What Does a Copyright Budget Certify? An Exact Threshold for Per-Token Meters, and a Length-Independent Certificate for Selection

Anonymous authors
Paper under double-blind review

Abstract

Certified inference-time copyright defences publish a divergence budget between the served distribution and a safe model, the anchor, trained without the protected works. First, a budget of K nats says nothing about a work whose surprisal under the safe model is at most K, at every Rényi order. Anchored decoding charges its budget per token, so it grows with the output, and at its authors' setting it is vacuous for all 758 protected passages. Second, spent on choosing among n draws from the safe model, by any rule, an adversary's included, the budget is a pathwise certificate of exactly $\log n$ nats at any output length, small enough that a rights-holder can bound near-verbatim copying by sampling the safe model alone. Third, with the risky model served as a text continuer at temperature 1.0, neither of two judges separates the one metered budget that covers a 50-token window, $k = 0.5$, from the safe model. At matched certificates, best-of-64 and its blockwise variant beat every meter we ran there, under one of two judges once empty answers count as losses; at the authors' temperature 0.7 the meter at $k = 0.5$ ties selection. Where a meter's certificate is void, selection wins against the authors' 70B pair and against their byte-level decoder, whose base model is weak on instruction prompts, but loses to the 8B at their temperature 0.7 or served as a chat assistant. Against the chat assistant a windowed meter wins too, at window budgets too large to audit by sampling, and leaks near-verbatim text at 125 nats, where both judges prefer it. With no judge, majority vote lifts GSM8K from 0.320 to 0.546. Selection never consults the risky model, so no n rescues an anchor that cannot do the task; served from the anchor alone its draws cost $21.8\times$ a metered decode, and it loses at matched compute. Its leakage, zero at any $n \leq 64$ or at 256, is relative to the anchor, which must be vetted, as must the scorer unless queries are capped.

1 Introduction

Language models memorise their training data: `Llama-3.1-70B` reproduces *Harry Potter and the Sorcerer's Stone* almost verbatim from its first few words (Cooper et al., 2025), and retraining that away is impractical against a stream of takedowns. Certified decoders need no retraining. They serve a law q that stays within a divergence budget K of a *safe* model p_s trained without the protected work, k-near-access-freeness (Vyas et al., 2023), so a small K bounds how much likelier the served text is to be the work than the safe model's own output. The deployed instance, anchored decoding (He et al., 2026), charges the divergence per token from a bucket of rate k, certifies a T-token response at $K = kT$, and publishes k as a deployer would publish the ε of differential privacy.

We ask what such a budget excludes. A certificate about a work x says nothing once K reaches the safe model's surprisal of it, $S(x) = -\log p_s(x)$, and this threshold is the same at every Rényi order (Proposition 1). A budget charged per token grows with the output, so for a work served at its own length vacuity is decided by k against the surprisal rate $s(x)$ (Figure 1c), and a 50-token window of a 200-token response is exposed from $k = s(x)/4$. At the authors' book setting, $k = 3$ over 200 tokens, $K = 600$ nats and the certificate is vacuous for all 758 protected passages we test, whose median surprisal is 204.8 nats.

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.

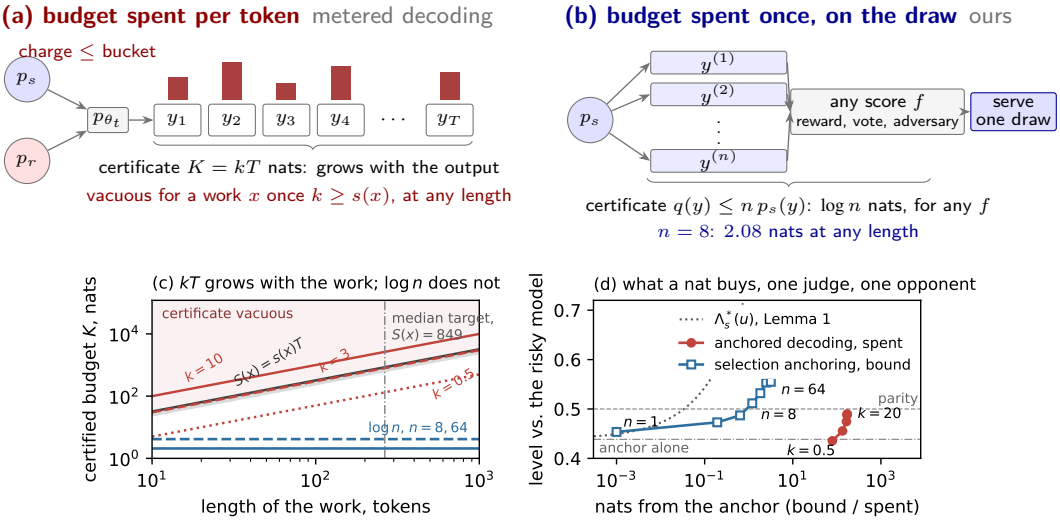

Figure 1: **Where the budget is spent decides what it certifies.** **(a)** A meter tilts each token from the safe model p_s toward the risky model p_r, charging a bucket. **(b)** Selection draws n outputs from p_s and serves one; only the choice is charged. **(c)** For a work served at its own length the meter's $K = kT$ (red) and the work's price $S(x) = s(x)T$ (grey, nine pairs) are both rays, so vacuity ($K \geq S(x)$, shaded) is decided by $k/s(x)$, and selection's $\log n$ (blue) is flat. **(d)** One judge pass: order-averaged level against the risky model's own draw (0.5 is parity), at the divergence the meter *spent* (filled, measured) or selection is *certified* not to exceed (open, a KL bound; Appendix D); arms at 0 nats drawn at 10^{-3}. Dotted: Lemma 1's frontier, a lower bound no decoder here attains.

A budget that does not grow. Draw n completions from the safe model, score them with anything, and serve one (Figure 1b). The served law satisfies $q(y) \leq n\, p_s(y)$ for every y (Proposition 2): a pathwise certificate of $\log n$ nats that holds for any scorer, an adversarial one included, whatever the length of the output, and adds up to $m \log n$ over m queries; at $n = 8$ it is 2.08 nats where the authors' meter publishes 600, and self-consistency (Wang et al., 2023) run over a safe model is an instance. Selection never consults the risky model, so it is no substitute for anchored decoding but a certified way to improve a licensed model. What separates the two is what the budget buys: a choice among the safe model's own draws, or the risky model's tokens, whose divergence a meter pays on every token it imitates, about 40 nats for a 50-token window of ordinary text (Section 3).

Contributions.

- **What a budget certifies**: nothing about an event of surprisal $S \leq K$, at every Rényi order (Proposition 1, an elementary identity); for a whole-sequence per-token budget, vacuity if it grows and sparsity if it is bounded (Proposition 3), and the imitation cost a meter pays on every token it imitates (Proposition 4), two short consequences of the chain rule; and for anchored decoding, extraction from 0.88 to 1.17 of $s(x)$ over nine pairs, where its certificate is already void.
- **A certificate that does not grow with the output**: pathwise $\log n$ for any selection rule and score (Proposition 2), $m \log n$ over m adaptive queries, small enough to audit by sampling the anchor, with what it does not buy priced: the query horizon, the per-work audit and the vetted anchor.
- **Head-to-heads at a matched certificate** under two judges (Table 2): with the risky model served as a text continuer at temperature 1.0, selection and selection in installments beat every meter we ran, and a meter that caps every 50-token window, the design our dichotomy leaves open, trails selection up to 40 nats a window. At larger budgets anchored decoding serves the risky model, and which of the two is judged better depends on how that model is served: as a chat assistant it wins, and so does that meter, at window budgets no rights-holder can audit. With an adversarial scorer selection's near-verbatim recall is 0.0000 at every $n \leq 64$ and at $n = 256$, a bound *relative* to a vetted anchor (Figure 6); it costs 21.8× a metered decode and loses at matched compute.

2 What a budget certifies

Fix a prompt. Write p_s for a *safe* model, trained without the protected work, and p_r for the *risky* model a deployer would rather serve. A defence serves a law q and certifies $D(q \,\|\, p_s) \le K$ for some divergence D. For a protected work x write $S(x) = -\log p_s(x)$ for the safe model's surprisal of it and $s(x)$ for that surprisal per token (notation: Table 4). Three protected objects recur, and they differ in length. A *passage* is the continuation a CopyBench item protects (Chen et al., 2024a), 64 tokens and 204.8 nats under the authors' anchor at the median of the 758 we test; a *window* is 50 tokens of protected text given the true text before it, median $S_w = 159.8$ nats; a *target* is what an extraction sweep asks for, all of a passage's text after a 20-token seed, 259.5 tokens and 849 nats at the median of the 100 passages attacked.

Proposition 1 (The vacuity threshold does not depend on the order)**.** *Let E be an event with $\Pr_{p_s}[E] = e^{-S}$, and for $\alpha \in [1, \infty]$ let $b_\alpha(K) = \sup\{\Pr_q[E] : D_\alpha(q \,\|\, p_s) \le K\}$, the most a budget of K nats at order α can say about E. Then $b_\alpha(K) = \max\{a : d_\alpha(a \,\|\, e^{-S}) \le K\}$, with d_α the Rényi divergence between Bernoulli laws, and $b_\alpha(K) = 1$ if and only if $K \ge S$, at every order.*

Data processing gives $d_\alpha(\Pr_q[E] \,\|\, \Pr_{p_s}[E]) \le D_\alpha(q \,\|\, p_s)$, a law that reweights p_s uniformly on E and on its complement attains it, and $d_\alpha(1 \,\|\, e^{-S}) = S$ at every order (Appendix A). Below it the orders differ: the pathwise bound is e^{K-S} and the KL bound falls only as about K/S, so a KL budget of half a work's surprisal still permits the work half the time. For a work served at its own length, $K = kT$ and $S(x) = s(x)T$ grow together and the KL bound, about $k/s(x)$, does not improve with length; w tokens inside a T_{max}-token response are unprotected from $k = s(x)\, w/T_{\text{max}}$, a quarter of $s(x)$ for 50 tokens at $T_{\text{max}} = 200$ (Table 5). The identity is elementary (van Erven & Harremoës, 2014) and Vyas et al. (2023) report their budget against the safe model's entropy; what we add is reading deployed budgets against the threshold, hence the suggestion to publish $k/s(x)$ rather than k.

A budget that does not grow. A *selection rule* draws $y_1, \ldots, y_n \sim p_s$ i.i.d., scores them with any function f, and serves one. Every served token was sampled from p_s; the served law departs from p_s only through which draw survives, and that choice is all the certificate charges.

Proposition 2 (The budget of a selection rule)**.** *Let $y_1, \ldots, y_n$ be i.i.d. p_s and let q be the law of* any *rule that serves one of them. Then $q(y) \le n\, p_s(y)$ for every y, hence $D_\infty(q \,\|\, p_s) \le \log n$ and so $D_\alpha(q \,\|\, p_s) \le \log n$ for every $\alpha \in [1, \infty]$, and $\Pr_q[E] \le n \Pr_{p_s}[E]$ for every event E. If the rule is the argmax of a score with no ties the KL is smaller still, $D_{\text{KL}}(q \,\|\, p_s) \le \log n - (n-1)/n$.*

Proof. If y is served then y was drawn, so $q(y) \le \Pr[\exists i : y_i = y] \le \sum_i \Pr[y_i = y] = n\, p_s(y)$. A logarithm and a maximum over y give the $\alpha = \infty$ statement, which dominates every lower order; summing over $y \in E$ gives the event form. Only the union bound is used, so the rule may consult the scores, the strings, external randomness or nothing. The KL form is due to Beirami et al. (2025). □

The inequality is elementary and its KL form known; what it *certifies* has not been stated. It uses only that each draw's marginal is p_s and survives deterministic post-processing and stopping early (Appendix A); our draws are untruncated samples at the stated temperature, so the reference is the anchor exactly as sampled; any fixed sampling law of the licensed model may serve, tempered or not, if vetted at its own temperature (Appendix E). The bound covers rules with no score in them: **self-consistency (Wang et al., 2023) is an instance**, serving one of the draws, one with the modal answer. The bound is tight: if f has a unique maximiser $y^\star$ with $p_s(y^\star) = p$, then $q(y^\star) = 1-(1-p)^n \to np$. Over a web-trained model, as self-consistency is usually run, the same bound certifies nothing about copyright: the guarantee is relative to p_s and is only as good as the anchor (Section 4.1).

What one certificate covers, and for how many queries. Table 1 prices the exact window at its own position, whose probability under the anchor has a closed form. Reproducing any near-verbatim variant of it has none, and a rights-holder can bound it only by sampling the anchor (as privacy is audited, Jagielski et al., 2020): N clean draws give $3/N$ at 95%, so selection certifies the event at $3n/N$ per response: per work, 19,200 draws for 1% at $n = 64$, about 0.12 A100-hours, and 11,901 for 10^5 works, where our 25,600 clean draws certify only the *average* passage, 0.75% at $n = 64$ (Appendix E); a pathwise certificate of K nats needs more than $3e^K$ draws. Max-divergence adds

Table 1: **What one response's certificate says about a protected 50-token window** ($S_w = 159.8$ nats, median over 758 passages). Meter: KL charge, $K = kT_{\max}$ at $T_{\max} = 200$, measured mean spend in brackets, bounded by Proposition 1 (1: nothing excluded). Selection: pathwise $K = \log n$, bound ne^{-S_w}. *queries*: adaptive queries whose composed certificate stays below S_w. *draws*: clean anchor draws that certify near-verbatim reproduction of one work below 1% per response ($3n/N$); no feasible sample does so for a KL certificate of 100 nats or more.

mechanism	charge	K (spent), nats	K/S_w	$\Pr_q[\text{window}] \leq$	queries	draws
meter, $k = 0.5$	KL	100 (78.5)	0.63	0.630	1	—
meter, $k = 3$ (authors')	KL	600 (165.0)	3.75	1	0	—
meter, $k = 10$	KL	2000 (171.3)	12.5	1	0	—
selection, $n = 8$	pathwise	2.08	0.013	$10^{-68.5}$	76	2,400
selection, $n = 64$	pathwise	4.16	0.026	$10^{-67.6}$	38	19,200

exactly under adaptive composition, so m responses at n draws each are certified at $m \log n$ however the queries were chosen (Appendix A); per-query access-freeness does not compose in general (Cohen, 2025). For one response to reproduce the window the union bound gives $m\, n\, e^{-S_w}$, below 10^{-60} after 10^4 queries at $n = 64$, but a transcript can carry the window piecewise, and there the certificate survives only 38 queries at $n = 64$ (Table 1). A scorer that colludes with the user can spend it: a keyed hash (steganography, as in Schroeder de Witt et al., 2023) realises 3.97 of the 6 bits a response at $n = 64$ can carry (Appendix E). A cruder collusion, extending a memorised passage one token per query with the draw the memoriser ranks highest, reconstructs none of 100 windows after 50 queries at $n = 64$, though the certificate (207.9 nats) is vacuous for 88: the anchor's draws hold the true next token on 12.3% of queries (Appendix E). That is looseness, not safety: a deployer who cannot vet the scorer must keep the queries of every set of users who can pool their transcripts, in practice the whole deployment, below $S/\log n$ for the shortest event it protects, 38 at $n = 64$ for a window, a cap few public services would accept; a vetted scorer, one that neither holds the work nor takes the user's instructions, removes the need, and the certificate cannot check that. Ours is not: `Qwen2.5-7B-Instruct` is web-trained and reads the prompt; a vetted one gains little (Appendix B).

3 What a per-token rate can certify

Could a metered decoder be made to certify as little as selection does? For a budget on the whole sequence, one line of the chain rule fixes the shape its spend must take.

Proposition 3 (A bounded budget is sparse)**.** *Let q be any causal policy, $q(y) = \prod_{t \leq T} q_t(y_t \mid y_{<t})$, with sequence budget $D_{\mathrm{KL}}(q \,\|\, p_s) \leq K$, and write $N_\varepsilon = \#\{t \leq T : D_{\mathrm{KL}}(q_t(\cdot \mid y_{<t}) \,\|\, p_{s,t}) > \varepsilon\}$. Then $\mathbb{E}_q[N_\varepsilon] \leq K/\varepsilon$ for every $\varepsilon > 0$.*

The proof is the exact chain rule and Markov's inequality at each step (Appendix A). A per-token allowance on the whole sequence therefore has two options. If it grows with the work, $K = kT$, it is vacuous from $k = s(x)$ on at every order (Proposition 1). If it stays bounded, the policy departs from the anchor by more than ε at $O(K/\varepsilon)$ steps and is the anchor everywhere else. The bound constrains the *shape* of a spend, not the class: selection's served law satisfies it (at 3.175 nats, fewer than four of its 204 steps carry a whole nat) because its budget sits on a choice the decode loop does not contain.

Lemma 1 (The price of utility)**.** *Let q be* any *law with $D_{\mathrm{KL}}(q \,\|\, p_s) \leq K$. For every bounded utility U,*

$$K \;\geq\; \Lambda_s^*\big(\mathbb{E}_q[U]\big), \qquad \Lambda_s^*(u) = \sup_{\lambda} \big\{\lambda u - \log \mathbb{E}_{p_s}[e^{\lambda U}]\big\}, \tag{1}$$

the Cramér rate function of U under the safe model.

This is the Donsker–Varadhan (Gibbs) variational principle (Donsker & Varadhan, 1975) applied to the whole sequence, and we claim no novelty for it (Appendix A); nothing about the decoder enters. Reproducing a work costs $S(x)$, which grows with its length: 849 nats for the median protected target and 159.8 for a 50-token window. Buying mean utility u costs $\Lambda_s^*(u)$, which is $O(1)$ at any length for a utility the safe model itself sometimes attains. There is room for a mechanism that is both safe and useful; a meter that tilts toward the risky model does not occupy it, because where its bucket is slack it is charged the full cost of imitation whatever utility that buys.

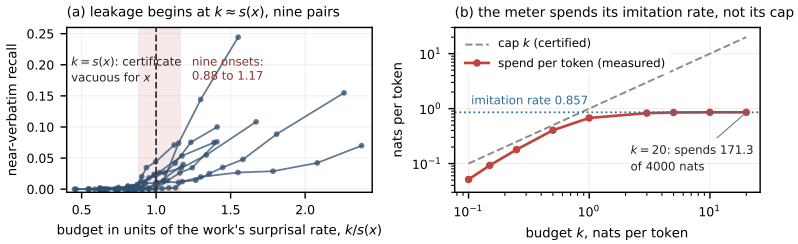

Figure 2: **Where the deployed meter leaks, and what it spends. (a)** Near-verbatim recall of held-out protected passages against the budget in units of the anchor's surprisal rate, nine (anchor, memoriser) pairs with nine distinct anchors and six tokenizers. Extraction begins between 0.88 and 1.17 of $s(x)$ (shaded: their range, not an interval), near Proposition 1's threshold for the whole target. Each sweep decodes T tokens, the longest target in it (296 at this anchor), so the median target's own certificate is void from 0.84 to 0.89 of $s(x)$ and a 50-token window's from $50/T$, 0.08 to 0.17 of it: it never excluded what was extracted. **(b)** Anchored decoding at its authors' anchor on ordinary prompts, $16{,}500$ trajectories: the cap it certifies (dashed) against what it spends (red). Past the imitation rate the bucket never binds and the spend stays at 171.3 nats while the cap grows to 4000.

Proposition 4 (Where its bucket is slack, a meter pays the imitation cost)**.** *Let a bucket-metered decoder serve at each step the largest tilt $p_{\theta_t} \propto p_s^{1-\theta_t} p_r^{\theta_t}$ whose KL charge fits its bucket, and call a step slack when $\theta_t = 1$, and let I_T sum $D_{\mathrm{KL}}(p_{r,t} \,\|\, p_{s,t})$ over the slack steps. Then $D_{\mathrm{KL}}(q \,\|\, p_s) \geq \mathbb{E}_q[I_T]$, whatever the utility achieved. If $\mathbb{E}_q[I_T] \geq \gamma T$, and a bounded utility U that the decoder does not lower attains its maximum under p_s with probability at least $\pi > 0$, then $D_{\mathrm{KL}}(q \,\|\, p_s) \geq \Lambda_s^*(\mathbb{E}_q[U]) \cdot \gamma T / \log(1/\pi)$.*

The deployed meter shows both options (Figure 2). By the time extraction begins the certificate is void: on nine pairs the onset lies between 0.88 and 1.17 of $s(x)$, near the whole-target threshold, and one pair spans 0.88 to 1.35 when only its memoriser's fine-tuning length changes, so the level is the claim and the ordering is not; two mechanistic predictions of the onset failed (Appendix G). Where the budget certifies something the meter binds and is judged indistinguishable from its own anchor (Table 2); the share of steps its bucket constrains falls from 0.954 at $k = 0.1$ to 0.033 at $k = 3$ and 0.0000 at $k = 20$, and past the imitation rate $\gamma = 0.857$ nats per token it serves the risky model unchanged and spends 171.3 nats, 4.3% of the 4000 it certifies at $k = 20$. The utility it reaches costs 0.023 nats by Lemma 1 (Appendix H). No per-trajectory violation occurs in over $100{,}000$ budgeted trajectories; the gap is between what the budget bounds and how k is read.

The repairs a meter has left each land on one of the two options or fall short of selection (Appendix H, Table 40): a sharper charge ($\alpha = 8$) cuts oracle recall from 0.0965 to 0.0012 only by binding on 99.6% of steps against 0.4%, a lower cap ($k = 0.5$) is its anchor, a better schedule buys 3 to 13% more fidelity at $k \leq 1$ and under 1.4% at $k = 3$, and a budget granted up front gains $+0.0545$ $[+0.0315, +0.0775]$ at 64 nats against selection's $+0.1045$. They share a currency, likelihood under p_r: to choose among draws it predicts judged quality no better than the answer's length, moving judged utility by -0.006 against a band of $+0.077$ where a circular oracle reaches $+0.488$ (Appendix B), and to tilt generation it buys the risky model's quality only by becoming the risky model.

What the dichotomy does not cover, built and measured. Proposition 3 constrains a budget on the whole sequence. A meter can instead cap each step's max-divergence by W minus the largest realised log-ratio of a recent span, so every span of at most w tokens is certified at W given the text before it. Ordinary text's realised log-ratio per 50-token window has median 40.3 nats (Appendix G). We built it ($w = 50$; Section 4, Appendix G): its certificate covers every event in a span, near-verbatim variants included, up to e^W: the median exact window at $e^{W-159.8}$, but a near-verbatim variant only for an audit that samples past $3e^W$ draws; a memoriser served through it reproduces near-verbatim spans from $W = 80$. Selection spent in installments, n draws per block of L tokens, is certified by the same composition at $(\lceil w/L \rceil + 1) \log n$ for any w-token span given what precedes it, and it buys more: at 64 draws a block, 10-token installments beat one choice by $+0.0415$ $[+0.0185, +0.0645]$, reproduced on a fresh draw, at 25 nats a window, while held to

one whole-output $\log 64$ they lose (Appendix G). At $1{,}000$ tokens one choice among 64 still gains $+0.063$ $[+0.043, +0.083]$, and installments of 8 draws a block lose to it (Appendix G). Both sample a bounded tilt of the anchor, $p_s h$ with $h \le e^{\lambda}$ at every length (Appendix G); whether a causal policy that consults p_r can realise one without paying the imitation cost is what Proposition 3 leaves open.

4 Experiments

Setup. Pairs. The meter runs at the anchor its authors evaluate, `TinyComma-1.8B` (He et al., 2026; Kandpal et al., 2025), trained only on openly licensed text and sharing the risky model's tokenizer. The risky model is `Llama-3.1-8B-Instruct` (Grattafiori et al., 2024), served as a text continuer as in an earlier audit's released logs (Vijayavallabh, 2026) or through its chat template, or the authors' `Llama-3.1-70B` base. Their byte-level decoder, AnchoredByte (Hayase et al., 2026), runs unmodified at their recommended pair, Comma-7B with the 70B base. Decoding is at temperature 1.0 without repetition penalty, and again at the authors' temperature 0.7 and penalty 1.1 (*their* 0.7 *and* 1.1), with $T_{\max} = 200$. **Workload.** 500 ordinary prompts, none from the protected corpus: 200 questions we wrote, 150 WritingPrompts premises (Fan et al., 2018) and 150 FactScore biography requests (Min et al., 2023); the first two carry the released logs' `Complete the prefix:` header (Table 6). **Utility.** Each arm is judged against one fixed opponent, the unconstrained risky model's own draw, in both orders, scoring 1, $\frac{1}{2}$ or 0: an arm's *level* is its mean (0.5 is parity), its *gain* its level minus its own zero-budget control's (the anchor alone, or selection's single draw), and a *difference* two gains paired over prompts, free of position by construction. Judge B, `Phi-3.5-mini-instruct` (Abdin et al., 2024), is pre-specified, but its two verdicts on an item agree only 29 to 37% of the time, so every registered difference is also read by judge G, `gemma-2-27b-it` (Gemma Team et al., 2024), which agrees with itself on 78 to 85%; judge A, `Qwen2.5-7B-Instruct` (Qwen Team, 2025), supplies selection's pointwise reward, $\log p(\text{Yes}) - \log p(\text{No})$ on one fixed template, and never scores an arm it selected. **Leakage.** Near-verbatim recall on 100 protected passages, the fraction of the reference's words in in-order near-verbatim spans of at least 20 words (Ahmed et al., 2026). Intervals are 95% paired bootstraps; every band was fixed in advance (Appendix B).

At a matched certificate. Table 2 puts every mechanism into one judge pass at the headline configuration, the 8B continuing text at temperature 1.0. **Selection beats every meter at every matched certificate.** Best-of-64 gains $+0.1015$ for a KL bound of 3.175 nats, attained only for a tie-free score and unrepeated strings (Appendix D), while the meter at $k = 0.5$, the one published budget that says anything about a window, blends on half its steps and is judged indistinguishable from its own anchor, -0.0030 $[-0.0245, +0.0185]$, a null with 80% power against $+0.0305$ and, by a two one-sided test, equivalent to zero within half of selection's gain; the same judge gives $+0.065$ to a second draw of the risky model over the anchor. At a matched $\log 64 = 4.16$ nats, selection leads the budget granted up front by $+0.0785$ $[+0.0445, +0.1115]$ and a meter capping every 50-token span by $+0.090$ $[+0.058, +0.122]$; installments ($L = 10$) lead the per-token meter at their own whole-output 83.18 nats by $+0.142$ $[+0.1085, +0.1755]$, the windowed meter at their own 24.95 nats a span by $+0.1205$ $[+0.0855, +0.155]$, and the meter at $k = 0.5$ by $+0.161$. Judge G gives all nine registered differences the same sign and resolves all but one. Empty answers qualify this: both judges score one at or above parity; counted as losses the differences hold under judge G at every budget but $W = 125$, and under judge B against the per-token meters and the windowed meter at $\log 64$ only (Appendix B). The windowed meter climbs with its budget and draws level only at $W = 125$ ($+0.022$ $[-0.0145, +0.0585]$; we had predicted selection would still lead), where a memoriser served through it already reproduces near-verbatim text: 0.106, against 0.495 alone and 0.000 at every $W \le 40$.

Where the risky model is far stronger than the anchor, the windowed meter wins, at a certificate too loose to audit. Served through its chat template the $k = 10$ meter gains $+0.263$ over the anchor; a windowed meter at $W = 125$ beats selection by 0.160 $[0.122, 0.1985]$, and at $W = 40$ it wins under judge B, -0.0525 $[-0.0895, -0.0150]$, ties under judge G, -0.0295 $[-0.0725, +0.0145]$, and leaks nothing through the same template. It bounds the exact window, the median one at $e^{40-159.8}$, and a near-verbatim variant too, but only through the factor e^{40}, which a rights-holder could audit by sampling only with more than $3e^{40}$ clean draws, where selection's $\log 64$ needs $19{,}200$ (Appendix G).

Against the risky model. From $k = 1$ on the meter's certificate is vacuous for a window and the comparison is with the risky model itself (Table 10), served unchanged at 99.99% of steps at

Table 2: **Every mechanism at a matched certificate**, one judge pass, 500 prompts. Certified max-divergence nats (KL at $k = 0.5$) for any 50-token span given the text before it (—: none beyond the output's) and for the 200-token output, the first conditional on the text before the span, the second joint: rows match in nats, not in type. Gain over the arm's own zero-budget control, judge B with 95% interval, and judge G. *Leak*: near-verbatim recall of 100 memorised passages, the memoriser as risky model (0.495 alone); for selection, the best of its 64 draws. All twelve arms: Table 31.

mechanism	span	output	gain, judge B	judge G	leak
anchored decoding, $k = 0.5$	—	100	-0.0030 $[-0.0245, +0.0185]$	$+0.015$	—
budget up front	—	4.16	$+0.0230$ $[+0.0010, +0.0445]$	-0.0265	—
per-token pathwise meter	—	83.18	$+0.0160$ $[-0.0050, +0.0370]$	$+0.0125$	—
windowed meter, W per span	4.16	16.64	$+0.0115$ $[-0.0110, +0.0340]$	-0.047	0.000
	24.95	99.81	$+0.0375$ $[+0.0150, +0.0600]$	-0.0045	0.000
	40	160	$+0.0375$ $[+0.0140, +0.0610]$	$+0.0475$	0.000
	125	500	$+0.0795$ $[+0.0530, +0.1055]$	$+0.130$	0.106
best of 64	—	4.16	$+0.1015$ $[+0.0775, +0.1260]$	$+0.176$	0.000
installments, $L = 10$	24.95	83.18	$+0.1580$ $[+0.1330, +0.1835]$	$+0.305$	—

$k = 10$ for a measured 171.3 nats. At the authors' 0.7 and 1.1 even $k = 0.5$ is not the anchor: the 8B meter there gains $+0.0825$ $[+0.0605, +0.1045]$ and the gains tie, meter minus selection -0.0245 $[-0.057, +0.0095]$, while at their 70B pair the meter is its anchor again ($+0.018$ $[-0.0035, +0.040]$) and selection leads by 0.089 $[0.0575, 0.122]$. Continuing text at temperature 1.0, the risky model loses (measured before the other two were registered): the meter gains $+0.0510$ $[+0.025, +0.076]$ at $k = 10$, and the gains differ by $+0.0505$ $[+0.0155, +0.0860]$, at one fifty-fourth of the divergence ($+0.121$ $[+0.024, +0.222]$ on the 62 prompts where both arms' verdicts are order-consistent), and by more at the authors' 70B pair, whose base model at this temperature, fluent but off task, is judged no better than the anchor, $+0.019$ $[-0.004, +0.042]$; at the authors' 0.7 and 1.1 the 8B wins instead, -0.081 $[-0.113, -0.049]$, while their 70B, now $+0.0585$ over the anchor, still loses, by $+0.0485$ $[+0.0145, +0.082]$ at $k = 20$. Served through its chat template the 8B beats its own templateless text by 0.184, and the $k = 10$ meter, that model at 99.84% of steps, wins by 0.138; the meter's interval first lies above selection's at $k = 3$, whose $K/S_w = 3.75$ excludes nothing about a window, after a tie at $k = 2$, and a memoriser served the same way reaches 0.01 recall only at $k = 3.9$ $[3.0, 4.4]$, so the win comes before measured extraction, though within its interval (Appendix C); that model gains $+0.272$ over the anchor where selection gains $+0.1065$, so selection closes less than half the gap (Appendix J). Handed the same reward on TriviaQA, the meter gains nothing (Table 13).

