QuantileMark: A Message-Symmetric Multi-bit Watermark for LLMs

Junlin Zhu, Baizhou Huang, Xiaojun Wan
Wangxuan Institute of Computer Technology, Peking University
zhujunlin@stu.pku.edu.cn, {hbz19, wanxiaojun}@pku.edu.cn

Abstract

As large language models become standard backends for content generation, practical provenance increasingly requires multi-bit watermarking. In provider-internal deployments, a key requirement is message symmetry: the message itself should not systematically affect either text quality or verification outcomes. Vocabulary-partition watermarks can break message symmetry in low-entropy decoding: some messages are assigned most of the probability mass, while others are forced to use tail tokens. This makes embedding quality and message decoding accuracy message-dependent. We propose QuantileMark, a white-box multi-bit watermark that embeds messages within the continuous cumulative probability interval $[0, 1)$. At each step, QuantileMark partitions this interval into M equal-mass bins and samples strictly from the bin assigned to the target symbol, ensuring a fixed $1/M$ probability budget regardless of context entropy. For detection, the verifier reconstructs the same partition under teacher forcing, computes posteriors over latent bins, and aggregates evidence for verification. We prove message-unbiasedness, a property ensuring that the base distribution is recovered when averaging over messages. This provides a theoretical foundation for generation-side symmetry, while the equal-mass design additionally promotes uniform evidence strength across messages on the detection side. Empirical results on C4 continuation and LFQA show improved multi-bit recovery and detection robustness over strong baselines, with negligible impact on generation quality. Our code is available at GitHub.

1 Introduction

Large language models (LLMs) have become standard backends for applications ranging from dialogue assistance and content creation to code generation and data analysis (OpenAI, 2022). As high-quality synthetic text becomes ubiquitous and inexpensive, provenance has emerged as a practical necessity: platforms and providers must attribute the origin of content and its deployment settings. This capability is critical not only for mitigating misinformation but also for policy enforcement, abuse response (e.g., toxic content), copyright disputes, and enterprise compliance (Yang et al., 2025).

Generative watermarking addresses this need by embedding a covert signal during the decoding process, enabling a detector to verify the model's authorship (Kirchenbauer et al., 2023). While early work focused on *zero-bit* schemes for binary presence detection, real-world deployments increasingly demand *multi-bit* provenance to encode metadata such as user IDs, model versions, or timestamps (Yoo et al., 2024). Multi-bit attribution is inherently **provider-internal**. The entity hosting the model must also control the detection service. After all, even if a third party extracts an embedded identifier, they cannot map it to a real-world identity without the provider's private user database. Consequently, the provider is the ultimate root of trust for identity resolution. This dynamic practically justifies white-box access to the model at detection time.

Unlike zero-bit watermarking, where the detector only answers a binary question of presence, multi-bit provenance requires decoding a specific message (e.g., a user ID). We represent the provenance message as a binary string segmented into m-bit symbols, so each symbol takes one of $M = 2^m$ discrete values. At each watermarking step, we embed one symbol and can convey up to m bits of information. For example, a 24-bit message can be encoded as 12 symbols when $m = 2$.

The process of encoding these messages introduces a critical new requirement: **message symmetry**. Message symmetry means that the message itself should not systematically affect either text quality or verification outcomes. In particular, this concern arises in two places. During generation,

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 6790–6806

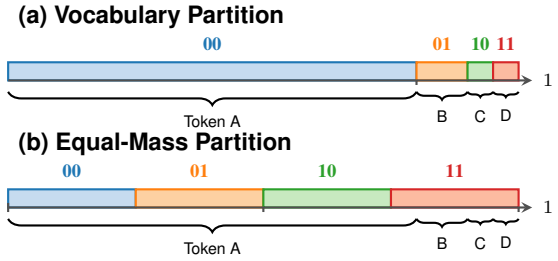

Figure 1: Impact of probability mass partitioning strategies on message symmetry under a low-entropy distribution. (a) Vocabulary Partition allocates uneven probability budgets to messages that should ideally be weighted equally. (b) Equal-Mass Partition assigns a fixed probability budget to each message.

messages should receive comparable probability-mass budgets so that text quality does not depend on the identifier. During detection, evidence should accumulate comparably across messages so that watermark detection are not message-dependent. Ensuring this symmetry is essential for operational fairness, as it guarantees that the provider can deliver consistent service quality and attribution reliability across the entire user base.

Existing generative watermarks can be broadly grouped into *distortionary* and *distribution-preserving* designs. Both design lines face challenges when message symmetry is required for multi-bit provenance.

Distortionary schemes typically employ a **vocabulary partition** to construct the signal (Yoo et al., 2024). For a message space of size M, these methods partition the vocabulary into M disjoint sets and apply a logit bias to the subset corresponding to the target symbol. This approach, however, introduces severe asymmetry under low-entropy decoding. Consider a step where the model assigns probabilities $\{0.8, 0.1, 0.05, 0.05\}$ to its top candidates, as shown in Figure 1. If the token with probability 0.8 falls into the partition set for one value, embedding that value is almost free: the model outputs its preferred token and the detector observes the signal with high probability. Conversely, for the other $M - 1$ values, their assigned sets may contain only low-probability tail tokens. Embedding these values forces the model to override its natural choice, which degrades text quality while simultaneously yielding weaker statistical evidence. Consequently, the expected evidence averaged over all messages is diluted.

Distribution-preserving methods ensure the watermarked output preserves the base distribution when marginalized over randomness. This goal can already mitigate message-dependent distortion on the generation side. However, constrained by the requirement of **black-box verification**, these schemes typically prioritize stealth through *minimal stepwise distortion* (Jiang et al., 2025). This creates a fundamental bottleneck for multi-bit provenance: the evidence is often too subtle for robust message recovery. In provider-internal settings with white-box access, such constraints are unnecessary. A more attractive direction is to enforce message symmetry explicitly, then exploit its benefit: allocate each symbol a fixed probability budget and permit larger, controlled deviations, ensuring effective evidence accumulation.

To overcome the structural asymmetry of vocabulary partitioning and the evidence limitations of existing distribution-preserving designs, we introduce **QuantileMark**, a white-box multi-bit watermark based on continuous probability mass partitioning. At each step, QuantileMark partitions the cumulative probability interval $[0, 1)$ into M bins of equal probability mass and encodes a symbol by sampling within the corresponding bin (Figure 2). During detection, the verifier reconstructs the same partition to compute posteriors for the latent bins. Equal-mass allocation enforces message symmetry by construction, and it turns symmetry into stronger and more stable stepwise evidence regardless of context entropy.

In summary, our contributions are threefold. First, we introduce QuantileMark, a provenance framework that guarantees a uniform probability budget for every message symbol via quantile partitioning, accompanied by a white-box detector that computes posteriors for decoding. Second, we formalize the notion of message-unbiasedness and prove that QuantileMark satisfies this property, providing a theoretical guarantee for generation-side symmetry. Finally, we demonstrate that QuantileMark outperforms vocabulary-based baselines on C4 and LFQA in recovery and robustness, while maintaining generation quality.

2 Preliminaries

We study a provider-internal watermarking setting where a model generates a sequence $x_{1:T}$ using a fixed secret key K. Let $h_t := x_{<t}$ denote the history at step t, and let $p_0(\cdot \mid h_t)$ denote the base next-token distribution. Our goal is to embed a message $S = (s_1, \ldots, s_H)$ consisting of

