PROVENANCELAB: AN AUDITABLE LONGITUDINAL CORPUS FOR LLM WATERMARK TRANSITION ANALYSIS

Anonymous authors
Paper under double-blind review

ABSTRACT

Major providers began embedding machine-readable watermarks in large language model (LLM) text output during 2026 amid the implementation of the EU AI Act's synthetic-content marking requirements, but for many pre-existing commercial models the exact date at which outputs on a particular serving surface begin carrying a watermark is not publicly disclosed, and marking is being rolled out gradually. This creates a narrow and irreproducible window: outputs a model produced before it was marked cannot be regenerated afterward. We present ProvenanceLab, a longitudinal, externally auditable corpus of Claude, GPT-5.6, and Gemini outputs collected across this deployment boundary. Each record carries cryptographic integrity hashing and, for the strongest cohort, service-generated cloud evidence (invocation logs, provider request identifiers), retention-protected storage, and external RFC 3161 timestamps, so that the record of what a model emitted during the transition survives in a form a third party can audit. We further validate the statistical instrument that a future analysis will use: a controlled watermark on/off calibration experiment with the SynthID-Text logits processor on Gemma-2B-IT (1,460 generations) shows, under prompt-clustered inference and prompt-disjoint held-out evaluation, that pre-specified lexical and bigram diversity features register a known watermark with reproducible on/off effects (Cliff's $\delta \approx +0.15$; 95% cluster-bootstrap CI excluding zero) that strengthen in longer outputs ($\delta \approx +0.26$ for 250–500-word outputs) and persist on held-out prompts. The corpus, the provenance architecture, and the calibrated instrument are the contribution and are complete in this paper; a provider-supported verifier, when one becomes available, supplies the final retrospective label. We do not claim watermark detection or attribution for any commercial model from text statistics.

1 INTRODUCTION

The EU AI Act, in Article 50(2), requires providers of generative AI systems to mark synthetic text, audio, image, and video content in a machine-readable form that is detectable as artificially generated, subject to technical-feasibility qualifications (European Parliament and Council, 2024). The general obligation applies from 2 August 2026, and under Article 111(4) systems placed on the market before that date have until 2 December 2026 to comply. In response, one major provider announced that models launched on or after 2 August 2026 support text watermarking at launch and that earlier models are being retrofitted over the following months, using an approach derived from SynthID-Text (Dathathri et al., 2024; Anthropic, 2026); another has described text-watermarking research without announcing deployment.

This regulatory transition creates an unusual measurement problem. The watermark state of a given commercial model changes at a date that is neither published nor observable from outside, and for pre-existing models the change is gradual. Once a model begins to be marked, its earlier, unmarked outputs cannot be collected again: the serving stack behind a fixed model name is known to change silently over time (Chen et al., 2024), and the natural experiment of observing the same model under two watermark states expires. The scientific value of the surrounding period is therefore tied to capturing observations while the answer is still open and preserving them in a form that a third party can later verify.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

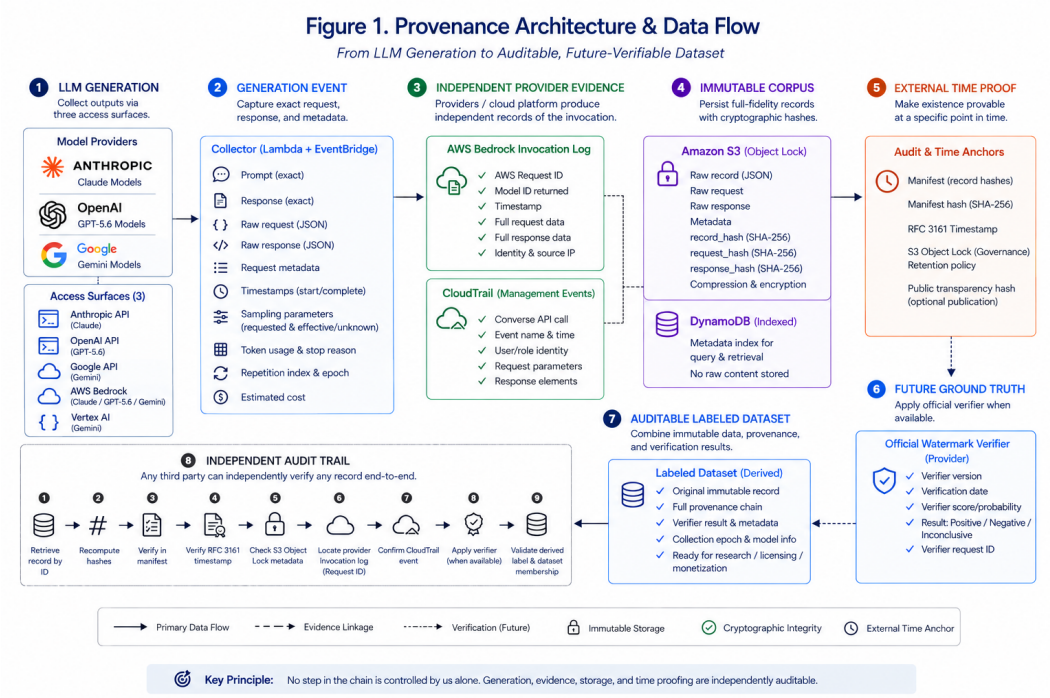


Figure 1: Provenance architecture and data flow, from generation to an auditable, future-verifiable dataset. Each generation event is captured with its exact request, response, and metadata; the managed cloud inference platform produces independent invocation and management-event logs; records are persisted with cryptographic hashes under write-once storage and anchored with an external RFC 3161 timestamp. A future provider verifier label can be attached retrospectively. Publicly available artifacts (hashes, the dated manifest, and the RFC 3161 proof) are verifiable by anyone; the provider log layer is verifiable by a party with access to those logs. No single stage of the chain is controlled by the collector alone. The optional public manifest commit used for supplementary time evidence is withheld during review to preserve anonymity and released with the de-anonymized version. The panel is schematic and illustrates representative access surfaces and providers rather than an exhaustive list; the authoritative set of models and surfaces is the registry in Table 3.