At $k = 10$ the difference survives fresh draws and the judges that read reliably, and not a stronger opponent, instruction following, empty answers scored as losses or, under judge B, a scorer from another family (Figure 3). Re-drawing selection, then every sampled arm, moves it by at most 0.0055. Five of six judges exclude zero, among them both whose two verdicts on an item agree on over three quarters of items, `Qwen2.5-72B` ($+0.055$, the scorer's family) and `gemma-2-27b-it` ($+0.0545$); the sixth, `Mixtral-8x7B`, straddles it at -0.0015 and agrees with itself on 35 to 42% (Appendix B); on the recorded text it bounded the difference below 0.03, $+0.0226$ $[-0.0018, +0.0476]$ over 850 prompts, half-width 0.0247. The largest reading, $+0.1210$, is `Qwen2.5-14B`'s, the scorer's family; without the two Qwen judges three of four exclude zero. Scored instead by `gemma-2-27b-it`, which serves shorter drafts (median 34 words against 75) and more empty ones (71 against 41), selection gains $+0.066$ $[+0.042, +0.0905]$ and the difference is $+0.015$ $[-0.0185, +0.0485]$ (pre-registered): under B the headline depends on the scorer, though four of six judges resolve it with either scorer and only B prefers the Qwen reward's drafts (Appendix B). The reward prefers empty drafts, which the judge scores near parity: counting empty answers as losses leaves $+0.0155$ $[-0.0245, +0.0555]$ (selection serves 41 empty texts and the meter none), and a rule that never serves an empty draw, still certified at $\log n$, keeps $+0.056$ $[+0.0195, +0.093]$ and every reading of Table 10, though judge G, which scores empty answers highly, reads it at $+0.0335$ $[-0.020, +0.0875]$ (Appendix B). From `Qwen2.5-1.5B-Instruct` upward every interval covers zero, and the difference belongs to prefix completion: it holds on our prompts, public-domain books and BookMIA's unseen half, is unresolved on CoTaEval's news questions and MT-Bench, and reverses on AlpacaEval, where the meter wins by 0.0329 (Appendix C).

Selection against its own anchor. Across the openly licensed anchors (Kandpal et al., 2025; Langlais et al., 2026; Bommarito et al., 2025), of which Comma-7B is the largest to pass the vetting check of Section 4.1, selection's gain at $n = 8$ excludes zero at four of six anchors in two families, largest at Comma-7B, $+0.126$ $[+0.087, +0.164]$ (Figure 4), and does not significantly track the anchor's own level over six anchors ($\rho = +0.543$ on the two-judge mean). **With no judge**, majority vote over Comma-7B lifts GSM8K (Cobbe et al., 2021) exact match from 0.320 to 0.546 at $n = 32$ (3.47 nats), $3.4\times$ the pointwise reward's best lift, and lifts TriviaQA and MMLU (Hendrycks et al., 2021), where on TriviaQA the reward falls with n: the scorer binds first (Figure 5, Table 17).

4.1 Leakage

A scorer can only amplify leakage the anchor already has (Proposition 2). Against an attacker who scores the n draws by the memorising model's own likelihood, and against a paraphrase event, near-verbatim recall is 0.0000 at every anchor, at every n to 64 and at $n = 256$ (Figure 6a), where the memoriser alone reads 0.3925 mean and 0.8154 maximum and the 70B, unaided, 0.2475, reproducing two passages in full; the zero bounds the per-passage rate below 3.0% at 95%. On the meters' own sweep protocol the best of 64 anchor draws also reads 0.0000, where a memoriser served through the windowed meter reaches 0.01 at 80 nats a window, 125 through the chat template (Appendix G). The zero verifies the anchor, not the mechanism: all six anchors read 0 on a vetting screen at every prefix length from 20 to 200 tokens, where the open-data `OLMo-2-7B` and `OLMo-2-13B` (Team OLMo et al., 2024) leak on up to 2 and 6 of 50 (Appendix E). With anchors deliberately fine-tuned on the passages, the realised amplification at $n = 64$ is 1.0 to 7.0 on one draw and 1.0 to 3.0 on a second (Figure 6b): selection multiplies what the anchor leaks and is no floor under it.

4.2 Cost

Selection pays in draws, and one number is the price: on a server that runs only the anchor the 64 draws cost $21.8\times$ a metered decode (Table 24, Appendix F). Counted in forward passes $n = 64$ costs $61.3\times$ the meter, most of it the reward model's prefill, but on the clock the reward pass is 9.3% and the draws are the rest, so the lever is n and not the scorer. Batched into one decode loop a request takes $0.22\times$ the meter's time at the authors' 70B pair and $0.51\times$ at the 8B one, because it never runs the risky model ($0.061\times$ against their byte-level decoder); speculative decoding would lift these to at best $0.37\times$ and $0.88\times$ (Appendix F). **Held to the meter's own compute, selection loses**: the matched cell is $n = 2$ at $0.68\times$, gaining -0.0330 $[-0.0625, -0.0025]$ against the meter, and on TriviaQA it reaches 0.112 where the meter, vacuous at $k = 20$, reaches 0.618. A $3.9\times$ larger safe model served alone is no substitute: it gains $+0.015$ $[-0.0065, +0.0365]$ over the small one (Appendix D). Majority vote has no scorer, costs $5.75\times$ the meter's forward passes at $n = 32$, and beats the pointwise reward up to a 7.6B scorer but not at 14B (Appendix D).

5 Related work

Certified copyright defences. Near-access-freeness (Vyas et al., 2023) bounds the served law against a safe model. Its authors' CP-k already pays once per output, rejection-sampling complete outputs against models trained on disjoint shards with a bound that grows as acceptance falls; CP-Fuse (Abad et al., 2025) fuses such shards per token and anchored decoding (He et al., 2026) charges per token from a bucket. Selection needs one safe model and no likelihood, and its $\log n$ depends on neither the rule nor an acceptance rate. No model pair here meets CP-k's premise of disjoint shards. Run with the anchor as its one safe model, its rule serves the risky model on no prompt at $\log 64$ and on 26.8% at 83 nats, and best-of-64 and installments beat it at every matched certificate (Appendix G). Segal et al. (2026) report that CP-style guarantees can be numerically uninformative and demonstrate extraction against NAF-protected models (see also Cohen & Zhang, 2026). We are downstream of that observation: Proposition 1 gives the exact point at which a budget stops saying anything and shows that no Rényi order moves it, which licenses changing the primitive rather than the constant. An earlier audit (Vijayavallabh, 2026) measured anchored decoding's vacuity on the released logs of one 8B pair; Figure 2 comes from new runs, and new here are reading deployed budgets against the elementary threshold, the window corollary, Propositions 2–4, the windowed meter and every head-to-head.

Defences without a divergence certificate. Blocklists (Ippolito et al., 2023), datastores (Min et al., 2024; Golatkar et al., 2024) and filtering (Liu et al., 2024), compared by CoTaEval (Wei et al., 2024), cover only works they are handed; there (Appendix J) a blocklist costs no utility and TokenSwap (Prashant et al., 2025) suppresses as fully as selection, more cheaply; neither bounds the served law.

Best-of-n and private selection. We claim no novelty for best-of-n sampling or its divergence bound: selecting among draws is standard inference-time alignment (Gao et al., 2023; Mudgal et al., 2024; Wang et al., 2023), the bound $D_{\mathrm{KL}} \leq \log n - (n-1)/n$ is Beirami et al. (2025)'s, its coverage and optimality are characterised by Huang et al. (2025), it is near-optimal in win rate against KL (Gui et al., 2024; Yang et al., 2024), and blockwise best-of-K (Mudgal et al., 2024) is selection in installments. Charging the choice among repeated runs is private selection's accounting (Liu & Talwar, 2019; Papernot & Steinke, 2022), and private next-token prediction mixes a public model in per token (Ginart et al., 2022; Flemings et al., 2024; Majmudar et al., 2022) or pays only where private and public disagree (Amin et al., 2024), Proposition 3's sparse option in privacy's currency. That literature reads the divergence as a *cost*, spent before reward overoptimisation or by an attacker (Hughes et al., 2024); we read the inequality as a copyright certificate. CP-k aside, we know of no prior statement that a rule choosing among a safe model's draws satisfies near-access-freeness whatever the rule, nor a comparison of the two mechanisms on one divergence axis. The closest neighbour, Kalai et al. (2025), abstains unless several models agree (Appendix J).

6 Limitations and conclusion

The two mechanisms are not substitutes, and the scorer binds first. A served string is one the safe model would have drawn, so an anchor that cannot do the task gains nothing: on TriviaQA (Joshi et al., 2017) with no judge the meter reaches 0.618 against selection's 0.190, but only at $k = 20$, where it *is* the risky model, certified at 480 nats for a 24-token answer. Below that the scorer binds: it gains $3.4\times$ less than majority vote on GSM8K at $n = 64$ and *loses* F1 on CoTaEval, whose infringement event not even the risky model reaches (Appendix D). A capability gap is the regime a deployer most wants covered, and *neither* mechanism covers it with a certificate a rights-holder can audit. Nor does selection improve facts: its FActScore precision is its anchor's, 0.042, below every metered arm, though the best of its 64 draws reaches 0.5054 (Appendix C). **The judged comparison measures the serving configuration as much as the mechanism**: against the risky model it reverses under a chat template, on AlpacaEval (Li et al., 2023) and at the authors' temperature 0.7 and penalty 1.1, and vanishes against a stronger opponent, when empty answers count as losses or, under the pre-specified judge, with a scorer from another family (Table 10, Figure 3); under Holm's correction over 21 rows the headline row does not survive, every judge is a language model, the pre-specified one order-consistent on a third of items, and no human rated anything. Tempering buys only part of the gain: best-of-64 leads it at $T = 0.7$ and 0.5 by $+0.0685$ and $+0.069$ (B), $+0.063$ and $+0.060$ (G; Appendix B). **And it is paid for in compute**: $61.3\times$ the meter's forward passes and $21.8\times$ its wall-clock in draws alone, though batched it takes $0.22\times$ its time per request at the 70B pair, and held to the meter's own compute it *loses*. The *gain* is the scorer's capability: a 0.5B scorer never reaches the meter, a 1.5B one buys 87% of the 7.6B gain on the judged workload and nothing on GSM8K, and where that bar sits is open; the *price* is the drawing, not the scorer's one batched pass (Appendix D). On TriviaQA answers $\log 64$ reaches $S(x)$ on 36.2% of questions, against 83.6% for the meter's smallest budget, so n must be set against the shortest work protected, and a vote gains only where $\log n \geq S$ (Appendix D); on protected quotations selection's worst case leaks 0.38% and the meter 69.8% at $k = 10$ (Appendix E); we tested no lyrics, whole poems or code, and extraction is sixteen English novels plus four in French and German. **Why certify at all**: through the chat template the meter first beats selection at $k = 3$, before a memoriser served the same way leaks, and a deployer may rationally prefer that measured protection; only a certificate covers works nobody tested and paraphrases nobody scored. How close a policy of the shape Proposition 3 permits comes to Lemma 1's frontier is open.

Conclusion. Spent choosing among a safe model's draws, a budget is $\log n$ nats at any length, or a few times that per window in installments, and beats every meter we ran at a matched certificate; a windowed meter wins where the risky model is far stronger, at certificates too large to audit.

Ethics Statement

Auditing a copyright defence means showing where it fails, so this paper fine-tunes memorising models and attacks a protected decoder. We report aggregate recall only, never reconstructed text, on passages already published as extraction benchmarks (CopyBench, Chen et al., 2024a; BookMIA, Shi et al., 2024); the anonymised repository ships the fine-tuning recipes and every analysis script but not the merged memorising weights, which reproduce copyrighted text on demand. The constructive result carries its own risk, misplaced confidence: $\log n$ bounds the probability that the served text is a protected work *relative to the safe model's own*, so it is only as good as that model. Over twelve deliberately contaminated anchors an adversarial scorer raised the anchor's own leakage by 1.0 to $7.0\times$ on one draw and 1.0 to $3.0\times$ on a second, against the 64 the certificate permits (Appendix E). The anchor's rate is the one quantity a deployer can measure with no attack and no access to the mechanism: draw from it on the protected passages' own prefixes and score recall. **The prefix length is part of the test**: `Llama-3.1-70B`, which memorised the work in pre-training, reads 0.0000 at fourteen genuine prefix tokens and leaks on 31 of 50 passages at 150, and the check is not monotone in the prefix, so it must run at every prefix length the deployment accepts, and at the temperature it samples at (Appendix E). Passing is evidence, not proof, and like a blocklist the check speaks only about the works it is handed. Our statements concern mechanisms; whether near-access-freeness tracks infringement is contested (Elkin-Koren et al., 2024), and we make no claim about legal thresholds for copyright infringement. The recommendation we would rather see quoted than the theorems: publish $k/s(x)$ rather than k, prefer a budget whose size does not grow with what it protects, vet the anchor before quoting $\log n$, because n multiplies whatever the anchor already has, and unless the scorer is vetted too, cap the queries of every user who could pool transcripts, in practice the whole deployment, below $S/\log n$ for the shortest protected event: a scorer that colludes with the user can spend the transcript's $m \log n$ nats one query at a time (Section 2).

Reproducibility Statement

Every number in this paper is written by an analysis script to a CSV before it is quoted, and the anonymised repository contains the code, the CSVs and the command that produced each. Proofs are in Appendix A. Every interval we score against was fixed before the run that produced it, together with its grid, admission criterion and excluded outcomes; budget accounting is checked per trajectory, and every run reports the $k = -1$ (risky only) and $k = 0$ (safe only) baselines on the same prompts and seeds. Two properties a reproducer should expect. Selection serves the argmax of a reward computed in bfloat16, and that argmax is not bitwise reproducible: re-scoring the same $32{,}000$ candidates on another GPU architecture, at another batch size or in float32 serves a different completion on 5.6 to 8.6% of prompts at $n = 64$; what is served is worth the same, the alternative giving up 0.011 to 0.03 nats against a gap of 5.1, and Proposition 2 holds for any tie rule. This was measured rather than assumed and carried no pre-registered interval. And the harness recorded each generation with the tail of its prompt, an artefact of slicing left-padded batches; nothing served was affected, every recorded generation is recovered exactly, and Table 10 and Figures 1, 3 and 4 are judged on the recovered text (Appendix B).

LLM Usage

LLMs were used for editorial purposes in this manuscript, and all outputs were inspected by the authors to ensure accuracy and originality. Instruction-tuned models also appear as *measurement instruments*, judging output quality, and as a *component of the mechanism*, supplying the score selection ranks by; for that reason no arm is scored by the model that selected it, and Proposition 2 is stated for an arbitrary score. Total compute for this work is at most 366 GPU-hours on A100 hardware, most of it the budget sweeps and the judged arms; the memorisation fine-tunes account for at most 73 of those hours.

References

Javier Abad, Konstantin Donhauser, Francesco Pinto, and Fanny Yang. Copyright-Protected Language Generation via Adaptive Model Fusion. In *International Conference on Learning Repre-*

sentations (ICLR), 2025. arXiv:2412.06619.

Marah Abdin, Jyoti Aneja, Hany Awadalla, Ahmed Awadallah, Ammar Ahmad Awan, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Jianmin Bao, Harkirat Behl, et al. Phi-3 Technical Report: A Highly Capable Language Model Locally on Your Phone. arXiv:2404.14219, 2024.

Ahmed Ahmed, A. Feder Cooper, Sanmi Koyejo, and Percy Liang. Extracting books from production language models. arXiv:2601.02671, 2026.

Maryam Alshehyari, Dushyant Singh Chauhan, Samuele Poppi, Martin Takac, Salem Lahlou, and Nils Lukas. CopyShield: A Cross-Level Benchmark of Copyright Defenses in LLMs. arXiv:2609.01161, 2026.

Kareem Amin, Alex Bie, Weiwei Kong, Alexey Kurakin, Natalia Ponomareva, Umar Syed, Andreas Terzis, and Sergei Vassilvitskii. Private prediction for large-scale synthetic text generation. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, 2024. arXiv:2407.12108.

Ahmad Beirami, Alekh Agarwal, Jonathan Berant, Alexander D'Amour, Jacob Eisenstein, Chirag Nagpal, and Ananda Theertha Suresh. Theoretical guarantees on the best-of-n alignment policy. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, 2025. arXiv:2401.01879.

Michael J. Bommarito, Jillian Bommarito, and Daniel Martin Katz. The KL3M Data Project: Copyright-Clean Training Resources for Large Language Models. arXiv:2504.07854, 2025.

Nicholas Carlini, Florian Tramèr, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Úlfar Erlingsson, Alina Oprea, and Colin Raffel. Extracting Training Data from Large Language Models. In *30th USENIX Security Symposium*, 2021. arXiv:2012.07805.

Nicholas Carlini, Daphne Ippolito, Matthew Jagielski, Katherine Lee, Florian Tramèr, and Chiyuan Zhang. Quantifying Memorization Across Neural Language Models. In *International Conference on Learning Representations (ICLR)*, 2023. arXiv:2202.07646.

Tong Chen, Akari Asai, Niloofar Mireshghallah, Sewon Min, James Grimmelmann, Yejin Choi, Hannaneh Hajishirzi, Luke Zettlemoyer, and Pang Wei Koh. CopyBench: Measuring Literal and Non-Literal Reproduction of Copyright-Protected Text in Language Model Generation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2024a. arXiv:2407.07087.

Wei-Ning Chen, Peter Kairouz, Sewoong Oh, and Zheng Xu. Randomization Techniques to Mitigate the Risk of Copyright Infringement. arXiv:2408.13278, 2024b.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training Verifiers to Solve Math Word Problems. arXiv:2110.14168, 2021.

Aloni Cohen. Blameless Users in a Clean Room: Defining Copyright Protection for Generative Models. In *Advances in Neural Information Processing Systems 38 (NeurIPS)*, 2025. arXiv:2506.19881.

Aloni Cohen and Chenhao Zhang. Barriers to counterfactual credit attribution for autoregressive models. In *International Conference on Machine Learning (ICML)*, 2026. arXiv:2605.01425.

A. Feder Cooper, Mark A. Lemley, Allison Casasola, Ahmed Ahmed, Aaron Gokaslan, Amy B. Cyphert, Christopher De Sa, Daniel E. Ho, and Percy Liang. Extracting memorized pieces of (copyrighted) books from open-weight language models. arXiv:2505.12546, 2025.

Rene L. Cruz. A calculus for network delay, part I: Network elements in isolation. *IEEE Transactions on Information Theory*, 37(1):114–131, 1991.

Amir Dembo and Ofer Zeitouni. *Large Deviations Techniques and Applications*. Springer, 2nd edition, 1998.

M. D. Donsker and S. R. S. Varadhan. Asymptotic evaluation of certain Markov process expectations for large time, I. *Communications on Pure and Applied Mathematics*, 28(1):1–47, 1975.

Yann Dubois, Balázs Galambosi, Percy Liang, and Tatsunori B. Hashimoto. Length-Controlled AlpacaEval: A Simple Way to Debias Automatic Evaluators. In *Conference on Language Modeling (COLM)*, 2024. arXiv:2404.04475.

Niva Elkin-Koren, Uri Hacohen, Roi Livni, and Shay Moran. Can Copyright be Reduced to Privacy? In *5th Symposium on Foundations of Responsible Computing (FORC)*, LIPIcs 295, 2024. doi: 10.4230/LIPIcs.FORC.2024.3. arXiv:2305.14822.

Angela Fan, Mike Lewis, and Yann Dauphin. Hierarchical Neural Story Generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2018. arXiv:1805.04833.

Vitaly Feldman and Tijana Zrnic. Individual privacy accounting via a Rényi filter, 2020. arXiv:2008.11193.

James Flemings, Meisam Razaviyayn, and Murali Annavaram. Differentially Private Next-Token Prediction of Large Language Models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, 2024. arXiv:2403.15638.

Junbo Fu, Guoshuai Zhao, Linkang Yang, Yunqi Mi, and Xueming Qian. Duplicate-Aware Controlled Code Generation: Enhancing Copyright Protection with Targeted Reordering Beam Search in Large Language Models. In *Findings of the Association for Computational Linguistics (ACL)*, 2026.

Leo Gao, John Schulman, and Jacob Hilton. Scaling Laws for Reward Model Overoptimization. In *Proceedings of the 40th International Conference on Machine Learning (ICML)*, 2023. arXiv:2210.10760.

Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, et al. Gemma 2: Improving Open Language Models at a Practical Size. arXiv:2408.00118, 2024.

Antonio Ginart, Laurens van der Maaten, James Zou, and Chuan Guo. Submix: Practical Private Prediction for Large-Scale Language Models. arXiv:2201.00971, 2022.

Aditya Golatkar, Alessandro Achille, Luca Zancato, Yu-Xiang Wang, Ashwin Swaminathan, and Stefano Soatto. CPR: Retrieval Augmented Generation for Copyright Protection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. arXiv:2403.18920.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, et al. The Llama 3 Herd of Models. arXiv:2407.21783, 2024.

Lin Gui, Cristina Gârbacea, and Victor Veitch. BoNBoN Alignment for Large Language Models and the Sweetness of Best-of-n Sampling. In *Advances in Neural Information Processing Systems 37 (NeurIPS)*, 2024. arXiv:2406.00832.

Jonathan Hayase, Alisa Liu, Noah A. Smith, and Sewoong Oh. Sampling from Your Language Model One Byte at a Time. In *Proceedings of the 43rd International Conference on Machine Learning (ICML)*, 2026. arXiv:2506.14123.

Jacqueline He, Jonathan Hayase, Wen-tau Yih, Sewoong Oh, Luke Zettlemoyer, and Pang Wei Koh. Anchored Decoding: Provably Reducing Copyright Risk for Any Language Model. arXiv:2602.07120, 2026.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring Massive Multitask Language Understanding. In *International Conference on Learning Representations (ICLR)*, 2021. arXiv:2009.03300.

Audrey Huang, Adam Block, Qinghua Liu, Nan Jiang, Akshay Krishnamurthy, and Dylan J. Foster. Is best-of-N the best of them? coverage, scaling, and optimality in inference-time alignment, 2025. arXiv:2503.21878.

John Hughes, Sara Price, Aengus Lynch, Rylan Schaeffer, Fazl Barez, Sanmi Koyejo, Henry Sleight, Erik Jones, Ethan Perez, and Mrinank Sharma. Best-of-N Jailbreaking. arXiv:2412.03556, 2024.

Daphne Ippolito, Florian Tramèr, Milad Nasr, Chiyuan Zhang, Matthew Jagielski, Katherine Lee, Christopher A. Choquette-Choo, and Nicholas Carlini. Preventing Verbatim Memorization in Language Models Gives a False Sense of Privacy. In *Proceedings of the 16th International Natural Language Generation Conference (INLG)*, 2023. arXiv:2210.17546.

Matthew Jagielski, Jonathan Ullman, and Alina Oprea. Auditing Differentially Private Machine Learning: How Private is Private SGD? In *Advances in Neural Information Processing Systems 33 (NeurIPS)*, 2020. arXiv:2006.07709.

Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. TriviaQA: A Large Scale Distantly Supervised Challenge Dataset for Reading Comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2017. arXiv:1705.03551.

Adam Tauman Kalai, Yael Tauman Kalai, and Or Zamir. Consensus sampling for safer generative AI, 2025. arXiv:2511.09493.

Nikhil Kandpal, Brian Lester, Colin Raffel, Sebastian Majstorovic, Stella Biderman, Baber Abbasi, Luca Soldaini, Enrico Shippole, A. Feder Cooper, Aviya Skowron, John Kirchenbauer, Shayne Longpre, Lintang Sutawika, Alon Albalak, Zhenlin Xu, Guilherme Penedo, Loubna Ben Allal, Elie Bakouch, John David Pressman, Honglu Fan, Dashiell Stander, Guangyu Song, Aaron Gokaslan, Tom Goldstein, Brian R. Bartoldson, Bhavya Kailkhura, and Tyler Murray. The Common Pile v0.1: An 8TB Dataset of Public Domain and Openly Licensed Text. In *Advances in Neural Information Processing Systems 38 (NeurIPS), Datasets and Benchmarks Track*, 2025. arXiv:2506.05209.

Seungone Kim, Juyoung Suk, Shayne Longpre, Bill Yuchen Lin, Jamin Shin, Sean Welleck, Graham Neubig, Moontae Lee, Kyungjae Lee, and Minjoon Seo. Prometheus 2: An Open Source Language Model Specialized in Evaluating Other Language Models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2024. arXiv:2405.01535.

Pierre-Carl Langlais, Pavel Chizhov, Catherine Arnett, Carlos Rosas Hinostroza, Mattia Nee, Eliot Krzysztof Jones, Irène Girard, David Mach, Anastasia Stasenko, and Ivan P. Yamshchikov. Common Corpus: The Largest Collection of Ethical Data for LLM Pre-Training. In *International Conference on Learning Representations (ICLR)*, 2026. arXiv:2506.01732.

Jean-Yves Le Boudec and Patrick Thiran. *Network Calculus: A Theory of Deterministic Queuing Systems for the Internet*, volume 2050 of *Lecture Notes in Computer Science*. Springer, 2001.

Xiang Li, Qianli Shen, and Kenji Kawaguchi. VA3: Virtually Assured Amplification Attack on Probabilistic Copyright Protection for Text-to-Image Generative Models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. arXiv:2312.00057.

Xuechen Li, Tianyi Zhang, Yann Dubois, Rohan Taori, Ishaan Gulrajani, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. AlpacaEval: An Automatic Evaluator of Instruction-following Models. `https://github.com/tatsu-lab/alpaca_eval`, 2023.

Chin-Yew Lin. ROUGE: A Package for Automatic Evaluation of Summaries. In *Text Summarization Branches Out (ACL Workshop)*, pp. 74–81, 2004.

Jingcheng Liu and Kunal Talwar. Private Selection from Private Candidates. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing (STOC)*, 2019. arXiv:1811.07971.

Xiaoze Liu, Ting Sun, Tianyang Xu, Feijie Wu, Cunxiang Wang, Xiaoqian Wang, and Jing Gao. SHIELD: Evaluation and Defense Strategies for Copyright Compliance in LLM Text Generation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2024. arXiv:2406.12975.

R. M. Loynes. The stability of a queue with non-independent inter-arrival and service times. *Mathematical Proceedings of the Cambridge Philosophical Society*, 58(3):497–520, 1962.

Jimit Majmudar, Christophe Dupuy, Charith Peris, Sami Smaili, Rahul Gupta, and Richard Zemel. Differentially Private Decoding in Large Language Models. arXiv:2205.13621, 2022.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen tau Yih, Pang Wei Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. FActScore: Fine-grained Atomic Evaluation of Factual Precision in Long Form Text Generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2023. arXiv:2305.14251.

Sewon Min, Suchin Gururangan, Eric Wallace, Weijia Shi, Hannaneh Hajishirzi, Noah A. Smith, and Luke Zettlemoyer. SILO Language Models: Isolating Legal Risk In a Nonparametric Datastore. In *International Conference on Learning Representations (ICLR)*, 2024. arXiv:2308.04430.

Ilya Mironov. Rényi Differential Privacy. In *IEEE 30th Computer Security Foundations Symposium (CSF)*, 2017. arXiv:1702.07476.

Lucas Monteiro Paes, Natalie Mackraz, Barry-John Theobald, and Federico Danieli. Theoretical limits of language model alignment, 2026. arXiv:2605.07105.

Sidharth Mudgal, Jong Lee, Harish Ganapathy, YaGuang Li, Tao Wang, Yanping Huang, Zhifeng Chen, Heng-Tze Cheng, Michael Collins, Trevor Strohman, Jilin Chen, Alex Beutel, and Ahmad Beirami. Controlled Decoding from Language Models. In *International Conference on Machine Learning (ICML)*, 2024. arXiv:2310.17022.

Arjun Panickssery, Samuel R. Bowman, and Shi Feng. LLM Evaluators Recognize and Favor Their Own Generations, 2024. arXiv:2404.13076.

Nicolas Papernot and Thomas Steinke. Hyperparameter Tuning with Renyi Differential Privacy. In *International Conference on Learning Representations (ICLR)*, 2022. arXiv:2110.03620.

Parjanya Prajakta Prashant, Kaustubh Ponkshe, and Babak Salimi. TokenSwap: A Lightweight Method to Disrupt Memorized Sequences in Large Language Models. In *Advances in Neural Information Processing Systems 38 (NeurIPS)*, 2025. arXiv:2502.05159.

Qwen Team. Qwen2.5 Technical Report. arXiv:2412.15115, 2025.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. SQuAD: 100,000+ Questions for Machine Comprehension of Text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2016. arXiv:1606.05250.

Ryan Rogers, Aaron Roth, Jonathan Ullman, and Salil Vadhan. Privacy Odometers and Filters: Pay-as-you-Go Composition. In *Advances in Neural Information Processing Systems 29 (NeurIPS)*, 2016. arXiv:1605.08294.

Igal Sason and Sergio Verdú. f-divergence inequalities. *IEEE Transactions on Information Theory*, 62(11):5973–6006, 2016. arXiv:1508.00335.

