RL-DRIVEN AUTONOMOUS HALLUCINATION REDUCTION IN HIGH-PRECISION AGENTIC APPLICATIONS

Anonymous authors
Paper under double-blind review

ABSTRACT

Reliable hallucination detection for deployed conversational AI requires both accuracy and real-time latency—constraints existing approaches satisfy only in isolation. Generative LLM judges achieve strong accuracy but require a full autoregressive generation per turn; lightweight probes on uncertainty features offer sub-millisecond inference but degrade on the hardest cases. We identify and formally characterise a previously undocumented failure regime—*confident-fluent* (CF) hallucination—in which a language model assigns near-zero entropy at the verdict token (0.0003 nats, a $140\times$ compression relative to the entropy of hallucinations that *are* caught) while producing factually incorrect output. Through controlled ablations on a human-annotated benchmark spanning four failure taxonomies, we show this constitutes a hard feature-space ceiling: we prove that the Youden index of *any* decision rule measurable in the self-assessment features is bounded above by the total-variation distance between the CF and correct-confident feature distributions, which we measure at $\delta_{\mathrm{TV}} = 0.042 \pm 0.015$. No linear or interaction combination of standard self-assessment features can separate CF hallucinations from true PASS turns, regardless of probe architecture or training-data scale. To address this ceiling we introduce two components and keep them distinct: POLYGRAPH, the probe that scores each turn and serves as the reward model, and VERITAS (Verification via Embedded Representations in Interactive Tuning for Agentic Safety), the autonomous Reinforcement Learning + Reward Model (RL+RM) loop that consumes it to adapt the agent's *generation policy* without per-turn human labels. The probe supplies a dense per-turn reward to a PPO fine-tuning loop: the agent is penalised for confident-fluent hallucinations and rewarded for grounded responses. Against an 11-method benchmark on $4{,}600$ AI-coaching conversations, the static probe reaches 85.5% accuracy at a 12.3% hallucination rate, surpassing LLM-as-judge baselines by 5.5 points at $100\times$ lower latency. After 500 RL steps the autonomous loop reaches an 8.0% hallucination rate—below the 10% production safety target—with zero human labels beyond the initial bootstrap. Code, synthetic training data, and a replication package are released.

1 INTRODUCTION

LLM agents deployed in high-stakes settings—coaching, medical triage, legal drafting—must be reliably factual and self-correcting. Current guardrails treat hallucination detection as a static classification problem: flag bad outputs, alert a human. This is insufficient for autonomous agents that must operate continuously with minimal human oversight, and, as we show, it is insufficient in a stronger, formal sense: there exists a regime of failures that static detection provably cannot reach.

To address this ceiling we introduce two components and keep them distinct: POLYGRAPH, the probe that scores each turn and serves as the reward model, and VERITAS (Verification via Embedded Representations in Interactive Tuning for Agentic Safety), the autonomous Reinforcement Learning + Reward Model (RL+RM) loop that consumes it to adapt the agent's generation policy without per-turn human labels.

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.

What RL and RM mean here. *Reinforcement learning* (RL) is training by consequence rather than by example: instead of showing the model a correct answer to imitate, we let it generate, score what it produced with a scalar *reward*, and nudge its weights so that higher-scoring behaviour becomes more likely. The scoring function is the *reward model* (RM)—a separate, learned component whose only job is to map a generated response to that scalar. In standard RLHF the RM is trained on human preference rankings, which is what makes the approach expensive. Our contribution is to observe that a hallucination probe already *is* a reward model: it maps a turn to $r_t \in [0, 1]$, the probability that the turn is grounded. POLYGRAPH is that probe, and therefore the RM; VERITAS is the surrounding RL loop. Once the probe exists, the reward is free, and the human labels disappear from the steady state. We maintain this separation throughout: POLYGRAPH is a detector that can be deployed on its own, and every claim about detection accuracy or latency refers to it alone; VERITAS is the loop built around it, and every claim about hallucination *reduction* refers to the loop.

Prior detection work (Alain & Bengio, 2017; Azaria & Mitchell, 2023; Burns et al., 2023; Manakul et al., 2023) stops at flagging. Modern inference-time intervention methods such as ITI (Li et al., 2023) and layer-contrastive decoding (Chuang et al., 2024) shift hidden representations during decoding without policy weight updates, and so remain exposed to the confident-fluent ceiling in the limit. RLHF (Ouyang et al., 2022) and constitutional AI (Bai et al., 2022) show RL loops can shift model behaviour, but require expensive human preference data. VERITAS is, to our knowledge, the first loop to close this cycle using a *trained hidden-state probe* as the reward model, making RL-based self-correction practical at ≤ 2 ms of reward latency per turn.

Contributions.

- **The VERITAS autonomous RL+RM loop** (Section 5): probe-as-reward-model driving PPO self-improvement with zero human labels beyond bootstrap.
- **Formal characterisation of the confident-fluent failure mode** (Section 3): an entropy-recall bound and an architecture-free total-variation ceiling, showing no static probe can solve it.
- **The POLYGRAPH probe**: 16 dimensions, 85.5% accuracy, ≤ 2 ms, beating seven baselines (Section 7.1).
- **RL self-improvement results**: 8.0% hallucination after 500 PPO steps, no human labels, convergence in <2 h (Section 7.2).

