T \to +\infty$.

If $w_i = 1$, then Condition (6) reduces to

$$\frac{1}{\sqrt{n}} \sum_{i=1}^n (1-p_i) \to +\infty, \tag{7}$$

which is precisely the condition under which the power of the randomization-based test, using the original test statistic (2), converges to 1, as established in Corollary 1 in Li et al. (2024a). When $w_i = (1 - p_i)/p_i$, Condition (6) becomes

$$\frac{\sum_{i=1}^n (1-p_i)^2/p_i}{\sqrt{\sum_{i=1}^n (1-p_i)^2/p_i^2}} \to +\infty. \tag{8}$$

We observe that tokens with smaller NTPs exert a more pronounced influence on the LR test compared to the original test. Specifically, the existence of a subset of tokens with small NTPs can substantially boost the detection power of the LR test. The following example illustrates a scenario where condition (8) is satisfied, while condition (7) is not. Consequently, the power of the LR test approaches one, whereas the power of the original method fails to reach one.

Example 1. Consider the scenario where $p_j = 1/\log(n)$ for $j \in \mathcal{I}_1$ and $p_j = 1$ for $j \notin \mathcal{I}_1$, with $|\mathcal{I}_1| = \log(n)$. In this case, Condition (7) does not hold; however, Condition (8) is satisfied. As a result, the likelihood ratio (LR) test can detect the watermark with high power, whereas the method proposed by Aaronson (2023) may fail. Although this setup may appear artificial, it effectively illustrates the power of the LR test, even when the signal is sparse, with only a small proportion of tokens containing information about the watermark.

3.3 Incorporating the Estimated Ntps Into Test Statistics

In practice, the true NTPs $\{p_i\}$ are unknown even with access to the LLM, as the true prompt is never known. Nevertheless, the NTPs can be estimated by calling the LLM and supplementing with a prompt. We shall provide more details about the estimation of NTPs in Section 4. For now, let us define q_i as a generic estimate of p_i. Replacing p_i with q_i in (4), we obtain the LR test statistic with the estimated NTPs:

$$\frac{1}{n} \sum_{i=1}^n \frac{1-q_i}{q_i} \left\{ \log(\xi_{i,y_i}) + 1 \right\},$$

which belongs to the general class of test statistics in (5) with $w_i = (1-q_i)/q_i$. When (8) holds, the following theorem states that if q_i is sufficiently close to p_i, the power of the proposed method with q_i will converge to one.

Theorem 3. Assuming that (8) holds and

$$\max_i \frac{|p_i - q_i|}{\min(p_i, 1-p_i)} = o(1),$$

then

$$\frac{\sum_{i=1}^n w_i(1-p_i)}{\sqrt{\sum_{i=1}^n w_i^2 p_i^2}} \to +\infty,$$

where $w_i = (1-q_i)/q_i$. Hence, the statistic in (5) with $w_i = (1 - q_i)/q_i$ has power approaching one.

3.3.1 Regularization

In practice, the estimated NTPs q_i can be extremely small, which makes the proposed method unstable. To address this issue, we found that performing regularization on the estimated NTPs often enhances the robustness of the proposed algorithm. Here, we focus on the shrinkage strategy. Specifically, given an estimated NTP q_i, we define the regularized NTP as

$$\mathrm{S}(q_i, \lambda) = \lambda q_i + (1-\lambda) p_{i,0},$$

where $\lambda \in (0, 1)$ and $(p_{1,0}, \ldots, p_{n,0})$ is a pre-specified vector of probabilities, e.g., $p_{i,0} = 0.5$ for all i when there is no prior information. Then, we can modify

the LR test statistic using the regularized NTPs:

$$\phi_{\text{shrinkage}}(\xi_{1:n}, \widetilde{y}_{1:n}; q_{1:n}, \lambda) = \frac{1}{n}\sum_{i=1}^{n} \frac{1-\mathrm{S}(q_i,\lambda)}{\mathrm{S}(q_i,\lambda)} \left\{\log(\xi_{i,\widetilde{y}_i}) + 1\right\}. \quad (9)$$

3.4 Watermarked Text Detection With Attack

In practice, the text published by the user, denoted as $\widetilde{y}_{1:m}$, can differ significantly from the text initially generated by the LLM using the key $\xi_{1:n}$. To account for this difference, we employ a transformation function $\mathcal{E}$ that takes $y_{1:n}$ as input and produces the published text $\widetilde{y}_{1:m}$ as output:

$$\widetilde{y}_{1:m} = \mathcal{E}(y_{1:n}). \quad (10)$$

This transformation can involve substitutions, insertions, deletions, or other edits to the input text. We emphasize that Theorem 1 remains valid when replacing $\widetilde{y}_{1:n}$ with $\widetilde{y}_{1:m}$.

Since the published text $\widetilde{y}_{1:m}$ can significantly differ from the initially generated text $y_{1:n}$, we do not expect every token in $\widetilde{y}_{1:m}$ to be related to the key sequence $\xi_{1:n}$. Instead, we anticipate that certain substrings of $\widetilde{y}_{1:m}$ are correlated with the key sequence under the alternative hypothesis H_a.

To measure this dependence, we employ a scanning method that examines every substring of $\widetilde{y}_{1:m}$ and a corresponding substring of $\xi_{1:n}$ with the same length B. Let ϕ be defined in (5). Given the block size B, we define the maximum test statistic as

$$\Phi(\xi_{1:n}, \widetilde{y}_{1:m}) = \max_{1\le a\le n-B+1} \max_{1\le b\le m-B+1} \phi(\xi_{a:a+B-1}, \widetilde{y}_{b:b+B-1}). \quad (11)$$

Theorem 4. Consider the maximum statistic defined in (11), with ϕ defined in (5). Suppose that

$$C_{N,B}^{-1} \max_{a,b} \mathbb{E}\left[\phi\left(\xi_{a:a+B-1}, \widetilde{y}_{b:b+B-1}\right) | \, \widetilde{y}_{1:m}, y_{-n_0:n}\right] \to +\infty, \quad (12)$$

where $N = \max\{n, m\}$ and $C_{N,B} = \sqrt{\Omega_{\max}}\log(N)/B$ with $\Omega_{\max} = \max_b \sum_{i=b}^{b+B-1} w_i^2$. Then (3) holds.

Remark 2. Instead of conducting the maximum test as described previously, we can employ an alternative method by combining the evidence against the null hypothesis from different substrings through a p-value combination approach. In this approach, we first calculate a p-value (denoted as p_b) to test for the presence of a watermark within each segment $\widetilde{y}_{b:b+B-1}$ using the randomization test. We then combine these p-values using either Bonferroni's method or the recent Cauchy combination method (Liu and Xie, 2020) to obtain a global p-value for testing the presence of a watermark within the entire text. It is important to note that both Bonferroni's method and the Cauchy combination method are designed to handle dependencies among the p-values $\{p_b\}_{b=1}^{m-B+1}$.

4 ESTIMATING THE NTPS

In this section, we propose a greedy method to estimate the prompt. Given an estimated prompt, the NTP for each token can be calculated conditioned on the estimated prompt as well as the previously generated tokens by calling the LLM.

4.1 Prompt Estimation

There is limited research on estimating the user prompt from texts generated by an LLM. Here, we consider a greedy approach that searches for the most likely prompt within a predefined set of strings to maximize the likelihood of the texts produced by the LLM.

Let $\mathcal{P}_{\text{opt}}$ be an instruction set (Taori et al., 2023; Zheng et al., 2024) containing all possible candidate prompts related to the potentially watermarked texts, and define

$$\hat{z} = \arg\max_{z\in\mathcal{P}_{\text{opt}}} \sum_{i=1}^{m} \log p(\widetilde{y}_i \mid z\widetilde{y}_{1:i-1}), \quad (13)$$

where $z\widetilde{y}_{1:i-1}$ is the string formed by concatenating the prompt z with the tokens $\widetilde{y}_{1:i-1}$. Given the estimated prompt from the instruction set, we can estimate the NTPs by

$$\hat{q}_i = p(\widetilde{y}_i \mid \hat{z}\widetilde{y}_{1:i-1}), \quad i = 1, 2, \ldots, n,$$

where $y_{1:0} = \emptyset$.