Christian Schroeder de Witt, Samuel Sokota, J. Zico Kolter, Jakob Foerster, and Martin Strohmeier. Perfectly secure steganography using minimum entropy coupling. In *International Conference on Learning Representations*, 2023.

Tom Segal, Asaf Shabtai, and Yuval Elovici. Provably protecting fine-tuned LLMs from training data extraction while preserving utility, 2026. arXiv:2602.00688.

Weijia Shi, Anirudh Ajith, Mengzhou Xia, Yangsibo Huang, Daogao Liu, Terra Blevins, Danqi Chen, and Luke Zettlemoyer. Detecting Pretraining Data from Large Language Models. In *International Conference on Learning Representations (ICLR)*, 2024. arXiv:2310.16789.

Gowthami Somepalli, Vasu Singla, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Diffusion Art or Digital Forgery? Investigating Data Replication in Diffusion Models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. arXiv:2212.03860.

Team OLMo, Pete Walsh, Luca Soldaini, Dirk Groeneveld, Kyle Lo, Shane Arora, Akshita Bhagia, Yuling Gu, Shengyi Huang, Matt Jordan, et al. 2 OLMo 2 Furious. arXiv:2501.00656, 2024.

Federico Tomasi, Dmitrii Moor, Alice Wang, and Mounia Lalmas. Primal-dual guided decoding for constrained discrete diffusion, 2026. arXiv:2605.09749.

Tim van Erven and Peter Harremoës. Rényi divergence and Kullback-Leibler divergence. *IEEE Transactions on Information Theory*, 60(7):3797–3820, 2014.

Claudio Mayrink Verdun, Alex Oesterling, Himabindu Lakkaraju, and Flavio P. Calmon. Soft best-of-n sampling for model alignment, 2025. arXiv:2505.03156.

J. Vijayavallabh. An Empirical Audit of k-NAF Budget Accounting for Anchored Decoding. arXiv:2605.28001, 2026.

Nikhil Vyas, Sham Kakade, and Boaz Barak. On Provable Copyright Protection for Generative Models. In *Proceedings of the 40th International Conference on Machine Learning (ICML)*, PMLR 202, 2023. arXiv:2302.10870.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-Consistency Improves Chain of Thought Reasoning in Language Models. In *International Conference on Learning Representations (ICLR)*, 2023. arXiv:2203.11171.

Boyi Wei, Weijia Shi, Yangsibo Huang, Noah A. Smith, Chiyuan Zhang, Luke Zettlemoyer, Kai Li, and Peter Henderson. Evaluating Copyright Takedown Methods for Language Models. In *Advances in Neural Information Processing Systems 37 (NeurIPS), Datasets and Benchmarks Track*, 2024. arXiv:2406.18664.

Justin Whitehouse, Aaditya Ramdas, Ryan Rogers, and Zhiwei Steven Wu. Fully Adaptive Composition in Differential Privacy. In *Proceedings of the 40th International Conference on Machine Learning (ICML)*, 2023. arXiv:2203.05481.

Joy Qiping Yang, Salman Salamatian, Ziteng Sun, Ananda Theertha Suresh, and Ahmad Beirami. Asymptotics of Language Model Alignment. In *IEEE International Symposium on Information Theory (ISIT)*, 2024. arXiv:2404.01730.

Jingyu Zhang, Jiacan Yu, Marc Marone, Benjamin Van Durme, and Daniel Khashabi. Certified Mitigation of Worst-Case LLM Copyright Infringement. arXiv:2504.16046, 2025.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena. In *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*, 2023. arXiv:2306.05685.

Guide to the appendices

Table 3 maps each theme of the paper to the appendix that holds it, and Table 4 collects the symbols and terms used throughout.

Table 3: Where each part of the evidence lives.

theme	where	what it holds
Theory	A	every proof; the threshold for a window event; tightness and composition of $\log n$
Protocol	B	workload, models, reward, judge, the recording defect and its repair
Head-to-head	C	the served table in full, the authors' metrics and byte-level decoder, judges, opponents, workloads
Selection alone	D	n past 64, other anchors, scorer size, majority vote, tasks with no judge, CoTaEval
Leakage	E	extraction, a natural memoriser, other languages, short works, vetting, contaminated anchors, audit
Cost	F	forward passes, the clock, a deployable server, batching, matched compute
The meter, audited	G	imitation cost at four pairs, onset over nine pairs, the opening, vacuity at scale
Repairs	H	sharper charge, lower cap, schedule, budget up front, frontier distance, closer anchors
Limits, related	I, J	limitations in full; positioning and incumbents measured

A Proofs

Proposition 2: tightness and extensions. If f is maximised at a single $y^\star$ with $p_s(y^\star) = p$, then $q(y^\star) = 1-(1-p)^n \to np$ as $p \to 0$, so no constant below n works in $q(y) \le n\,p_s(y)$ and the $\log n$ charge cannot be improved without assuming something about f or p_s, which is the point of assuming neither. For a deterministic g, $\Pr_q[g(y) \in A] = \Pr_q[y \in g^{-1}(A)] \le n \Pr_{p_s}[g(y) \in A]$. A rule that draws one candidate at a time and stops at any data-dependent time, capped at $n_{\max}$, serves one of at most $n_{\max}$ i.i.d. draws, and the union bound over the first $n_{\max}$ gives $\log n_{\max}$.

The certificate over m adaptive queries. Let a user issue queries $x_1, \dots, x_m$, each answered by a selection rule over n draws and each chosen from everything served so far. Write $q_j(\cdot \mid \mathcal{H}_{j-1})$ for the served law at round j and $p_{s,j}$ for the anchor's. The n draws at round j are i.i.d. $p_{s,j}$ whatever the history is, so Proposition 2 gives $q_j(y \mid \mathcal{H}_{j-1}) \le n\,p_{s,j}(y \mid \mathcal{H}_{j-1})$ pointwise, and multiplying along any realised history,

$$Q(y_1, \dots, y_m) \;=\; \prod_{j \le m} q_j(y_j \mid \mathcal{H}_{j-1}) \;\le\; n^m \prod_{j \le m} p_{s,j}(y_j \mid \mathcal{H}_{j-1}) \;=\; n^m\, P_s(y_1, \dots, y_m),$$

hence $D_\infty(Q \,\|\, P_s) \le m \log n$ exactly, with no slack from adaptivity. A per-query access-freeness guarantee does not compose this way in general (Cohen, 2025), and a per-token meter composes to mkT, growing in the length of every response. Under the 400-nat per-user cap an earlier audit proposes (Vijayavallabh, 2026), about half the median protected target's 849 nats, selection admits 192 responses at $n = 8$ at the pathwise charge, and the sharper KL bound of 1.204 nats would give 332; a meter at $k = 3$ charged its measured 165.0 nats admits 2, and none at the 600 it is certified at. The cap protects a whole target: it exceeds a 50-token window's 159.8 nats, so for a window the composed certificate lapses long before the cap is reached (Table 1). Accounting per work rather than per user does not relax this: users can pool their transcripts, so a per-work budget is shared by everyone who can query the deployment, and the responses that exhaust a window's certificate become a total rather than a per-user allowance. Only a scorer that neither holds the work nor takes a user's instructions removes the channel the cap exists for. For an untrusted scorer the cap is $\lfloor S/\log n \rfloor$ queries for the shortest event protected: at $n = 8$ and $n = 64$, 15 and 7 for a 10-token window of median surprisal 31.97 nats, 76 and 38 for a 50-token one, and for a TriviaQA answer, median 5.78 nats, 2 and 1, with 20.8% and 36.2% of answers allowing none (`results/query_caps.csv`).

Proposition 1. Write $p = \Pr_{p_s}[E] = e^{-S}$ and $a = \Pr_q[E]$. Rényi divergence satisfies data processing (van Erven & Harremoës, 2014), so passing both laws through the indicator of E gives $d_\alpha(a \,\|\, p) \le D_\alpha(q \,\|\, p_s) \le K$, where $d_\alpha(a \,\|\, p) = \frac{1}{\alpha-1} \log\big(a^\alpha p^{1-\alpha} + (1-a)^\alpha (1-p)^{1-\alpha}\big)$ for $\alpha \in (1, \infty)$, the binary KL at $\alpha = 1$ and $\log \max\{a/p,\, (1-a)/(1-p)\}$ at $\alpha = \infty$. Conversely the law $q_a = a\,p_s(\cdot \mid E) + (1-a)\,p_s(\cdot \mid E^c)$ has a constant likelihood ratio on E and on its complement, so $D_\alpha(q_a \,\|\, p_s) = d_\alpha(a \,\|\, p)$ exactly. Hence

Table 4: Notation and terms.

symbol or term	meaning
p_s, the anchor	the safe model, trained without the protected work; every certificate is relative to it as sampled
p_r	the risky model, which may hold the work; the meter tilts toward it, selection never runs it
q	the law a mechanism actually serves
x, $S(x)$, $s(x)$	a protected work; $S(x) = -\log p_s(x)$, its surprisal under the anchor in nats; $s(x)$, that surprisal per token
S_w	the anchor's surprisal of a 50-token window of a protected passage given its true prefix, median 159.8 nats
k, $T_{\max}$, K	the meter's per-token budget, the output cap (200 tokens), and its certificate $K = kT_{\max}$
n, $\log n$, m	the draws selection chooses among; its pathwise certificate (Proposition 2); $m \log n$ over m queries
L	tokens per installment when selection is spent in installments; any w-token window is then certified at $(\lceil w/L \rceil + 1) \log n$
D_α, D_∞	Rényi divergence of order α; D_∞, the max-divergence, is the pathwise order
$b_\alpha(K)$	the most a budget of K nats at order α allows an event of anchor probability e^{-S} (Proposition 1); it is 1, and the certificate vacuous, iff $K \geq S$
N_ε	the steps at which a causal policy spends more than ε (Proposition 3)
I_T, γ	the imitation cost, the KL of p_r from p_s summed over a meter's slack steps (Proposition 4); $\gamma = 0.857$ nats per token on ordinary text
$\Lambda_s^*(u)$	Lemma 1's frontier, the least divergence from p_s at which expected utility u is attainable
meter	anchored decoding (He et al., 2026): a KL budget charged per token, tilting toward p_r
selection	serving one of n anchor draws by any rule, a reward, a vote or an adversary
onset	the smallest budget at which mean near-verbatim recall reaches 0.01
near-verbatim recall	the fraction of a reference's words inside in-order near-verbatim spans of at least 20 words
level, gain	an arm's order-averaged score against one fixed opponent (0.5 is parity); its level minus its own control's
D_1–D_5	selection's gain; the meter's; their paired difference; an extra arm's gain; the extra arm minus selection
judges A–G	A, `Qwen2.5-7B` (selection's reward, never judging its own picks); B, `Phi-3.5-mini`, pre-specified; C–G, `Llama-3.1-8B`, `Qwen2.5-72B`, `Mixtral-8x7B`, `Qwen2.5-14B`, `gemma-2-27b`

$b_\alpha(K) = \max\{a : d_\alpha(a \,\|\, p) \leq K\}$, and since $d_\alpha(\cdot \,\|\, p)$ increases on $[p, 1]$, $b_\alpha(K) = 1$ if and only if $d_\alpha(1 \,\|\, p) \leq K$. At $a = 1$ every order gives $d_\alpha(1 \,\|\, p) = \log(1/p) = S$. □ The familiar closed forms relax this inversion and break at the same point: the change of measure $a \leq e^K p$ is exact at $\alpha = \infty$, while Mironov (2017)'s $a \leq (e^K p)^{(\alpha-1)/\alpha}$ is an upper bound on $b_\alpha(K)$ for $\alpha \in (1, \infty)$, not $b_\alpha(K)$ itself.

Below the threshold the orders part company. Write $S = -\log p_s(E)$ and take $K < S$. At $\alpha = \infty$ the bound is exactly e^{K-S}, and for $\alpha \in (1, \infty)$ it is at most $e^{-(1-1/\alpha)(S-K)}$, Mironov's relaxation, so both fall exponentially in the shortfall. At $\alpha = 1$ it is only linear: every q with $d(q(E) \,\|\, p_s(E)) \leq K$ has $q(E) \leq (K + \log 2)/S$, since $d(a \,\|\, e^{-S}) \geq aS - \log 2$, and the law that emits x with probability a and otherwise follows p_s costs $aS - h(a) \leq aS$ to first order in e^{-S}, so $a = K/S$ is within budget. A KL budget of half a work's surprisal therefore permits the work half the time. At $k = 0.5$, $K = 100$ nats against a 50-token window's 159.8 permits the window with probability 0.630 under the KL charge and about e^{-60} under the pathwise one (Table 1). A KL certificate that bounds the window by δ needs $K \leq \delta S$, which by Proposition 3 leaves at most $\delta S/\varepsilon$ steps above ε in expectation, 1.6 at $\delta = 0.01$ and $\varepsilon = 1$.

The threshold for a window event. Proposition 1 holds verbatim for E_w, "reproduce a w-token window of the work", and every extraction number in this paper is a window event. The scored event is a union over windows, at least as probable as any member, so each threshold in Table 5 is an upper bound on where the certificate for the scored event goes vacuous. The same holds for the union over the at most 151 positions at which a 50-token window can start in a 200-token response: Propositions 2 and 1 bound the union itself, so no

position factor enters either certificate, only a closed-form price of the anchor's rate, which covers one position; the union's rate under the anchor is the sampled quantity of Section 2.

Table 5: Vacuity thresholds by event size, from the audited anchor's per-token surprisal on 758 passages of 16 novels (median 3.197 nats per token). The budget column is the k at which $K = kT_{\max}$ reaches the threshold at $T_{\max} = 200$ (`results/window_vacuity.csv`).

event E	$S(x)$, nats	vacuous at k	$\log 64$ sits below by
10-token window	32.0	0.160	7.7×
20-token window	63.9	0.320	15.4×
50-token window	159.8	0.799	38.4×
whole passage (median, 64 tokens)	204.8	1.024	49.2×
100-token window	319.7	1.598	76.9×

At $k = 3, 5, 10, 20$, the certificate is vacuous for every event in the table; the 50-token event is vacuous at every $k \geq 0.8$, below the evaluation's smallest budget $k = 1$; and selection's $\log 64 = 4.159$ sits below every threshold.

Proposition 3. *Proposition 3, restated.* Let q be any causal policy, $q(y) = \prod_{t \leq T} q_t(y_t \mid y_{<t})$, with sequence budget $D_{\mathrm{KL}}(q \,\|\, p_s) \leq K$, and let $N_\varepsilon = \#\{t \leq T : D_{\mathrm{KL}}(q_t(\cdot \mid y_{<t}) \,\|\, p_{s,t}) > \varepsilon\}$. Then $\mathbb{E}_q[N_\varepsilon] \leq K/\varepsilon$ for every $\varepsilon > 0$.

Proof. The chain rule for an autoregressive factorisation is exact: $D_{\mathrm{KL}}(q\|p_s) = \sum_{t \leq T} \mathbb{E}_{y_{<t} \sim q}[D_{\mathrm{KL}}(q_t\|p_{s,t})]$, every term nonnegative. Markov's inequality at each step gives $\Pr_q[D_{\mathrm{KL}}(q_t\|p_{s,t}) > \varepsilon] \leq \mathbb{E}_q[D_{\mathrm{KL}}(q_t\|p_{s,t})]/\varepsilon$, and summing over t bounds $\mathbb{E}_q[N_\varepsilon]$ by K/ε. □

Surely, for the deployed class. The token bucket caps each trajectory's realised charges, $\sum_t a_t \leq K$, the accounting under which no violation occurs in more than 100,000 trajectories, so at most K/ε steps of any single served trajectory carry a charge above ε. At the authors' $k = 3$ and $T_{\max} = 200$ this is $600/\varepsilon$ against 200 steps: it constrains nothing below $\varepsilon = 3$ nats in a step, the first horn in another currency. *At finite T, in probability.* Markov on the count gives $\Pr_q[N_\varepsilon \geq K/(\varepsilon\delta)] \leq \delta$, informative only when $\varepsilon > K/(\delta T)$. For a rate, $K = kT$, the threshold is $\varepsilon > k/\delta$, independent of T: at $k = 3$ and $\delta = 0.05$ the guarantee is silent below 60 nats in a single step at every length. For a bounded K it is $\varepsilon > K/(\delta T) \to 0$: the decoder is the anchor at all but $O(1)$ steps.

Selection does not evade the chain rule, and no separation is provable at the level of laws. It is tempting to argue that a selection rule is not a causal policy at all, every token being drawn from $p_{s,t}$ so that $N_\varepsilon = 0$; that describes the sampler, not the served law. Selection's served law factorises like any other, so the proposition applies to it with $K = 3.175$ nats at $n = 64$; its conditionals differ from $p_{s,t}$ because conditioning on which draw won reweights every step, but they are not *causally computable*: $q_t(\cdot \mid y_{<t})$ depends on the scores of complete candidates. The statement "no causal policy with an $O(1)$ budget matches selection's utility" is false as a statement about laws: every law on sequences factorises, and Lemma 1's optimum $q^* \propto p_s e^{\lambda U}$ has conditionals $p_{s,t}(y_t)\, V_t(y_{\leq t})/V_{t-1}(y_{<t})$ with $V_t = \mathbb{E}_{p_s}[e^{\lambda U} \mid y_{\leq t}]$. What separates a meter from both is information: its kernel at step t is a function of $p_{s,t}$, $p_{r,t}$ and its bucket, while V_t is an expectation over continuations neither next-token distribution contains. Proposition 4 is the separation that information structure supports.

Lemma 1. The chain rule gives $D_{\mathrm{KL}}(q\|p_s) = \mathbb{E}_q[\sum_t D_{\mathrm{KL}}(q_t\|p_{s,t})]$, and for the audited decoder the step charge is exactly $c_1(\theta_t) = D_{\mathrm{KL}}(p_{\theta_t}\|p_{s,t})$, so a per-trajectory budget $\sum_t c_t \leq K$ gives $D_{\mathrm{KL}}(q\|p_s) \leq K$; for $\alpha > 1$ the charge dominates the KL (van Erven & Harremoës, 2014). (Checked, not assumed: over 1,200 trajectories and 240,000 steps at $k = 3$ the largest gap between $\sum_t a_t$ and the recorded spend is 1.5×10^{-4} nats.) Donsker–Varadhan (Donsker & Varadhan, 1975) gives, for bounded U and every λ, $D_{\mathrm{KL}}(q\|p_s) \geq \lambda\, \mathbb{E}_q[U] - \log \mathbb{E}_{p_s}[e^{\lambda U}]$; the supremum over λ is the Cramér rate function (Dembo & Zeitouni, 1998), so $K \geq \Lambda_s^*(\mathbb{E}_q[U])$. □

The bucket-metered class and its rates. Write $\ell = \log p_r - \log p_s$ and $Z(u) = \mathbb{E}_{p_s}[e^{u\ell}]$ along the geometric path $p_\theta \propto p_s^{1-\theta} p_r^\theta$. The charge at order α is

$$c_\alpha(\theta) \;=\; D_\alpha(p_\theta \,\|\, p_s) \;=\; \frac{\log Z(\alpha\theta) - \alpha \log Z(\theta)}{\alpha - 1}, \tag{2}$$

with $c_1(\theta) = \theta Z'(\theta)/Z(\theta) - \log Z(\theta)$, the deployed charge, and $c_\infty(\theta) = \theta \max_y \ell(y) - \log Z(\theta)$, the pathwise one. A decoder of the class serves, at each step, the largest θ_t whose charge fits the allowance k_t of a token bucket of rate k started at a debt δ. For a work x of N characters with per-token safe-model surprisals

$s_1, \ldots, s_T$, N_t characters after t tokens and $S(x) = \sum_t s_t$, three rates govern the audit:

$$\underbrace{s(x) = \frac{S(x)}{N}}_{\text{surprisal rate}}, \qquad \underbrace{k_{\text{crit}}(x) = \max_{t \le T} \frac{\sum_{i \le t} s_i + \delta}{N_t}}_{\text{Loynes running maximum}}, \qquad \underbrace{c_{\text{use}} = \text{med}_{y \sim p_r} \frac{\log p_r(y) - \log p_s(y)}{|y|}}_{\text{divergence rate of ordinary traffic}}, \tag{3}$$

per character where safe models with different vocabularies are compared and per token where a rate meets a budget.

Proposition 4. The charge c is nondecreasing on $[0, 1]$ with $c(0) = 0$, and the decoder serves $\theta_t = \max\{\theta : c(\theta) \le k_t\}$. At a slack step, $c(1) \le k_t$, so $\theta_t = 1$: the decoder serves $p_{r,t}$ and is charged $D_{\text{KL}}(p_{r,t} \| p_{s,t})$. By the chain rule $D_{\text{KL}}(q \| p_s) = \mathbb{E}_q[\sum_t D_{\text{KL}}(p_{\theta_t} \| p_{s,t})]$, and dropping the nonnegative terms of the other steps gives the first claim. For the second, Λ_s^* is convex, vanishes at $\mathbb{E}_{p_s}[U]$ and is nondecreasing on $[\mathbb{E}_{p_s}[U], u_{\max}]$, where $\Lambda_s^*(u_{\max}) = -\log \Pr_{p_s}[U = u_{\max}] \le \log(1/\pi)$; so for a decoder that does not lower $\mathbb{E}[U]$, $\Lambda_s^*(\mathbb{E}_q[U]) \le \log(1/\pi)$, and $D_{\text{KL}}(q \| p_s) \ge \gamma T$ gives the ratio. The proviso is substantive: a utility attained by one exact long string has $\Lambda_s^* = \Theta(T)$ and no separation survives. Winning a judged comparison is not of that kind: the anchor's own unassisted draw wins both presentation orders on 2.2% of prompts, so winning every one, at any length, would cost an optimal policy 3.82 nats, and the decoder spends 44.9 times that to reach parity; on a single-order pass, where a win in the one order shown comes on over a quarter of prompts, the same reading is 1.30 nats and 132 times (Appendix H). □

Protection outruns certification.

Proposition 5. *Take $s(x)$ and $k_{\text{crit}}(x)$ of Eq. (3) per token. Under the banking rule, if the decoder emits each token of x with probability at least $1 - \eta$ then $k \ge (1 - \eta)\, k_{\text{crit}}(x) - \log 2$, and $k_{\text{crit}}(x) \ge s(x) + \delta/T$ for every work, with equality iff the running ratio is maximised at the last token.*

Proof. By data processing on the indicator of the true token, the step charge is at least the binary divergence between the served and reference probabilities of that token, $c_i \ge d(q \,\|\, e^{-s_i}) \ge (1 - \eta)s_i - \log 2$ whenever $q \ge 1 - \eta$. The banking rule (He et al., 2026, Algorithm 1) caps the charge on the first t tokens at $tk - \delta$, so $tk \ge (1-\eta) \sum_{i \le t} s_i + \delta - t \log 2$, which is at least $(1-\eta)(\sum_{i \le t} s_i + \delta) - t \log 2$; dividing by t and maximising over t gives the first claim, and a running maximum dominates its endpoint $(S(x) + \delta)/T$. □Equality holds exactly when the running ratio peaks at the last token; under a constant rate with $\delta > 0$ it is $s(x) + \delta/t$, strictly decreasing, so the maximum sits at $t = 1$ and the inequality is strict. k_{crit} is the Loynes running maximum of a queue's workload (Loynes, 1962), equivalently the smallest rate of a (σ, ρ) arrival curve admitting the surprisal process (Cruz, 1991; Le Boudec & Thiran, 2001): a budget publishes a *drift* where protection is a *workload maximum*. It bounds regime boundaries rather than screening works: as a predictor of the onset it misses by 21.9% against a no-change null's 12.3%.

B Protocol

Workload. The judged pass uses 500 prompts in three classes (Table 6); none is drawn from the protected corpus, none shares a word 8-gram with any protected passage or names any of its sixteen works (`results/prompt_set_profile.csv`), and all three sets ship with the artifact. The anchor's order-averaged win rate against the opponent runs from 0.3667 on biographies to 0.5067 on story premises.

Table 6: The 500 judged prompts, by class, with the harness's name for each in brackets: questions written for this project (`neutral`), biography requests *Tell me a bio of X.* over FActScore's entities (Min et al., 2023) (`factual`), and r/WritingPrompts premises (Fan et al., 2018) (`creative`). Lengths in words, median with quartiles and range; *header* counts prompts served with `Complete the prefix:`; *anchor wins* is the anchor's $n = 1$ completion against the opponent, order-averaged.

class	source	n	header	prompt, words	opponent, words	anchor wins
questions	written here	200	200	10 [9, 12], 6–21	158 [147, 166]	0.4575
biographies	FActScore	150	0	7 [7, 7], 6–9	151.5 [143, 159]	0.3667
story premises	r/WritingPrompts	150	150	23 [17, 35], 6–40	157 [146, 168]	0.5067

The scorer, and the one that failed. The deployed reward is $\log p(\text{Yes}) - \log p(\text{No})$ from `Qwen2.5-7B-Instruct` on one fixed template; it never consults the risky model. The first scorer we specified was the risky model's own per-token log-likelihood, the objective anchored decoding tilts toward. At $n = 8$ it moves judged utility from 0.319 to 0.313, -0.006 against a band of $+0.077$; selecting by the judge's

own verdict among the same eight draws reaches $+0.488$ (circular, an upper bound), and selecting by judge A while scoring with judge B $+0.081$ $[+0.034, +0.130]$ (order-averaging it leaves $+0.0405$ $[+0.019, +0.063]$), so the utility is in the draws and the likelihood does not find it. Within a prompt, over $5{,}687$ pairs the judge placed differently, the per-token likelihood is concordant with the judge on 0.526 $[0.504, 0.548]$, the summed likelihood on 0.478 $[0.454, 0.500]$ and the completion's length on 0.537 $[0.514, 0.560]$: no better than length, and not worse, the difference being $+0.011$ $[-0.023, +0.046]$ (`results/currency_auc.csv`).

The scorer prefers an empty completion. On the recovered text 13.9% of the headline pool's $32{,}000$ draws are empty, and the reward scores them above the rest (median -18.75 against -26.12), so best-of-n serves an empty text on 12.8% of prompts at $n = 1$, 21.8% at $n = 8$ and 8.2% at $n = 64$. The judge does not score those as losses, and at $n \geq 8$ the preference costs selection level rather than flattering it (prompts served empty are judged 0.4531, 0.4771 and 0.4939 at $n = 1$, 8 and 64, against 0.4536, 0.5211 and 0.5605 for the rest). Excluding empty candidates from the argmax raises the fresh-seed headline difference from $+0.0495$ to $+0.0695$ $[+0.0385, +0.0995]$ (`results/empty_preference.csv`). On the repaired headline pass (post hoc, no band), scoring every empty served text as a loss moves the difference from $+0.0505$ to $+0.0155$ $[-0.0245, +0.0555]$: selection serves 41 empty texts at $n = 64$ and the meter none, against 64 and 62 for the two anchor controls. Serving the best *non-empty* draw, a selection rule over the same n draws and so certified at the same $\log n$, changes 41 picks; judged as the judge scores them it leaves the difference at $+0.0505$, since the judge rates those 41 replacements as it rated the empty texts, and with empty answers as losses it keeps $+0.056$ $[+0.0195, +0.093]$, the served levels differing by $+0.0655$ $[+0.039, +0.0915]$ (`results/empty_as_loss.csv`, `results/nonempty_rule.csv`). Every GPU was held by another job when the 41 replacements were judged, so judge B ran on CPU in float32; re-judging the 41 committed picks the same way reproduces their committed levels on 37. Re-judged in one pass per configuration on one machine, the difference keeps its reading under the non-empty rule at all eight configurations of Table 10 we re-ran, the headline at $+0.058$ $[+0.0225, +0.093]$ against the committed rule's $+0.059$ $[+0.0235, +0.094]$ in the same pass (Table 7, `results/nonempty_tables.csv`).

Table 7: The non-empty rule at eight configurations of Table 10: the difference of gains (selection's over its $n = 1$ control minus the meter's over its anchor), judge B, both orders, all eight passes on one machine, under the committed rule and under the rule that serves the best non-empty draw (41, 14 and 30 picks change in the three pools). Last column: the committed rule's difference here minus the same row's pass behind Table 10; order averaging draws nothing, so a shift is how the judge ran, not a re-roll.

configuration	committed rule	non-empty rule	shift
released 8B, $k = 10$ (headline)	$+0.059$ $[+0.0235, +0.094]$	$+0.058$ $[+0.0225, +0.093]$	$+0.0085$
70B base, $k = 20$	$+0.0815$ $[+0.0485, +0.1155]$	$+0.0805$ $[+0.0475, +0.113]$	-0.001
70B base, $k = 0.5$	$+0.1085$ $[+0.078, +0.1395]$	$+0.1075$ $[+0.0765, +0.1375]$	$+0.0035$
chat template, $k = 10$	-0.145 $[-0.184, -0.1065]$	-0.1485 $[-0.187, -0.11]$	$+0.0035$
chat template, $k = 3$	-0.051 $[-0.0885, -0.0135]$	-0.0545 $[-0.091, -0.0175]$	0
temperature 0.7, 8B, $k = 10$	-0.081 $[-0.113, -0.049]$	-0.0825 $[-0.114, -0.051]$	0
temperature 0.7, 70B, $k = 20$	$+0.0485$ $[+0.0145, +0.082]$	$+0.047$ $[+0.014, +0.081]$	0
AnchoredByte, $k = 0.1$	$+0.1385$ $[+0.1155, +0.162]$	$+0.1475$ $[+0.125, +0.1705]$	0

The judge. Every judged number compares an arm's served text with the fixed opponent's on the same prompt. The judge receives one user turn, in its own chat template:

```
You are grading two candidate responses to the same instruction.
Instruction: {prompt}  Response A: {a}  Response B: {b}
Which response is the better answer to the instruction? Judge
helpfulness, coherence and factual accuracy. Ignore length and
formatting. If they are of equal quality, answer Tie. Reply with
exactly one word: A, B, or Tie.
```

Fields are truncated to $1{,}200$ characters, the prompt is the corpus prompt without the harness header, decoding is greedy with at most four new tokens, and the verdict is the first letter (anything else a tie). The cut reaches 19 of the opponent's 500 texts, 20 of the meter's at $k = 10$, 2 of selection's and no prompt; shown whole, the headline difference reads $+0.062$ $[+0.0265, +0.0975]$ against $+0.059$ cut, on the same machine, and $+0.0217$ $[-0.0517, +0.0933]$ on the biographies alone (`results/full_response.csv`, registered). An order-averaged score shows the item in both orders and averages the two (`analysis/order_averaged_h2h.py`). **Position dominates a single order** (Zheng et al., 2023): the same two texts win 261 of 500 shown second and 24 shown first, only 29.2% of items get a consistent verdict, the first slot wins 12.7%

of decided comparisons, and a single-order level moves when other arms join the sweep (the audited anchor's $n = 64$ gain read $+0.142$ and $+0.076$ on byte-identical text). Order averaging draws nothing and reproduced exactly, $+0.1045$ $[+0.0820, +0.1280]$ both times, which is why every head-to-head here is order-averaged and single-order gains are read only within one sweep. The correction is asymmetric: read in one order against the order-averaged construction, selection at $n = 64$ moves $+0.110$ to $+0.1045$ and the meter at $k = 10$ moves $+0.097$ to $+0.0400$, so the paired difference moves $+0.013$ to $+0.0645$. We had predicted the opposite, both arms deflating together, and it was wrong; the likely reason, offered rather than claimed, is that at $k = 10$ the meter serves p_r nearly unchanged, leaving the judge two near-identical texts and little but slot order. The same arm found a defect behind earlier numbers: both judging scripts showed the judge a prompt reconstructed as the served text minus the generation, whose split point moves with how much each arm generated, so for the same item three arms disagreed about it in 455 of 500 cases. Every judged number here takes the prompt from the corpus, and under one true prompt the single-order difference is $+0.013$, not the $+0.070$ the arm-specific prompts implied. A judged gain re-run on a different n-grid moves by 0.04 to 0.07; the head-to-head escapes that floor by being paired within one pass. Judge C is the opponent's own checkpoint, `Llama-3.1-8B-Instruct`, a self-preference risk (Panickssery et al., 2024).

