TextSeal: A Localized LLM Watermark for Provenance & Distillation Protection

Tom Sander[*†] **Hongyan Chang**[†] **Tomáš Souček** **Sylvestre-Alvise Rebuffi** **Valeriu Lacatusu**
Alexandre Mourachko **Tuan Tran** **Hady Elsahar**[†] **Pierre Fernandez**[*†]
FAIR, Meta Superintelligence Labs
tomsander@meta.com, pfz@meta.com
[*]Equal contribution. [†]Core team. Code: https://github.com/facebookresearch/textseal

Abstract

We introduce *TextSeal*, a state-of-the-art watermark for large language models. Building on Gumbel-max sampling, TextSeal introduces dual-key generation to restore output diversity, along with entropy-weighted scoring and multi-region localization for improved robustness. It supports serving optimizations such as speculative decoding and multi-token prediction, and does not add any inference overhead. TextSeal outperforms baselines like SynthID-text in detection strength at every diversity level and is robust to dilution, maintaining confident localized detection even in heavily mixed human/AI documents. The scheme is theoretically distortion-free, and evaluation across reasoning benchmarks confirms that it preserves downstream performance; while a multilingual human evaluation (6,000 A/B comparisons, 5 languages) shows no perceptible quality difference. Beyond its use for provenance detection, TextSeal is also "radioactive": its watermark signal transfers through model distillation, enabling detection of unauthorized use.

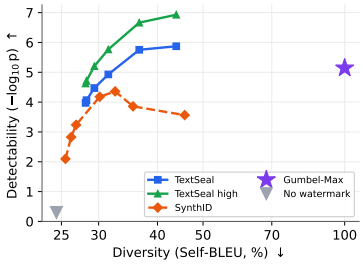
1 Introduction

LLM watermarking embeds an imperceptible yet detectable signal during text generation. It enables identifying AI-generated content (as required by the EU AI Act [European Parliament and Council of the European Union, 2024]), tracking how model outputs propagate through downstream pipelines, and detecting unauthorized distillation [Sander et al., 2024, Sablayrolles et al., 2020].

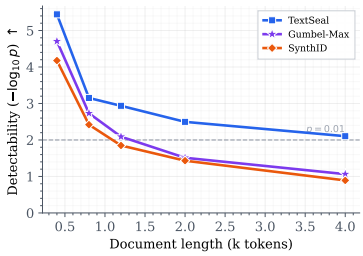
For production deployment, *distortion-free* watermarking is highly desirable: it keeps the exact decoding configuration (temperature, top-p) the model was deployed and tuned for, embedding the watermark at zero cost to any individual generation's quality. A recent comparison [Fernandez et al., 2025] shows that Gumbel-Max watermark [Aaronson and Kirchner, 2023] achieves the best detectability-perplexity Pareto frontier by far among all methods. However, Gumbel-Max has two drawbacks: it is fully *deterministic* (a fixed prompt and secret key always produce the same output, eliminating diversity), and its single-key design can trigger degenerate loops when repeated n-grams cause the PRF to lock onto the same token (Remark 1). SynthID [Dathathri et al., 2024] (deployed in Google's Gemini) resolves this determinism while remaining distortion-free, but its tournament-sampling design forces detection to rely on a Z-test over binary scores. Building on the Gumbel-max framework, we benefit from the more powerful Gamma test, which we further extend to leverage token-level entropy while preserving its calibration. We also propose a localization scheme that identifies watermarked segments, essential when watermarked and human-written text are mixed.

We introduce *TextSeal*, a distortion-free, non-deterministic watermark for LLMs that outperforms SynthID at every diversity level. Building upon the Gumbel-max framework, TextSeal introduces three improvements, each evaluated in section 4 and further in section 5:

40th Conference on Neural Information Processing Systems (NeurIPS 2026).



(a) Diversity–detectability trade-off.



(b) Detectability under dilution.

Task	None	TextSeal
AIME	40.1	41.1
MATH	79.8	79.8
GSM8K	95.4	96.0
HumanEval	97.0	93.3
MBPP	50.2	49.2
ARC-C	88.3	88.5
ARC-E	93.4	93.7
GPQA	50.5	50.0
HellaSwag	94.7	94.8
MMLU	49.2	51.5
SimpleQA	15.8	16.0
WinoGrande	93.2	93.5
Average	**70.6**	**70.6**

(c) Performance.

Figure 1: **TextSeal overview** (Qwen3.5-27B). **(a)** TextSeal outperforms SynthID on the diversity-detectability frontier (ELI5, 400 tok, T=0.8, top-p=0.9; SynthID with Google DeepMind's g-values). **(b)** Localized detection stays confident at 10× dilution where global baselines fail. **(c)** Downstream accuracy is preserved across all 12 reasoning benchmarks at T=0.6.

1. **Dual-Key Generation** overcomes determinism by randomly alternating between two secret keys during generation, restoring diversity at a small cost in detection power. This supports speculative decoding and Multi-Token Prediction without additional latency (subsection 3.1, App. E.7).
2. **Entropy-Weighted Detection** uses early-fusion scoring tailored to dual keys, entropy-aware weighting, and moment-matched Gamma approximations for calibrated p-values (subsection 3.2).
3. **Localized Detection** identifies individual watermarked segments within a document via a multi-region geometric cover search, dramatically boosting detection under dilution (subsection 3.3).

TextSeal adds only $\leq 0.3\%$ sampling overhead (3× faster than SynthID; section 5). Beyond provenance, TextSeal is *radioactive* [Sander et al., 2024]: its signal transfers through model distillation, providing a safeguard against unauthorized use of model outputs (section 6). Figure 1 summarizes our results: TextSeal outperforms SynthID on the diversity-detectability frontier, maintains confident detection at 10× dilution, and preserves full accuracy across all 12 reasoning benchmarks.

2 Background and Related Work

2.1 LLM Watermarking

Early text watermarking relied on edit-based methods [Topkara et al., 2005, 2006c] with low robustness. For LLMs, two concurrent approaches appeared after ChatGPT: green-red list biasing [Kirchenbauer et al., 2023a] and Gumbel-max sampling [Aaronson and Kirchner, 2023], both using pseudo-random seeds from a secret key and preceding tokens, enabling lightweight detection without access to the model. Some subsequent work explores multi-bit watermarking [Fernandez et al., 2023, Yoo et al., 2024, Qu et al., 2024], undetectable constructions [Christ et al., 2023, Kuditipudi et al., 2023], low-entropy optimizations [Lee et al., 2023, Huang et al., 2023], semantic watermarks [Liu et al., 2023, Liu and Bu, 2024, Hou et al., 2023], adaptive green-red variants [Wang et al., 2025], distillation for open-weights model [Gu et al., 2023], etc. See Appendix A.3 for detailed scheme descriptions.

Beyond detection, watermark radioactivity [Sander et al., 2024] has been leveraged for data protection (RAG [Jovanović et al., 2025], contamination [Sander et al., 2025], copyright [Zhang et al., 2025]), which we extend in section 6 to the distillation of reasoning traces.

2.2 Distortion-Freeness and Choice of Baselines

A key distinction among watermarking methods is whether they are *distortion-free* (also called *unbiased*) [Hu et al., 2023]: at each generation step, the next-token distribution is preserved, i.e., $\mathbb{P}(\text{output}_t = v) = p_v^{(t)}$ for all v, where the probability is taken over key selection. Each token is drawn from the unmodified LLM distribution; only *diversity* across repeated generations is reduced. This per-token property does not make a whole answer distributed as the model's: when a context window repeats within an answer, the same pseudo-random values are reused. Answer-level distortion-

freeness requires deduplication at generation time, i.e. watermarking each context window at most once per key within an answer, with a fallback for repeated windows (Remark 1).

Green-red list methods [Kirchenbauer et al., 2023a] and low-entropy filtering methods (e.g., SWEET [Lee et al., 2023], which skips watermarking on low-entropy tokens) are *not* distortion-free: they shift the output distribution, degrading every generation. MorphMark [Wang et al., 2025] adaptively scales the green-red bias based on the natural green-list probability mass, reducing distortion in low-entropy contexts, but remains non-distortion-free since it still applies a logit bias. Semantic watermarks [Liu et al., 2023, Liu and Bu, 2024, Hou et al., 2023] require auxiliary semantic encoders, making them harder to deploy. Gumbel-max [Aaronson and Kirchner, 2023], DiPMark [Wu et al., 2023] (distortion-free green-red via pseudorandom permutations), δ-reweight and γ-reweight [Hu et al., 2023], SynthID-Text [Dathathri et al., 2024] (in Gemini), and WaterMax [Giboulot and Furon, 2024] (costly multiple generations per query) are distortion-free.

Recent large-scale evaluations [Fernandez et al., 2025] show that Gumbel-max and SynthID achieve the best detectability-quality Pareto frontier, strictly dominating DiPMark, green-red variants, and semantic watermarks; we therefore compare TextSeal against these two practical baselines. Because all three are distortion-free, we can fix the LLM, temperature, and top-p, and vary only their watermark-specific diversity parameter, isolating the watermark's effect.

2.3 Gumbel-Max Watermarking

TextSeal builds directly on the Gumbel-max scheme [Aaronson and Kirchner, 2023], which achieves the best detectability-quality trade-off among existing methods [Fernandez et al., 2023, 2025]. We briefly summarize it here; our improvements are presented in section 3.

Embedding. At each step t, the LLM predicts a distribution $\boldsymbol{p}^{(t)}$ over vocabulary $\mathcal{V}$. A secret key K and the k preceding tokens $\mathbf{w} = (x_{t-k}, \ldots, x_{t-1})$ seed a Pseudo-Random Function (PRF) that assigns $R_v = \text{PRF}(v, \mathbf{w}, K) \in [0, 1]$ to each candidate token v. The watermark selects:

$$x_t = \arg\max_{v \in \mathcal{V}} \; R_v^{1/p_v^{(t)}},$$

favoring tokens with high model probability *and* high PRF value, creating a statistical correlation with K that can later be detected (see subsection A.1 for PRF implementation details).

Detection. The detector re-computes PRF values using the secret key and preceding tokens, then checks whether the score exceeds what chance would produce. For tokens $x^{(1)}, \ldots, x^{(T)}$ with PRF values $R_t := R^{(t)}_{x^{(t)}}$, the detection score is $S_T = -\sum_{t=1}^{T} \ln(1-R_t)$. Watermarked tokens tend to have high R_t (since the selection rule favors them), inflating S_T; for unwatermarked text, the R_t values are uniform and independent of the selection, yielding a lower score on average.

Theoretical guarantees. The watermark is provably distortion-free at the token level: $\mathbb{P}(V^\star{=}v) = p_v$ where $V^\star = \arg\max_v R_v^{1/p_v}$ and $R_v \overset{\text{iid}}{\sim} \mathcal{U}[0, 1]$, i.e., the probability over the PRF randomness matches the LLM distribution (Proposition 1). Under $\mathcal{H}_0$, $S_T \sim \Gamma(T, 1)$, yielding exact p-values (Proposition 2). Under $\mathcal{H}_1$, $\mathbb{E}[S_T] \geq T + (\pi^2/6 - 1)H_T$ where H_T is the completion entropy (Proposition 3): detectability scales with entropy, motivating the entropy-weighted detection of subsection 3.2. Proofs of all statements in this section are given in Appendix B.

3 Method: TextSeal

TextSeal addresses three key limitations of the standard Gumbel-max watermark: its deterministic outputs, its suboptimal detection in mixed-entropy text, and the lack of localized detection capability. We describe each improvement below, and present an overview in Figure 2.

3.1 Dual-Key Routing for Diversity and Speculative Decoding Compatibility

Standard Gumbel-Max always produces the same output for a given prompt and secret key, triggering repetitive patterns and degenerate loops [Holtzman et al., 2019]. TextSeal breaks this determinism

by randomly alternating between two secret keys $k^{(1)}$ and $k^{(2)}$, restoring output diversity while preserving both detectability and the distortion-free guarantee of Gumbel-max.

Embedding. At each generation step t, one key is selected at random: $k^{(1)}$ with probability $1-\alpha$, or $k^{(2)}$ with probability α. The token is produced via Gumbel-Max using the selected key's PRF:

$$x_t = \arg\max_v \; R_v^{(k),\, 1/p_v^{(t)}}, \quad k \in \{k^{(1)}, k^{(2)}\} \tag{1}$$

The routing probability $\alpha \in [0, 0.5]$ controls the diversity-detectability trade-off. To keep whole answers distortion-free, TextSeal deduplicates at generation time: a repeated context window is watermarked with the key not yet used for it, and sampled without watermark once both keys are used (Remark 1); dual-key routing thus doubles the tolerance to repeated n-grams.

Detection. The detector does not know which key generated each token. To capture signal from both potential paths, we compute scores under both keys and aggregate them as a weighted sum:

$$s_i = (1-\alpha)\cdot s_i^{(1)} + \alpha \cdot s_i^{(2)}, \quad \text{where } s_i^{(j)} = -\ln(1-R_i^{(j)}) \tag{2}$$

We call this "early fusion": scores are combined at the token level, rather than running two independent tests and aggregating p-values ("late fusion"). Under $\mathcal{H}_0$, s_i is a weighted combination of independent exponentials with mean 1 and variance $\theta_R = \alpha^2 + (1-\alpha)^2$. The final p-value is computed using the unified framework in subsection 3.2. We show that this early-fusion approach is better than Fisher or Bonferroni aggregations in subsubsection C.1.1, and support it empirically in subsection 5.1. Dual-key routing also natively supports speculative decoding and Multi-Token Prediction (MTP), where draft-accepted tokens use $k^{(1)}$ and rejected tokens use $k^{(2)}$ (details in Appendix E.7).

3.2 Entropy-Weighted Detection

Entropy Weighting. When the next-token distribution has low entropy, the top token already has probability close to 1, so the choice is weakly influenced by the PRF value R_v and carries little watermark. TextSeal therefore weights each token's detection score by its local entropy H_i, so that

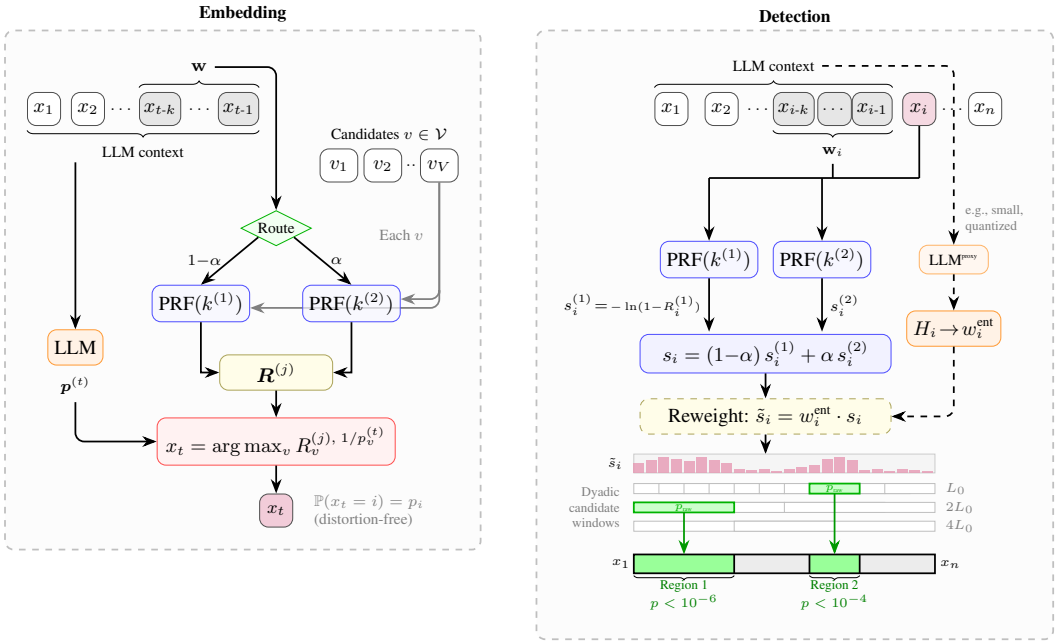


Figure 2: TextSeal overview. **Left (Embedding):** One of two keys is randomly selected (probability α for $k^{(2)}$, $1-\alpha$ for $k^{(1)}$), and the token is chosen via Gumbel-Max using the selected key's PRF (subsection 3.1). **Right (Detection):** Per-token scores under both keys are fused and entropy-weighted (subsection 3.2); a geometric cover search then localizes watermarked regions (subsection 3.3).

high-entropy positions contribute more to the final statistic. We estimate H_i with a single forward pass of an auxiliary model, e.g., a smaller or quantized model from the same family as the generator.

Formally, we assign each token-level score s_i an entropy weight w_i^{ent} and compute

$$S_{\text{combined}} = \sum_{i=1}^{n} w_i^{\text{ent}} \cdot s_i, \qquad \text{where } w_i^{\text{ent}} = 0.1 + 0.9 \cdot \frac{H_i - H_{\min}}{H_{\max} - H_{\min}}. \tag{3}$$

This attenuates low-entropy tokens instead of letting them dilute the score. Unlike prior entropy-filtering approaches [Lee et al., 2023] that threshold the entropies, our scheme is continuous: every token still contributes, but with strength matched to its expected usefulness.

Moment-Matched Gamma Approximation. S_{combined} is a weighted sum of independent, non-identical exponentials, whose exact CDF is numerically unstable. We fit $S_{\text{combined}} \sim \text{Gamma}(k_{\text{new}}, \theta_{\text{new}})$ by matching the first two moments under $\mathcal{H}_0$:

$$\theta_{\text{new}} = \theta_R \frac{\sum (w_i^{\text{ent}})^2}{\sum w_i^{\text{ent}}}, \qquad k_{\text{new}} = \frac{(\sum w_i^{\text{ent}})^2}{\theta_R \sum (w_i^{\text{ent}})^2} \tag{4}$$

The resulting p-value is:

$$p\text{-value}(S_{\text{combined}}) = 1 - F_\Gamma(S_{\text{combined}}; k_{\text{new}}, \theta_{\text{new}}) \tag{5}$$

We show in section 5.1 that this approximation is well-calibrated under $\mathcal{H}_0$, and in subsection 5.1 that it significantly outperforms unweighted detection. This framework handles dual-key routing (α) and entropy gating (w_i^{ent}) in a single frequentist test with calibrated p-values.

3.3 Multi-Region Localization and Adaptive Ensemble

When a document contains multiple scattered watermarked regions (e.g., distinct AI-generated paragraphs pasted into a human-written essay), evaluating a global score suffers from two critical flaws. First, the unwatermarked background tokens severely dilute the statistical signal. Second, it cannot identify the specific provenance of individual segments within the document. We do not claim a new primitive: the contribution is an ensemble with a family-wise error guarantee, built on a geometric cover and a Bonferroni correction, instantiated for the dual-key score, with its evaluation on mixed documents; any score with a known null, such as SynthID's, can use it (subsection G.7).

Geometric Cover Search & Greedy Extraction. Our goal is to extract a set of disjoint watermarked intervals $\{[a_1, b_1], \ldots, [a_y, b_y]\}$. A naive search over all $\mathcal{O}(n^2)$ possible start and end pairs is computationally prohibitive and incurs a massive multiple-testing penalty. Instead, we employ a geometric cover search, reducing the space to dyadic window lengths $L \in \{L_0, 2L_0, 4L_0, \ldots, 2^{\lfloor \log_2 n \rfloor}\}$, sliding each window across the text at half-length offsets. This yields a strictly bounded number of candidate windows, $M \approx 4n/L_{\min}$. The extraction proceeds in two stages: first, all M windows are ranked by their raw score sum (computed in $\mathcal{O}(1)$ per window via prefix sums); then, for the top candidates only, we compute the rigorous entropy-weighted Gamma p-value. The greedy extraction selects the window with the lowest p-value, flags it as watermarked, masks its tokens, and repeats on the residual text, aggregating intervals as long as their combined significance overcomes the multiple-testing tax. Full mathematical details and complexity analysis are in Appendix D.

Adaptive Ensemble Detection. Since the detector does not know *a priori* whether the document is fully watermarked, partially edited, or contains scattered insertions, we let it adaptively choose the strongest evidence. The ensemble selects the most significant among three strategies, applying a flat Bonferroni correction (k=3): (1) **Global** full-text test (no search penalty), (2) **Single-Best** window (penalized by $\log_{10} M$), and (3) **Multi-Region** aggregation over y zones (penalized by $\log_{10} \binom{M}{y} + \log_{10} Y_{\max}$):

$$\log_{10} p_{\text{final}} = \min\big(\log_{10} p_{\text{global}}, \log_{10} p_{\text{single}}, \log_{10} p_{\text{multi}}\big) + \log_{10} 3 \tag{6}$$

For largely unedited text, this defaults to the global test with a negligible $\log_{10}(3)$ penalty; under heavy dilution, localization rescues detection (analysis and proofs in Appendix D).

Table 1: Accuracy across benchmarks with and without *TextSeal*. No significant performance drop is observed, confirming that TextSeal preserves the capabilities of the underlying model.