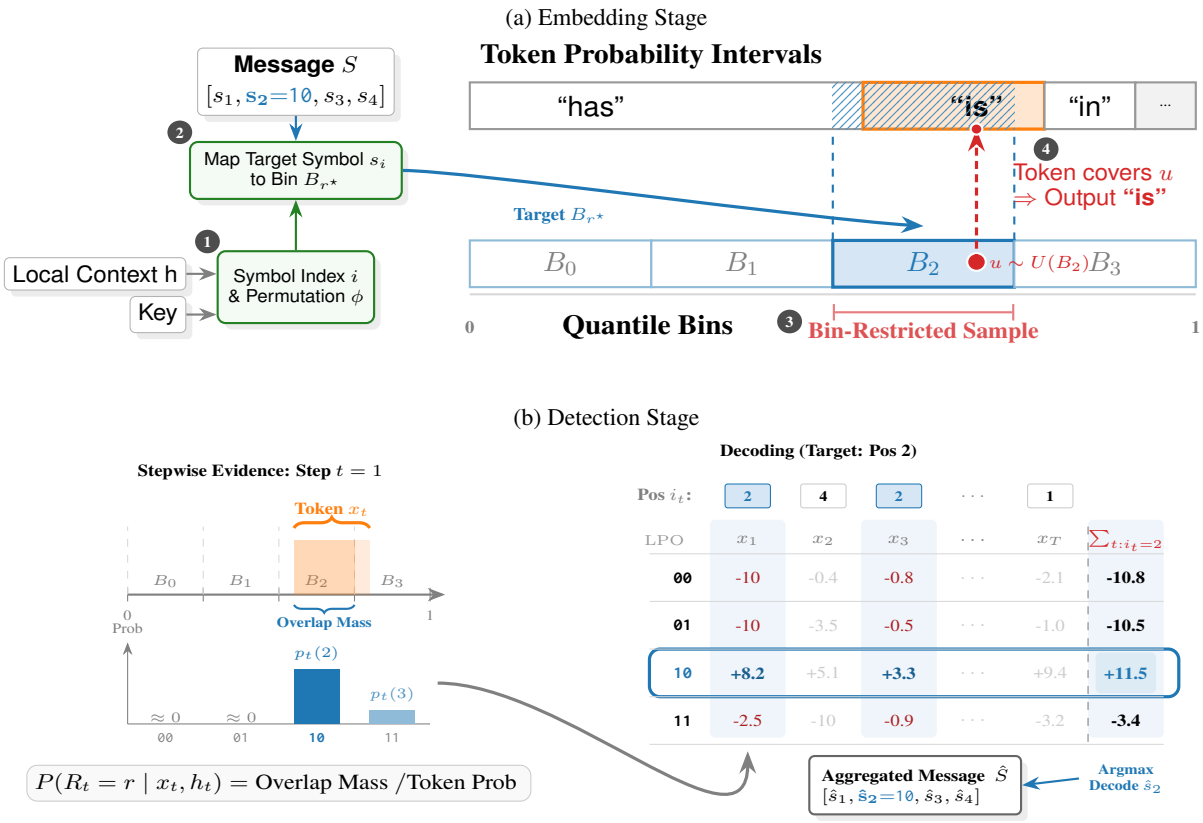

Figure 2: Overview of QuantileMark, where $m = 2$, $H = 4$. (a) **Embedding:** The key-derived logic (green) maps symbol to target bin $B_{r^\star}$ (blue frame). A random value u (red dashed line) is sampled uniformly within this bin, forcing the selection of the token *is* whose interval covers u. (b) **Detection:** Posteriors (blue bars) form the evidence LPO at this step. These scores are aggregated across steps assigned to position $i_t = 2$ (right, blue columns) to decode $\hat{s}_2$.

H discrete symbols. Each symbol s_h encodes m bits of information and is drawn from the set $[M] \coloneqq \{0, \ldots, M-1\}$, where $M = 2^m$.

Position Allocation. Unlike zero-bit presence detection, recovering a multi-bit message requires locating where each symbol is embedded. A standard abstraction is *position allocation* (Yoo et al., 2024): a keyed function maps each step t to an index $i_t \in \{1, \ldots, H\}$. This implies that step t carries evidence *only* about the specific message symbol s_{i_t}.

Vocabulary Partitioning as Multi-Bit Channel. Many existing methods construct the channel by partitioning the vocabulary. At step t, the vocabulary is split into M disjoint sets $\{\mathcal{V}_t^{(r)}\}_{r=0}^{M-1}$. To embed symbol s_{i_t}, the sampler promotes tokens in $\mathcal{V}_t^{(s_{i_t})}$. In this paradigm, the abstract message symbol is tied directly to discrete token identities.

CDF and Quantile Function View of Sampling. It is convenient to represent discrete sampling using the cumulative distribution function (CDF) F and its inverse, the quantile function $Q(u) = F^{-1}(u)$. Under an ordering of tokens at step t, each token corresponds to a specific interval on the cumulative probability interval $[0, 1)$ whose length equals its probability. Standard sampling is operationally equivalent to drawing $u \sim \mathrm{Unif}[0, 1)$ and selecting the output token via $x = Q(u)$, **which returns the token whose interval contains** u. This geometric view allows us to decouple the channel from discrete token identities by defining the watermark logic directly on the continuous domain of the quantile function. We refer to embedding strategies that operate on this continuous interval as **quantile watermarking**.

3 Methodology: Equal-Mass Quantile Watermarking

We propose **QuantileMark**, a white-box watermarking framework designed to ensure **message**

symmetry by defining the embedding channel (i.e., the medium where the watermark signal is injected) on the cumulative probability interval rather than vocabulary space.

The framework consists of two procedures (Figure 2). During generation, QuantileMark partitions the stepwise cumulative probability interval into M equal-mass quantile bins and samples a token strictly from the bin assigned to the target message symbol. During detection, the verifier reconstructs this quantile geometry, computing the posterior probability of the latent bin to aggregate evidence and decode the message.

3.1 Setup: Message and Position Allocation

Let $x_{1:T}$ be the generated sequence and h_t the history at step t. We embed a message $S = (s_1, \ldots, s_H)$ consisting of H symbols, where each symbol $s_h \in \{0, \ldots, M-1\}$ and $M = 2^m$.

At each step t, we derive pseudo-random parameters from a secret key K and a local context window g_t (the previous w tokens). Specifically, we generate a position allocation index $i_t \in \{1, \ldots, H\}$ that determines which message symbol s_{i_t} to embed at the current step, alongside a keyed permutation ϕ_t that maps this symbol to a specific target bin index on the CDF.

The permutation ϕ_t ensures that a fixed symbol value is not systematically bound to the same quantile bin across steps, which is essential for the unbiasedness guarantees in Section 3.4.

3.2 Embedding via Quantile Partitioning

The core of our embedding strategy is to enforce a uniform probability budget for every symbol by operating on the continuous domain of the quantile function. This requires partitioning the unit interval and calculating the mass overlap to bridge the continuous design with the discrete vocabulary.

Quantile Geometry and Overlap Mass. At step t, we sort the vocabulary by probability to map each token v to a sub-interval $I_t(v) \subset [0, 1)$ of length $p_0(v|h_t)$. We simultaneously partition $[0, 1)$ into M fixed bins $B_r = [r/M, (r+1)/M)$. The alignment between token intervals and bins is captured by the **overlap mass**:

$$\mu_t(v, r) = |I_t(v) \cap B_r| \,. \tag{1}$$

This explicitly quantifies the mass of v falling within B_r, bridging the continuous design with the discrete vocabulary and enabling the use of partial mass from tokens that straddle bin boundaries.

Bin-Restricted Sampling. To embed the target symbol s_{i_t}, we identify the target bin $r_t^\star = \phi_t(s_{i_t})$. Instead of standard sampling (drawing $u \in [0, 1)$ globally), QuantileMark samples a random value u_t strictly within the target bin $B_{r_t^\star}$. The output token x_t is determined by finding the token whose interval $I_t(x_t)$ contains u_t. The resulting watermarked distribution for a chosen bin r is:

$$p_{\mathrm{wm}}(v \mid h_t, r) = M \cdot \mu_t(v, r). \tag{2}$$

This sampling mechanism guarantees that every target symbol receives an identical probability budget $1/M$ regardless of context entropy. Consequently, this structurally enforces message symmetry while effectively exploiting it: in contrast to minimal-distortion baselines like StealthInk, we concentrate the entire probability mass into the target bin, substantially boosting the stepwise evidence available for detection.

3.3 Decoding and Detection

The detection process follows a *decode-then-test* paradigm: we first reconstruct the quantile geometry to recover the most likely message $\hat{S}$, and then compute a detection score based on the confidence of this recovery.

Stepwise Evidence Extraction. The core of our detection strategy is to recover soft evidence about the embedding channel using teacher forcing. At each step t, the detector reconstructs the base distribution $p_0(\cdot \mid h_t)$ and the quantile geometry. The observed token x_t occupies the interval $I_t(x_t)$. Given that the sampling value must lie within this interval, the posterior probability that it originated from bin r is the proportion of the token's mass overlapping that bin:

$$p_t(r) = P(R_t = r \mid x_t, h_t) = \frac{\mu_t(x_t, r)}{p_0(x_t \mid h_t)}. \tag{3}$$

Crucially, this posterior provides soft evidence. Unlike vocabulary partitioning, which forces a hard assignment (casting a vote for a single symbol and rejecting the other $M-1$), $p_t(r)$ distributes support proportional to the overlap mass. For example, a token straddling bins r_1 and r_2 yields positive support for both while decisively ruling out non-overlapping bins.

To aggregate this evidence numerically, we compute the Log Posterior Odds (LPO):

$$\mathrm{LPO}_t(r) = \log\left(\frac{p_t(r)}{1 - p_t(r)}\right). \tag{4}$$

For numerical stability, we clip $p_t(r)$ to the range $[\epsilon, 1-\epsilon]$.

Message Decoding. Since the position allocation i_t is deterministic given the key, we aggregate evidence for each message symbol s_h independently. The decoder identifies the symbol $\hat{s}_h$ that maximizes the cumulative LPO across all steps assigned to index h:

$$\hat{s}_h = \underset{s\in\{0,\dots,M-1\}}{\arg\max} \sum_{t:i_t=h} \mathrm{LPO}_t\big(\phi_t(s)\big). \quad (5)$$

Here, $\phi_t(s)$ maps the candidate symbol s to its corresponding bin index at step t.