We present ProvenanceLab, a system and corpus built for exactly this window (Figure 1). Our contributions are threefold. First, we describe a longitudinal, externally auditable corpus of commercial LLM outputs collected across the deployment boundary, with per-record provenance whose strength is explicitly labeled by access surface. Second, we describe a provenance and temporal architecture that keeps the corpus useful once provider verification exists, separating immutable raw records from future derived labels and fixing a train/development/test partition rule in advance. Third, we validate the statistical instrument that the eventual analysis will depend on, using a controlled watermark on/off calibration experiment under known ground truth, and report effect sizes with cluster-bootstrap uncertainty and prompt-disjoint held-out evaluation. These three contributions are complete within this paper; a provider-supported verifier is the final labeling step that can be applied to the preserved records once one exists, not a precondition for what the corpus, architecture, and calibration already establish.

We frame the study around two hypotheses. The first is that the activation of a statistical text watermark can induce detectable changes in one or more observable response-level statistics under fixed model, prompt, and access-surface conditions; we operationalize this falsifiably in the controlled calibration by requiring that at least one pre-specified feature show a positive on/off effect whose held-out cluster-bootstrap confidence interval excludes zero. The second is that a longitudinal corpus with frozen prompts, multiple models, and externally anchored time provenance can retrospectively stratify outputs by verifier-assigned watermark status once a provider verifier becomes available, enabling before-and-after comparisons that cannot be constructed after the transition has passed. An

earlier hypothesis about reconstructing the latent token-selection preference by Markov chain Monte Carlo was scope-limited because production logits are unavailable and repeated generations diverge in context after the first stochastic token.

We are deliberate about scope. We do not claim to detect or attribute a watermark in any commercial model from text statistics, and the corpus does not presently establish the watermark state of Claude, GPT-5.6, or Gemini. What the calibration establishes is that the features we intend to use are sensitive to a known SynthID-family watermark when one is present, which is a necessary precondition for interpreting any future signal. Taken together, the resource, its provenance, and the validated instrument make it possible to answer the watermark-transition question when a verifier becomes available. Collecting the corpus now is not merely convenient but necessary: the pre-marking outputs are temporally irreproducible, so the only scientifically valid way to construct the before-and-after comparison later is to have preserved the observations, with auditable provenance, while the window is still open. At that point retrospective labeling of temporally irreproducible observations becomes possible.

2 Related Work

Several strands of prior work inform this study. On watermark construction, Kirchenbauer et al. (2023) introduced the green-list/red-list framework for LLM text watermarking, and Dathathri et al. (2024) developed SynthID-Text, a production scheme using Tournament sampling and context-dependent g-functions deployed on Gemini. Ye et al. (2026) introduce a complementary semantic watermark embedded in Abstract Meaning Representation structures rather than token-selection preferences, and Gloaguen et al. (2026) derive most existing schemes from a constrained optimization problem that exposes a quality-diversity-power trade-off. Omidi et al. (2026) provide the first theoretical analysis of SynthID-Text and design attacks on mean-score detection.

On detection and robustness, post-hoc detectors that require no watermark have been studied extensively: Mitchell et al. (2023) detect machine-generated text from the curvature of a model's log-probability function, and Yang et al. (2024) propose a training-free detector based on divergent n-gram regeneration, a signal related to the lexical and bigram diversity features we study. Sadasivan et al. (2023) demonstrate fundamental challenges in reliably detecting AI-generated text under paraphrase and spoofing, Han et al. (2025) report degradation of SynthID under targeted modification, and Tamim & Khan (2026) find that tested schemes lose their signal after meaning-preserving paraphrase in nearly all cases when evaluated against forensic admissibility criteria. Liang et al. (2025) build a unified platform to evaluate watermarkers under a broad battery of attacks and report that robustness varies sharply across schemes, and Pan et al. (2025) show that watermark signals inherited through knowledge distillation can be removed by paraphrasing or inference-time neutralization. These robustness limits reinforce our decision to treat any observed distributional shift as suggestive rather than dispositive and to defer confirmation to a provider verifier.

Most directly relevant to our work is the literature on black-box detection and third-party verification. Gloaguen et al. (2025) develop statistical tests that detect watermark presence from limited black-box queries on commercial models, and Wang et al. (2026) introduce a third-party verification framework motivated by the barrier that secret-key schemes create for independent auditing. Li et al. (2026) distinguish detection, attribution, extraction, and localization as separate forensic capabilities with different sample complexity. Phan et al. (2026) survey deployment and adoption.

A related line of work in LLM security concerns generation-time mechanisms and their observable effects on outputs. Xiang et al. (2024) show that inference-time chain-of-thought prompting can induce targeted behavioral change under black-box access, and Xu et al. (2024) modify token-selection probabilities during generation to suppress unsafe responses. These security objectives differ from watermarking, but they demonstrate that generation-time interventions produce measurable changes in model outputs, the class of signal that a statistical text watermark induces and that our features are designed to detect.

Prior work primarily studies watermark construction, robustness, or detection under controlled or model-query settings. ProvenanceLab addresses a complementary problem: preserving longitudinal, externally time-anchored observations of commercial model outputs across an unknown deployment boundary, so that future provider-supported verification can be applied retrospectively.

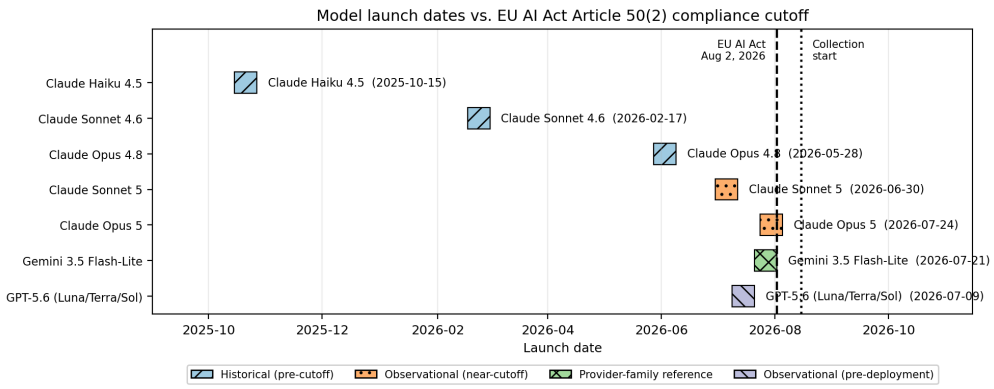


Figure 2: Model launch dates relative to the EU AI Act Article 50(2) compliance cutoff (2 August 2026) and the start of collection. All observed models were released before the cutoff, so the corpus captures their outputs during the transition period in which their watermark state is undetermined.