		Math				Code			Knowledge				Common Sense					
Reasoning temp.	WM	AIME	MATH	GSM8K	Avg	HE	MBPP	Avg	MMLU	GPQA	SQA	Avg	HS	WG	ARC-E	ARC-C	Avg	Avg
0.6	✓	41.1	79.8	96.0	72.3	93.3	49.2	71.2	51.5	50.0	16.0	39.2	94.8	93.5	93.7	88.5	92.6	**70.6**
	✗	40.1	79.8	95.4	71.7	97.0	50.2	73.6	49.2	50.5	15.8	38.5	94.7	93.2	93.4	88.3	92.4	**70.6**
1.0	✓	37.1	77.9	96.1	70.4	94.5	48.5	71.5	48.9	45.5	13.7	36.0	94.6	93.8	92.8	86.0	91.8	**69.1**
	✗	35.8	78.4	96.1	70.1	98.2	49.3	73.7	46.4	42.9	15.5	34.9	94.6	93.6	93.8	86.6	92.2	**69.3**

4 Main Experiments

4.1 Experimental Setup

Models & Datasets Unless stated otherwise, we use Qwen 3.5-27B [Qwen Team, 2026] for generation, with $T = 0.8$, top-$p = 0.9$, and reasoning disabled. We evaluate on 1k prompts from the ELI5 dataset [Fan et al., 2019] (with 5 different seeds), truncating answers to 400 tokens. For entropy-aware detection (subsection 3.2), we by default use the lightweight Qwen 3.5-0.8B model.

We compare *TextSeal* (default $\alpha = 0.1$) to *Gumbel-Max* [Aaronson and Kirchner, 2023] and *SynthID-Text* [Dathathri et al., 2024] with depth 10. SynthID embeds a watermark via multi-layered tournament sampling, and proposes two detection methods: (i) a frequentist Z-test over the mean tournament score, and (ii) a Bayesian detector that estimates the posterior $\mathbb{P}(\text{watermarked} \mid \text{scores})$ via a logistic regression or MLP trained on a representative dataset. We use the frequentist Z-test, because the Bayesian detector provides no controlled false-positive rate, does not generalize across domains (its posteriors depend on the training distribution), and is incompatible with localized multi-window testing (full discussion in Appendix A.2). All methods use watermark context window k=3 and deduplicate at generation time, so that every answer is distortion-free (TextSeal per key, subsection 3.1; SynthID by its repeated-context masking); at detection time, each (context window, current token) pair is scored only once, to satisfy the independence assumption of the tests [Fernandez et al., 2023].

4.2 Detectability-Diversity Trade-off

A practical watermark must embed a robust signal without changing the output distribution. For Figure 1a, we vary for *TextSeal* the routing probability α from subsection 3.1 from 0.05 to 0.5 ($\alpha = 0$ is deterministic Gumbel-Max, the star at Self-BLEU 100). For *SynthID*, we vary the depth from 2 to 30, with Google DeepMind's released g-values and its own test. TextSeal uses entropy-weighted detection with the 0.8B proxy, and *TextSeal high* with the generator itself as entropy model, which adds about one order of magnitude at its default $\alpha = 0.1$ (median $-\log_{10} p$ of 6.7 against 5.8) at a higher detection cost. *TextSeal* detects more than SynthID at every diversity level, by about two orders of magnitude at its default $\alpha = 0.1$ (median $-\log_{10} p$ of 5.8 against 3.8). Entropy-weighted detection can also be applied to SynthID's mean g-value per token: SynthID then detects more in the most diverse regime (Self-BLEU below 33), while TextSeal stays ahead at lower diversity (subsection G.1) and remains simpler, with a single race and no added latency (section 5). On Llama-3.1-8B-Instruct, TextSeal also outperforms SynthID at every diversity level (subsection G.4); subsection G.6 gives the TPR at fixed FPR for every α, to choose α from a diversity requirement.

4.3 Performance on Benchmarks

We evaluate TextSeal on 12 benchmarks spanning math, code, general knowledge, and common sense (Table 1) with Qwen 3.5-27B at T=0.6 and T=1.0, each with top-p=0.95 and a 3,000-token reasoning budget. TextSeal preserves the average accuracy of the underlying model (70.6 vs. 70.6 at T=0.6). We observe a slight drop on code benchmarks (1–2 points on HumanEval/MBPP); inspecting the failures shows they all stem from formatting omissions (missing imports) rather than incorrect reasoning. Re-running with multiple random seeds and secret keys confirms that all differences remain within stochastic variance; see subsection E.2 for details.

Table 2: Human preference evaluation (majority vote aggregation over 3 annotators per sample). Net Win Rate: $(n_{\text{WM}} - n_{\text{Base}})/N$. p-value: two-sided binomial test on decisive samples against 50%. No test reaches significance after Bonferroni correction ($\alpha/6 = 0.008$).

Language	WM Wins	Base Wins	Ties	WM Preference Rate	p-value	Net Win Rate
English	124	146	1,730	45.9%	0.20	-1.1%
Arabic	201	181	618	52.6%	0.33	$+2.0\%$
Chinese	84	74	842	53.2%	0.47	$+1.0\%$
Hindi	91	89	820	50.6%	0.94	$+0.2\%$
Japanese	143	130	727	52.4%	0.47	$+1.3\%$
Overall	643	620	4,737	50.9%	0.54	$+0.4\%$

4.4 Human Evaluation of Imperceptibility

We assess whether the watermark introduces perceptible quality degradation through a human A/B preference study. Following the methodology of Dathathri et al. [2024], we generate paired responses to questions from ELI5 [Fan et al., 2019] (2,000 English samples) and CaLMQA [Arora et al., 2025] (1,000 each for Arabic, Chinese, Hindi, Japanese), totaling 6,000 question-answer pairs.

Each pair is evaluated by three annotators (via Appen) with qualifications requiring post-graduate education, native-level language fluency, and at least two years of experience. Annotators select among four options: *A is preferred*, *B is preferred*, *both equally good*, or *both equally bad*, without knowing which output is watermarked. We aggregate via majority vote, merging the two tie categories and defaulting split votes (one vote per category, no majority) to a tie.

Results. Table 2 reports preference rates after majority vote aggregation. We test whether the watermark win rate among decisive (non-tie) samples differs from 50% using a two-sided binomial test. No individual language reaches significance (all $p > 0.05$), and no test is significant after Bonferroni correction for the six comparisons ($\alpha/6 = 0.008$). The majority of samples (79%) result in ties, and inter-annotator agreement is high (88% of samples have at least 2/3 consensus on the 4-class scale). We also report the *net win rate*, defined as $(n_{\text{WM}} - n_{\text{Base}})/N$, with n_{WM} and n_{Base} the counts of samples where the watermarked or baseline response is preferred, and N the total number of samples. The overall net win rate is $+0.38\%$, indicating a negligible difference.

To rigorously establish imperceptibility rather than failing to detect a difference, we apply the Two One-Sided Tests (TOST) procedure [Schuirmann, 1987] for equivalence testing. We test $|P(\text{WM preferred}) - P(\text{Base preferred})| < \Delta$ over all N samples (including ties), which provides greater statistical power than restricting to decisive samples alone. With a smallest effect size of interest $\Delta = 5\%$, equivalence is established overall and for four of five individual languages ($p < 0.05$); Arabic marginally fails ($p = 0.06$) due to its wider confidence interval. The overall 90% CI is $[-0.6\%, +1.4\%] \subset [-5\%, +5\%]$ (full breakdowns and methodology in Appendix E.8).

4.5 Localization in Mixed Documents

In practice, watermarked text often forms only a fraction of a document. A global detector scoring the entire text faces two challenges: *dilution*, where unwatermarked tokens degrade the signal-to-noise ratio, and *fragmentation*, where watermarked content is scattered across non-contiguous regions. We evaluate TextSeal's adaptive ensemble (subsection 3.3) by embedding 400-token watermarked answers (T=1.0, top-p=0.95) into unwatermarked Wikipedia texts. Under **dilution** (K=1), a single contiguous block is placed in documents of increasing length up to 12,000 tokens; under **fragmentation** (K>1), the 400 tokens are split into $K \in \{1, 2, 3, 5\}$ equal fragments interleaved in an 8,000-token document. Figure 3 shows that global detection drops below p=0.01 around 4k tokens, while the adaptive ensemble stays at $-\log_{10} p > 4$ even at 12k tokens (3.3% watermarked) and handles up to K=3 scattered fragments before multiple-testing dominates. Localization is not specific to TextSeal: SynthID's score in the same ensemble also gains under dilution; TextSeal detects more single blocks in long documents, and SynthID more 5-fragment blocks (subsection G.7).

5 Ablations and Analyses

5.1 Diversity Strategies Comparison And FPRs

Strategy. We compare four strategies for restoring diversity in Gumbel-max watermarking, each with a tunable control parameter (full descriptions and proofs in Appendix C): **(1) Stochastic Mixing** mixes the PRF value with a Bernoulli coin (control: mixing rate a); **(2) Periodic Skip** disables watermarking at fixed intervals (control: skip rate α); **(3) Entropy-Normalized Skip** skips with probability τ uniformly across entropy regimes, preserving distortion-freeness (control: τ); **(4) Dual-Key Routing** (subsection 3.1) alternates between two secret keys (control: routing probability $\alpha \in [0, 0.5]$). We evaluate Qwen 3.5-27B on 1,000 ELI5 prompts (T=1.0, top-p=0.95). Figure 4 traces the Pareto frontier for each method. Entropy-weighted detection boosts every method by several orders of magnitude without changing diversity. Early-fusion dual-key routing outperforms Fisher-style aggregation at comparable Self-BLEU (proof in subsubsection C.1.1); stochastic mixing is consistently dominated. We adopt dual-key routing as our default diversity mechanism for its native compatibility with speculative decoding and multi-token prediction.

Robustness to post-editing. We attack TextSeal, SynthID, and Gumbel-Max outputs with a Llama-3.2-3B rephraser at four strengths, iterative rephrasing, back-translation, and random word insertions/deletions, and report TPR at fixed FPR from 10^{-2} to 10^{-4} (subsection G.3, threat model in subsection G.2). At matched diversity, TextSeal is ahead of SynthID on clean text at low FPR (72.8% vs 68.4% at 10^{-4}, entropy-weighted) and after word-level edits or back-translation (42.7% vs 34.3% and 36.0% vs 27.6% at 10^{-2}), and level under rephrasing; aggressive or repeated rephrasing

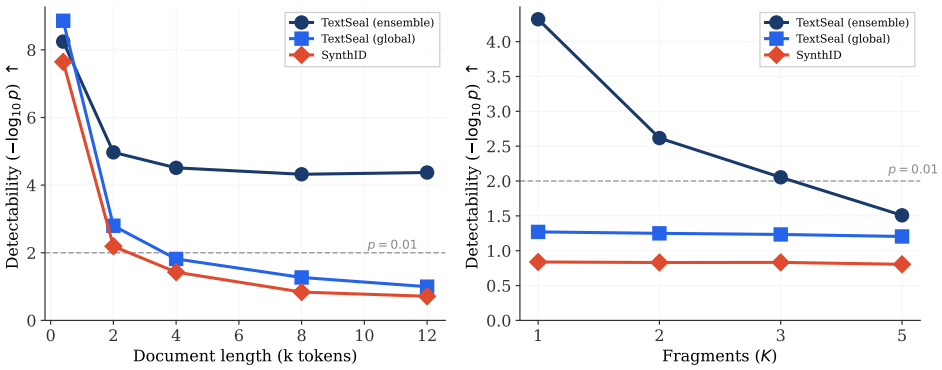


Figure 3: Localized detection in mixed documents containing 400 watermarked tokens. **(Left)** *Dilution*: Global detection drops below p=0.01 around 4k tokens; the adaptive ensemble maintains $-\log_{10} p > 4$ even at 12k tokens. **(Right)** *Fragmentation*: The ensemble detects the watermark for up to K=3 fragments in an 8,000-token document.

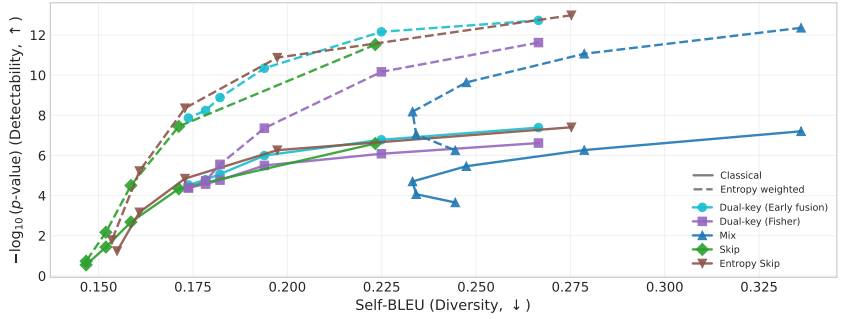


Figure 4: **Pareto frontier of diversity strategies.** Self-BLEU (lower is more diverse) vs. median $-\log_{10}(p)$ (higher is more detectable). Solid lines: classical detector; dashed: entropy weighting. Dual-key early fusion and entropy skip define the strongest Pareto frontier.

removes all watermarks we evaluate. subsection G.5 adds dual-key routing, early fusion, and entropy weighting one at a time, reporting Self-BLEU and TPR at fixed FPR.

False positive rate calibration. We validate FPR on 1M unwatermarked Wikipedia passages (256 tokens each). Under standard detection, all methods track the nominal FPR down to $\tau \approx 10^{-4}$, and so does TextSeal under entropy-weighted (0.8B) detection, confirming validity (Appendix E.4).

5.2 Embedding and Detection Efficiency

Generation overhead. TextSeal only evaluates the PRF on the top-p survivor tokens ($\sim$200 tokens), adding only $\sim$0.07 ms per token ($\leq$0.3% overhead). SynthID's tournament sampling costs $\sim$0.6 ms per token (0.4–1.1%). Both operate entirely on the logits with no model or KV-cache changes. The full per-model-size latency and memory breakdown is given in Appendix E.6.

Detection efficiency and proxy scaling (Figure 5). Standard unweighted detection requires no proxy model (0.007 ms/token) but yields a baseline median $-\log_{10} p$ of 4.8. Entropy weighting with the full 27B generation model boosts this to 8.2 but incurs 50.1 GB VRAM. The 4-bit 0.8B proxy reaches similar detectability (6.2) with 1.4 GB VRAM and 0.115 ms/token, and scales identically under attacks, which makes it our default entropy model for detection.

Appendix E provides additional analyses: multilingual QA (5 languages, 6,000 samples) and the compatibility of TextSeal with MTP speculative decoding (Appendix E.7).

6 Watermark Radioactivity: Detecting Distillation via Learnability

A watermark is *radioactive* [Sander et al., 2024] if, when a model is trained on watermarked data, it inherits a detectable token bias. This enables a powerful application beyond text provenance: detecting whether a competitor has distilled your model's outputs into their own.

Setup. We distill DeepSeek-R1-Distill-Qwen-14B [Guo et al., 2025] (teacher) into Qwen2.5-3B [Team, 2024] (student) on 5,000 curated problems from OpenR1-Math-220k [Hugging Face, 2025]. The teacher generates reasoning traces under four watermark schemes: Gumbel-Max [Aaronson and Kirchner, 2023], TextSeal (α=0.1), SynthID depth 10 [Dathathri et al., 2024], and an unwatermarked control (all with watermark windows of size 3). Following Muennighoff et al. [2025], we retain traces only if they close their `</think>` block, contain a `\boxed{}` answer (when required), match the reference solution under `math_verify`, and have no 100-character span recurring ≥ 3 times. We also remove problems that the base student already solves correctly, so the distillation set only includes traces that teach the student something new. We then apply LoRA fine-tuning on the remaining traces. To test watermark transfer, we use the open-model radioactivity test [Sander et al., 2024]. We feed each training trace into the student with *teacher forcing* (providing the ground-truth prefix at each position) and record its top-1 prediction. If the student has learned the watermark, its predictions should favor high-PRF tokens. We score each prediction with the watermark PRF and aggregate into

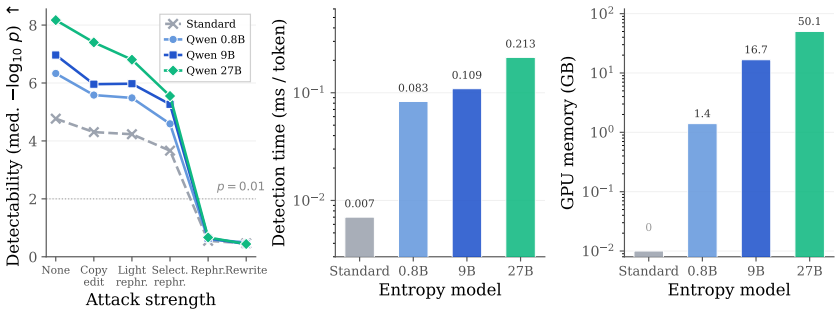


Figure 5: Entropy-weighted detection: performance vs. cost. The 4-bit 0.8B proxy captures most of the detectability boost of the 27B entropy model at 35$\times$ less GPU memory.

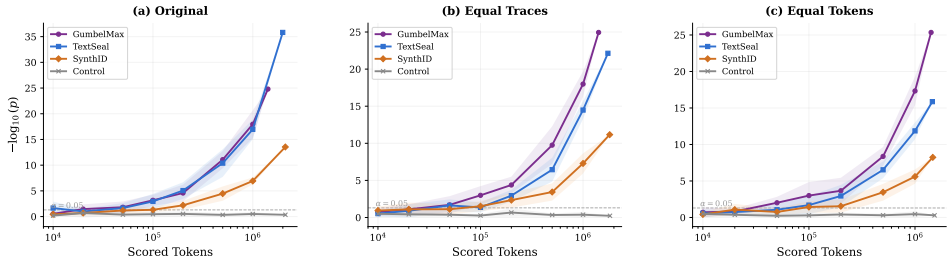


Figure 6: **Watermark radioactivity via distillation.** Detection power ($-\log_{10}(p)$) vs. number of unique scored tokens under three settings: original traces, equal-trace control (1,991 each), and equal-token control ($\sim$15.1M chars each). TextSeal yields the strongest signal in the original setup (more retained traces); Gumbel-Max dominates both controlled settings (stronger per-token signal).

a p-value, deduplicating at two levels to avoid spurious signal from context copying and to ensure independence across traces (subsection A.4), yielding $\sim$1.4–2.2M unique scored tokens per method. The full experimental setup and hyperparameters are given in subsection E.5.

Results. Figure 6 shows that all three watermarks reliably transfer through distillation, with detection power far exceeding the significance threshold. Under the original setup (each method uses all its retained traces), TextSeal achieves the strongest signal thanks to higher data volume. Once data volume is equalized (controlled conditions in Figure 6b,c), Gumbel-Max dominates, confirming a stronger per-token signal via deterministic argmax; TextSeal achieves comparable overall detectability by retaining more training data. All distilled students substantially improve over the base model (+10–15% on GSM8K), and distilling on watermarked traces does not lead to significant changes compared to the unwatermarked control. We further give entropy-weighting ablations, controlled comparisons, as well as the full benchmark accuracy numbers in subsection E.5.

7 Conclusion

We introduced TextSeal, a distortion-free watermark for LLMs that achieves state-of-the-art detectability through dual-key generation, entropy-weighted detection, and localized multi-region search. TextSeal outperforms SynthID on the diversity-detectability frontier, preserves model performance across 12 benchmarks, supports speculative decoding and MTP, and its watermark transfers through distillation, which enables radioactive tracing of unauthorized use.

Limitations. Like all distortion-free sampling watermarks, TextSeal trades diversity for detectability, not quality. While invisible for a single generation, this may affect workflows requiring diverse outputs (best-of-N reranking, creative brainstorming); modern RL-trained reasoning models already exhibit collapsed entropy, limiting the marginal loss. A capable LLM rephraser with a meaning-preserving prompt removes TextSeal's watermark, as it removes SynthID's and Gumbel-Max's (subsection G.3); our threat model assumes server-side keys and detector, without unlimited detector queries (subsection G.2). Several providers' watermarks can coexist without conflict, since each detection is keyed and calibrated separately for each provider (subsection G.8).

Note. Between the submission and the publication of this paper, we explored a potential public/private decoding scheme for TextSeal [Milis et al., 2026] and further improved the signal-to-noise ratio and detectability of the watermark with TextSeal2 [Sander et al., 2026].

Acknowledgements and Disclosures

We thank Surya Parimi, Christophe Ropers, Rashel Moritz, and Vanessa Stark for project support, and the annotators of the human evaluation. **Funding.** This work was funded by Meta; no third-party funding supported it. **Competing interests.** All authors are employed by Meta, which develops large language models and may use watermarking in its own products and services.

References

Scott Aaronson and Hendrik Kirchner. Watermarking GPT outputs, 2023. URL `https://scottaaronson.blog/?m=202302`.

Sahar Abdelnabi and Mario Fritz. Adversarial watermarking transformer: Towards tracing text provenance with data hiding. In *2021 IEEE Symposium on Security and Privacy (SP)*, pages 121–140. IEEE, 2021.

Shane Arora, Marzena Karpinska, Hung-Ting Chen, Ipsita Bhattacharjee, Mohit Iyyer, and Eunsol Choi. Calmqa: Exploring culturally specific long-form question answering across 23 languages. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11772–11817, 2025.

Guorui Bian, Michael McAleer, and Wing-Keung Wong. A trinomial test for paired data when there are many ties. *Mathematics and Computers in Simulation*, 81(6):1153–1160, 2011.