Consider the instruction set defined by

$$\mathcal{P}_{\text{opt}} = \{z_1, \ldots, z_{n_{\text{opt}}}, \emptyset\}, \quad (14)$$

where $\{z_i\}_{i=1}^{n_{\text{opt}}}$ represents the set of possible prompts that may have been used to generate the watermarked text. To assess the quality of the instruction set, we propose using the Levenshtein distance between $\mathcal{P}_{\text{opt}}$ and the true prompt. Specifically, for any two strings y and $\widetilde{y}$, the Levenshtein distance is defined as

$$d_{\text{Lev}}(y, \widetilde{y}) := \begin{cases} d(y_{2:}, \widetilde{y}_{2:}) & \text{if } y_1 = \widetilde{y}_1, \\ 1 + \min\{d(y_{2:}, \widetilde{y}), d(y, \widetilde{y}_{2:})\} & \text{if } y_1 \neq \widetilde{y}_1, \end{cases} \quad (15)$$

where $d(y, \emptyset) = d(\emptyset, y) = |y|$, and $y_{2:} = y_2 y_3 \cdots y_{|y|}$. We define the Levenshtein distance between $\mathcal{P}_{\text{opt}}$ and the true prompt $y_{-n_0:0}$ as

$$d_{\text{Lev}}(\mathcal{P}_{\text{opt}}, y_{-n_0:0}) \coloneqq \min_{z\in\mathcal{P}_{\text{opt}}} d_{\text{Lev}}(z, y_{-n_0:0}).$$

A smaller value of $d_{\text{Lev}}(\mathcal{P}_{\text{opt}}, y_{-n_0:0})$ indicates a higher quality of $\mathcal{P}_{\text{opt}}$.

5 NUMERICAL EXPERIMENTS

We perform numerical experiments using two LLMs: `Meta-Llama-3-8B` from Meta (AI@Meta, 2024) and `Mistral-7B-v0.1` from Mistral AI (Jiang et al., 2023). In these experiments, we employ the EMS method as the watermarking technique for watermarked text generation. In this section, we present the results from the `Meta-Llama-3-8B` model, while the results for `Mistral-7B-v0.1` are provided in the Supplementary Materials Section B.

We compare four methods, all based on the randomization test proposed in Section 2.2, differing only in the choice of test statistics. The `baseline` method uses the original test statistic defined in (2). The `oracle` method employs the LR test statistic defined in (4). The `empty` method uses the test statistic in (9) with a shrinkage parameter $\lambda = 0.5$ and $q_i = q_{\emptyset,i}$, where $q_{\emptyset,i}$ represents the estimated NTPs with an empty prompt. The `optim` method uses the test statistic in (9) with a shrinkage parameter $\lambda = 0.5$ and $q_i = q_{\text{opt},i}$, where $q_{\text{opt},i}$ is the estimated NTPs with the prompt determined by solving the optimization problem in (13) using the instruction set $\mathcal{P}_{\text{opt}}$ defined in (14). In the numerical studies below, we fix $d_{\text{Lev}}(\mathcal{P}_{\text{opt}}, y_{-n_0:0}) = 5$. Results for other values of $d_{\text{Lev}}(\mathcal{P}_{\text{opt}}, y_{-n_0:0})$ are provided in the Supplementary Materials.

Each experiment is conducted using $T = 999$ permutations and with 1000 independent Monte Carlo replications.

5.1 Detection Power Without Attack

We begin by analyzing the detection power in the absence of any attacks. We fix $B = m$ and focus on the cases where $m \in \{10, 20, 30\}$, corresponding to approximately 8, 15, and 22 words, respectively (OpenAI, 2024). The detection power of the four methods with respect to m is presented in Figure 1 with p-value threshold set to be 0.05.

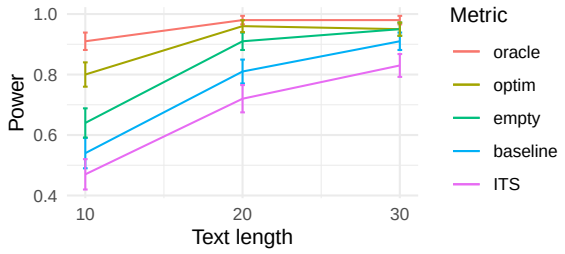

Figure 1: Detection power without attacks. Error bars represent one standard error.

In Figure 1, as the text length m increases, all methods exhibit higher power since longer texts lead to easier watermark detection. The `oracle` method consistently outperforms all other methods, aligning with our theoretical findings. All adaptive methods perform better than the `baseline` method. Specifically, when the text length is 10, the detection power of the `oracle` method increases by approximately 65% compared to the `baseline` method. Additionally, when implementing the `optim` method, the greedy algorithm selects the best prompt more than 95% of the time across all three text lengths. Here, the best prompt refers to the prompt within the instruction set that has the smallest Levenshtein distance to the true prompt. Notably, the `optim` method consistently outperforms the `empty` method, suggesting that a more accurately estimated prompt enhances the power of the proposed approach.

5.2 Detection Power With Attacks

In practice, the text being tested is not necessarily the same as watermarked text generated by the LLM. As mentioned in Section 3.4, we employ a transformation function $\mathcal{E}$, which can involve deletion, insertion, or substitution. In this work, we focus on semantically meaningful attacks, which are more realistic since the user-modified text will likely preserve its original meaning. We define semantically meaningful attacks as follows:

- **Deletion**. Suppose $y_{1:n}$ is generated by an autoregressive LLM. The attacked text $\widetilde{y}_{1:m} = y_{1:i_0, i_1+1:n}$ is obtained by deleting a random sentence $y_{i_0+1:i_1}$ of the original text.
- **Insertion**. Suppose $y_{1:n}$ is generated by an autoregressive LLM. The attacked text $\widetilde{y}_{1:m} = y_{1:i_0} z_{1:i_1} y_{i_0+1:n}$ is obtained by inserting a random relevant sentence $z_{1:i_1}$.
- **Substitution**. Define a binary vector $\boldsymbol{e} = \{e_1, \ldots, e_n\}$, where $e_i \in \{0, 1\}$ and $e_i = 1$ denotes a substitution at position i. The substitution attack is performed by having the autoregressive LLM generate each token $\widetilde{y}_i$ through the following process:

$$\widetilde{y}_i = \begin{cases} \Gamma(\xi_i, p(\cdot \mid \widetilde{y}_{-n_0:i-1})) & \text{if } e_i = 0, \\ \text{Multinomial}(p(\cdot \mid \widetilde{y}_{-n_0:i-1})) & \text{if } e_i = 1, \end{cases}$$

for $i = 1, \ldots, n$. By specifying an attack percentage, random $s\% \cdot n$ consecutive e_i's are set to be 1 while the rest are set to be 0.

Table 1: Detection power under substitution attack with p-value threshold 0.05.

Text length	$m = 10$			$m = 20$			$m = 30$		
Attack pct.	10%	20%	30%	10%	20%	30%	10%	20%	30%
`baseline`	0.518	0.405	0.280	0.743	0.587	0.457	0.836	0.730	0.568
`empty`	0.587	0.454	0.325	0.824	0.693	0.541	0.894	0.823	0.674
`optim`	0.686	0.550	0.401	0.877	0.751	0.618	0.931	0.870	0.733

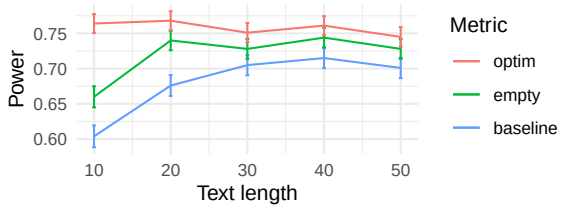

Figure 2: Detection powers under deletion attacks. Error bars represent one standard error.

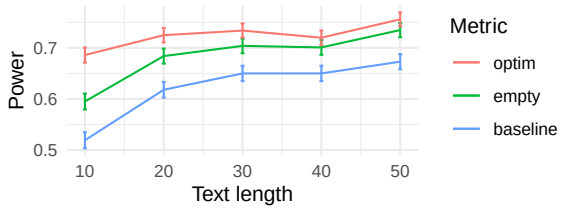

Figure 3: Detection powers under insertion attacks. Error bars represent one standard error.

In the presence of attacks, true prompts no longer exist, rendering the `oracle` method inapplicable. Therefore, we demonstrate the results using methods defined in Section 5 except the `oracle` method.

Table 1 presents the results for the substitution attack. As the attack percentage increases, the power of all methods decreases; conversely, as the text length increases, the power of all methods increases. In all settings, the two likelihood-based methods outperform the `baseline` method. Additionally, the `optim` method outperforms the `empty` method, further emphasizing that better prompt estimation leads to improved performance of the proposed method.

Table 2: Average attack percentages for different text lengths (%).

Text length	10	20	30	40	50
Deletion	9.57	18.98	25.18	28.37	32.45
Insertion	9.66	15.56	18.88	21.83	21.80

Results for deletion and insertion attacks are presented in Figures 2 and 3, respectively. For deletion and insertion attacks, to ensure semantic meaningfulness, we cannot control the attack percentage exactly. For example, we cannot control the length of the inserted sentence under an insertion attack. It is evident from the figures that in all settings, the two likelihood-based methods outperform the `baseline` method, indicating the benefit of the proposed approach. We emphasize that the power of the proposed methods does not increase monotonically with text length because the attack percentage also increases as text length increases. The average attack percentage is summarized in Table 2.