Content-provenance standards such as C2PA pursue a related goal for media assets, but an independent security analysis finds that current C2PA specifications fall short of their claimed guarantees and should not yet be relied on for high-stakes use (Golaszewski et al., 2026); our provenance architecture therefore draws on cloud service-generated evidence and write-once storage to reduce reliance on self-reported claims and on any single trust anchor. Additional detail on model cohorts, provenance layers, infrastructure, and the collection protocol appears in the appendix.

3 Corpus and Provenance Design

3.1 Cohorts and Epistemic Labeling

The corpus registry spans nine models across three providers, of which seven are populated with observations at the snapshot reported here; the two remaining models (GPT-5.6 Terra and Sol) were registered for forward compatibility but were still at pilot or zero records, which accounts for the nine-registered, seven-populated figures used elsewhere in the paper (see Appendix A for the full registry). Five Claude models (Haiku 4.5, Sonnet 4.6, Opus 4.8, Sonnet 5, Opus 5) and one Gemini model were launched before the 2 August 2026 cutoff; three GPT-5.6 models are included as an observational cohort. These are real production models queried on their live serving surfaces, and every date, count, and effect reported here comes from actual collected records rather than simulation. Every model observed here was released before the compliance cutoff (Figure 2), which is precisely why their pre-marking outputs can only be captured during the current window.

We deliberately avoid labeling any model as watermarked or unwatermarked. Instead, each record carries an epistemic status label that records only what is verifiably known: pre-rollout observation for near-cutoff models, legacy-unknown for older models under gradual retrofit, provider-family-claimed for Gemini (where the provider claims a SynthID family but the specific API surface is unverified by us), and a label for providers that have deployed watermarking in other modalities but not announced text watermarking. We refer to the current material as an initial observation wave rather than a pre-watermark baseline, because we do not yet know which outputs are watermark-negative.

3.2 Provenance by Access Surface

Provenance strength is not uniform across the interfaces through which outputs are obtained, and we label each record accordingly. Records obtained via the AWS Bedrock managed inference surface used for Claude carry an AWS-corroborated evidence chain: service-generated invocation logs, CloudTrail management-event logs, provider request identifiers, and per-request metadata linkage. Records for the GPT-5.6 cohort obtained through the Bedrock Mantle endpoint carry a partial class: integrity hashes, write-once retention, and external timestamps, but no invocation-log corrobora-

tion. Records obtained through a provider REST API carry a self-recorded class: our timestamps and hashes only. Every record is hashed with SHA-256 over prompt, response, and canonicalized metadata; batches are periodically committed to a manifest that is timestamped with an RFC 3161 trusted-timestamp authority and stored under write-once retention (Figure 1). The full evidence-layer table, hashing specification, and an end-to-end independent verification protocol appear in the appendix. Naming a specific cloud platform describes the provenance mechanism and does not identify the authors.

3.3 Identification Strategy

The strongest available natural experiment fixes the model, access surface, and prompt set and varies only time, asking whether the distribution of observable text features undergoes a persistent change. We treat within-model longitudinal comparison as the primary strategy because it holds model identity fixed and removes the largest confound; a provider verifier, when available, supplies the label. For models whose decoding configuration is provider-controlled and potentially time-varying, a silent serving change coinciding with watermark activation is an unresolvable confound without provider cooperation, so we treat a detected shift as evidence consistent with watermark activation rather than proof, and rely on the eventual verifier to disambiguate. Cross-model comparison is secondary and confounded by model style. Prompts are frozen and stratified by qualitative expected watermark opportunity, and the corpus deliberately over-samples long-form outputs (creative prose, extended essays, full translations) because both the provider description and our own calibration indicate that the recoverable signal strengthens with output length; this is where continued collection effort is directed. All features are computed at multiple length strata so that a candidate signal present only in long outputs is not obscured by short ones.

This strategy is already operational on the collected records, which is what makes the corpus usable before any verifier exists. Comparing each legacy Claude model against its own earlier baseline on the same frozen prompts and the same AWS-corroborated Bedrock surface, holding model identity fixed and varying only time, we find no coherent, length-consistent watermark-signature shift on any credibly comparable model at the snapshot reported here. This clean null is the expected outcome given the provider's statement that legacy models are being retrofitted over subsequent months, and it illustrates the analysis the corpus supports today rather than a claim about watermark state. The comparison also surfaced a documented within-model serving change unrelated to watermarking: one model that answered directly at the earlier baseline later produced extended internal reasoning that consumed the token budget, which alters response statistics independently of any watermark and which we therefore excluded from the within-model test rather than reporting as a shift. That exclusion is itself a concrete demonstration of why fixed-model, fixed-prompt provenance is necessary to keep silent serving changes from masquerading as signal.

4 Controlled Watermark Calibration Experiment

4.1 Setup and Success Criteria

Before interpreting any subtle distributional change in a commercial model as a watermark, we ask whether the intended features can register a watermark that is definitely present under controlled ground truth. We generate text with Gemma-2B-IT under watermark on and off conditions using the SynthID-Text Tournament-sampling logits processor (Dathathri et al., 2024; Google DeepMind, 2024) (`ngram_len`$= 5$, 30 keys from the official demo, `sampling_table_size`$= 2^{16}$), over the same 73 frozen canonical prompts, 10 repetitions per condition, for 1,460 generations. Success is defined in advance: at least one pre-specified feature must show a reproducible on/off effect on prompt-disjoint held-out prompts, with a cluster-bootstrap confidence interval excluding zero and with direction and magnitude reported by length stratum. Multiplicity across eight features is controlled with a Benjamini–Hochberg false-discovery-rate (FDR) procedure at $q = 0.05$.