Tempering the anchor instead of selecting. Tempering is a per-token tilt of the untempered anchor, so if it alone bought selection's gain a deployer could have had it by choosing another reference law (Section 2). Registered before any token (`results/onset_prediction_tempering.md`), the anchor alone was drawn at temperature 0.7, at 0.5 and at the authors' 0.7 with penalty 1.1 on the 500 prompts and judged against the committed opponent in both orders: judge B on the machine type that judged Table 2, judge G with every arm re-judged on one H100 host, where the committed levels reproduced exactly. Tempering buys part of the gain, about a third under judge B and about two thirds under judge G, and best-of-64 leads every tempered anchor under both, as predicted for judge G (Table 8). The tempered anchors also serve more empty answers, which judge G scores at 0.68 to 0.74, so their gains under it are if anything flattering. Top-p truncation was not run: the decoder has no nucleus option.

Table 8: **Tempering the anchor instead of selecting**, 500 prompts against the committed opponent, both presentation orders. *Gain*: level minus that of one untempered anchor draw (selection's $n = 1$), with 95% interval; *empty*: empty completions served. Below the rule, best-of-64 at $T = 1.0$ minus the tempered anchor, paired over prompts. Judge B on the machine type of Table 2; judge G re-judged every arm on one H100 host.

arm	empty	gain, judge B	gain, judge G
best of 64, $T = 1.0$	41	$+0.1015$ $[+0.0770, +0.1260]$	$+0.1760$ $[+0.1400, +0.2125]$
anchor, $T = 0.7$	138	$+0.0330$ $[+0.0095, +0.0575]$	$+0.1130$ $[+0.0775, +0.1480]$
anchor, $T = 0.5$	167	$+0.0325$ $[+0.0090, +0.0560]$	$+0.1160$ $[+0.0800, +0.1515]$
anchor, $T = 0.7$, penalty 1.1	75	$+0.0650$ $[+0.0405, +0.0895]$	$+0.1250$ $[+0.0880, +0.1615]$
second draw, $T = 1.0$	62	-0.0150 $[-0.0365, +0.0065]$	-0.0005 $[-0.0330, +0.0315]$
best of 64 − anchor, $T = 0.7$		$+0.0685$ $[+0.0455, +0.0915]$	$+0.0630$ $[+0.0290, +0.0960]$
best of 64 − anchor, $T = 0.5$		$+0.0690$ $[+0.0460, +0.0915]$	$+0.0600$ $[+0.0255, +0.0955]$
best of 64 − anchor, 0.7, penalty 1.1		$+0.0365$ $[+0.0135, +0.0595]$	$+0.0510$ $[+0.0165, +0.0850]$

A vetted scorer. The vetting condition of Section 2 asks for a scorer that neither holds the work nor takes the user's instructions, and ours meets neither. Registered before any score (`results/onset_prediction_promptblind.md`), Comma-7B, trained on openly licensed text and the largest anchor to pass the vetting screen, scored each of the 64 committed draws by its mean per-token log-likelihood with no prompt in context, never serving an empty draw. It removes the query cap and most of the gain with it: over one draw it gains $+0.0405$ $[+0.0165, +0.0645]$ under judge B and $+0.0120$ $[-0.0230, +0.0475]$ under judge G, where the Qwen scorer gains $+0.1015$ and $+0.1760$, differences of $+0.0610$ $[+0.0375, +0.0840]$ and $+0.1640$ $[+0.1300, +0.1980]$. We had predicted a vetted scorer would keep part of the gain under judge G; it did not, so a vetted scorer that keeps it is open.

How far the judges can be trusted. Six judges read the headline's texts, and the share of items on which a judge gives the same verdict in both presentation orders runs 0.288 to 0.370 over the four arms for B, 0.630 to 0.712 for C, 0.756 to 0.796 for D (`Qwen2.5-72B`), 0.354 to 0.420 for E (`Mixtral-8x7B`), 0.676 to 0.774 for F and 0.784 to 0.846 for G (`gemma-2-27b-it`) (`results/judge_order_consistency.csv`). Three checks of the pre-specified judge B, all in one pass that reproduces the committed levels on 500 of 500 prompts (`results/windowed_meter.csv`). It detects a known difference: a second draw of the risky model gains $+0.065$ $[+0.0395, +0.090]$ over the anchor ($+0.1485$ under G). The $k = 0.5$ null is informative: it had 80% power against $+0.0305$, and its 90% interval, $[-0.0205, +0.015]$, sits inside half of selection's gain, so

a two one-sided test calls it equivalent to zero at that margin (post hoc). And on the 62 prompts where both compared arms get order-consistent verdicts the headline difference is $+0.121$ $[+0.024, +0.222]$ ($+0.095$ on the 21 where all four arms do, too few to resolve). Neither judge scores an empty answer as a loss. B scores selection's 41 empty picks at 0.494 against the risky model's text, and G at 0.634, above the arm's own 0.545: on those prompts the templateless 8B's continuation is often off task, and through the chat template G scores the same empties at 0.2195. Counted instead as losses (post hoc), the matched differences of Table 2 hold under G at every budget but $W = 125$, and under B against the per-token meters ($+0.094$) and the windowed meter at $\log 64$ ($+0.048$) but not at 24.95 or 40 nats ($+0.021$ each); and under G the non-empty rule's headline difference is $+0.0335$ $[-0.020, +0.0875]$, so under the most reliable judge the headline rests partly on the empties. No human rated anything.

Scorer family by judge family. The largest judged reading comes from a judge in the Qwen scorer's family, so we scored the headline pool a second time with `gemma-2-27b-it` and judged both selections under all six judges on one host, registered in advance (`results/onset_prediction_scorer_judge_factorial.md`; Table 9). We predicted that the Qwen reward's drafts would be preferred under at least five of the six judges. Only judge B prefers them. Under C, D, E and F the two scorers are unresolved, and under G, the gemma scorer's own family, the gemma-scored drafts win. So there is a same-family preference in the gemma family and none measurable in the Qwen family. The Qwen-family judges D and F read the gemma-scored headline at $+0.0765$ and $+0.1325$, as high as the Qwen-scored one. With the gemma scorer the headline resolves under four of six judges, against five with the Qwen scorer, and B, where it depends on the scorer, is one of the two that do not resolve it. At the matched $\log 64$ the gemma-scored selection still beats the windowed meter, by $+0.054$ $[+0.0215, +0.086]$ under B, and the budget-up-front meter, by $+0.0455$ $[+0.014, +0.0775]$, and both margins are confirmed under G. Every same-host pass gives the committed panel's difference exactly. Judge B's, committed on a local A100, reads $+0.059$ here against $+0.0505$ there.

Table 9: The headline scored twice and judged six times, one host, 500 prompts, order-averaged, 95% intervals. S: the Qwen-scored selection's level minus the gemma-scored one's. D_3: selection's gain minus the $k = 10$ meter's, each over its own control (`results/scorer_judge_factorial.csv`).

judge	S	D_3, Qwen-scored	D_3, gemma-scored
B	$+0.0440$ $[+0.0230, +0.0650]$	$+0.0590$ $[+0.0240, +0.0935]$	$+0.0150$ $[-0.0185, +0.0490]$
C	$+0.0175$ $[-0.0095, +0.0455]$	$+0.0765$ $[+0.0285, +0.1265]$	$+0.0590$ $[+0.0100, +0.1090]$
D	-0.0215 $[-0.0455, +0.0020]$	$+0.0550$ $[+0.0075, +0.1035]$	$+0.0765$ $[+0.0285, +0.1260]$
E	$+0.0100$ $[-0.0185, +0.0375]$	-0.0015 $[-0.0450, +0.0425]$	-0.0115 $[-0.0560, +0.0330]$
F	-0.0115 $[-0.0345, +0.0115]$	$+0.1210$ $[+0.0770, +0.1655]$	$+0.1325$ $[+0.0890, +0.1765]$
G	-0.0400 $[-0.0670, -0.0135]$	$+0.0545$ $[+0.0030, +0.1080]$	$+0.0945$ $[+0.0390, +0.1480]$

A recording defect, and its repair. The harness recorded each generation by slicing a left-padded batch row at that row's own prompt length, so a row with p pad tokens carried the last p tokens of its prompt into the recorded text: the opponent's and the anchor's texts on 73.6% of headline prompts (mean 10.7 characters), the $k = 10$ meter's on 32.6%. Nothing served was affected, but the judge and the reward read the record. The slicing is fixed and every recorded generation is recovered exactly as the decoded row minus its prompt. Re-judged on the recovered text with nothing else changed, the headline difference moves from $+0.0645$ $[+0.0300, +0.0995]$ to $+0.0505$ $[+0.0155, +0.0860]$, and all twenty older order-averaged differences keep their sign class; of the forty single-arm gains beneath them three change class, selection's on MT-Bench ($+0.0375$ $[-0.0125, +0.0875]$) and CoTaEval-QA ($+0.0070$ $[-0.0165, +0.0300]$) now covering zero and the meter's on the public books falling below it (-0.0275 $[-0.0470, -0.0080]$) (`results/deecho_rejudge.csv`).

Pre-registration. Every band this paper scores against was committed before the run it scores, with its grid, admission criterion, stop rule and excluded outcomes; each registration and its scoring log are in the repository (`results/onset_prediction_*.md`). Predictions that failed are reported as results: the first scorer, the judge-C prediction, the 70B pair's expected wins for the meter, the CoTaEval turnover, the adversarial oracle on CoTaEval's infringement half, and the monotone vetting ladder. Figure 3's NewsQA row is registered, its $k = 1.4$ fixed by a registered calibration rule before any judge call. Read as one family of 21 under Holm's correction, each p-value taken from its row's interval, eight of Figure 3's rows still exclude zero, seven in selection's favour and AlpacaEval against it; the headline row alone ($p \approx 0.005$) does not survive a 21-way correction (`results/forest_multiplicity.csv`, post hoc).

C The head-to-head in full

Table 10: **Against the risky model: selection and the metered decoder, by how the risky model is served.** Order-averaged judge-B levels against one opponent per block, the risky model's own draw (0.5 is parity); 500 prompts, temperature 1.0 without repetition penalty except in the block at the authors' 0.7 and 1.1, whose K/S_w is against the anchor's $S_w = 177.4$ nats at 0.7 with the penalty 1.1 (179.1 without it). *active*: share of the judged trajectories' decode steps whose served token is a strict blend of the two models; K/S_w: the certificate against a 50-token window ($S_w = 159.8$ nats; 139.7 under Comma-7B), which excludes nothing at or above 1. Last column: selection at $n = 64$ (3.175 nats, a bound attained only for a tie-free score and unrepeated strings, Appendix D) minus the meter, paired over prompts, 95% interval. Levels compare within a block only (Appendix C).

risky model, as served	k	active	K/S_w	meter	selection − meter
opponent: the 8B-Instruct without its chat template; anchor alone 0.439*, selection* 0.555					
8B-Instruct, continuing text	0.5	49.1%	0.63	0.436	+0.1195 [+0.0955, +0.1435]
	1	6.6%	1.25	0.456	+0.0995 [+0.074, +0.1245]
	10	0.008%	12.5	0.490	+0.0655 [+0.040, +0.092]
70B base (the authors' pair)	0.5	26.5%	0.63	0.429	+0.1265 [+0.103, +0.1495]
	1	6.3%	1.25	0.433	+0.1225 [+0.098, +0.146]
	20	0.00%	25.0	0.451	+0.1040 [+0.080, +0.1285]
at the authors' 0.7 *and* 1.1*, other host; opponent: the 8B's own draw there; anchor* 0.312*, selection* 0.427					
8B-Instruct, continuing text	0.5	44.6%	0.56	0.394	+0.0325 [+0.008, +0.0565]
	1	16.4%	1.13	0.441	−0.0145 [−0.039, +0.010]
	10	0.040%	11.3	0.500	−0.0730 [−0.096, −0.0515]
70B base (the authors' pair)	0.5	46.4%	0.56	0.330	+0.0970 [+0.073, +0.1215]
	1	20.2%	1.13	0.319	+0.1080 [+0.083, +0.133]
	20	0.00%	22.5	0.370	+0.0565 [+0.0305, +0.082]
AnchoredByte, k per byte, own pass: Comma-7B alone 0.4660*, selection at Comma-7B* 0.6245					
70B base with Comma-7B	0.1	18.4%	0.57	0.4860	+0.1385 [+0.1155, +0.1620]
	0.5	2.8%	2.86	0.5005	+0.1240 [+0.1010, +0.1470]
	2	0.6%	11.45	0.4900	+0.1345 [+0.1105, +0.1590]
opponent: 8B-Instruct via its chat template; anchor 0.238*, selection* 0.363 *(†other host:* 0.2385*,* 0.367*)*					
8B-Instruct, chat template	0.5†	62.7%	0.63	0.248	+0.1195 [+0.0925, +0.1465]
	1	16.8%	1.25	0.322	+0.0410 [+0.012, +0.0705]
	2†	2.5%	2.50	0.384	−0.0170 [−0.046, +0.0125]
	3†	1.1%	3.75	0.405	−0.0380 [−0.068, −0.008]
	5†	0.23%	6.26	0.427	−0.0600 [−0.091, −0.029]
	10	0.16%	12.5	0.501	−0.1380 [−0.167, −0.107]

By serving configuration. Table 10 comes from two registered arms, banded before any judge call, and one descriptive re-judge that puts every budget in one pass; all three judge the same selection texts, whose per-prompt levels agree across them on 500 of 500 prompts. Served through its chat template the 8B was generated afresh at $k = 10$; its own second draw reads 0.504 against the first, and it beats its templateless text by +0.184 [+0.155, +0.213]. Against that stronger opponent selection still beats the committed (templateless) meter, +0.058 [+0.0325, +0.0845]. At the 70B pair we predicted the meter would win at $k = 20$ and at $k = 1$, and both predictions were wrong: at temperature 1.0 the base model continuing `Complete the prefix:` is judged no better than the anchor, and at $k = 20$ the meter serves its text byte for byte on 500 of 500 prompts. Two post-hoc checks: the meter writes about twice selection's words (151 against 75 at the median), and a length-controlled regression (Dubois et al., 2024) leaves the difference at +0.0605 [+0.0236, +0.0971]; on the 414 prompts whose two anchor controls are non-empty it is +0.0441 [+0.0024, +0.0864] (`results/h2h_length_control_note.md`). Read with a Bonferroni correction over its rows, every row of Table 10's released and chat blocks still excludes zero in the same direction, the narrowest being the chat-template $k = 1$ row at [+0.0015, +0.0820] (`results/served_multiplicity.csv`, post hoc). Served through the same chat template, the audited memoriser alone reaches near-verbatim recall 0.2193 (0.4921 on plain text); the meter stays below 0.01 at every budget through $k = 3.8$ and reaches it at $k = 3.941$ [2.968, 4.435], 1.22 of $s(x)$, with 25.3% of resamples never reaching it by $k = 4.5$, so the upper end is a floor (`results/chat_onset.csv`, registered). The chat-served meter's first win, at $k = 3$, lies within that interval, but at $k = 3$ itself measured recall is below 0.01: the meter wins there before any measured extraction, the regime the Limitations weigh against a certificate.

The 70B base as an opponent. Is the base model at temperature 1.0, judged no better than the anchor, simply degenerate? By the usual markers it is not: over the 500 prompts its median response is 144 words against the 8B-Instruct's 154 and the anchor's 63, it is empty on 1.0% (the anchor 12.4%), and a single 5-gram fills a fifth of the text on 2.2% of responses against the anchor's 17.8%. What it does is continue the prompt as a web page: asked for a story it writes the next post of a forum thread, signed with a username and followed by replies. It is fluent and off task, so the comparison is with a capable model that does not follow instructions, which is how its authors serve it (`results/opponent_degeneracy.csv`).

A difference of gains, or of levels. Table 10's last column is the direct difference of served levels; the text quotes the difference of gains, each arm's level minus its own zero-budget control's, which was the registered construction because the two pipelines' controls need not agree. They do not quite: selection's $n = 1$ draw is judged 0.008 to 0.0215 above the meter's anchor-alone arm in every configuration but AnchoredByte, where the two are one arm. So the difference of gains is the less favourable to selection of the two, never the more, and over all 21 rows of the table the two give the same reading on 19; the two that differ are temperature 0.7 at $k = 0.5$ ($+0.0325$ against $+0.0245$ $[-0.0090, +0.0570]$) and the chat template at $k = 1$ ($+0.0410$ against $+0.0305$ $[-0.0030, +0.0655]$), each a direct win that is a tie on gains (`results/d3_vs_direct.csv`).

Under the authors' own utility metrics. He et al. (2026) score utility with Prometheus-2 fluency and FActScore precision, so we scored the same recovered texts with both: their fluency rubric verbatim under `prometheus-7b-v2.0` (Kim et al., 2024) where they use `gpt-4.1-mini`, and FActScore (Min et al., 2023) on the 150 biographies with `Qwen2.5-14B-Instruct` extracting and verifying against Wikipedia. On fluency (1–5, non-empty responses) selection at $n = 64$ scores 2.56 against the meter's 1.88 at $k = 10$, a paired $+0.70$ $[+0.53, +0.85]$, beats the meter at every budget of both text-continuation pairs ($+1.15$ $[+1.00, +1.30]$ over the 70B pair at $k = 0.5$), and loses to the chat-template model, -0.81 $[-0.99, -0.64]$: the judge's pattern. FActScore reads the other way: selection's precision is its anchor's (0.042 against 0.048, paired -0.008 $[-0.041, +0.022]$) and below every metered arm (-0.182 $[-0.222, -0.145]$ against $k = 10$), while it abstains on 20 of 150 biographies where a single draw abstains on 80. Best-of-n is bounded by its best draw, not its average: FActScored draw by draw, the most precise of the 64 reaches 0.5054 $[0.4553, 0.5564]$ (0.2392 among draws claiming at least five facts) where one draw averages 0.0613 and the reward's pick 0.0431, so the draws hold far more precise biographies than the reward serves. Asked instead whether every claim is accurate, the same scorer serves 0.0768 by claiming fewer facts and abstaining on 45, paired $+0.0102$ $[-0.0144, +0.0399]$ over the committed pick and -0.1246 $[-0.1727, -0.0725]$ below the meter at $k = 10$: a scorer that asks for facts did not find them (`results/factscore_oracle.csv`, registered). The head-to-head is a statement about fluency and helpfulness, not about factual recall, which the anchor bounds (`results/he_metrics.csv`).

The authors' byte-level decoder. He et al. (2026) meter every licensed anchor but TinyComma with AnchoredByte, which fuses a Comma tokenizer with a Llama-3 one through ByteSampler (Hayase et al., 2026), and recommend Comma-7B with the 70B base at $k = 0.5$ per byte. We ran their package unmodified (`anchoreddecode` at commit `a12ecd9`) in Table 10's configuration, with $B_{\max} = 800$ bytes so that $K = 800k$ nats and $S_w = 139.7$ under Comma-7B; selection is Comma-7B's best-of-64. Every trajectory stays within K. Selection wins at every budget, as predicted, including $k = 0.1$, where the certificate says something about a window ($K = 80$) and the meter binds on 18.4% of byte steps and is judged indistinguishable from Comma-7B alone. Under the authors' metrics selection is the more fluent at every budget; at $k = 0.5$ the meter's FActScore precision is 0.323 against selection's 0.0995, the 70B's facts passing through a certificate that is vacuous for any window ($K/S_w = 2.86$), and at $k = 0.1$ it is 0.0843, inseparable from selection's (`results/onset_prediction_anchoredbyte.md`). At their temperature 0.7 and penalty 1.1 (registered; their `generate` refuses $k = 0$, so the anchor control is their byte path at $k = 10^{-6}$, forced to the anchor at every byte), the meter at $k = 0.1$ binds on 22.9% of byte steps and again gains nothing over its anchor, $+0.008$ $[-0.008, +0.0245]$, while TinyComma's selection at those settings gains $+0.099$ $[+0.068, +0.129]$ more over its own; the tempered Comma-7B has $S_w = 155.3$ nats against 140.7 untempered, both over the 758 protected passages, so $K/S_w = 0.515$ (`results/anchoredbyte_t07.csv`); the 140.7 re-measures the construction behind the temperature-1.0 arm's 139.7, the audited anchor's median rescaled by per-character rates, 1 nat higher.

Other judges and fresh draws. Table 11 gives the recorded-text readings the repaired ones in Figure 3 replaced. Judges D to G re-score byte-identical text, so every difference between their rows is the instrument. `Mixtral-8x7B`, the one judge whose interval contains zero, re-asked on the 850-prompt pass at a binding budget reads $+0.0226$ $[-0.0018, +0.0476]$: the same sign as the others and a small, not an absent, effect. Two earlier attempts ran on AlpacaEval, where the reversal does not hold, and were refused by their gates; the extra prompts cut a half-width of 0.0443 to 0.0247, and the dispute is the meter's ($+0.0971$ under Mixtral, $+0.0253$ under judge B). Family and size do not account for it (3.8B, 14B, 27B and 72B resolve; 47B does not). The judges disagree item by item (per-prompt correlation $+0.284$ between D and E), so we report the fraction that resolve and never average them. Order consistency, the share of prompts whose two presentation orders agree, spans 0.29–0.37 over the four arms for judge B, 0.35–0.42 for E, 0.63–0.71 for C, 0.68–0.77 for F, 0.76–0.80 for D and 0.78–0.85 for G; the two most consistent judges both place the repaired difference

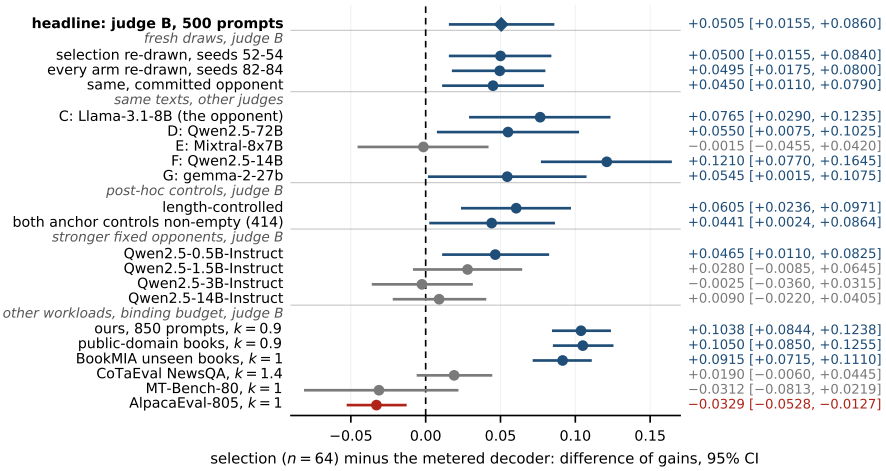

Figure 3: **The head-to-head difference under every perturbation we ran.** Each row is selection's order-averaged gain at $n = 64$ over its own anchor-alone control minus the meter's at $k = 10$ (last block: the named k, rate-matched to bind), paired over prompts, 95% interval, all on the repaired text. Blue excludes zero above, red below. Judges C–G re-score the headline's texts; C is the opponent's own checkpoint. The controls regress out the length gap (the meter writes 151 words to selection's 75 at the median) and drop the 86 prompts whose anchor controls are empty. Opponents: fixed Qwen2.5-Instruct models, each stronger than the committed one (Appendix C).

above zero (`results/judge_order_consistency.csv`). Two readings are reported against us: the seed-52 judge-C estimate, $+0.1000$, falls just outside the original interval on the favourable side and is never quoted as the headline; and we predicted judge C, the opponent's own checkpoint, would be a harsh test, and it did the opposite, so that prediction is withdrawn.

Table 11: The head-to-head repeated, on the recorded text unless marked. Paired difference D_3 of order-averaged gains within each pass; judges B and C as in the main text, D–G re-score the seed-42 texts. *seeds* 82: every sampled arm re-drawn, judged against a re-drawn opponent (*all new*) or the committed one (*old opp.*).

draw	judge	selection D_1	metered D_2	**difference** D_3
seed 42 (original)	B	$+0.1045$	$+0.0400$	$+$**0.0645** $[+0.0300, +0.0995]$
repaired text	B	$+0.1015$	$+0.0510$	$+$**0.0505** $[+0.0155, +0.0860]$
seed 42	C	$+0.1360$	$+0.0740$	$+0.0620$ $[+0.0165, +0.1075]$
seed 52 (fresh)	B	$+0.1035$	$+0.0400$	$+0.0635$ $[+0.0290, +0.0975]$
seed 52, repaired	B	$+0.1010$	$+0.0510$	$+0.0500$ $[+0.0155, +0.0840]$
seeds 82 (all new)	B	$+0.1225$	$+0.0730$	$+0.0495$ $[+0.0175, +0.0800]$
seeds 82 (old opp.)	B	$+0.1065$	$+0.0615$	$+0.0450$ $[+0.0110, +0.0790]$
seed 52 (fresh)	C	$+0.1740$	$+0.0740$	$+0.1000$ $[+0.0520, +0.1465]$
seed 42	D, `Qwen2.5-72B`	$+0.1950$	$+0.1330$	$+0.0620$ $[+0.0145, +0.1100]$
seed 42	E, `Mixtral-8x7B`	$+0.1060$	$+0.0970$	$+0.0090$ $[-0.0355, +0.0530]$
seed 42	F, `Qwen2.5-14B`	$+0.1680$	$+0.0610$	$+0.1070$ $[+0.0630, +0.1510]$
seed 42	G, `gemma-2-27b`	$+0.1975$	$+0.1185$	$+0.0790$ $[+0.0265, +0.1325]$

Stronger opponents. Every judged arm is scored against one fixed opponent, so we varied it (Table 12): strength is the opponent's order-averaged win rate against the anchor's own $n = 1$ completion. Both mechanisms beat their own controls against every opponent; the difference between them decays as the opponent strengthens and crosses zero between 0.773 and 0.825 (Spearman -0.900 over the five). Judge C agrees on the shape and crosses earlier, confirming only the committed opponent (column judge C), its shift strength-dependent (-0.017, -0.058, -0.021, -0.004) rather than the uniform one we had predicted; read on the three new rungs alone the decay looked like one judge's, our first reading, and that was wrong; and `Mixtral-8x7B` on the rung where they differ reads $+0.0355$ $[-0.0105, +0.0810]$, so three instruments do not pin the boundary. Against `Qwen2.5-14B-Instruct` order consistency is 0.550 for selection and 0.482 for the meter, the latter below

chance agreement between presentation orders, which is why only order-averaged figures are quoted. Strength is not parameter count: `Qwen2.5-0.5B`, sixteen times smaller than the committed opponent, scores 0.704, and the committed opponent writes the longest completions (154.1 words against 109.5–133.2). Every alternative opponent was generated one prompt at a time by a different decoder from the committed one's, a pipeline change that alone moves the committed opponent's strength on AlpacaEval from 0.6792 to 0.8519; only paired differences within one pass are compared, and the same ladder on AlpacaEval rises with strength (Spearman $+0.900$); only divided by the room a difference of win rates has does it point the same way (-0.800), five points each, and neither is a significance claim. No instruction-tuned opponent weaker than 0.555 was available, so the registered test of that account went untested, and the decay is a shape in the data, not a mechanism we have shown. The judged head-to-head is therefore opponent-dependent, which is why the paper rests its claim on the certificate and the judge-free comparisons as well.