Igor A Bolshakov. A method of linguistic steganography based on collocationally-verified synonymy. In *International Workshop on Information Hiding*, pages 180–191. Springer, 2004.

Jack T Brassil, Steven Low, Nicholas F Maxemchuk, and Lawrence O'Gorman. Electronic marking and identification techniques to discourage document copying. *IEEE Journal on Selected Areas in Communications*, 13(8):1495–1504, 1995.

Ching-Yun Chang and Stephen Clark. Practical linguistic steganography using contextual synonym substitution and a novel vertex coding method. *Computational linguistics*, 40(2):403–448, 2014.

Mark Chapman, George I Davida, and Marc Rennhard. A practical and effective approach to large-scale automated linguistic steganography. In *International Conference on Information Security*, pages 156–165. Springer, 2001.

Miranda Christ, Sam Gunn, and Or Zamir. Undetectable watermarks for language models. *Cryptology ePrint Archive*, 2023.

Sumanth Dathathri, Abigail See, Sumedh Ghaisas, Po-Sen Huang, Rob McAdam, Johannes Welbl, Vandana Bachani, Alex Kaskasoli, Robert Stanforth, Tatiana Matejovicova, et al. Scalable watermarking for identifying large language model outputs. *Nature*, 634(8035):818–823, 2024.

European Parliament and Council of the European Union. Regulation (EU) 2024/1689 laying down harmonised rules on artificial intelligence (AI Act), 2024. URL `https://eur-lex.europa.eu/eli/reg/2024/1689/oj`.

Angela Fan, Yacine Jernite, Ethan Perez, David Grangier, Jason Weston, and Michael Auli. ELI5: Long form question answering. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3558–3567. Association for Computational Linguistics, 2019.

Pierre Fernandez, Antoine Chaffin, Karim Tit, Vivien Chappelier, and Teddy Furon. Three bricks to consolidate watermarks for large language models. *2023 IEEE International Workshop on Information Forensics and Security (WIFS)*, 2023.

Pierre Fernandez, Tom Sander, Hady Elsahar, Hongyan Chang, Tomáš Souček, Valeriu Lacatusu, Tuan Tran, Sylvestre-Alvise Rebuffi, and Alexandre Mourachko. How good is post-hoc watermarking with language model rephrasing? *arXiv preprint arXiv:2512.16904*, 2025.

Yu Fu, Deyi Xiong, and Yue Dong. Watermarking conditional text generation for ai detection: Unveiling challenges and a semantic-aware watermark remedy. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 18003–18011, 2024.

Eva Giboulot and Teddy Furon. Watermax: breaking the llm watermark detectability-robustness-quality trade-off. *arXiv preprint arXiv:2403.04808*, 2024.

Fabian Gloeckle, Badr Youbi Idrissi, Baptiste Rozière, David Lopez-Paz, and Gabriel Synnaeve. Better & faster large language models via multi-token prediction. *arXiv preprint arXiv:2404.19737*, 2024.

Chenchen Gu, Xiang Lisa Li, Percy Liang, and Tatsunori Hashimoto. On the learnability of watermarks for language models. *arXiv preprint arXiv:2312.04469*, 2023.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. The curious case of neural text degeneration. *arXiv preprint arXiv:1904.09751*, 2019.

Abe Bohan Hou, Jingyu Zhang, Tianxing He, Yichen Wang, Yung-Sung Chuang, Hongwei Wang, Lingfeng Shen, Benjamin Van Durme, Daniel Khashabi, and Yulia Tsvetkov. Semstamp: A semantic watermark with paraphrastic robustness for text generation. *arXiv preprint arXiv:2310.03991*, 2023.

Abe Bohan Hou, Jingyu Zhang, Yichen Wang, Daniel Khashabi, and Tianxing He. k-semstamp: A clustering-based semantic watermark for detection of machine-generated text. *arXiv preprint arXiv:2402.11399*, 2024.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Liang Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *Iclr*, 1(2):3, 2022.

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

Baihe Huang, Banghua Zhu, Hanlin Zhu, Jason D. Lee, Jiantao Jiao, and Michael I. Jordan. Towards optimal statistical watermarking, 2023.

Hugging Face. Open r1: A fully open reproduction of deepseek-r1, January 2025. URL `https://github.com/huggingface/open-r1`.

Nikola Jovanović, Robin Staab, Maximilian Baader, and Martin Vechev. Ward: Provable rag dataset inference via llm watermarks. *ICLR*, 2025.

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

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

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

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the 29th symposium on operating systems principles*, pages 611–626, 2023.

Gregory Kang Ruey Lau, Xinyuan Niu, Hieu Dao, Jiangwei Chen, Chuan-Sheng Foo, and Bryan Kian Hsiang Low. Waterfall: Framework for robust and scalable text watermarking. In *ICML 2024 Workshop on Foundation Models in the Wild*, 2024.

Taehyun Lee, Seokhee Hong, Jaewoo Ahn, Ilgee Hong, Hwaran Lee, Sangdoo Yun, Jamin Shin, and Gunhee Kim. Who wrote this code? watermarking for code generation. *arXiv preprint arXiv:2305.15060*, 2023.

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

Yepeng Liu and Yuheng Bu. Adaptive text watermark for large language models. *arXiv preprint arXiv:2401.13927*, 2024.

Quinn McNemar. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, 12(2):153–157, 1947.

Hasan Mesut Meral, Bülent Sankur, A Sumru Özsoy, Tunga Güngör, and Emre Sevinç. Natural language watermarking via morphosyntactic alterations. *Computer Speech & Language*, 23(1): 107–125, 2009.

Georgios Milis, Tom Sander, Tomáš Souček, Heng Huang, and Pierre Fernandez. Could llm watermark detection be public? *arXiv preprint arXiv:2610.12106*, 2026.

Niklas Muennighoff, Zitong Yang, Weijia Shi, Xiang Lisa Li, Li Fei-Fei, Hannaneh Hajishirzi, Luke Zettlemoyer, Percy Liang, Emmanuel Candès, and Tatsunori B Hashimoto. s1: Simple test-time scaling. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 20286–20332, 2025.

Leyi Pan, Aiwei Liu, Zhiwei He, Zitian Gao, Xuandong Zhao, Yijian Lu, Binglin Zhou, Shuliang Liu, Xuming Hu, Lijie Wen, et al. Markllm: An open-source toolkit for llm watermarking. *arXiv preprint arXiv:2405.10051*, 2024.

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

Jipeng Qiang, Shiyu Zhu, Yun Li, Yi Zhu, Yunhao Yuan, and Xindong Wu. Natural language watermarking via paraphraser-based lexical substitution. *Artificial Intelligence*, 317:103859, 2023.

Wenjie Qu, Dong Yin, Zixin He, Wei Zou, Tianyang Tao, Jinyuan Jia, and Jiaheng Zhang. Provably robust multi-bit watermarking for ai-generated text via error correction code. *arXiv preprint arXiv:2401.16820*, 2024.

Qwen Team. Qwen3.5: Towards Native Multimodal Agents. `https://qwen.ai/blog?id=qwen3.5`, 2026. Alibaba Cloud.

Alexandre Sablayrolles, Matthijs Douze, Cordelia Schmid, and Hervé Jégou. Radioactive data: tracing through training. In *International Conference on Machine Learning*, pages 8326–8335. PMLR, 2020.

Tom Sander, Pierre Fernandez, Alain Durmus, Matthijs Douze, and Teddy Furon. Watermarking makes language models radioactive. *NeurIPS*, 2024.

Tom Sander, Pierre Fernandez, Saeed Mahloujifar, Alain Durmus, and Chuan Guo. Detecting benchmark contamination through watermarking. *arXiv preprint arXiv:2502.17259*, 2025.

Tom Sander, Pierre Fernandez, Itai Gat, Haiming Bao, Jiaxin Shi, Jason Park, Valeriu Lacatusu, Sylvestre-Alvise Rebuffi, Hongyan Chang, Hady Elsahar, et al. Textseal2: Distortion-free llm watermarking with copula mixing and memoryless races. *arXiv preprint arXiv:2610.40613*, 2026.

Donald J Schuirmann. A comparison of the two one-sided tests procedure and the power approach for assessing the equivalence of average bioavailability. *Journal of Pharmacokinetics and Biopharmaceutics*, 15(6):657–680, 1987.

M Hassan Shirali-Shahreza and Mohammad Shirali-Shahreza. A new synonym text steganography. In *2008 international conference on intelligent information hiding and multimedia signal processing*, pages 1524–1526. IEEE, 2008.

Qwen Team. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024. URL `https://arxiv.org/abs/2412.15115`.

Mercan Topkara, Cuneyt M Taskiran, and Edward J Delp III. Natural language watermarking. In *Security, Steganography, and Watermarking of Multimedia Contents VII*, volume 5681, pages 441–452. SPIE, 2005.

Mercan Topkara, Giuseppe Riccardi, Dilek Hakkani-Tür, and Mikhail J Atallah. Natural language watermarking: Challenges in building a practical system. In *Security, Steganography, and Watermarking of Multimedia Contents VIII*, volume 6072, pages 106–117. SPIE, 2006a.

Mercan Topkara, Umut Topkara, and Mikhail J Atallah. Words are not enough: sentence level natural language watermarking. In *Proceedings of the 4th ACM international workshop on Contents protection and security*, pages 37–46, 2006b.

Umut Topkara, Mercan Topkara, and Mikhail J Atallah. The hiding virtues of ambiguity: quantifiably resilient watermarking of natural language text through synonym substitutions. In *Proceedings of the 8th workshop on Multimedia and security*, pages 164–174, 2006c.

Honai Ueoka, Yugo Murawaki, and Sadao Kurohashi. Frustratingly easy edit-based linguistic steganography with a masked language model. *arXiv preprint arXiv:2104.09833*, 2021.

Ashish Venugopal, Jakob Uszkoreit, David Talbot, Franz Josef Och, and Juri Ganitkevitch. Watermarking the outputs of structured prediction with an application in statistical machine translation. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 1363–1372, 2011.

Zongqi Wang, Tianle Gu, Baoyuan Wu, and Yujiu Yang. Morphmark: Flexible adaptive watermarking for large language models. *arXiv preprint arXiv:2505.11541*, 2025.

Alex Wilson and Andrew D Ker. Avoiding detection on twitter: embedding strategies for linguistic steganography. *Electronic Imaging*, 28:1–9, 2016.

Keith Winstein. Lexical steganography through adaptive modulation of the word choice hash. *Unpublished. http://www. imsa. edu/˜ keithw/tlex*, 1998.

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

Lingyun Xiang, Xinhui Wang, Chunfang Yang, and Peng Liu. A novel linguistic steganography based on synonym run-length encoding. *IEICE transactions on Information and Systems*, 100(2): 313–322, 2017.

Xiaojun Xu, Jinghan Jia, Yuanshun Yao, Yang Liu, and Hang Li. Robust multi-bit text watermark with llm-based paraphrasers. *arXiv preprint arXiv:2412.03123*, 2024.

KiYoon Yoo, Wonhyuk Ahn, Jiho Jang, and Nojun Kwak. Robust multi-bit natural language watermarking through invariant features. *arXiv preprint arXiv:2305.01904*, 2023.

KiYoon Yoo, Wonhyuk Ahn, and Nojun Kwak. 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, 2024.

Jingqi Zhang, Ruibo Chen, Yingqing Yang, Peihua Mai, Heng Huang, and Yan Pang. Leave no trace: Black-box detection of copyrighted dataset usage in large language models via watermarking. *arXiv preprint arXiv:2510.02962*, 2025.

Ruisi Zhang, Shehzeen Samarah Hussain, Paarth Neekhara, and Farinaz Koushanfar. {REMARK-LLM}: A robust and efficient watermarking framework for generative large language models. In *33rd USENIX Security Symposium (USENIX Security 24)*, pages 1813–1830, 2024.

A More Technical Details on the Methods

A.1 Hash Function Implementation

The PRF takes as input the candidate token x, a context window $\mathbf{w} = (w_1, \ldots, w_k)$ of k token IDs, and the secret key K (all of them are integers), and outputs a random integer in $[0, M)$.

We compute the hash as follows:

$$h'(x, \mathbf{w}, K) = \left(p_2 \cdot x + \sum_{i=1}^{k} w_i \cdot q_i + p_3 \cdot K \right) \cdot p_4, \tag{7}$$

$$h(x, \mathbf{w}, K) = \text{XORShift}(h'(x, \mathbf{w}, K)) \mod M, \tag{8}$$

where $q_1, \ldots, q_k$ are distinct large primes (so that different orderings of the same tokens give different values), and p_2, p_3, p_4 are additional primes. h' undergoes XOR-shift for better bit dispersion: $h = (h' \cdot p_{\text{mix}}) \oplus ((h' \cdot p_{\text{mix}}) \gg s)$, where p_{mix} is a mixing prime and s is a shift constant.

Finally, we normalize to obtain the uniform pseudo-random value:

$$u = \frac{h(x, \mathbf{w}, K)}{M} \in [0, 1]$$

A.2 Details on SynthID-Text Evaluation

In our experiments, we evaluate *SynthID-Text* [Dathathri et al., 2024] as the state-of-the-art generation-time, distortion free and non deterministic watermark. While traditional methods (such as Gumbel-Max or Soft Red List) apply a single, global shift to the logit distribution, SynthID-Text embeds its signal through a multi-layered *Tournament sampling* mechanism.

Tournament Generation. At each generation step t, the method seeds a pseudo-random function using the preceding k tokens (the context window). Using this seed, the vocabulary $\mathcal{V}$ is pseudo-randomly partitioned into a tournament structure with m layers. At each layer $l \in \{1, \ldots, m\}$, a pseudo-random g-value $g_{t,l}$ is computed. Instead of a single binary split, SynthID-Text iteratively reshapes the target LLM's probability distribution across these m layers. Tokens that consistently win their tournament matches (i.e., those assigned high g-values across multiple layers) see their sampling probabilities exponentially increased. By distributing the watermark across multiple layers, SynthID-Text preserves text quality while embedding a robust signal. The diversity–detectability frontier (Figure 1a) and Appendix G use the g-values of Google DeepMind's released implementation (Bernoulli, one key per layer, hashed with the context window and the token). The other SynthID experiments (radioactivity, efficiency, and Figure 1b) use our earlier implementation, which derives layer l's g-values from token $v + l$ under one key and yields weaker detection.

Why we avoid SynthID's Bayesian detector. The original SynthID-Text framework proposes a Bayesian neural network (logistic regression or MLP) trained on a representative dataset to estimate posterior probabilities $P(w|g)$. We avoid this approach for several reasons.

(i) No false-positive-rate guarantee. A Bayesian posterior score has no frequentist calibration: there is no principled way to set a decision threshold that guarantees, for example, at most one false accusation in 10^4 documents. This is essential for any legal or regulatory use of watermark detection, where a false positive can constitute a wrongful accusation of AI generation.

(ii) Distribution dependence. The trained classifier learns the joint distribution of token scores and watermark presence from its training corpus. Deploying on a different model, domain, language, or decoding strategy invalidates these learned posteriors; we observed that the Bayesian detector degrades sharply on out-of-domain text, requiring retraining for every new deployment setting.

(iii) Incompatibility with multiple-testing correction. Localized detection (subsection 3.3) requires evaluating thousands of candidate windows and a Bonferroni correction to control the family-wise error rate. This demands a calibrated null distribution per window, which a learned classifier cannot provide. Bayesian scores are not p-values and cannot be validly combined or corrected.

(iv) Opacity and reproducibility. A learned classifier is a black box whose decision boundary cannot be formally audited. For provenance claims that may carry legal weight, a closed-form statistical test

with an analytically derived null distribution is far more defensible. Moreover, the Bayesian detector is not open-sourced, and despite following the specification in the original supplementary material, we were unable to reproduce comparable results, making fair comparison infeasible.

Our frequentist alternative. To ensure a fair, threshold-independent comparison, we implement a mathematically rigorous frequentist detection pipeline. At detection time, for a given token x_t and its context, we reconstruct the PRF-seeded tournament and extract the sequence of m layer-wise g-values $g_{t,1}, \ldots, g_{t,m}$. Because earlier layers in the tournament contribute more watermarking evidence than later layers, we compute a *Weighted Mean Score* for the token:

$$s_t = \sum_{l=1}^{m} \alpha_l g_{t,l} \tag{9}$$

where $\alpha_1 \geq \cdots \geq \alpha_m \geq 0$ are linearly decaying weights. Over a sequence of N valid tokens, we sum the scores to obtain a test statistic $S = \sum_{t=1}^{N} s_t$. Under the null hypothesis $\mathcal{H}_0$ (unwatermarked text), the g-values follow the unwatermarked uniform or Bernoulli distribution. We analytically compute the mean μ_0 and variance σ_0^2 of the weighted sum under $\mathcal{H}_0$. We then compute a final Z-score for the sequence:

$$Z_{\text{SynthID}} = \frac{S - N\mu_0}{\sigma_0 \sqrt{N}} \tag{10}$$

The significance is given by the standard normal survival function $p = 1 - \Phi(Z_{\text{SynthID}})$.

A.3 Other Watermark Schemes

We describe below the other watermarking schemes referenced in this work.

Green-list/Red-list. Kirchenbauer et al. [2023a] modify the logit vector based on the watermark context window and secret key K. A token v is "green" if $\text{PRF}(v, \mathbf{w}, K) < \gamma$ (typically γ=0.5), and its logit is incremented by δ: $\tilde{\ell}_v = \ell_v + \delta$ for green tokens, $\tilde{\ell}_v = \ell_v$ otherwise. Detection counts green tokens with a binomial test. It is *not* distortion-free: the additive bias alters every generation.

MorphMark. Wang et al. [2025] adaptively adjust watermark strength based on context. Let $P_G = \sum_{v \in \text{GreenList}} p_v$ be the total probability mass on green tokens. If $P_G \leq p_0$ (a threshold), no watermark is applied; otherwise, probabilities are rescaled with an adaptive boost factor $r = \min(\kappa P_G, 1)$. This reduces distortion compared to vanilla green-red but is still *not* distortion-free.

DiPMark. Wu et al. [2023] introduce a distortion-free variant of green-red watermarks using a pseudorandom permutation π (seeded by context and K) to reorder tokens before applying a bias. The bias preserves the original distribution in expectation over the randomness of π.

WaterMax. Giboulot and Furon [2024] generate several candidate chunks from the original LLM distribution and select outputs with the highest watermark score. This is distortion-free by construction but requires multiple generations per query, making it impractical for production.

A.4 Radioactivity Test Protocol

We detail the radioactivity test methodology from Sander et al. [2024, 2025].

Teacher-forcing setup. We feed the watermarked training traces into the suspect (student) model using teacher forcing: at each position t, the model receives the ground-truth prefix $x_{<t}$ from the watermarked trace and produces a prediction. Let $\hat{x}_t = \arg\max_{v \in \mathcal{V}} P_\theta(v \mid x_{<t})$ denote the student's top-1 prediction at step t. The key insight is that teacher forcing isolates the model's learned token preferences from confounding factors such as sampling noise, requiring only a single forward pass over the existing traces rather than expensive autoregressive generation.

Test statistic. We score each top-1 prediction using the watermark's PRF: $R_t = \text{PRF}(\hat{x}_t, \mathbf{w}_t, K)$, where $\mathbf{w}_t$ is the context window of teacher tokens preceding position t. If the student internalized the watermark bias during training, its top-1 predictions $\hat{x}_t$ will be systematically skewed toward high-PRF tokens, producing a statistically significant test statistic.

Deduplication. We deduplicate at two levels, each for a different reason: (i) *within each trace*, each context window $\mathbf{w}_t$ is scored once. This is needed because the teacher's watermarked tokens appear in the student's context during teacher forcing: if an n-gram that was biased toward high-PRF values appears repeatedly, the student might copy it from context rather than predicting it from internalized preferences, creating a false signal [Sander et al., 2024]; (ii) *across traces*, all (context window, predicted token) pairs are pooled and deduplicated, because the PRF is deterministic and shared (context, token) tuples across different training examples would yield identical scores, violating independence [Fernandez et al., 2023]. After deduplication, under $\mathcal{H}_0$ (the student is unaware of K), the scores are independent and follow their null distribution, enabling exact p-values.

B Gumbel-max proofs

The following results were presented by Aaronson and Kirchner [2023] and formalized by Fernandez et al. [2023]. Some elements of these proofs are used later, so we restate them here. An overview of the standard Gumbel-max generation scheme is presented in Figure 7.

Proposition 1 (Sampling probability)**.** *Consider a discrete distribution* $\boldsymbol{p} = (p_1, \ldots, p_V)$ *and* $V = |\mathcal{V}|$ *random variables* $\boldsymbol{R} = (R_1, \ldots, R_V)$ *s.t.* $R_v \overset{iid}{\sim} \mathcal{U}_{[0,1]}$. *Let* $V^\star = \arg\max_v R_v^{1/p_v}$. *Then:* $\mathbb{P}(V^\star = v) = p_v$.