The 1,460 generations are not independent: the ten repetitions that share a prompt are correlated. We therefore compute uncertainty with a cluster bootstrap over the 73 prompts, and permutation tests randomize the watermark condition within each prompt block, preserving the matched paired-prompt design. All statistics use a fixed seed with 5,000 resamples, and are reproducible from the released generation files.

Table 1: Overall watermark on/off feature effects across all 1,460 generations, sorted by magnitude. Confidence intervals are prompt-level cluster bootstrap. The "FDR sig." column reports the Benjamini–Hochberg decision at $q = 0.05$ using within-prompt permutation p-values; the "CI excl. 0" column reports whether the cluster-bootstrap interval excludes zero. Mean sentence length is FDR-significant by permutation but its bootstrap interval marginally includes zero, and mean word length shows the reverse, which is why we report the two criteria separately rather than as a single flag.

Feature	**ON mean**	**OFF mean**	**Cliff's δ**	**95% CI**	**FDR sig.**	**CI excl. 0**
bigram diversity	0.856	0.831	$+0.155$	$[+0.112, +0.203]$	yes	yes
lexical diversity	0.559	0.531	$+0.154$	$[+0.112, +0.204]$	yes	yes
word entropy	5.855	5.759	$+0.069$	$[+0.047, +0.099]$	yes	yes
comma frequency	0.055	0.053	$+0.060$	$[+0.020, +0.100]$	yes	yes
mean word length	5.205	5.174	$+0.050$	$[+0.021, +0.084]$	no	yes
mean sentence length	17.00	16.14	$+0.032$	$[-0.005, +0.070]$	yes	no
char entropy	4.320	4.299	$+0.018$	$[-0.008, +0.047]$	no	no
word count	220.8	224.6	-0.018	$[-0.042, +0.003]$	no	no

Table 2: Cliff's δ for the two largest features, stratified by output length. Both grow with length, peaking at 250–500 words.

Length stratum (words)	**n (on+off)**	**lexical div. δ**	**bigram div. δ**
0–50	217	$+0.106$	$+0.075$
50–100	118	$+0.069$	-0.083
100–250	425	$+0.164$	$+0.183$
250–500	700	$\mathbf{+0.265}$	$\mathbf{+0.256}$

4.2 Results

Table 1 reports the overall on/off effects. We call a feature significant only when it satisfies both criteria we fixed in advance: it is Benjamini–Hochberg significant under the within-prompt permutation test at $q = 0.05$ *and* its prompt-level cluster-bootstrap confidence interval for Cliff's δ excludes zero. Four of eight features meet this combined bar (bigram diversity, lexical diversity, word entropy, and comma frequency). A fifth, mean sentence length, is Benjamini–Hochberg significant by permutation but its bootstrap interval $[-0.005, +0.070]$ marginally includes zero, so we do not count it as a reliable effect; conversely mean word length has an interval excluding zero but is not permutation-significant. We report the two criteria in separate columns rather than as a single flag precisely because they can disagree at the margin, and treat only their conjunction as significant. The two largest features are bigram diversity and lexical diversity, each about twice the next-largest effect. All significant effects are positive, indicating greater lexical and bigram diversity under the on condition, consistent with a watermark perturbation of token-selection probabilities. We report Cliff's δ, a nonparametric effect size in $[-1, +1]$; positive values mean the feature is larger under the on condition.

The effect is not constant across output lengths. Table 2 shows that the two largest features grow with output length, reaching their largest separation at 250–500 words, roughly two to three times the effect seen in short outputs. The growth is not strictly monotonic: bigram diversity turns slightly negative in the 50–100-word stratum ($\delta = -0.083$), the smallest stratum at $n = 118$, where the estimate is noisy and its bootstrap interval spans zero, so we read it as sampling variation rather than a reversal of the effect. This is broadly consistent with the provider statement that detectability increases with more generated tokens, and it directs continued collection toward long-form prompts. Because the strata are formed from realized output length and a watermark could in principle affect length, this stratification can introduce a selection effect; the near-zero overall word-count effect ($\delta = -0.018$, CI $[-0.042, +0.003]$) limits its severity here, but we treat the pattern as suggestive rather than confound-free and discuss a matched-length sensitivity design in Appendix E.

The eight candidate features and the 52/21 prompt partition were fixed before the held-out outcomes were examined; no feature selection, threshold selection, or classifier tuning used the 21 held-out prompts. To guard against a classifier memorizing prompt-specific style, we split the data so that no prompt appears in both training and test sets: a logistic regression on the eight features trained on 52 prompts and evaluated on 21 completely held-out prompts (420 generations) achieves ROC-AUC $= 0.591$ with a prompt-level cluster-bootstrap 95% CI of $[0.551, 0.643]$, which excludes chance. The relevant benchmark for a calibration is 0.5, not the near-1.0 AUC one would demand of a deployable detector: the claim being tested is only whether the feature vector carries any out-of-prompt information about a treatment we control, so a CI that excludes 0.5 is exactly the right bar and 0.591 clears it, while remaining far too low to serve as a detector. The classifier is intentionally not presented as a practical detector; its modest AUC provides only that independent check, which is what we claim and not more. Recomputing the per-feature effects on the 21 held-out prompts alone, and applying FDR across all eight features on that set, lexical and bigram diversity remain significant with CIs excluding zero, so the effect is not an artifact of the prompts used during feature selection; per-feature held-out effect sizes, confidence intervals, and adjusted significance are reported in Appendix E.

4.3 What This Establishes

The pre-specified primary criterion is met, and specifically on the held-out prompts. The calibration shows that our features are sensitive to a known SynthID-family watermark when one is present under controlled ground truth. It establishes nothing about the watermark state of any commercial model: those models are larger, observed through black-box APIs, use unknown keys and configurations, and may change serving behavior silently. Gemma-2B-IT is much smaller than production systems, so absolute magnitudes here are not expected to transfer directly. The calibration is a necessary precondition for interpreting a future commercial signal, and we intend it as such rather than as evidence of one.