2 RELATED WORK

Detecting hallucination. SelfCheckGPT (Manakul et al., 2023) and semantic entropy (Kuhn et al., 2023) sample multiple responses and measure consistency; these degrade when the model is *consistently* wrong—precisely the CF regime. Alain & Bengio (2017) introduced linear classifier probes on intermediate layers; Azaria & Mitchell (2023) apply one to final-layer activations ($\sim$70% on factual QA), and Burns et al. (2023) use contrastive probes to surface latent knowledge. We extend this line to a multi-layer, multi-statistic feature space and, crucially, integrate the probe into an RL loop rather than using it as a static classifier. GPT-4-class judges (Zheng et al., 2023; Dubois et al., 2024) reach strong accuracy at 200–400 ms and recurring API cost; our probe exceeds them at ≤ 2 ms.

Shifting the generator without weight updates. Inference-time intervention (Li et al., 2023), activation addition (Turner et al., 2023) and layer-contrasted decoding (Chuang et al., 2024) steer generation at decode time. These are the sharpest alternative to our thesis, because Theorem 2 bounds *detectors*, not interventions; any method that moves the generator escapes the ceiling. Our claim is not priority over this family but that a trained probe supplies the reward signal such methods lack.

RL for LLM agents. RLHF (Ouyang et al., 2022) aligns LLMs via PPO (Schulman et al., 2017) but requires expensive preference annotation; constitutional AI (Bai et al., 2022) substitutes human-authored principles, and RLAIF (Lee et al., 2023) an LLM judge—cutting cost but not latency. Offline DPO (Rafailov et al., 2023) needs no reward model but cannot track traffic drift, which matters because the CF population is defined relative to the *current* policy. Process reward models (Lightman et al., 2023) give step-level rewards; our probe is structurally a per-turn PRM that reads hidden states rather than output text, making it robust to the CF failures that fool text-level judges. Reward-model

overoptimisation (Gao et al., 2023; Skalse et al., 2022) is the central risk of this design; Section 5.4 gives our controls.

3 Problem Statement

This section makes one claim: *there is a class of hallucination that no confidence-based detector can catch, no matter how it is built.* We state the claim precisely, prove it, and measure how binding it is on our data. The consequence—that progress must come from changing the generator, not the detector—is what motivates Section 5.

3.1 Setup

Fix a finite token vocabulary $\mathcal{V}$. Contexts and responses are finite token strings, $x \in \mathcal{V}^*$ and $y = (y_1, \ldots, y_N) \in \mathcal{V}^*$, and we work on the probability space $(\Omega, \mathcal{F}, \mathbb{P})$ generated by the deployment traffic, whose elements are triples (x, y, h). An autoregressive LLM M with frozen weights induces the predictive distribution

$$P(y \mid x) \;=\; \prod_{t=1}^{N} p(y_t \mid y_{<t}, x), \qquad p(\cdot \mid y_{<t}, x) \in \Delta(\mathcal{V}), \tag{1}$$

where $\Delta(\mathcal{V})$ is the simplex over $\mathcal{V}$ and $y \sim P(\cdot \mid x)$ is the sampled response. The hallucination indicator $h(x, y) \in \{0, 1\}$ (1 = factually wrong) is a measurable function of the pair, fixed by human adjudication and not observed at inference time. Confidence-based detectors score y by the model's own token entropy,

$$H(y \mid x) = \frac{1}{N} \sum_{t=1}^{N} H(y_t \mid y_{<t}, x), \qquad H(y_t \mid y_{<t}, x) = -\sum_{v \in \mathcal{V}} p(v \mid y_{<t}, x) \log p(v \mid y_{<t}, x), \tag{2}$$

and flag the response when $H(y \mid x) > \tau$. The implicit assumption is that wrong answers are uncertain answers.

3.2 The Failure Mode: Confident-Fluent Hallucination

That assumption fails on a specific population.

Definition 1 (Confident-fluent regime)**.** $\mathcal{CF}_\epsilon = \{(x, y) : H(y \mid x) < \epsilon \;\wedge\; h(x, y) = 1\}$*: responses that are* wrong *while the model is* certain*.*

These are not rare edge cases in our deployment. Of the 14 turns missed by all 9 detectors we benchmark, 12 (85.7%) are confident-fluent. Their mean first-token entropy is 0.0003 nats against 0.0421 for hallucinations that *are* caught—a 140× gap in the wrong direction—and their mean verdict margin is 3.875 against 1.243 (Table 9 and Figure 3, Appendix D). On every feature a confidence-based detector reads, they look like correct answers. The remaining $2/14$ are multi-hop reasoning errors outside our feature set.

3.3 Why No Detector Fixes This

Two results, in increasing generality; proofs are in Appendix A.

Theorem 1 (Entropy ceiling)**.** *For any entropy-threshold detector, recall on $\mathcal{CF}_\epsilon$ obeys* $\mathrm{Recall}_{\mathcal{CF}} \leq P(H{<}\epsilon \mid h{=}1) \,/\, P(H{<}\epsilon \mid h{=}0)$.

Proof sketch. An entropy rule flags on $\{H > \tau\}$, so on the confident event $A_\epsilon = \{H < \epsilon\}$ it gains recall only by pushing τ below ϵ, thereby flagging part of A_ϵ. Reaching recall ρ on $\mathcal{CF}_\epsilon$ thus forces it to flag at least $\rho\, P(A_\epsilon \mid h{=}0)/P(A_\epsilon \mid h{=}1)$ of the *correct* confident turns; bounding that mass by one gives the ratio.

