Improved Unbiased Watermark for Large Language Models

Anonymous ACL submission

Abstract

As artificial intelligence surpasses human capabilities in text generation, the necessity to authenticate the origins of AI-generated content has become paramount. Unbiased watermarks offer a powerful solution by embedding statistical signals into language model-generated text without distorting the quality. In this paper, we introduce MCMARK, a family of unbiased, **M**ulti-**C**hannel-based watermarks. MCMARK works by partitioning the model's vocabulary into segments and promoting token probabilities within a selected segment based on a watermark key. We demonstrate that MCMARK not only preserves the original distribution of the language model but also offers significant improvements in detectability and robustness over existing unbiased watermarks. Our experiments with widely-used language models demonstrate an improvement in detectability of over 10% using MCMARK, compared to existing state-of-the-art unbiased watermarks. This advancement underscores MCMARK's potential in enhancing the practical application of watermarking in AI-generated texts.

1 Introduction

As artificial intelligence outstrips human ability in text generation, verifying the authenticity and source of AI-created texts is increasingly crucial. Watermarking of language models (Aaronson, 2022; Kirchenbauer et al., 2023a; Christ et al., 2023; Kuditipudi et al., 2023; Hu et al., 2023; Wu et al., 2023) offers an effective method to differentiate between texts generated by humans and machines. This approach involves covertly embedding a statistical signal within the text via a watermark generator that uses specific watermark keys. Detection of this statistical signal is carried out using statistical hypothesis testing, enabling the confirmation of the text's provenance.

Watermarks that do not introduce distortion (Aaronson, 2022; Christ et al., 2023; Kuditipudi et al., 2023; Hu et al., 2023; Wu et al., 2023) are particularly crucial in the watermarking of language models. These watermarks are essential as they are provably capable of maintaining the original output distribution of the language model. The expected distribution of the watermarked model, conditioned on the watermark keys, aligns perfectly with that of the original language model, thus maintaining utility and relevance in practical applications.

However, current unbiased watermarking approaches face practical challenges. For instance, the unbiased watermark (Hu et al., 2023) requires access to language model (LM) prompts and APIs, EXP-edit (Kuditipudi et al., 2023) incurs a substantial time cost for detection, and DiPmark (Wu et al., 2023) exhibits lower detection accuracy compared to biased watermarks like those reported in (Kirchenbauer et al., 2023a). Consequently, enhancing the practicality of watermarks is an imperative issue. In our work, we introduce a novel family of unbiased watermarks, termed MCMARK, which exhibit enhanced detectability and robustness. In MCMARK, we partition the vocabulary into l segments. During the watermark generation process, a watermark key is used to randomly select a segment. Then, the token probabilities within the selected segment are promoted using our MCMARK-based unbiased algorithm. During detection, the presence of the watermark is detected by verifying whether the current token corresponds to the segment associated with the watermark key.

Our contribution can be summarized as follows:

- We introduced MCMARK, a family of unbiased watermarks that provably preserve the output distribution of language models. MCMARK is adaptable, robust to text modification, and does not require access to prompt and language model APIs during detection.

- We theoretically demonstrate that when the number of segments equals two, MCMARK

offers superior detectability compared to DiPmark(Wu et al., 2023) and STA-1 (Mao et al., 2024). We further discuss the trade-offs between detectability and robustness in MCMARK.

- Through comprehensive experiments, we validate the unbiasedness, detectability, and robustness of MCMARK on popular language models, such as LLAMA-3. Our results show an over 10% improvement in detectability compared to the state-of-the-art unbiased watermarks.

2 Related Work

Statistical watermarks. Kirchenbauer et al. (2023a) refined the statistical watermarking framework initially introduced by Aaronson (2022), showcasing the efficacy of this technique via comprehensive experiments on large language models. They divided the language model tokens into red and green lists and favored the green list tokens by adjusting their logits with a fixed increment δ. Zhao et al. (2023) introduced a unigram watermark approach that employs single-gram hashing to generate watermark keys, enhancing the robustness of statistical watermarks. Liu et al. (2023b) further increased the robustness of statistical watermarking by using the semantics of generated texts as watermark keys. Additionally, Liu et al. (2023a) developed a scheme for unforgeable watermarks that utilizes neural networks to alter token distributions, moving away from conventional watermark keys. Nevertheless, such methods can substantially alter the text distribution, potentially diminishing the quality of the content.

Unbiased watermarks. To maintain the original output distribution in watermarked content, several researchers have investigated novel approaches for token distribution modification. Aaronson (2022) pioneered an unbiased watermarking method using Gumbel-max to adjust token distribution and employing prefix n-grams as watermark keys. Christ et al. (2023) used inverse sampling for modifying the token distributions of watermarked content on a binary language model with watermark keys based on token positioning. ITS-edit and EXP-edit (Kuditipudi et al., 2023) utilized inverse-sampling and Gumbel-max respectively for modifying the token distributions of watermarked content, with a predetermined watermark key list. Hu et al. (2023) combined inverse-sampling and γ-reweight strategies for watermarking, though their detection method is not model-agnostic. DiPmark (Wu et al., 2023) enhanced the γ-reweight technique and introduced a model-agnostic detector. STA-1 (Mao et al., 2024) optimized the quality of the watermarked text under the low-entropy scenarios.

3 Preliminary

Notation. We follow the notations in the previous work (Hu et al., 2023; Wu et al., 2023; Mao et al., 2024) to represent the generation task of LLMs. Let the set of vocabulary tokens be denoted by V with cardinality $N = |V|$. We define $\mathcal{V}$, which includes all possible token sequences of any length including zero. In the context of a language model, token sequences are generated in response to a specific prompt. The probability of generating the next token x_{t+1} from V, conditioned on the preceding token sequence $x_1, \ldots, x_t$, is represented as $P_M(x_{t+1} \mid \boldsymbol{x}_{1:t})$.

In the watermark generator, the LLM utilizes a private key $k \in \mathcal{K}$ to reweight the distribution from $P_M(x_{t+1} \mid \boldsymbol{x}_{1:t})$ to $P_{M,w}(x_{t+1} \mid \boldsymbol{x}_{1:t}, k)$, where $P_{M,w}$ indicates a watermarked model and the private key k is randomly selected from a key space $\mathcal{K}$ according to a known distribution $P_{\mathcal{K}}(k)$. According to (Hu et al., 2023), an unbiased watermark requires that the expectation of the reweighted distribution equals that of the original distribution, i.e.,

$$\mathbb{E}_{k\sim P_{\mathcal{K}}}[P_{M,w}(x_{t+1} \mid \boldsymbol{x}_{1:t}, k)] = P_M(x_{t+1} \mid \boldsymbol{x}_{1:t}).$$

During watermark detection, the user only has access to the watermark key, the reweight strategy, and the generated audio. The detector employs a hypothesis testing approach to ascertain the presence of the watermark signal. The null hypothesis H_0 is defined as *"The content is generated without the presence of watermarks"*. The detector adopts a score function based on the watermark key and the reweight strategy, which exhibits statistical bias between the watermarked and unwatermarked token sequences.

4 Method

Definition 4.1 (Distribution Channel)**.** *Given the original LM distribution $P_M(\cdot|x_{1:t})$, the reweighting method and the key space, we define each unique watermarked distribution $P_{M,w}(\cdot|x_{1:t}, k)$ as a distribution channel. The set of distribution*

channels is the set of all possible LM distributions after watermarking, i.e., $\{P_{M,w}(\cdot|x_{1:t}, k)|k \in \mathcal{K}\}$.

For example, for the inverse-sampling and the Gumbel-max method, the set of distribution channels is $\{\delta(x)|x \in V\}$, where δ is the Dirac distribution, $\delta(x)$ means the probability of sampling token x is 1. It is easy to see the cardinality of the distribution channel set is $|V|$.