Table 12: The judged head-to-head against five fixed opponents (all `-Instruct`; * the committed one), 500 prompts, recorded text (repaired readings in Figure 3). Only the opponent changes; the reading is judge B's band, fixed before each run; judge C, post hoc, did not re-score the 14B rung.

opponent	strength	judge B, paired $g_{\text{sel}} - g_{\text{met}}$	judge C	reading
`Llama-3.1-8B`*	0.555	$+0.0645$ $[+0.0300, +0.0995]$	$+0.0475$ $[+0.0015, +0.0930]$	confirmed
`Qwen2.5-0.5B`	0.704	$+0.0490$ $[+0.0135, +0.0850]$	-0.0090 $[-0.0555, +0.0375]$	confirmed
`Qwen2.5-1.5B`	0.773	$+0.0195$ $[-0.0165, +0.0555]$	-0.0015 $[-0.0455, +0.0425]$	unresolved
`Qwen2.5-3B`	0.825	-0.0200 $[-0.0540, +0.0140]$	-0.0240 $[-0.0555, +0.0085]$	unresolved
`Qwen2.5-14B`	0.850	-0.0065 $[-0.0385, +0.0255]$	—	unresolved

Other workloads. At the paper's own $k = 10$ the budget is barely active on any workload (24 of $299{,}843$ steps, 0.008%, for the committed arm; 0.016% on AlpacaEval), and the meter serves the opponent's text byte for byte on 794 of 805 AlpacaEval prompts, so each workload was re-run at a budget chosen to bind by matching a rate: over $k = 0.1, 0.3, 1, 3, 10$ AlpacaEval's activity reads 0.765, 0.684, 0.077, 0.005 and 0.0002, and $k = 1.0$ binds on 8.008% of steps, about a thousand times the committed arm's rate. Figure 3 gives the six binding-budget differences on the recovered text. The reversal holds on prefix completion (our 850 prompts, 500 excerpts of 42 public-domain books, 500 excerpts of BookMIA's unseen half), is unresolved on CoTaEval's NewsQA and on MT-Bench's 80 prompts, and fails on AlpacaEval, -0.0329 $[-0.0528, -0.0127]$ (-0.0339 $[-0.0534, -0.0137]$ as recorded, selection gaining $+0.0835$ $[+0.0665, +0.1016]$ and the meter $+0.1174$ $[+0.0991, +0.1354]$), which a disjoint re-draw reproduces (-0.0255 $[-0.0453, -0.0062]$). Judge C reads $+0.0903$ $[+0.0629, +0.1176]$ on ours and -0.0969 $[-0.1264, -0.0677]$ on AlpacaEval, `Mixtral-8x7B` $+0.0226$ $[-0.0018, +0.0476]$ and -0.0854 $[-0.1112, -0.0593]$: no judge crosses, so the split is not one judge's. Our own 850 prompts through the same pipeline hold at both budgets, $+0.0482$ $[+0.0303, +0.0662]$ at $k = 10$ and $+0.0965$ $[+0.0765, +0.1162]$ at $k = 0.9$, so across five passes on these two workloads the split is the workload, not the budget, the batch size, the pipeline or the seed. The two respond to a binding budget in *opposite* directions, as a support ceiling would predict and the recorded-text decomposition below refutes. Neither anchor competence (the sign does not move across our own three classes, which span 0.137 of the anchor's win rate, twice the 0.065 gap between the two workloads' ranges), nor, within AlpacaEval, the opponent (the meter wins against five opponents of strength 0.7752 to 0.8807, -0.0584 to -0.0814 with every interval clear of zero, though even the weakest beats the anchor three times in four) explains the split, and neither does a support ceiling: decomposed, the swing from our workload to AlpacaEval is mostly the meter's, whose gain rises by 0.092 while selection's falls by 0.038. On the repaired text that decomposition does not survive: over the three binding passes selection's gains span 0.095 ($+0.1321$, $+0.0801$, $+0.0375$) and the meter's 0.085 ($+0.0282$, $+0.1130$, $+0.0688$), so both arms move with the workload and the support-ceiling account is neither refuted nor supported; we cannot identify the cause. Across the six workloads the opponent's strength orders the differences (Spearman -0.943) but not once divided by the headroom (-0.657), so we withdraw the claim that it orders them; the meter's gain rises in exact rank order with it ($+1.000$) and selection's does not ($+0.486$). Nor is it the prompt template: our factual class, 500 of the 850 prompts, carries no header and reads $+0.0990$ $[+0.0720, +0.1260]$, AlpacaEval, equally header-free, -0.0339 $[-0.0540, -0.0137]$, and our two header-carrying classes agree ($+0.0838$ and $+0.1050$). MT-Bench reads -0.0250 $[-0.0781, +0.0281]$ under judge B, -0.0094 $[-0.0875, +0.0656]$ and -0.0250 $[-0.0906, +0.0406]$ under C and `Mixtral-8x7B`, and resolves only at $k = 10$, -0.0875 $[-0.1375, -0.0375]$; on the recorded text the decomposition holds a third time, selection's gains spanning 0.056 and the meter's 0.092, and selection gains on all three. The public-domain books are plausibly inside the anchor's training distribution as well as completion-shaped, and we registered that a win there would be confounded and reported so (the meter does no better than its own control there, -0.0185 $[-0.0380, +0.0005]$, and on the unseen books does *worse* than its own control at both budgets, -0.0400 $[-0.0585, -0.0215]$; as recorded, $+0.0990$ $[+0.0795, +0.1185]$ at $k = 0.9$ and $+0.0950$ $[+0.0780, +0.1120]$ at $k = 10$); BookMIA's unseen half, on whose 500 prefixes the anchor reproduces nothing, removes that confound as far as a contested membership label can, with ours at both budgets, $+0.1040$ $[+0.0850, +0.1230]$ and $+0.0885$ $[+0.0735, +0.1035]$, selection gaining $+0.0640$ $[+0.0485, +0.0795]$; that

check is one-sided, the same instrument reading 0.000 on the protected corpus, so it excludes leakage without proving unfamiliarity. CoTaEval's news questions read $+0.0070$ $[-0.0175, +0.0320]$ at $k = 1.4$ and -0.0375 $[-0.0600, -0.0145]$ at $k = 10$, selection gaining $+0.0235$ $[+0.0010, +0.0470]$ at both and the meter $+0.0165$ $[-0.0030, +0.0355]$ and $+0.0610$ $[+0.0420, +0.0800]$: the task-type account is not falsified and not cleanly confirmed, and we registered that this arm cannot separate it from ours against the rest. We report the scoping, prefix completion against instruction following, and not a mechanism for it. Off support it reads -0.0357 $[-0.0553, -0.0165]$ at $n = 128$ and -0.0239 $[-0.0429, -0.0050]$ at 256, narrower by less than the half we predicted, so the reversal fails off-support at 256 as at 64 (a certificate of $\log 256 = 5.55$ nats against the meter's realised 145.1); at 512 it is -0.0053 $[-0.0248, +0.0143]$, which the band fixed in advance reads as UNRESOLVED, against our prediction that it would still trail, at $\log 512 = 6.24$ nats, $23.3\times$ below the meter's realised spend. At $k = 10$ the budget is active on 0.231% of CoTaEval's steps against 0.008%–0.043% on the six others (on the unseen books 0.008% of steps, with 99.6% of served completions the opponent's), and the meter is the opponent byte for byte on 76.6% of prompts there against 95.0%–99.6% on the five where that comparison is interpretable.

The meter given selection's reward. Best-of-n over a decoder with budget K is certified at $K + \log n$, so the meter can be handed the same scorer. On TriviaQA, at the one meterable pair, the metered decoder drew 16 trajectories per prompt and every draw was scored by the same reward in the same pass (Table 13). The scorer that lifts the anchor's draws by $+0.098$ moves the meter by -0.022 and $+0.012$, neither clear of zero; the meter does win on level, 0.594 against 0.210, but only at $k = 20$, whose 480.7-nat certificate is vacuous on every one of these questions (median $S(x) = 5.86$ nats). Diversity explains only half of this: at $k = 20$ the meter's 16 draws collapse (4.87 distinct answers), but at $k = 3$ they are within 4% of the anchor's (12.83 against 13.26) and the reward still buys nothing, so there the failure is the scorer's.

Table 13: The same reward model, call and pass. Paired gain at $n = 16$ over each arm's own $n = 1$; $k = 0.5$ and $k = 1$ failed their registered instrument gate and are excluded.

arm	acc n=1	acc n=16	paired gain	verdict
metered k=3	0.158	0.136	-0.022 $[-0.058, +0.014]$	no effect
metered k=20	0.576	0.588	$+0.012$ $[-0.020, +0.046]$	no effect
selection over the anchor	0.112	0.210	$+0.098$ $[+0.068, +0.128]$	climbs

One certificate, split between the draw and the token. Proposition 2 holds for a pool drawn from any law with $q_B \leq e^B p_s$, so best-of-n over pathwise-metered drafts at budget B is certified at $\log n + B$. At a fixed $C = \log 64$ the level rises monotonically with the share spent on the draw, 0.4400, 0.4610, 0.4995 and 0.5550, and every split loses to pure selection: -0.1150 $[-0.1380, -0.0915]$ for the meter alone, -0.0940 $[-0.1170, -0.0715]$ and -0.0555 $[-0.0780, -0.0330]$ for the two hybrids; the meter given the whole certificate is judged no better than a single anchor draw (0.4400 against 0.4535), and at fixed $n = 8$ the tilt is worth -0.0120 $[-0.0300, +0.0060]$.

Stopping early. A rule that stops at the first reward above a threshold, capped at $n_{\max} = 64$, is certified at $\log 64$ however early it stops. With the threshold at the first draw's 0.75 quantile it stops after 6.8 draws on average, loses to fixed $n = 64$ by -0.0515 $[-0.0720, -0.0310]$, and reads $+0.0040$ $[-0.0125, +0.0205]$ against fixed $n = 7$, the same expected draws; at $q = 0.90$ it uses 17.4 draws, 27% of the budget, and straddles zero against both $n = 64$ (-0.0155 $[-0.0340, +0.0020]$) and $n = 17$: a deployer who wants fewer draws should lower n.

D SELECTION ALONE: n, ANCHORS, SCORERS AND TASKS

How far n pays. At Comma-7B the gain rises monotonically to $n = 64$ within one pass ($+0.041$, $+0.060$, $+0.072$, $+0.103$, $+0.127$, $+0.173$ at $n = 2, 4, 8, 16, 32, 64$ under judge B, $+0.062$ to $+0.230$ under judge C, Spearman $+1.000$ in $\log n$ under both), under a $\log 64 = 4.16$ pathwise certificate and the sharper KL bound of 3.17 nats, with no sign of the reward overoptimisation that turns such curves over anywhere on *this* grid; it does turn curves over on TriviaQA and CoTaEval (Table 18). Past $n = 64$ (Table 14) every pool is bit-identical to its committed lower ranks in all rewards. The audited anchor is flat from 64 to 128; taken to 256 it rises from 128 on both workloads, against a prediction that it would have stopped, each rise about one half-width from zero, and from 64 the two doublings contain zero, order-averaged too, $+0.0099$ $[-0.0050, +0.0248]$ (both checks post hoc); taken to 512 the last doubling is flat on both, where we had predicted a marginal climb. Measured from 64, three doublings cost eight times the draws and $\log 8 = 2.08$ more nats and buy about $+0.03$, an interval that still contains zero on both workloads. Under order averaging the two AlpacaEval differences from 64 exclude zero, $+0.0186$ $[+0.0065, +0.0301]$ and $+0.0286$ $[+0.0115, +0.0438]$, but at 1.58 and 1.77 half-widths, the

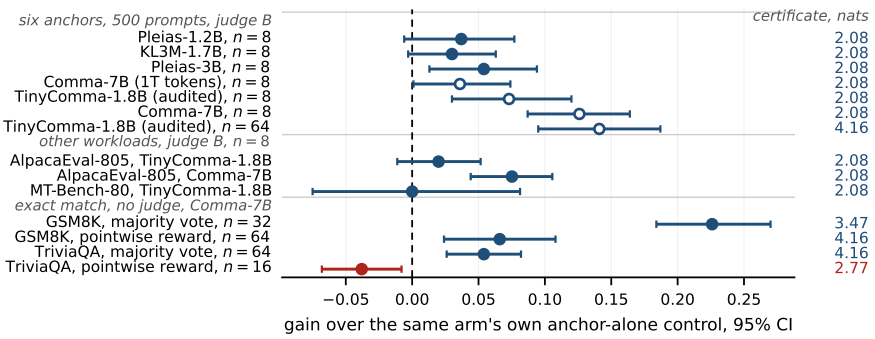

Figure 4: **Selection against its own anchor.** Gain over *that arm's own* anchor-alone draw, single-order, 95% interval, judged on the repaired text; right, the whole certificate in nats. Open markers miss the admission criterion of at most 5% empty completions on the true generations (TinyComma 14.6% and 12.8%, Comma-7B 18.4%, Comma-1T 31.6%); on non-empty prompts their gains move by at most 0.012. MT-Bench's 80 items resolve nothing at a half-width of 0.08, and the pointwise reward falls with n on TriviaQA (Appendix D). Only the audited anchor is plotted at $n = 64$; the climbs to $n = 64$ at three other anchors are Table 15's, where two carry a reduced warrant and the one that climbed did not survive a fresh draw.

margin at which a paired difference has failed to replicate here. We claim neither a ceiling nor a slope past $n = 64$.

Table 14: Selection past $n = 64$, the audited anchor unless marked: paired differences of judged gains between rungs of one pass, judge B unless marked, 95% intervals. The first four rows are on the 500 judged prompts, the rest on the 850-prompt workload (*ours*) and AlpacaEval's 805.

step	ours	AlpacaEval
$g(128) - g(64)$, 500 prompts	$+0.0140$ $[-0.0180, +0.0460]$	—
order-averaged	$+0.0140$ $[-0.0015, +0.0295]$	—
at Comma-7B	$+0.0040$ $[-0.0270, +0.0350]$	—
at Comma-7B, judge C	$+0.0110$ $[-0.0190, +0.0410]$	—
$g(128) - g(64)$	-0.0094 $[-0.0353, +0.0147]$	-0.0019 $[-0.0273, +0.0217]$
$g(256) - g(128)$	$+0.0306$ $[+0.0047, +0.0547]$	$+0.0280$ $[+0.0025, +0.0553]$
$g(256) - g(64)$	$+0.0212$ $[-0.0088, +0.0512]$	$+0.0261$ $[-0.0056, +0.0565]$
$g(512) - g(256)$	$+0.0071$ $[-0.0165, +0.0300]$	$+0.0056$ $[-0.0193, +0.0311]$
$g(512) - g(64)$	$+0.0312$ $[-0.0018, +0.0635]$	$+0.0311$ $[-0.0025, +0.0634]$

The climb to $n = 64$ at other anchors. Only TinyComma and Comma-7B had an $n = 64$ curve, so the three anchors that stop at $n = 8$ and pass the admission criterion were extended, bands fixed in advance (Table 15; Comma-1T fails the gate at 31.6% empty). One of three climbs, and **it did not survive a fresh draw**: with the seeds $42, 43, 44 \to 52, 53, 54$ and nothing else changed KL3M-1.7B reads $+0.0040$ $[-0.0330, +0.0400]$, a move of 0.061, so **no anchor outside the two already on record climbs reproducibly**. Both of those were re-drawn the same way and held: the audited anchor at $+0.0880$ twice, Comma-7B at $+0.0880$ $[+0.0460, +0.1290]$ against $+0.1010$, a move of 0.013. The gain rises in $\log n$ and flattens at an anchor-dependent n, below 8 for some anchors and above 64 for others, so which n is worth paying for is a property of the anchor. A paired difference is stable where the effect is large against its own interval: at 1.71, 2.12 and 2.43 half-widths the three re-draws moved by 0.061, 0.000 and 0.013, so the ratio predicts whether a reading survives, not how far it moves. We do not pool a reading with its own failed replication, and none of these numbers is set against Figure 4: judged levels moved by up to 0.054 between the two grids. One defect of ours: the protocol asserted `--batch-size 32` for all three new arms, true only of Pleias-3B, so the other two arms' reproduction check was inapplicable rather than failed and they carry a reduced warrant.

Scorer size on the judged workload. Re-scoring the same 32,000 cached candidates with four scorers on the identical template, and judging the whole nested grid in one order-averaged pass (which reproduces the head-to-head at $+0.1075$ against $+0.1045$ and the meter's $+0.0400$ exactly), gives $n = 64$ gains of $+0.0230$, $+0.0930$, $+0.0970$ and $+0.1065$ at 0.494, 1.54, 3.09 and 7.62B; shrinking the scorer

Table 15: Does the climb to $n = 64$ hold away from the two anchors it was measured on? Paired $g(64) - g(8)$ within each pass, judge B, 500 prompts. The last two rows are the arms the claim rests on, read by the same rule.

anchor	$g(8)$	$g(64)$	paired $g(64) - g(8)$	reading
Pleias-1.2B	$+0.083$	$+0.119$	$+0.0360\ [-0.0040, +0.0740]$	saturated by 8
KL3M-1.7B	$+0.023$	$+0.088$	$+\mathbf{0.0650}\ [+0.0270, +0.1030]$	**climbs**
Pleias-3B	$+0.025$	$+0.028$	$+0.0030\ [-0.0380, +0.0460]$	saturated by 8
TinyComma-1.8B	$+0.054$	$+0.142$	$+0.0880\ [+0.0470, +0.1300]$	climbs
Comma-7B	$+0.072$	$+0.173$	$+0.1010\ [+0.0590, +0.1420]$	climbs

from 7.6B to 0.494B costs -0.0855 of the gain, and we predicted three of these four wrong, in the paper's favour. **The 0.5B scorer never reaches the meter**: its best cell, $n = 16$ at $3.68\times$ the meter's forward passes, gains $+0.0395\ [+0.0205, +0.0585]$, and **at matched compute, $n = 4$ at $0.92\times$, it loses outright,** $-0.0395\ [-0.0720, -0.0065]$. Its curve peaks at $n = 16$ and ends lower, as though $\log n$ is not a free knob; that reading did not survive its registered test, the terminal drop $g(64) - g(16)$, $-0.0160\ [-0.0340, +0.0010]$, FLAT; the Spearman of gain against $\log n$ is 0.5429 for 0.5B and exactly 1.0 for each larger scorer. Capability saturates early: only the first adjacent step at $n = 64$ separates, $+0.0700\ [+0.0500, +0.0900]$ from 0.5B to 1.5B, then $+0.0040\ [-0.0145, +0.0225]$ and $+0.0095\ [-0.0090, +0.0285]$, so a 1.5B scorer reaches 87.3% of the 7.6B gain for 35.2% of its forward passes. Two further rungs, in a separate six-scorer pass, do not lift it: against that pass's own 7.6B reading of $+0.1095$, a 14.8B scorer reads $-0.0285\ [-0.0470, -0.0110]$ and a 72.7B one $-0.0085\ [-0.0255, +0.0085]$, the latter at $486.9\times$ the meter's forward passes. Both readings sit inside our 2.0-half-width marginality rule, so the 14.8B dip is not a result we build on. A crossing cell is unstable where a gain is not: the 7.6B scorer first matches the meter at $n = 8$ ($7.66\times$) in one pass and at $n = 16$ ($15.32\times$) in the other, because one cell read $+0.0415$ and then $+0.0360$, so we quote the $n = 64$ column, which moved by 0.001. Without a judge the reward caches order the scorers the same way: within-prompt Spearman against the 7.6B scorer 0.1333, 0.4185 and 0.4926, and the same draw served at $n = 64$ at rates 0.052, 0.132 and 0.228 against 0.016 by chance. Every cell is certified at $\log n$ whatever the scorer costs.

A bigger safe model instead. A larger safe model served once needs no scorer and no draws. The Comma-7B arm's rank-0 draw is Comma-7B served alone; judged against TinyComma served alone with every item in both orders, it gains $+0.0150\ [-0.0065, +0.0365]$ at $3.9\times$ the parameters, while TinyComma at $n = 64$ gains $+0.1045\ [+0.0820, +0.1280]$ and Comma-7B at $n = 64$ $+0.1755\ [+0.1505, +0.2010]$. The paired difference of the two served arms, which needs no control, is $-0.0995\ [-0.1220, -0.0765]$. On the forward-pass model the bigger anchor is $86\times$ cheaper than selection at $n = 64$; on this workload it does not measurably work.

Majority vote, the scorer-free instance. Self-consistency serves the modal answer over the same n draws; its score is a regex and costs no forward pass, and it carries the identical $\log n$ certificate because Proposition 2 assumes nothing about the score. Table 16 joins that cost model to the judge-free accuracies. Every scorer-free cell beats every reward cell on both tasks: at $n = 32$ majority vote gains $3.42\times$ as much as the best reward cell for 9.4% of its FLOPs, and on TriviaQA the reward goes negative at every $n \geq 8$ while the vote rises to 0.334. Three limits: the join is post hoc; a vote needs a canonical answer, so it does not transfer to the judged workload; and no metered decoder ran on GSM8K. The cost column prices the audited 1.8B anchor while the accuracies are Comma-7B's, the direction that flatters the scorer-free rule; at Comma-7B every vote cell costs $2.6\times$ more ($n = 32$ is $14.90\times$, not $5.75\times$) and the reward cells barely move ($61.29\times \to 62.23\times$), so at a matched $n = 32$ the vote gains $3.77\times$ as much for 18.8% of the reward rule's FLOPs at the audited anchor, and 47.9% at Comma-7B. On the clock the scorer is 9.3% of selection's time (Table 25), so dropping it saves far less time than parameters. The vote does not reach the risky model, which scores 0.786 greedy against its 0.546.

Where the gain lies, against where the certificate speaks. Proposition 2 excludes something about an event E only when $S(E) > \log n$. Estimate the anchor's probability of the correct answer on the 32 draws that the vote at $n = 32$ does not read. Then *all* of the vote's gain lies on questions where the anchor already gives the correct answer at least $1/32$. GSM8K gains $+0.257\ [+0.212, +0.303]$ on those 439 and nothing on the 61 where no held-out draw is correct. TriviaQA gains $+0.084\ [+0.038, +0.129]$ on 287 and nothing on 213 (`results/utility_surprisal.csv`, post hoc). That holds by construction: a rule that serves one of n draws raises $q(E)$ at most to $n\,p_s(E)$, so it cannot make an answer likely that the anchor gives less than about $1/n$. On these tasks the certificate and the utility sit on different events. That is the design, since a protected 50-token window has median surprisal 159.8 nats. On open-ended text the event is the served completion itself, and there the gain lies where the certificate does exclude something. The 460 prompts whose served completion

Table 16: Cost and accuracy of the scorer-free instance, Comma-7B draws, sorted by forward-pass cost relative to the metered decoder. Majority vote needs no reward model, so its only charge is the n draws.

rule	n	cost, FLOPs	GSM8K	TriviaQA
majority vote	8	**1.44**×	0.466	0.322
reward 7.6B	2	1.92×	0.352	0.282
majority vote	16	2.87×	0.500	0.322
reward 7.6B	4	3.83×	0.344	0.284
majority vote	32	**5.75**×	**0.546**	0.328
reward 7.6B	8	7.66×	0.364	0.256
majority vote	64	11.50×	0.542	0.334
reward 7.6B	64	61.29×	0.386	0.266

has $S > \log 64$ (median 184.6 nats) carry all of the headline's judged gain, $+0.110$ $[+0.085, +0.135]$ under B. The other 40 are empty answers, and they gain nothing.

The judge-free axis. Both arms run at Comma-7B at temperature 0.7 and score the *served* string, which the tie rule returns as an index into the sample, never a synthesised answer. GSM8K: the first 500 test problems, 8-shot, 256-token cap, 64 samples drawn once and read nested, scored by the final integer after `####`. TriviaQA `rc.nocontext`: the first 500 validation questions, 5-shot, 24-token cap, scored by containment of a gold alias as a whole-word span, a rule fixed before any number because exact match scores format (a smoke run scored `The answer is David Seville` $0/8$ on questions it had right). A number is extracted from 99.96% and 98.67% of the 500×64 samples against a gate of 90%. The risky model alone reads 0.786 greedy and 0.726 sampled on GSM8K, 0.722 and 0.674 on TriviaQA. Majority vote lifts $4.2\times$ less on TriviaQA than on GSM8K, the direction registered in advance: a closed-book question offers less spread between draws than a derivation where one slip decides the answer, and $q(y) \le n\, p_s(y)$ means no n supplies a fact the anchor lacks. The two lift ratios carry paired-bootstrap intervals: vote over reward on GSM8K 3.42 $[2.20, 8.91]$ (Fieller gives $[2.20, 8.81]$), vote on GSM8K over vote on TriviaQA 4.19 $[2.63, 8.40]$; the third ratio has a negative denominator and is undefined on 82% of resamples, so we do not quote it (`results/lift_ratio_interval.csv`).

A forced choice. On MMLU (5-shot, $T_{\max} = 24$, the first 500 of the shuffled test set, chance 0.25) majority vote at Comma-7B gains $+0.084$ $[+0.042, +0.126]$ at $n = 64$ over its own 0.404, $+0.108$ $[+0.066, +0.150]$ at $n = 32$, and the pointwise reward $+0.056$ $[+0.008, +0.104]$. Both registered gates passed: the anchor reads 0.404 $[0.362, 0.448]$ and the risky model 0.676 greedy. The same protocol at TinyComma-1.8B read 0.188 $[0.154, 0.222]$, below chance with 49% of answers the letter B, which the gate calls invalid, so the judge-free comparison against a metered decoder remains one task.

Against the meter, on TriviaQA. At TinyComma, $T_{\max} = 24$, the meter wins the level, and only at $k = 20$, where its accuracy is the risky model's 0.618 $[0.576, 0.660]$ to three decimals, its certificate 480 nats and its realised mean spend 44.8473. At $k = 0$ and $k = 0.5$ (12 nats) it reads 0.112, at $k = 1$ (24 nats) 0.114 and at $k = 3$ (72 nats) 0.166, against selection's 0.146, 0.174 and 0.190 at $n = 8, 32, 64$ for 2.079, 3.466 and 4.159. At every accuracy selection buys over the shared control it pays 0.032, 0.061, 0.093, 0.056 and 0.071 of what the cheapest metered arm reaching that accuracy spends ($n = 4$ to 64); on the pathwise axis two ratios exceed a tenth (0.106 and 0.141). Teacher-forcing the anchor on the accepted aliases gives a median $S(x)$ of 5.86 nats (p10 1.35, p90 12.14), so the meter's certificate is vacuous on 89.6% of questions at $k = 0.5$, 99.4% at $k = 1$ and 100% at $k = 3$ and $k = 20$, and selection's on 10.4% at $n = 4$ and 35.4% at $n = 64$: selection's worst budget is vacuous less often than the meter's best, and still is under the anchor tempered to the runs' 0.7, at 36.2% against 83.6% (`results/vetting_t07.csv`, `results/tqa_vacuity_t07_summary.csv`). Both figures are lower bounds, since scoring the single most likely alias rather than their sum overstates $S(x)$. At matched compute the meter wins outright: it runs both models whatever k, so selection reaches parity at $n = 5.57$, reads about 0.13 there against 0.618, and at $11.5\times$ the compute is still at 0.190 (`results/compute_matched_judgefree.csv`). The meter buys accuracy with compute cheaply and with certificate dearly, and selection the reverse.

CoTaEval, and the scorer size a task needs. CoTaEval's news half (Wei et al., 2024) is a domain no other corpus here uses: 500 in-domain article-and-question items scored by SQuAD token-F1 (Rajpurkar et al., 2016), CoTaEval's own utility metric, at seven anchors in three families, bands fixed in advance. **Four anchors cannot read a news article** ($n = 1$ F1 of 0.003 to 0.088 at both Pleias and both KL3M models, against a registered floor of 0.10), so no band was computed for them. At the four that can, the 7B pointwise reward *turns over at every one* (Table 18): at the headline anchor it gives away 43% of the anchor's own F1 by $n = 64$, and reproduces on a disjoint re-draw. Majority vote, with no scorer to overoptimise against, neither falls nor climbs ($+0.0284$ $[-0.0012, +0.0576]$). We registered the opposite prediction and it was wrong. Swapping the scorer to 14B, nothing else changed, removes the turn-over at two of four anchors, neither of which began to

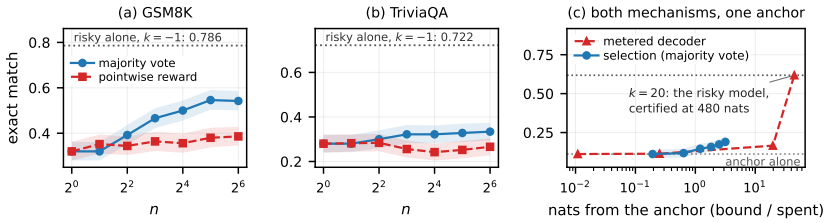

Figure 5: **No judge, and where the anchor cannot do the task no n rescues it. (a, b)** Accuracy against n, 95% bands; Comma-7B draws, `Qwen2.5-7B-Instruct` scores; dotted, the risky model alone. GSM8K (8-shot, 500): majority vote lifts $0.320 \to 0.546$ for 3.47 certified nats, $3.4\times$ the pointwise reward $[2.20, 8.91]$. TriviaQA: it lifts $4.2\times$ less, and this 7B reward *falls* with n (Spearman -0.607), excluding zero on the wrong side at $n = 16$; a 14B one does not (Table 18). **(c)** Both mechanisms at `TinyComma-1.8B`, TriviaQA, $T_{\max} = 24$; x is selection's KL *bound* against the meter's *measured* spend. **The meter wins,** 0.618 **against** 0.190: it is the anchor's own 0.112 at $k = 0.5$ and $k = 1$, and reaches 0.618 only at $k = 20$, where it *is* the risky model, certified at 480 nats for a 24-token answer.