Sequence-Level Scoring. Once the message $\hat{S} = (\hat{s}_1, \dots, \hat{s}_H)$ is decoded, we evaluate the overall presence of the watermark. We define the implied target bin path as $\hat{r}_t = \phi_t(\hat{s}_{i_t})$ and compute the average evidence along this path: $T(x_{1:T}) = \frac{1}{T}\sum_{t=1}^{T} \mathrm{LPO}_t(\hat{r}_t)$

The watermark is detected if $T(x_{1:T}) > \theta$, where θ is a threshold calibrated to control the false positive rate on non-watermarked text.

3.4 Unbiasedness Properties of Equal-Mass Channelization

We assume a uniform prior over the message space. Under this assumption, the target symbol s at any given step is uniformly distributed over $[M]$.

Definition 1 (Message-unbiasedness at a step)**.** Fix a context h_t and key K. Let s be a target symbol drawn uniformly from $[M]$, and let $R = \phi_t(s)$ be the relabeled bin index. The scheme is message-unbiased if, for all tokens v,

$$\mathbb{E}_s\big[p_{\mathrm{wm}}(v \mid h_t, R)\big] = p_0(v \mid h_t).$$

Lemma 1 (Equal-mass bins imply message-unbiasedness)**.** Assume ϕ_t maps a uniform symbol to a uniform bin in $[M]$, then QuantileMark is message-unbiased at step t.

The proof follows from the linearity of expectation and the partition of unity property of the quantile bins; see Appendix A.2 for details.

Message-unbiasedness ensures that, on average, the watermark introduces no distortion if the message is unknown. Crucially, this property serves as a structural pathway to satisfying the dual notion of **cipher-unbiasedness** (Jiang et al., 2025), which requires the distribution to be preserved when the message is fixed but the key is randomized.

Definition 2 (Cipher-unbiasedness at a step)**.** Fix a context h_t, a key K, and a symbol $s \in [M]$. Let Z be a seed random variable, and let $\Phi_{t,Z}$ denote the resulting key-conditioned permutation on $[M]$ produced from Z. The scheme is cipher-unbiased if, for all tokens v,

$$\mathbb{E}_Z[p_{\mathrm{wm}}(v \mid h_t, \Phi_{t,Z}(s))] = p_0(v \mid h_t).$$

QuantileMark satisfies Definition 2 whenever, for any fixed s, the induced bin index $\Phi_{t,Z}(s)$ is uniform on $[M]$ under the seed distribution. The proof is given in Appendix A (Lemma 3).

4 Experiments

We evaluate QuantileMark in a provider-internal setting, focusing on accurate *multi-bit* recovery from clean generations and robustness against realistic deployment mismatches. Our results confirm that QuantileMark significantly improves detection stability over vocabulary-based baselines while maintaining generation quality.

4.1 Experimental Setup

We conduct experiments on two tasks: open-ended continuation on C4 (Raffel et al., 2020) using Llama-2-7B (Touvron et al., 2023), and long-form QA on ELI5/LFQA (Fan et al., 2019) using Llama-3.1-8B-Instruct (Grattafiori et al., 2024). We pair each task with a suitable model, using a base model for continuation and an instruction-tuned model for QA, to ensure the evaluation reflects realistic deployment behavior. Unless otherwise stated, results are averaged over 500 watermarked and 500 non-watermarked samples per task, all fixed to a length of $T = 300$ tokens. We use human references as non-watermarked text to simulate realistic detection against organic text. Appendix E.1 additionally evaluates against non-watermarked model generations to isolate the watermark signal from distributional mismatch.

We embed a 24-bit message using top-k=128 sampling and temperature τ=1.0. We compare QuantileMark (m=2) against MPAC (Yoo et al., 2024) (m=2, γ=0.25, δ=2.0) and StealthInk (Jiang et al., 2025) (m=1), following the recommended settings for best trade-offs. Detection of QuantileMark reconstructs geometry via teacher forcing on output tokens (adding chat template headers for Instruct models) with matching top-k/τ; detector mismatch is analyzed separately

Method	C4					LFQA				
	Bit Acc ↑	AUC ↑	TPR@ 1%FPR ↑	PPL ↓	Time (s) ↓	Bit Acc ↑	AUC ↑	TPR@ 1%FPR ↑	PPL ↓	Time (s) ↓
No watermark	–	–	–	7.684	–	–	–	–	2.647	–
MPAC	0.9702	0.9887	0.9800	10.351	**0.103**	0.8770	0.9756	0.8056	3.793	**0.111**
StealthInk	0.9003	0.9869	0.7920	8.235	0.645	0.7447	0.8301	0.2425	**2.636**	1.013
QuantileMark (ours)	**0.9893**	**0.9995**	**0.9840**	**7.404**	0.343	**0.9500**	**0.9997**	**1.0**	2.759	0.348

Table 1: Generation and detection performance on C4 and LFQA with 24 bits embedded in 300 tokens. Time (s) denotes average detection time per sample.

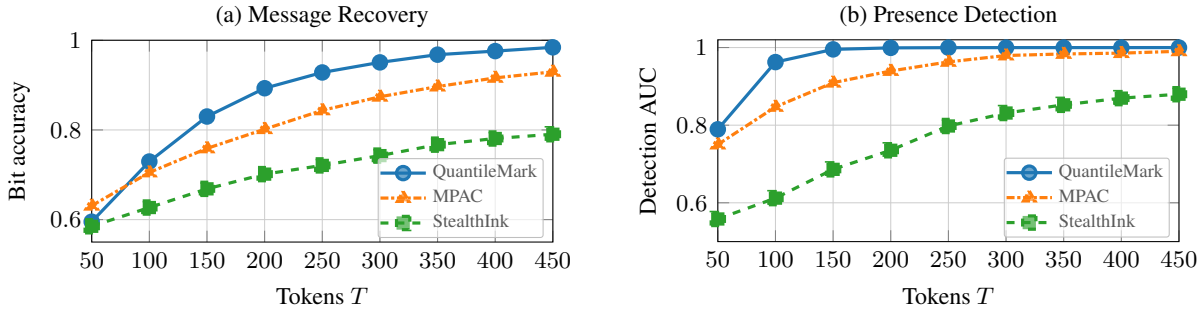

Figure 3: Varying detection length T on Llama-3.1-8B-Instruct for LFQA.

in Section 4.4. We report Bit Acc (bit-wise accuracy, averaged over 500 watermarked samples), detection AUC, TPR@1%FPR, and perplexity (PPL).

4.2 Generation and Detection Results

Table 1 summarizes the generation and detection performance of different methods on two tasks. QuantileMark achieves consistently strong multi-bit recovery and near-saturated detection, with high bit accuracy and AUC and TPR@1%FPR close to 1.0 on both C4 and LFQA. Notably, these gains persist on LFQA, where the next-token distribution is often more peaked and vocabulary-partition methods can allocate highly uneven probability mass across symbol values, destabilizing detection.

In terms of generation quality, QuantileMark maintains PPL close to the unwatermarked baseline; minor fluctuations (e.g., 7.404 vs 7.684 on C4) fall within expected finite-sample randomness, consistent with our theoretical message-unbiasedness. In contrast, MPAC incurs a noticeable PPL increase due to distortionary green-list bias, while StealthInk shows weaker recovery at the same message length. This performance comes with negligible operational overhead: although detection requires a model forward pass, the computational cost is strictly bounded by the generation latency itself, making verification inherently affordable for any provider capable of hosting the service.

Varying the generated length T (Figure 3) confirms that QuantileMark accumulates evidence efficiently, reaching high recovery and detection rates with fewer tokens than baselines.

We next vary the symbol size m while keeping the message length fixed at 24 bits. Figure 4 shows a trade-off between symbol resolution and reliable evidence. Moderate choices ($m = 2, 3$) perform best, while $m = 4$ sharply reduces recovery even though detection remains strong. Intuitively, since a single observed token reveals limited information about the latent bin (Appendix A.3), increasing m under a fixed-length budget fails to provide sufficient supporting tokens for M-ary decisions. This effectively dilutes the information per symbol, explaining the drop in bit accuracy observed at $m = 4$.

The lower AUC at $m = 1$ stems from the sharp CDF geometry of human text. Human tokens often fall in the tail, yielding nearly deterministic bin posteriors. With binary partitions ($M = 2$), the detector's maximization step can easily find a spurious message that fortuitously aligns with these strong signals. Increasing to $M = 4$ makes such chance agreement statistically much harder—as each token supports a specific bin while rejecting three others—thereby improving separation.