We can define the unbiased watermark from the perspective of distribution channels. Let $\{P_1, \ldots, P_l\}$ be a set of distribution channels. Define $\mathcal{K}_i$ as the subset of watermark keys that satisfy $P_{M,w}(\cdot \mid x_{1:t}, k) = P_i$ for all $k \in \mathcal{K}_i$. An unbiased watermark should meet the following condition: $\forall x \in V$, it should have:

$$\sum_{i=1}^{l} P_i(x)\mathbb{E}_{k\sim P_{\mathcal{K}}}\left[\mathbf{1}_{\{k\in\mathcal{K}_i\}}\right] = P_M(x \mid x_{1:t}).$$

4.1 MCMARK Overview

By utilizing the distribution channels, we can design a new watermarking framework. During generation, the watermark key is used to pseudorandomly select one of these channels as the basis for the next token distribution. For detection, the algorithm verifies whether a given token was generated by our model by assessing whether the token aligns with the recovered distribution channel i_t. The detailed algorithms for the generation and detection processes are provided in Alg. 3 and Alg. 4.

Based on this framework, we propose MCMARK, a family of **M**ulti-**C**hannel-based unbiased watermarking algorithms. MCMARK operates by constructing a set of l distribution channels. We divide the vocabulary V into l equal parts, $V_1, \ldots, V_l$. For each distribution channel P_i, we increase the probability of tokens in V_i. During detection, given the recovered distribution channel index i_t, if the generated token $x_t \in V_{i_t}$, we assume this token is generated by the watermarked distribution. The detailed algorithms for the generation and detection processes are shown in Alg. 1 and Alg. 2.

The effectiveness of our proposed watermarking approach hinges on designing optimal distribution channels that maximize $\mathbb{E}_{i_t\in\{1,2,\cdots,l\}}\sum_{x\in V_{i_t}} P_{i_t}(x)$, that is if the selected channel is i_t, the probability of generating tokens within V_{i_t} should be maximized to ensure that more tokens can be effectively detected.

In the following sections, we detail the process for obtaining the optimal distribution channels.

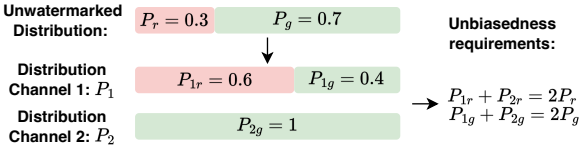


Figure 1: Illustration of the optimal solution for the binary channel example, where we set the red and green list token probabilities to $P_r = 0.3$ and $P_g = 0.7$ respectively.

4.2 Finding Unbiased Watermarks as an Optimization Problem

A binary channel example. Using the concept of distribution channels, we can explore new unbiased watermarks by identifying sets $\{P_1, \ldots, P_l\}$ and $\{\mathcal{K}_1, \ldots, \mathcal{K}_l\}$. Consider the simplest case where $l = 2$ and $\mathbb{E}_{k\sim P_{\mathcal{K}}}[\mathbf{1}_{k\in\mathcal{K}_1}] = \mathbb{E}_{k\sim P_{\mathcal{K}}}[\mathbf{1}_{k\in\mathcal{K}_2}] = \frac{1}{2}$. In this scenario, the watermark possesses two distribution channels with equal probability. To ensure detectability of the watermark (in the presence of a watermark key), the distributions P_1 and P_2 must be sufficiently distinct. A practical approach involves dividing all tokens into two lists, red and green. In distribution channel P_1, we increase the sum of the probability of tokens in the red list ($P_{1r} := \sum_{x\in V_r} P_1(x)$), and in distribution channel P_2, we increase the probability of tokens in the green list (P_{2g}). Let P_r and P_g denote the cumulative probabilities of the red and green tokens in the original language model distribution, respectively. The probabilities $P_{1r}, P_{1g}, P_{2r}, P_{2g}$ must satisfy the following constraints for maintaining unbiased properties:

$$\begin{cases} P_{1r} + P_{1g} = 1, \\ P_{2r} + P_{2g} = 1, \\ P_{1r} + P_{2r} = 2P_r, \\ P_{1g} + P_{2g} = 2P_g. \end{cases} \quad (1)$$

The first two constraints ensure that the sum of probabilities within a distribution equals 1, while the last two constraints are necessary to uphold the unbiased nature of the watermark.

In order to maximize the detection efficiency, we expect the variation between P_1 and P_2 to be as large as possible. Thus our optimization objective is

$$\max P_{1r} + P_{2g}.$$

Assuming w.l.o.g. $P_r \leq 0.5$, it is easy to identify the optimal solution is $P_{1r} = 2P_r$ and $P_{2g} = 1$ with the constrains in Eq. 1. See Figure 1 for an illustration of the optimal solution. In this configuration, the probabilities of all red tokens are

doubled in the first distribution channel, while the probabilities of all green tokens are doubled in both channels.

Optimization problem. We can generalize the binary case ($l = 2$) to more complex scenarios. Consider $\mathbb{E}_{k\sim P_\mathcal{K}}[\mathbf{1}_{k\in\mathcal{K}_1}] = \cdots = \mathbb{E}_{k\sim P_\mathcal{K}}[\mathbf{1}_{k\in\mathcal{K}_l}] = \frac{1}{l}$. Similarly to the $l = 2$ case, we can divide the token set into l equal parts, $V_1, \ldots, V_l$, where $|V_i| = \frac{|V|}{l}$. For each distribution channel P_i, we increase the probability of tokens in V_i. Denoting by $P_{i,V_j} := \sum_{x\in V_j} P_i(x)$ the probability of part V_j in channel P_i, and P_{V_j} the sum of token probabilities within V_j (i.e. $\mathbb{E}_{x\sim P_M}[\mathbf{1}_{x\in V_j}]$), we formulate the following optimization problem:

$$\max \sum_{i=1}^{l} P_{i,V_i}, \quad \text{s.t.} \begin{cases} \sum_{j=1}^{l} P_{i,V_j} = 1, & \forall i = 1, \ldots, l, \\ \sum_{i=1}^{l} P_{i,V_j} = lP_{V_j}, & \forall j = 1, \ldots, l. \end{cases} \tag{2}$$

The constraint $\sum_{j=1}^{l} P_{i,V_j} = 1$ ensures that the total probability within each distribution channel P_i sums to 1. This requirement guarantees that each P_i represents a valid probability distribution. Additionally, the constraint $\sum_{i=1}^{l} P_{i,V_j} = lP_{V_j}$ ensures that the expected probability of V_j across all distribution channels equals the original probability of V_j in the model distribution P_M. This maintains the unbiased nature of the watermark.

Optimization Solution. Given that $\max \sum_{i=1}^{l} P_{i,V_i} \leq \sum_{i=1}^{l} \max P_{i,V_i}$, we first calculate each $\max P_{i,V_i}$ individually and then demonstrate the feasibility of $\sum_{i=1}^{l} \max P_{i,V_i}$. With the constraint $\sum_{i=1}^{l} P_{i,V_j} = lP_{V_j}$, we have $P_{i,V_i} \leq \min\{1, lP_{V_i}\}$ and $\max P_{i,V_i} = \min\{1, lP_{V_i}\}$. Therefore, we obtain:

$$\max \sum_{i=1}^{l} P_{i,V_i} \leq \sum_{i=1}^{l} \min\{1, lP_{V_i}\}.$$

We now show that $\sum_{i=1}^{l} \min\{1, lP_{V_i}\}$ is feasible. We propose one solution:

$$P_{i,V_j} = \begin{cases} \min\{1, lP_{V_i}\}, & \text{if } i = j, \\ (1 - lP_{V_i})_+ (lP_{V_j} - 1)_+, & \text{if } i \neq j. \end{cases} \tag{3}$$

where $(1 - lP_{V_i})_+ := \max\{0, 1 - lP_{V_i}\}$ and $(lP_{V_j} - 1)_+ := \max\{0, lP_{V_j} - 1\}$. Please refer

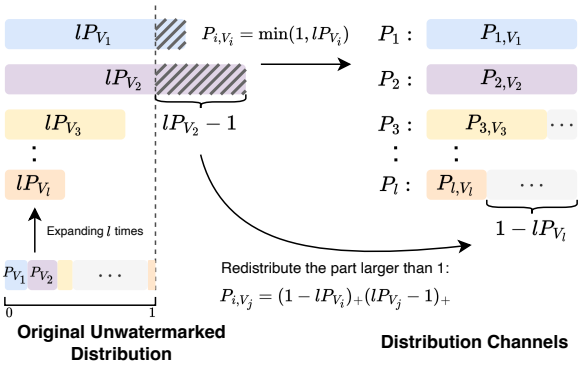


Figure 2: Illustration of the optimal solution (Eq. 3) for creating the distribution channels for MCMARK.