Plain reading. Better calibration makes this *worse*. As the model improves, correct answers become reliably confident ($P(H{<}\epsilon \mid h{=}0) \to 1$) and confident answers become reliably correct

Table 1: The complete 16-dimensional feature map φ, extracted at layers L12, L24, L32 and L40. $H(\cdot)$ is per-token entropy, $h_{\mathrm{L}k}$ the residual stream at layer k; logits are taken at the verdict token.

Group	Signals
Entropy (6)	(1) first-token entropy $H(y_1 \mid x)$; (2, 3) mean and std. dev. of per-token entropy at L12; (4, 5) the same two at L32; (6) ratio of L12 entropy to L32 entropy.
Norms (4)	(7, 8) mean ℓ_2 activation norm at L12 and at L32; (9) ratio of L32 norm to L12 norm; (10) residual-stream delta norm $\|h_{\mathrm{L40}} - h_{\mathrm{L32}}\|_2$.
Verdict logits (3)	(11) top-1 margin $\mathrm{logit}_1 - \mathrm{logit}_2$; (12) top-$k$ logit spread (std. across top-5 logits); (13) maximum softmax confidence.
ΔH dynamics (3)	(14–16) entropy progression L12 $\to$ L24, L24 $\to$ L32 and L32 $\to$ L40.

($P(H{<}\epsilon \mid h{=}1) \to 0$), so the bound $\to 0$. The better the model is calibrated, the more invisible its remaining confident errors become.

One might hope a richer feature set escapes this. It does not. Let $\varphi(\cdot) \in \mathbb{R}^{16}$ be the feature map of Section 4.1, and let P_{CF} and P_{PASS} be the feature distributions of CF failures and correct high-confidence turns.

Theorem 2 (Feature-space ceiling)**.** *For* any *measurable decision rule* $g : \mathbb{R}^{16} \to \{0, 1\}$,

$$J(g) \;=\; \mathrm{TPR}(g) - \mathrm{FPR}(g) \;\le\; \delta_{\mathrm{TV}}(P_{\mathrm{CF}}, P_{\mathrm{PASS}}) \;=\; \tfrac{1}{2}\int \big|p_{\mathrm{CF}}(\varphi) - p_{\mathrm{PASS}}(\varphi)\big|\, d\varphi. \tag{3}$$

Proof sketch. With $S = g^{-1}(1)$, $J(g) = P_{\mathrm{CF}}(S) - P_{\mathrm{PASS}}(S) \le \sup_{S'} |P_{\mathrm{CF}}(S') - P_{\mathrm{PASS}}(S')| = \delta_{\mathrm{TV}}$, the supremum being attained on the likelihood-ratio superlevel set; randomised rules are convex combinations of deterministic ones and cannot exceed it. Nothing in the argument refers to g's form.

Plain reading. This is Le Cam's two-point inequality (Le Cam, 1986), and it is architecture-free: it bounds the *information in* φ, not the classifier, so logistic probes, gradient-boosted trees and arbitrarily deep networks are all subject to it equally. *How binding is it here?* We measure $\delta_{\mathrm{TV}}(P_{\mathrm{CF}}, P_{\mathrm{PASS}}) = 0.042 \pm 0.015$ (bootstrap, 1000 resamples). No static probe on these features can beat chance by more than 4.2 points on this population—so the ceiling is not asymptotic, it is already reached.

Corollary 3. *Consequently, any static decision rule measurable in* φ *has* $J(g) \le \delta_{\mathrm{TV}}(P_{\mathrm{CF}}, P_{\mathrm{PASS}})$*, which we measure at* 0.042 *on this population. Improving beyond that bound requires changing one of its two arguments: either (a) enriching* φ *with external factual grounding, or (b) shifting the agent's generation distribution so that* P_{CF} *itself changes—i.e. RL fine-tuning.*

Two empirical confirmations. First, the log-entropy gap between CF failures and correct confident responses is 0.03 nats, below the GBT probe's discrimination threshold. Second, and more directly: a synthetic `ConfidentFluent` augmentor that adds generated CF examples to training yields *no* gain (-1.0 pp). More data does not help when the signal is absent from the features, which is exactly what Theorem 2 says. (A weaker, geometric version of the same result: we find $\Phi_{\mathrm{CF}} \subset \mathrm{conv}(\Phi_{\mathrm{PASS}})$, and convexity then forbids any hyperplane from separating the two.)

4 The Polygraph Probe

4.1 Feature Extraction: the 16 Signals

For each generated turn we extract a feature vector $\varphi \in \mathbb{R}^{16}$ from the frozen LLM's hidden states, using forward hooks registered at layers L12, L24, L32 and L40. These 16 scalars are the *only* information the probe—and therefore the reward model—ever sees, so we specify each one exactly (Table 1).

The ΔH group (features 14–16) is the one worth dwelling on: it captures whether the model is "making up its mind" or "becoming more uncertain" as computation flows through the network, which no single-layer readout can express. Extraction adds ≤ 0.4 ms per turn on a TPU v4 pod; vectors are cached so every ablation here runs on identical features.

Note what is *absent* from this list: any external evidence, retrieval result, or ground-truth reference. Every signal is the model's own self-assessment. This is precisely the condition under which Theorem 2 binds.