5 Discussion and Conclusion

ProvenanceLab began from an observation about timing. The EU AI Act created a moment in which large commercial language models would begin to mark their text output, yet for many pre-existing models the exact date at which outputs on a particular serving surface begin carrying a watermark is not publicly disclosed, and marking is being rolled out gradually. That uncertainty defines a limited observation window, because the outputs a model produces before it is marked cannot be regenerated afterward. Our response was to observe these models continuously and preserve what we see in a form a third party can later verify, so that the question of whether a given output predates watermarking can be answered with evidence rather than recollection.

The corpus reflects that intent. It records outputs from seven populated models across three providers, anchors each record with cryptographic hashes and, where the interface allows, with service-generated logs and externally issued timestamps, and protects the stored material against silent alteration. Alongside the corpus, the controlled calibration shows under known ground truth that the features we plan to use are sensitive to a SynthID-style watermark, that this sensitivity survives evaluation on prompts held out from feature development, and that it strengthens in longer passages as the provider's own description predicts. The resource, the provenance architecture, and the validated instrument are finished pieces of work, not placeholders waiting on an external event; each stands on its own and each was necessary to attempt now, because the observations they preserve cannot be recovered once the transition has passed.

We are restrained about what has been demonstrated. We do not determine the watermark state of any commercial model from text statistics, and nothing in the present corpus should be read as establishing that a commercial model is or is not watermarked. What the work provides is the infrastructure, the auditable provenance, and the calibrated measurement that make the question answerable at all. A provider verifier is the final labeling step in that pipeline rather than its scientific core: when one arrives, its labels can be applied retrospectively to observations captured while the answer was still open, producing a before-and-after picture that could not be assembled after the transition had already passed. The value of the present paper does not depend on when that step happens, only on having preserved the evidence and validated the instrument before the opportunity

closed. Continued collection through the transition period, expansion of the prompt inventory and the range of languages, and adoption of a verifier interface when it becomes available are the natural next steps, and each is supported by the design described here.

5.1 LIMITATIONS

Several limitations bound what the corpus and calibration currently support. There are no provider-verified watermark labels yet, so the corpus preserves observations whose watermark status is genuinely unknown rather than a labeled benchmark; its value is realized when a verifier becomes available. The identification strategy assumes that the only provider-side change over time is watermark state, but for models whose decoding configuration is provider-controlled and possibly time-varying, a silent serving change coinciding with watermark activation cannot be separated from the watermark without provider cooperation. Provenance strength differs across access surfaces, and self-recorded records carry weaker corroboration than AWS-corroborated ones; we label this per record rather than averaging it away. The repeated-sampling instrument supports whole-response distributional features but not per-token reconstruction, because conditioning contexts diverge across repetitions after the first stochastic token. The calibration model, Gemma-2B-IT, is much smaller than production systems, and the write-once retention used for storage is bypassable by a sufficiently privileged principal, so the provenance chain protects most strongly against external tampering and accidental loss. Finally, the corpus is English-dominated and uses a frozen 73-prompt panel, which bounds ecological validity; expanding the panel and language coverage is planned.

ETHICS STATEMENT

This work studies provider watermarking and content provenance, topics with direct governance and transparency relevance under the EU AI Act. The experimental corpus contains model-generated responses to a fixed set of neutral prompts and does not intentionally collect human-subject or personal data; operational provenance artifacts (such as service logs) are sanitized before any release to remove account- or user-identifying metadata. We do not attempt to reverse engineer provider watermark keys or configurations, to extract or reconstruct proprietary models, or to train competing models; the design explicitly excludes these. We are careful not to assign confirmed watermark labels to any commercial model, because doing so from text statistics alone would be unsupported and could mislead downstream users, regulators, or the public. Provenance strength is labeled per record so that no reader overstates what a given observation can support. Any future release of the corpus will preserve these epistemic labels and the distinction between raw observations and derived verifier annotations.

REPRODUCIBILITY STATEMENT

The controlled calibration experiment is fully specified for reproduction across Section 4, Appendix E, and the accompanying anonymized supplementary artifacts. The model, watermark configuration, prompt set, repetition count, and random seed are given in Section 4, and the statistical procedure (prompt-level cluster bootstrap, within-prompt permutation, Benjamini–Hochberg FDR, and the 52/21 prompt-disjoint classifier split) is described there and expanded in the appendix. All reported numbers are computed from the released watermarked and unwatermarked generation files with a fixed seed. The corpus schema, hashing canonicalization, and the independent verification protocol needed to audit any record are given in the appendix, and any public release will follow established dataset-documentation practice (Gebru et al., 2021). Anonymized code and generation files are provided as supplementary material.

AI USE STATEMENT

We used generative AI tools to assist with drafting and editing prose and with routine code scaffolding for the collection and analysis pipeline. We did not use generative AI to produce experimental results, statistical analyses, or reported numbers, which were computed by the described pipeline. All AI-assisted text and code were reviewed and verified by the authors, who take responsibility for the final content of this work.

REFERENCES

Ryan P. Adams and David J. C. MacKay. Bayesian online changepoint detection. *arXiv preprint arXiv:0710.3742*, 2007.

Anthropic. How claude's text watermark works, 2026. Provider announcement, August 14, 2026.

Lingjiao Chen, Matei Zaharia, and James Zou. How is ChatGPT's behavior changing over time? *Harvard Data Science Review*, 6(2), 2024.

Sumanth Dathathri, Abigail See, Sumedh Ghaisas, Po-Sen Huang, Rob McAdam, Johannes Welbl, et al. Scalable watermarking for identifying large language model outputs. *Nature*, 634:818–823, 2024.

European Parliament and Council. Regulation (eu) 2024/1689 (artificial intelligence act), article 50(2); as amended by regulation (eu) 2026/1744 adding article 111(4), 2024.