Figure 2 for an illustration of the optimal solution. The token probabilities within a channel are given by $P_i(x) = \frac{P_{i,V_j}}{P_{V_j}} P_M(x|\boldsymbol{x}_{1:t}), \forall x \in V_j$.

4.3 Watermarking Methodology

With the optimization solution defined, we can establish our watermarking algorithm. Following the approach outlined in (Kirchenbauer et al., 2023a), we utilize a fixed secret key sk and the n-gram preceding content $x_{t-n:t-1}$ as the watermark key to generate the token x_t. During watermark generation, we first split the vocabulary into l parts and compute the set of distribution channels according to Eq. 3. We then pseudorandomly select a probability channel based on the watermark key and sample the next token from the selected probability channel. The detailed algorithm is presented in Algorithm 1.

During detection, we are given the generated content and the secret key. With this information, we can recover the index i_t of the probability channel used for generating the token x_t at step t. As in P_{i_t}, we increase the probability of tokens in V_{i_t}, thus we can detect the watermark signal by checking whether x_t is in V_{i_t} or not. We define a hypothesis test with null hypothesis H_0: the content is generated without watermarking. Denoted by $\boldsymbol{x}_{1:T}$ a given content sequence, we can use the test statistic $\Phi(\boldsymbol{x}_{1:T}) = \sum_{t=1}^{T} \mathbf{1}_{x_t \in V_{i_t}}$, where i_t is the index of the probability channel at step t recovered from the watermark keys.

Under the null hypothesis, $\Phi(\boldsymbol{x}_{1:T})$ follows a binomial distribution with a success rate of $\frac{1}{l}$. Thus, we have the following tail bound:

$$\Pr(\Phi(\boldsymbol{x}_{1:T}) \geq z) = \sum_{i=\lceil z \rceil}^{T} \binom{T}{i} \left(\frac{1}{l}\right)^i \left(\frac{l-1}{l}\right)^{T-i}$$

The theoretical false positive rate is therefore:

$$\sum_{i=\Phi(\boldsymbol{x}_{1:T})}^{T} \binom{T}{i} \left(\frac{1}{l}\right)^{i} \left(\frac{l-1}{l}\right)^{T-i} \tag{4}$$

Algorithm 1 MCMARK generator.

Input: pretrained LM P_M, secret key sk, prompt $\boldsymbol{x}_{-m:0}$, generate length $T \in \mathbb{N}$.
for $t = 1, \ldots, T$ **do**
  Split the vocabulary into l parts $\{V_1, ..., V_l\}$.
  Get the probability distribution of t-th token $P_M(\cdot \mid \boldsymbol{x}_{-m:t-1})$.
  Get $P_{V_j} = \sum_{x \in V_j} P_M(x \mid \boldsymbol{x}_{-m:t-1})$, $j = 1, ..., l$
  Generate the distribution channels $\{P_1, ..., P_l\}$ by Eq. 3
  Pseudorandomly select a probability channel P_i based on the watermark key $(\mathsf{sk}, \boldsymbol{x}_{t-n:t-1})$.
  Sample the next token x_i from P_i.
return $\boldsymbol{x}_{1:T}$.

Algorithm 2 MCMARK detector.

Input: pretrained LM P_M, generated tokens $\boldsymbol{x}_{1:T}$, false positive rate threshold p_0.
Initialize $\Phi = 0$
for $t = 1, \ldots, T$ **do**
  Recover the index of the selected distribution channel i_t based on the watermark key.
  $\Phi = \Phi + \mathbf{1}_{x_t \in V_{i_t}}$.
Get theoretical false positive rate p by Eq. 4.
if $p \leq p_0$ **then**
  return $\boldsymbol{x}_{1:T}$ is watermarked.
else
  return $\boldsymbol{x}_{1:T}$ is not watermarked.

4.4 Theoretical analysis

During the watermark detection process, situations may arise where the current token x_t is sampled from the distribution channel P_{i_t} but does not belong to the designated subset V_{i_t}. This discrepancy occurs when $\mathbb{E}_{x \sim P_M}[\mathbf{1}_{x \in V_{i_t}}] < \frac{1}{l}$. Under such conditions, detection of the watermark signal is not possible even though x_t is generated from the watermarked distribution. This leads us to define the true-negative rate, which quantifies the frequency of these undetectable instances, as follows:

Definition 4.2 (Expected True-Negative Rate)**.** *We define the true-negative rate, denoted as $P_{TN}(x, P_{M,w}(\cdot \mid k), \Phi)$, as the probability that a token is generated from the watermarked distribution (true) but cannot be detected by the watermark detector (negative): $P_{TN}(x, P_{M,w}(\cdot \mid k), \Phi) := \Pr(\{x \sim P_{M,w}(\cdot \mid k), \Phi(x) = 0\})$. The expected true-negative rate is then defined as*

$$E_{TN} := \mathbb{E}_{k \sim P_{\mathcal{K}}}[P_{TN}(x, P_{M,w}(\cdot \mid k), \Phi)].$$

The true-negative rate is relevant for many statistical watermarks, including Soft watermark (Kirchenbauer et al., 2023a), γ-reweight (Hu et al., 2023), DiPmark (Wu et al., 2023), and STA-1 (Mao et al., 2024). Notably, γ-reweight is a special case of DiPmark. In the subsequent analysis, we will compare the expected true-negative rates of DiPmark, STA-1, and our proposed method.

Both DiPmark and STA-1 implement a red-green list strategy. We denote the red list probability as $P_{V_r} := \mathbb{E}_{x \sim P_M}[\mathbf{1}_{x \in V_r}]$. The expected true negative rate of DiPmark (E_{TN}^{DiP}), STA-1 (E_{TN}^{STA}) and MCMARK (E_{TN}^{MCmark}) are given by:

$$\begin{cases} E_{TN}^{\text{DiP}} = \max\{P_{V_r} - \alpha, 0\} + \max\{P_{V_r} - (1-\alpha), 0\}. \\ E_{TN}^{\text{STA}} = P_{V_r}^2. \\ E_{TN}^{MCmark} = \sum_{i=1}^{l} \max\{0, 1/l - P_{V_i}\}. \end{cases}$$

If the probabilities P_{V_i} are evenly distributed, that is, $P_{V_i} = \frac{1}{l}$, then the expected true-negative rate for MCMARK, E_{TN}^{MCmark}, equals zero, and watermark signals are embedded uniformly across all tokens. In practice, increasing l tends to result in more unevenly distributed P_{V_i}. In the most extreme case, when $l = |V|$, each segment V_i contains exactly one token, which represents the most uneven distribution of P_{V_i} and thus the expected true-negative rate will be large.