4.3 Robustness to Scrubbing Attacks on C4

We evaluate robustness on C4 under four scrubbing attacks commonly used in prior multi-bit wa-

Method	No attack		Copy-paste ($\epsilon = 0.2$)		Synonym ($\epsilon = 0.2$)		Deletion ($\epsilon = 0.1$)		Paraphrase (Dipper)	
	Bit Acc ↑	AUC ↑	Bit Acc ↑	AUC ↑	Bit Acc ↑	AUC ↑	Bit Acc ↑	AUC ↑	Bit Acc ↑	AUC ↑
MPAC	0.9702	0.9887	0.9438	0.9849	0.9476	0.9876	**0.8811**	**0.9750**	0.7201	**0.7936**
StealthInk	0.9003	0.9869	0.8497	0.9550	0.8608	0.9701	0.7839	0.8854	0.6829	0.6536
QuantileMark	**0.9893**	**0.9995**	**0.9730**	**0.9972**	**0.9712**	**0.9963**	0.8712	0.9426	**0.7640**	0.7609

Table 2: Robustness on C4 with 24 bits embedded in 300 tokens. The attack intensities ϵ denote the fraction of tokens modified.

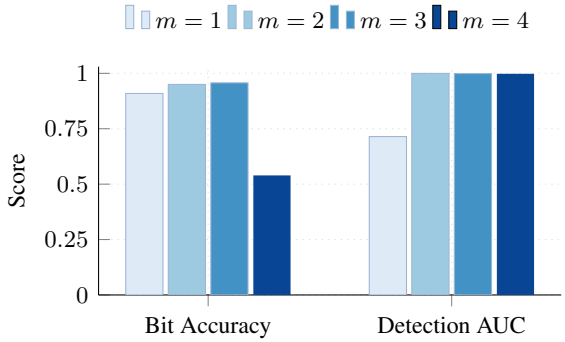

Figure 4: LFQA ablation of m for QuantileMark with 24 bits embedded. We vary the bits per symbol m ($M = 2^m$ quantile bins).

termark evaluations (Jiang et al., 2025): copy-paste mixing with non-watermarked text, synonym substitution, random deletion, and paraphrasing. Table 2 reports bit accuracy and detection AUC; full attack specifications are described in Appendix D.2.

While the watermark resists local lexical edits (copy-paste and substitution), deletions and paraphrasing prove significantly more damaging. These attacks induce synchronization drift and degrade the context quality required to reconstruct the quantile geometry, thereby hindering the recovery of long multi-bit messages. Robust detection under paraphrasing remains an open challenge across all existing multi-bit watermarking schemes.

4.4 Detector-Generator Mismatch on LFQA

We evaluate detector-generator mismatch when text is generated with $(k{=}128, \tau{=}1.0)$ but detector varies $(\hat{k}, \hat{\tau})$. Figure 5 shows that performance degrades smoothly under moderate mismatch: bit accuracy remains high across $\hat{\tau} \in [0.8, 1.2]$ and remains stable even for small detector top-$\hat{k}$ (e.g., $\hat{k} = 8$). This robustness is expected in our white-box setting because the detector reconstructs quantile geometry from logits: moderate changes in truncation or temperature often preserve the probability ordering on the head of the distribution, so

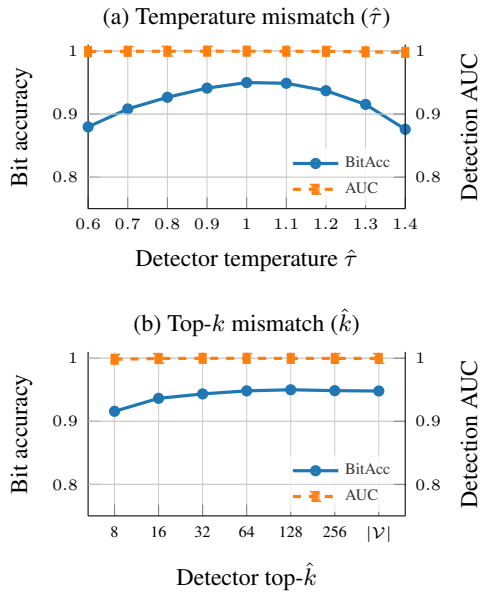

Figure 5: Detector-generator mismatch analysis on Llama-3.1-8B-Instruct.

overlap masses and channel posteriors shift gradually rather than catastrophically.

4.5 Utility on Downstream Tasks

Finally, we assess whether watermarking degrades downstream utility under top-k=128 sampling. For text summarization, we use BART-large (Lewis et al., 2020) evaluated on CNN/DailyMail (Hermann et al., 2015). For machine translation, we run Multilingual BART (mBART) (Liu et al., 2020) on the WMT'14 En-Ro corpus (Bojar et al., 2014). We also include a reference-free LFQA evaluation using GPT-4o, with the full experimental setup and results detailed in Appendix E.2.

Table 3 shows that QuantileMark consistently outperforms other watermarking baselines, achieving utility scores that are nearly identical to the no-watermark upper bound across both tasks.

5 Related Work

We focus on generative watermarking schemes that modify the sampling process, distinguishing between distortionary partitioning and distribution-

Method	Machine translation			Summarization		
	BLEU ↑	R-1 ↑	BERT ↑	R-1 ↑	BERT ↑	PPL ↓
No watermark	17.55	49.13	35.82	37.42	21.40	5.18
MPAC	16.89	47.98	34.89	36.81	20.57	5.81
StealthInk	**17.42**	48.97	35.75	37.08	20.83	5.72
QuantileMark	17.41	**48.99**	**35.99**	**37.44**	**21.37**	**5.21**

Table 3: Downstream generation quality under top-k=128 sampling. For machine translation, we report BLEU, ROUGE-1 (R-1), and BERTScore (BERT). For summarization, we report ROUGE-1, BERTScore, and perplexity (PPL).

preserving methods.

5.1 Distortionary Vocabulary Partitioning Multi-bit Watermarks

A dominant line of work constructs a channel by partitioning the vocabulary and biasing the next token distribution toward a key dependent subset. The watermark of Kirchenbauer et al. (2023) introduces green list promotion with statistical tests on set hits, followed by analyses of calibration and robustness under edits (Piet et al., 2023; Kirchenbauer et al., 2024). Multi bit extensions retain the same mechanism while adding structure across positions, for example via position allocation and multiple colorings in MPAC (Yoo et al., 2024), or via coding designs over partition assignments (Wang et al., 2024; Fernandez et al., 2023). Several works improve robustness by adding error correcting codes from the view of message encoding (Chao et al., 2025; Qu et al., 2024).

However, standard partitioning schemes inherently suffer from probability mass imbalance, particularly in peaked distributions. While recent heuristics like rank-based partitioning (Park et al., 2025) or entropy-based gating (Gu et al., 2025) mitigate this issue, they remain adaptive approximations. In contrast, QuantileMark addresses this structurally by redefining channels in CDF interval, guaranteeing an equal probability budget by construction rather than adaptive rejection.

5.2 Distribution-preserving and Unbiased Watermarks

A parallel line of work aims to preserve the base model distribution in expectation over the watermark randomness. Two complementary perspectives predominate in the literature: *RNG-space* constructions, which define a measure-preserving transformation on the sampling randomness (Kuditipudi et al., 2024), and *reweighting* constructions, which formulate watermarking as a randomized modification of the stepwise token distribution (Hu et al., 2024).

RNG-space distortion-free watermarks. Kuditipudi et al. (2024) propose distortion-free watermarking by mapping a key-derived random number sequence to samples from the language model, and instantiate the framework with inverse transform sampling and exponential minimum sampling. SynthID-Text introduces Tournament sampling (Dathathri et al., 2024). Multi-bit extensions encode information by transforming the sampling randomness based on the message: DISC employs cyclic shifts (Kordi Boroujeny et al., 2024), while MirrorMark utilizes measure-preserving mirroring (Jiang et al., 2026).

Expectation-unbiased reweighting. Hu et al. (2024) formalize unbiased watermarking as randomized reweighting of the stepwise distribution. For autoregressive generation, preserving the sequence distribution requires appropriate independence of the watermark codes across steps (Hu et al., 2024). DiPmark (Wu et al., 2024) and StealthInk (Jiang et al., 2025) further explore distribution-preserving designs, with StealthInk emphasizing text-only detection and multi-bit provenance.

QuantileMark shares the design of defining watermarks through structured sampling randomness but leverages white-box access to optimize for detection reliability rather than black-box stealth. Unlike prior schemes that rely on randomized vocabulary partitions, QuantileMark enforces equal-mass channelization on the cumulative probability interval. This structural design substantially boosts the stepwise statistical evidence available for decoding, all while maintaining the theoretical unbiasedness of the generation process.