Timnit Gebru, Jamie Morgenstern, Briana Vecchione, Jennifer Wortman Vaughan, Hanna Wallach, Hal Daumé III, and Kate Crawford. Datasheets for datasets. *Communications of the ACM*, 64 (12):86–92, 2021.

Thibaud Gloaguen, Nikola Jovanović, Robin Staab, and Martin Vechev. Black-box detection of language model watermarks. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2025.

Thibaud Gloaguen, Nikola Jovanović, Robin Staab, and Martin Vechev. A unified framework for LLM watermarks. *arXiv preprint arXiv:2602.06754*, 2026.

Enis Golaszewski, Neal Krawetz, Alan T. Sherman, Edward Zieglar, Sai K. Matukumalli, Roberto Yus, Carson L. Kegley, Michael Barthel, William Bowman, Bharg Barot, and Kaur Kullman. Verifying provenance of digital media: Why the C2PA specifications fall short. *arXiv preprint arXiv:2604.24890*, 2026.

Google DeepMind. Synthid-text: open-source reference implementation. Software repository, 2024.

X. Han, Q. Li, J. Ni, and M. Zulkernine. Robustness assessment and enhancement of text watermarking for Google's SynthID. *IEEE TrustCom*, pp. 942–949, 2025.

John Kirchenbauer, Jonas Geiping, Yuxin Wen, Jonathan Katz, Ian Miers, and Tom Goldstein. A watermark for large language models. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2023.

X. Li, Z. Gao, X. Feng, J. Jiang, Y. Sui, and J. Hu. Watermark forensics for generative models: An information-theoretic perspective. *arXiv preprint arXiv:2607.13003*, 2026.

Jiacheng Liang, Zian Wang, Spencer Hong, Shouling Ji, and Ting Wang. Watermark under fire: A robustness evaluation of LLM watermarking. In *Findings of the Association for Computational Linguistics: EMNLP*, 2025.

Eric Mitchell, Yoonho Lee, Alexander Khazatsky, Christopher D. Manning, and Chelsea Finn. DetectGPT: Zero-shot machine-generated text detection using probability curvature. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2023.

R. Omidi, Y. Dong, and B. Wang. On Google's SynthID-Text LLM watermarking system: Theoretical analysis and empirical validation. *arXiv preprint arXiv:2603.03410*, 2026.

Leyi Pan, Aiwei Liu, Shiyu Huang, Yijian Lu, Xuming Hu, Lijie Wen, Irwin King, and Philip S. Yu. Can LLM watermarks robustly prevent unauthorized knowledge distillation? *arXiv preprint arXiv:2502.11598*, 2025.

H. Phan, K. Dang, O. Dulal, A. AL Shukairi, A. Shine, C. Garner, and P. Lai. A survey on LLM watermarking: Theory and deployment. *arXiv preprint arXiv:2607.10103*, 2026.

Vinu Sankar Sadasivan, Aounon Kumar, Sriram Balasubramanian, Wenxiao Wang, and Soheil Feizi. Can AI-generated text be reliably detected? *arXiv preprint arXiv:2303.11156*, 2023.

S. R. Tamim and A. L. Khan. AI watermark evidence fails forensic readiness: An empirical evaluation. *arXiv preprint arXiv:2607.16010*, 2026.

Z. Wang, Y. Ren, Y. Cao, F. Fang, X. Li, and L. Guo. Rethinking LLM watermark detection in black-box settings: A non-intrusive third-party framework. *Findings of the Association for Computational Linguistics (ACL)*, pp. 19773–19790, 2026.

Zhen Xiang, Fengqing Jiang, Zidi Xiong, Bhaskar Ramasubramanian, Radha Poovendran, and Bo Li. BadChain: Backdoor chain-of-thought prompting for large language models. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2024.

Zhangchen Xu, Fengqing Jiang, Luyao Niu, Jinyuan Jia, Bill Yuchen Lin, and Radha Poovendran. SafeDecoding: Defending against jailbreak attacks via safety-aware decoding. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, 2024.

Xianjun Yang, Wei Cheng, Yue Wu, Linda Petzold, William Yang Wang, and Haifeng Chen. DNA-GPT: Divergent n-gram analysis for training-free detection of GPT-generated text. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2024.

Zhiyuan Ye, Guhan Dey, Christos Christodoulopoulos, Charith Peris, Anil Ramakrishna, Wei Ruan, Aram Galstyan, Kai-Wei Chang, Rahul Gupta, and Ninareh Mehrabi. SWAN: Semantic watermarking with abstract meaning representation. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 36304–36315, 2026.

This appendix supplies the material referenced from the main text: the model registry (Appendix A), the provenance evidence layers, hashing canonicalization, and independent verification protocol (Appendix B), the record schema (Appendix C), the prompt inventory (Appendix D), and the full calibration reproducibility details including feature definitions, the statistical procedure, and the held-out effect table (Appendix E). Reviewers are not required to read the appendix.

A Model Registry

The registry contains nine models across three providers. Seven are populated with observations; two (GPT-5.6 Terra and Sol) were registered but remained at pilot/zero records at the snapshot reported here, which is the source of the "nine registered, seven populated" distinction in the main text. All launch dates precede the 2 August 2026 compliance cutoff.

Provider	Model	Public launch	Cohort	Populated
Anthropic	Claude Haiku 4.5	2025-10-15	historical (pre-cutoff)	yes
Anthropic	Claude Sonnet 4.6	2026-02-17	historical (pre-cutoff)	yes
Anthropic	Claude Opus 4.8	2026-05-28	historical (pre-cutoff)	yes
Anthropic	Claude Sonnet 5	2026-06-30	observational (near-cutoff)	yes
Anthropic	Claude Opus 5	2026-07-24	observational (near-cutoff)	yes
Google	Gemini 3.5 Flash-Lite	2026-07-21	provider-family reference	yes
OpenAI	GPT-5.6 Luna	2026-07-09	observational (pre-deployment)	yes (pilot)
OpenAI	GPT-5.6 Terra	2026-07-09	observational (pre-deployment)	no (registered)
OpenAI	GPT-5.6 Sol	2026-07-09	observational (pre-deployment)	no (registered)

Table 3: Full model registry. The "Public launch" column records each model's public general-availability date. The GPT-5.6 family reached general availability on 2026-07-09; our corpus first observed it on the AWS Bedrock serving surface we collect from on 2026-07-13, and we record that surface-observed date separately in the record metadata, consistent with our practice of labeling provenance by access surface. The two unpopulated OpenAI models were registered for forward compatibility but not yet collected at the reported snapshot; the corpus counts in the main text refer to the seven populated models.