To compare with DiPmark and STA-1, we consider a special case of MCMARK where $l = 2$, meaning there are two distribution channels, and the vocabulary is segmented into a red and a green list. In this scenario, the expected true-negative rate for MCMARK simplifies to:

$$\begin{aligned} E_{TN}^{MCmark} &= \max\left\{0, \frac{1}{2} - P_{V_r}\right\} \\ &\quad + \max\left\{0, P_{V_r} - \frac{1}{2}\right\} = \left|\frac{1}{2} - P_{V_r}\right|. \end{aligned} \tag{5}$$

Note that all of E_{TN}^{DiP}, E_{TN}^{STA}, and E_{TN}^{MCmark} are correlated with P_{V_r}. Assuming a uniform distribution of P_{V_r} on $[0, 1]$, we compute $\mathbb{E}_{P_{V_r}}[E_{TN}^{\text{DiP}}]$,

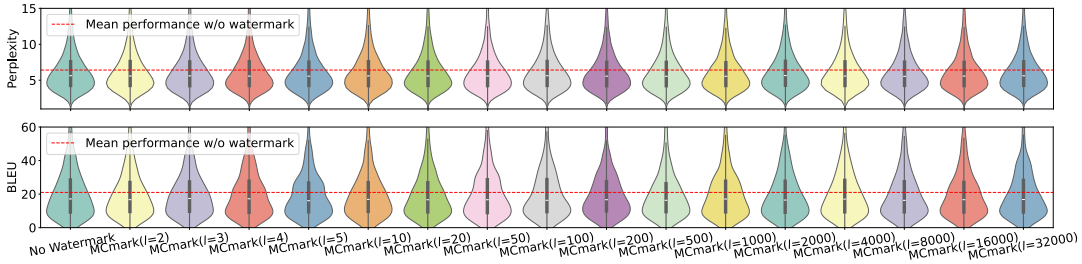


Figure 3: Validating unbiased property of MCMARK. **Top**: text summarization task with perplexity metric. **bottom**: machine translation task with BLEU metric.

Table 1: Detectability comparison with LLama2 and C4 dataset on text generation task. Notice, as the detectors of ITS-edit and EXP-edit do not provide a theoretical guarantee of the false positive rate, we report the true positive rate at the empirical false positive rate instead. '-' refers to unavailable results. The lack of results for ITS-edit and EXP-edit at an FPR of 0.01% is because, in their original setting, the lowest achievable FPR does not reach 0.01%.

	TPR@FPR=1%↑	TPR@FPR=0.1%↑	TPR@FPR=0.01%↑	Median p-value↓
KGW(δ=0.5)	38.78%	19.16%	10.20%	2.809e-2
KGW(δ=1.0)	86.88%	74.44%	61.55%	5.494e-6
KGW(δ=1.5)	96.69%	93.83%	90.08%	1.556e-12
KGW(δ=2.0)	99.34%	98.79%	97.79%	6.580e-22
Unigram(δ=0.5)	78.63%	63.51%	47.54%	1.607e-4
Unigram(δ=1.0)	96.99%	92.59%	88.08%	1.745e-9
Unigram(δ=1.5)	98.94%	97.54%	96.13%	1.051e-16
Unigram(δ=2.0)	99.88%	99.52%	98.93%	5.387e-25
ITS-edit	61.77%	54.49%	-	4.000e-4
EXP-edit	89.01%	86.35%	-	2.000e-4
γ-reweight	89.17%	81.79%	75.83%	4.467e-8
DiPmark(α=0.4)	87.66%	78.77%	71.77%	1.236e-7
DiPmark(α=0.3)	81.88%	69.88%	61.65%	5.284e-6
STA-1	84.93%	71.58%	57.76%	2.656e-5
MCMARK(l=20)	**98.96%**	**98.38%**	**97.69%**	**8.098e-30**

$\mathbb{E}_{P_{V_r}}[E_{TN}^{\text{STA}}]$, and $\mathbb{E}_{P_{V_r}}[E_{TN}^{MCmark}]$ as $(\alpha-\frac{1}{2})^2+\frac{1}{4}$, $\frac{1}{3}$, and $\frac{1}{4}$, respectively. This implies that MCMARK achieves superior detectability compared to DiPmark and STA-1, as indicated by the lower expected true negative rate. Furthermore, the variances of E_{TN}^{DiP}, E_{TN}^{STA}, and E_{TN}^{MCmark} are calculated as $\frac{5}{48}-(\alpha-\frac{1}{2})^2\left[(\alpha+\frac{1}{6})^2+\frac{1}{18}\right]$, $\frac{4}{45}$, and $\frac{1}{48}$, respectively. Given that in DiPmark $\alpha \leq \frac{1}{2}$, MCMARK also achieves the minimum variance among all three methods. This indicates that MCMARK can more consistently generate watermarked sentences with a low true negative rate.

4.5 Robustness-detectability trade-off

An adversary may attempt to alter the output token to disrupt the watermark detection. In MCMARK detection, if a token x_t is modified to x'_t and $x'_t \notin V_{i_t}$, the watermark signal is effectively removed. Consequently, the probability that a watermark is removed due to such an alteration is given by $\frac{|V_{i_t}|}{|V|}$. Given that $|V_{i_t}| = \frac{|V|}{l}$, the probability that a watermark is removed simplifies to $\frac{1}{l}$. Therefore, increasing l decreases the robustness of the watermark, as it increases the likelihood that an adversary can successfully remove the watermark by modifying the token.

On the other hand, we show that moderately increasing l can enhance the detectability of the watermark (see Appendix B for a detailed discussion). Thus, we identify a fundamental trade-off: increasing the number of distribution channels l enhances the detectability of the watermark, yet it simultaneously reduces its robustness. We empirically validate our analysis in Figure 5 and 6.

5 Experiments

The experiments consist of two main parts. First, we validate that our proposed method is unbiased

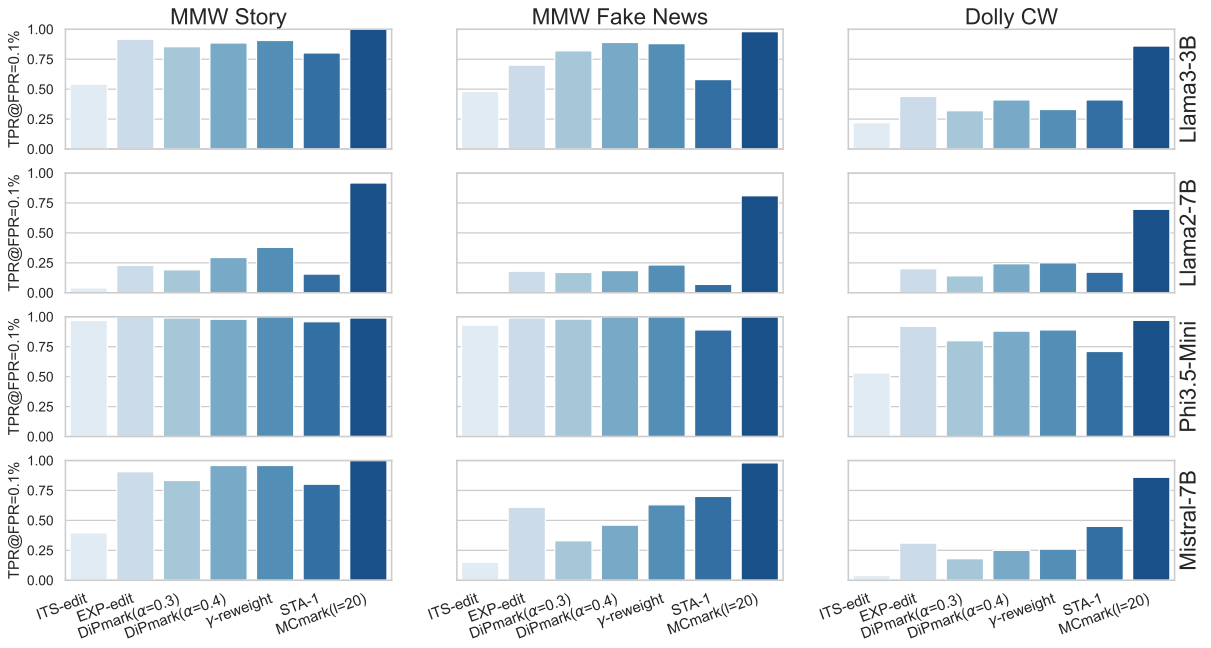


Figure 4: Comparative analysis of MCMARK against SOTA unbiased watermarks across various language models and datasets on watermark detectability.

by demonstrating that its output quality for machine translation and text summarization tasks is similar to the baseline without watermarking. Second, we showcase the effectiveness of MCMARKthrough comprehensive experiments on text generation. After that, we discuss the robustness of our method and the detectability-robustness trade-off in MCMARK. The detailed experimental settings can be found in Appendix C.