Proof of Proposition 1. For any $v \in \mathcal{V}$, $R_v \overset{iid}{\sim} \mathcal{U}_{[0,1]}$ so, $-\ln(R_v)$ follows an exponential distribution $\mathcal{E}(1)$. Let $Z_v := -\frac{1}{p_v}\ln(R_v)$. By construction, $Z_v \sim \mathcal{E}(p_v)$, with density $f_{Z_v}(z) = p_v e^{-p_v \cdot z}$. We now have:

$$V^\star = \arg\max_v R_v^{\frac{1}{p_v}} = \arg\min_v Z_v. \tag{11}$$

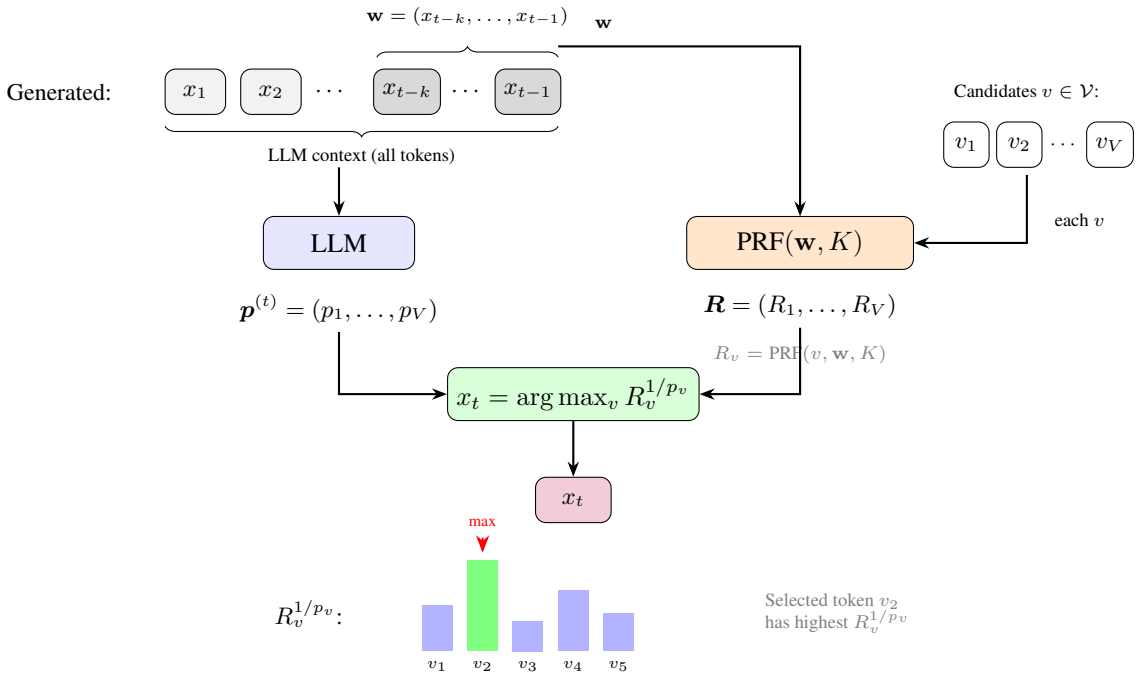


Figure 7: Standard Gumbel-Max watermarking (see section 2). The LLM uses all previous tokens to predict probabilities, while the PRF uses only the last k tokens (watermark context $\mathbf{w}$) to generate pseudo-random values R_v for each candidate token v. The token maximizing R_v^{1/p_v} is selected.

A well known result about exponential laws is that:

$$\underline{Z} = \min_v Z_v \sim \mathcal{E}\left(\sum_v p_v\right) = \mathcal{E}(1), \tag{12}$$

$$\mathbb{P}(V^\star = v) = \frac{p_v}{\sum_j p_j} = p_v. \tag{13}$$

This shows that for a given secret vector $\boldsymbol{r}$, the watermarking chooses a word which may be unlikely (low probability $p_{V^\star}$). Yet, on expectation over the secret keys, i.e., over r.v. $\boldsymbol{R} = (R_1, \ldots, R_V)$, the distribution of the chosen token follows the distribution given by the LLM. □

Corollary 1. *Conditionally on $V^\star = v$, $R_{V^\star} \sim$ Beta$(1/p_v, 1)$.*

Proof of Corollary 1. From the proof above, $\underline{Z} = \min_v Z_v \sim \mathcal{E}(1)$ and $V^\star = \arg\min_v Z_v$. A standard property of competing exponentials is that the identity of the winner is independent of the winning time: $V^\star \perp \underline{Z}$. Conditioning on $V^\star = v$, we therefore still have $\underline{Z} \sim \mathcal{E}(1)$, and:

$$\underline{Z} = Z_v = -\frac{1}{p_v}\ln(R_v) \sim \mathcal{E}(1), \tag{14}$$

so $R_v = e^{-p_v E}$ with $E \sim \mathcal{E}(1)$ and density $f_{R_v}(r) = \frac{r^{1/p_v - 1}}{p_v}$, i.e. $R_v \mid V^\star = v \sim$ Beta$(1/p_v, 1)$. □

Remark 1 (Repeated n-grams and strict distortion-freeness)**.** The distortion-free property (Proposition 1) holds assuming each context window $\mathbf{w}$ appears at most once per key during generation. When the same k-gram context repeats with the same key, the PRF produces identical pseudo-random values, making the selection deterministic rather than stochastic, introducing distortion.

To guarantee strict distortion-freeness for every token within a single generation, one must maintain a set $\mathcal{S}$ of seen context windows *for that generation* and apply this protocol [Dathathri et al., 2024]:

1. **First occurrence** of a context window: watermark with a randomly chosen key (as in standard dual-key routing) and record the window and the key used in $\mathcal{S}$.
2. **Second occurrence**: the first key is exhausted, so watermark with the other key and record it in $\mathcal{S}$.
3. **Third occurrence and beyond**: both keys are exhausted; use unwatermarked sampling.

Dual-key routing thus doubles the number of watermarkable slots per context window versus single-key schemes before any fallback is needed. The memory overhead of $\mathcal{S}$ is negligible (one hash per generated token), but requires stateful tracking within each generation. The diversity–detectability frontier (Figure 1a) and Appendix G enforce this protocol for TextSeal and the equivalent repeated-context masking for SynthID-Text; the other evaluations do not (except subsection E.2), as repeated k-grams are rare in practice with $k \geq 3$, so the protocol rarely triggers.

Proposition 2 (p-value under $\mathcal{H}_0$)**.** *Under $\mathcal{H}_0$ (text not watermarked), the score S_T follows a $\Gamma(T, 1)$ distribution. The p-value associated to a score s is:*

$$p\text{-value}(s) = \mathbb{P}(S_T > s \mid \mathcal{H}_0) = \frac{\Gamma(T, s)}{\Gamma(T)}, \tag{15}$$

where $\Gamma(T, s)$ is the upper incomplete gamma function.

This provides an *exact* false positive rate: for any significance level α, we can compute a threshold τ such that the probability of wrongly flagging unwatermarked text as watermarked is exactly α.

Proposition 3 (Expected score under $\mathcal{H}_1$)**.** *Under $\mathcal{H}_1$ (text is watermarked), $\mathbb{E}(S_T) \geq T + \left(\frac{\pi^2}{6} - 1\right) H_T$, where $H_T = -\sum_{t=1}^{T} p_t \ln(p_t)$ is the entropy of the completion.*

Proof of Proposition 3. From the corollary above, $R_t = \exp(-p_t E)$ with $E \sim \mathcal{E}(1)$, so:

$$\begin{aligned}
\mathbb{E}(S) &= -\mathbb{E}\left[\sum_{t=1}^{T} \ln(1 - \exp(-p_t E))\right] \\
&= -\sum_{t=1}^{T} \int_0^\infty \ln(1 - e^{-p_t x}) e^{-x} dx \\
&= -\sum_{t=1}^{T} \int_0^1 \frac{1}{p_t} r^{1/p_t - 1} (-\ln(1-r)) dr \\
&\qquad \text{(by change of variable } x = -1/p_t \ln(r) \text{)}
\end{aligned}$$

Then, using integration by parts with $u = 1 - r^{1/p_t}$ and $v = \ln(1-r)$, the integral becomes:

$$-\int_0^1 \frac{1}{p_t} r^{1/p_t - 1} \ln(1-r) dr = \int_0^1 \frac{1 - r^{1/p_t}}{1-r} dr = \mathcal{H}_{1/p_t}$$

where $\mathcal{H}_z$ is the z-th harmonic number also defined as $\mathcal{H}_z = \sum_{n=1}^{\infty} \frac{1}{n} - \frac{1}{n+z}$. Therefore, we have:

$$\begin{aligned}
-\int_0^1 \frac{1}{p_t} r^{1/p_t - 1} \ln(1-r) dr &= \sum_{n=1}^{\infty} \frac{1}{n} - \frac{1}{n + 1/p_t} \\
&= 1 + \sum_{n=1}^{\infty} \frac{1}{n+1} - \frac{1}{n + 1/p_t}.
\end{aligned}$$

Now, $\forall n \in \mathbb{N}^\star$, we have:

$$\begin{aligned}
(n+1)^2 \left(\frac{1}{n+1} - \frac{1}{n+1/p_t}\right) &= \frac{(n+1)(n+1/p_t) - (n+1)^2}{n+1/p_t} \\
&= \frac{1+n}{1/p_t + n} (1/p_t - 1) \\
&\geq -\frac{1+n}{1/p_t + n} \ln(p_t) \\
&\geq -p_t \ln(p_t).
\end{aligned}$$

Therefore, by summing over all $t \in [1, T]$,

$$\begin{aligned}
\mathbb{E}(S) &\geq T + \left(\sum_{n=1}^{\infty} \frac{1}{(n+1)^2}\right) \left(\sum_{t=1}^{T} -p_t \ln(p_t)\right) \\
&= T + \left(\frac{\pi^2}{6} - 1\right) H_T.
\end{aligned}$$

□

C Proofs on Diversity Schemes for Gumbel Max

We derive bounds on the expected detection score $\mathbb{E}[S_T]$ under $\mathcal{H}_1$ for each diversity strategy described in subsection 5.1 and illustrated in Figure 8. All bounds decompose as the standard Gumbel bound (Proposition 3) plus a correction term capturing the cost of the diversity mechanism.

Recall that under the standard Gumbel scheme, $R_t \sim \text{Beta}(1/p_t, 1)$ and the expected per-token score is $\mathbb{E}[s_t] = \mathcal{H}_{1/p_t}$, leading to $\mathbb{E}[S_T] \geq T + (\frac{\pi^2}{6} - 1) H_T$. In each case below, some tokens are either unwatermarked or have a modified distribution of R_t. For unwatermarked tokens, $R_t \sim \mathcal{U}[0,1]$ and $\mathbb{E}[s_t] = 1$.

C.1 Dual-Key Routing

Dual-key routing (subsection 3.1) maintains two secret keys $k^{(1)}$ and $k^{(2)}$. At each step, key $k^{(1)}$ is selected with probability $1-\alpha$ and $k^{(2)}$ with probability α. The token is sampled by Gumbel-Max with the selected key. Detection fuses both keys' scores: $s_i = (1-\alpha) \cdot s_i^{(1)} + \alpha \cdot s_i^{(2)}$.

Proposition 4 (Bound on expected score under dual-key routing, single-key detection)**.** *Under dual-key routing with parameter $\alpha \in [0,1]$ (key $k^{(1)}$ selected with probability $1-\alpha$, key $k^{(2)}$ with probability α), detection using a single key $k^{(1)}$ yields:*

$$\mathbb{E}[S_T^{(1)}] \geq T + (1-\alpha)\left(\frac{\pi^2}{6} - 1\right) H_T \tag{16}$$

Proof. At each step t, key $k^{(1)}$ is selected with probability $1-\alpha$ and key $k^{(2)}$ with probability α. For detection using key $k^{(1)}$:

- With probability $1-\alpha$: the PRF value $R_t^{(1)}$ is the one that was used for generation, so $R_t^{(1)} \sim \text{Beta}(1/p_t, 1)$ (Corollary 1) and $\mathbb{E}[s_t^{(1)}] = \mathcal{H}_{1/p_t}$.
- With probability α: the token was generated using key $k^{(2)}$, so $R_t^{(1)}$ is independent of the generation process. It is effectively uniform and $\mathbb{E}[s_t^{(1)}] = 1$.

Summing over T tokens:

$$\mathbb{E}[S_T^{(1)}] = (1-\alpha) \sum_{t=1}^{T} \mathcal{H}_{1/p_t} + \alpha T$$

Applying the standard bound (Proposition 3) to $\sum_t \mathcal{H}_{1/p_t} \geq T + (\frac{\pi^2}{6} - 1)H_T$:

$$\mathbb{E}[S_T^{(1)}] \geq (1-\alpha)\left[T + \left(\frac{\pi^2}{6} - 1\right) H_T\right] + \alpha T = T + (1-\alpha)\left(\frac{\pi^2}{6} - 1\right) H_T$$

□

This bound matches the random skip bound (Proposition 8) with α playing the role of the skip rate: from the perspective of a single-key detector, tokens generated with the other key look exactly like skipped tokens. The advantage of dual-key routing is that the aggregated score (Equation 2) lets every token contribute signal from at least one key, as formalized below.

Proposition 5 (Expected score under dual-key Early Fusion detection)**.** *Under dual-key routing with parameter $\alpha \in [0,1]$, detecting with the aggregated score $s_i = (1-\alpha) \cdot s_i^{(1)} + \alpha \cdot s_i^{(2)}$ yields:*

$$\mathbb{E}[S_T] \geq T + \left(\alpha^2 + (1-\alpha)^2\right)\left(\frac{\pi^2}{6} - 1\right) H_T \tag{17}$$

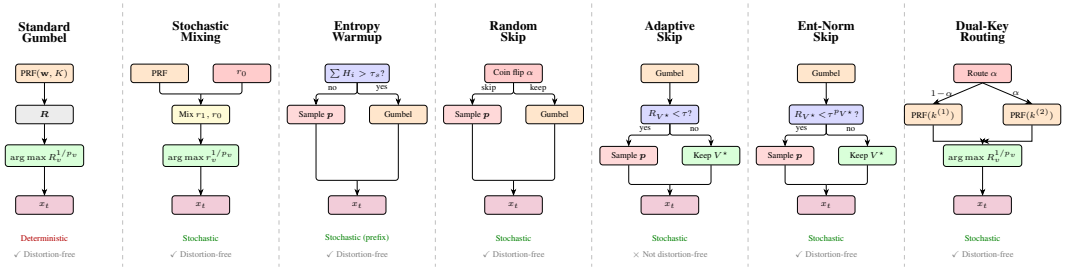


Figure 8: Overview of diversity mechanisms for Gumbel-Max watermarking. Each column shows how the token x_t is generated. All methods except Adaptive Skip preserve the distortion-free property. The key distinction lies in *where* randomness is injected: in the PRF value (Mixing), in the decision to watermark (Skip variants, Warmup), or in the key selection (Dual-Key Routing).

Proof. At each step t, key $k^{(1)}$ is active with probability $1-\alpha$ and key $k^{(2)}$ with probability α. The aggregated (early-fusion) per-token score is $T_t = (1-\alpha)s_t^{(1)} + \alpha s_t^{(2)}$.

- If $k^{(1)}$ was used (prob. $1-\alpha$): $s_t^{(1)}$ has the watermarked distribution ($\mathbb{E}[s_t^{(1)}] = \mathcal{H}_{1/p_t}$) and $s_t^{(2)}$ is uniform ($\mathbb{E}[s_t^{(2)}] = 1$), giving $\mathbb{E}[T_t] = (1-\alpha)\mathcal{H}_{1/p_t} + \alpha$.
- If $k^{(2)}$ was used (prob. α): $s_t^{(1)}$ is uniform ($\mathbb{E}[s_t^{(1)}] = 1$) and $s_t^{(2)}$ has the watermarked distribution ($\mathbb{E}[s_t^{(2)}] = \mathcal{H}_{1/p_t}$), giving $\mathbb{E}[T_t] = (1-\alpha) + \alpha\mathcal{H}_{1/p_t}$.

Taking expectation over the key choice:

$$\begin{aligned}\mathbb{E}[T_t] &= (1-\alpha)\big[(1-\alpha)\mathcal{H}_{1/p_t} + \alpha\big] + \alpha\big[(1-\alpha) + \alpha\mathcal{H}_{1/p_t}\big] \\ &= \big[(1-\alpha)^2 + \alpha^2\big]\mathcal{H}_{1/p_t} + 2\alpha(1-\alpha) \\ &= \theta_R\,\mathcal{H}_{1/p_t} + (1-\theta_R)\end{aligned}$$

where $\theta_R = \alpha^2 + (1-\alpha)^2$. Summing over T tokens:

$$\mathbb{E}[S_T] = \theta_R \sum_{t=1}^{T} \mathcal{H}_{1/p_t} + (1-\theta_R)\,T$$

Applying the standard bound $\sum_t \mathcal{H}_{1/p_t} \geq T + (\frac{\pi^2}{6} - 1)H_T$:

$$\mathbb{E}[S_T] \geq \theta_R \left[T + \left(\frac{\pi^2}{6} - 1\right) H_T\right] + (1-\theta_R)\,T = T + \theta_R \left(\frac{\pi^2}{6} - 1\right) H_T$$

□

Note that $\theta_R = \alpha^2 + (1-\alpha)^2 \leq 1-\alpha$ for $\alpha \leq 0.5$, so the expected score under Early Fusion is actually lower than under single-key detection (Proposition 4). The power advantage of Early Fusion comes not from a higher expected score, but from the reduced null variance (θ_R per token instead of 1), which yields a better Z-score, as the power analysis below shows.

C.1.1 Power Analysis: Early vs. Late Fusion

We analyze the statistical power of the Early Fusion test compared to a classical single-key baseline and alternative Late Fusion strategies using the Z-score (Signal-to-Noise Ratio) separation:

$$Z = \frac{\mathbb{E}[S|H_1] - \mathbb{E}[S|H_0]}{\sqrt{\mathrm{Var}(S|H_0)}}$$

Assume a standard Gumbel-Max test where an unwatermarked token yields an expected score of 1 with a variance of 1, and a successfully watermarked token yields an expected score $\mu_w > 1$.

Single-Key Baseline. For a traditional single-key test with n scored tokens, the expected score sum under H_1 is $n\mu_w$, and under H_0 it is n. The null variance is also n.

$$Z_{\text{base}} = \frac{n\mu_w - n}{\sqrt{n}} = \sqrt{n}(\mu_w - 1)$$

Early Fusion: Unweighted ($w = 0.5$). For the unweighted test, the expected score per token is $\mathbb{E}[\bar{s}_i] = \frac{\mu_w+1}{2}$ regardless of which key generated it. The null variance is $\mathrm{Var}(\bar{s}_i) = \frac{1^2+1^2}{2^2} = 0.5$.

$$Z_{\text{early}} = \frac{n(\frac{\mu_w+1}{2}) - n}{\sqrt{0.5n}} = \frac{n(\mu_w - 1)}{2\sqrt{0.5n}} = \frac{\sqrt{n}(\mu_w - 1)}{\sqrt{2}}$$

Thus, $Z_{\text{early}} = \frac{1}{\sqrt{2}} Z_{\text{base}} \approx 0.707 Z_{\text{base}}$. Hence, unweighted Early Fusion is **invariant to** α, but requires exactly twice as many tokens ($2n$) as the single-key baseline to reach the same confidence.

Early Fusion: Optimal Weighted ($w = \alpha$). If the routing probability α is known (e.g., via speculative decoding acceptance rates) and we use optimal weights $w = \alpha$, the expected token score under H_1 becomes $\mathbb{E}[s_i] = \alpha(\alpha\mu_w + 1 - \alpha) + (1 - \alpha)((1 - \alpha)\mu_w + \alpha)$. Simplifying this and calculating the Z-score yields:

$$Z_\alpha = \sqrt{n}(\mu_w - 1)\sqrt{\alpha^2 + (1 - \alpha)^2}$$

When $\alpha = 0.5$ (maximum diversity), $Z_\alpha = Z_{\text{early}} \approx 0.707 Z_{\text{base}}$. When $\alpha = 0.1$ (typical for draft model acceptance in speculative decoding), $Z_\alpha = \sqrt{0.1^2 + 0.9^2} Z_{\text{base}} \approx 0.905 Z_{\text{base}}$. Thus the weighted test recovers nearly 30% of the power lost to diversity under skewed generation rates.

Superiority over Late Fusion. We can now formally demonstrate why token-level aggregation outperforms independent per-key testing (late fusion). Late fusion evaluates each key's scores independently across the entire sequence ($S^{(1)} = \sum s_i^{(1)}$ and $S^{(2)} = \sum s_i^{(2)}$) and then combines their resulting p-values (e.g., via Fisher's method or by taking the minimum p-value).

Assuming without loss of generality that $\alpha \geq 0.5$, the expected signal for the dominant key over the null is $n\alpha(\mu_w - 1)$. The variance remains n. The statistical power of the combined Late Fusion test is ultimately bounded by the strongest independent signal it receives, which achieves at best:

$$Z_{\text{late}} \approx \frac{n\alpha(\mu_w - 1)}{\sqrt{n}} = \sqrt{n}\alpha(\mu_w - 1) = \alpha Z_{\text{base}}$$

To prove optimal Early Fusion natively dominates Late Fusion, we compare their Z-scores. We must show that $Z_\alpha > Z_{\text{late}}$, which simplifies to proving $\sqrt{\alpha^2 + (1 - \alpha)^2} > \alpha$ for any $\alpha \in (0, 1)$:

$$\alpha^2 + (1 - \alpha)^2 = \alpha^2 + (1 - 2\alpha + \alpha^2) = 2\alpha^2 - 2\alpha + 1$$

We test the inequality $2\alpha^2 - 2\alpha + 1 > \alpha^2$:

$$\alpha^2 - 2\alpha + 1 > 0 \implies (\alpha - 1)^2 > 0$$

Since $(\alpha - 1)^2$ is strictly positive for all $\alpha \in (0, 1)$, it follows that $Z_\alpha > Z_{\text{late}}$. Therefore, token-level aggregation strictly dominates independent per-key testing by preserving the complementary signal distributed across both keys ($k^{(1)}$ and $k^{(2)}$) at the token level, rather than systematically treating the minority key's tokens as noise during independent sequence-level evaluations.