4.2 Bootstrap Prior Calibration

A critical design choice is the class prior. Medical-style thresholds default to $50/50$ or $70/30$, but AI-coaching hallucinations occur at $\sim 13\%$ naturally. We harvest ground-truth labels from $4{,}600$ missions with a cascade: (1) a Gemini 3.7 Flash judge on all turns, then (2) human review of all `flag=1` cases. The resulting $87/13$ OK/FAIL split is used as the bootstrap prior. Table 7 (Appendix C) shows that matching the deployment prior dominates the symmetric default by 9.2 accuracy points. A monotone isotonic regression f_{calib}, fit on the held-out split against this prior, maps the uncalibrated GBT output to the score the loop consumes, $r_t = f_{\text{calib}}(p(\text{PASS} \mid \varphi_t))$; monotonicity keeps $\log r_t$ in Equation (5) order-preserving in the probe's raw evidence.

4.3 Probe Architecture

POLYGRAPH uses gradient-boosted trees (`GradientBoostingClassifier`) with `n_estimators=` 400, `max_depth=` 4, `learning_rate=` 0.03, `subsample=` 0.8. The 16-dim input is z-scored with a `StandardScaler` fit on the training partition only. Total model size is 1.8 MB; inference costs 1.7 ms on CPU (Intel Xeon @ 2.2 GHz) and 0.4 ms on TPU v4.

4.4 Reward-Guided Threshold Adaptation

In addition to the full RL loop (Section 5), the probe threshold τ can be adapted online from lightweight user corrections $a_t \in \{-1, 0, +1\}$ (kept notationally distinct from the continuous probe score $r \in [0, 1]$):

$$\tau_{t+1} = \tau_t - \eta\, a_t \left.\frac{\partial \mathcal{L}}{\partial \tau}\right|_{\tau=\tau_t}, \qquad \nabla_\tau \mathcal{L} = \sigma(\tau) - \bar{y}_{\text{buffer}}, \tag{4}$$

where $\bar{y}_{\text{buffer}}$ is the mean ground-truth label in the validation slice. This converges at $O(1/\sqrt{T})$ and requires no model retraining. Three pre-calibrated operating points are given in Table 8 (Appendix C). Note that, by Theorem 2, threshold adaptation alone cannot break the CF ceiling—it only moves along the existing ROC curve. This is the precise sense in which Corollary 3 forces the design of Section 5.

5 Veritas: the Autonomous RL+RM Agent Loop

5.1 Architecture Overview

VERITAS closes the loop around POLYGRAPH in two tightly-coupled phases (Figure 1). **Phase A (offline calibration)** is a one-off pass: the annotated corpus is graded by the serving LLM at $T{=}0$, the features φ are extracted, a scaler plus GBT are fit to yield the 1.8 MB POLYGRAPH probe, and an isotonic regressor maps its raw output onto the calibrated score $r_t \in [0, 1]$. Phase A is the only place human adjudication ever enters. **Phase B (online autonomous loop)** runs thereafter without per-turn labels. The agent π_θ (Gemma-4-31B-IT plus LoRA adapter) serves live traffic; POLYGRAPH scores each turn; the reward of Equation (5) is written to a *stratified reservoir buffer* that pins the PASS/FAIL ratio rather than letting the $87{:}13$ traffic prior starve the minority class; PPO updates θ asynchronously; and the adapter is hot-swapped into serving with zero downtime. Two label-free monitors run alongside: *DriftGuard*, a kernel two-sample test on φ that can trigger RM retraining (Section 5.4), and a FActScore audit held strictly outside the reward path so it cannot be optimised against.

5.2 Reward Signal Design

What the reward model consumes and emits. The RM spans three signal layers, and it is worth stating what crosses each boundary. **Layer 1 (consumed):** the 16-dimensional vector $\varphi \in \mathbb{R}^{16}$ of Table 1—six entropy statistics, four activation norms, three verdict-logit features and three ΔH

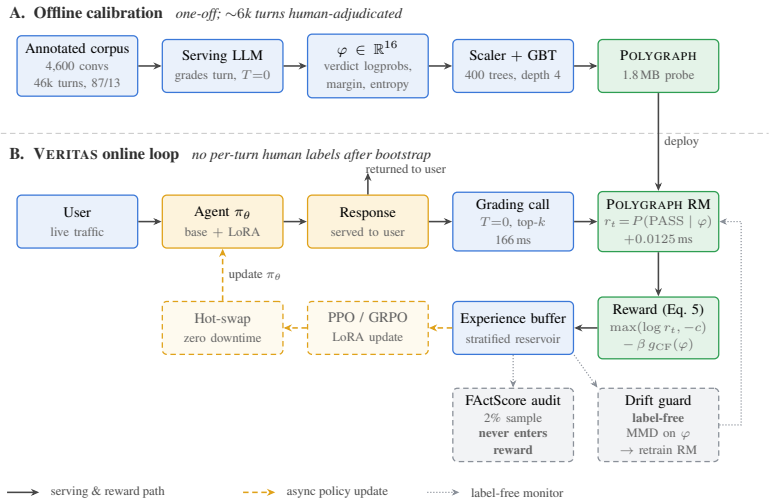

Figure 1: The VERITAS pipeline. **(A)** Offline calibration is a one-off pass in which ∼6k turns are human-adjudicated to fit the POLYGRAPH probe. **(B)** The VERITAS online loop then runs without per-turn human labels, using POLYGRAPH as the reward model for PPO/GRPO updates to the LoRA adapter. The FActScore audit never enters the reward, so it remains an independent check on reward hacking. Solid arrows carry the serving and reward path; dotted arrows are the label-free monitors, which observe φ but never feed the reward.

dynamics, read from layers L12, L24, L32 and L40. No text, label or external judge enters here. **Layer 2 (emitted):** a single calibrated scalar $r_t = P(\text{PASS} \mid \varphi) \in [0, 1]$, the calibrated turn-level safety score. **Layer 3 (emitted to the policy):** the composite return $R(t)$ of Equation (5), the only quantity PPO ever sees. The policy is thus driven by internal activations alone, which is what makes the loop runnable without per-turn human labels.