5.1 Unbiased Property Validation

Since MCMARK is **provably** unbiased, we use this empirical experiment as a support for this property. We follow the evaluation process of (Hu et al., 2023). Specifically for MCMARK, we select number of the distribution channels l from the set {2, 3, 4, 5, 10, 20, 50, 100, 200, 500, 1000, 2000, 4000, 8000, 16000, 32000}. Upon examining Figure 3, we find across all l values, the BLEU scores in the machine translation tasks and the perplexity values in the text summarization tasks remain consistently similar between MCMARK and the original language model. In appendix Table 3 and Table 4, we provide additional unbiasedness evaluation on both biased and unbiased watermarks, the results indicate that MCMARK can preserve the LM distribution compared to the biased watermarks.

5.2 Detectability

Following the evaluation metric of the previous works (Kuditipudi et al., 2023; Wu et al., 2023), we report the true positive rate at guaranteed false positive rates, i.e., TPR@FPR=$\{1\%, 0.1\%, 0.01\%\}$. Notice, as the detectors of ITS-edit and EXP-edit do not provide a theoretical guarantee, we report the true positive rate at the empirical false positive rate following their original setting. From Table 1, we see that MCMARK achieved the best detectability comparing with all unbiased watermarks, at least 14% improvement on all TPR@FPR metrics. Besides, MCMARK outperformed the biased watermarking algorithm KGW and Unigram when $\delta \in \{0.5, 1.0, 1.5\}$, and achieved comparable performance with them when $\delta = 2.0$. In Figure 4, we present a comparative analysis of MCMARK against SOTA unbiased watermarks across various language models and datasets. Our method, represented by the last bar in each plot, consistently outperforms all comparisons across the board. Further experimental results are detailed in Figure 7.

Time Efficiency. Similar to the KGW watermark, Unigram, and DiPmark, the time cost introduced by the MCMARK generator occurs only during the modification of token probabilities in the generation process. Additionally, the MCMARK detector is model-agnostic. Empirically, the time efficiency of the MCMARK method matches that of KGW, Unigram, γ-reweight, DiPmark, and STA-1. Notably, the detectors of ITS-edit and EXP-edit require thousands of inferences (Wu et al., 2023), which significantly reduces their detection efficiency compared

Table 2: Robustness comparison of unbiased watermarks, we use metrics TPR@FPR=0.1% and the median p-value.

	ϵ=0.05		ϵ=0.1		ϵ=0.2	
	TPR↑	p-value↓	TPR↑	p-value↓	TPR↑	p-value↓
ITS-edit	50.40%	7.998e-4	43.69%	6.399e-3	33.90%	4.839e-2
EXP-edit	81.35%	2.000e-4	78.27%	2.000e-4	74.88%	2.000e-4
γ-reweight	72.38%	4.241e-6	59.40%	1.975e-4	31.07%	1.195e-2
DiPmark(α=0.4)	69.63%	8.154e-6	58.13%	1.975e-4	29.06%	1.996e-2
DiPmark(α=0.3)	59.53%	1.363e-4	46.24%	2.123e-3	20.59%	4.059e-2
STA-1	60.84%	2.253e-4	47.15%	1.462e-3	21.35%	1.843e-2
MCMARK(l=20)	**97.11%**	**6.068e-23**	**96.07%**	**2.140e-18**	**88.79%**	**3.731e-10**

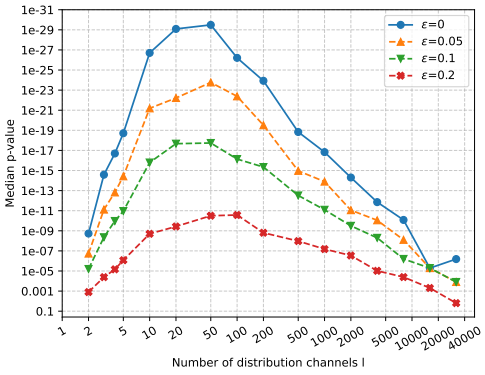


Figure 5: Median p-value vs number of distribution channels l in MCMARK.

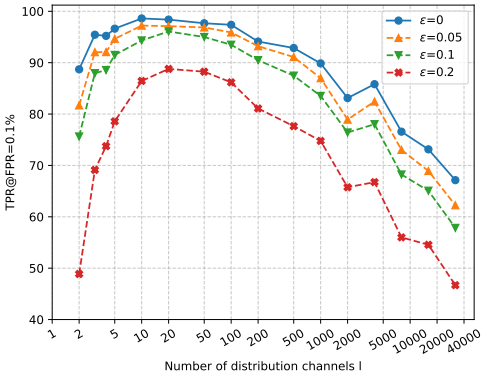


Figure 6: TPR@FPR=0.1% vs number of distribution channels l in MCMARK.

to other methods.

5.3 Robustness

We compare the robustness of MCMARK ($l = 20$) with the SOTA unbiased watermark approaches ITS-edit, EXP-edit, γ-reweight, DiPmark, and STA-1. In this context, we use the text generation task with 1,000 generated sequences on LLaMA-2. For robustness evaluation, we manipulate $\epsilon \in \{0.05, 0.1, 0.2\}$ portion of the text tokens through token replacement attack (Kirchenbauer et al., 2023b; Wu et al., 2023; Chen et al., 2024).

In Table 2, we present the TPR at an FPR of 0.1% and the median p-value for various watermarks across different attack strengths, denoted by ϵ. MCMARK consistently demonstrates superior robustness, outperforming all baseline methods in effectively detecting watermarked sentences.

5.4 Detectability-robustness trade-off

In Section 4.5, we discuss the detectability-robustness trade-off of MCMARK w.r.t. l. In this experiment, we empirically verify this trade-off by comparing the TPR@FPR=0.1% and the median p-value with the number of the distribution channels $l \in \{2, 3, 4, 5, 10, 20, 50, 100, 200, 500, 1000, 2000, 4000, 8000, 16000, 32000\}$. We use Llama2 model with C4 subset on the text generation task. For robustness evaluation, we modify $\epsilon \in \{0.05, 0.1, 0.2\}$ portion of the text tokens through token replacement attack (Kirchenbauer et al., 2023b; Wu et al., 2023; Chen et al., 2024).

In Figures 5 and 6, we report on the detectability of MCMARK using two metrics: TPR@FPR=0.1% and median p-value. The robustness of MCMARK initially increases with l before subsequently decreasing, aligning with the analysis presented in Section 4.5.

6 Conclusion

In summary, we present MCMARK, a novel family of unbiased watermarks that significantly enhance detectability and robustness in large language models without distorting text output. Our experimental results demonstrate a more than 10% improvement in detectability over existing state-of-the-art unbiased watermarking approaches, validated across various models and datasets. MCMARK represents a significant advancement in the practical application of watermarking technology in LMs.