6 DISCUSSION

In this study, we employ a greedy method to search for the most likely prompt within an instruction set. However, this process can be computationally intensive, especially with a large instruction set. To reduce the computational burden, it is beneficial to explore alternative approaches for estimating the NTPs. Promising methods in this direction include in-context learning (Brown, 2020) and model inversion, which involves training an inverse model (Morris et al., 2024) to predict prompts from the responses of an LLM.

While the LR test is the optimal testing procedure in the absence of an attack, it remains unclear whether an optimal testing procedure exists in the presence of attacks. As an initial step toward addressing this issue, it is necessary to formally define an attack model, a challenging task that warrants further investigation.

Most existing watermark generation and detection methods do not consider the semantics of texts. An intriguing area for future research is the development of methods for generating and detecting watermarks at the semantic level with certain forms of optimality.

References

Aaronson, S. (2023). Watermarking of large language models. `https://simons.berkeley.edu/talks/scott-aaronson-ut-austin-openai-2023-08-17`.

AI@Meta (2024). Llama 3 model card. https://github.com/meta-llama/llama3/blob/main/MODEL_CARD.md.

Brown, T. B. (2020). Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*.

Cai, Z., Liu, S., Wang, H., Zhong, H., and Li, X. (2024). Towards better statistical understanding of watermarking llms. *arXiv preprint arXiv:2403.13027*.

Edgington, E. and Onghena, P. (2007). *Randomization tests*. Chapman and Hall/CRC.

Good, P. (2013). *Permutation tests: a practical guide to resampling methods for testing hypotheses*. Springer Science & Business Media.

Hu, Z., Chen, L., Wu, X., Wu, Y., Zhang, H., and Huang, H. (2023). Unbiased watermark for large language models. *arXiv preprint arXiv:2310.10669*.

Huang, B., Zhu, B., Zhu, H., Lee, J. D., Jiao, J., and Jordan, M. I. (2023). Towards optimal statistical watermarking. *arXiv preprint arXiv:2312.07930*.

Jiang, A. Q., Sablayrolles, A., Mensch, A., Bamford, C., Chaplot, D. S., Casas, D. d. l., Bressand, F., Lengyel, G., Lample, G., Saulnier, L., et al. (2023). Mistral 7b. *arXiv preprint arXiv:2310.06825*.

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

Kirchenbauer, J., Geiping, J., Wen, Y., Shu, M., Saifullah, K., Kong, K., Fernando, K., Saha, A., Goldblum, M., and Goldstein, T. (2023b). On the reliability of watermarks for large language models. *arXiv preprint arXiv:2306.04634*.

Kuditipudi, R., Thickstun, J., Hashimoto, T., and Liang, P. (2024). Robust distortion-free watermarks for language models. *Transactions on Machine Learning Research*.

Li, X., Li, G., and Zhang, X. (2024a). Segmenting watermarked texts from language models. *arXiv preprint arXiv:2410.20670*.

Li, X., Ruan, F., Wang, H., Long, Q., and Su, W. J. (2024b). Robust detection of watermarks for large language models under human edits. *arXiv preprint arXiv:2411.13868*.

Li, X., Ruan, F., Wang, H., Long, Q., and Su, W. J. (2024c). A statistical framework of watermarks for large language models: Pivot, detection efficiency and optimal rules. *arXiv preprint arXiv:2404.01245*.

Liu, A., Pan, L., Lu, Y., Li, J., Hu, X., Zhang, X., Wen, L., King, I., Xiong, H., and Yu, P. (2024). A survey of text watermarking in the era of large language models. *ACM Computing Surveys*.

Liu, Y. and Bu, Y. (2024). Adaptive text watermark for large language models. *arXiv preprint arXiv:2401.13927*.

Liu, Y. and Xie, J. (2020). Cauchy combination test: a powerful test with analytic p-value calculation under arbitrary dependency structures. *Journal of the American Statistical Association*, 115(529):393–402.

Morris, J. X., Zhao, W., Chiu, J. T., Shmatikov, V., and Rush, A. M. (2024). Language model inversion. In *The Twelfth International Conference on Learning Representations*.

Nemecek, A., Jiang, Y., and Ayday, E. (2024). Topic-based watermarks for llm-generated text. *arXiv preprint arXiv:2404.02138*.

OpenAI (2024). What are tokens and how to count them?

Papandreou, G. and Yuille, A. L. (2011). Perturb-and-map random fields: Using discrete optimization to learn and sample from energy models. In *2011 International Conference on Computer Vision*, pages 193–200.

Taori, R., Gulrajani, I., Zhang, T., Dubois, Y., Li, X., Guestrin, C., Liang, P., and Hashimoto, T. B. (2023). Stanford alpaca: An instruction-following llama model. https://github.com/tatsu-lab/stanford_alpaca.

Vershynin, R. (2018). *High-dimensional probability: An introduction with applications in data science*, volume 47. Cambridge university press.

Wu, Y., Hu, Z., Zhang, H., and Huang, H. (2023). Dipmark: A stealthy, efficient and resilient watermark for large language models. *arXiv preprint arXiv:2310.07710*.

Zhao, X., Li, L., and Wang, Y.-X. (2024). Permute-and-flip: An optimally robust and watermarkable decoder for llms. *arXiv preprint arXiv:2402.05864*.

Zhao, X., Wang, Y.-X., and Li, L. (2023). Protecting language generation models via invisible watermarking. In *International Conference on Machine Learning*, pages 42187–42199. PMLR.

Zheng, Y., Zhang, R., Zhang, J., Ye, Y., and Luo, Z. (2024). Llamafactory: Unified efficient fine-tuning of 100+ language models. *arXiv preprint arXiv:2403.13372*.

Checklist

1. For all models and algorithms presented, check if you include:

(a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [Yes]
(b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [Yes]
(c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [Yes]

2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. [Yes]
   (b) Complete proofs of all theoretical results. [Yes] Proofs are available in the supplementary materials if not present in the main content.
   (c) Clear explanations of any assumptions. [Yes]

3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [Yes]
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [Yes]
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [Yes]
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [Yes]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. [Yes]
   (b) The license information of the assets, if applicable. [Yes]
   (c) New assets either in the supplemental material or as a URL, if applicable. [Yes]
   (d) Information about consent from data providers/curators. [Yes]
   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Not Applicable]

5. If you used crowdsourcing or conducted research with human subjects, check if you include:
   (a) The full text of instructions given to participants and screenshots. [Not Applicable]
   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Not Applicable]
   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Not Applicable]

A likelihood based approach for watermark detection: Supplementary Materials

A PROOFS OF THE MAIN RESULTS

Proof of Theorem 1.
(i) For simplicity of notation, let $\varphi \coloneqq \phi(\xi_{1:n}, \widetilde{y}_{1:n})$ and $\varphi^{(t)} \coloneqq \phi(\xi^{(t)}_{1:n}, \widetilde{y}_{1:n})$ for $t = 1, \ldots, T$. Under the null hypothesis, $\xi_{1:n}$ is independent of $\widetilde{y}_{1:n}$, which implies that the pairs $(\xi_{1:n}, \widetilde{y}_{1:n}), (\xi^{(1)}_{1:n}, \widetilde{y}_{1:n}), \ldots, (\xi^{(T)}_{1:n}, \widetilde{y}_{1:n})$ follow the same distribution. Therefore, $\varphi, \varphi^{(1)}, \ldots, \varphi^{(T)}$ are exchangeable. This exchangeability ensures that the rank of φ relative to $\{\varphi, \varphi^{(1)}, \ldots, \varphi^{(T)}\}$ is uniformly distributed. Let the order statistics be denoted as $\varphi_{(1)} \leq \cdots \leq \varphi_{(T+1)}$. Then, we have

$$P\left(\varphi = \varphi_{(j)}\right) = \frac{1}{T+1}, \qquad j = 1, \ldots, T+1.$$

Thus, for $j = 1, \ldots, T+1$, we obtain

$$P\left(\mathsf{p}_T \leq \frac{j}{T+1}\right) = P\left(\varphi \in \left\{\varphi_{(T+2-j)}, \ldots, \varphi_{(T+1)}\right\}\right) = \frac{j}{T+1}.$$

Finally, we have

$$P\left(\mathsf{p}_T \leq \alpha\right) = \frac{\lfloor (T+1)\alpha \rfloor}{T+1} \leq \alpha.$$

(ii) By Chebyshev's inequality we have

$$P\left(|\phi(\xi'_{1:n}, \widetilde{y}_{1:n}) - \mathsf{E}_{\xi'}| \geq \epsilon \mathrm{Sd}_{\xi'} | \mathcal{F}_n\right) \leq \frac{\mathrm{Var}(\phi(\xi'_{1:n}, \widetilde{y}_{1:n})|\mathcal{F}_n)}{\epsilon^2 \mathrm{Sd}^2_{\xi'}} = \frac{1}{\epsilon^2},$$

for all $\epsilon \geq 0$. Since $\mathsf{E}_{\xi'} = 0$, we have $\phi(\xi'_{1:n}, \widetilde{y}_{1:n}) = O_p(\mathrm{Sd}_{\xi'})$ given $\mathcal{F}_n$.