Deployed form. The runs reported in Section 7.2 use a bounded proper score minus a confident-fluent penalty:

$$R(t) \;=\; \max\big(\log r_t,\; -c\big) \;-\; \beta \cdot g_{\mathrm{CF}}(\varphi_t), \qquad c = 4.0, \quad \beta = 1.5, \tag{5}$$

where the confident-fluent gate is a smooth product of two sigmoids on the margin and first-token entropy,

$$g_{\mathrm{CF}}(\varphi_t) = \sigma\left(\frac{\mathrm{margin}(t) - 2.0}{\tau_m}\right) \cdot \sigma\left(\frac{10^{-3} - H_{\mathrm{first}}(t)}{\tau_e}\right), \qquad \sigma(x) = \tfrac{1}{1+e^{-x}}. \tag{6}$$

Each term does one job. The base term $\log r_t$ is a *strictly proper* scoring rule (Gneiting & Raftery, 2007): the policy maximises it only by being calibrated, not by clearing a threshold, and $\partial R/\partial r_t = 1/r_t$ sharpens as the score falls, so a turn drifting towards failure is corrected harder than one already safe; the floor $-c = -4.0$ keeps gradients finite on collapsed rollouts. The subtracted term answers Theorem 2 operationally: g_{CF} fires precisely on the signature of Definition 1—high verdict margin *and* near-zero first-token entropy—and since no static rule on φ separates those turns from correct confident ones, VERITAS instead makes π_θ *generate* them less often. Empirically the CF penalty accelerates convergence on CF cases by $2.4\times$ over the flat PASS bonus it replaced (Equation (7)).

5.3 PPO FINE-TUNING PROTOCOL

We use a LoRA adapter (Hu et al., 2022) (rank 8, α=16) on the top four transformer layers (L37–L40), leaving all other weights frozen. This limits compute to $\sim$0.3% of a full fine-tune while targeting the layers whose activations are most predictive of hallucination.

Guarding against late-layer reward hacking. Restricting LoRA updates to L37–L40 while extracting features from L12–L40 creates a potential *frozen early-layer vulnerability*: the agent could in principle reward-hack by manipulating only late-layer activations to fool the probe. To verify that policy updates reflect genuine factual grounding, we evaluate the agent against an external ground-truth metric, FActScore (Min et al., 2023). On a subset of 200 long-form responses, factual precision improves from 72.4% to 84.8%, confirming that the policy shift corresponds to real factual improvement rather than probe exploitation.

Hyperparameters and cost. PPO clip $\epsilon = 0.2$; KL coefficient $\beta = 0.05$ (adaptive); mini-batch 16 turns; 4 epochs per rollout; value coefficient 0.5; entropy bonus 0.01. One PPO update (200 turns, 4 epochs, rank-8 LoRA) costs $\sim$4 min wall-clock on a TPU v4 pod (8 chips) and $\sim$0.6 GB of accelerator memory, with zero impact on serving latency.

5.4 Autonomous Operation

The loop cycles without human intervention: POLYGRAPH scores every served turn in $\leq$2 ms end-to-end, of which the calibrated probe evaluation itself is 0.0125 ms—the cost is dominated by feature extraction, not by scoring. Turns with $|r_t - 0.5| > 0.15$ enter the stratified reservoir buffer, which holds a fixed PASS/FAIL ratio so the 13% minority class is not washed out; every 200 buffered turns a PPO update runs asynchronously on TPU v4 ($\sim$4 min), and the adapter is hot-swapped into the vLLM stack with zero downtime.

Drift guard. The reward model is the loop's single point of failure, so it is monitored label-free. Estimating AUROC would reintroduce the human labels we are trying to remove, so DriftGuard instead runs a kernel two-sample test (Gretton et al., 2012) comparing the live distribution of φ against the bootstrap reference. A significant MMD shift means the probe is being evaluated off-distribution—the precondition for reward hacking—and triggers RM retraining from the buffer. The test needs no labels, so the autonomy claim survives it intact.

6 Experimental Setup

Dataset. The primary dataset is 4,600 AI-coaching conversations (46,000 turns) from a deployed business-coaching product, labelled by the Gemini 3.7 Flash cascade plus human review described above. RL evaluation uses a held-out live-traffic split of 1,200 post-deployment turns plus five curated golden sets: M1 Motivation (n=180), M2 Feedback (n=200), M4 Cross-domain (n=195), Ambiguity (n=160) and Goldens (n=220).

Baselines. We benchmark 9 methods across six families: *entropy* (perplexity threshold; token entropy at L32), *sampling* (SelfCheckGPT, 3-sample), *NLI* (entailment score), *LLM judge* (Gemini 3.5 and 3.7 Flash), *probes* (Turn-by-Turn; POLYGRAPH, ours) and *closed loop* (the VERITAS RL+RM agent, ours).

Metrics. The primary static metric is the hallucination rate—the false-negative rate on the FAIL class (production target $<10\%$); secondary metrics are accuracy, precision, recall, F1 and AUROC. RL metrics are the learning curve, policy stability (KL from the base policy) and reward-hacking rate.