C.2 Stochastic Mixing

Stochastic mixing injects true randomness by mixing the deterministic PRF value r_1 with a Bernoulli coin. For $a \in (0, 1)$, the mixed value is $r = a \cdot r_1$ with probability a, or $r = a + (1 - a) \cdot r_1$ with probability $1 - a$. r remains uniform (distortion-free), but detection uses only r_1.

Proposition 6 (Bound on expected score under mixing)**.** *Under stochastic mixing with parameter $a \in (0, 1)$, detection is performed using r_1 (the deterministic PRF value). The expected score satisfies:*

$$\mathbb{E}[S_T] > T + \left(\frac{\pi^2}{6} - 1\right) H_T + \sum_{t=1}^{T} (1 - a^{1/p_t}) \ln(1 - a) \tag{18}$$

Proof. Let $R \sim \text{Beta}(1/p, 1)$ be the selected value. The score for a single token is $s = -\ln(1 - r_1)$, where $r_1 = R/a$ if $R \in [0, a]$ and $r_1 = (R - a)/(1 - a)$ otherwise.

We decompose $\mathbb{E}[s]$ by interval:

$$\mathbb{E}[s] = \underbrace{\int_0^a -\ln(1 - r/a) f_R(r) dr}_{I_1} + \underbrace{\int_a^1 -\ln\left(1 - \frac{r - a}{1 - a}\right) f_R(r) dr}_{I_2}$$

For I_2: using $1 - \frac{r-a}{1-a} = \frac{1-r}{1-a}$:

$$I_2 = \int_a^1 [-\ln(1 - r) + \ln(1 - a)] f_R(r) dr = \int_a^1 -\ln(1 - r) f_R(r) dr + (1 - a^{1/p}) \ln(1 - a)$$

since $\mathbb{P}(R > a) = 1 - a^{1/p}$.

For I_1: since $r/a \geq r$ for $r \in [0, a]$, we have $-\ln(1-r/a) \geq -\ln(1-r)$, giving $I_1 \geq \int_0^a -\ln(1-r) f_R(r) dr$.

Summing I_1 and I_2 yields $\mathbb{E}[s] \geq \mathbb{E}[s_{\text{std}}] + (1-a^{1/p})\ln(1-a)$ where $\mathbb{E}[s_{\text{std}}] = \mathcal{H}_{1/p}$. Applying the standard bound and summing over all T tokens gives the result. □

Proposition 7 (Distortion-freeness of mixing)**.** *The mixed variable r follows $\mathcal{U}[0,1]$, so mixing is distortion-free: the sampled token follows the model distribution $\boldsymbol{p}$.*

Proof. Let $F_R(x) = \mathbb{P}(r \leq x)$. For $x \leq a$: $r \leq x$ requires $r_0 = 0$, giving $\mathbb{P}(r \leq x) = a \cdot \mathbb{P}(r_1 \leq x/a) = a \cdot x/a = x$. For $x > a$: $\mathbb{P}(r \leq x) = a + (1-a) \cdot \frac{x-a}{1-a} = x$. Hence, $r \sim \mathcal{U}[0,1]$. □

Behavior of the penalty. The penalty $(1-a^{1/p})\ln(1-a)$ is always non-positive (since $\ln(1-a) < 0$) and vanishes at both extremes: as $a \to 0$, $\ln(1-a) \to 0$; as $a \to 1$, $(1-(1-\epsilon)^{1/p})\ln(\epsilon) \sim \frac{\epsilon}{p}\ln(\epsilon) \to 0$. As expected, in both extremes all tokens take the same route, as in vanilla Gumbel-max.

C.3 Random Skip

Random skip disables the watermark independently at each token with probability α, reverting to standard (unwatermarked) sampling from $\boldsymbol{p}$. This blindly injects randomness to break deterministic loops, uniformly attenuating the detection signal regardless of entropy.

Proposition 8 (Bound on expected score under periodic skip)**.** *Under periodic skip with rate $\alpha \in [0,1]$ (each token is independently skipped with probability α), the expected score satisfies:*

$$\mathbb{E}[S_T] \geq T + (1-\alpha)\left(\frac{\pi^2}{6} - 1\right) H_T \tag{19}$$

Proof. At each step t, with probability $1-\alpha$ the watermark is active and $\mathbb{E}[s_t] = \mathcal{H}_{1/p_t}$; with probability α the watermark is skipped and $\mathbb{E}[s_t] = 1$. Summing:

$$\mathbb{E}[S_T] = (1-\alpha)\sum_{t=1}^{T} \mathcal{H}_{1/p_t} + \alpha T$$

Applying the standard bound (Proposition 3) to $\sum_t \mathcal{H}_{1/p_t} \geq T + (\frac{\pi^2}{6} - 1)H_T$:

$$\mathbb{E}[S_T] \geq (1-\alpha)\left[T + \left(\frac{\pi^2}{6} - 1\right) H_T\right] + \alpha T = T + (1-\alpha)\left(\frac{\pi^2}{6} - 1\right) H_T$$

□

The entropy-dependent signal is uniformly attenuated by a factor $(1-\alpha)$, regardless of the token entropy. This is wasteful compared to adaptive strategies that selectively skip only low-signal tokens.

C.4 Adaptive Skip

Adaptive skip disables the watermark selectively when the model is highly confident. At each step, the token is drawn via Gumbel-Max, but if the winning PRF value $R_{V^\star}$ falls below a threshold τ, the watermark is discarded and the token is resampled from $\boldsymbol{p}$. Low $R_{V^\star}$ indicates the token won by high probability mass rather than a favorable PRF draw, so skipping it loses little detection signal.

Proposition 9 (Adaptive skip is not distortion-free)**.** *Under adaptive skip with threshold $\tau \in (0,1)$, the output distribution is:*

$$\mathbb{P}(\textit{output} = v) = p_v\left(1 - \tau^{1/p_v} + \sum_{w \in \mathcal{V}} p_w \, \tau^{1/p_w}\right) \tag{20}$$

which differs from p_v unless $\boldsymbol{p}$ is uniform.

Proof. Let $V^\star$ be the initial token selected by the Gumbel-max trick, where $\mathbb{P}(V^\star = v) = p_v$. By Corollary 1, the conditional distribution of the pseudo-random value R_v is Beta$(1/p_v, 1)$. The watermark is skipped (and the token resampled from $\boldsymbol{p}$) if $R_{V^\star} < \tau$.

The marginal probability of outputting a specific token v decomposes into two disjoint events: keeping the initially selected v, or skipping and resampling $X_t = v$ from the original distribution $\boldsymbol{p}$:

$$\begin{aligned}
\mathbb{P}(\text{output} = v) &= \mathbb{P}(V^\star = v, R_v \geq \tau) + \mathbb{P}(\text{skip})\mathbb{P}(X_t = v) \\
&= \mathbb{P}(V^\star = v)\mathbb{P}(R_v \geq \tau \mid V^\star = v) + \left(\sum_{w \in \mathcal{V}} \mathbb{P}(V^\star = w)\mathbb{P}(R_w < \tau \mid V^\star = w)\right)\mathbb{P}(X_t = v) \\
&= p_v \cdot \mathbb{P}(R_v \geq \tau \mid V^\star = v) + \left(\sum_{w \in \mathcal{V}} p_w \mathbb{P}(R_w < \tau \mid V^\star = w)\right) p_v \\
&= p_v \left(1 - \tau^{1/p_v}\right) + p_v \sum_{w \in \mathcal{V}} p_w \tau^{1/p_w} \\
&= p_v \left(1 - \tau^{1/p_v} + \sum_{w \in \mathcal{V}} p_w \tau^{1/p_w}\right)
\end{aligned}$$

For the mechanism to be distortion-free, we require $\mathbb{P}(\text{output} = v) = p_v$ for all $v \in \mathcal{V}$. This implies:

$$\tau^{1/p_v} = \sum_{w \in \mathcal{V}} p_w \tau^{1/p_w}$$

The right-hand side is a constant across all tokens, whereas the left-hand side strictly depends on p_v. This equality holds if and only if all tokens have the exact same probability $p_v = 1/|\mathcal{V}|$. □

Remark 2. The distortion shifts mass from high-confidence tokens (large p_v, frequently skipped since τ^{1/p_v} is large) toward low-confidence tokens (small p_v, rarely skipped). For example, with $p_1 = 0.9$, $p_2 = 0.1$, and $\tau = 0.5$: the output probabilities become $(0.858, 0.142)$ instead of $(0.9, 0.1)$. In practice, τ is small (e.g., $\tau = 0.1$), so the resulting distortion is mild.

Proposition 10 (Bound on expected score under adaptive skip)**.** *Under adaptive skip with threshold $\tau \in [0, 1]$ (the watermark is disabled when $R^{(t)}_{V^\star} < \tau$), the expected score satisfies:*

$$\mathbb{E}[S_T] \geq T + \left(\frac{\pi^2}{6} - 1\right) H_T + \ln(1 - \tau) \sum_{t=1}^{T} \tau^{1/p_t} \tag{21}$$

The correction term is always non-positive, vanishing as $\tau \to 0$.

Proof. We condition on the identity of the selected token. By Proposition 1, $\mathbb{P}(V^\star = v) = p_v$. By Corollary 1, *conditioned on* $V^\star = v$, the PRF value $R_v \sim$ Beta$(1/p_v, 1)$ with density $f(r) = \frac{1}{p_v} r^{1/p_v - 1}$ and CDF $F(r) = r^{1/p_v}$. The skip condition $R_v < \tau$ therefore has conditional probability $\mathbb{P}(R_v < \tau \mid V^\star = v) = \tau^{1/p_v}$. This decreases with entropy: for confident tokens ($p_v \to 1$), $\tau^{1/p_v} \to \tau$ (frequent skipping); for unlikely tokens ($p_v \to 0$), $\tau^{1/p_v} \to 0$ (rare skipping).

We now bound $\mathbb{E}[s_t \mid V^\star = v]$. Decomposing over the skip decision:

$$\mathbb{E}[s_t \mid V^\star = v] = \underbrace{\mathbb{E}[-\ln(1 - R_v) \cdot \mathbf{1}_{R_v \geq \tau} \mid V^\star = v]}_{\text{not skipped: use watermarked token}} + \underbrace{\mathbb{E}[-\ln(1 - R_{X_t}) \cdot \mathbf{1}_{R_v < \tau} \mid V^\star = v]}_{\text{skipped: resample } X_t \sim \boldsymbol{p}}$$

The first term integrates the score over the non-skip region using the conditional density of R_v:

$$\mathbb{E}[-\ln(1 - R_v) \cdot \mathbf{1}_{R_v \geq \tau} \mid V^\star = v] = \int_\tau^1 \frac{-\ln(1 - r)}{p_v} r^{1/p_v - 1} dr$$

For the second term, the replacement token $X_t \sim \boldsymbol{p}$ is drawn with independent randomness, but its PRF value R_{X_t} comes from the *same* realization $\boldsymbol{R}$, so we cannot claim its expected score is 1 (see Remark 4). Since $-\ln(1 - R_{X_t}) \geq 0$, the second term is non-negative, so:

$$\mathbb{E}[s_t \mid V^\star = v] \geq \int_\tau^1 \frac{-\ln(1 - r)}{p_v} r^{1/p_v - 1} dr = \mathcal{H}_{1/p_v} - \int_0^\tau \frac{-\ln(1 - r)}{p_v} r^{1/p_v - 1} dr$$

where we used $\int_\tau^1 = \int_0^1 - \int_0^\tau$ and $\int_0^1 \frac{-\ln(1-r)}{p_v} r^{1/p_v-1} dr = \mathcal{H}_{1/p_v}$. Since $-\ln(1-r) \le -\ln(1-\tau)$ for $r \in [0,\tau]$:

$$\begin{aligned}
\int_0^\tau \frac{-\ln(1-r)}{p_v} r^{1/p_v-1} dr &\le \int_0^\tau \frac{-\ln(1-\tau)}{p_v} r^{1/p_v-1} dr \\
&= \frac{-\ln(1-\tau)}{p_v} \int_0^\tau r^{1/p_v-1} dr \\
&= \frac{-\ln(1-\tau)}{p_v} \left[p_v \cdot r^{1/p_v} \right]_0^\tau \\
&= \frac{-\ln(1-\tau)}{p_v} \left(p_v \cdot \tau^{1/p_v} - 0 \right) \\
&= -\ln(1-\tau) \cdot \tau^{1/p_v}
\end{aligned}$$

and therefore $\mathbb{E}[s_t \mid V^\star = v] \ge \mathcal{H}_{1/p_v} + \tau^{1/p_v} \ln(1-\tau)$. Since this holds for every v, it holds for the realized token probability $p_t = p_{V^\star}$. Summing over T steps and applying the standard bound (Proposition 3) to $\sum_t \mathcal{H}_{1/p_t} \ge T + (\frac{\pi^2}{6} - 1) H_T$ gives the result. □

Remark 3. The penalty $\ln(1-\tau) \sum_t \tau^{1/p_t}$ is always non-positive (since $\ln(1-\tau) < 0$), confirming that skipping can only reduce the detection signal. For small τ (e.g., $\tau = 0.1$), the penalty is negligible: τ^{1/p_t} is small for all but deterministic tokens ($p_t \approx 1$), and those tokens carry no watermark signal anyway ($\mathcal{H}_1 = 1$, equal to the null baseline). The bound is conservative because we dropped the skip contribution entirely; in practice, skipped tokens still contribute positively to the score.

Remark 4 (Skip contribution)**.** A tempting (but incorrect) approach is to claim that skipped tokens contribute expected score 1, arguing that the replacement token $X_t \sim \boldsymbol{p}$ is drawn independently and therefore its PRF value R_{X_t} is uniform. This would yield the decomposition:

$$\mathbb{E}[s_t] = \int_\tau^1 \frac{-\ln(1-r)}{p_t} r^{1/p_t-1} dr + \tau^{1/p_t} \cdot 1$$

leading to a correction $\tau^{1/p_t}(1 + \ln(1-\tau))$ that is *positive* for $\tau < 1 - 1/e$—implying that skipping *improves* detection, which is impossible since skipping can only reduce the detection signal.

The error is that while X_t is drawn independently of $\boldsymbol{R}$, the PRF value $R_{X_t} = \boldsymbol{R}[X_t]$ shares the *same* realization $\boldsymbol{R}$. Since the skip event $\{R_{V^\star} < \tau\}$ constrains $\boldsymbol{R}$ (the winning PRF value is low), the conditional expectation $\mathbb{E}[-\ln(1 - R_{X_t}) \mid R_{V^\star} < \tau] \ne 1$. A simple counterexample: for a deterministic token ($p_t = 1$), there is only one possible token, so skipping changes nothing and $\mathbb{E}[s_t] = \mathcal{H}_1 = 1$. Yet the incorrect formula gives $1 + \tau(1 + \ln(1-\tau)) > 1$.

C.5 Entropy-Normalized Adaptive Skip

This variant of adaptive skip replaces the fixed threshold τ with an entropy-dependent threshold $\tau^{p_{V^\star}}$, which ensures every token is skipped with exactly the same probability τ regardless of its confidence level. This restores the distortion-free property lost by standard adaptive skip.

For a target skip rate $\tau \in (0,1)$, the watermark is now disabled and the token is resampled from $\boldsymbol{p}$ if:

$$R_{V^\star} < \tau^{p_{V^\star}}$$

Proposition 11 (Distortion-freeness of entropy-normalized skip)**.** *The entropy-normalized adaptive skip mechanism is distortion-free, i.e., :* $\mathbb{P}(\textit{output} = v) = p_v$ *for all* $v \in \mathcal{V}$.

Proof. We first evaluate the conditional probability of a skip occurring given that token v was initially selected. By Corollary 1, $R_v \mid V^\star = v \sim \text{Beta}(1/p_v, 1)$, which has the cumulative distribution function $F(r) = r^{1/p_v}$. Therefore, the conditional skip probability is:

$$\mathbb{P}(\text{skip} \mid V^\star = v) = \mathbb{P}(R_v < \tau^{p_v} \mid V^\star = v) = (\tau^{p_v})^{1/p_v} = \tau$$

Because this conditional probability is exactly τ for *every* token in the vocabulary, the unconditional probability of a skip is also exactly τ. Indeed, by the law of total probability:

$$\mathbb{P}(\text{skip}) = \sum_{w \in \mathcal{V}} \mathbb{P}(\text{skip} \mid V^\star = w) \mathbb{P}(V^\star = w) = \sum_{w \in \mathcal{V}} \tau \cdot p_w = \tau \sum_{w \in \mathcal{V}} p_w = \tau$$

The total marginal probability of outputting token v can then be found by partitioning over the two mutually exclusive generation paths (whether a skip occurs or not):

$$\begin{aligned}\mathbb{P}(\text{output} = v) &= \mathbb{P}(\text{output} = v \cap \text{not skipped}) + \mathbb{P}(\text{output} = v \cap \text{skip}) \\ &= \mathbb{P}(V^\star = v \cap R_{V^\star} \geq \tau^{p_{V^\star}}) + \mathbb{P}(\text{skip}) \cdot \mathbb{P}(X_t = v) \\ &= \mathbb{P}(V^\star = v) \cdot \mathbb{P}(R_v \geq \tau^{p_v} \mid V^\star = v) + \mathbb{P}(\text{skip}) \cdot \mathbb{P}(X_t = v)\end{aligned}$$

We know the unconditional probability of a skip is τ, so the conditional probability of not skipping is $1 - \tau$. Furthermore, the replacement token X_t is sampled from the original distribution independently of the skip event, so $\mathbb{P}(X_t = v) = p_v$. Substituting these values yields:

$$\begin{aligned}\mathbb{P}(\text{output} = v) &= p_v \cdot (1 - \tau) + \tau \cdot p_v \\ &= p_v - p_v\tau + p_v\tau \\ &= p_v\end{aligned}$$

Thus, the marginal distribution is preserved and the mechanism is distortion-free. □

Proposition 12 (Bound on expected score under entropy-normalized skip)**.** *Under the entropy-normalized adaptive skip with target skip rate $\tau \in (0, 1)$, the expected score satisfies:*

$$\mathbb{E}[S_T] \geq T + \left(\frac{\pi^2}{6} - 1\right) H_T + \tau \sum_{t=1}^{T} \ln(1 - \tau^{p_t}) \tag{22}$$

Proof. We condition on the identity of the selected token $V^\star = v$. We decompose the expected score into the non-skipped and skipped cases:

$$\mathbb{E}[s_t \mid V^\star = v] = \mathbb{E}[-\ln(1 - R_v) \cdot \mathbf{1}_{R_v \geq \tau^{p_v}} \mid V^\star = v] + \mathbb{E}[-\ln(1 - R_{X_t}) \cdot \mathbf{1}_{R_v < \tau^{p_v}} \mid V^\star = v]$$

As discussed in Remark 4, the replacement token X_t relies on the same PRF realization $\boldsymbol{R}$, so its contribution is difficult to isolate but strictly non-negative. Dropping the second term provides a conservative lower bound:

$$\mathbb{E}[s_t \mid V^\star = v] \geq \int_{\tau^{p_v}}^{1} \frac{-\ln(1 - r)}{p_v} r^{1/p_v - 1} dr = \mathcal{H}_{1/p_v} - \int_{0}^{\tau^{p_v}} \frac{-\ln(1 - r)}{p_v} r^{1/p_v - 1} dr$$

Since the function $-\ln(1 - r)$ is monotonically increasing, for $r \in [0, \tau^{p_v}]$, we have $-\ln(1 - r) \leq -\ln(1 - \tau^{p_v})$. We can bound the subtracted integral:

$$\begin{aligned}\int_{0}^{\tau^{p_v}} \frac{-\ln(1 - r)}{p_v} r^{1/p_v - 1} dr &\leq -\ln(1 - \tau^{p_v}) \int_{0}^{\tau^{p_v}} \frac{1}{p_v} r^{1/p_v - 1} dr \\ &= -\ln(1 - \tau^{p_v}) \left[r^{1/p_v}\right]_{0}^{\tau^{p_v}} \\ &= -\ln(1 - \tau^{p_v}) \left(\tau^{p_v}\right)^{1/p_v} \\ &= -\tau \ln(1 - \tau^{p_v})\end{aligned}$$

Substituting this back yields:

$$\mathbb{E}[s_t \mid V^\star = v] \geq \mathcal{H}_{1/p_v} + \tau \ln(1 - \tau^{p_v})$$

Since this inequality holds for any chosen v, it holds for the realized token probability p_t. Summing over the sequence of T tokens and applying the standard Gumbel bound (Proposition 3) gives:

$$\mathbb{E}[S_T] \geq \sum_{t=1}^{T} \mathcal{H}_{1/p_t} + \tau \sum_{t=1}^{T} \ln(1 - \tau^{p_t}) \geq T + \left(\frac{\pi^2}{6} - 1\right) H_T + \tau \sum_{t=1}^{T} \ln(1 - \tau^{p_t})$$

The correction term is strictly non-positive because $\tau^{p_t} \in (0, 1)$, meaning $\ln(1 - \tau^{p_t}) < 0$. This accurately reflects the expected loss in signal when skipping exactly τ fraction of the tokens. □

Remark 5 (Skip behavior and score penalty)**.** Unlike standard adaptive skip where the skip probability τ^{1/p_v} depends on token confidence (skipping high-confidence tokens more often), the entropy-normalized threshold τ^{p_v} ensures a *uniform* skip rate of exactly τ for all tokens regardless of their probability. However, the per-token score penalty $\tau \ln(1 - \tau^{p_t})$ still varies with entropy:

- For high-confidence tokens ($p_t \to 1$): the penalty approaches $\tau \ln(1-\tau)$, which is mild. These tokens contribute little watermark signal anyway ($\mathcal{H}_1 = 1$, equal to the null baseline score), so skipping them has minimal impact on detection power.
- For low-confidence tokens ($p_t \to 0$): the threshold $\tau^{p_t} \to 1$, making the penalty bound $\tau \ln(1-\tau^{p_t}) \to -\infty$, making the bound useless. Such tokens occur rarely ($\mathbb{P}(V^\star = v) = p_v$), so their contribution to the penalty is attenuated by their low frequency.