Limitations

There is an inherent trade-off between the robustness and detectability of our proposed MCMARK. Increasing the number of distribution channels l can potentially enhance detectability but reduce robustness against attacks where an adversary may modify output tokens to disrupt the watermark detection.

References

Scott Aaronson. 2022. My AI safety lecture for UT effective altruism,.

Marah Abdin, Jyoti Aneja, Hany Awadalla, Ahmed Awadallah, Ammar Ahmad Awan, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Jianmin Bao, Harkirat Behl, et al. 2024. Phi-3 technical report: A highly capable language model locally on your phone. *arXiv preprint arXiv:2404.14219*.

Ond rej Bojar, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Matthias Huck, Antonio Jimeno Yepes, Philipp Koehn, Varvara Logacheva, Christof Monz, Matteo Negri, Aurelie Neveol, Mariana Neves, Martin Popel, Matt Post, Raphael Rubino, Carolina Scarton, Lucia Specia, Marco Turchi, Karin Verspoor, and Marcos Zampieri. 2016. Findings of the 2016 conference on machine translation. In *Proceedings of the First Conference on Machine Translation*, pages 131–198, Berlin, Germany. Association for Computational Linguistics.

Ruibo Chen, Yihan Wu, Yanshuo Chen, Chenxi Liu, Junfeng Guo, and Heng Huang. 2024. A watermark for order-agnostic language models. *arXiv preprint arXiv:2410.13805*.

Miranda Christ, Sam Gunn, and Or Zamir. 2023. Undetectable watermarks for language models. *arXiv preprint arXiv:2306.09194*.

Mike Conover, Matt Hayes, Ankit Mathur, Jianwei Xie, Jun Wan, Sam Shah, Ali Ghodsi, Patrick Wendell, Matei Zaharia, and Reynold Xin. 2023. Free dolly: Introducing the world's first truly open instruction-tuned llm.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Zhengmian Hu, Lichang Chen, Xidong Wu, Yihan Wu, Hongyang Zhang, and Heng Huang. 2023. Unbiased watermark for large language models. *arXiv preprint arXiv:2310.10669*.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

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

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

Rohith Kuditipudi, John Thickstun, Tatsunori Hashimoto, and Percy Liang. 2023. Robust distortion-free watermarks for language models. *arXiv preprint arXiv:2307.15593*.

Mike Lewis. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.

Aiwei Liu, Leyi Pan, Xuming Hu, Shu'ang Li, Lijie Wen, Irwin King, and Philip S Yu. 2023a. An unforgeable publicly verifiable watermark for large language models. *arXiv preprint arXiv:2307.16230*.

Aiwei Liu, Leyi Pan, Xuming Hu, Shiao Meng, and Lijie Wen. 2023b. A semantic invariant robust watermark for large language models. *arXiv preprint arXiv:2310.06356*.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742.

Minjia Mao, Dongjun Wei, Zeyu Chen, Xiao Fang, and Michael Chau. 2024. A watermark for low-entropy and unbiased generation in large language models. *arXiv preprint arXiv:2405.14604*.

Julien Piet, Chawin Sitawarin, Vivian Fang, Norman Mu, and David Wagner. 2023. Mark my words: Analyzing and evaluating language model watermarks. *arXiv preprint arXiv:2312.00273*.

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

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073–1083, Vancouver, Canada. Association for Computational Linguistics.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Shangqing Tu, Yuliang Sun, Yushi Bai, Jifan Yu, Lei Hou, and Juanzi Li. 2023. Waterbench: Towards holistic evaluation of watermarks for large language models. *arXiv preprint arXiv:2311.07138*.

Yihan Wu, Zhengmian Hu, Hongyang Zhang, and Heng Huang. 2023. Dipmark: A stealthy, efficient and resilient watermark for large language models. *arXiv preprint arXiv:2310.07710*.

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

A Algorithms

Algorithm 3 Generation framework.

Input: pretrained LM P_M, prompt $\boldsymbol{x}_{-m:0}$, generate length $T \in \mathbb{N}$.
for $t = 1, \ldots, T$ **do**
    Get the probability distribution of t-th token $P_M(\cdot \mid \boldsymbol{x}_{-m:t-1})$.
    Generate the probability channels $\{P_1, ..., P_l\}$
    Pseudorandomly select a probability channel P_i based on the watermark key.
    Sample the next token x_i from P_i.
return $\boldsymbol{x}_{1:T}$.

Algorithm 4 Detection framework.

Input: pretrained LM P_M, generate length $T \in \mathbb{N}$, generated tokens $\boldsymbol{x}_{1:T}$.
for $t = 1, \ldots, T$ **do**
    Recover the index of the selected distribution channel i_t based on the watermark key.
    Check whether token x_i is generated from the i_t-th distribution channel.

B Robustness-detectability trade-off

An adversary may attempt to alter the output token to disrupt the watermark detection. In MCMARK detection, if a token x_t is modified to x'_t and $x'_t \notin V_{i_t}$, the watermark signal is effectively removed. Consequently, the probability that a watermark is removed due to such an alteration is given by $\frac{|V_{i_t}|}{|V|}$. Given that $|V_{i_t}| = \frac{|V|}{l}$, the probability that a watermark is removed simplifies to $\frac{1}{l}$ Therefore, increasing l decreases the robustness of the watermark, as it increases the likelihood that an adversary can successfully remove the watermark by modifying the token.

On the other hand, moderately increasing l can enhance the detectability of the watermark. Recall that $\Pr(\Phi(\boldsymbol{x}_{1:T}) \geq z) = \sum_{i=\lceil z \rceil}^{T} \binom{T}{i} \left(\frac{1}{l}\right)^i \left(\frac{l-1}{l}\right)^{T-i}$. The derivative with respect to l of each term in the sum is given by: $\frac{d}{dl}\left(\left(\frac{1}{l}\right)^i \left(\frac{l-1}{l}\right)^{T-i}\right) = \left(\frac{1}{l}\right)^{i+2} \left(\frac{l-1}{l}\right)^{T-i} \left(-il + (T-i)\frac{l}{l-1}\right)$, indicating that when $l \geq 2$ and $i > \frac{T}{2}$, $\left(\frac{1}{l}\right)^i \left(\frac{l-1}{l}\right)^{T-i}$ decreases with increasing l. Typically, the score $\Phi(\boldsymbol{x}_{1:T})$ is greater than $\frac{T}{2}$, so increasing l may reduce the detection p-value. This reduction makes the statistical distinction between watermarked and unwatermarked text more significant, thereby improving the detectability of the watermark. Thus, we identify a fundamental trade-off: increasing the number of distribution channels l enhances the detectability of the watermark, yet it simultaneously reduces its robustness.

However, blindly increasing l may also lead to bad detectability. The detectability is not only related to the distribution of $\Phi(\boldsymbol{x}_{1:n})$ under the null hypothesis but also the scale of $\Phi(\boldsymbol{x}_{1:n})$. If l is too large, the expected true negative rate $E_{TN}^{MCmark} = \sum_{i=1}^{l} \max\{0, 1/l - p_{V_i}\}$ might be poor, since p_{V_i} are more likely to be unevenly distributed. We empirically validate our analysis in Figure 5 and 6.

C Experimental settings

Baselines We evaluate the performance of our methods against various baselines, including two biased watermarking approaches, KGW (Kirchenbauer et al., 2023a) and Unigram (Zhao et al., 2023), as well as five unbiased watermarking algorithms, ITS-edit (Kuditipudi et al., 2023), EXP-edit (Kuditipudi et al., 2023), γ-reweight (Hu et al., 2023), DiPmark (Wu et al., 2023) and STA-1 (Mao et al., 2024).