6 Conclusion

We presented **QuantileMark**, a white-box multi-bit watermark designed to mitigate the challenges of provenance embedding in low-entropy regimes. Instead of relying on vocabulary partitioning, QuantileMark defines the channel in cumulative probability interval by dividing it into M equal-mass bins. This structural approach ensures that every symbol receives a fixed probability budget, providing a basis for *message symmetry* where messages are treated with consistent statistical weight. Com-

plementing this embedding, we derived a teacher-forced detector that exploits the reconstructed quantile geometry to compute posteriors over latent bins, enabling a coherent decode-then-test process. Empirical results across base and instruction-tuned models demonstrate the feasibility and effectiveness of this design, showing improved recovery rates and detection performance compared to vocabulary-based baselines, while preserving generation quality. QuantileMark thus offers a practically viable solution for provider-internal attribution that balances message symmetry with operational utility.

Limitations

Our work has several limitations that suggest directions for future research. First, robustness remains limited under heavy paraphrasing and other edits that substantially rewrite local contexts, since such changes can disrupt position allocation and the seed stream used for per-step relabeling and evidence aggregation. Improving paraphrase resilience may require stronger position allocation strategies that tolerate sequence misalignment and a PRF or seeding logic that depends less on fragile local token neighborhoods while remaining reproducible for the key holder. Second, our study focused on a provider-internal setting and assumed white-box access at detection time, which rules out public verification from text alone in the current form. Although detection may transfer to settings with a proxy model, a distilled model, or a closely related model family sharing the same tokenizer, we did not systematically study how model mismatch and distribution shift affect the reconstructed geometry and downstream recovery. Finally, our experimental evaluation was limited in scope, utilizing a small set of tasks and model families. Extending evaluation across broader decoding policies settings would better characterize the generality of quantile-based channelization for multi-bit provenance.

Ethical Considerations

The main ethical goal of multi-bit watermarking is accountability. It allows providers to trace malicious outputs, such as disinformation, back to the responsible user. While this naturally raises privacy concerns about user surveillance, the risk is limited because the watermark is passive. It only matters if a user shares the generated text publicly; if the text remains private, the watermark is harmless. Furthermore, our provider-internal design protects user privacy better than black-box schemes. The embedded ID is meaningless to third parties. Only the model provider can extract the signal and link it to a real user. Ultimately, while providers have this tracking ability, they must use it strictly for safety and compliance, guided by strong privacy policies, rather than for invasive profiling.

Acknowledgements

This work was supported by Beijing Natural Science Foundation (L253001), Key Laboratory of Science, Technology and Standard in Press Industry (Key Laboratory of Intelligent Press Media Technology) and National Engineering Research Center of New Electronic Publishing Technologies. We appreciate the anonymous reviewers for their helpful comments. Xiaojun Wan is the contact author.

References

Ondrej Bojar, Christian Buck, Christian Federmann, Barry Haddow, Philipp Koehn, Johannes Leveling, Christof Monz, Pavel Pecina, Matt Post, Herve Saint-Amand, Radu Soricut, Lucia Specia, and Ale s Tamchyna. 2014. Findings of the 2014 workshop on statistical machine translation. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*, pages 12–58, Baltimore, Maryland, USA. Association for Computational Linguistics.

Patrick Chao, Yan Sun, Edgar Dobriban, and Hamed Hassani. 2025. Watermarking language models with error correcting codes. *arXiv preprint arXiv:2406.10281*.

Sumanth Dathathri, Abigail See, Sumedh Ghaisas, Po-Sen Huang, Rob McAdam, Johannes Welbl, Vandana Bachani, Alex Kaskasoli, Robert Stanforth, Tatiana Matejovicova, Jamie Hayes, Nidhi Vyas, Majd Al Merey, Jonah Brown-Cohen, Rudy Bunel, Borja Balle, A. Taylan Cemgil, Zahra Ahmed, Kitty Stacpoole, and 5 others. 2024. Scalable watermarking for identifying large language model outputs. *Nat.*, 634(8035):818–823.

Angela Fan, Yacine Jernite, Ethan Perez, David Grangier, Jason Weston, and Michael Auli. 2019. ELI5: long form question answering. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 3558–3567. Association for Computational Linguistics.

Pierre Fernandez, Antoine Chaffin, Karim Tit, Vivien Chappelier, and Teddy Furon. 2023. Three bricks to consolidate watermarks for large language models.

In *2023 IEEE International Workshop on Information Forensics and Security (WIFS)*. IEEE.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, and 542 others. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Tianle Gu, Zongqi Wang, Kexin Huang, Yuanqi Yao, Xiangliang Zhang, Yujiu Yang, and Xiuying Chen. 2025. Invisible entropy: Towards safe and efficient low-entropy LLM watermarking. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 6727–6744, Suzhou, China. Association for Computational Linguistics.

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. *Advances in neural information processing systems*, 28.

Zhengmian Hu, Lichang Chen, Xidong Wu, Yihan Wu, Hongyang Zhang, and Heng Huang. 2024. Unbiased watermark for large language models. In *The Twelfth International Conference on Learning Representations*.

Ya Jiang, Massieh Kordi Boroujeny, Surender Suresh Kumar, and Kai Zeng. 2026. Mirrormark: A distortion-free multi-bit watermark for large language models. *Preprint*, arXiv:2601.22246.

Ya Jiang, Chuxiong Wu, Massieh Kordi Boroujeny, Brian L. Mark, and Kai Zeng. 2025. Stealthink: A multi-bit and stealthy watermark for large language models. *arXiv preprint arXiv:2506.05502*.

John Kirchenbauer, Jonas Geiping, Yuxin Wen, Jonathan Katz, Ian Miers, and Tom Goldstein. 2023. A watermark for large language models. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 17061–17084. PMLR.

John Kirchenbauer, Jonas Geiping, Yuxin Wen, Manli Shu, Khalid Saifullah, Kezhi Kong, Kasun Fernando, Aniruddha Saha, Micah Goldblum, and Tom Goldstein. 2024. On the reliability of watermarks for large language models. In *ICLR*.

Massieh Kordi Boroujeny, Ya Jiang, Kai Zeng, and Brian Mark. 2024. Multi-Bit Distortion-Free Watermarking for Large Language Models. *arXiv preprint arXiv:2402.16578*.

Kalpesh Krishna, Yixiao Song, Marzena Karpinska, John Wieting, and Mohit Iyyer. 2023. Paraphrasing evades detectors of ai-generated text, but retrieval is an effective defense. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA. Curran Associates Inc.

Rohith Kuditipudi, John Thickstun, Tatsunori Hashimoto, and Percy Liang. 2024. Robust distortion-free watermarks for language models. *Transactions on Machine Learning Research*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880.

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.

OpenAI, :, Aaron Hurst, Adam Lerer, Adam P. Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, Aleksander Mądry, Alex Baker-Whitcomb, Alex Beutel, Alex Borzunov, Alex Carney, Alex Chow, Alex Kirillov, and 401 others. 2024. Gpt-4o system card. *Preprint*, arXiv:2410.21276.

OpenAI. 2022. ChatGPT: Optimizing language models for dialogue. Website. `https://openai.com/blog/chatgpt`.

Shinwoo Park, Hyejin Park, Hyeseon Ahn, and Yo-Sub Han. 2025. Watermod: Modular token-rank partitioning for probability-balanced llm watermarking. *Preprint*, arXiv:2511.07863.

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*.

Wenjie Qu, Dong Yin, Zixin He, Wei Zou, Tianyang Tao, Jinyuan Jia, and Jiaheng Zhang. 2024. Provably Robust Multi-bit Watermarking for AI-generated Text via Error Correction Code. *arXiv preprint arXiv:2401.16820*.

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).

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, and 1 others. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Lean Wang, Wenkai Yang, Deli Chen, Hao Zhou, Yankai Lin, Fandong Meng, Jie Zhou, and Xu Sun. 2024. Towards codable watermarking for injecting multi-bits information to LLMs. In *The Twelfth International Conference on Learning Representations*.

Yihan Wu, Zhengmian Hu, Hongyang Zhang, and Heng Huang. 2024. Dipmark: A stealthy, efficient and resilient watermark for large language models. In *Proceedings of the 41st International Conference on Machine Learning*.

Zhiguang Yang, Gejian Zhao, and Hanzhou Wu. 2025. Watermarking for large language models: A survey. *Mathematics*, 13(9):1420.

KiYoon Yoo, Wonhyuk Ahn, and Nojun Kwak. 2024. Advancing beyond identification: Multi-bit watermark for large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 4031–4055, Mexico City, Mexico. Association for Computational Linguistics.

A Channel Geometry and Message-Unbiasedness

This appendix collects auxiliary results that complement Section 3.2–3.4. We reuse the stepwise channel geometry from the main text but switch to a context-level notation by dropping the step index. Fix a context h and its decoding distribution $p_0(\cdot \mid h)$. Define the discrete CDF F_h, token CDF intervals $I_h(y)$, equal-width quantile bins $\{B_r\}_{r\in[M]}$, and overlap mass

$$\mu_h(y, r) \coloneqq |I_h(y) \cap B_r|$$

exactly as in Section 3.2.