The mechanism thus achieves distortion-freeness while concentrating the detection penalty on tokens that either carry little signal (high-confidence tokens) or appear rarely (low-confidence tokens), preserving most of the watermark power from medium-entropy tokens.

D Fast Localization and Statistical Penalties

In practical settings, the exact start and end indices of a watermarked insertion are unknown. Our objective is to determine a set of disjoint watermarked intervals $\{[a_1, b_1], \ldots, [a_y, b_y]\}$. A naive exhaustive search over all possible intervals in a sequence of length n requires evaluating $\binom{n}{2} \approx n^2/2$ windows. This $\mathcal{O}(n^2)$ search space not only introduces severe computational bottlenecks but also imposes an insurmountable statistical penalty via multiple-testing correction, as false positives become significantly more likely as the number of tested hypotheses grows. To optimize both computational and statistical efficiency, we utilize a geometric cover search space [Kirchenbauer et al., 2023b] combined with a fast two-stage extraction pipeline and rigorous Bonferroni correction.

D.1 The Geometric Cover Search

To avoid testing all $\mathcal{O}(n^2)$ intervals, we constrain our search to a dyadic grid of overlapping windows. We define a set of window lengths $L \in \{L_0, 2L_0, 4L_0, \ldots, 2^{\lfloor \log_2 n \rfloor}\}$, where $L_0 = 2^{\lceil \log_2 L_{\min} \rceil}$ is the smallest power of two at least as large as the minimum zone length $L_{\min}$. For each window length L, we slide the window across the entire text with a stride of $L/2$ tokens.

This geometric grid guarantees that any arbitrary watermarked region of length $L^* \geq L_{\min}$ will be at least 50% covered by at least one grid window. The total number of candidate windows M in this grid is strictly bounded:

$$M = \sum_{k=\lceil \log_2 L_{\min} \rceil}^{\lfloor \log_2 n \rfloor} \left\lfloor \frac{n - 2^k}{2^{k-1}} \right\rfloor + 1 \approx \frac{4n}{L_0} \tag{23}$$

By restricting the search to $M \approx \mathcal{O}(n/L_{\min})$ windows, we reduce the hypothesis space by orders of magnitude, dramatically lowering the statistical tax required to claim significance.

D.2 Fast Two-Stage Pipeline and Greedy Extraction

Evaluating the rigorous, entropy-weighted Gamma distribution for all M windows can be computationally heavy. To process arbitrarily large documents efficiently, we utilize a two-stage pipeline:

1. **Fast Filtering:** We pre-calculate prefix sums of the unweighted raw scores s_i. The sum for any candidate interval in the dyadic grid can then be computed in $\mathcal{O}(1)$ time. We select the top candidates based on these raw sums for rigorous scoring.
2. **Rigorous Scoring:** For the most promising candidates, we compute the exact entropy-weighted moment-matched Gamma p-value p_{raw} (as defined in Section 3.2).

To extract multiple zones, we proceed greedily. We find the window I^* with the smallest p_{raw}. If its penalized significance (accounting for the search tax) is high enough, we flag it as watermarked, mask its tokens (zeroing their scores), and repeat on the residual text. We aggregate disjoint intervals until the combined p-value fails to pass the multiple-testing threshold, up to a maximum of $Y_{\max}$ zones.

D.3 The Bonferroni Tax and False Positive Guarantee

Evaluating M intervals introduces the multiple comparisons problem. To maintain a strict family-wise error rate (FWER) ϵ under the null hypothesis $\mathcal{H}_0$, we apply a union bound.

For a single-zone search, the Bonferroni correction factor is simply M. For a multi-zone search identifying y disjoint regions, we must account for the number of ways to choose y windows from the grid, $\binom{M}{y}$, as well as the optimization over $y \in [1, Y_{\max}]$. The corrected p-value in log-space is:

$$\ln p_{\text{corrected}} = \ln p_{\text{raw}} + \ln \binom{M}{y} + \ln Y_{\max} \tag{24}$$

Under $\mathcal{H}_0$, the probability that the most significant window combination exceeds our threshold is strictly bounded:

$$\mathbb{P}\left(\bigcup_{i=1}^{K} \left\{ p_i \leq \frac{\epsilon}{K} \right\} \;\middle|\; \mathcal{H}_0 \right) \leq \sum_{i=1}^{K} \mathbb{P}\left(p_i \leq \frac{\epsilon}{K} \;\middle|\; \mathcal{H}_0 \right) = K \cdot \frac{\epsilon}{K} = \epsilon \tag{25}$$

where K is the total number of tested hypotheses in the search space. Thus the probability of falsely accusing an entirely human-written text remains $\leq \epsilon$, regardless of document length.

D.4 Asymptotic Power Comparison: Global vs. Localized Detection

The crossover point is where the localized multi-zone test rejects $\mathcal{H}_0$ more strongly than the global test. Let n be the document length and $\rho \in (0, 1]$ the fraction of watermarked tokens ($w = \rho n$).

Setup and Approximation. Let the weighted token score $\tilde{s}_i$ have mean μ_0 and variance σ^2 under $\mathcal{H}_0$. Under $\mathcal{H}_1$, the mean shifts to $\mu_w > \mu_0$. Let $\delta = (\mu_w - \mu_0)/\sigma$ be the per-token signal-to-noise ratio. Using a Gaussian tail approximation, the log p-value of a Z-score is $\ln p \approx -\frac{1}{2}Z^2$. We define $\Delta^2 = \delta^2/2$ as the expected log p-value accumulation rate per watermarked token.

Power of the Global Test. The global test evaluates all n tokens. The expected Z-score is:

$$Z_{\text{global}} = \frac{\rho n \sigma \delta}{\sigma\sqrt{n}} = \rho\delta\sqrt{n} \implies \mathbb{E}[\ln p_{\text{global}}] \approx -\rho^2 n \Delta^2 \tag{26}$$

The signal is quadratic in ρ; the $(1-\rho)n$ human tokens add no signal but inflate the variance.

Power of the Localized Test. Assuming a localized test correctly isolates the ρn watermarked tokens into y zones, the variance is reduced to the watermarked subset $\rho n \sigma^2$. The expected raw log p-value is:

$$Z_{\text{local}} = \frac{\rho n \sigma \delta}{\sigma\sqrt{\rho n}} = \delta\sqrt{\rho n} \implies \mathbb{E}[\ln p_{\text{local, raw}}] \approx -\rho n \Delta^2 \tag{27}$$

Accounting for the combinatorial tax $\ln \binom{n}{2y} \approx 2y \ln n$, the penalized localized score is:

$$\mathbb{E}[\ln p_{\text{local, final}}] \approx -\rho n \Delta^2 + 2y \ln n \tag{28}$$

The Crossover Point. The localized test dominates when $\mathbb{E}[\ln p_{\text{local, final}}] < \mathbb{E}[\ln p_{\text{global}}]$:

$$-\rho n \Delta^2 + 2y \ln n < -\rho^2 n \Delta^2 \tag{29}$$

$$n\Delta^2(\rho - \rho^2) > 2y \ln n \tag{30}$$

$$\rho(1-\rho) > \frac{2y \ln n}{n\Delta^2} \tag{31}$$

Hence, localized detection is optimal when the signal is sufficiently concentrated (low ρ) such that the variance reduction from excluding human tokens outweighs the logarithmic search tax.

D.5 The Dilution Rescue Effect

For largely unedited text, the ensemble gracefully defaults to the global test, paying a negligible worst-case penalty of $\log_{10}(3)$. Consider w watermarked tokens with expected per-token score $\mu > 1$ (as given by Proposition 3), split into y chunks within a document of length n. Under extreme dilution ($n \gg w$), the global test's significance drops as its signal-to-noise ratio scales by $\mathcal{O}(w^2(\mu-1)^2/n)$. The multi-region strategy isolates the pure signal ($\mathcal{O}(w(\mu-1)^2)$) but pays a combinatorial tax scaling as $\mathcal{O}(y \log_{10} n)$. Localization rescues detection when the isolated signal outpaces this logarithmic tax: $w(\mu-1)^2 \gtrsim y \log_{10} n$. For instance, $w = 800$ tokens ($\mu = 1.2$) in $y = 5$ chunks easily overcome the $5 \log_{10} n$ tax, allowing confident detection even within $n = 100,000$ tokens—a scenario where the global signal is destroyed.

Table 3: Benchmark accuracy (%) across 5 random seeds (non-watermarked) or 5 secret keys (watermarked). We report Mean $\pm$ Std to quantify generation variance. "Dedup" refers to n-gram deduplication at generation-time (see Remark 1). Differences between conditions fall within one standard deviation, indicating no systematic degradation from watermarking.

Benchmark	No Watermark	WM (no dedup)	WM (dedup)
AIME	41.0 ± 1.2	40.6 ± 1.2	40.8 ± 0.9
GSM8K	95.9 ± 0.3	95.6 ± 0.3	95.9 ± 0.3
HumanEval	97.1 ± 0.9	97.6 ± 2.4	97.8 ± 0.7
MBPP	50.2 ± 0.6	49.8 ± 0.4	49.7 ± 0.6
MMLU	87.6 ± 0.5	87.1 ± 1.6	87.8 ± 0.6

D.6 High-Resolution Boundary Annotation (mIoU)

While the greedy ensemble rigorously bounds the False Positive Rate, the harsh combinatorial tax forces it to prematurely discard small fragments, making it suboptimal for exact boundary estimation (mean Intersection over Union, mIoU). To achieve high-resolution localization, we decouple *detection* from *annotation*. If the ensemble rejects the null hypothesis $\mathcal{H}_0$, we drop the search taxes and apply a localized density smoother. Tokens satisfying a normalized weighted moving average $\bar{S}_i > \tau$ are annotated as watermarked, recovering fine-grained, sentence-level provenance.

E Additional Experiments and Details

E.1 Localization Experiments

Detailed dilution and fragmentation results are presented in the main body (Figure 3, subsection 4.5).

E.2 Benchmark Variance Analysis

Evaluating LLM performance on benchmarks with chain-of-thoughts involves variance due to the stochastic generation, and the final answer extraction heuristics (e.g., regex-based parsing for numbers or letters, or code execution for programming tasks). To quantify this variance and assess whether watermarking introduces systematic degradation or improvement, we re-ran some benchmarks with multiple random seeds for non-watermarked and multiple secret keys for watermarked generation.

Experimental Setup. We use Qwen 3.5-27B with reasoning enabled (reasoning temperature 0.6, top-$p = 0.95$, max 3,000 reasoning tokens). We evaluate on five benchmarks: AIME (math), GSM8K (math), HumanEval (code), MBPP (code), and MMLU (multiple choice). For non-watermarked generation, we use 5 different random seeds. For watermarked generation, we use the same values as secret keys, with both n-gram deduplication enabled and disabled (see Remark 1): when enabled, watermark contexts that have already appeared in the generation fall back to vanilla sampling.

Results. The standard deviation across seeds/keys ranges from 0.3% to 2.4%, depending on the benchmark and condition. Code benchmarks exhibit variance due to the binary nature of test execution and sensitivity to minor formatting differences. Crucially, the differences between the watermarked and the non-watermarked conditions fall within approximately one standard deviation, indicating no systematic performance degradation (or improvement) from watermarking.

Effect of n-Gram Deduplication. Enabling n-gram deduplication (vanilla sampling for repeated context windows) tends to lower variance, especially on MMLU (0.6 vs 1.6 std) and HumanEval (0.7 vs 2.4 std). This matches the observation that repeated n-gram contexts in reasoning chains yield more deterministic (and potentially repetitive) generation when not deduplicated.

E.3 Multilingual QA

This section provides the full experimental setup for the multilingual question-answering experiment.

Table 4: Quality comparison between watermarked (WM) and non-watermarked (Non-WM) answers on 6,000 multilingual QA pairs. SBERT: cosine similarity between WM and Non-WM answers (all-MiniLM-L6-v2). Reasoning/Answer tokens: average per response. Refusal/Wrong Script: percentage of responses with a refusal or with the wrong language script.

		Reasoning tokens		Answer tokens		Refusal %		Wrong Script %	
Language	SBERT	WM	Non-WM	WM	Non-WM	WM	Non-WM	WM	Non-WM
English	0.870	168	149	124	121	0.8	0.7	0.0	0.0
Arabic	0.914	296	232	161	155	0.6	0.6	0.6	0.6
Chinese	0.815	224	203	144	146	0.6	0.3	0.7	0.6
Hindi	0.920	293	229	166	161	1.1	1.1	1.1	1.1
Japanese	0.761	294	280	181	181	0.6	0.2	9.0	7.8
Overall	0.858	240	207	150	147	0.7	0.6	1.9	1.7

Table 5: Watermark detection performance on multilingual QA.

Metric	English	Arabic	Chinese	Hindi	Japanese	Overall
TPR@0.1%	53.6%	83.3%	79.5%	59.0%	51.0%	63.3%
Median $\log_{10} p$	-3.15	-5.23	-4.62	-3.51	-3.04	-3.72

Experimental Configuration. We use GPT-OSS-20B with reasoning enabled (max 2,000 reasoning tokens) and watermarking applied to the reasoning trace. Generation uses temperature 0.7, top-$p = 0.95$, and a maximum length of 4,096 tokens. The watermark employs Gumbel-Max with 3-gram context, dual-key early fusion ($\alpha = 0.1$), and a fixed secret key.

Datasets. We use 6,000 QA pairs across five languages: English (2,000 samples from ELI5), and Arabic, Chinese, Hindi, and Japanese (1,000 samples each from CaLMQA [Arora et al., 2025]).

System Prompt. The following system prompt was used for all languages:

> *"You are answering questions. Give a clear, concise explanation in plain language. Answer in the same language as the question. Keep your answer to 50–150 words. No bullet points, headers, or markdown formatting—just natural prose."*

Quality Comparison. Table 4 compares the quality of watermarked and non-watermarked outputs. Semantic similarity between WM and Non-WM answers is high (SBERT mean 0.86), so watermarking does not meaningfully alter content. Reasoning lengths increase slightly with watermarking ($\sim$16% more reasoning tokens, especially in non-English languages), likely due to sampling variance rather than a systematic effect. Refusal rates are low under both conditions ($< 1\%$). Script consistency is $> 98\%$ for all languages except Japanese (90%), with a small increase of 1% for WM.

Watermark Detection Results. Arabic and Chinese show the strongest detection, which is likely due to higher per-token entropy. Japanese shows the lowest detection (51% TPR@0.1%) due to the more constrained vocabulary and lower per-token entropy in CJK scripts.

Statistical Tests for Differences. We apply McNemar's test [McNemar, 1947] to assess whether watermarking systematically affects script consistency or refusal rates. For script consistency, we observe 52 discordant pairs where WM was wrong but Non-WM was correct, versus 39 where Non-WM was wrong but WM was correct; with continuity correction, this yields $\chi^2 = 1.58$ and $p = 0.21$. For refusal rates, we find 21 pairs where WM refused but Non-WM answered, versus 15 where Non-WM refused but WM answered, giving $\chi^2 = 0.69$ and $p = 0.41$. Both p-values are well above $\alpha = 0.05$, so watermarking does not systematically increase script errors or refusals.

E.4 False Positive Rate Calibration

Figure 9 validates the FPR calibration on 1 million unwatermarked Wikipedia passages (256 tokens each). Under standard unweighted dual-key detection (Figure 9, left), all methods tightly track the

diagonal down to $\tau \approx 10^{-4}$. Under lightweight (0.8B) entropy-weighted detection (Figure 9, right), TextSeal remains strictly well-calibrated, with its curve above the diagonal.

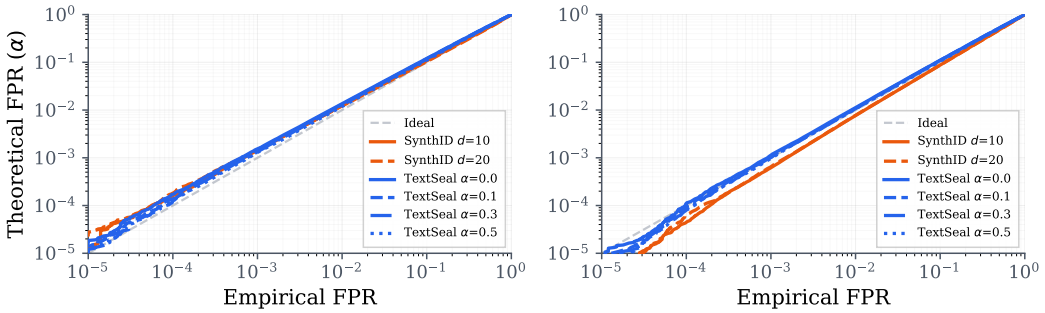


Figure 9: Theoretical FPR (τ) vs. empirical FPR under standard detection (left) and entropy-weighted detection (right), on 1M unwatermarked Wikipedia texts (256 tokens each). The dashed diagonal (empirical = theoretical FPR) indicates perfect calibration. All curves lie above the diagonal (i.e., are conservative): the empirical FPR never exceeds the nominal level τ.

E.5 Learnability Experimental Details

Table 6: **Teacher trace quality and detectability.** The teacher generates 5,000 traces per method; pass rate is the fraction retained by the four-stage quality filter. Teacher $-\log_{10}(p)$ reports the mean watermark detection power across individual teacher traces. Accuracy is measured on GSM8K (1,319 problems, greedy decoding); the baseline is the pre-training Qwen2.5-3B.

Method	**Retained Traces**	**Pass Rate**	**Teacher $-\log_{10}(p)$**	**GSM8K Acc**	**Δ vs Base**
Base Model (Qwen2.5-3B)	—	—	—	64.5%	—
Gumbel-Max	1,991	39.8%	14.89	78.8%	+14.3
TextSeal	2,352	47.0%	33.15†	79.9%	+15.4
SynthID	2,408	48.2%	14.39	75.2%	+10.7
Control	2,400	48.0%	0.39	75.5%	+11.0

Models & Dataset. The teacher is DeepSeek-R1-Distill-Qwen-14B [Guo et al., 2025], an R1-style reasoning model that produces long chain-of-thought traces enclosed in `<think>` tags, and the student is Qwen2.5-3B [Team, 2024]. We train on a subset of 5,000 problems drawn from OpenR1-Math-220k [Hugging Face, 2025], curated via a three-stage pipeline: (i) malformed or incomplete problems are removed; (ii) only problems that the student model fails to solve are retained, ensuring the training data teaches new capabilities; (iii) diversity sampling across the 14 math categories with a 15% cap per category prevents the overrepresentation of any single math topic.

Watermarked Trace Generation. We compare the three sampling-based methods introduced in subsection 4.1: *Gumbel-Max* [Aaronson and Kirchner, 2023], *TextSeal* (dual-key routing probability $\alpha = 0.1$), and *SynthID* [Dathathri et al., 2024] (plus an unwatermarked control), all with watermark context window $k = 3$. Secret keys are calibrated per method via a Kolmogorov–Smirnov test to ensure uniform PRF hashes on unwatermarked text as done in Fernandez et al. [2025]. The teacher generates 5,000 solutions using vLLM [Kwon et al., 2023] with flash-attention-2 on 4×H200 GPUs (tensor parallel), with $T = 1.0$, top-$p = 0.95$, and max 8,192 generated tokens.

Quality Filtering. Each teacher trace passes through four sequential filters (the first failure rejects the trace): (i) *think closure*—the trace must contain a closing `</think>` tag; (ii) *boxed presence*—the trace must include a `\boxed{...}` final answer (skipped for multiple-choice datasets); (iii) *repetition detection*—a sampled sliding-window check (window size 100 characters, ∼200 evenly spaced samples) rejects any trace where a substring occurs ≥3 times (responses ≤200 characters auto-pass); (iv) *answer verification*—the extracted answer is compared to the gold answer using the `math_verify` library in fail-open mode: if either side fails to parse, the trace is kept.