Each record carries an epistemic status label: `legacy_unknown` (pre-compliance model, possibly retrofitted, no confirmation either way), `pre_rollout_observation` (released before the cutoff, status genuinely ambiguous), `provider_family_claimed` (provider claims a watermark family on the model family, but the specific API text surface is unverified by us), and a

label for providers that have deployed watermarking in other modalities without an announced text watermark. We never assign a `watermarked`/`unwatermarked` label from statistics; verifier-confirmed labels (`verifier_positive`/`negative`/`inconclusive`) are reserved for the future when a provider detection interface exists.

B PROVENANCE: EVIDENCE LAYERS, HASHING, AND VERIFICATION

B.1 EVIDENCE LAYERS AND PROVENANCE CLASSES

Provenance strength is heterogeneous and is labeled per record by access surface. Table 4 lists the evidence layers and what each does and does not prove. We deliberately distinguish four kinds of guarantee that should not be conflated: cryptographic integrity (hashes prove content was not altered after hashing, but not time), retention control (write-once storage resists deletion but is administratively bypassable, so it is a control rather than a mathematical guarantee), independent time evidence (an external RFC 3161 authority attests a manifest existed before a time), and provider-generated evidence (service logs the collector does not author). "Immutable" in the main text refers to this layered combination, not to mathematical impossibility of alteration.

Layer	Guarantee type	What it establishes
Provider invocation log	provider-generated	Independent service record of full request/response with a service timestamp and request ID.
Provider management-event log	provider-generated	Log of each inference API call (event name, time, caller identity).
Provider request ID	provider-generated	Joins our record to the provider's service log.
SHA-256 record/prompt/response hashes	cryptographic integrity	Content has not changed since hashing; does not prove time.
RFC 3161 trusted timestamp	independent time	External authority attests the manifest existed before a time T.
Write-once retention (governance mode)	retention control	Resists deletion under ordinary permissions; bypassable by a sufficiently privileged principal.
Public manifest commit (optional; withheld during review)	independent time (supplementary)	Manifest hash visible with an external commit timestamp; withheld during review to preserve anonymity and released with the de-anonymized version.
Provider detection verifier (future)	provider-generated label	Watermark likelihood score, when a verifier becomes available.

Table 4: Provenance evidence layers grouped by guarantee type. The three provenance classes are: cloud-corroborated (managed inference surface with invocation and management-event logs and request-metadata linkage), a partial class (integrity hashes, retention, and external timestamps but no invocation-log corroboration), and self-recorded (our timestamps and hashes only, e.g. a provider REST API).

B.2 HASHING AND CANONICALIZATION

Every record computes

$$H_{\text{record}} = \text{SHA-256}\big(\texttt{prompt} \,\|\, \texttt{0x00} \,\|\, \texttt{response} \,\|\, \texttt{0x00} \,\|\, \texttt{canonical_metadata}\big).$$

The manifest hash is $H_{\text{manifest}} = \text{SHA-256}$ of the record hashes rendered as lowercase hexadecimal, sorted lexicographically ascending, joined by a single newline (`0x0A`) with no trailing newline. Canonicalization is specified exactly so that an independent auditor reproduces identical digests: metadata is parsed from JSON and re-serialized with lexicographically sorted keys and compact separators (`(',', ':')`), UTF-8 encoded, with no insignificant whitespace and no trailing newline; `prompt_hash` and `response_hash` are SHA-256 over the exact UTF-8 bytes of the respective texts with no normalization; the metadata component excludes the `record_hash` field itself, the raw prompt/response text (already in the preimage), and any later-added verifier annotations. The

authoritative ordered field set is versioned and recorded in a `configuration_hash`. We do not apply Unicode normalization; bytes are hashed as produced by the provider. We document this construction as record-hash *schema v1*: the single-null-byte delimiter is uniquely decodable only under the assumption that the prompt, response, and serialized metadata fields do not themselves contain the byte `0x00`, which holds for the text records in this corpus. Because this framing is not delimiter-safe against arbitrary binary content, a future schema v2 will move to explicit length-prefix framing or a hash over a single canonical JSON object; schema v1 hashes are retained unchanged for backward compatibility with already-anchored manifests.

B.3 Independent Verification Protocol

Given a record identifier, an auditor can (1) retrieve the retention-protected raw record; (2) recompute `prompt_hash`, `response_hash`, the request and response payload hashes, and `record_hash` using the canonicalization above; (3) verify record membership in the dated manifest; (4) verify the RFC 3161 token covers the manifest hash; (5) check the write-once retention metadata; (6) for cloud-corroborated records, retrieve the provider invocation-log entry by request ID and confirm the model identifier, timestamp, and input/output match, and confirm the per-request metadata identifier matches; and (7) when a verifier exists, attach and check the derived label against the verified text hash. Because anchoring runs on a daily cadence, a record's external time lower bound during the up-to-24-hour window before its manifest is timestamped rests on the provider log timestamps rather than the external token; a stronger deployment would use non-bypassable retention and per-record timestamping, which we note as future hardening.

C Record Schema

Each record contains core, provenance, verifier-placeholder, and retry fields. Core fields include a UUID sample identifier; request and response UTC timestamps; provider, access surface, and requested/returned model identifiers; prompt identifier, verbatim prompt text, prompt hash, and category; requested and effective sampling parameters (marked `unknown` for models that reject explicit sampling parameters); max tokens; input and output token counts; stop reason; latency; the full raw response text and its hash; the raw request and response payloads and their hashes; the provider request ID; the epistemic watermark-status label and its evidence string; the collection epoch; and an estimated cost. Eight verifier fields are defined but empty, to be populated as a separate derived annotation without mutating raw records. Field availability is access-surface dependent; fields that a surface does not support are null. Raw records are never mutated; derived annotations (including verifier labels) are stored as separate mappings.