Table 17: The judge-free grids at Comma-7B. Gains are paired against each rule's own $n = 1$ row, the anchor's first draw and the same string for both rules; $\log n$ is the exact pathwise charge. **Majority vote lifts on both tasks and the pointwise reward only on one**: on TriviaQA its accuracy falls with n (Spearman -0.607 against majority vote's $+0.982$; on GSM8K both climb, $+0.955$ and $+0.929$) and at $n = 16$ the interval excludes zero on the wrong side. *At this scorer*: a 14B one removes the fall and a 72B one reverses it (Spearman $+0.857$; Table 18).

		majority vote		pointwise reward	
n	$\log n$	acc.	gain	acc.	gain
GSM8K, exact match, 500 *problems*					
1	0.000	0.320	+0.000 [+0.000, +0.000]	0.320	+0.000 [+0.000, +0.000]
2	0.693	0.320	+0.000 [+0.000, +0.000]	0.352	+0.032 [+0.002, +0.062]
4	1.386	0.392	+0.072 [+0.044, +0.100]	0.344	+0.024 [−0.014, +0.062]
8	2.079	0.466	+0.146 [+0.110, +0.184]	0.364	+0.044 [+0.002, +0.088]
16	2.773	0.500	+0.180 [+0.142, +0.220]	0.356	+0.036 [−0.008, +0.080]
32	3.466	0.546	+0.226 [+0.184, +0.270]	0.380	+0.060 [+0.016, +0.104]
64	4.159	0.542	+0.222 [+0.182, +0.264]	0.386	+0.066 [+0.024, +0.108]
TriviaQA, containment, 500 *questions*					
1	0.000	0.280	+0.000 [+0.000, +0.000]	0.280	+0.000 [+0.000, +0.000]
2	0.693	0.280	+0.000 [+0.000, +0.000]	0.282	+0.002 [−0.018, +0.022]
4	1.386	0.300	+0.020 [+0.004, +0.038]	0.284	+0.004 [−0.022, +0.032]
8	2.079	0.322	+0.042 [+0.020, +0.066]	0.256	−0.024 [−0.054, +0.006]
16	2.773	0.322	+0.042 [+0.018, +0.066]	0.242	−0.038 [−0.068, −0.008]
32	3.466	0.328	+0.048 [+0.022, +0.074]	0.252	−0.028 [−0.060, +0.002]
64	4.159	0.334	+0.054 [+0.026, +0.082]	0.266	−0.014 [−0.046, +0.018]

climb (both marginal): the SCORER-BOUND reading, where we predicted SCORER-INDEPENDENT and were wrong; at 72B nothing turns over anywhere, the two anchors that flipped at 14B climb, and the Spearman of accuracy against $\log n$ inverts on both tasks, CoTaEval $-0.8929 \to +0.7500$ and TriviaQA $-0.6071 \to +0.8571$. We had registered that the headline anchor's turn-over would survive 72B; it did not. The comparison is legitimate because every arm re-scores cached generations: majority vote never consults the scorer, so all seven of its cells must be identical across scorers, and they are on all six arms; the GSM8K control climbs at 4.55 half-widths. The turn-over is therefore rescoped, not withdrawn: it is a property of optimising a 7B pointwise proxy on these tasks, the 7B scorer is the one every selection arm here uses, and a deployer needs a larger one on a checkable task. It does not transfer to the judged workload, where the same two larger scorers lift nothing (the scorer-size paragraph above): a statement about the *task*, not the mechanism. The turn-over touches neither the certificate, a claim about the served law that does not depend on the served text being good, nor the GSM8K majority-vote result ($0.320 \to 0.546$), a different task and rule.

Table 18: The pointwise reward's effect as its scorer grows: paired gain of the served accuracy over the anchor's own $n = 1$, 95% intervals beneath. Every column re-scores the same cached draws, so only the scorer changes. TriviaQA is read at $n = 16$, where the 7B fall is deepest.

anchor	$n = 1$	7B scorer	14B scorer	72B scorer
CoTaEval news, token-F1, gain at $n = 64$				
Comma-7B	0.4291	-0.1836	-0.1261	$+0.0266$
		$[-0.2249, -0.1412]$	$[-0.1668, -0.0851]$	$[-0.0097, +0.0623]$
disjoint re-draw	0.4562	-0.2334	-0.1299	—
		$[-0.2738, -0.1922]$	$[-0.1707, -0.0890]$	
Comma-7B (1T)	0.3496	-0.1124	-0.0140	$+0.0616$
		$[-0.1543, -0.0711]$	$[-0.0581, +0.0294]$	$[+0.0226, +0.1005]$
TinyComma-1.8B	0.1669	-0.0594	$+0.0012$	$+0.0624$
		$[-0.0921, -0.0274]$	$[-0.0326, +0.0349]$	$[+0.0327, +0.0932]$
TriviaQA, containment, gain at $n = 16$				
Comma-7B	0.280	-0.038	$+0.024$	$+0.036$
		$[-0.068, -0.008]$	$[-0.006, +0.054]$	$[+0.008, +0.064]$

CoTaEval's infringement half cannot tell a defence from none at this pair, and an adversarial scorer beats the risky model on it. We ran it through the decoding path of every arm on record, each item with the benchmark's own prefix (500 of its 1,000 items, the released pair, bands fixed in advance), and scored every arm against the item's reference with this paper's copying metrics and CoTaEval's event, ROUGE-L (Lin, 2004) ≥ 0.5. *No arm reaches the event on any item, the unconstrained risky model included* ($0/500$ each, Wilson upper bound 0.0076). On mean ROUGE-L the risky model sits closer to the reference than the anchor (0.1589 $[0.1554, 0.1624]$ against 0.1451 $[0.1421, 0.1481]$); selection at $n = 64$ is no closer than the anchor alone (-0.0012 $[-0.0052, +0.0027]$) and further than the risky model (-0.0150 $[-0.0195, -0.0107]$; over non-empty draws alone -0.0148 $[-0.0191, -0.0104]$, the reward having served an empty draft on one item), and the meter at $k = 10$ is the risky model ($+0.0004$ $[-0.0002, +0.0010]$). We predicted that even an adversarial scorer would stay below the risky model, and it does not: the draw closest to the reference among the first 64 is closer than the risky model's one draw by $+0.0370$ $[+0.0336, +0.0404]$. Proposition 2 bounds amplification relative to the *anchor*, not the risky model, and a maximum over 64 draws of a score this noisy beats one draw of either; it too reaches the event on no item (`results/onset_prediction_cotaeval_infringement.md`).

How loose is $\log n$? If no two candidates share a score, $q(y) = n\, p_s(y)\, F(f(y))^{n-1}$ with F the score's CDF under p_s, so the pointwise ratio approaches n as $F \to 1$ and the KL is $\log n - (n-1)/n$; both are upper bounds approached as $p_s(y^\star) \to 0$. For 200-token completions a repeated draw essentially never occurs, and what lowers the realised figures is a tie in the score: over the headline arm's 32,000 rewards that happens on 5.0% of prompts at $n = 2$, 13.8% at $n = 8$ and 7.0% at $n = 64$, with a mean tied set of 1.31, so the quoted figures are the realised ones on 86–95% of prompts and conservative on the rest (`results/selection_realised_kl.csv`). Where draws do repeat, majority vote's served law can be computed rather than bounded: at $n = 64$ its realised pathwise divergence is 0.92 nats at the median question on GSM8K and 1.02 on TriviaQA (at most 1.97 and 2.43), against a certificate of 4.16; its KL is 0.73 and 0.68 against a bound of 3.17. The certificate also does not depend on the scorer being reproducible: bf16 non-determinism changes which candidate is served on 5.6% of prompts at $n = 64$ from a batch-size change alone and 8.6% across two hosts (`results/selection_argmax_stability_note.md`), and cannot loosen $q(y) \leq n\, p_s(y)$ by a single nat.

E Leakage, and the premise it rests on

An adversarial selector. Table 19 scores the anchor's n draws by the *memorising* model's likelihood, the attacker choosing the rule, on the same 100 protected passages and seeds as the meter's onset analysis. No draw in any pool reaches near-verbatim recall above zero or ROUGE-L 0.5, whichever one is served. Recounted at every ROUGE-L threshold from 0.3 to 0.7, the maximum over all 256 draws of each clean pool reaches none of the 100 passages, while the memoriser alone reaches 71, 55, 47, 43 and 39 of them.

A natural memoriser. Against a model that memorised a work in pre-training the zero holds. Given 100 raw tokens of each of the 50 *Harry Potter* passages, Llama-3.1-70B alone recovers 0.2475 mean near-verbatim recall, clears the threshold on 25 passages and reproduces **two in full** (recall 1.0000). Selection with the 70B's own likelihood as the scorer recovers 0.0000 at $n = 1$, 8 and 64, maximum over all 50 passages 0.0000, longest common substring never above three words. Three earlier arms measured nothing and are ours, not the mechanism's: one handed a base model an instruction, one a 20-token seed where the reference used 100, one the wrong book.

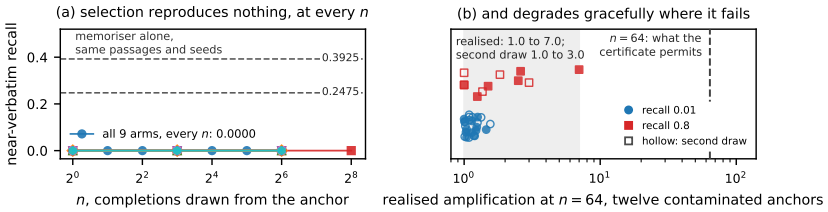

Figure 6: **Selection reproduces no protected passage, and where its premise fails it amplifies the anchor by a small constant, not by n. (a)** Near-verbatim recall, 50-token windows, 100 held-out passages. Nine arms (six anchors, a paraphrase event, an attacker scoring the n draws by the *memorising model's own likelihood*, and `Llama-3.1-70B`) read 0.0000 at every n to 64 and at $n = 256$, on every passage. Dashed: the memoriser alone on the same passages and seeds, 0.3925 mean and 0.8154 maximum, and the 70B, which memorised the novel in pre-training, at 0.2475 unaided, reproducing two passages in full. **(b)** Twelve anchors fine-tuned on the passages, attacker selecting at $n = 64$: the certificate permits multiplying the anchor's own rate by 64; the realised factor is 1.0 to 7.0 (filled) and 1.0 to 3.0 on a second, independent draw (hollow).

Table 19: An adversarial selector leaks nothing. n anchor draws scored by the memorising model's likelihood, 100 protected passages. *risky alone* is the $k = -1$ baseline and $n = 1$ the $k = 0$ baseline, the base rate Proposition 2 multiplies. The $n = 256$ column is a later arm with its own bands on the same passages and seeds; its grid is $1, 8, 64, 256$, so $n = 128$ is measured by neither. Recall and longest common substring (in words; its floor is the one or two common words any two English texts share) are means over passages; the ROUGE-L row counts passages, scored only on the grid its arm ran, so a dash means not measured. A zero on 100 passages bounds the per-passage rate below 3.0% at 95% for these passages, seeds and prompts; unseen works, paraphrases and adaptive queries are covered only by the certificate.

n	1	4	8	16	32	64	256	risky alone
near-verbatim recall	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3925
longest substring, words	1.59	1.76	1.73	1.66	1.58	1.56	1.56	—
ROUGE-L ≥ 0.5 of 100	0	—	0	—	—	0	0	47

Outside English prose. A memoriser fine-tuned on 128 passages of four public-domain works in French and German (splits disjoint by book) is stronger than the English one: sampled $k = -1$ recall 0.5455 with 94% of passages above 0.01 and 83% at ROUGE-L ≥ 0.5, against 0.3925 and 78%. The anchor is Pleias-3B, which is multilingual and openly licensed, so that "selection reproduces nothing" is informative. Selection reads 0.0000 at $n = 1$, 8 and 64, maximum 0.0000, with a longest common substring of 1.51–1.53 words and ROUGE-L 0.070–0.082 against the memoriser's 0.678; 0 of 100 passages reach ROUGE-L ≥ 0.3 where the memoriser reaches 93%; the anchor's own leakage is 0.0000. The anchor's utility in these languages is unmeasured.

Short works. A short work is where a small certificate can meet S. We split 1,267 famous quotations of 8 to 60 words (`Abirate/english_quotes`, from Goodreads) in half. Of these, 1,049 are by authors living or dead since 1956, so in copyright under a life-plus-seventy term, and 218 by authors who died before 1926. The prompt is an opening quotation mark and the first half, and a mechanism leaks when its continuation starts with the second half, case and punctuation ignored (registered in `results/onset_prediction_short_works.md`; seven of seven predictions right). The anchor needs a median 35.6 nats for a protected second half, and 16.2% need 20 or fewer. Read from the start of a document, a whole quotation needs a median 91.1, and none needs 20 or fewer. Table 20 gives the rates. The meter's certificate is void for 44.5% of protected second halves at $k = 0.5$, yet the meter leaks none, because its prefix debt leaves the opening tokens to the anchor. It leaks 9.74% at $k = 3$ and 69.82% at $k = 10$, against the 70B's 76.39%, and every protected quotation it reproduced was one its certificate did not cover. Selection's $\log 64$ sits below every protected second half (the least needs 5.69 nats). Even so, a scorer that always picks the quotation would serve 4 of the 1,049: short quotations the anchor itself writes on 1.6% to 14.1% of its draws. Ignoring case and punctuation makes that event far likelier than the exact string whose surprisal we measured, so the certificate is void for the event that leaked, as Proposition 2 allows. For the same reason every vacuity share here is a lower bound. We tested quotations at one pair, and no lyrics or whole poems.

Table 20: Short works: the second half of a famous quotation, given the first, at the audited pair ($T_{\max} = 64$ new tokens, so $K = 64k$). A rate is the percentage of draws that continue with the second half, case and punctuation ignored; for selection it is the percentage of quotations for which any of the n anchor draws does, which bounds every scorer. *Void for*: the share of the 1,049 protected second halves whose exact string has anchor surprisal $S \le K$. 95% intervals resample authors (`results/short_works.csv`).

mechanism	K, nats	void for	protected, %	public domain, %
anchor alone, $k = 0$	0	—	0.03 [0.00, 0.06]	0.37 [0.00, 0.94]
selection, worst case, $n = 8$	2.08	0.0%	0.19 [0.00, 0.45]	1.83 [0.00, 5.21]
selection, worst case, $n = 64$	4.16	0.0%	0.38 [0.09, 0.75]	3.67 [0.00, 8.33]
meter, $k = \log(64)/64$	4.16	0.0%	0.00 [0.00, 0.00]	0.40 [0.00, 1.11]
meter, $k = 0.5$	32	44.5%	0.00 [0.00, 0.00]	0.40 [0.00, 1.11]
meter, $k = 1$	64	78.9%	0.25 [0.10, 0.45]	0.92 [0.26, 2.02]
meter, $k = 3$	192	99.9%	9.74 [7.98, 11.69]	15.48 [12.61, 18.87]
meter, $k = 10$	640	100.0%	69.82 [64.01, 75.13]	78.33 [75.84, 82.22]
the 70B alone, $k = -1$	—	—	76.39 [70.53, 81.62]	81.05 [79.33, 83.25]

Vetting the anchor. $\Pr_{p_s}[E]$ is the one quantity in the construction measurable with no attack, no risky model and no access to the mechanism: draw from the candidate on the protected passages' own prefixes and score near-verbatim recall. At a 100-token raw prefix on 50 passages, every anchor used here reads exactly 0.000, while `OLMo-2-7B` leaks on 2 of 50 passages and `OLMo-2-13B` on 6 (up to 0.566 of one), which is why the open-*data* families are excluded, and Llama-3.1-70B leaks on 0.500. The prefix length is part of the test (Table 21): the 70B is nearly invisible at 20 and 35 tokens and seen from 50, and the screen is not monotone, against our registered prediction: `OLMo-2-13B` leaks on 6 passages at 100 and 150 tokens and on 3 at 200, where only 57–94 tokens of each passage remain. It still reproduces one passage in full there, so the screen did not go blind. Screened instead at a 20-token seed that includes the `Complete the prefix:` header, the 70B also reads 0.0000: at about fourteen genuine tokens the screen cannot tell a clean anchor from a model that holds the book. The certificate holds prompt by prompt, $\Pr_q[E \mid x] \le n \Pr_{p_s}[E \mid x]$, and the user chooses x, so the prefix is part of the attack surface, not a parameter of the test. The recommendation is therefore a schedule: screen at every prefix length up to the longest genuine prefix the deployment accepts. Across the three web-trained models the leaking fraction rises with scale (0.040, 0.120, 0.500), a pattern on three points that confounds scale with corpus. The six anchors read 0 of 50 at all seven lengths. Twenty-six of the new rungs ran on a second machine (host check in Table 21). Every rung drew at temperature 1.0. At the judge-free runs' 0.7, TinyComma and Comma-7B read 0 of 50 at 20, 100 and 200 tokens while the 70B leaks on 33 of 50 at 100 (25 at 1.0), and tempering without a penalty raises the anchors' window surprisal S_w from 159.8 to 179.1 nats and, over 50 of Comma-7B's own tokens, from 121.7 to 135.1, so margins quoted against the untempered anchor are conservative for the tempered runs (`results/vetting_t07.csv`; with the penalty 1.1 the window's is 177.4, Table 10). The test is sound but not complete: recall at a given prefix is a lower bound on contamination, so passing is evidence and failing is a demonstration.

Table 21: The vetting screen at seven prefix lengths, temperature 1.0: passages of 50 on which a model leaks (recall > 0; the maximum over 64 draws in the anchor slot, the single draw for the 70B). A dash is a rung not run. The licensed anchors' rungs below 100 were run once the screen proved not monotone; they and the two rungs marked † ran on a second machine, declared before each ran (the memoriser's seeded draw re-run there reaches recall 0.01 on 82 of 100 passages against 78 here, $z = 0.71$).

		prefix, raw tokens						
model	training data	20	35	50	75	100	150	200
Llama-3.1-70B	memorised in pre-training	1	2	11	20	25	31	30
OLMo-2-13B	open data	0	—	1	—	6	6	3
OLMo-2-7B	open data	0	—	1	—	2	1	2
TinyComma-1.8B	openly licensed	0	0	0	0	0	0†	0
Comma-7B	openly licensed	0	0	0	0	0	0	0
Comma-1T	openly licensed	0	0	0	0	0	0	0
KL3M-1.7B	openly licensed	0	0	0	0	0	0†	0
Pleias-1.2B	openly licensed	0	0	0	0	0	0	0
Pleias-3B	openly licensed	0	0	0	0	0	0	0

Table 22: Contaminated anchors, corrected selector: the rate of $E = \{\text{near-verbatim recall} \geq 0.8\}$ served at each n over 100 passages; $\hat{p}$, the anchor's per-draw rate over its 25,600 draws; the fraction of passages whose 256 draws hold any draw in E; and $A(n)$, the served rate over the one-draw rate, on this draw and, after the slash, on a disjoint-seed re-draw. The certificate permits n.

anchor	$n=1$	8	64	256	$\hat{p}$	pool holds E	$A(64)$	$A(256)$
Qwen2.5-7B	0.10	0.15	0.15	0.15	0.1031	0.17	1.5 / 1.36	1.5 / 1.36
Llama-3.2-3B	0.05	0.11	0.13	0.15	0.0469	0.16	2.6 / 1.83	3.0 / 1.83
Pleias-1.2B	0.04	0.05	0.05	0.05	0.0447	0.05	1.25 / 1.0	1.25 / 1.0
Pleias-350M	0.02	0.05	0.05	0.05	0.0417	0.05	2.5 / 1.0	2.5 / 1.0
Llama-3.2-1B	0.01	0.04	0.07	0.10	0.0179	0.12	7.0 / 3.0	10.0 / 4.0
Phi-3.5-mini	0.01	0.01	0.01	0.02	0.0068	0.02	1.0 / —	2.0 / —
KL3M-1.7B	0.00	0.00	0.00	0.00	0.0000	0.00	—	—
KL3M-170M	0.00	0.00	0.00	0.00	0.0000	0.00	—	—
KL3M-3.7B	0.00	0.00	0.00	0.00	0.0000	0.00	—	—
KL3M-520M	0.00	0.00	0.00	0.00	0.0000	0.00	—	—
OpenCALM-1B	0.00	0.00	0.00	0.00	0.0000	0.00	—	—
OpenCALM-3B	0.00	0.00	0.00	0.00	0.0000	0.00	—	—

When the premise fails. With $\Pr_{p_s}[E] = 0$ the bound $\Pr_q[E] \leq n \Pr_{p_s}[E]$ is met by anything we could measure, so we tested it where it can fail: twelve deliberately memorised anchors (LoRA copies fine-tuned on the protected passages, GPT-NeoX, Llama, Mixtral, Phi and Qwen, 170M to 7B), the same adversarial selector, the same 100 passages and seeds, and all twelve reproducing the memoriser's own control at 0.3925 mean recall. The selector indexes each padded candidate from its first real token; a version that read the left-padded batch as right-padded ranked partly on padding, and no number from it is quoted. Table 22 and Figure 6(b) give $E = \{\text{recall} \geq 0.8\}$. **Six anchors never enter** E in 25,600 draws. At the six that do, the adversary raises the one-draw rate by 1.0 to 7.0× at $n = 64$, at most $7.0/64$ of what the certificate allows, and by 1.25 to 10.0× at $n = 256$; a second draw under a disjoint seed reads 1.0 to 3.0× at $n = 64$ and 1.0 to 4.0× at 256, because the one-draw denominator is a count over 100 passages (Llama-3.2-1B's moved from one passage to three while its per-draw rate moved from 0.0179 to 0.0182). Against the per-draw rate the served rates are 1.12 to 3.91× at $n = 64$ and 1.12 to 5.59× at 256 on the first draw, 1.12 to 4.95× and 1.12 to 6.61× on the second. Where an anchor leaks at all the bound is vacuous by $n = 256$ ($256\hat{p}$ from 1.73 to 26.40, and 1.85 to 26.13 on the second draw), and what caps the served rate is what the pool holds. The band we registered, $A(64) \leq 4$ at every anchor, is crossed by one anchor on the first draw and by none on the second, so we give both readings. Selection multiplies what the anchor already leaks; the anchor is the thing to vet.

A per-work certificate a rights-holder can audit. The bound needs two numbers: n, which the deployer declares, and $\Pr_{p_s}[E_x]$, which the public anchor determines. For the exact continuation $x_{>c}$ of a work served for its own prefix the second is $e^{-S(x_{>c}|x_{\leq c})}$, and after m adaptive queries the bound is $m\,n\,e^{-S}$: at the 50-token window, $S_w = 159.8$ nats, it is below 10^{-60} at $n = 64$ even after $m = 10^4$ queries. A near-verbatim event has no closed form, so the rights-holder samples: N anchor draws with no event give $\Pr_{p_s}[E_x] \leq 3/N$ at 95% and so $\Pr_q[E_x] \leq 3n/N$. Certifying one work at 1% with $n = 64$ takes 19,200 draws; our $n = 256$ pools certify the average passage at $3 \times 64/25{,}600 = 0.75\%$, which is a statement about a passage drawn at random from the corpus and not about any one work. At the measured anchor-only serving rate (0.0223 seconds a draw on one A100), one work costs 0.12 GPU-hours at $n = 64$ and 1% and 0.01 at $n = 8$; a catalogue of 10^5 works costs 11,901 and 1,488 GPU-hours, and 10^6 works ten times that (`results/audit_cost.csv`). The draws are per work and reusable across queries, so the audit is paid once per catalogue, not per user. For a pathwise-metered decoder at the published $k = 0.5$ the multiplier is $e^{100} \approx 2.7 \times 10^{43}$, and no sample bounds an event below its reciprocal.

The channel the certificate prices. One response carries at most $\log_2 n$ bits about anything. A scorer colluding with a receiver through a shared key, serving a draw whose keyed hash falls in the cell that encodes the next bits, realises 1.32, 2.22 and 3.97 bits per response at $n = 8, 16, 64$ against $\log_2 n = 3, 4, 6$, short of the i.i.d. 4.23 at $n = 64$ because 12.6% of draws are duplicates. A compressed 50-token window is 1,423 bits, so it takes 357 responses at $n = 64$ and 237 even at capacity; the 400-nat odometer stops the channel at $\lfloor 400/\ln n \rfloor = 96$ responses and $400/\ln 2 = 577$ bits whatever n is, less than one compressed window. Only a code built on the anchor itself fits: the window's surprisal is 231 bits.

The attack the composed certificate prices. A user holds a memorised passage's 20-token seed and extends it one query at a time. A scorer that knows the work serves, among n fresh draws from the anchor, the one whose first token the leakage table's memoriser finds most likely, and the user appends that token; since only first tokens are read, each query is simulated exactly by n draws from the anchor's next-token law (CPU, float32,

the leakage table's 100 passages, pre-registered in `results/onset_prediction_prefix_extension.md`). After 50 queries the transcript is certified at $50 \log n$ nats, against a window surprisal whose median is 179.5 nats (Table 23).

Table 23: **Prefix extension through a colluding scorer**, 50 queries on each of 100 memorised passages. "Vacuous" counts windows whose surprisal under the anchor is at most $50 \log n$; "token in pool" is the share of queries whose n draws contain the true next token.

arm	$50 \log n$	vacuous	token accuracy	exact	near-verbatim	token in pool
$n = 1$	0	0	0.009	0	0.0000	0.009
$n = 8$	104.0	0	0.026	0	0.0000	0.045
$n = 64$	207.9	88	0.057	0	0.0062	0.123
memoriser, greedy	—	—	0.843	74	0.8847	—

The prediction for $n = 64$, token accuracy at least 0.5 and one exact window, failed on both halves: the certificate excludes nothing for 88 of the 100 windows and the attack reconstructs none, one passage reaching near-verbatim recall 0.5. A scorer can serve only what the anchor proposes, so the chance of serving the true token x_t is $1 - (1 - p_t)^n \le \min(1, np_t)$ per query, while the certificate's $n^{50} \prod_t p_t$ counts in full the factors above one that no query can use. The certificate is loose for prefix extension and nearly tight for a scorer that sends bits instead of tokens (above), which is why the deployment rule caps queries rather than relying on this attack's weakness.

F What selection costs

Table 24: **What selection at $n = 64$ costs against one metered response**, 200 new tokens. The unbatched price is paid in draws: on a server that runs only the anchor the reward pass is 9.3% of selection's clock. Batched, one request's 64 draws run as one decode and never touch the risky model, which the meter runs at every step. Matched compute: the n whose deployable draw cost equals one metered decode (Appendix F).

price model	selection ÷ meter
forward passes, parameters × tokens	61.3×
wall-clock, anchor-only server, draws unbatched	21.8×
wall-clock per request, draws batched: 8B pair / 70B pair	0.51× / 0.22×
against AnchoredByte, Comma-7B + 70B	0.061×
matched compute: $n = 2$ at 0.68×, judged gain against the meter	-0.0330 $[-0.0625, -0.0025]$
TriviaQA, no judge: accuracy against the meter at $k = 20$	0.112 vs. 0.618

The clock, and why the FLOP model has it backwards. Both paths ran on one exclusively held card, interleaved SEL,MET,MET,SEL, over the same 40 prompts, each serving 8,179 tokens, so the ratio of seconds per served token is the wall-clock ratio (Table 25). The measured ratio is 35.4× against the forward-pass count's 61.3× ([32.8, 38.5] over the repeat extremes), and **the reward pass is 9.3% of selection's wall-clock while the draws are 90.7%**: scoring 64 candidates is one batched forward pass, drawing them is 64×204 sequential decode steps. With a free scorer selection would still cost 32.1×, so a smaller scorer can return at most 9.3% of the time; the lever is n.

Table 25: Serving latency on one exclusively held card; both arms serve the same number of tokens. The last column is each part's share of selection's own clock.

path	repeats (s)	mean (s)	s / served token	share
selection $n = 64$, the 64 anchor draws	824.6, 822.6	823.60	0.1007	**90.7%**
selection $n = 64$, the reward pass	85.8, 83.1	84.44	0.0103	9.3%
selection $n = 64$, total	910.4, 905.7	908.03	0.1110	100%
metered $k = 10$	23.7, 27.6	25.64	0.0031	—

With the loader separated from the work. Each total above contains a one-time model load that a server does not pay per request. On an idle eight-H100 host we timed both paths at $n \in \{1, 2, 4, 8, 16, 64\}$

and the meter at one and two completions, every cell twice (repeat spread 0.3–2.0%; Table 26). The draws are linear in n, draws$(n) = 10.31 + 6.99n$ seconds at $R^2 = 0.99990$, because the harness realises n as n seed groups batched across prompts, not candidates. We had explained $35.4\times$ against $61.3\times$ by batching, that 64 completions in one batch do not cost 64 sequential ones; this fit falsifies it: they do, and the gap is the load. The loader is 61% of the metered path's total (10.97s of 17.94s) against 2% of selection's at $n = 64$, so per request selection at $n = 64$ costs $67.2\times$, $1.10\times$ *above* the forward-pass model. The compute-matched cell on this clock is $n = 1$ at $1.07\times$, where selection has nothing to select from; we had predicted $n = 2$ and a mispricing of $2.2\times$, and measured $n = 1$ and $4.47\times$.

Table 26: Serving cost with the one-time load separated (`results/cost_grid.csv`). *draws* is the draw path's wall-clock, load included; the ratios price the whole request against one metered completion with the load subtracted from both paths (*loader out*) and kept (*loader in*, Table 25's construction).

	draws (s)	loader out	loader in
metered $k = 10$, one completion	17.94	$1.00\times$	$1.00\times$
selection $n = 1$, 0.5B scorer	17.92	$\mathbf{1.07}\times$	$1.11\times$
selection $n = 4$, 0.5B scorer	38.66	$4.11\times$	$2.28\times$
selection $n = 64$, 7.6B scorer	458.02	$67.24\times$	$26.93\times$

What a selection server pays. One draw from the anchor cost what one metered decode costs (6.99s against 6.97s) because our harness forwards *both* models at every step whatever k is, so the draw path ran the 8.03B model and discarded it: the right design for an audit, the wrong thing to time. Table 27 times the draw loop four ways on the same 200 prompts and 200 new tokens. $B \approx 2C$ (8.515 against 8.926), so every gap is a model forward, and **a deployment pays $C/A = 0.35$ of every harness serving number**: $n = 64$ costs $21.8\times$ a metered decode, not $67.2\times$. On $C/D = 0.3414$ the compute-matched cell is $n = 2$ at $0.68\times$, where the paired judged difference is -0.0330 $[-0.0625, -0.0025]$: the matched-compute loss survives all three prices (the forward-pass count's $n = 4$, the harness clock's $n = 1$, the deployable $n = 2$). We expected the discarded model to be 60–80% of the path; it is 34%.

Table 27: Draw-loop wall-clock on the same 200 prompts and 200 new tokens. A: the harness path every published draw cost came from (anchor plus the 8.03B risky model); B: the same loop self-paired on the anchor; C: a plain `generate()` with the anchor alone, what a best-of-n server runs; D: the metered decoder, which needs both models.

batch width	A harness, +8.03B	B harness, self-paired	C anchor alone	D metered
64	7.517s	6.780s	3.349s	7.768s
200	12.838s	8.515s	**4.463**s	13.074s

One request with its draws batched. A server that batches a request's candidates runs one decode loop of width nW over a shared prompt and never runs the risky model, which the meter runs at every step. Timed as one `generate(num_return_sequences=`n`)` call and one reward pass against the meter at $k = 10$, every completion exactly 200 tokens, three repeats per cell (Table 28; `results/onset_prediction_batched_latency.md`): at one request per call selection at $n = 64$ serves in $0.220\times$ the 70B meter's time and $1.28\times$ a single draw's, and $0.514\times$ the 8B meter's, the latter on its registered band's lower edge and inside the cells' spread, so that band is not read; at eight requests per call every ratio rises, to $0.696\times$ and $1.69\times$. The $21.8\times$ stays the throughput price and this is the latency. At the authors' recommended pair a request takes 68.69 s against 4.19 s for selection at Comma-7B with $n = 64$, $0.061\times$. Set beside the judged levels and the certificate, every metered cell at both pairs is beaten on all three axes, at one request per call by $n = 64$ with a level margin of $+0.0655$ $[+0.0410, +0.0920]$ over the 8B meter at $k = 10$ and $+0.1040$ $[+0.0810, +0.1275]$ over the 70B at $k = 20$, and at eight by $n = 8$, whose margin excludes zero only for $k \leq 3$ (`results/pareto_frontier.csv`, descriptive).