A.1 Basic Overlap Identities

Lemma 2 (Basic overlap identities)**.** For every context h:

$$\sum_{r=0}^{M-1} \mu_h(y, r) = |I_h(y)| = p_0(y \mid h), \qquad \forall y \in \mathcal{V}, \tag{6}$$

$$\sum_{y\in\mathcal{V}} \mu_h(y, r) = |B_r| = \frac{1}{M}, \qquad \forall r \in [M]. \tag{7}$$

Proof. Equation (6) holds because $\{B_r\}_r$ partitions $[0, 1)$. Equation (7) holds because $\{I_h(y)\}_y$ partitions $[0, 1)$.

A.2 Unbiasedness Notions: Message and Cipher

We recall two notions introduced in the main text. Message-unbiasedness is defined in Definition 1, and cipher-unbiasedness (a StealthInk-style viewpoint) is defined in Definition 2. In this appendix we use a context-level notation by dropping the step index: fix a context h and the corresponding base distribution $p_0(\cdot \mid h)$. The overlap mass $\mu_h(y, r)$ is defined exactly as in Section 3.2, and the per-channel distribution satisfies

$$p_{\mathrm{wm}}(y \mid h, r) = M\, \mu_h(y, r). \tag{8}$$

We now give the proof of Lemma 1, which is stated in the main text.

Proof of Lemma 1. Fix h. By assumption, ϕ maps a uniform symbol to a uniform bin, so $R = \phi(s)$ is uniform on $[M]$ when s is uniform on $[M]$. Therefore, for any token $y \in \mathcal{V}$,

$$\begin{aligned}
\mathbb{E}_s[p_{\mathrm{wm}}(y \mid h, R)] &= \mathbb{E}_R[p_{\mathrm{wm}}(y \mid h, R)] \\
&= \frac{1}{M} \sum_{r=0}^{M-1} p_{\mathrm{wm}}(y \mid h, r) \\
&= \frac{1}{M} \sum_{r=0}^{M-1} M\, \mu_h(y, r) \\
&= \sum_{r=0}^{M-1} \mu_h(y, r) \\
&= p_0(y \mid h),
\end{aligned} \tag{9}$$

where the last equality uses (6).

Lemma 3 (Uniform cipher-induced relabeling implies cipher-unbiasedness)**.** Fix (h, s). Assume that under the seed prior in Definition 2, the key-derived permutation Φ induces $R = \Phi(s)$ that is uniform on $[M]$. Then for every token $y \in \mathcal{V}$,

$$\mathbb{E}[p_{\mathrm{wm}}(y \mid h, \Phi(s))] = p_0(y \mid h). \tag{10}$$

Proof. Under the assumption, R is uniform on $[M]$ even with s fixed. Hence

$$\begin{aligned}
\mathbb{E}[p_{\mathrm{wm}}(y \mid h, \Phi(s))] &= \mathbb{E}_R[p_{\mathrm{wm}}(y \mid h, R)] \\
&= \frac{1}{M} \sum_{r=0}^{M-1} p_{\mathrm{wm}}(y \mid h, r) \\
&= \frac{1}{M} \sum_{r=0}^{M-1} M\, \mu_h(y, r) \\
&= \sum_{r=0}^{M-1} \mu_h(y, r) = p_0(y \mid h),
\end{aligned} \tag{11}$$

where the last step uses (6).

A.3 Signal Dilution and Information Capacity

This subsection analyzes a fundamental decoding limitation: the information recoverable about the bin index is capped by the token surprisal $-\log_2 p_0(y \mid h)$, regardless of watermark capacity m.

Posterior Bound. Fix a context h and observed token y. Assuming a uniform prior $\mathbb{P}(R = r) = 1/M$, the posterior for bin r is given by Eq. (26): $\mathbb{P}(R = r \mid y, h) = \mu_h(y, r)/p_0(y \mid h)$. Since the overlap mass satisfies $\mu_h(y, r) \le |B_r| = 1/M$, the posterior is uniformly bounded:

$$\max_{r\in[M]} \mathbb{P}(R = r \mid y, h) \;\le\; \min\left\{1, \frac{1}{M p_0(y \mid h)}\right\}. \tag{12}$$

Intuitively, if a token's probability mass p_0 spans many bins ($Mp_0 \gg 1$), the probability is diluted across all contained bins, preventing confident identification of any single bin.

Information Cap. The information gain $\Delta(y) \coloneqq H(R) - H(R \mid y, h)$ is bounded by the min-entropy. Using (12):

$$\Delta(y) \le \min\{\log_2 M,\; -\log_2 p_0(y \mid h)\}\,. \tag{13}$$

This bound implies that high-probability tokens cannot carry more bits of watermark information than their own self-information. For instance, if $p_0(y \mid h) \ge 0.5$, observing y yields at most 1 bit of information about the bin location, regardless of how large M is. Reliable multi-bit recovery therefore relies on aggregating partial evidence across the sequence rather than identifying the bin from a single step.

A.4 Time-Averaged Bin Occupancy under a Fixed Message

This subsection models the empirical occupancy of quantile bins along a long continuation. Fix a message $s_{1:H} \in [M]^H$ and consider a length-T generated continuation $x_{1:T}$. At step t, the embedding rule selects a target bin index

$$r_t^\star \coloneqq \phi_t(s_{i_t}) \in [M],$$

where (i_t, ϕ_t) are derived from the key K and the step seed (e.g., $z_t = \mathrm{Hash}(g_t)$ as in the main text). Define the empirical bin occupancy

$$\hat\pi_T(r) \coloneqq \frac{1}{T}\sum_{t=1}^{T} \mathbf{1}\{r_t^\star = r\}, \qquad r \in [M]. \tag{14}$$

When $\hat\pi_T$ is close to uniform, the scheme exhibits a channel-hopping effect over time: although each step samples within a single bin, the visited bins cover $[M]$ nearly evenly at the sequence level.

We formalize this effect under an idealized PRF model, conditioning on the realized sequence of step seeds. Let $\mathcal{Z}_T$ be the set of distinct seeds appearing among $\{z_t\}_{t=1}^T$. For each $z \in \mathcal{Z}_T$, define its multiplicity

$$n_z \coloneqq \big|\{t \in [T] : z_t = z\}\big|, \qquad \sum_{z\in\mathcal{Z}_T} n_z = T. \tag{15}$$

Because PRF outputs are functions of z, repeated seeds reuse the same (i_t, ϕ_t). Thus, for each distinct seed z, define the induced bin label

$$R_z \coloneqq \phi_z(s_{i_z}) \in [M],$$

where (i_z, ϕ_z) denotes the PRF-derived pair associated with seed z. Abbreviate

$$S_T \coloneqq \sum_{z\in\mathcal{Z}_T} n_z^2, \qquad T_{\text{eff}} \coloneqq \frac{T^2}{S_T}. \tag{16}$$

Here T_{eff} acts as an effective sample size: repeated seeds inflate S_T and reduce concentration.

Proposition 1 (Key-random occupancy is near-uniform, with an effective sample size)**.** Assume an idealized PRF model in which, conditional on the realized seed sequence $(z_1, \ldots, z_T)$, the pairs $\{(i_z, \phi_z)\}_{z\in\mathcal{Z}_T}$ are independent across distinct seeds, and each ϕ_z is a uniform random permutation on $[M]$ independent of i_z. Then for any fixed message $s_{1:H}$ and any $r \in [M]$,

$$\mathbb{E}[\hat\pi_T(r) \mid (z_1, \ldots, z_T)] = \frac{1}{M}. \tag{17}$$

Moreover, for any $\epsilon > 0$, let $\mathbf{z} \coloneqq (z_1, \ldots, z_T)$.

$$\mathbb{P}\left(\left|\hat\pi_T(r) - \frac{1}{M}\right| \ge \epsilon \,\middle|\, \mathbf{z}\right) \le 2\exp\left(-\frac{2\epsilon^2 T^2}{S_T}\right) = 2\exp\left(-2\epsilon^2 T_{\text{eff}}\right). \tag{18}$$

Proof. Condition on $(z_1, \ldots, z_T)$. Grouping repeated seeds rewrites (14) as a weighted sum over distinct seeds:

$$\hat\pi_T(r) = \sum_{z\in\mathcal{Z}_T} \frac{n_z}{T}\,\mathbf{1}\{R_z = r\}. \tag{19}$$

Under the permutation assumption, for any fixed symbol $d \in [M]$ the image $\phi_z(d)$ is uniform on $[M]$, so $R_z = \phi_z(s_{i_z})$ is uniform on $[M]$ and $\mathbb{E}[\mathbf{1}\{R_z = r\} \mid (z_1, \ldots, z_T)] = 1/M$, proving (17).