Let the distribution of $\phi(\xi'_{1:n}, \widetilde{y}_{1:n})$ conditional on $\mathcal{F}_n$ be denoted by F, and the empirical distribution of $\{\varphi^{(t)}\}_{t=0}^{T}$ by F_T, where we set $\varphi^{(0)} = \varphi$. Let $q_{1-\alpha,T} = \varphi_{(T+2-j_\alpha)}$ with $j_\alpha = \lfloor (T+1)\alpha \rfloor$. Note that $F_T(q_{1-\alpha,T}) = 1 - (j_\alpha - 1)/(T+1)$. Our test rejects the null whenever $\mathsf{p}_T \leq \alpha$, which is equivalent to rejecting the null if $\varphi \geq \varphi_{(T+2-j_\alpha)}$. By the Dvoretzky–Kiefer–Wolfowitz inequality, we have

$$P(|F_T(q_{1-\alpha,T}) - F(q_{1-\alpha,T})| > \epsilon | \mathcal{F}_n) \leq P\left(\sup_x |F_T(x) - F(x)| > \epsilon | \mathcal{F}_n\right) \leq C_1 \exp(-2T\epsilon^2)$$

for some constant $C_1 > 0$, which implies that, with probability greater than $1 - C_1 \exp(-2T\epsilon^2)$, $F(q_{1-\alpha,T}) < 1 - (j_\alpha - 1)/(T+1) + 2\epsilon$. Define $F^{-1}(t) = \inf\{s : F(s) \geq t\}$ and the event $\mathcal{A}_T = \{q_{1-\alpha,T} < F^{-1}(1 - (j_\alpha - 1)/(T+1) + 2\epsilon)\}$. Then we have $P(\mathcal{A}_T | \mathcal{F}_n) \geq 1 - C_1 \exp(-2T\epsilon^2)$. In addition, as $\phi(\xi'_{1:n}, \widetilde{y}_{1:n}) = O_p(\mathrm{Sd}_{\xi'})$, we have $F^{-1}(s) = O(\mathrm{Sd}_{\xi'})$ for any $s < 1$.

Notice that $\mathrm{Sd}_\xi^{-1}\big(\phi(\xi_{1:n},\widetilde{y}_{1:n})-\mathsf{E}_\xi)|\mathcal{F}_n]\big)=O_p(1)$. Hence, for $T>2/\epsilon-1$,

$$\begin{aligned}
&P(\phi(\xi_{1:n},\widetilde{y}_{1:n})\geq q_{1-\alpha,T}|\mathcal{F}_n)\\
\geq&P\left(\mathrm{Sd}_\xi^{-1}\big(\phi(\xi_{1:n},\widetilde{y}_{1:n})-\mathsf{E}_\xi\big)+\mathrm{Sd}_\xi^{-1}\mathsf{E}_\xi\geq \mathrm{Sd}_\xi^{-1}q_{1-\alpha,T},\mathcal{A}_T|\mathcal{F}_n\right)\\
\geq&P\Big(O_p(1)+\mathrm{Sd}_\xi^{-1}\mathsf{E}_\xi\geq \mathrm{Sd}_\xi^{-1}F^{-1}(1-(j_\alpha-1)/(T+1)+2\epsilon),\mathcal{A}_T|\mathcal{F}_n\Big)\\
\geq&P\Big(O_p(1)+\mathrm{Sd}_\xi^{-1}\mathsf{E}_\xi\geq \mathrm{Sd}_\xi^{-1}F^{-1}(1-\alpha+3\epsilon),\mathcal{A}_T|\mathcal{F}_n\Big)\\
\geq&P\Big(O_p(1)+\mathrm{Sd}_\xi^{-1}\mathsf{E}_\xi\geq \mathrm{Sd}_\xi^{-1}O(\mathrm{Sd}_{\xi'}),\mathcal{A}_T|\mathcal{F}_n\Big)\\
\geq&1-C_1\exp(-2T\epsilon^2)+o(1),
\end{aligned}$$

where we have used Condition $\mathrm{Sd}_\xi=o(\mathsf{E}_\xi)$ and $\mathrm{Sd}_{\xi'}=o(\mathsf{E}_\xi)$ to establish the convergence. □

Proof of Theorem 2. Define

$$f_1(\xi_{1:n},\widetilde{y}_{1:n})=\prod_{i=1}^n\exp\left(\frac{\log(\xi_{i,\widetilde{y}_i})}{p_i}\right)\frac{1}{p_i},\quad f_0(\xi_{1:n},\widetilde{y}_{1:n})=\prod_{i=1}^n\exp\left(\log(\xi_{i,\widetilde{y}_i})\right),$$

and the likelihood ratio

$$\lambda(\xi_{1:n},\widetilde{y}_{1:n})=\frac{f_1(\xi_{1:n},\widetilde{y}_{1:n})}{f_0(\xi_{1:n},\widetilde{y}_{1:n})}.$$

It is straightforward to verify that there exists a constant $k_{1-\alpha}$ such that $L(\xi_{1:n},\widetilde{y}_{1:n})>q_{1-\alpha}$ if and only if $\lambda(\xi_{1:n},\widetilde{y}_{1:n})>k_{1-\alpha}$. Hence,

$$\psi(\xi_{1:n},\widetilde{y}_{1:n})=\mathbf{1}\left\{\lambda(\xi_{1:n},\widetilde{y}_{1:n})>k_{1-\alpha}\right\}.$$

Consider another test function ψ' with size α, meaning that $\mathbb{E}_0[\psi'(\xi_{1:n},\widetilde{y}_{1:n})]\leq\alpha$. We claim that

$$\int\big(\psi(\xi_{1:n},\widetilde{y}_{1:n})-\psi'(\xi_{1:n},\widetilde{y}_{1:n})\big)\big(f_1(\xi_{1:n},\widetilde{y}_{1:n})-k_1f_0(\xi_{1:n},\widetilde{y}_{1:n})\big)\,d\xi\geq 0. \tag{A.1}$$

To see this, consider the following cases:

- If $f_1(\xi_{1:n},\widetilde{y}_{1:n})>k_1f_0(\xi_{1:n},\widetilde{y}_{1:n})$, then $\psi(\xi_{1:n},\widetilde{y}_{1:n})=1$. Since $\psi'(\xi_{1:n},\widetilde{y}_{1:n})\leq 1$, the integral is non-negative.
- If $f_1(\xi_{1:n},\widetilde{y}_{1:n})<k_1f_0(\xi_{1:n},\widetilde{y}_{1:n})$, then $\psi(\xi_{1:n},\widetilde{y}_{1:n})=0$. Since $\psi'(\xi_{1:n},\widetilde{y}_{1:n})\geq 0$, the integral is non-negative.
- If $f_1(\xi_{1:n},\widetilde{y}_{1:n})=k_1f_0(\xi_{1:n},\widetilde{y}_{1:n})$, the integral is zero.