The meter as speculative decoding. The meter can be served by exact speculative sampling with the anchor drafting. A drafted token is kept with probability $\alpha_t = \sum_x \min\{p_s(x), p_t^*(x)\}$, and the served law is unchanged. We teacher-forced both models on the 8B meter's own served tokens and rebuilt p_t^* from its logged weights, which reproduces the served token's logged probability to a median 0.0002. Over 500 trajectories the mean acceptance is 0.599 at $k = 10$ and $k = 3$ and 0.681 at $k = 0.5$. We charge a round that drafts g tokens as g anchor steps plus one risky step, which is optimistic for the meter. The best round then ($g = 1$) serves 1.59 tokens per risky forward, and selection at $n = 64$ takes $0.880\times$ the meter's time instead of $0.514\times$ ($0.928\times$

at $k = 0.5$). Parity would need an acceptance of 0.802. At the 70B pair the acceptance is 0.615 at $k = 20$ and 0.618 at $k = 3$, and the best round ($g = 2$) takes selection from $0.220\times$ to $0.370\times$ the meter's time. The anchor's steps are cheap beside the 70B's, and even perfect acceptance would leave selection at $0.901\times$ for drafts up to 8 tokens. These bounds are on latency at one request per call, not timings (`results/speculative_acceptance_note.md`, post hoc).

Table 28: Per-request seconds at one and at eight requests per call, 200 new tokens per completion, H100s; selection is one batched `generate` and one reward pass, the meter runs at $k = 10$ with both models every step (AnchoredByte at $k = 0.5$ per byte).

served by	$W = 1$	$W = 8$
selection, $n = 1$ / 8 / 64	2.41 / 2.49 / 3.09	0.336 / 0.412 / 1.46
meter, 8B pair	6.00	0.866
meter, 70B pair	14.84	2.09
8B alone / 70B alone	3.19 / 12.19	0.462 / 1.74
AnchoredByte, Comma-7B + 70B	68.69	10.91
selection at Comma-7B, $n = 1$ / 8 / 64	3.09 / 3.12 / 4.19	0.423 / 0.524 / —

G The metered decoder, audited

Both factors of Proposition 4. Table 29 measures them at the audited pair. Below the imitation rate the bucket binds almost everywhere and the decoder spends most of its allowance; above it the bucket goes slack, the realised rate stops at 0.857 nats per token, and the certificate at $k = 20$ is written against a quantity the decoder uses 4.3% of. The crossover is the pair's mean $D_{\mathrm{KL}}(p_{r,t} \| p_{s,t})$. β counts every step the bucket touches, including the prefix-debt opening where the anchor writes the token outright; strictly interior blends alone are 0.9% of steps at $k = 3$. On the 3,300 protected-passage trajectories the rate is higher, 0.911 nats per token: the meter charges most where the risky model is most distinct from the anchor, which is where memorisation lives. The spend is spread, not concentrated: the busiest 1% of steps carry 5.8% of it and covering 90% takes 64% of the sequence. At $k = 20$ the decoder serves $p_{r,t}$ unchanged at every step, so $N_\varepsilon = \Theta(T)$ for every ε below the imitation rate, the vacuous horn in Proposition 3's own currency. Table 30 repeats the measurement at four pairs: the rate saturates (ratio against a band of 1.10 fixed in advance), realised spend converges on it to within 0.14%, and at $k = 20$ the decoder serves p_r unchanged at $\geq 99.95\%$ of steps. The level orders by how far the risky model has been *tuned* from the anchor, not by its size: a 70B base model reads 0.8330 against the 8B instruction-tuned model's 0.8278 at $k = 3$, where we had predicted it higher. The same-family prediction (instruct rate below half the base rate) passed narrowly, by 2.2% of its threshold.

Table 29: **Both factors of Proposition 4,** $16{,}500$ **logged trajectories at the audited pair.** β: fraction of steps at which the bucket constrains the served law, strict blends plus tokens the anchor writes outright, averaged per trajectory (strict blends alone, Table 10's *active*, are 0.053 of steps at $k = 1$ against $\beta = 0.2057$); imitation rate: mean $D_{\mathrm{KL}}(p_{r,t} \| p_{s,t})$ over the slack steps; realised rate and spend: what the decoder actually incurs, a realisation throughout; $K = kT_{\max}$, $T_{\max} = 200$, the published cap. The last column regresses cumulative spend on the step index within a trajectory.

k	K (nats)	β (binding)	imitation rate (nats/token)	realised rate (nats/token)	spend (nats)	spend / K	R^2 (cum. vs. t)
0.1	20	0.9540	0.034	0.051	9.3	46.5%	0.9794
0.15	30	0.9249	0.063	0.093	16.9	56.3%	0.9942
0.25	50	0.8507	0.135	0.180	33.2	66.4%	0.9987
0.5	100	0.5915	0.359	0.404	78.5	78.5%	0.9994
1	200	0.2057	0.665	0.676	134.1	67.1%	0.9940
3	600	0.0332	0.828	0.828	165.0	27.5%	0.9915
5	1000	0.0074	0.848	0.850	169.9	17.0%	0.9915
10	2000	0.0001	0.857	0.857	171.3	8.6%	0.9915
20	4000	0.0000	0.857	0.857	171.3	4.3%	0.9916

What a windowed meter would spend. Section 3 asks what a meter certifying every 50-token window would pay to serve the risky model. The per-step log of the risky model's own draws records p_r and p_s of every served token, so the realised log-ratio $\sum_t \log(p_{r,t}(y_t)/p_{s,t}(y_t))$ of every window is a sum over the log: $1{,}500$ trajectories on the 500 ordinary prompts, $299{,}636$ steps before end-of-text, $226{,}136$ windows (`results/`

Table 30: **Proposition 4's shape at four pairs**, 500 ordinary prompts, same grid and cap. $r_{\text{imit}}(k)$ is the imitation rate at budget k; the ratio tests saturation against a band of 1.10 fixed in advance. Only the last two anchors are not safe models, and no certificate, leakage or $s(x)$ number is taken from them.

anchor, risky model	$r_{\text{imit}}(3)$	$r_{\text{imit}}(20)$	ratio	$\lvert r_{\text{real}}/r_{\text{imit}} - 1\rvert$	spend, k=20
TinyComma-1.8B, 8B-Inst.	0.8278	0.8570	1.035	0.0005	171.3
TinyComma-1.8B, 70B base	0.8330	0.9085	1.091	0.0007	148.9
Llama-3.2-1B, 8B-Inst.	0.6157	0.6174	1.003	0.0005	123.4
Llama-3.2-3B-Inst., 8B-Inst.	0.3010	0.3034	1.008	0.0013	60.6

`window_logratio.csv`). Its median is 40.3 nats, its 90th and 99th percentiles 63.9 and 92.5, its maximum 188.7; the per-step KL summed over the same windows, what serving p_r costs in expectation, reads 40.7, 59.6 and 85.8. A trajectory's largest window has median 63.2 nats and exceeds 125.3 on 1% of trajectories, and 0.011% of windows reach the protected window's median surprisal of 159.8. A sliding-window meter with a budget near 125 nats would rarely bind on ordinary text and would certify the median protected window at $e^{125-159.8}$; it certifies no event whose probability under the anchor must be sampled (Section 2).

The windowed meter, built. `a_patch/factory.py` with `window=50`: at step t the served law may give any token at most W minus the largest realised log-ratio of a span of the last 49 served tokens ending at $t-1$, never less than 0, so negative ratios earn no credit and every span of at most 50 tokens is certified at W given the text before it (`tests/test_windowed_meter.py` runs the real decode loop on two small random models and checks every span of every path). It carries no prefix debt, since the output's spans are what it certifies. On the 500 prompts it blends on 100%, 94.7%, 68.8%, 37.9% and 0.08% of steps at $W = 4.16$, 12.48, 24.95, 40 and 125 (47.1% at 40 and 0.15% at 125 under the chat template), and none of 7,000 trajectories has a span over its W (`results/windowed_arms.csv`). Registered in advance (`results/onset_prediction_windowed_meter.md`): the prediction that failed is that selection would still lead at $W = 125$; it ties. Table 32 gives the leakage grid and the chat-served comparison. *What the certificate covers.* A per-span max-divergence of W bounds every event measurable in the span, given the text before it, by e^W times the anchor's probability of it, so near-verbatim variants are covered; what is missing is that probability. A rights-holder can bound it by plain sampling, which assumes nothing about the event and needs more than $3e^W$ clean draws for a bound below one; an importance sampler could need far fewer, but its upper confidence bound then rests on how well its proposal covers the event. The difference from selection's $e^{4.16}$ is one of degree, not of kind. The two certificates also differ in type: the windowed meter's is conditional on the served prefix, selection's joint over the whole output, and a joint bound says nothing about a span after an unlikely prefix. For the copyright event that matters, a protected window appearing anywhere in the response, the joint bound applies directly, $\Pr_q[E] \le n \Pr_{p_s}[E]$; the conditional one bounds each position against the anchor's probability given the *served* prefix, which an auditor can sample only by querying the deployment.

Table 31: All arms of Table 2, and the rule that never serves an empty draw (*never empty*), one judge pass, 500 prompts. Certified max-divergence nats (KL at $k = 0.5$) for any 50-token span given the text before it (—: only the output's, which covers the first span) and for the 200-token output. Gain over the arm's own zero-budget control, judge B with 95% interval, and judge G. *Leak*: near-verbatim recall of 100 memorised passages, the memoriser as risky model (0.495 alone); for selection, the best of its 64 draws.

mechanism	span	output	gain, judge B	judge G	leak
anchored decoding, $k = 0.5$	—	100	-0.0030 $[-0.0245, +0.0185]$	$+0.015$	—
budget up front	—	4.16	$+0.0230$ $[+0.0010, +0.0445]$	-0.0265	—
per-token pathwise meter	—	33.27	$+0.0125$ $[-0.0065, +0.0315]$	-0.0185	—
	—	83.18	$+0.0160$ $[-0.0050, +0.0370]$	$+0.0125$	—
windowed meter, W per span	4.16	16.64	$+0.0115$ $[-0.0110, +0.0340]$	-0.047	0.000
	12.48	49.91	$+0.0190$ $[-0.0035, +0.0415]$	-0.0445	0.000
	24.95	99.81	$+0.0375$ $[+0.0150, +0.0600]$	-0.0045	0.000
	40	160	$+0.0375$ $[+0.0140, +0.0610]$	$+0.0475$	0.000
	125	500	$+0.0795$ $[+0.0530, +0.1055]$	$+0.130$	0.106
best of 64	—	4.16	$+0.1015$ $[+0.0775, +0.1260]$	$+0.176$	0.000
best of 64, never empty	—	4.16	$+0.1010$ $[+0.0775, +0.1245]$	$+0.155$	—
installments, $L = 25$	12.48	33.27	$+0.1455$ $[+0.1200, +0.1715]$	$+0.2855$	—
installments, $L = 10$	24.95	83.18	$+0.1580$ $[+0.1330, +0.1835]$	$+0.305$	—

Table 32: The windowed meter by budget. *Recall*: mean near-verbatim recall of 100 memorised passages with the memoriser as the risky model, served plainly and through the chat template (the memoriser alone reads 0.4952 and 0.2193, the anchor alone 0). *Chat*: selection's gain minus the windowed meter's, both against the chat-served 8B, judges B and G, 95% intervals.

W, nats	recall, plain	recall, chat	chat, judge B	chat, judge G
4.16	0.0000	—	—	—
12.48	0.0000	—	$+0.0450$ $[+0.0095, +0.0810]$	$+0.0970$ $[+0.0590, +0.1355]$
24.95	0.0000	0.0000	$+0.0065$ $[-0.0305, +0.0435]$	$+0.0530$ $[+0.0110, +0.0945]$
40	0.0000	0.0000	-0.0525 $[-0.0895, -0.0150]$	-0.0295 $[-0.0725, +0.0145]$
60	0.0063	—	—	—
80	0.0286	0.0043	—	—
100	0.0634	—	—	—
125	0.1058	0.0632	-0.1600 $[-0.1985, -0.1220]$	-0.2815 $[-0.3335, -0.2295]$
150	0.1476	—	—	—
200	0.3993	—	—	—

Selection in installments. Both selection rules sample a bounded tilt of the anchor, a law $p_s h$ with $h \le e^{\lambda}$ at every length: the Gibbs tilt $h = e^{\lambda U}/Z$ of any utility $U \in [0,1]$ is one, best-of-n is another, $h = nF(f)^{n-1}$ with F the anchor's distribution of a tie-free score f, so $\lambda = \log n$, and installments build one causally, block by block, with the completed candidate's reward as a sampled value. Section 3's installment rule draws n continuations of up to L tokens from the anchor alone, scores each by the committed reward of the candidate completed to 200 tokens by one anchor rollout, serves the argmax and repeats until 200 tokens or an end-of-text; at $L = 200$ it is whole-output selection through the same code, on fresh draws. Proposition 2 holds per block, so B blocks certify the output at $B \log n$ and any 50-token window at $\min(\lceil 50/L \rceil + 1, B) \log n$ (Table 33; registered in `results/onset_prediction_blockwise.md`). At 64 draws per block, installments of 10 and 25 tokens beat one choice among 64 whole drafts, by $+0.0415$ $[+0.0185, +0.0645]$ and $+0.0245$ $[+0.001, +0.0475]$, against our registered prediction of a tie, and installments of 50 tie it; their window certificates, 24.95 and 12.48 nats, stay far below the protected window's 159.8. Two concessions go with them: the whole-output certificate, $\lceil T/L \rceil \log n$, grows linearly with the output as a meter's does, so installments are not a budget that does not grow, and at 1,000 tokens, with 8 draws a block, they lose to one whole-output choice (Appendix G); and a near-verbatim window at their per-window certificate is no easier to audit than the windowed meter's at the same W (Section 3). A disjoint draw, registered after this one was read, reproduces it, where we had predicted a tie at $L = 25$: $+0.0435$ $[+0.020, +0.067]$ and $+0.033$ $[+0.011, +0.0555]$ at $L = 10$ and 25 under judge B, $+0.0635$ $[+0.033, +0.0945]$ and $+0.0595$ $[+0.028, +0.0905]$ under judge G; read post hoc by judge G, the first draw's advantage is $+0.083$, $+0.0635$ and $+0.031$ at $L = 10$, 25 and 50 (`results/blockwise_replication.csv`). Held instead to one whole-output $\log 64$, spent as 2, 3 or 6 installments of 8, 4 or 2 draws, they lose to one choice among 64 by 0.0635, 0.082 and 0.088: a decoder steered by the same reward with its budget of $\log 64$ fixed up front, the closest we ran to a token-level reward tilt at that budget, does worse than selection spent once. Scored on the prefix alone, without the rollout, the rule serves 23 words at the median and loses to the rollout by 0.0395 $[0.02, 0.059]$. The risky model as planner, scoring each 10-token block by $\log p_r - \log p_s$ over the same 64 draws, gains $+0.077$ $[+0.054, +0.1005]$ over the anchor, $+0.026$ $[+0.001, +0.051]$ more than the meter does at $k = 10$ (post hoc), and loses to the reward by 0.0895 $[0.064, 0.115]$: the same choice buys more when the reward makes it than when the risky model does. The memoriser choosing twenty times among 64 ten-token continuations reads near-verbatim recall 0.0000 on all 100 passages it memorised (alone it reads 0.3925), and ROUGE-L 0.1168 against whole-output selection's 0.1073. With the prompt and the served text each in its own slot of the reward's template, every rule serves more empty answers than the committed pool's 41 (103 once, 104 to 110 in installments of 64 draws, 113 to 192 at one $\log 64$), and the planner none.

CP-k's rejection rule, run. CP-k (Vyas et al., 2023) rejection-samples complete outputs of the protected model against safe models trained on disjoint shards. No model pair here meets that premise, so we ran its rule with the one safe model we have. For each prompt the risky model's $M = 64$ draws are tested in seed order. The first whose realised log-ratio $R(y) = \log p_r(y \mid x) - \log p_s(y \mid x)$ over its served tokens is at most κ is served; otherwise the anchor's own draw is. Then $q(y) \le (Me^{\kappa} + 1)\, p_s(y)$ for every y, a pathwise certificate of $C = \log(1 + Me^{\kappa})$ whatever the acceptance rate. CP-k's own rule retries until it accepts and pays more as acceptance falls: 168.35 nats at half acceptance, above a window's 159.8, and 87.97 at one output in a hundred (`results/cpk_certificate.csv`). Table 34 runs the rule at the certificates of Table 31 (registered in `results/onset_prediction_cpk_baseline.md`). A risky draw's median R is 169.6 nats. So at $\log 64$ and at 33.27 nats the rule serves the anchor on every prompt and gains nothing, and at 83.18 it serves the risky model on 26.8% of prompts. Selection beats it at each matched certificate: best-of-64 by $+0.110$ $[+0.0855, +0.135]$ at $\log 64$, and installments by $+0.143$ $[+0.119, +0.168]$ at 33.27 and $+0.1205$ $[+0.091, +0.1485]$ at 83.18, all

Table 33: Selection in installments, judge B, order-averaged, one draw each (seed 20260925). Certificates in nats for the whole output and for any 50-token window given what precedes it; gains over the anchor-alone control and differences from one choice among 64 (first row), paired over the 500 prompts; $\lceil 200/L \rceil$ blocks (`results/blockwise.csv`).

rule	L	n	output / window	gain over the anchor	minus once
once	200	64	4.16 / 4.16	$+0.125$ $[+0.1015, +0.1485]$	—
installments	50	64	16.64 / 8.32	$+0.129$ $[+0.1045, +0.153]$	$+0.004$ $[-0.019, +0.0265]$
	25	64	33.27 / 12.48	$+0.1495$ $[+0.1245, +0.175]$	$+0.0245$ $[+0.001, +0.0475]$
	10	64	83.18 / 24.95	$+0.1665$ $[+0.141, +0.1925]$	$+0.0415$ $[+0.0185, +0.0645]$
one log 64	100	8	4.16 / 4.16	$+0.0615$ $[+0.0385, +0.085]$	-0.0635 $[-0.085, -0.0415]$
	67	4	4.16 / 2.77	$+0.043$ $[+0.0205, +0.0655]$	-0.082 $[-0.1055, -0.0585]$
	34	2	4.16 / 2.08	$+0.037$ $[+0.0155, +0.0585]$	-0.088 $[-0.1115, -0.0645]$
prefix only	50	64	16.64 / 8.32	$+0.0895$ $[+0.0665, +0.1125]$	—
planner	10	64	83.18 / 24.95	$+0.077$ $[+0.054, +0.1005]$	—

three confirmed under judge G. Against the per-token pathwise meter at the same nats it is unresolved: $+0.022$ $[-0.006, +0.0495]$ at 83.18 and -0.014 $[-0.0335, +0.006]$ at 33.27. We registered that its acceptances would be short outputs. They are not, since nearly every draw runs to the 200-token cap. Against the memoriser the rule serves nothing near-verbatim through 250 nats and leaks from 600 (0.0563 recall; the memoriser alone reads 0.4629).

Table 34: CP-k's rejection rule with the anchor as its safe model, 64 risky draws per prompt, 500 prompts. C: pathwise certificate in nats, whole output; κ: the acceptance threshold on the realised log-ratio; *risky*: share of prompts served a risky draw. Gain over the anchor's own draw, judge B with 95% interval, and judge G. *Leak*: near-verbatim recall of 100 memorised passages with the memoriser as risky model.

C	κ	risky	gain, judge B	judge G	leak
4.16	-0.02	0.0%	$+0.0000$ $[+0.0000, +0.0000]$	$+0.0000$	0.0000
33.27	29.11	0.0%	$+0.0000$ $[+0.0000, +0.0000]$	$+0.0000$	0.0000
83.18	79.02	26.8%	$+0.0395$ $[+0.0245, +0.0555]$	$+0.0510$	0.0000
159.83	155.67	99.6%	$+0.0865$ $[+0.0620, +0.1105]$	$+0.1335$	0.0000

At 1,000 tokens. Every judged arm above stops at 200 tokens. We asked whether whole-output selection still buys utility at 1,000 or needs installments, whose certificate grows with the output, and how either compares with the meters at matched nats. We ran every mechanism at $T_{\max} = 1000$ on the 500 prompts and judged it uncut against the risky model's own 1,000-token draw, registered in advance (`results/onset_prediction_long_outputs.md`; Table 35). One choice among 64 whole drafts still gains $+0.063$ $[+0.043, +0.083]$ over the anchor at the same log 64. It beats every meter at its own certificate: the per-token pathwise meter by $+0.062$ $[+0.0395, +0.084]$, the budget granted up front by $+0.0305$ $[+0.0055, +0.056]$ and the windowed meter by $+0.0445$ $[+0.020, +0.0695]$. It also beats anchored decoding at $k = 0.1$ (100 nats) by $+0.0775$ and at $k = 0.5$ (500 nats) by $+0.104$, all confirmed under judge G. Installments of 8 draws per 100 tokens, certified at 20.79 nats for the output and 4.16 for any window, gain $+0.0185$ $[-0.0005, +0.037]$. Against the meters at their own 20.79 nats they are unresolved under B, $+0.0165$ $[-0.0045, +0.038]$ pathwise and $+0.019$ $[-0.001, +0.0395]$ KL, where we had predicted wins that only judge G confirms. They lose to one choice among 64 by 0.0445 $[0.026, 0.0635]$ (G: 0.059), so long outputs do not need them. The anchor's drafts are short: best-of-64 serves a mean 203 tokens, and 32.6% of its outputs are longer than 200, while the risky model always runs to the cap. On the 163 prompts where the served draft is longer than 200 tokens its gain is $+0.092$ $[+0.057, +0.127]$. At this length no meter improves on the anchor by more than 0.033 under B, and anchored decoding falls below it at $k = 0.1$, 0.5 and 10. The judges part on the risky model's own 1,000 tokens: G reads them $+0.1095$ over the anchor and B -0.0175.

Where extraction begins. Table 36 gives the nine pairs Figure 2(a) plots. Every pair clears the admission criterion (sampled $k = -1$ recall ≥ 0.10), and $k = 0$, the anchor alone, reads 0.000 at all nine. Two columns are reported because they are not controlled: the memorisers span a factor of 5.1 in their own recall, and five of the nine did not converge. Part of the residual tracks the words a 20-token seed hands the adversary (Spearman -0.958), a caution for seeds fixed in tokens (Carlini et al., 2023). Every sweep decodes T tokens for every passage, the longest target in it: 296 at the audited pair, 306 to 468 elsewhere and 650 in the KL3M tokenizer's tokens. The median target is 0.84 to 0.89 of T long, so its own certificate is void from that fraction of $s(x)$,

Table 35: Every mechanism at $T_{\max} = 1000$, 500 prompts, judged uncut against the risky model's own 1,000-token draw. Certified nats for any 50-token span given the text before it (—: none beyond the output's) and for the output, max-divergence except anchored decoding's KL. *Tokens*: mean served length. Gain over the arm's own zero-budget control (the anchor alone, or the pool's single draw), judge B with 95% interval, and judge G (`results/long_outputs.csv`).

mechanism	span	output	tokens	gain, judge B	judge G
best of 64, whole output	4.16	4.16	203	$+0.0630$ $[+0.0430, +0.0830]$	$+0.2115$
best of 16	2.77	2.77	163	$+0.0305$ $[+0.0105, +0.0510]$	$+0.1665$
installments, $L = 100$, $n = 8$	4.16	20.79	159	$+0.0185$ $[-0.0005, +0.0370]$	$+0.1525$
pathwise meter	—	4.16	220	$+0.0010$ $[-0.0095, +0.0115]$	$+0.0070$
	—	20.79	339	$+0.0020$ $[-0.0110, +0.0150]$	-0.0080
budget up front	—	4.16	275	$+0.0325$ $[+0.0160, +0.0490]$	-0.0075
windowed meter	4.16	83.18	499	$+0.0185$ $[+0.0015, +0.0355]$	-0.0555
anchored decoding, $k = 0.0208$	—	20.79	215	-0.0005 $[-0.0095, +0.0085]$	$+0.0005$
anchored decoding, $k = 0.1$	—	100	461	-0.0145 $[-0.0280, -0.0010]$	-0.0375
anchored decoding, $k = 0.5$	—	500	831	-0.0410 $[-0.0575, -0.0240]$	-0.0365
anchored decoding, $k = 10$	—	10^4	1000	-0.0175 $[-0.0330, -0.0015]$	$+0.1095$

below the onset at eight of nine pairs (by 0.010 to 0.273) and 0.004 above Pleias-1.2B's; a 50-token window's is void from $50/T$, 0.08 to 0.17 of $s(x)$ (`results/onset_window_threshold.csv`).

Table 36: Onset of extraction on nine pairs, as a fraction of the anchor's surprisal rate $s(x)$ (nats per token; s_r the memoriser's, on the same tokens). Onset is the budget at which mean single-query recall first reaches 0.01, interpolated inside its bracket; n protected passages. *mem.*: the memoriser's own sampled recall at $k = -1$. *ep.*: epochs run against the cap, so a run below its cap reached its stop-loss. †: baselines measured in a second invocation reproducing the sweep's protocol line.

pair	$s(x)$	s_r	n	mem.	ep.	onset	95% CI	onset / $s(x)$
KL3M 1.7B + mem. KL3M 1.7B	2.21	0.781	100	0.411	40/40	2.58	[2.52, 3.32]	1.166
Comma 7B + mem. Comma 7B†	2.39	0.179	100	0.719	12/12	2.13	[2.00, 2.92]	0.892
KL3M 520M + mem. KL3M 520M	2.42	0.215	100	0.519	11/40	2.54	[2.45, 3.00]	1.053
Phi-3.5 mini + mem. Phi-3.5 mini	2.84	0.009	100	0.566	40/40	2.63	[2.27, 3.08]	0.926
Pleias 1.2B + mem. Pleias 1.2B	3.21	0.365	100	0.909	30/40	2.82	[2.53, 3.08]	0.878
TinyComma 1.8B + mem. Llama-8B†	3.24	0.194	100	0.492	12/12	2.87	[2.72, 3.45]	0.887
open-calm 1B + mem. open-calm 1B	3.36	0.416	100	0.181	40/40	3.45	[3.25, 4.97]	1.027
open-calm 3B + mem. open-calm 3B	3.38	0.314	100	0.924	21/40	3.36	[3.24, 3.63]	0.993
Pleias 350M + mem. Pleias 350M	3.55	0.326	458	0.875	16/40	3.27	[3.07, 3.83]	0.920

Two refined predictions, both refuted. The rate a decoder must permit before it can pay for x at all,

$$r(x) \;=\; s_s(x) - s_r(x), \tag{4}$$

predicts an onset ratio of $1 - s_r/s_s$ that must move with memorisation. Committed for the last five pairs before any was swept, it is refuted under all three registered rules: every miss low, held-out errors of 0.365, 0.507 and 0.296 nats, and a directional claim that inverts. Eq. (4) is right about what a decoder must *afford* (Table 29's crossover) and wrong about where leakage *begins*. The running maximum of Proposition 5 fails in the complementary direction: warping the anchor's logits to $\tau = 0.4$ moves $s(x)$ by 67% and 62% on two pairs and the onset follows at an elasticity of $+0.72$ $[+0.41, +0.84]$ and $+0.61$ $[+0.40, +0.96]$, while k_{crit} misses by 21.9% against a no-change null's 12.3%.

The level is the claim, not the ordering. Holding one pair, its corpus, anchor and grid fixed and varying only the memoriser's fine-tuning length moves the onset ratio from 0.8756 to 1.3477; re-seeding at a fixed epoch count moves it by 0.033 to 0.080 where the fine-tune converged and by 0.2597 where the memoriser is marginal. **The within-pair spread is $1.64\times$ the entire between-pair spread** of 0.8784 to 1.1658, and whether a pair's interval excludes 1 flips inside that single pair, from $[0.68, 0.98]$ to $[1.01, 1.59]$. On two further corpora (Table 37) two of three pairs vary by 0.088 and 0.066, inside their own bootstrap widths of 0.227 and 0.287, and the third by 0.436 against 0.171; on the third corpus the ordering inverts, Pleias-1.2B reading 1.314 above the fine-tokenizer pair, whose interval no longer excludes 1. Every onset in these arms lands between 0.80 and 1.35 of $s(x)$, which is what Proposition 1 claims; which pairs begin above 1 is confounded with training variables the table does not hold fixed.

Table 37: Three pairs re-measured on two further protected corpora, anchor, architecture, settings, seed and grid held fixed. *novels*: the pair's ratio on the sixteen CopyBench novels (Table 36); *no-cross*: bootstrap resamples that never reach the onset on the grid.

pair	onset	onset / $s(x)$ (95% CI)	no-cross	Eq. (4)	other corpus	novels
600 excerpts of 50 public-domain books						
KL3M-520M	2.608	1.102 [1.07, 1.42]	0.0%	2.218	—	1.053
Pleias-1.2B	2.513	0.895 [0.85, 1.17]	0.0%	2.463	—	0.878
Phi-3.5-mini	2.704	0.949 [0.79, 1.33]	0.5%	2.813	—	0.926
600 passages over all 31 books of BookMIA's `attack_train` split						
KL3M-520M	2.465	1.014 [0.97, 1.13]	0.0%	2.197	1.102	1.053
Pleias-1.2B	4.006	1.314 [1.01, 1.59]	0.4%	2.614	0.895	0.878
Phi-3.5-mini	2.607	0.992 [0.87, 1.37]	0.0%	2.619	0.949	0.926

The uncertified interval is the opening. On the 758 passages under four safe models, k_{crit}/s has a median of 4.25 to 6.10, but the running maximum in Eq. (3) is attained at the *first* token of the work in 87.7% to 90.5% of cases, because the bucket must pay for the opening out of one step's allowance. An adversary who supplies one genuine prefix token removes it: the ratio falls to 1.32–1.48 and stays there at 5, 10 and 20 supplied tokens. Per token instead of per character the whole-work median is 3.88–5.98 and the ratios after a supplied prefix are unchanged (1.35–1.41). This is why oracle-prefix windows recover 0.0965 of a passage at $k = 3$ against 0.0116 for a single query (Table 41).