For concentration, let $X_z \coloneqq \mathbf{1}\{R_z = r\} \in \{0, 1\}$ and weights $w_z \coloneqq n_z/T$. The variables $\{X_z\}_{z \in \mathcal{Z}_T}$ are independent and $w_z(X_z - \mathbb{E}X_z) \in [-w_z, w_z]$. Applying the standard weighted Hoeffding inequality yields

$$\mathbb{P}\left(\left|\sum_{z \in \mathcal{Z}_T} w_z \left(X_z - \frac{1}{M}\right)\right| \geq \epsilon \,\middle|\, \mathbf{z}\right) \leq 2\exp\left(-\frac{2\epsilon^2}{\sum_{z \in \mathcal{Z}_T} w_z^2}\right). \tag{20}$$

Substituting $\sum_z w_z^2 = \sum_z (n_z/T)^2 = S_T/T^2$ gives (18).

This proposition highlights a purely *key-driven* channel-hopping effect. Even with a fixed message $s_{1:H}$, the target mapping $r_t^\star$ visits bins nearly uniformly over time. This uniform coverage is critical for preventing *structural bias*, ensuring that a deterministic message does not continuously target a small subset of bins, which would otherwise induce a perceptible distributional shift.

Crucially, this uniformity ($\hat{\pi}_T(r) \approx 1/M$) describes the distribution of the *target labels*, not necessarily the token statistics of unwatermarked text. The PRF construction achieves a dual goal: it generates a target distribution that is statistically near-uniform (preventing the systematic overuse of any single bin) while remaining fully deterministic and reproducible for the key holder (enabling decoding). Finally, the effective sample size T_{eff} accounts for the redundancy caused by seed collisions: repeated seeds force the reuse of PRF outputs, thereby reducing the number of independent samples and slowing the convergence to uniformity.

B Detection as a Composite GLRT with LPO Evidence

This appendix formalizes the detection process as a Generalized Likelihood Ratio Test (GLRT) and justifies the use of Log Posterior Odds (LPO) as the stepwise evidence metric.

B.1 Hypotheses and the Composite Alternative

Let $x_{1:T}$ be an observed continuation and $h_t = x_{<t}$. We test

$$H_0 : x_{1:T} \sim p_0, \tag{21}$$

$$H_1 : x_{1:T} \sim p_{\text{wm}}(\cdot \mid s) \text{ for some message } s \in \mathcal{S}. \tag{22}$$

We assume the detector has white-box access to the model and can reconstruct $p_0(\cdot \mid h_t)$ via teacher forcing. All PRF-derived assignments are reproducible because they are derived per step from the secret key and a hash of a local context $g_t = x_{t-w:t-1}$ in the generated stream.

B.2 Sequence Likelihood under a Fixed Message

At each step, a message symbol determines a target channel index $r_t^\star \in [M]$ (after relabeling), and the per-step watermarked distribution is $p_{\text{wm}}(x_t \mid h_t, s) = M\, \mu_{h_t}(x_t, r_t^\star)$. Using the standard autoregressive factorization, we write

$$\begin{aligned} p_{\text{wm}}(x_{1:T} \mid s) &= \prod_{t=1}^{T} p_{\text{wm}}(x_t \mid h_t, s) \\ &= \prod_{t=1}^{T} M\, \mu_{h_t}(x_t, r_t^\star), \\ p_0(x_{1:T}) &= \prod_{t=1}^{T} p_0(x_t \mid h_t). \end{aligned} \tag{23}$$

B.3 Decode-then-Test Equals a GLRT

The generalized likelihood ratio statistic for H_0 versus the composite alternative H_1 is

$$\Lambda_{\text{GLRT}}(x_{1:T}) \coloneqq \frac{\max_{s \in \mathcal{S}} p_{\text{wm}}(x_{1:T} \mid s)}{p_0(x_{1:T})}. \tag{24}$$

Let $\hat{s} \in \arg\max_{s \in \mathcal{S}} \log p_{\text{wm}}(x_{1:T} \mid s)$ be the decoded message. Then the decode–then–test score

$$T_{\text{GLRT}}(x_{1:T}) \coloneqq \log p_{\text{wm}}(x_{1:T} \mid \hat{s}) - \log p_0(x_{1:T}) \tag{25}$$

equals $\log \Lambda_{\text{GLRT}}(x_{1:T})$ up to tie-breaking.

B.4 Channel Posterior and Token Evidence

Introduce a latent channel variable $R_t \in [M]$ indicating which quantile bin generated token x_t. Given teacher-forced reconstruction of $p_0(\cdot \mid h_t)$ and the overlap geometry,

$$\mathbb{P}(R_t = r \mid x_t, h_t) = \frac{\mu_{h_t}(x_t, r)}{p_0(x_t \mid h_t)}. \tag{26}$$

We define per-token log-posterior odds (LPO) evidence for the channel event $R_t = r$:

$$\mathrm{LPO}_t(r) := \log \frac{\mathbb{P}(R_t = r \mid x_t, h_t)}{1 - \mathbb{P}(R_t = r \mid x_t, h_t)}. \quad (27)$$

To avoid overloading watermark-level hypotheses, define the event-level hypotheses $G_1(r) : R_t = r$ and $G_0(r) : R_t \neq r$. Assume a uniform prior $\mathbb{P}(R_t = r) = 1/M$.

Lemma 4 (Token LPO as an event-level likelihood ratio up to a constant)**.** Let the mixture under $G_0(r)$ be

$$p(x_t \mid G_0(r)) = \frac{1}{M-1} \sum_{j \neq r} p(x_t \mid R_t = j).$$

Then

$$\mathrm{LPO}_t(r) = \log \frac{p(x_t \mid G_1(r))}{p(x_t \mid G_0(r))} - \log(M-1). \quad (28)$$

Proof. By Bayes' rule,

$$\begin{aligned} &\frac{\mathbb{P}(R_t = r \mid x_t, h_t)}{\mathbb{P}(R_t \neq r \mid x_t, h_t)} \\ &\quad = \frac{p(x_t \mid R_t = r)\,\mathbb{P}(R_t = r)}{\sum_{j \neq r} p(x_t \mid R_t = j)\,\mathbb{P}(R_t = j)} \\ &\quad = \frac{p(x_t \mid R_t = r)}{\sum_{j \neq r} p(x_t \mid R_t = j)} \\ &\quad = \frac{1}{M-1} \cdot \frac{p(x_t \mid R_t = r)}{p(x_t \mid G_0(r))}. \end{aligned}$$

Taking logs yields the claim.

B.5 Comparison: LPO vs. LLR

For a fixed decoded message, the standard Log-Likelihood Ratio (LLR) evidence is:

$$\mathrm{LLR}_t(r) := \log \frac{p_{\mathrm{wm}}(x_t \mid h_t, r)}{p_0(x_t \mid h_t)} = \log M + \log p_t(r). \quad (29)$$

While LLR and LPO are functionally related via $\mathrm{LPO}_t(r) = \mathrm{LLR}_t(r) - \log M - \log(1 - p_t(r))$, LPO offers distinct numerical advantages.

Symmetry and Dynamic Range. We implement a clipped posterior $p_t(r) \in [\epsilon, 1-\epsilon]$ (with $\epsilon = 10^{-6}$) for stability.

- **LLR** is asymmetric and capped from above: $\mathrm{LLR}_t(r) \leq \log M$. Even if the token is perfectly aligned ($p_t(r) \approx 1$), the positive evidence is limited by the bit-width m.
- **LPO** is symmetric and unbounded (before clipping): $\mathrm{LPO}_t \in [-C, C]$ where $C \approx 13.8$. This allows near-certain alignments to contribute significantly more evidence than $\log M$, improving separation when $p_t(r) \to 1$.

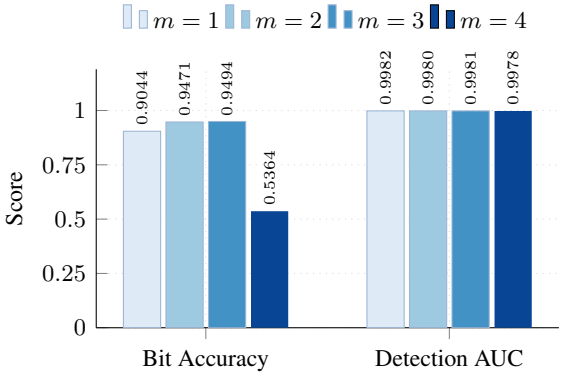

Figure 6: Performance comparison on LFQA with varying bits per symbol (m) using LLR as the token-level evidence. This serves as a complement to Figure 4, which uses LPO.