D Prompt Inventory

The frozen canonical set (version 1.0) contains 73 prompts stratified by qualitative expected watermark opportunity: creative prose, essays, conversation, and translation (higher opportunity, more interchangeable token choices), and factual question answering and code (lower opportunity, more constrained outputs). A separate constrained probe set of 15 prompts forces the first several response tokens to be highly predictable, enabling shared-context analysis deeper into generation. A distinct daily change-point "canary" panel of fixed prompts monitors distributional shift over time, in the spirit of online change-point detection (Adams & MacKay, 2007), and is not part of the frozen inventory. The canonical set is never modified; new probes are added as separate versioned sets to preserve longitudinal comparability.

E Calibration Reproducibility and Additional Results

E.1 Configuration

The controlled experiment uses Gemma-2B-IT (the instruction-tuned 2B Gemma model) as the base model and the SynthID-Text Tournament-sampling watermark logits processor from the HuggingFace Transformers implementation, configured with `ngram_len`$= 5$, the 30-key watermarking configuration from the official SynthID-Text demonstration, and `sampling_table_size`$= 2^{16}$.

Generation covers the same 73 frozen canonical prompts, 10 repetitions per prompt per condition, for 730 watermarked and 730 unwatermarked generations. Length targets by category are 512 tokens for creative and essay prompts, 256 for factual and code, and 128 for the short repeated prompts. All statistics use a fixed random seed (`seed=` 20260831) with 5,000 resamples for confidence intervals and 5,000 permutations for p-values, so the reported numbers are reproducible from the released watermarked and unwatermarked generation files. The exact model revision, tokenizer and library versions, prompt/chat template, decoding parameters, and the specific watermark-key values are recorded in the released configuration and code accompanying the generation files rather than transcribed here; we provide these artifacts as supplementary material so that the configuration is reproducible from the released files rather than from a paraphrase.

E.2 Feature Definitions

All features are computed on the response text after a fixed tokenization: word tokens are whitespace-separated tokens lowercased and stripped of leading/trailing punctuation from the set `.,;:!?"'()[]{}`. For a response with word tokens $w_1, \ldots, w_n$: lexical diversity is $|\{w_i\}|/n$ (type-token ratio); bigram diversity is the number of distinct adjacent word bigrams divided by the total number of bigrams; word entropy is the Shannon entropy (base 2) of the word-token distribution; character entropy is the Shannon entropy of the character distribution over the raw text; mean word length is the mean token length in characters; mean sentence length is the mean number of whitespace tokens per sentence, where sentences are split on `[.!?]+`; comma frequency is the number of commas divided by the number of whitespace tokens; and word count is n. Cliff's δ is computed as $(\#\{a > b\} - \#\{a < b\})/(|A||B|)$ over the on-condition values A and off-condition values B. We use Cliff's δ only as a descriptive marginal on-versus-off effect size; the matched, prompt-clustered structure of the design is handled entirely by the inference procedure rather than by the effect-size definition. The cluster bootstrap resamples the 73 prompts with replacement, retaining all within-prompt repetitions of each selected prompt; the permutation test randomizes the on/off condition within each prompt block, preserving the matched paired-prompt design; and multiplicity is controlled with the Benjamini–Hochberg procedure at $q = 0.05$ across all eight features.

E.3 Held-Out Analysis and Leakage Control

The eight candidate features and the 52/21 prompt partition were fixed before the held-out outcomes were examined; no feature selection, threshold selection, or classifier tuning used the 21 held-out prompts. The classifier is a logistic regression on the eight standardized features; it is intentionally not presented as a practical detector. Its modest held-out ROC-AUC of 0.591 (95% cluster-bootstrap CI $[0.551, 0.643]$, excluding chance) provides only an independent check that the feature vector carries weak out-of-prompt information about the controlled treatment, and this weak predictive value is expected rather than a shortcoming of the contribution. Table 5 reports the per-feature on/off effects recomputed on the 21 held-out prompts alone, with Benjamini–Hochberg correction applied across all eight features on that set; lexical and bigram diversity remain significant with CIs excluding zero, so the effect is not an artifact of the prompts used during feature selection. On this held-out set the remaining five features had negligible effects that did not survive the Benjamini–Hochberg correction, consistent with their small effects in the full-corpus analysis (Table 1); the complete per-feature held-out outputs, including raw permutation p-values and BH-adjusted q-values, accompany the released analysis artifacts. Table 5 summarizes these held-out effects.

E.4 Length Stratification and Selection Effects

The length strata reported in the main text are formed from realized output length in words. Because a watermark could in principle affect generated length, stratifying on realized length can introduce a selection effect, and the strata have unequal sizes. We report the on/off word-count effect as negligible (Cliff's $\delta = -0.018$, CI $[-0.042, +0.003]$), which indicates the on and off length distributions are close on this corpus and limits the severity of this concern, but we do not treat the length-stratified pattern as confound-free. A stronger design would match or truncate text to equal lengths across conditions, or stratify on target prompt-length category rather than realized length; we note this as a limitation and a direction for the next iteration.

Feature (held-out prompts only)	**Cliff's δ**	**95% CI**	**FDR sig.**
lexical diversity	$+0.165$	$[+0.071, +0.279]$	yes
bigram diversity	$+0.141$	$[+0.072, +0.228]$	yes
word entropy	$+0.079$	$[+0.046, +0.140]$	no

Table 5: On/off effects on the 21 prompt-disjoint held-out prompts (prompt-level cluster bootstrap, within-prompt permutation, Benjamini–Hochberg across all eight features). Lexical and bigram diversity remain significant on prompts that played no role in feature selection. Word entropy is positive with a CI excluding zero but does not survive held-out FDR. The remaining features are not significant on the held-out set.