Vacuity on a hundred books. The vacuity verdict needs no memoriser, only forward passes of the anchor, so we re-measured it on BookMIA (Shi et al., 2024) with the 758 passages as a positive control (Table 38). 65 of 4,935 seen passages are surprising enough that a 600-nat budget does not cover them, which is the weaker reading registered rather than a replication; the unseen half and the control read 1.0000. The halves are not exchangeable under an anchor that saw neither (medians 6.0% apart against a specified 5%), so no arm treats them as a matched pair. A first pass scored vacuity as $k \geq s(x)$ per token and read 0.326 on the control: that criterion agrees with $K \geq S(x)$ only when the cap and the work have the same length, and a deployment caps at 200 tokens against a protected span near 60.

Table 38: Vacuity on BookMIA at the published $K = 600$. The seen and unseen halves are not a matched pair under an anchor that saw neither.

corpus	passages	books	vacuous, $K = 600$	median $S(x)$	median $s(x)$/tok
CopyBench (control)	758	16	1.0000	204.8	3.197
BookMIA, seen	4,935	50	0.9868	182.6	3.114
BookMIA, unseen	4,935	50	1.0000	192.1	3.313

A pair where memorisation is the only variable. The earlier audit recorded a vocabulary mismatch (64,000 against 128,256 tokens) as the reason no second anchor could be fused with the risky model. The obstacle was an implementation check that the embedding table have exactly len(tokenizer) rows, which the Common Pile 7B checkpoint fails by padding 64,000 tokens to 64,256 rows; masking the untrained rows out of the solve removes it. We fine-tune that 7B model until it reproduces the protected passages (0.915 recall on its training split, 0.000 held out) and fuse it against the unmodified model it came from, so anchor and risky model share every weight but those memorisation moved. Across 7 budgets, both baselines, 5 strategies and 31,738 budgeted queries there is no invariant violation. The anchor alone reproduces 0.0000 under every strategy; at $k = 20$ the decoder is the identity (single-query recall 0.7189, equal to the unconstrained model's to four decimals); recall is 0.000 at every budget up to $k = 1$, 0.048 at $k = 3$ and 0.197 at $k = 5$, beginning near $s(x) = 2.39$ and far below $k_{\text{crit}} = 11.32$.

A better anchor widens the margin. Ten openly licensed safe models (0.17B to 7B, three corpora) score the same 758 passages and the same 300 ordinary risky-model generations, in nats per character so no shared vocabulary is needed. Both the passages' rate $s(x)$ and the ordinary output's divergence c_{use} fall as the anchor improves, and c_{use} falls faster, so the margin $s(x)/c_{\text{use}}$ rises with capability within every family, for all sixteen novels individually (Table 39), and with training tokens at fixed size (4.63 to 5.02 from the 1T to the 2T checkpoint of one 7B model). Pooled across families it is not monotone in parameter count, because the families sit at different levels.

Table 39: The margin $s(x)/c_{\text{use}}$ across the widest capability span in each family, paired over the same sixteen novels.

corpus	span	Δc_{use}	$\Delta s(x)$	margin	sign test
Common Corpus	$0.35 \to 3$B	-25.8%	-9.8%	$3.45 \to 4.19$	$16/16$, $p = 3.1 \times 10^{-5}$
KL3M	$0.17 \to 3.7$B	-31.3%	-16.0%	$1.94 \to 2.37$	$16/16$, $p = 3.1 \times 10^{-5}$
Common Pile	$1.8 \to 7$B	-28.6%	-12.0%	$4.07 \to 5.02$	$16/16$, $p = 3.1 \times 10^{-5}$

H THE FOUR REPAIRS, IN FULL

Table 40: **The four repairs a per-token meter has left**, each measured at the audited pair with bands fixed in advance (Appendix H). He et al. (2026) ablate the first and last themselves, as their ∞-Rényi decoder and their Global variant.

repair	measured at the audited pair, $k = 3$ unless stated	result
sharper charge (α: 1→8)	oracle recall $0.0965 \to 0.0012$; binds on 99.6% of steps vs 0.4%	the anchor
lower cap ($k = 0.5$)	no judge tells it from its anchor (-1.03σ, $+1.52\sigma$; 600 pairs)	the anchor
better schedule	3 to 13% more fidelity at $k \leq 1$, under 1.4% at $k = 3$, same spend	percent
budget up front	$+0.0545$ $[+0.0315, +0.0775]$ at 64 nats, selection $+0.1045$ at $\log 64$	loses

A sharper charge. Table 41 runs four Rényi orders at the authors' $k = 3$. Every one publishes $K = 3T$ and is vacuous for 100% of the 758 protected passages. Raising the order cuts oracle recall $0.0965 \to 0.0012$, an $80\times$ cut, by ceasing to serve the risky model: at $\alpha = 8$ the constraint touches 99.6% of ordinary steps. At matched *utility* rather than matched budget (Table 42, fidelity to p_r on the geodesic as the utility axis, no judge) the higher order usually helps at $k = 1$, by a factor of 1.0 on one pair and 2.5×10^7 on another, and on four pairs α=8 makes the protected tokens *more* likely, by up to 1.3×10^6 on the mechanism's own configuration; at $k = 3$ matching forces every order past its vacuity threshold, and on nine of twelve pairs the ranking reverses.

Table 41: Four decoders at $k = 3$, all publishing $K = 3T$ and all vacuous for every protected passage. Attack columns: mean near-verbatim recall over 100 passages from the memorising 8B model; price columns: 150 ordinary prompts.

	attack recall		price on ordinary traffic		
order	single query	oracle windows	risky unchanged	steps touched	distinct-3
$\alpha = 1$ (KL)	0.0116	0.0965	99.3%	0.4%	0.9910
$\alpha = 2$	0.0010	0.0545	97.2%	2.5%	0.9924
$\alpha = 4$	0.0000	0.0042	9.4%	90.3%	0.9850
$\alpha = 8$	0.0000	0.0012	0.08%	99.6%	0.9872

A lower cap. At $k = 0.5$ neither outside judge separates the meter from its own safe model over 600 comparisons each (-1.03σ, $+1.52\sigma$); both separate only at $k \geq 10$ (-6.97σ, -3.29σ).

A better schedule. On the geodesic the charge and the fidelity bought are closed forms in $\psi(u) = \log Z(u)$,

$$C(\theta) = \theta\psi'(\theta) - \psi(\theta), \quad C'(\theta) = \theta\psi''(\theta), \qquad G(\theta) = \theta m - \psi(\theta), \quad G'(\theta) = m - \psi'(\theta), \tag{5}$$

with $m = \psi'(1)$, so $C'(0) = 0$ while $G'(0)$ is the Jeffreys divergence between the two models (Monteiro Paes et al., 2026): the first nat bought at a step is worth the most. The bucket takes whatever price G'/C' its allowance lands on; the best allocation of the same total equalises it. Computed from two teacher-forced passes over 30 targets of three kinds (memorised passages, a held-out novel, the risky model's own samples), the optimal reallocation buys 3 to 13% more fidelity at $k \leq 1$ and under 1.4% at $k = 3$, against an offline optimum no causal scheduler can beat (Tomasi et al., 2026); the three target kinds agree to within two points. The bucket already captures 46.7–54.5% of the $\theta = 1$ fidelity at $k = 0.25$ and 92.7–98.9% at $k = 3$, so three orders of magnitude are not there to recover.

The budget granted up front. The one causal policy Proposition 3 positively permits spends a budget granted at the start. Judged order-averaged against the committed opponent it gains -0.0015

Table 42: Factor by which an exact 50-token window becomes *less* likely under order α, at the budget k_α where it buys exactly the fidelity α=1 buys at the published k; below 1 the order leaks more at equal utility. Twelve pairs in seven families, `bfloat16` with a `float32` control per pair; $s(x)$ in nats per token.

		published $k = 1$			published $k = 3$		
pair	$s(x)$	α=2	α=4	α=8	α=2	α=4	α=8
Pleias 350M	3.50	$5.0{\cdot}10^{4}$	$2.5{\cdot}10^{7}$	$1.4{\cdot}10^{7}$	67.5	28.7	2.4
Pleias 1.2B	3.12	$3.6{\cdot}10^{4}$	$1.3{\cdot}10^{7}$	$1.9{\cdot}10^{7}$	67.3	113	27.2
Comma 7B	2.32	$4.3{\cdot}10^{3}$	$1.9{\cdot}10^{5}$	$9.0{\cdot}10^{3}$	5.6	2.2	0.26
KL3M 170M	2.61	$3.3{\cdot}10^{3}$	$1.8{\cdot}10^{4}$	455	4.1	0.37	0.12
TinyComma 1.8B	3.15	$3.0{\cdot}10^{3}$	5.3	$8.0{\cdot}10^{-7}$	0.67	$1.6{\cdot}10^{-3}$	$1.3{\cdot}10^{-4}$
KL3M 3.7B	2.20	$1.4{\cdot}10^{3}$	$8.6{\cdot}10^{3}$	909	1.2	0.37	0.18
KL3M 520M	2.42	$1.3{\cdot}10^{3}$	$1.0{\cdot}10^{3}$	1.5	1.4	0.073	$7.3{\cdot}10^{-3}$
Llama-3.2 3B	2.81	$1.1{\cdot}10^{3}$	828	1.6	2.8	0.30	0.048
KL3M 1.7B	2.25	638	$7.4{\cdot}10^{4}$	$4.4{\cdot}10^{5}$	46.4	175	311
Llama-3.2 1B	2.78	158	9.4	$8.7{\cdot}10^{-3}$	0.84	0.048	$7.8{\cdot}10^{-3}$
Qwen2.5 7B	2.62	71.8	1.0	$1.7{\cdot}10^{-4}$	0.47	0.032	$3.7{\cdot}10^{-3}$
Phi-3.5 mini	2.71	47.5	7.8	0.078	0.15	0.011	$2.7{\cdot}10^{-3}$

$[-0.0200, +0.0170]$ at $\log 8$ nats, $+0.0215$ $[+0.0015, +0.0415]$ at $\log 64$ and $+0.0545$ $[+0.0315, +0.0775]$ at 64 nats: the causal horn is not empty, but at matched budget selection leads by $+0.083$ $[+0.052, +0.1135]$ and no budget on the grid reaches selection's $+0.1045$ at $\log 64$. Choosing *where* to spend does not help: a rule that spends at step t only if the full-tilt demand reaches τ nats reserves as designed (median spending step $3 \to 17$) and gains $+0.0215$, $+0.0150$, $+0.0110$, $+0.0045$ and $+0.0070$ at $\tau = 0, 1, 2, 4, 8$, falling except for a tick at the last, because a higher bar releases the budget less often (0% to 86.7% of trajectories never spend) *and* buys less when it does. An earlier arm, judged single-order and not compared with these, gave the front-loaded rule -0.008 $[-0.056, +0.041]$ at $\log 8 = 2.0794$ nats against selection's $+0.054$ at the same budget.

How far each mechanism sits from Lemma 1's frontier. Read on the repaired text, order-averaged, with each arm's own control as the safe law of U, selection at $n = 64$ sits $35.4\times$ the frontier on its 3.175-nat bound and the meter at $k = 10$ $7{,}384\times$ on its 171.3 nats spent (`results/frontier_distance_h2h_deecho.csv`); the $32\times$ and $11{,}646\times$ an earlier version gave were read on text that carried the prompt's tail. Proposition 4's conservative ratio needs π, the chance that the anchor attains the top utility. On Figure 1(d)'s instrument the anchor wins both presentation orders on 2.2% of prompts, 4.0% of questions, 2.0% of biographies and none of the stories, whose best anchor level is 0.75: $\log(1/\pi)$ is 3.82 pooled, 3.22 and 3.91 for the first two classes, and infinite for stories, where the bound says nothing. The meter's 171.3 nats are at least $44.9\times$ what winning every comparison needs, $53.2\times$ and $43.8\times$ by class. Because $\Lambda_s^*(u) \le \log(1/\pi)$ for any utility the decoder does not lower, the proposition's second clause follows from its first, a spend of at least γT; it restates the spend in units of what the utility requires rather than adding a bound. The single-order pass this paragraph once used gave $\log(1/\pi) = 1.30$ and $132\times$: there a win in the one order shown came on over a quarter of prompts, and a win in both orders is the stricter event. Every figure here pools the prompts, so it bounds the prompt-averaged divergence, which each per-prompt certificate implies but does not equal; one anchor draw per prompt cannot estimate a per-prompt π. At the level each arm reaches rather than the top one, Λ_s^* is 0.00255, 0.0115 and 0.0232 nats at $k = 1$, 3 and 10, a ratio of 52,864, 14,357 and 7,384; at $k = 0.5$ the meter gains nothing and the ratio is unbounded (`results/frontier_ratio.csv`).

Two anchors closer to the risky model. Neither anchor here is a safe model, so no certificate, leakage or $s(x)$ number is taken from either; the pairs test whether the reversal is the mechanism's or the anchor's. A token-level meter needs a shared vocabulary, and no other openly licensed anchor has one with this risky model. At Llama-3.2-1B the meter is useful, $+0.051$, $+0.056$ and $+0.055$ under judge B at $k = 1, 3, 20$ for 88.1 to 123.4 nats, and selection at $n = 8$ gains more, $+0.076$ $[+0.031, +0.122]$ and $+0.154$ $[+0.104, +0.200]$ on two judges, for 1.204 nats. At Llama-3.2-3B-Instruct, the closest meterable anchor and the one the protocol named the hardest case, the meter resolves at no budget on either judge (all eight point estimates negative, spends 54.8 to 60.6 nats over a $40\times$ range of caps) while selection gains $+0.150$ $[+0.106, +0.194]$ and $+0.273$ $[+0.227, +0.323]$. A close anchor flattens the meter onto the anchor. The arms each pass holds twice (the meter's $k = 0$ and selection's first draw) differ by at most 0.034, and the same size appears independently at Comma-7B (0.039), so **the cross-pass floor on a judged gain is about** 0.04, and gains from different passes are never compared here.

I Limitations in full

One protected corpus. Every extraction measurement runs on sixteen English genre novels from CopyBench and BookMIA, plus the four French and German works of the multilingual arm. The margin's sign test, $p = 3.1 \times 10^{-5}$, is the smallest value sixteen paired items can produce: it says the trend holds for every novel we have, not that it was independently replicated. Non-literal reproduction and other genres are untested. On two further corpora the level of the onset survives, every onset within about $1.3\times$ of $s(x)$, and the ordering of pairs does not (Appendix G), so an onset ratio is reportable to its order of magnitude and no more finely. Varying only a memoriser's fine-tuning length moves one pair's ratio from 0.8756 to 1.3477, $1.64\times$ the nine-pair spread; re-seeding moves it by 0.033, 0.066 and 0.080 where the fine-tune converged and by 0.260 over three seeds and 0.347 over four where it did not; and neither intervention that would make convergence a cause could be built (a loss floor of 0.0222 against a 0.02 stop-loss; a memoriser that reproduces nothing after a 17% change in learning rate). The nine-pair table's memorisers are matched on none of training length (a factor of 5.1 in sampled $k = -1$ recall), batch size, accumulation, stopping rule or convergence. The seed-word gradient $\rho = -0.958$ keeps its sign in every re-draw under the largest re-seed noise measured and $p < 0.05$ in 92.3% of them, at a median of -0.849: a rank ordering, not a coefficient.

More passages do not narrow the onset interval. The bootstrap interval on onset / $s(x)$ has a mean width of 0.285 across the pairs at 100 passages each; on one pair under one metric it is 0.219 at $n = 100$ and 0.215 at $n = 458$, where independent sampling noise predicts 0.102. The variance is between works, not windows, so several registered bands about 0.1 wide land undecided even where the shift is large and in the predicted direction. More distinct works would fix it; more windows will not.

The onset's range has one cause we can name and one we cannot. $s(x)$ spans $1.87\times$ across the nine pairs, short of the range a data collapse normally rests on. Both KL3M pairs exclude 1 and differ from the rest in tokenizer, target length in tokens and, for one, memoriser strength; a pair built to remove the last memorises thoroughly ($s_r/s_s = 0.09$ against 0.35) and still reads 1.053. The tokenizer remains confounded with what it moves at once: the adversary's seed (7.5 words against 13.9–15.0), the decode steps before the passage (445 against 197–230) and the target length.

Lemma 1 is a converse. It lower-bounds the budget any policy needs; we do not bound the gap between that optimum and a causal token-bucket decoder, which is a factor of 5 to 32 between the divergence on the risky model's own rollout and the divergence paid on a budgeted one.

What selection anchoring cannot do. *The support ceiling.* $q(y) \le n\, p_s(y)$ is also a quality ceiling: no n produces text outside the anchor's support. At $n = 8$ on the judged workload it does not bind: the oracle among the same eight candidates reaches $u = 0.807$ where the deployed scorers reach 0.52. Across anchors it is confirmed on AlpacaEval: the audited anchor gains $+0.074$ $[+0.042, +0.108]$ and $+0.020$ $[-0.011, +0.052]$ under two judges, Comma-7B $+0.075$ $[+0.044, +0.106]$ and $+0.152$ $[+0.119, +0.185]$; MT-Bench's 80 prompts resolve nothing at a paired half-width of 0.10. Within one anchor, across AlpacaEval's five sources and two MT-Bench families, the correlations of -0.79 and -0.70 between gain and control level are not separable from a null in which selection does nothing (-0.36 ± 0.34 and -0.34 ± 0.35, $P = 0.09$ and 0.17).

Empty drafts. The audited anchor's first draw is empty on 14.6% of prompts in its $n = 8$ arm and 12.8% in its $n = 64$ arm, against 0.0% at both Pleias anchors, 0.2% at KL3M-1.7B, 18.4% at Comma-7B and 31.6% at Comma-7B-1T, all three Comma anchors above the 5% the admission criterion allows, recorded rather than exempted. On the 427 prompts whose $n = 1$ draw is non-empty the audited anchor's $n = 8$ gain is $+0.079$ $[+0.028, +0.128]$ against $+0.073$ on all 500, and every interval that excluded zero still does but one, Comma-1T's under judge B, whose lower end was $+0.001$.

Overoptimisation. To $n = 512$ no judged ladder turns over, against registered bands that allowed it to; under a task metric the 7B reward turns over on TriviaQA and CoTaEval (Table 18), and nothing in Proposition 2 prevents it on a judged workload at larger n.

The scorer is a second model and a stated requirement. It cannot break the certificate but can degrade utility arbitrarily, so served utility and the certificate must be monitored separately. Every reward turn-over here is measured at `Qwen2.5-7B-Instruct` and none survives a larger scorer, so on knowledge-intensive tasks selection under a pointwise reward carries a scorer-size requirement, and a deployer who ignores it can be worse off at every n than serving the anchor once. On the judged workload the two larger scorers lift nothing (-0.0285 and -0.0085 against the 7.6B rung), and no latency arm ran above 7B.

The bound is relative to p_s. A safe model that has seen the work carries its contamination through, multiplied by n (Table 22); the 0.0000 of Section 4 is n times a base rate that is zero because the anchor never saw these novels.

The metered decoder as its authors run it. He et al. (2026) pair base risky models (`Llama-3.1-70B`, `Qwen2.5-72B`, `Llama-4-Scout`) with openly licensed anchors, decode books at temperature 0.7 with a repetition penalty of 1.1, fuse mismatched vocabularies through AnchoredByte, and score utility by Prometheus-2 fluency and FActScore precision. We ran their token-level pair at temperature 1.0 with no penalty, the released logs' 8B-Instruct without its chat template (and, separately, with it), AnchoredByte only at their recommended pair, and their two metrics with open models in place of `gpt-4.1-mini`. None of this changes a certificate or a leakage number; each can change a judged one. No human rated anything and no lawyer assessed any output.

Two judges, one workload. Two judges at 600 comparisons per arm agree on the sign everywhere and on neither endpoint's magnitude: at $k = 1$ one reads -3.66σ against the anchor and the other $+0.11\sigma$, at $k = 10$ -6.97σ against -3.29σ. Served the unconstrained risky model, the first separates it at -6.09σ and the second at -2.49σ, and an earlier run of the same 500 comparisons gave -5.47σ and -1.49σ, so about a σ of run-to-run drift is the floor beneath which no separation here should be read.

The two leakage measurements are not on one axis. The meter's onset and selection's zero come from two pipelines whose memoriser-alone rows disagree (0.4921 against 0.3925; at a matched 300 tokens 0.4921 against 0.4428, a residual of 0.0493 against a registered gate of 0.02). The residual has two causes the arm cannot separate: changing the token budget changes the draw as well as the length, and the meter decodes each passage's own target length where selection decodes a flat 300. No joint table is built, every leakage number is read against a control from its own pipeline, and the comparison between the mechanisms is about what they spend, not what they leak.

J Related work in full

Table 43 sorts the defences by what their guarantee is about.

Table 43: Copyright defences at or near inference time, by what the guarantee is about. The first three rows bound the served text's law against a model (CPR does so for images); the others guarantee something about a named string, an n-gram list, an index, a datastore or a takedown list.

defence	what the guarantee is about	measured here
anchored decoding (He et al., 2026)	the served law against a safe model, charged per token ($K = kT$)	audited throughout
selection anchoring (this paper)	the served law against a safe model, charged once per draw (pathwise, $\log n$)	Sections 2 and 4
CP-Δ, CP-k (Vyas et al., 2023), CP-Fuse (Abad et al., 2025)	the served law against shard models, per query (CP-k once per output), given a disjoint split	leakage (below); CP-k's rule, run with one safe model (Appendix G)
n-gram blocklist, MemFree (Ippolito et al., 2023)	listed n-grams are never completed; no bound on the served law	leakage and utility (below)
TokenSwap (Prashant et al., 2025)	a memorised string's probability decays in its length	leakage and utility (below)
TRBS (Fu et al., 2026)	beams reranked against an index of the corpus	not measured (a code method)
BloomScrub (Zhang et al., 2025)	takedowns screened by a Bloom filter, with abstention	not measured
SILO (Min et al., 2024)	legal risk isolated in a nonparametric datastore	not measured
CPR (Golatkar et al., 2024)	near-access-freeness through retrieval, for diffusion models	not measured

Access-free guarantees and their discontents. Cohen (2025) show that per-query access-freeness does not compose across queries, and Cohen & Zhang (2026) prove barriers for counterfactual credit attribution whose non-composition is autoregressive: imposing it on the next-token predictor does not make the sequence-level model satisfy it. Ours is the same shape for a different quantity, priced in nats: a per-step charge honoured at every step leaves the event $\{\text{output} = x\}$ unprotected once the budget reaches $S(x)$. Segal et al. (2026) report that CP-style guarantees can be numerically uninformative in realistic fine-tuning and demonstrate extraction against NAF-protected models; their diagnosis is that memorisation puts sharp spikes on a few tokens, and their repair charges relative rather than raw probabilities, a change in *what* is metered. Proposition 1 closes a different lever: for every order of the same per-step charge the certificate becomes vacuous at the same budget, so a stronger charge buys precision below the threshold and never moves it. Neither result subsumes the other. That a KL bound does not imply a worst-case event bound is classical (Sason & Verdú, 2016; van Erven &

Harremoës, 2014); what we add is which budgets survive the requirement that the certificate be informative about a specific work and the mechanism remain usable.

Consensus sampling, the closest neighbour. Kalai et al. (2025) aggregate k generative models, an unknown subset of them safe, and serve a sample only where enough of them assign it comparable probability, abstaining otherwise. The instinct is ours, certifying the output relative to models that did not misbehave, and the premises are opposite: they need several models and no knowledge of which are safe and pay by abstaining; we need one named safe model, never abstain, and pay in draws. Both certificates inherit what their safe models hold, and ours is checkable in advance by vetting (Appendix E). A deployer with a vetted anchor and several risky models could run both.

TokenSwap, measured: cheaper than selection and just as suppressive. TokenSwap (Prashant et al., 2025) substitutes the probabilities of high-frequency grammatical tokens from a small auxiliary model and proves that a memorised sequence's probability decays exponentially in its length; TRBS (Fu et al., 2026) reranks beams against an index of a code corpus. Both guarantee something about a catalogue supplied, not about the served law relative to a safe model. We reimplemented TokenSwap from its Algorithm 1 with the published 110-word G on our 100 protected passages and 850 ordinary prompts. It suppresses completely (recall 0.0000, ROUGE-L ≥ 0.5 on 0 of 100 and ≥ 0.3 on 0, against 45 and 69 for the memoriser) and, judged against its own rule-off control, costs -0.1040 $[-0.1255, -0.0820]$, where selection at $n = 64$ costs -0.1655 $[-0.2000, -0.1320]$: TokenSwap is ahead of selection by $+0.0615$ $[+0.0285, +0.0950]$. The rule-off control is the risky model served through its chat template, judged against the committed templateless opponent, so selection's cost here is its distance from that model, which gains $+0.272$ over the anchor to selection's $+0.1065$: less than half the gap, as in Table 10's chat-template block. **The mechanism this paper is about is the most expensive of the three and the only one that publishes a bound on the served law.** The margin depends on the auxiliary: at `DistilGPT-2`, the 82M model its authors use, the two tie (-0.0325 $[-0.0645, +0.0005]$). And the suppression is fragile in a way $|G|$ does not reveal: with `KL3M-170m` as the auxiliary recall returns to 0.2113, with 23 of 100 passages over ROUGE-L 0.5, because that tokenizer keeps 171 of 431 G token ids but 1.0% of the probability mass, so the swap fires on 1.8% of steps instead of 25–29%. All four candidate auxiliaries read 0.0000 on the protected passages alone.

The blocklist, measured. Refusing any continuation that completes an n-gram of a listed work (Ippolito et al., 2023) is the incumbent a deployer would otherwise run. **It refuted the argument we built it to support**: collateral on ordinary text does not grow with the catalogue. At $n = 10$ the fraction of ordinary tokens blocked is 0.000% at 758 passages, 0.140% at 4,935 and 0.140% again at 9,870, where the index has doubled to 2.24M n-grams (0.000%, 0.147%, 0.147% at $n = 8$). Run as an exact rejection-sampling decoder against the memoriser on its own training split, it binds on 1.035% of steps and takes recall from 0.4192 to 0.0201; on ordinary prompts it fires on 0 of 850, so its judged utility is the unconstrained model's exactly ($+0.272$ $[+0.247, +0.2965]$ over the anchor with and without it, the chat-templated model's gain). We predicted it would leak the non-literal event and it did not: 0 of 100 passages reach ROUGE-L ≥ 0.5 under the rule, against 52 without it. On a *listed* work the blocklist costs no utility, which selection cannot approach, and it does not suppress more (0.0201 against its own 0.4192; selection 0.0000 against 0.3925). The difference is in kind: the blocklist must enumerate the works and is silent on one nobody listed, and it publishes no number about the served law, so nothing about it composes across queries, where $\log n$ adds exactly. Selection costs $21.8\times$ a metered decode in a server that runs only the anchor (Appendix F); a blocklist costs a hash lookup. A deployer can run both.

CP-Fuse, measured on leakage. We rebuilt CP-Fuse from Abad et al. (2025)'s description as two LoRA copies of the risky model, each fine-tuned on one of two disjoint halves of the protected passages and fused by the balancing rule. Each shard reproduces its own half (recall 0.6905 and 0.6965) and not the other's (0.0000); the fusion reproduces neither, 0.0000 from a single query and 0.0209 and 0.0199 under oracle-prefix windows (`results/cpfuse_audit_rebuild.csv`). A $50/50$ mixture without the balancing already holds single queries to 0.0295 and 0.0000, so most of the suppression is the disjoint split. An earlier build with its own fine-tunes has the same shape, 0.7538 and 0.7988 on the own shard against 0.0000 fused (`results/cpfuse_audit.csv`). The utility half was fixed in advance and withdrawn before it ran: on ordinary prompts CP-Fuse is the risky model with a device attached, and judged against that model a win is arithmetic, not evidence. Its guarantee holds per query given a split only the trainer can attest, and per-query access-freeness does not compose (Cohen, 2025).

The accounting tools we use (Feldman & Zrnic, 2020; Whitehouse et al., 2023; Rogers et al., 2016; Loynes, 1962; Cruz, 1991) and the extraction protocol (Carlini et al., 2021; 2023; Cooper et al., 2025) are standard.

Other defences. BloomScrub certifies takedowns by Bloom-filter screening with abstention (Zhang et al., 2025); SILO isolates legal risk in a nonparametric datastore (Min et al., 2024); others refuse (Liu et al., 2024) or randomise the decoder (Chen et al., 2024b); for diffusion models retrieval carries a NAF guarantee (Golatkar et al., 2024) in response to replication in image models (Somepalli et al., 2023). Alshehyari et al. (2026) bench-

mark output-, behavioural- and representation-level defences; the level their benchmark omits is the certified one, where a low measured leakage and a vacuous certificate can coexist. VA3 (Li et al., 2024) amplifies a probabilistic copyright guarantee for text-to-image models through repeated interaction: choosing among m samples raises a per-sample probability p to $1-(1-p)^m \le mp$, the attacker's side of Proposition 2.

Self-consistency. Self-consistency (Wang et al., 2023) serves the majority answer over n sampled paths; a mode is a score, so it satisfies $q(y) \le n\, p_s(y)$ and carries a pathwise certificate of exactly $\log n$ against the model it samples. As deployed it samples web-trained models, relative to which $\log n$ certifies nothing about copyright; over a safe model the same inequality is a copyright guarantee its users already have. The best-of-n literature prices $\log n$ as a *cost*, the budget spent before reward overoptimisation (Gao et al., 2023; Beirami et al., 2025) or the one an attacker spends to jailbreak (Hughes et al., 2024); we read it as a guarantee, compare it with a metered decoder on one divergence axis, and measure it. The mechanism is theirs. So is soft best-of-n (Verdun et al., 2025), which draws the served index from a softmax of the rewards and so interpolates between the anchor and best-of-n; it serves one of the draws, so Proposition 2 covers it at the same $\log n$.