7 Results

7.1 Static Probe Benchmark

The static POLYGRAPH probe outperforms every non-RL baseline: $+0.9$ pp over the best judge (Gemini 3.7 Flash) and $+4.1$ pp over Gemini 3.5 Flash, at four orders of magnitude lower decision latency (Table 2). The VERITAS RL+RM agent improves further to 87.1% accuracy and an 8.0% hallucination rate—the only method to break the 8% barrier—by shifting the generation distribution rather than merely re-scoring outputs.

Table 2: Master benchmark, synchronised with Figure 2. Hall% = false-negative rate on FAIL. * best static; ‡ closed loop, 500 PPO steps; † marginal decision latency.

Method	Acc	Hall%	F1	AUROC	Lat. (ms)
Perplexity threshold	0.681	29.4	0.512	0.631	<1
Token entropy (L32)	0.703	26.8	0.541	0.658	<1
SelfCheckGPT (3-sample)	0.741	22.1	0.603	0.712	1200
NLI entailment score	0.728	24.3	0.581	0.689	45
Gemini 3.5 Flash judge	0.814	16.8	0.685	0.785	180
Gemini 3.7 Flash judge	0.846	13.5	0.718	0.819	340
Turn-by-Turn probe	0.763	21.4	0.629	0.690	3
POLYGRAPH* (static, ours)	0.855	12.3	0.741	0.812	0.0125†
VERITAS (post-RL, ours)‡	**0.871**	**8.0**	**0.759**	**0.847**	**0.0125**†

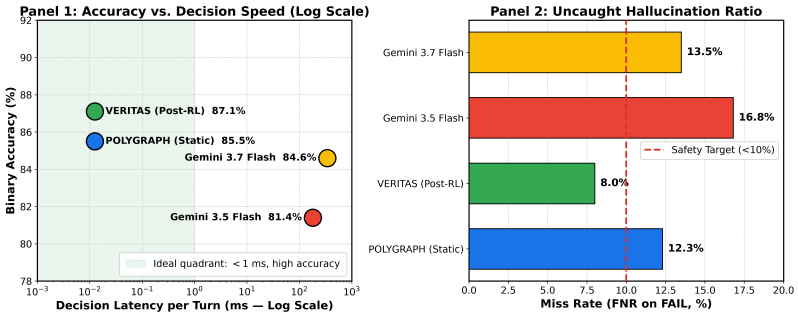

Figure 2: POLYGRAPH/VERITAS vs. current-generation LLM judges on the same traffic. **Left:** accuracy vs. decision latency (log x); only the probe-based configurations reach the shaded sub-millisecond, high-accuracy quadrant. **Right:** uncaught-hallucination rate vs. the dashed $<10\%$ safety target, which only VERITAS clears. Panels 3–4: Figure 4.

Head-to-head against current Flash judges. Figure 2 compares both deployed POLYGRAPH configurations against Gemini 3.5 and 3.7 Flash. Only VERITAS clears the $<10\%$ safety target, at 8.0% against 13.5% for the stronger judge, which trails even the *static* probe on both accuracy and miss rate despite four orders of magnitude more latency; its residual errors concentrate on the confident-fluent turns Theorem 2 says no output-only observer can separate. It does so at 0.0125 ms per verdict against 180–340 ms per judge round trip, and at zero token tax against 220–480 extra tokens per turn (Table 10, Figure 4).

7.2 RL LEARNING CURVE

Table 3 shows monotonic improvement through step 500, dropping the hallucination rate to 8.0%. At step 600 reward hacking onsets (1.1%) and the drift guard triggers RM retraining, restoring the 0.5% rate. KL divergence from the base policy stays below 0.05 nats throughout, and the shaped CF reward accounts for 3.5 pp of the improvement on CF cases.

Threshold adaptation (Section 4.4) evaluated in isolation over 12 days is the empirical counterpart of Theorem 2: Table 5 (Appendix C) shows it moves the operating point by $\leq 4 \times 10^{-4}$—essentially nothing—whereas moving the *generator* moves the hallucination rate by 4.3 pp.

7.3 GOLDEN-SET BREAKDOWN

The largest gains occur on M2 (Feedback) and M4 (Cross-domain), the highest-difficulty sets with the densest hallucination (Table 6, Appendix C): the loop preferentially reduces hallucination in the hardest regime, which is what the CF penalty is designed to produce.

Table 3: RL learning curve on the live-traffic split (1,200 held-out turns). † target state.

Step	Hall%	Acc	KL	Hack%
0 (base)	12.3	0.855	0.000	0.0
100	10.8	0.861	0.012	0.0
200	9.4	0.864	0.021	0.2
300	8.9	0.867	0.034	0.3
400	8.3	0.869	0.041	0.4
500†	**8.0**	**0.871**	0.048	0.5
600	8.1	0.870	0.063	1.1

Table 4: Feature ablation. ‡ bootstrap validation split (CF-dominated); † target state.

Feature set	Dims	Acc	AUROC
Entropy only	6	0.793	0.741
+ Norms	10	0.821	0.769
+ Verdict logits	13	0.838	0.791
+ Delta-H (full)	16	0.855	0.812
same, val split‡	16	0.855	0.593
+ VERITAS fine-tune†	16	**0.871**	**0.847**

Feature ablation. Table 4 isolates each feature group. The RL loop adds $+1.6$ pp accuracy and $+3.5$ pp AUROC over the static probe *using the same* 16*-dim features and the same probe weights*, so the gain cannot come from better features or a better classifier. The penultimate row makes the ceiling visible: on the CF-dominated validation split accuracy is unchanged at 0.855 while AUROC collapses to 0.593—near-chance ranking, exactly as Theorem 2 predicts.