Models and Datasets we ustilize Llama-2-7b-chat-hf (Touvron et al., 2023), Llama-3.2-3B-Instruct (Dubey et al., 2024), Mistral-7B-Instruct-v0.3 (Jiang et al., 2023), Phi-3.5-mini-instruct (Abdin et al., 2024)for text generation tasks to evaluate the effectiveness of our proposed MCMARK.

Following Kirchenbauer et al. (2023a); Hu et al. (2023), we use a subset from the C4 dataset (Raffel et al., 2020) for text generation experiments. Additionally, we also include three MMW datasets (Piet et al., 2023), Dolly CW (Conover et al., 2023) and two tasks from WaterBench (Tu et al., 2023).

For unbiasedness validation, we adopt the settings from Hu et al. (2023); Wu et al. (2023), employing MBart (Liu et al., 2020) for machine translation and BART (Lewis, 2019) for text summarization.

In the machine translation experiments, we use the WMT16 ro-en dataset (Bojar et al., 2016). For text summarization, while for text summarization, we utilize the CNN/DailyMail dataset (See et al., 2017).

Watermarking parameters. We evaluate the detectability of MCMARK on the text generation task with different language models. We generate 1,000 examples for each tasks. We use the prefix 2-gram together with a secret key as the watermark keys. We select $\alpha \in \{0.3, 0.4\}$ for DiPmark, and $\delta \in \{0.5, 1.0, 1.5, 2.0\}$ and $\gamma = 0.5$ for KGW watermark (Kirchenbauer et al., 2023a), $\delta \in \{0.5, 1.0, 1.5, 2.0\}$ for Unigram (Zhao et al., 2023). For ITS-edit (Kuditipudi et al., 2023), EXP-edit (Kuditipudi et al., 2023), γ-reweight (Hu et al., 2023) and STA-1 (Mao et al., 2024), we follow the settings in the original papers. For MCMARK we set the number of distribution channels $l = 20$.

D Additional Experiments

In this section, we provide additional comparative analysis regarding the unbiased property and detectability of MCMARK. We also include an ablation study on the number of distribution channels l in MCMARK.

Unbiased Property. In Tables 3 and 4, we conduct an additional evaluation of unbiasedness for both biased and unbiased watermarks. The results confirm that MCMARK effectively preserves the language model's distribution, outperforming the biased watermark alternatives.

Detectability. In Figure 7, we assess the detectability of MCMARK on tasks such as MMW Book Report, Longform QA, and Finance QA. The results demonstrate that MCMARK consistently exhibits superior detectability across all tested models and datasets.

Ablation Study with l. In Figures 8, 9, 10, and 11, we present an analysis of the relationship between detectability and the number of distribution channels l in MCMARK. Our findings indicate that detectability initially increases and then decreases with respect to l, illustrating a critical trade-off in the parameter's configuration.

Table 3: Unbiasedness evaluation on text summarization tasks.

	BERT Score↑	PPL↓	Rouge-1↑
Baseline	0.3175	6.3932	0.3768
KGW(δ=0.5)	0.3152	6.4894	0.3746
KGW(δ=1.0)	0.3125	6.8647	0.3742
KGW(δ=1.5)	0.3067	7.4633	0.3673
KGW(δ=2.0)	0.2996	8.4847	0.3605
Unigram(δ=0.5)	0.3160	6.5302	0.3754
Unigram(δ=1.0)	0.3132	6.8145	0.3717
Unigram(δ=1.5)	0.3081	7.4693	0.3647
Unigram(δ=2.0)	0.2990	8.4182	0.3545
ITS-edit	0.3147	6.5302	0.3758
EXP-edit	0.3209	5.9945	0.3775
γ-reweight	0.3164	6.4414	0.3765
DiPmark($\alpha = 0.4$)	0.3178	6.4127	0.3773
DiPmark($\alpha = 0.3$)	0.3169	6.3867	0.3765
STA-1	0.3182	6.4118	0.3777
MCMARK(l=20)	0.3168	6.3864	0.3763

Table 4: Unbiasedness evaluation on machine translation tasks.

	BERT Score↑	BLEU↑
Baseline	0.5576	20.35
KGW(δ=0.5)	0.5560	20.25
KGW(δ=1.0)	0.5555	20.02
KGW(δ=1.5)	0.5489	18.95
KGW(δ=2.0)	0.5420	18.28
Unigram(δ=0.5)	0.5570	20.49
Unigram(δ=1.0)	0.5576	20.02
Unigram(δ=1.5)	0.5459	19.05
Unigram(δ=2.0)	0.5330	18.51
ITS-edit	0.5700	21.29
EXP-edit	0.5600	20.00
γ-reweight	0.5548	20.12
DiPmark($\alpha = 0.4$)	0.5614	20.65
DiPmark($\alpha = 0.3$)	0.5563	20.48
STA-1	0.5532	19.83
MCMARK(l=20)	0.5588	20.16

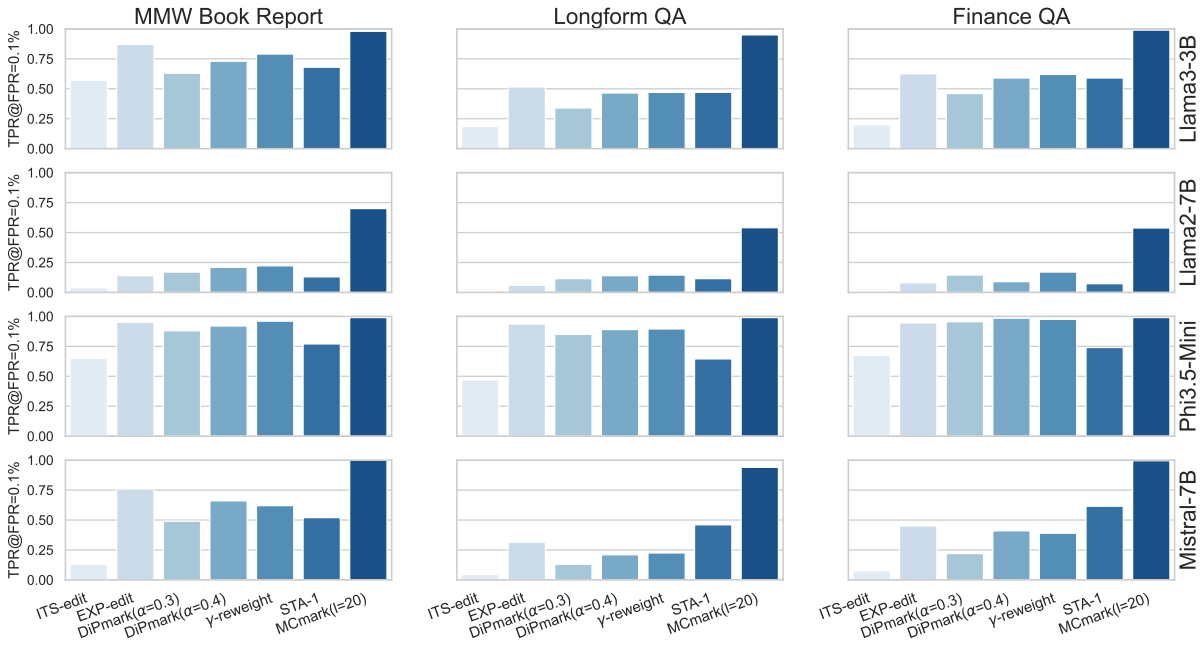


Figure 7: Comparative analysis of MCMARK against SOTA unbiased watermarks across various language models and datasets on watermark detectability.