Thus, by rearranging (A.1), we obtain

$$\begin{aligned}
&\int\big(\psi(\xi_{1:n},\widetilde{y}_{1:n})-\psi'(\xi_{1:n},\widetilde{y}_{1:n})\big)f_1(\xi_{1:n},\widetilde{y}_{1:n})\,d\xi\\
\geq&k_1\int\big(\psi(\xi_{1:n},\widetilde{y}_{1:n})-\psi'(\xi_{1:n},\widetilde{y}_{1:n})\big)f_0(\xi_{1:n},\widetilde{y}_{1:n})\,d\xi\\
\geq&0,
\end{aligned}$$

where we use the fact that $\mathbb{E}_0[\psi']\leq\alpha=\mathbb{E}_0[\psi]$ to get the second inequality. Therefore, we conclude that $\mathbb{E}_1[\psi]\geq\mathbb{E}_1[\psi']$, meaning that ψ is the most powerful test at level α. □

Proof of Theorem 3. Denote $S_1=\sum_{i=1}^n(1-p_i)^2/p_i$ and $S_2=\sum_{i=1}^n(1-p_i)^2/p_i^2$. Write $d_i=p_i-q_i$. The goal is to show

$$\frac{\sum_{i=1}^n(1-q_i)(1-p_i)/q_i}{\sqrt{\sum_{i=1}^n(1-q_i)^2/q_i^2}}\to\infty. \tag{A.2}$$

Consider the summand in the numerator of the LHS of (A.2). Since $d_i/p_i = o(1)$, by Taylor expansion, we have

$$\begin{aligned}\frac{(1-q_i)(1-p_i)}{q_i} &= \frac{(1-p_i+d_i)(1-p_i)}{p_i\left(1-\frac{d_i}{p_i}\right)} \\ &= \frac{(1-p_i)^2}{p_i}\left\{1+\frac{d_i}{p_i}+o\left(\frac{d_i}{p_i}\right)\right\}+\frac{d_i(1-p_i)}{p_i}\left\{1+\frac{d_i}{p_i}+o\left(\frac{d_i}{p_i}\right)\right\} \\ &= \frac{(1-p_i)^2}{p_i}+\frac{d_i}{p_i}\left\{\frac{(1-p_i)^2}{p_i}+(1-p_i)\right\}+o\left(\frac{d_i}{p_i}\right).\end{aligned}$$

Ignoring the smaller order term, we have

$$\sum_{i=1}^n \frac{(1-q_i)(1-p_i)}{q_i} \approx S_1+\sum_{i=1}^n \frac{d_i}{p_i}\left\{\frac{(1-p_i)^2}{p_i}+(1-p_i)\right\}.$$

For the denominator term, we have

$$\begin{aligned}\frac{(1-q_i)^2}{q_i^2} &= \frac{(1-p_i+d_i)^2}{(p_i-d_i)^2} \\ &= \frac{\left\{(1-p_i)^2+2(1-p_i)d_i+d_i^2\right\}}{p_i^2}\left\{1+2\frac{d_i}{p_i}+o\left(\frac{d_i}{p_i}\right)\right\} \\ &= \frac{(1-p_i^2)}{p_i^2}+\frac{d_i}{p_i}\left(\frac{2(1-p_i)^2}{p_i^2}+\frac{2(1-p_i)}{p_i}\right)+o\left(\frac{d_i}{p_i}\right).\end{aligned}$$

Again, ignoring the smaller order term, we obtain

$$\begin{aligned}\sum_{i=1}^n \frac{(1-p_i+d_i)^2}{(p_i-d_i)^2} &\approx \sum_{i=1}^n \frac{(1-p_i)^2}{p_i^2}+\sum_{i=1}^n \frac{d_i}{p_i}\left\{\frac{2(1-p_i)}{p_i}+\frac{2(1-p_i)^2}{p_i^2}\right\} \\ &= \left\{\sum_{i=1}^n \frac{(1-p_i)^2}{p_i^2}\right\}\left[1+\frac{\sum_{i=1}^n 2\left\{(1-p_i)/p_i+(1-p_i)^2/p_i^2\right\}d_i/p_i}{\sum_{i=1}^n(1-p_i)^2/p_i^2}\right],\end{aligned}$$

and

$$\sqrt{\sum_{i=1}^n \frac{(1-p_i+d_i)^2}{(p_i-d_i)^2}} \approx \sqrt{S_2}\sqrt{\left(1+\frac{\sum_{i=1}^n 2(1-p_i)/p_i\cdot d_i/p_i}{\sum_{i=1}^n(1-p_i)^2/p_i^2}+\frac{\sum_{i=1}^n 2(1-p_i)^2/p_i^2\cdot d_i/p_i}{\sum_{i=1}^n(1-p_i)^2/p_i^2}\right)}.$$

Hence, we have

$$\frac{\sum_{i=1}^n(1-q_i)(1-p_i)/q_i}{\sqrt{\sum_{i=1}^n(1-q_i)^2/q_i^2}} = \frac{S_1+\sum_{i=1}^n \frac{d_i}{p_i}\left(\frac{(1-p_i)^2}{p_i}+(1-p_i)\right)}{\sqrt{S_2}\sqrt{\left(1+\frac{\sum_{i=1}^n 2(1-p_i)/p_i\cdot d_i/p_i}{\sum_{i=1}^n(1-p_i)^2/p_i^2}+\frac{\sum_{i=1}^n 2d_i/p_i\cdot(1-p_i)^2/p_i^2}{\sum_{i=1}^n(1-p_i)^2/p_i^2}\right)}}+\text{small order terms.}$$

Since $d_i = o(p_i)$, we have $\sum_{i=1}^n d_i/p_i\cdot(1-p_i)^2/p_i = o\left(\sum_{i=1}^n(1-p_i)^2/p_i\right)$. By the same derivation, we have $\sum_{i=1}^n d_i/p_i\cdot(1-p_i)^2/p_i^2 = o\left(\sum_{i=1}^n(1-p_i)^2/p_i^2\right)$. Since $d_i = o(1-p_i)$, we have $(1-p_i)d_i = o\big((1-p_i)^2\big)$, which implies that $\sum_{i=1}^n d_i/p_i\cdot(1-p_i) = o\left(\sum_{i=1}^n(1-p_i)^2/p_i\right)$. By the same derivation, we have $\sum_{i=1}^n d_i/p_i\cdot(1-p_i)/p_i = o\left(\sum_{i=1}^n(1-p_i)^2/p_i^2\right)$. Combining all the above results, we obtain

$$\frac{\sum_{i=1}^n(1-q_i)(1-p_i)/q_i}{\sqrt{\sum_{i=1}^n(1-q_i)^2p_i^2/q_i^2}} = \frac{S_1+o(S_1)}{\sqrt{S_2}\sqrt{1+o(1)}}+\text{small order terms.}$$

Hence, (8) holds because of (A.2), which completes the proof. □

Proof of Theorem 4. Suppose a random variable $X \sim \exp(\lambda)$. Then, X is a sub-exponential random variable, and its sub-exponential norm is given by $\|X\|_{\psi_1} = 2/\lambda$. For more details on sub-exponential random variables, please refer to Vershynin (2018).

Recall that $-\log(\xi_{i,y_i}) \mid \mathcal{F}_m \sim \exp(1/p_i)$, where $\mathcal{F}_m = [\widetilde{y}_{1:m}, y_{-n_0:0}]$. Let $S_\xi = -\sum_{i=1}^n w_i \log(\xi_{i,y_i})$. By Bernstein's inequality (Vershynin, 2018, Thm. 2.8.2), it follows that

$$P\left(|S_\xi - \mathbb{E}[S_\xi]| \geq t \mid \mathcal{F}_m\right) \leq 2\exp\left(-c\min\left\{\frac{t^2}{K^2\sum_{i=1}^n w_i^2}, \frac{t}{K\max_i w_i}\right\}\right),$$

where $K = \max_i \| - \log(\xi_{i,y_i})\|_{\psi_1} \leq 2$ and $c > 0$ is a fixed constant. Hence, we obtain

$$\begin{aligned}
&P\left(|\phi(\xi_{a:a+B-1}, \widetilde{y}_{b:b+B-1}) - \mathbb{E}\left[\phi(\xi_{a:a+B-1}, \widetilde{y}_{b:b+B-1}) \mid \mathcal{F}_m\right]| > t \mid \mathcal{F}_m\right) \\
&\leq 2\exp\left(-c\min\left\{\frac{B^2t^2}{4\sum_{i=b}^{b+B-1} w_i^2}, \frac{Bt}{2\max_{b\leq i\leq b+B-1} w_i}\right\}\right) \\
&\leq 2\exp\left(-c\min\left\{\frac{B^2t^2}{4\Omega_{\max}}, \frac{Bt}{2\max_{i\in[n]} w_i}\right\}\right).
\end{aligned}$$

By the union bound, we have

$$\begin{aligned}
&P\Bigg(\max_{1\leq a\leq n-B+1, 1\leq b\leq m-B+1} |\phi(\xi_{a:a+B-1}, \widetilde{y}_{b:b+B-1}) \\
&- \mathbb{E}[\phi(\xi_{a:a+B-1}, \widetilde{y}_{b:b+B-1})|\mathcal{F}_m]| > t|\mathcal{F}_m\Bigg) \\
\leq& 2(n-B+1)(m-B+1)\exp\left(-c\min\left\{\frac{B^2t^2}{4\Omega_{\max}}, \frac{Bt}{2\max_{i\in[n]} w_i}\right\}\right).
\end{aligned}$$