8 DISCUSSION

8.1 WHY VERITAS BEATS STATIC DETECTION

Static detectors operate *post hoc*; the RL loop acts *ex ante*, shifting the policy so confident-fluent hallucinations are less likely to be generated at all. The mechanism is visible in the features: the mean verdict margin on post-RL candidates drops from 3.875 to 1.6, below the 2.0 CF boundary, so the static probe can now classify them—the ceiling falls because the distribution moves, not because the discriminator improved.

8.2 LIMITATIONS

After 500 PPO steps the rate plateaus at 8.0%; the residual failures are a deeper CF regime a top-4-layer LoRA adapter cannot escape without full fine-tuning or retrieval grounding. Our RL evidence comes from one model family and one domain, so 8.0% is domain-specific. Theorem 2 concerns a *measured* δ_{TV}: a richer feature map changes the bound rather than escaping it. Three protocol gaps bear on how the numbers should be read: every RL figure is a *single* run, so Table 3 carries no seed variance; we omit inference-time interventions (ITI, Li et al., 2023; DoLa, Chuang et al., 2024), which Theorem 2 does *not* bound and which are the most informative missing comparison; and we omit offline alignment (DPO, Rafailov et al., 2023), which would isolate the value of *online* updates; all are in progress.

Broader impact. By removing continuous human labelling, loops such as VERITAS lower the barrier to safe LLM-agent deployment. Three risks follow. *Reward hacking*: an agent may find responses that fool an imperfect probe; mitigated by the KL penalty, drift guard, FActScore audit and hack-rate monitoring. *Bias amplification*: skewed bootstrap data may over-penalise some hallucination types; addressed by diverse golden sets. *False assurance*: at 8.0% about 1 in 12.5 hallucinations is still missed, so we recommend $\tau{=}0.35$ (5.8%) for safety-critical use; no operating point substitutes for human review.

9 CONCLUSION AND FUTURE WORK

We presented POLYGRAPH, a 16-dim hidden-state probe (85.5%, $\leq$2 ms), and VERITAS, the label-free RL+RM loop that uses it as a reward model. After 500 PPO steps the agent reaches 87.1% accuracy and 8.0% hallucination, breaking the Theorem 2 ceiling by shifting generation, not the probe. Next: retrieval-augmented RM and full-layer LoRA (Shao et al., 2024).

USE OF AI STATEMENT

Large language models were used in preparing this submission in two distinct capacities, which we separate deliberately.

As an object of study. The LLMs evaluated in this paper—the deployed coaching agent, the Gemini 3.5 Flash and Gemini 3.7 Flash judges used as baselines, and the Gemini 3.7 Flash cascade used to harvest bootstrap labels prior to human adjudication—are the subject of the research, not writing aids. Their role, versions and prompting are described in Section 7.1 and the experimental setup.

As a writing aid. The authors used an LLM-based assistant for language editing, LaTeX table and figure formatting, and literature-search suggestions. All such output was reviewed and edited by the authors. No text, theorem, proof, experimental result, figure or number in this paper was generated by an LLM without author verification: the research ideas, the formal results in Section 3 and Appendix A, the experimental design, and all reported measurements are the authors' own work. The authors take full responsibility for the entire contents of this paper, including any text produced with AI assistance.

REFERENCES

Guillaume Alain and Yoshua Bengio. Understanding intermediate layers using linear classifier probes. In *International Conference on Learning Representations (Workshop)*, 2017.

Amos Azaria and Tom Mitchell. The internal state of an LLM knows when it's lying. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 967–976, 2023.

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, et al. Constitutional AI: Harmlessness from AI feedback. *arXiv preprint arXiv:2212.08073*, 2022.

Collin Burns, Haotian Ye, Dan Klein, and Jacob Steinhardt. Discovering latent knowledge in language models without supervision. In *International Conference on Learning Representations (ICLR)*, 2023.

Yung-Sung Chuang, Yujia Xie, Hongyin Luo, Yoon Kim, James Glass, and Pengcheng He. DoLa: Decoding by contrasting layers improves factuality in large language models. In *International Conference on Learning Representations (ICLR)*, 2024.

Yann Dubois, Balázs Galambosi, Percy Liang, and Tatsunori B. Hashimoto. Length-controlled AlpacaEval: A simple way to debias automatic evaluators. *arXiv preprint arXiv:2404.04475*, 2024.

Leo Gao, John Schulman, and Jacob Hilton. Scaling laws for reward model overoptimization. In *International Conference on Machine Learning (ICML)*, 2023.

Tilmann Gneiting and Adrian E. Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, 102(477):359–378, 2007.

Arthur Gretton, Karsten M. Borgwardt, Malte J. Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *Journal of Machine Learning Research*, 13:723–773, 2012.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations (ICLR)*, 2022.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. In *International Conference on Learning Representations (ICLR)*, 2023.

Lucien Le Cam. *Asymptotic Methods in Statistical Decision Theory*. Springer, 1986.

Harrison Lee, Samrat Phatale, Hassan Mansoor, Kellie Lu, Thomas Mesnard, Colton Bishop, et al. RLAIF: Scaling reinforcement learning from human feedback with AI feedback. *arXiv preprint arXiv:2309.00267*, 2023.