Student Fine-Tuning. The student is fine-tuned on the filtered traces using LoRA [Hu et al., 2022] (rank 128, scaling factor 128, dropout 0.05) with learning rate 2×10^{-5} and 3 epochs. The loss covers the full teacher response (reasoning trace and final answer), with prompt tokens masked out.

Watermark Detection. We evaluate watermark transfer using the *open-model* radioactivity test of Sander et al. [2024, 2025]. The test uses a *teacher-forcing* setup: each training trace is fed to the student, and the student's top-1 prediction $\hat{x}^{(t)} = \arg\max_{v \in \mathcal{V}} P_\theta(v \mid x_{<t})$ is recorded at every response position t. We score the student's *predictions* rather than newly generated text: this isolates the watermark signal from confounding factors such as sampling noise and generation quality, while requiring only one forward pass over existing traces rather than expensive autoregressive generation. If the student internalized the watermark's token preferences during fine-tuning, its top-1 predictions will be systematically biased toward high-PRF tokens—even without access to the secret key.

We score each prediction using the watermark's PRF: $R_t = \text{PRF}(\hat{x}^{(t)}, \mathbf{w}_t, K)$, where $\mathbf{w}_t = (x^{(t-k)}, \ldots, x^{(t-1)})$ is the trigram context window of *teacher* tokens preceding position t (as defined in section 2). Within each trace, each context window is scored only once; across traces, all (context, predicted token) pairs are pooled and deduplicated so that repeated tuples are counted only once, satisfying the independence assumption required by the statistical test [Fernandez et al., 2023]. This yields $\sim$1.4–2.2M unique scored tokens per watermarking method.

For Gumbel-Max, a single pooled Gamma test produces the p-value: we compute $s_t = -\ln(1 - R_t)$ for each unique pair and sum over all n unique scored tokens to obtain $S_n = \sum_{t=1}^{n} s_t$, which under $\mathcal{H}_0$ follows $\Gamma(n, 1)$ (Proposition 2). For TextSeal, we use the entropy-weighted early-fusion score with $w_i^{\text{ent}} = \sqrt{\hat{H}_i}$ weighting (subsection 3.2), with entropy estimated from the student's forward pass, and compute the p-value via the moment-matched Gamma approximation of Equation 5; the weighting function is validated by the ablation in Figure 10. For SynthID, we apply the frequentist test described in subsection 4.1, computing a depth-weighted Z-score over tournament layers.

Teacher Trace Quality. The filter pass rates across methods are as follows. Pass rates are 48% for the control, 48.2% for SynthID, 47% for TextSeal, and 39.8% for Gumbel-Max, yielding 2,400, 2,408, 2,352, and 1,991 well-formed traces respectively. Gumbel-Max traces are also notably shorter on average ($\sim$2,400 response tokens vs. $\sim$3,300–3,500 for other methods), because its deterministic argmax selection causes more repetition loops at T=1.0; the filter removes these long repetitive traces, leaving only shorter clean ones. Thus the student is trained on different amounts of data per configuration; we do not normalize for this, as the variation in sample count is modest ($\sim$20%).

Table 7: **Full learnability statistics (OpenR1, N=5,000 prompts).** Teacher $-\log_{10}(p)$ is the detection power of the watermark in the teacher's own traces (mean and median across individual traces). Student $-\log_{10}(p)$ is the pooled detection power after distillation (original setting, all retained traces). †TextSeal uses entropy-weighted scoring.

Method	**Retained Traces**	**Pass Rate**	**Teacher** $-\log_{10}(p)$ Mean	Median	**Student** $-\log_{10}(p)$
Gumbel-Max	1,991	39.8%	14.89	9.09	24.80
TextSeal	2,352	47.0%	33.15†	27.50†	35.82†
SynthID	2,408	48.2%	14.39	12.12	13.54
Control	2,400	48.0%	0.39	0.25	—

Controlled Comparisons. The results above are obtained with each method's full set of well-formed traces, which differ in count (1,991–2,408) and average length (Gumbel-Max traces average $\sim$2,400 tokens vs. $\sim$3,300–3,500 for other methods). To rule out training data volume as a confound, we repeat the experiment under two controlled conditions (Figure 6): (i) *equal traces*, where each method uses exactly 1,991 traces (the Gumbel-Max minimum, randomly subsampled for the other methods), and (ii) *equal tokens*, where each method is allocated $\sim$15.1M characters (subsampling traces for methods with more tokens, using all available traces for Gumbel-Max). Under equal traces, TextSeal achieves the highest student accuracy (81.0%), followed by SynthID and Control (78.8% each) and Gumbel-Max (77.7%). Under equal tokens, the spread narrows (79.7%/78.6%/79.6%/77.6% for TextSeal/Gumbel-Max/SynthID/Control). In both settings all watermarked students substantially improve over the pre-training baseline (64.5%). Detection results

confirm that all three watermarks remain strongly detectable under both controls, validating that the learnability conclusions of Figure 6 are not artifacts of unequal training data volume.

Entropy Weighting Ablation. The TextSeal results above use $\sqrt{\hat{H}}$ entropy-aware scoring by default (subsection 3.2). We validate this choice by comparing eight weighting functions in the same teacher-forcing setup. Each variant computes the weighted statistic $S_{\text{combined}} = \sum_{i=1}^{n} w_i^{\text{ent}} \cdot s_i$, where s_i is TextSeal's early-fusion score (Equation 2) and $w_i^{\text{ent}} = f(H_i)$ depends on the local entropy H_i at position i, estimated via one forward pass of the student. The p-value is computed via the moment-matched Gamma approximation of Equation 5, accounting for the heterogeneous weights.

Figure 10 compares eight weighting functions spanning two families: (i) transformations of the min-max normalized entropy $\hat{H}_i = (H_i - H_{\min})/(H_{\max} - H_{\min})$—Sqrt ($\sqrt{\hat{H}_i}$), Log ($\log(1+\hat{H}_i)$), Linear ($\hat{H}_i$, the default form of Equation 3), and Tanh ($\tanh(\hat{H}_i)$)—and (ii) raw entropy power functions H_i^{β} for $\beta \in \{0.5, 1.0, 1.5\}$. The concave $\sqrt{\hat{H}}$ weighting achieves the strongest detection ($p = 1.5 \times 10^{-36}$), improving over the uniform baseline ($p = 2.5 \times 10^{-22}$) by about 14 orders of magnitude. Concave functions outperform linear and superlinear alternatives because they moderately upweight high-entropy positions—where the watermark has more room to influence token selection (Proposition 3)—without over-amplifying noisy extreme-entropy tokens. Conversely, unnormalized power functions ($H_i^{1.0}$, $H_i^{1.5}$) are sensitive to the absolute entropy scale and perform no better than the unweighted baseline. This validates our default choice of $\sqrt{\hat{H}}$ weighting: concave, normalized transformations extract the most learnable signal, and the improvement over the unweighted baseline confirms that entropy-aware scoring is beneficial not only for direct generation-based detection but also for detecting watermark transfer to a student through distillation.

E.6 Embedding and Detection Efficiency

Generation Overhead. Table 8 details the sampling overhead during autoregressive decoding. TextSeal evaluates a fused dual-key PRF restricted to the top-p survivor tokens ($\sim$200 tokens), avoiding full-vocabulary hashes. This adds only $\sim$0.07 ms per token ($\leq$0.3% overhead). In contrast, SynthID's iterative tournament sampling requires d sequential rounds of top-p reweightings and a multinomial sampling step, costing $\sim$0.6 ms per token (0.4–1.1% overhead). Both methods operate entirely on the logits, requiring no model parameter changes or KV-cache modifications.

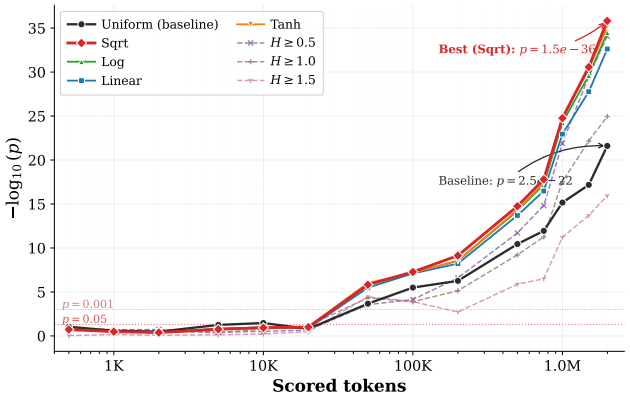


Figure 10: **Entropy-aware scoring for watermark learnability detection.** Detection power ($-\log_{10}(p)$) vs. unique scored tokens in the teacher-forcing radioactivity test for TextSeal (α=0.1), comparing entropy weighting functions $w_i^{\text{ent}} = f(H_i)$. We test transformations of the normalized entropy $\hat{H}_i = (H_i - H_{\min})/(H_{\max} - H_{\min})$—Sqrt ($\sqrt{\hat{H}_i}$), Log ($\log(1+\hat{H}_i)$), Linear ($\hat{H}_i$, cf. Equation 3), Tanh ($\tanh(\hat{H}_i)$)—and raw entropy power functions H_i^{β} for $\beta \in \{0.5, 1.0, 1.5\}$, against a uniform baseline. The concave $\sqrt{\hat{H}}$ weighting achieves the strongest signal ($p = 1.5 \times 10^{-36}$), improving on the uniform baseline ($p = 2.5 \times 10^{-22}$) by about 14 orders of magnitude, while superlinear weightings ($H^{1.5}$) and raw power functions perform at or below the baseline.

Table 8: Per-token sampling overhead of TextSeal and SynthID watermarking on a single H200 GPU. TextSeal uses dual-key Gumbel-Max (α=0.1, n-gram=3); SynthID uses tournament depth d=10. Values are medians over 30 prompts of ELI-5 $\times$ 400 tokens each.

		No Watermark		TextSeal		SynthID	
Model	**Fwd** (ms)	**Sample** (ms)	**tok/s**	**Sample** (ms)	**Overhead**	**Sample** (ms)	**Overhead**
Qwen 3.5-0.8B	21.4	0.37	45.9	0.43	0.3%	0.61	1.1%
Qwen 3.5-2B	21.5	0.36	45.8	0.43	0.3%	0.60	1.1%
Qwen 3.5-4B	30.3	0.38	32.6	0.44	0.2%	0.62	0.8%
Qwen 3.5-9B	31.3	0.38	31.5	0.45	0.2%	0.62	0.8%
Qwen 3.5-27B	61.9	0.39	16.1	0.46	0.1%	0.63	0.4%

E.7 MTP Speculative Decoding

Multi-token prediction (MTP) speculative decoding accelerates inference via lightweight draft heads that propose multiple tokens in parallel [Gloeckle et al., 2024]. TextSeal natively supports this by assigning key A to draft-accepted tokens and key B to target-resampled tokens. The mixing parameter α thus matches the empirical acceptance rate. We evaluate Qwen 3.5 (2B, 9B, 27B) generating 400-token ELI5 answers under three conditions: standard MTP, MTP with TextSeal, and autoregressive TextSeal. As shown in Figure 11, MTP draft acceptance rates remain identical (29–46%) with and without TextSeal, confirming the dual-key approach is distortion-free and preserves all speculative efficiency gains. While MTP TextSeal's detection signal is slightly lower than standard TextSeal due to key mixing dilution ($\alpha < 1$), entropy weighting recovers strong significance well above the p=0.01 threshold. Perplexity is identical across all conditions and model sizes.

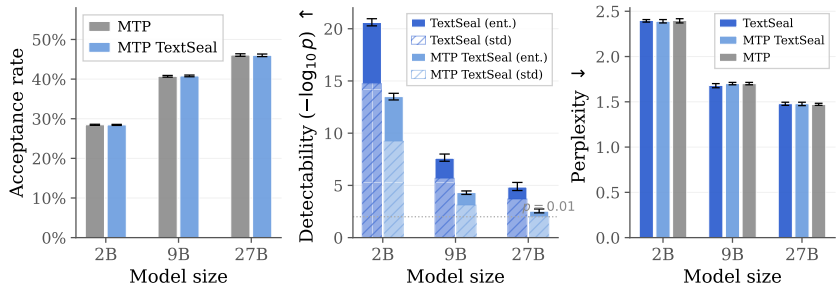


Figure 11: MTP speculative decoding with TextSeal watermarking across Qwen 3.5 model sizes (2B, 9B, 27B) at temperature 0.8, top-p 0.9. *Left:* Draft acceptance rate is unchanged by dual-key watermarking. *Center:* Both TextSeal and MTP TextSeal are well above the p=0.01 detection threshold. *Right:* Perplexity is identical across all conditions and model sizes.

E.8 Human Evaluation Details

This section provides methodology and detailed results for the human evaluation study summarized in subsection 4.4. The experimental setup (model, datasets, and generation parameters) is shared with the multilingual QA experiment described in detail in subsection E.3.

Preference Distribution. Table 9 shows the complete four-class preference breakdown before merging tie categories. Annotators chose among: *A is preferred*, *B is preferred*, *Both equally good*, and *Both equally bad*. We aggregate via majority vote (at least 2/3 annotators agree); samples with a three-way split (one vote per distinct category) are assigned to "Tie." For the final analysis, "Both Good," "Both Bad," and splits are merged into a single Tie category.

Net Win Rate. We define the *net win rate* as

$$\text{Net Win Rate} = \frac{n_{\text{WM}} - n_{\text{Base}}}{N}, \tag{32}$$

Table 9: **Full four-class preference breakdown** (majority vote, 3 annotators per sample). Split: items where no majority exists (three-way tie), counted as Tie in the final analysis.

Language	N	Prefer WM	Prefer Base	Both Good	Both Bad	Split
English	2,000	124	146	1,482	92	156
Arabic	1,000	201	181	168	287	163
Chinese	1,000	84	74	514	272	56
Hindi	1,000	91	89	435	278	107
Japanese	1,000	143	130	275	228	224
Overall	6,000	643	620	2,874	1,157	706

Table 10: **TOST equivalence test results** ($\Delta = 5\%$, $\alpha = 0.05$). Proportions computed over all N samples. 90% CI: Wald interval for the difference $P(\text{WM}) - P(\text{Base})$.

Language	N	$\hat{d}$	90% CI	p_{TOST}	Result
English	2,000	-1.10%	$[-2.5\%, +0.3\%]$	< 0.001	Equivalent
Arabic	1,000	$+2.00\%$	$[-1.2\%, +5.2\%]$	0.062	Marginal
Chinese	1,000	$+1.00\%$	$[-1.1\%, +3.1\%]$	< 0.001	Equivalent
Hindi	1,000	$+0.20\%$	$[-2.0\%, +2.4\%]$	< 0.001	Equivalent
Japanese	1,000	$+1.30\%$	$[-1.4\%, +4.0\%]$	0.013	Equivalent
Overall	6,000	$+0.38\%$	$[-0.6\%, +1.4\%]$	< 0.001	Equivalent

where n_{WM} and n_{Base} are the number of samples where the watermarked or baseline response was preferred (by majority vote), and N is the total number of samples including ties. The overall net win rate is $+0.38\%$, indicating a negligible difference between conditions.

Binomial Test. Among decisive (non-tie) samples, we test the null hypothesis H_0 : $P(\text{WM preferred}) = 0.5$ using a two-sided exact binomial test. No individual language reaches significance at $\alpha = 0.05$ (English: $p = 0.20$; Arabic: $p = 0.33$; Chinese: $p = 0.47$; Hindi: $p = 0.94$; Japanese: $p = 0.47$). The pooled test yields $p = 0.54$, also non-significant. The overall preference is nearly even (50.9% WM among decisive samples), indicating no quality degradation.

Equivalence Testing (TOST with Ties). To establish imperceptibility, rather than failing to detect a difference, we apply the Two One-Sided Tests (TOST) procedure [Schuirmann, 1987]. We test:

$$H_0\colon |P(\text{WM preferred}) - P(\text{Base preferred})| \geq \Delta \quad \text{vs.} \quad H_1\colon |P(\text{WM preferred}) - P(\text{Base preferred})| < \Delta \tag{33}$$

where proportions are computed over *all* N samples (including ties in the denominator). This formulation is more powerful than restricting to decisive samples, as ties are direct evidence of imperceptibility (annotators could not distinguish the outputs) and contribute to the sample size.

Let $\hat{d} = \hat{p}_{\text{WM}} - \hat{p}_{\text{Base}}$ with standard error $\text{SE} = \sqrt{(\hat{p}_{\text{WM}} + \hat{p}_{\text{Base}} - \hat{d}^2)/N}$. TOST computes $z_1 = (\hat{d} - \Delta)/\text{SE}$ and $z_2 = (\hat{d} + \Delta)/\text{SE}$ and rejects H_0 when $\max(\Phi(z_1),\, 1 - \Phi(z_2)) < \alpha$.

Table 10 reports results for $\Delta = 5\%$. Equivalence is established for four of the five languages and overall ($p < 0.05$); only Arabic marginally fails ($p = 0.062$) as its upper 90% confidence bound (5.2%) slightly exceeds the ± 5 percentage-point equivalence margin Δ.

On Trinomial Tests. An alternative approach is the trinomial test for paired data with ties [Bian et al., 2011, Dathathri et al., 2024], which models the three-category distribution (WM, Base, Tie) directly. We experimented with this approach but found that the chi-square statistic converges rapidly with the number of ties: once more than a handful of ties are present, the p-value stabilizes to the second decimal place and equals the standard binomial test on decisive samples. Since 79% of our samples are ties, the trinomial test provides no additional discriminative power, which motivates our use of the TOST procedure that explicitly leverages ties as evidence of imperceptibility.

Inter-Annotator Agreement. We measure agreement using two metrics: (i) unanimous agreement rate (fraction of samples where all 3 annotators selected the same four-class option), and (ii) mean

Table 11: **Inter-annotator agreement statistics by language** (four-class scale).

Language	Unanimous Rate	Majority ($\geq$2/3)	Pairwise Agreement
English	54.0%	92.2%	0.667
Arabic	23.9%	83.7%	0.438
Chinese	48.5%	94.4%	0.638
Hindi	37.0%	89.3%	0.544
Japanese	17.0%	77.6%	0.372
Overall	39.1%	88.2%	0.555

pairwise agreement (average fraction of annotator pairs that agree on the four-class label). Table 11 shows that agreement varies by language, with English and Chinese exhibiting the highest consistency. The lower agreement rates for Arabic and Japanese may reflect the inherent subjectivity of quality judgments and cultural differences in evaluation norms across languages.

Rationale Analysis. When annotators preferred one response over another, they provided rationales categorized into four types: "Not factual," "Unnatural language," "Bad word choice," and "Other." Table 12 shows the distribution of rationales for rejected responses. Factual errors and unnatural language are the two most common concerns across both watermarked and non-watermarked rejected responses, with no systematic difference between the two conditions.

Table 12: **Rationale distribution for rejected responses.** Counts indicate how often each rationale type was cited when annotators preferred the alternative response.

Response Type	Not Factual	Unnatural	Bad Word Choice	Other
Watermarked rejected	341	303	119	20
Non-watermarked rejected	356	298	139	35

Text Quality Metrics. We compare objective quality metrics of watermarked and non-watermarked responses. Mean lengths are nearly identical (WM: 176.8 ± 119.2; Non-WM: 177.9 ± 122.9 tokens), and repetition rates (the fraction of repeated n-grams) show no significant difference. Thus watermarking does not systematically affect surface-level text characteristics.

F Extended Related Work

F.1 Post-Hoc Text Watermarking

Early text watermarking altered surface-level text characteristics such as characters or spacing [Brassil et al., 1995]. Later methods modify grammatical or syntactical structures via pre-established rules [Topkara et al., 2005], including synonym substitution [Topkara et al., 2006c] and word reordering through passivization or topicalization [Topkara et al., 2006b,a, Meral et al., 2009]. Text steganography follows similar principles [Winstein, 1998, Chapman et al., 2001, Bolshakov, 2004, Shirali-Shahreza and Shirali-Shahreza, 2008, Chang and Clark, 2014, Xiang et al., 2017]. These edit-based systems exhibit low robustness and payload, e.g., 1–2 bits per sentence [Wilson and Ker, 2016]. Deep learning methods followed, including masked language models for steganography [Ueoka et al., 2021], infilling models [Yoo et al., 2023], neural lexical substitution [Qiang et al., 2023], and encoder-decoders [Abdelnabi and Fritz, 2021, Zhang et al., 2024, Xu et al., 2024].

F.2 Generation-Time LLM Watermarking

The first watermarks for machine-generated text date back to a method presumably used in Google Translate to filter translations from future training data [Venugopal et al., 2011]. For LLM-generated text, two concurrent approaches appeared shortly after the release of ChatGPT: Kirchenbauer et al. [2023a] bias a subset of the vocabulary ("green-red list"), while Aaronson and Kirchner [2023] alter the sampling via the Gumbel-max trick. Both use pseudorandom seeds generated from a secret key and preceding tokens, enabling lightweight statistical detection without access to the model.