Evidence	Bit Acc ↑	Detection AUC ↑	Score on wm ($\mu \pm \sigma$)	Score on no wm ($\mu \pm \sigma$)
C4				
LPO	0.9868	0.9989	2.215 ± 1.134	-2.791 ± 0.531
LLR	0.9879	0.9751	-0.9069 ± 0.414	-3.336 ± 0.659
LFQA				
LPO	0.9500	0.9997	-0.046 ± 0.949	-3.152 ± 0.410
LLR	0.9470	0.9980	-0.899 ± 0.359	-4.291 ± 0.554

Table 4: Comparison of LPO and LLR evidence within QuantileMark ($m = 2, H = 12$) on C4 and LFQA. Score entries report mean ± standard deviation.

Empirical Validation. Table 4 and Figure 6 compare decoding using LPO versus LLR. While bit accuracy is similar (indicating decoding is driven by the rank order of posteriors), LPO yields higher detection AUC and better separation on C4.

C Relation to RNG Space and Reweighting Viewpoints

Section 5 surveys distribution preserving watermarking from the RNG space and reweighting viewpoints. This appendix clarifies how QuantileMark fits into the broader landscape of distribution-preserving watermarks, specifically comparing the handling of randomness and reweighting mechanisms.

C.1 RNG-Space Discretization and Unbiasedness

Standard distortion-free schemes typically define a measure-preserving map on the sampling randomness $u_t \in [0, 1)$, ensuring $x_t \sim p_0$ for *every* fixed message. In contrast, QuantileMark discretizes the RNG space $[0, 1)$ into M disjoint bins and restricts sampling to a specific bin $B_{r^\star}$. Conceptually, this partitions the entropy source into a watermark-controlled variable (the bin index) and residual stochasticity (uniform sampling within the bin).

While this restriction implies the distribution is distorted for a *fixed* message, Lemma 1 proves that the scheme recovers p_0 when marginalized over the message or key. This design prioritizes **message symmetry** over strict per-message stealth. In provider-internal settings, this trade-off is advantageous: it maximizes the statistical evidence for detection while maintaining fairness and unbiasedness on average across the user base.

C.2 Vocabulary Shuffling versus CDF Message Relabeling

Some unbiased reweighting rules rely on shuffling a vocabulary order before applying accept amplify style operations (Hu et al., 2024; Jiang et al., 2025). QuantileMark does not require vocabulary shuffling. It keeps the CDF geometry fixed by the model induced probability ordering and uses only a low dimensional relabeling on $[M]$.

At each step, the PRF produces a message index assignment i_t and a permutation ϕ_t on $[M]$. The permutation relabels message symbols into bin indices, and the bins themselves are defined by equal mass partitioning of the stepwise CDF. The randomized object is thus an element of $[M]$ or a permutation on $[M]$, rather than a randomized mask over $\mathcal{V}$. This keeps the embedding rule simple and makes reconstruction straightforward in a white box setting, since the detector can rebuild the same quantile geometry from logits and evaluate overlap masses directly.

D Detailed Experiment Setup

This appendix provides additional details on data preprocessing, filtering protocols, and the configuration of robustness attacks used in Section 4.

D.1 Dataset Details

C4 (Open-ended continuation). We utilize the `realnewslike` subset of the C4 dataset (Raffel et al., 2020). For each example, we truncate the first 50 tokens to serve as the prompt and treat the subsequent text as the human reference. To ensure valid comparisons, we first filter the dataset to retain only examples where the human reference exceeds the target evaluation length T.

LFQA (Long-form Question Answering). We use the ELI5/LFQA dataset (Fan et al., 2019), where the provided questions serve directly as prompts for the model. Similar to C4, we filter for human answers that exceed length T.

D.2 Attack Specifications

We evaluate robustness against four types of post-hoc editing attacks. For the first three attacks, the parameter ϵ controls the intensity of the distortion.

- **Copy-Paste Mixing:** This attack simulates a scenario where watermarked content is embedded within a larger non-watermarked context. We construct the attacked text by mixing a fraction $1 - \epsilon$ of the watermarked generation with a fraction ϵ of non-watermarked text. The watermarked segment is inserted contiguously to simulate a copy-paste operation.
- **Synonym Substitution:** We replace a fraction ϵ of the tokens in the watermarked text with their synonyms using a predefined synonym dictionary, while preserving the original sentence structure.
- **Random Deletion:** We randomly select and remove a fraction ϵ of tokens from the watermarked sequence.
- **Paraphrasing:** We employ DIPPER (Krishna et al., 2023), a paraphrase generation model designed for controlling lexical and syntactic diversity. Following standard evaluation protocols, we configure DIPPER with lexical diversity $L = 20$ and order diversity $O = 0$.

E Additional Results

E.1 Model Generation as Non-Watermarked Text

In the main text, we use human references as detection negatives (non-watermarked text) to reflect

Method	C4		LFQA	
	AUC ↑	TPR@ 1%FPR ↑	AUC ↑	TPR@ 1%FPR ↑
MPAC	0.9993	0.986	0.9744	0.8337
StealthInk	0.9851	0.6960	0.8081	0.1824
QuantileMark	**0.9998**	**0.9900**	**0.9985**	**0.9499**

Table 5: Detection performance on C4 and LFQA with 24 bits embedded in 300 tokens. Negatives are non-watermarked text generated by model.

Method	GPT-4o ↑
No watermark	4.115
MPAC	3.921
StealthInk	4.047
QuantileMark	**4.109**

Table 6: LFQA answer quality judged by GPT-4o. Scores are on a 1–5 scale; higher is better. We randomly sample 500 prompts from LFQA and generate answers with a maximum length of T=300 new tokens. During evaluation, we exclude outputs shorter than 50 tokens to mitigate potential length-related bias.

a deployment setting where provenance is verified among organic text. Since our detector is model-assisted and reconstructs token-level geometry via teacher forcing, the choice of negatives can interact with distribution mismatch between human text and the model. To isolate the watermark signal from such mismatch, we additionally evaluate detection where negatives are unwatermarked generations from the same base model, produced with the same prompts and decoding parameters as the watermarked texts. We report AUC and TPR at 1% FPR under this matched-model control, as Table 5 shows.

E.2 GPT-4o Judging for LFQA

We employ GPT-4o (OpenAI et al., 2024) as a reference-free judge for LFQA. The evaluator assesses the generated text directly and assigns a single integer score on a scale of 1 to 5. Following the protocol established by MirrorMark (Jiang et al., 2026), we design the prompt to focus exclusively on linguistic quality while explicitly ignoring artifacts resulting from truncation. Table 6 shows the result.

GPT-4o Judge Prompt

You are a strict and consistent text-quality evaluator. Use ONLY the given text; do NOT assume the author's intent. The text may start or end abruptly because the generation length is fixed. Do NOT penalize truncation or incompleteness. Do NOT judge factual correctness.
Rate each field as an integer from 1 to 5. **overall** is an independent judgment; do NOT compute overall from the other fields (no arithmetic).
Rate the following text. Return only a JSON object in exactly the following structure:

```
{
  "coherence": int,
  "clarity": int,
  "naturalness": int,
  "overall": int
}
```

Text: [TEXT]

E.3 Scaling to Longer Messages

We evaluate QuantileMark with longer messages embedded in a fixed sequence of T=300 tokens on LFQA, using m=2 bits per symbol.

Message length (bits)	Bit Acc ↑	AUC ↑
24	0.9500	0.9997
32	0.9196	0.9997
40	0.9025	0.9997
48	0.8801	0.9994
56	0.8690	0.9995

Table 7: Effect of increasing message length on LFQA (T=300, m=2). Detection AUC remains near-perfect, while bit accuracy degrades gracefully as each symbol receives fewer supporting tokens.

These results demonstrate that QuantileMark scales well to longer messages, maintaining robust detection performance while degrading gracefully in bit recovery accuracy. As the message grows, each symbol is allocated fewer supporting tokens, reducing the per-symbol evidence budget.

E.4 Mitigating Repeated Seeds

As noted in Section 3.1, the per-step parameters (i_t, ϕ_t) are derived from a hash of the local context window g_t. When consecutive steps produce identical context hashes (seed collisions), the same bin assignment is reused, which can introduce sentence-level distributional bias even though per-step message-unbiasedness holds.

A simple mitigation is to **skip embedding** at any step whose seed has already appeared in the current sequence, treating the skipped step as an erasure.

Table 8 evaluates this strategy on LFQA with 24 bits embedded in 300 tokens.

Method	Bit Acc ↑	AUC ↑	PPL ↓
QuantileMark	0.9500	0.9997	2.759
+ skip repeated seed	0.9438	0.9996	2.644

Table 8: Effect of skipping repeated seeds on LFQA. Skipping slightly reduces PPL and mitigates sentence-level bias, while maintaining detection performance.

As expected, skipping reduces perplexity (confirming that sentence-level bias is mitigated) while detection performance remains essentially unchanged, thanks to the robust evidence aggregation of the detector.