Integrating out the strings in $y_{1:n}$ that are not contained in $\widetilde{y}_{1:m}$, we obtain

$$\begin{aligned}
&P\Bigg(\max_{1\leq a\leq n-B+1, 1\leq b\leq m-B+1} |\phi(\xi_{a:a+B-1}, \widetilde{y}_{b:b+B-1}) \\
&- \mathbb{E}[\phi(\xi_{a:a+B-1}, \widetilde{y}_{b:b+B-1})|\widetilde{y}_{1:m}, y_{-n_0:n}]| > t|\mathcal{F}_m\Bigg) \\
\leq& 2(n-B+1)(m-B+1)\exp\left(-c\min\left\{\frac{B^2t^2}{4\Omega_{\max}}, \frac{Bt}{2\max_{i\in[n]} w_i}\right\}\right).
\end{aligned}$$

Thus, conditional on $\mathcal{F}_m$, we have

$$\max_{a,b} |\mathbb{E}\left[\phi\left(\xi_{a:a+B-1}, \widetilde{y}_{b:b+B-1}\right) \mid \mathcal{F}_m\right] - \phi\left(\xi_{a:a+B-1}, \widetilde{y}_{b:b+B-1}\right)| = O(C_{N,B}).$$

Note that

$$\begin{aligned}
\phi(\xi_{1:n}, \widetilde{y}_{1:m}) &= \max_{a,b} \phi(\xi_{a:a+B-1}, \widetilde{y}_{b:b+B-1}) \\
&\geq \max_{a,b} \mathbb{E}\left[\phi\left(\xi_{a:a+B-1}, \widetilde{y}_{b:b+B-1}\right) \mid \mathcal{F}_m\right] \\
&\quad - \max_{a,b} |\mathbb{E}\left[\phi\left(\xi_{a:a+B-1}, \widetilde{y}_{b:b+B-1}\right) \mid \mathcal{F}_m\right] - \phi\left(\xi_{a:a+B-1}, \widetilde{y}_{b:b+B-1}\right)| \\
&= \max_{a,b} \mathbb{E}\left[\phi\left(\xi_{a:a+B-1}, \widetilde{y}_{b:b+B-1}\right) \mid \mathcal{F}_m\right] + O(C_{N,B}).
\end{aligned}$$

On the other hand, for a randomly generated key $\xi'_{1:n}$, we have $-\log(\xi'_{i,y_i}) \sim \exp(1)$ and

$\mathbb{E}\left[\phi\left(\xi'_{a:a+B-1}, \widetilde{y}_{b:b+B-1}\right) \mid \mathcal{F}_m\right] = 0$ for all a, b. Using the same argument, we obtain

$$\begin{aligned}
&P\left(\phi(\xi'_{1:n}, \widetilde{y}_{1:m}) > t \mid \mathcal{F}_m\right) \\
&= P\left(\max_{1\le a\le n-B+1,\, 1\le b\le m-B+1} \phi\left(\xi'_{a:a+B-1}, \widetilde{y}_{b:b+B-1}\right) > t \mid \mathcal{F}_m\right) \\
&\le 2(n-B+1)(m-B+1)\exp\left(-c\min\left\{\frac{B^2t^2}{4\Omega_{\max}}, \frac{Bt}{2\max_{i\in[n]} w_i}\right\}\right),
\end{aligned}$$

which suggests that $F^{-1}(s) = O(C_{N,B})$, where F is the distribution of $\phi(\xi'_{1:n}, \widetilde{y}_{1:m})$ conditional on $\mathcal{F}_m$ and $F^{-1}(t) = \inf\{s : F(s) \ge t\}$. The remaining arguments are similar to those in the proof of Theorem 1, and we omit the details. □

Proof of Lemma 1. Note that $E_{ik} \coloneqq -\log(\xi_{ik})/p(k \mid y_{-n_0:i-1}) \sim \mathrm{Exp}(p(k \mid y_{-n_0:i-1}))$. Since $\xi'_{1:n}$ is independent of $y_{1:n}$, we have $-\log(\xi'_{i,y_i}) \mid y_{-n_0:n} \sim \mathrm{Exp}(1)$. Given $y_{-n_0:n}$, we know that $E_{i,y_i} = \min_{1\le k\le V} E_{ik}$, which implies $-\log(\xi_{i,y_i})/p_i \mid y_{-n_0:n} \sim \mathrm{Exp}(1)$. It is worth noting that

$$\begin{aligned}
P\left(-\log(\xi_{i,y_i}) \ge t\right) &= P\left(-\frac{\log(\xi_{i,y_i})}{p(y_i \mid y_{-n_0:i-1})} \ge \frac{t}{p(y_i \mid y_{-n_0:i-1})}\right) \\
&= \exp\left(-\frac{t}{p(y_i \mid y_{-n_0:i-1})}\right).
\end{aligned}$$

Thus, $-\log(\xi_{i,y_i}) \mid y_{-n_0:n} \sim \mathrm{Exp}\left(1/p_i\right)$. □

Proof of Corollary 1. By Lemma 1, given test statistic $\phi(\xi_{1:n}, \widetilde{y}_{1:n})$ defined by (5), we have

$$\begin{aligned}
&\mathbb{E}[\phi(\xi'_{1:n}, \widetilde{y}_{1:n})] = 0, \\
&\mathrm{Var}\big(\phi(\xi'_{1:n}, \widetilde{y}_{1:n})\big) = \frac{1}{n}\sum_{i=1}^n w_i^2, \\
&\mathbb{E}[\phi(\xi_{1:n}, \widetilde{y}_{1:n})] = \frac{1}{n}\sum_{i=1}^n w_i(1-p_i), \\
&\mathrm{Var}\big(\phi(\xi_{1:n}, \widetilde{y}_{1:n})\big) = \frac{1}{n}\sum_{i=1}^n w_i^2 p_i^2 \le \mathrm{Var}\big(\phi(\xi'_{1:n}, \widetilde{y}_{1:n})\big).
\end{aligned}$$

Hence, by Theorem 1, the power of the randomization test converges to one if

$$\frac{\sum_{i=1}^n w_i(1-p_i)}{\sqrt{\sum_{i=1}^n w_i^2}} \to +\infty.$$

□

B ADDITIONAL NUMERICAL RESULTS

B.1 Additional Numerical Results For `Meta-Llama-3-8B`

We also evaluate the impact of the hyperparameter B. We consider three values of B: $B = 0.3m$, $0.6m$, and m, where m denotes the text length. Results under deletion and insertion attacks are provided in Figures B.1 and B.2, respectively. In most cases, the two likelihood-based methods outperform the `baseline` method. When $m = 10$, the optimal selection is $B = 10$, as the token length is very short. If the block size is too small, the maximum test cannot detect any watermark. As m increases, $B = 0.6m$ and $B = 0.3m$ outperform $B = m$ because not every token is related to the watermark key sequence, and using only a subset of tokens yields better performance.

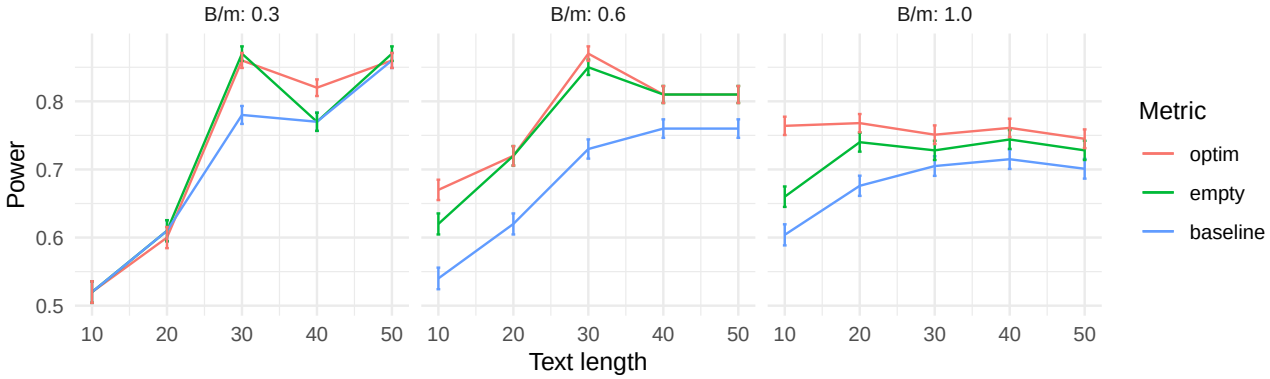

Figure B.1: Detection powers under deletion attacks using `Meta-Llama-3-8B`. Error bars represent one standard error.