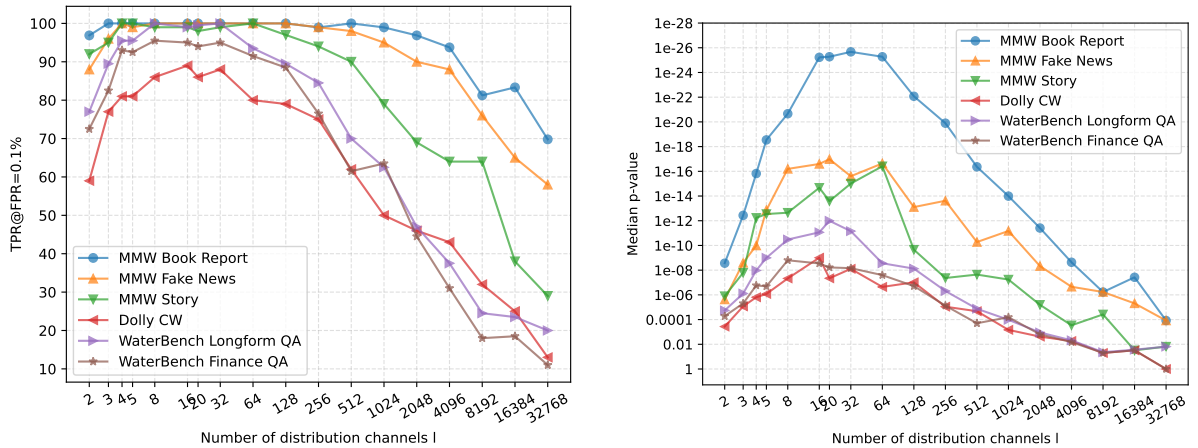


Figure 8: **Left**: Median p-value vs number of distribution channels l in MCMARK with Mistral-7B. **Right**: TPR@FPR=0.1% vs number of distribution channels l in MCMARK with Mistral-7B.

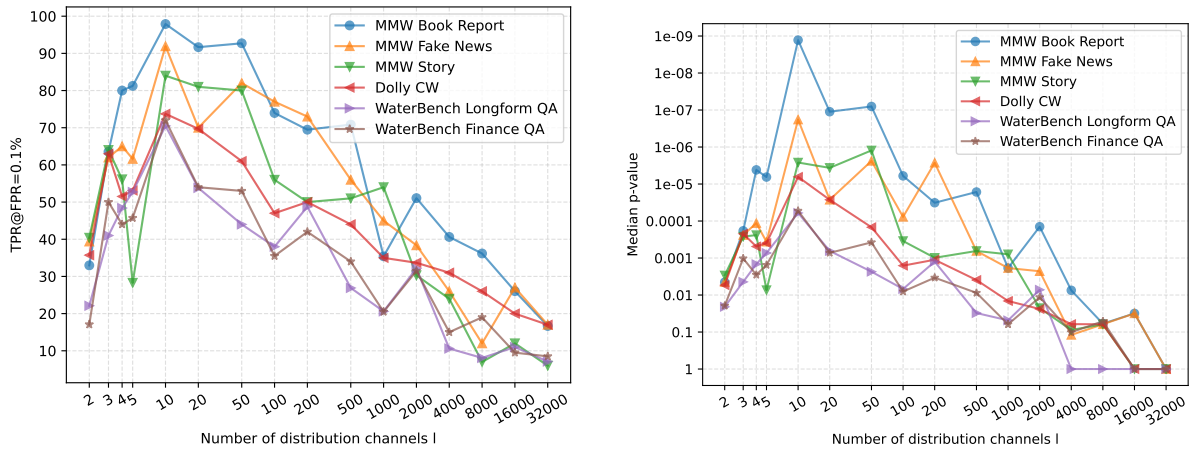


Figure 9: **Left**: Median p-value vs number of distribution channels l in MCMARK with Llama2-7B. **Right**: TPR@FPR=0.1% vs number of distribution channels l in MCMARK with Llama2-7B.

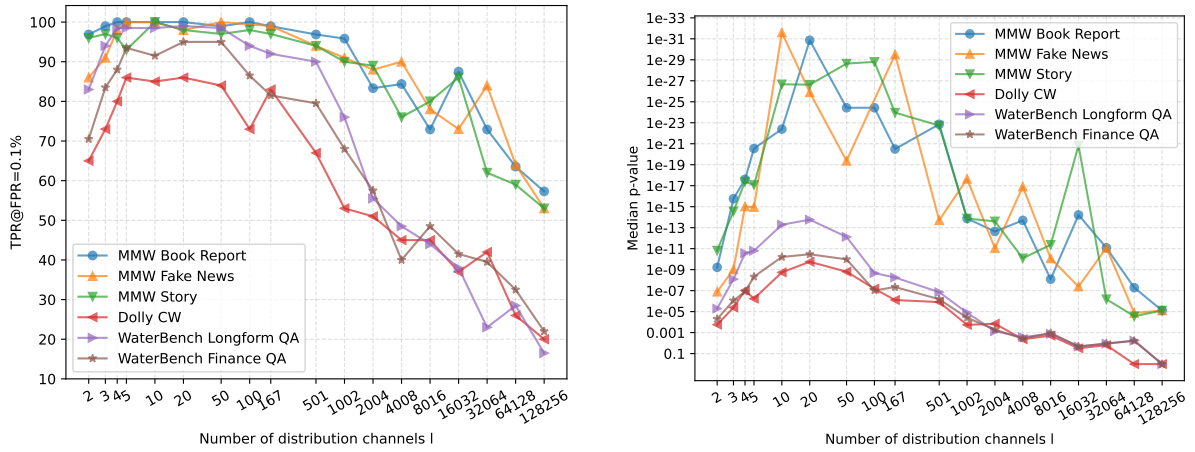
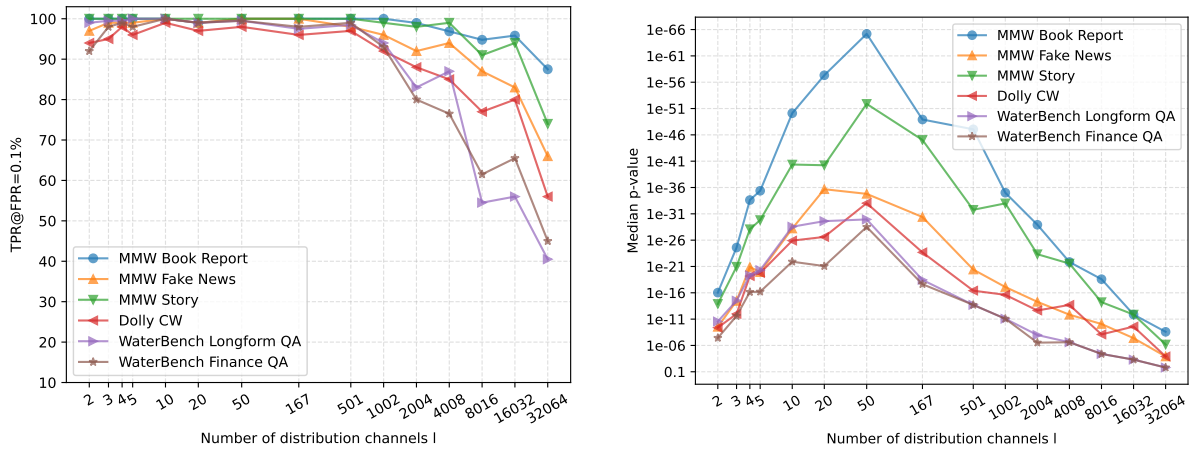


Figure 10: **Left**: Median p-value vs number of distribution channels l in MCMARK with Llama3-7B. **Right**: TPR@FPR=0.1% vs number of distribution channels l in MCMARK with Llama3-7B.



(a) TPR@FPR=0.1% vs l

(b) Median p-value vs l

Figure 11: **Left**: Median p-value vs number of distribution channels l in MCMARK with Phi3.5-Mini. **Right**: TPR@FPR=0.1% vs number of distribution channels l in MCMARK with Phi3.5-Mini.