Subsequent work explores improved tests and multi-bit watermarking [Fernandez et al., 2023, Yoo et al., 2024, Qu et al., 2024], position-dependent seeds [Christ et al., 2023, Kuditipudi et al., 2023], low-entropy optimizations [Lee et al., 2023, Christ et al., 2023, Huang et al., 2023], and semantic watermarks for improved robustness [Liu et al., 2023, Liu and Bu, 2024, Fu et al., 2024, Hou et al., 2023, 2024]. DiPMark [Wu et al., 2023] provides distortion-free green-red watermarks, and MorphMark [Wang et al., 2025] adaptively adjusts watermark strength based on the green token probability mass. WaterMax [Giboulot and Furon, 2024] generates several candidate chunks and selects outputs with high watermark scores, preserving the original LLM distribution. SynthID-Text [Dathathri et al., 2024] deploys tournament-based sampling in Google Gemini. Toolkits have also been introduced to benchmark these methods [Piet et al., 2023, Pan et al., 2024].

TextSeal builds on the Gumbel-max framework but introduces dual-key generation for diversity, entropy-weighted detection, and localized multi-region search—none present in prior work.

F.3 Post-Hoc LLM Watermarks for Data Protection

Recent works apply LLM watermarks to training or evaluation data via paraphrasing. Most exploit watermark radioactivity [Sander et al., 2024], i.e., the detectable traces left when watermarked text is used for training. Applications include detection of texts used in retrieval-augmented generation [Jovanović et al., 2025], benchmark contamination detection [Sander et al., 2025], and training data copyright [Zhang et al., 2025]. Waterfall [Lau et al., 2024] evaluates post-hoc watermarking through LLM paraphrasing for provenance on code and natural text. In section 6, we demonstrate that TextSeal's watermark transfers through distillation, extending it to reasoning traces.

G Robustness, Other Models, and Further Analyses

All experiments in this section use the setting of subsection 4.2 (ELI5, T=0.8, top-p=0.9, 400 tokens, window k=3, (window, token) deduplication, 1,000 answers per configuration) unless stated otherwise. SynthID-Text uses Google DeepMind's released g-values; TextSeal uses $\alpha = 0.1$ and SynthID-Text 20 layers, whose Self-BLEU (36.6 and 35.5) is the closest match on the frontier of Figure 1a; all runs use one key and one seed, so values differ slightly from section 5. Each detector is reported with its key-only test and with TextSeal's entropy-weighted test (subsection 3.2, Qwen 3.5-0.8B entropies); for SynthID-Text, the weighted test is applied to $2\bar{g}_t$, the mean g-value of token t over the layers, which has null mean 1 and variance $1/m$. TPR is reported at a nominal FPR, i.e. the share of watermarked answers with p below it; on 10 keys $\times$ 2,000 unwatermarked answers, every test's empirical FPR is at most its nominal level at 10^{-2}, 10^{-3}, and 10^{-4}.

G.1 Entropy Weighting for SynthID-Text

With its own test, SynthID-Text detects less than TextSeal at every diversity level (Figure 1a). Since entropy weighting is not specific to TextSeal, Figure 12 also scores both methods with the same three tests: key-only, entropy-weighted with Qwen 3.5-0.8B, and entropy-weighted with the generator (Qwen 3.5-27B, "high"). Entropy weighting helps both methods by a similar amount, and the comparison between them is the same with every test: SynthID-Text detects more in the most diverse regime (Self-BLEU below 33), and TextSeal detects more at lower diversity, including its default $\alpha = 0.1$ (Self-BLEU 36.6; median $-\log_{10} p$ of 6.7 against 6.1 for SynthID-Text with 20 layers, both with 27B entropies).

G.2 Threat Model

G.3 Post-Editing Attacks

Attacks. The attacker is Llama-3.2-3B-Instruct. *Rephrase* sends one user turn, "Rephrase the following text while preserving its meaning. Output only the rephrased version, nothing else. Text: {text} Rephrased:", sampled at temperature 0.5 / 0.8 / 1.0 / 1.2 (mild / moderate / aggressive / extreme, top-p=1). *Iterative rephrase* applies the aggressive preset 2 or 4 times in a row. *Back-translation* translates to French and back with the instruction "Translate the following text into {language}. Output only the translation, nothing else.", greedy. *Insert/delete* deletes 10% of the words and inserts a random word of the same text after another 10%. The attacked text is re-tokenized with

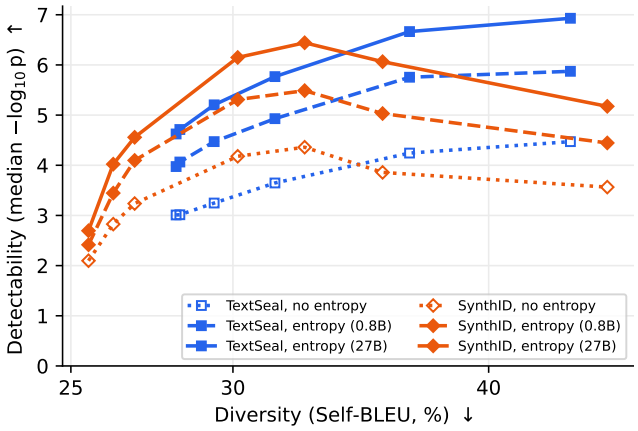


Figure 12: TextSeal and SynthID-Text (Google DeepMind's g-values, 2–30 layers) with the same detectors, Qwen 3.5-27B, 1,000 answers per point (median over the answers).

Table 13: Threat model. Generation and detection run on the provider's servers.

	In scope	Out of scope
Generation	provider-side; keys are secret	white-box access to the keys or the logits
Detector	provider-side	unlimited queries to the detector (score descent, key recovery)
Post-processing	rephrasing, iterative rephrasing, back-translation, word-level edits, dilution	full human re-authoring (no longer derived from the output)
False positives	calibrated operation at FPR 10^{-2} to 10^{-4}	–

the generator's tokenizer and its first 400 tokens are scored. Table 14 reports, for each attack, the word-level edit rate (Levenshtein distance divided by the original length) and the SBERT cosine similarity (all-MiniLM-L6-v2) to the original, measured on the TextSeal answers.

Table 14: TPR (%) under post-editing attacks at nominal FPR 10^{-2} and 10^{-4}, Qwen 3.5-27B. GM: Gumbel-Max (single key); SID: SynthID-Text, 20 layers; TS: TextSeal, α=0.1; "+EW": entropy-weighted test. Edit, SBERT: similarity to the original. Wilson intervals are about ±3 points.

			FPR 10^{-2}					FPR 10^{-4}				
Attack	Edit	SBERT	GM	SID	SID+EW	TS	TS+EW	GM	SID	SID+EW	TS	TS+EW
No attack	–	–	92.5	88.0	95.2	89.4	95.6	67.7	47.8	68.4	54.6	72.8
Rephrase, mild (T=0.5)	0.37	0.94	41.1	34.3	43.5	35.1	44.3	14.1	8.9	12.5	8.9	14.8
Rephrase, moderate (T=0.8)	0.48	0.93	24.0	20.1	26.6	18.2	25.2	6.4	3.2	4.1	3.4	6.8
Rephrase, aggressive (T=1.0)	0.61	0.92	10.1	7.9	10.4	9.5	12.0	1.2	0.4	0.7	1.2	2.3
Rephrase, extreme (T=1.2)	0.78	0.90	4.3	2.4	2.7	2.2	3.4	0.2	0.0	0.1	0.0	0.2
Iterative rephrase, 2 rounds	0.75	0.88	2.6	3.8	3.7	2.9	4.0	0.3	0.1	0.1	0.1	0.1
Iterative rephrase, 4 rounds	0.84	0.84	1.8	0.5	0.9	1.7	2.1	0.0	0.1	0.0	0.0	0.1
Back-translation EN→FR→EN	0.33	0.94	31.1	21.2	27.6	24.2	36.0	5.1	1.6	2.7	3.0	5.4
Insert/delete 20% of words	0.19	0.96	47.9	31.0	34.3	40.4	42.7	10.6	2.9	3.5	7.7	8.8

Results. At matched diversity, TextSeal detects slightly more than SynthID-Text on clean text, and the gap widens at low FPR (with entropy weighting, 72.8% vs 68.4% at 10^{-4}, and 59.2% vs 50.7% at 10^{-5}, where p-values are analytic since calibration is verified down to 10^{-4}). After word-level edits and back-translation TextSeal keeps more answers detected (42.7% vs 34.3% and 36.0% vs 27.6% at 10^{-2}, entropy-weighted), while under LLM rephrasing both are within the intervals of each other. Entropy weighting helps both methods under every attack that leaves a detectable signal. Gumbel-Max, which gives up diversity, is generally the most robust. Aggressive, extreme, and iterative rephrasing remove all three watermarks in most answers, while keeping SBERT similarity at 0.84 or above; we state this as a limitation: a capable rephrasing model with a meaning-preserving prompt is an effective attack on every distortion-free scheme we evaluate, and robustness to it would require a different design (e.g., semantic watermarks), outside our distortion-free setting.

G.4 Another Model Family

Table 15 repeats the frontier of Figure 1a with Llama-3.1-8B-Instruct as the generator (Llama-3.2-1B-Instruct for entropies; 2,000 unwatermarked answers $\times$ 10 keys for the empirical 10^{-3} thresholds; Self-BLEU over 16 answers to each of 50 questions). The trade-off and the gain of entropy weighting carry over. At similar diversity, TextSeal and SynthID-Text are level in the most diverse regime (Self-BLEU 28.8: SynthID-Text with 10 layers detects 69.5% / 87.2% with the key-only / weighted test; TextSeal 68% / 85%, interpolated between $\alpha = 0.3$ and 0.5), and TextSeal is ahead at lower diversity (Self-BLEU 35.5: 79.8% / 93.1% for $\alpha = 0.1$ against 59.8% / 87.1% for 20 layers).

Table 15: Llama-3.1-8B-Instruct: Self-BLEU (lower is more diverse), TPR (%) at FPR 10^{-3}, and median $-\log_{10} p$, key-only and entropy-weighted tests; 1,000 answers per row.

		TPR @ 10^{-3}		median $-\log_{10} p$	
Method	Self-BLEU	key-only	weighted	key-only	weighted
Gumbel-Max	100.0	89.3	97.7	5.71	8.67
SynthID-Text, 20 layers	35.5	59.8	87.1	3.68	5.44
SynthID-Text, 10 layers	28.8	69.5	87.2	4.03	5.48
SynthID-Text, 5 layers	26.7	54.1	76.9	3.25	4.20
SynthID-Text, 2 layers	25.7	29.2	46.5	2.12	2.67
TextSeal, $\alpha = 0.1$	35.5	79.8	93.1	4.75	6.83
TextSeal, $\alpha = 0.2$	30.9	75.1	88.7	4.31	6.26
TextSeal, $\alpha = 0.3$	29.1	69.4	85.1	3.94	5.61
TextSeal, $\alpha = 0.5$	27.7	63.0	82.3	3.61	5.08

G.5 Component Ablation

Table 16 adds TextSeal's components one at a time on clean text. Dual-key routing buys diversity (Self-BLEU from 100 to 36.6) at a cost in detection, which early fusion partly recovers over a Fisher combination of the two keys' p-values, and entropy weighting more than recovers. Localization is a capability for mixed documents rather than a gain on clean text (subsection 4.5, G.7).

Table 16: Incremental ablation, Qwen 3.5-27B, clean text. TPR (%) at nominal FPR.

Configuration	Self-BLEU	median $-\log_{10} p$	TPR @ 10^{-2}	TPR @ 10^{-4}
Gumbel-Max (single key)	100.0	5.14	92.5	67.7
+ dual-key routing, late fusion	36.6	3.72	82.1	46.0
+ early fusion	36.6	4.24	89.4	54.6
+ entropy-weighted detection	36.6	5.75	95.6	72.8

Table 17: TextSeal's components at a glance.

Component	Role	Cost	Where
Dual-key routing	restores diversity; natively compatible with speculative decoding and MTP	one Bernoulli draw per token	subsection 3.1, E.7
Early fusion	one Gamma test on the routing-weighted sum of both keys' scores	none	subsection 3.1
Entropy-weighted detection	up-weights high-entropy positions, keeps calibrated p-values	one forward pass of a small model at detection	subsection 3.2
Localization	finds watermarked regions in mixed documents, with a family-wise error guarantee	windowed tests at detection	subsection 3.3

G.6 Choosing α

Table 18 lists, for each α, the diversity and the TPR at three target FPRs. Given an application's diversity requirement (a maximum Self-BLEU) and a target FPR, we recommend the smallest α that meets the diversity requirement, and reading its TPR from the table; entropy weighting adds detection

at no diversity cost. For instance, a Self-BLEU of at most 37 allows $\alpha = 0.1$ (87% TPR at FPR 10^{-3} with entropy weighting), and at most 30 requires $\alpha \geq 0.3$ (73%). Beyond $\alpha = 0.3$, diversity barely improves while detection keeps decreasing, so we do not recommend larger values.

Table 18: Diversity and TPR (%) at nominal FPR for each routing probability α, Qwen 3.5-27B.

		key-only			entropy-weighted		
α	Self-BLEU	10^{-2}	10^{-3}	10^{-4}	10^{-2}	10^{-3}	10^{-4}
0.00	100.0	92.5	82.5	67.7	97.4	91.9	84.1
0.05	43.9	90.4	76.3	58.2	96.7	90.0	76.9
0.10	36.6	89.4	71.6	54.6	95.6	87.2	72.8
0.20	31.5	83.7	63.4	42.6	92.8	82.2	66.8
0.30	29.4	78.3	55.3	37.0	89.8	72.6	57.7
0.40	28.3	76.7	50.7	30.0	87.6	72.1	50.9
0.50	28.1	73.5	50.3	29.8	86.0	69.3	49.6

G.7 A Localized SynthID-Text Baseline

Localization is not specific to TextSeal: SynthID-Text's frequentist score can enter the same adaptive ensemble. We build documents from unwatermarked Qwen 3.5-27B answers and insert the first 400 tokens of one watermarked answer (at least 300 tokens long), whole (dilution) or split into equal fragments at random positions (fragmentation). TextSeal's fused score and SynthID-Text's per-token score $2\bar{g}_t$ go through the same ensemble (global test, best dyadic window, greedy multi-region, Bonferroni-corrected, $L_{\min} = 50$, $Y_{\max} = 5$), key-only; SynthID-global is its binomial test on the whole document. Table 19 reports the median $-\log_{10} p$ and the share of documents detected at $p < 10^{-2}$ over 200 documents per row. Localization lifts SynthID-Text's detection under dilution (21.5–41% against 3–29% for its global test) but barely under fragmentation. TextSeal detects more single blocks in long documents (36.5% vs 23.5% at 8k tokens, 34.5% vs 21.5% at 12k; intervals about ± 7 points), with similar medians, while the localized SynthID-Text detects more blocks split into 5 fragments. These values are lower than in Figure 3 because the setting differs: key-only tests, answers generated at $T{=}0.8$ with Self-BLEU-matched parameters, and unwatermarked model answers instead of Wikipedia as filler. Neither flags documents without a watermark.

Table 19: Localized detection, median $-\log_{10} p$ and detection rate (%) at $p < 10^{-2}$, 200 documents per row.

	TextSeal, localized		SynthID-Text, localized		SynthID-Text, global	
Document	median	detected	median	detected	median	detected
Dilution, 2k tokens (about 20% watermarked)	1.96	47.5	1.83	41.0	1.39	29.0
Dilution, 4k tokens (about 10% watermarked)	1.71	44.5	1.44	34.0	1.00	16.5
Dilution, 8k tokens (about 5% watermarked)	1.20	36.5	1.21	23.5	0.81	5.5
Dilution, 12k tokens (about 3% watermarked)	1.04	34.5	1.09	21.5	0.68	3.0
Fragmentation, 8k tokens, 2 fragments	0.59	17.0	0.77	10.5	0.71	5.0
Fragmentation, 8k tokens, 3 fragments	0.35	11.0	0.73	10.5	0.77	7.5
Fragmentation, 8k tokens, 5 fragments	0.24	4.0	0.75	10.0	0.81	8.0
No watermark, 8k tokens	0.00	0.0	0.15	0.0	0.37	0.5

G.8 Coexistence of Several Watermarks

A text is watermarked once, by the provider that generated it, so two schemes never act on the same sampling step. Detection is keyed: under independent keys, TextSeal's PRF values on text watermarked by another scheme are uniform, so each provider's detector keeps its calibrated FPR on the other's text. With n providers each checking a document at FPR τ, the probability of any false flag is at most $n\tau$ by the union bound, which a Bonferroni or FDR correction controls. Sequential provenance, e.g., a model distilled from TextSeal outputs that adds its own watermark, could yield two detections, which indicate the chain of provenance rather than a conflict (section 6).

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction claim that TextSeal is distortion-free, outperforms SynthID, preserves benchmark accuracy, supports speculative decoding, and transfers through distillation. Each claim is supported by experiments: diversity-detectability frontier (Section 4.1, Figure 1a), benchmarks (Section 4.2, Table 1), human evaluation (Section 4.3), localization (Section 4.4), efficiency (Section 5.2), and radioactivity (Section 6); limitations are discussed in the Conclusion and in Appendix G.3 (attacks).

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The Conclusion includes a dedicated Limitations paragraph discussing the diversity-quality trade-off inherent to distortion-free watermarks and its potential impact on workflows relying on diverse outputs, the removal of the watermark by strong LLM rephrasing (Appendix G.3), and the threat model (Appendix G.2).

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best

judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: All theoretical results (distortion-freeness, p-value distribution under $\mathcal{H}_0$, expected score under $\mathcal{H}_1$, early vs. late fusion, diversity bounds) are stated with explicit assumptions and proved in Appendix B–C. The main text states the results with explicit references to the corresponding propositions and proofs in the appendix.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: All experimental settings are fully specified: models (Qwen 3.5-27B, Qwen 3.5-0.8B for entropy proxy), hyperparameters (temperature, top-p, watermark context window k=3, routing probability α=0.1), datasets (ELI5, CaLMQA, Wikipedia, OpenR1-Math-220k, GSM8K, and 12 benchmarks), and evaluation protocols (deduplication, p-value computation, TOST equivalence testing). The PRF hash function is fully specified in Appendix A.1. SynthID detection details are in Appendix A.2.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.

(b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.

(c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The code is released at `https://github.com/facebookresearch/textseal`. All datasets and models used are publicly available (ELI5, CaLMQA, OpenR1-Math-220k, s1K, Wikipedia, standard benchmarks, GPT-OSS-20B).

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Generation parameters (temperature, top-p, max tokens), watermark parameters (k, α, SynthID depth), detection parameters (entropy proxy model, deduplication), and learnability training details (LoRA rank, learning rate, epochs, quality filter) are specified in Sections 4–6 and Appendix E (Additional Experiments and Details).

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: All watermark detection results are p-values from frequentist tests (Gamma test for Gumbel-max/TextSeal, Z-test for SynthID). The human evaluation uses a two-sided binomial test and TOST equivalence testing with explicit confidence intervals (Section 4.3). Benchmark variance is analyzed with multiple seeds/keys in Appendix E.2 with standard deviations. McNemar's tests are used for multilingual QA (Appendix E.3).

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
   - If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Generation overhead is measured per-token on a single NVIDIA H200 GPU across 5 Qwen 3.5 model sizes (Table in Appendix E.4). Detection efficiency is reported in ms/token and GB of VRAM (Section 5.2, Figure 4). Learnability experiments (teacher generation) use $4\times$H200 GPUs with tensor parallelism (Appendix E.5).

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

Justification: The work concerns watermarking for provenance and transparency, aligning with responsible AI goals. No deceptive practices, privacy violations, or harmful applications are involved. Human evaluators were qualified professionals paid above minimum wage.

Guidelines:

- The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: Positive impacts are discussed throughout: regulatory compliance (EU AI Act), provenance tracking, distillation detection. The Limitations paragraph acknowledges the diversity cost. A potential negative impact is that watermark detection could be used to falsely accuse human-written text of being AI-generated; we mitigate this by providing rigorous FPR calibration (Section 5.1) and formal p-value guarantees.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: The paper does not release pre-trained models, datasets, or other assets with a risk of misuse. The contribution is a text watermarking algorithm, which is inherently a safeguard mechanism supporting responsible AI deployment.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.

- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: All datasets (ELI5, CaLMQA, OpenR1-Math-220k, s1K, Wikipedia, GSM8K, AIME, MATH, HumanEval, MBPP, ARC, GPQA, HellaSwag, MMLU, SimpleQA, WinoGrande) and models (Qwen 3.5, DeepSeek-R1) are cited with their original publications. All of these assets are publicly available under permissive licenses.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: The paper does not release new datasets, models, or code assets.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [Yes]

Justification: The human evaluation (Section 4.3, Appendix E.6) describes the annotation protocol: 3 annotators per sample via Appen, with qualifications (post-graduate education, native fluency, 2+ years experience), four response options, and majority vote aggregation. The full system prompt and task description are provided in the paper. Annotators were compensated above minimum wage through the Appen platform.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A]

Justification: The human evaluation involves only preference judgments on AI-generated text (choosing which of two answers is better). This poses no risk to participants and does not constitute human subjects research requiring IRB approval.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [Yes]

Justification: LLMs are central to this work: Qwen 3.5-27B is the primary generation model for watermark embedding and evaluation, Qwen 3.5-0.8B serves as the entropy proxy for detection, and DeepSeek-R1-Distill-Qwen-14B is the teacher model for learnability experiments. All models and their usage are described in Sections 4–6.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.