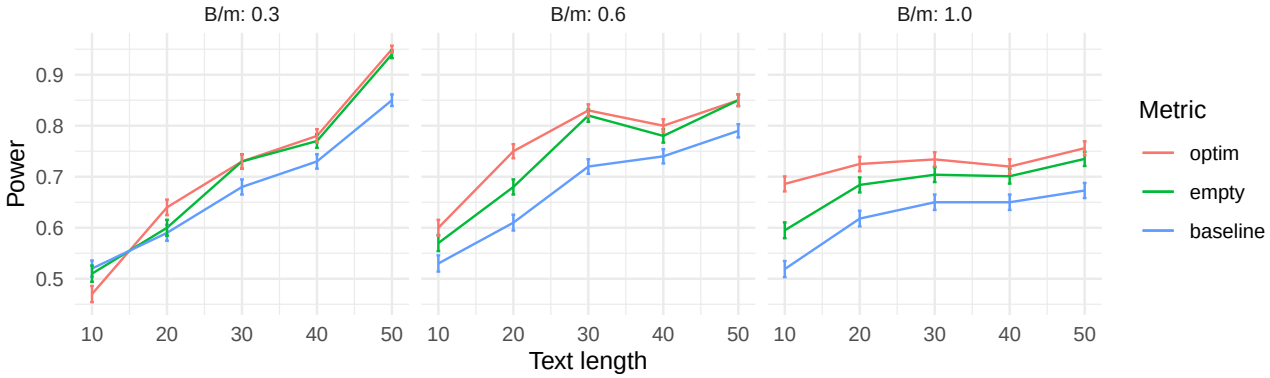

Figure B.2: Detection powers under insertion attacks using `Meta-Llama-3-8B`. Error bars represent one standard error.

B.2 Additional Numerical Results For `Mistral-7B-v0.1`

In this section, we present the results for `Mistral-7B-v0.1`. Detection powers with respect to text length under deletion and insertion attacks are shown in Figures B.3 and B.4, respectively. The phenomena are consistent with the results obtained using the LLM `Meta-Llama-3-8B`, indicating that our method is stable across different LLMs.

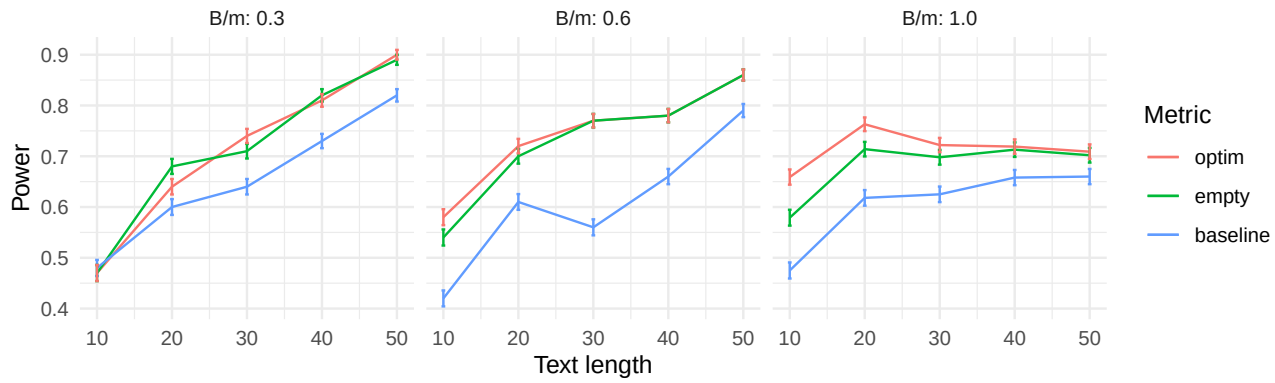

Figure B.3: Detection powers under deletion attacks using `Mistral-7B-v0.1`. Error bars represent one standard error.

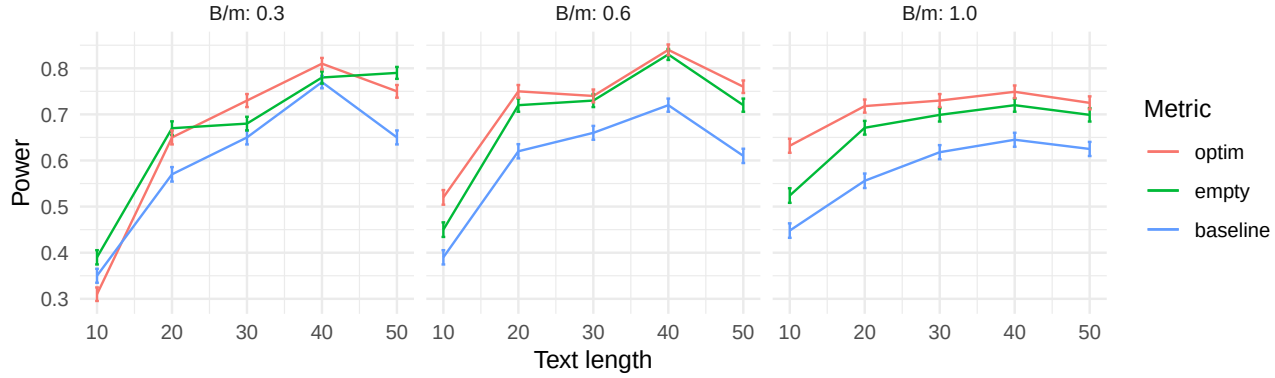

Figure B.4: Detection powers under insertion attacks using `Mistral-7B-v0.1`. Error bars represent one standard error.

C SELECTION OF HYPERPARAMETERS

In this section, we perform an ablation study on the hyperparameter λ introduced in the shrinkage regularization (9). Note that the `baseline` method corresponds to $\lambda = 0$. Tables C.1, C.2, and C.3 present the detection power under substitution, deletion, and insertion attacks, respectively. When using $\emptyset$ as the estimated prompt, setting $\lambda = 1$, i.e., no regularization, results in very low power for the proposed method, indicating that regularization is necessary. For λ values ranging from 0.1 to 0.9, the proposed method outperforms the `baseline` method, demonstrating its robustness to the choice of λ. We observe that selecting $\lambda = 0.5$ achieves stable performance across all settings.

Table B.1: Average attack percentages for different text lengths using `Mistral-7B-v0.1`.

Text length	10	20	30	40	50
Deletion	8.56	14.59	23.21	24.72	29.31
Insertion	7.25	13.99	17.96	19.49	21.98

Table C.1: Detection power using different λ under 30% substitution attack.

Model	Meta-Llama-3-8B					Mistral-7B-v0.1				
Text length	10	20	30	40	50	10	20	30	40	50
baseline	0.280	0.457	0.568	0.693	0.753	0.241	0.345	0.493	0.592	0.645
$S(p_{\emptyset,1:n}, 1.0)$	0.220	0.212	0.211	0.220	0.206	0.207	0.222	0.262	0.264	0.302
$S(p_{\emptyset,1:n}, 0.9)$	0.347	0.550	0.638	0.749	0.828	0.293	0.465	0.638	0.699	0.779
$S(p_{\emptyset,1:n}, 0.8)$	0.347	0.554	0.662	0.776	0.844	0.297	0.467	0.653	0.716	0.803
$S(p_{\emptyset,1:n}, 0.7)$	0.337	0.550	0.672	0.779	0.856	0.289	0.471	0.665	0.726	0.816
$S(p_{\emptyset,1:n}, 0.6)$	0.334	0.545	0.673	0.787	0.851	0.276	0.465	0.663	0.731	0.814
$S(p_{\emptyset,1:n}, 0.5)$	0.325	0.541	0.674	0.791	0.853	0.277	0.458	0.660	0.735	0.808
$S(p_{\emptyset,1:n}, 0.4)$	0.319	0.531	0.668	0.780	0.848	0.270	0.453	0.645	0.731	0.797
$S(p_{\emptyset,1:n}, 0.3)$	0.314	0.519	0.655	0.774	0.845	0.263	0.427	0.623	0.721	0.783
$S(p_{\emptyset,1:n}, 0.2)$	0.306	0.512	0.633	0.754	0.824	0.257	0.410	0.591	0.697	0.760
$S(p_{\emptyset,1:n}, 0.1)$	0.297	0.494	0.609	0.732	0.795	0.255	0.372	0.544	0.645	0.721
$S(p_{\text{opt},1:n}, 1.0)$	0.493	0.499	0.488	0.504	0.484	0.406	0.433	0.460	0.463	0.447
$S(p_{\text{opt},1:n}, 0.9)$	0.443	0.620	0.702	0.786	0.837	0.366	0.533	0.648	0.744	0.782
$S(p_{\text{opt},1:n}, 0.8)$	0.441	0.623	0.729	0.817	0.860	0.356	0.556	0.687	0.776	0.817
$S(p_{\text{opt},1:n}, 0.7)$	0.430	0.623	0.741	0.823	0.870	0.345	0.561	0.702	0.787	0.825
$S(p_{\text{opt},1:n}, 0.6)$	0.414	0.613	0.737	0.829	0.869	0.326	0.559	0.701	0.790	0.835
$S(p_{\text{opt},1:n}, 0.5)$	0.401	0.618	0.733	0.823	0.871	0.313	0.539	0.701	0.782	0.832
$S(p_{\text{opt},1:n}, 0.4)$	0.378	0.598	0.724	0.813	0.870	0.297	0.530	0.690	0.768	0.828
$S(p_{\text{opt},1:n}, 0.3)$	0.348	0.573	0.705	0.806	0.864	0.285	0.503	0.668	0.756	0.809
$S(p_{\text{opt},1:n}, 0.2)$	0.329	0.554	0.675	0.778	0.847	0.274	0.449	0.626	0.720	0.778
$S(p_{\text{opt},1:n}, 0.1)$	0.311	0.508	0.629	0.740	0.805	0.260	0.388	0.561	0.666	0.731

D EXPLORE QUALITY OF INSTRUCTION SET

In this section, we study the impact of distances between the instruction set and the true prompt on the detection power, focusing on the no-attack setting introduced in Section 5.1. Detection power with respect to the distance $d(\mathcal{P}_{\text{opt}}, y_{-n_0:0})$ is presented in Table D.1. We observe that as the distance decreases, the power increases. Therefore, a better-estimated prompt results in higher power for the proposed method.

Table C.2: Detection power using different λ under deletion attack.

Model	Meta-Llama-3-8B					Mistral-7B-v0.1				
Text length	10	20	30	40	50	10	20	30	40	50
baseline	0.604	0.676	0.705	0.715	0.701	0.475	0.618	0.625	0.658	0.660
$S(p_{\emptyset,1:n}, 1.0)$	0.291	0.261	0.237	0.206	0.168	0.335	0.337	0.287	0.277	0.264
$S(p_{\emptyset,1:n}, 0.9)$	0.658	0.720	0.700	0.707	0.684	0.591	0.697	0.675	0.675	0.650
$S(p_{\emptyset,1:n}, 0.8)$	0.665	0.733	0.712	0.733	0.708	0.594	0.713	0.685	0.705	0.675
$S(p_{\emptyset,1:n}, 0.7)$	0.666	0.738	0.719	0.739	0.722	0.590	0.716	0.694	0.709	0.691
$S(p_{\emptyset,1:n}, 0.6)$	0.659	0.741	0.724	0.739	0.729	0.587	0.720	0.695	0.714	0.701
$S(p_{\emptyset,1:n}, 0.5)$	0.660	0.740	0.728	0.744	0.728	0.579	0.714	0.698	0.713	0.702
$S(p_{\emptyset,1:n}, 0.4)$	0.649	0.735	0.730	0.744	0.728	0.566	0.707	0.698	0.710	0.707
$S(p_{\emptyset,1:n}, 0.3)$	0.644	0.725	0.730	0.743	0.726	0.561	0.697	0.695	0.708	0.702
$S(p_{\emptyset,1:n}, 0.2)$	0.635	0.717	0.727	0.742	0.727	0.535	0.685	0.694	0.702	0.699
$S(p_{\emptyset,1:n}, 0.1)$	0.625	0.702	0.720	0.735	0.720	0.507	0.663	0.670	0.689	0.683
$S(p_{\text{opt},1:n}, 1.0)$	0.771	0.719	0.669	0.606	0.563	0.601	0.578	0.495	0.482	0.422
$S(p_{\text{opt},1:n}, 0.9)$	0.793	0.763	0.734	0.753	0.740	0.694	0.741	0.703	0.678	0.662
$S(p_{\text{opt},1:n}, 0.8)$	0.790	0.771	0.744	0.755	0.743	0.695	0.758	0.714	0.703	0.682
$S(p_{\text{opt},1:n}, 0.7)$	0.787	0.774	0.745	0.757	0.741	0.682	0.761	0.719	0.715	0.698
$S(p_{\text{opt},1:n}, 0.6)$	0.772	0.774	0.750	0.760	0.742	0.672	0.767	0.723	0.719	0.706
$S(p_{\text{opt},1:n}, 0.5)$	0.764	0.768	0.751	0.761	0.745	0.659	0.763	0.722	0.719	0.709
$S(p_{\text{opt},1:n}, 0.4)$	0.744	0.769	0.751	0.762	0.746	0.638	0.754	0.716	0.725	0.708
$S(p_{\text{opt},1:n}, 0.3)$	0.717	0.761	0.750	0.753	0.744	0.604	0.739	0.716	0.722	0.710
$S(p_{\text{opt},1:n}, 0.2)$	0.688	0.749	0.741	0.751	0.732	0.572	0.714	0.705	0.709	0.700
$S(p_{\text{opt},1:n}, 0.1)$	0.638	0.717	0.726	0.739	0.724	0.525	0.677	0.682	0.695	0.684

Table C.3: Detection power using different λ under insertion attack.

Model	`Meta-Llama-3-8B`					`Mistral-7B-v0.1`				
Text length	10	20	30	40	50	10	20	30	40	50
`baseline`	0.519	0.618	0.650	0.650	0.673	0.448	0.556	0.618	0.645	0.625
$S(p_{\emptyset,1:n}, 1.0)$	0.249	0.219	0.205	0.183	0.189	0.285	0.277	0.260	0.249	0.218
$S(p_{\emptyset,1:n}, 0.9)$	0.614	0.679	0.703	0.691	0.727	0.523	0.661	0.680	0.724	0.688
$S(p_{\emptyset,1:n}, 0.8)$	0.610	0.681	0.711	0.699	0.737	0.522	0.673	0.698	0.729	0.697
$S(p_{\emptyset,1:n}, 0.7)$	0.612	0.684	0.711	0.706	0.740	0.517	0.678	0.695	0.727	0.697
$S(p_{\emptyset,1:n}, 0.6)$	0.602	0.683	0.709	0.702	0.741	0.523	0.679	0.700	0.723	0.699
$S(p_{\emptyset,1:n}, 0.5)$	0.595	0.684	0.704	0.701	0.735	0.524	0.671	0.699	0.720	0.699
$S(p_{\emptyset,1:n}, 0.4)$	0.587	0.672	0.701	0.700	0.730	0.514	0.663	0.689	0.719	0.692
$S(p_{\emptyset,1:n}, 0.3)$	0.573	0.671	0.687	0.688	0.725	0.506	0.650	0.685	0.717	0.688
$S(p_{\emptyset,1:n}, 0.2)$	0.559	0.662	0.679	0.680	0.714	0.491	0.629	0.671	0.713	0.672
$S(p_{\emptyset,1:n}, 0.1)$	0.544	0.646	0.663	0.666	0.696	0.480	0.600	0.656	0.693	0.652
$S(p_{\text{opt},1:n}, 1.0)$	0.715	0.669	0.585	0.555	0.504	0.582	0.542	0.490	0.422	0.335
$S(p_{\text{opt},1:n}, 0.9)$	0.731	0.728	0.732	0.723	0.764	0.677	0.699	0.720	0.752	0.721
$S(p_{\text{opt},1:n}, 0.8)$	0.729	0.729	0.738	0.721	0.772	0.677	0.720	0.723	0.754	0.723
$S(p_{\text{opt},1:n}, 0.7)$	0.716	0.730	0.735	0.726	0.771	0.665	0.727	0.726	0.755	0.726
$S(p_{\text{opt},1:n}, 0.6)$	0.705	0.729	0.737	0.722	0.762	0.653	0.722	0.729	0.755	0.724
$S(p_{\text{opt},1:n}, 0.5)$	0.686	0.725	0.734	0.720	0.756	0.632	0.718	0.730	0.749	0.725
$S(p_{\text{opt},1:n}, 0.4)$	0.674	0.712	0.728	0.715	0.745	0.616	0.710	0.722	0.740	0.711
$S(p_{\text{opt},1:n}, 0.3)$	0.654	0.699	0.711	0.708	0.736	0.586	0.694	0.712	0.735	0.709
$S(p_{\text{opt},1:n}, 0.2)$	0.609	0.679	0.697	0.695	0.726	0.546	0.671	0.698	0.724	0.677
$S(p_{\text{opt},1:n}, 0.1)$	0.572	0.658	0.674	0.673	0.703	0.502	0.624	0.661	0.697	0.654

Table D.1: Average detection power as a function of the distance between the instruction set and the true prompt with token length $m = 10$ and 100 replications.

$d(\mathcal{P}_{\text{opt}}, y_{-n_0:0})$	50	40	30	20	10	1
`Meta-Llama-3-8B`	0.65	0.66	0.69	0.70	0.79	0.84
`Mistral-7B-v0.1`	0.56	0.57	0.59	0.65	0.72	0.73