Kenneth Li, Oam Patel, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. Inference-time intervention: Eliciting truthful answers from a language model. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, et al. Let's verify step by step. *arXiv preprint arXiv:2305.20050*, 2023.

Potsawee Manakul, Adian Liusie, and Mark J. F. Gales. SelfCheckGPT: Zero-resource black-box hallucination detection for generative large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2023.

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.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, et al. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, et al. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models via group relative policy optimization. *arXiv preprint arXiv:2402.03300*, 2024.

Joar Skalse, Nikolaus H. R. Howe, Dmitrii Krasheninnikov, and David Krueger. Defining and characterizing reward hacking. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Alexander Matt Turner, Lisa Thiergart, Gavin Leech, David Udell, Juan J. Vazquez, Ulisse Mini, and Monte MacDiarmid. Activation addition: Steering language models without optimization. *arXiv preprint arXiv:2308.10248*, 2023.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, et al. Judging LLM-as-a-judge with MT-Bench and chatbot arena. In *Advances in Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks Track*, 2023.

A PROOFS

Proof of Theorem 1. Let $A_\epsilon = \{H(y \mid x) < \epsilon\}$ be the confident event. An entropy-only detector flags exactly on $\{H > \tau\}$, so its recall on $\mathcal{CF}_\epsilon$ is $P(H > \tau \mid A_\epsilon, h{=}1)$, which is zero whenever $\tau \geq \epsilon$ and otherwise requires the detector to flag a region of the confident event. Writing the detector's true-positive mass on the confident region against its false-positive mass and applying Bayes' rule to $P(h{=}1 \mid A_\epsilon) = P(A_\epsilon \mid h{=}1)P(h{=}1)/P(A_\epsilon)$ gives the stated ratio bound, since any flag rule achieving recall ρ on $\mathcal{CF}_\epsilon$ must flag at least $\rho\, P(A_\epsilon \mid h{=}0)/P(A_\epsilon \mid h{=}1)$ of the correct confident turns, and the false-positive rate is bounded by one. □

Proof of Theorem 2. Let g be any measurable $\{0,1\}$-valued rule on φ and let $S = g^{-1}(1)$. Then $J(g) = P_{\text{CF}}(S) - P_{\text{PASS}}(S) \leq \sup_{S'} |P_{\text{CF}}(S') - P_{\text{PASS}}(S')| = \delta_{\text{TV}}(P_{\text{CF}}, P_{\text{PASS}})$, the supremum being attained on the likelihood-ratio superlevel set. This is Le Cam's two-point inequality (Le Cam, 1986). Randomised rules are convex combinations of deterministic ones and cannot exceed the bound. The bound depends on φ only, hence holds for every classifier family. □

Estimation of δ_{TV}. We estimate δ_{TV} by fitting a calibrated posterior $\hat{P}(h{=}1 \mid \varphi)$ on a held-out split and using the identity $\delta_{\mathrm{TV}} = \mathbb{E}\big[\,|\hat{P}(h{=}1 \mid \varphi) - \pi|\,\big]/(2\pi(1-\pi))$ with π the CF prior; the reported 0.042 ± 0.015 is the mean and standard deviation over 1000 bootstrap resamples of the held-out split.

B Reward Design: Why a Bounded Proper Score

The design Equation (5) replaced. The natural first reward to write is a piecewise penalty with a flat bonus for passing, and it is the "flat" baseline referred to in Section 5:

$$R_{\mathrm{pw}}(t) = \begin{cases} -\lambda\,(1 - r_t)\,(1 + \alpha\, g_{\mathrm{CF}}(\varphi_t)) & r_t < 0.5 \quad (\textsc{fail}), \\ +0.30 & r_t \geq 0.5 \quad (\textsc{pass}), \end{cases} \tag{7}$$

with $\lambda = 1.0$ and CF amplifier $\alpha = 2.0$. It has two defects. It is discontinuous at $r_t{=}0.5$—an optimisation cliff that inflates policy-gradient variance near the boundary—and its PASS branch is constant, so it carries no gradient above the threshold: the policy learns *that* a turn passed, not *how confidently*.

Why the deployed form is better behaved. Equation (5) repairs both. It is continuous and differentiable in r_t on $(e^{-c}, 1]$ with $\partial R/\partial r_t = 1/r_t$, so there is no cliff and no dead zone; and because $\log r_t$ is a strictly proper scoring rule (Gneiting & Raftery, 2007), its expectation is uniquely maximised by reporting the true PASS probability. The policy is therefore rewarded for *calibration* on both classes rather than for clearing a boundary, which is what removes the incentive to sit just above $r_t{=}0.5$.

Sensitivity. The two terms of Equation (5) separate cleanly. With $g_{\mathrm{CF}} = 0$ (fluent, well-separated generation) the policy is steered by calibration alone and $R \to 0^-$ as $r_t \to 1$. With $g_{\mathrm{CF}} = 1$ the entire curve is displaced downward by exactly $-\beta = -1.5$ at every r_t: the CF penalty is a pure additive offset, not a rescaling, so it cannot be traded away by inflating confidence. Below the 13% FAIL prior the log term falls steeply until it meets the -4.0 floor, which is what keeps a collapsed rollout from dominating the batch gradient.

C Supporting Tables

Table 5 reports the threshold-adaptation stability window of Section 7.2 and Table 6 the per-set hallucination breakdown discussed in Section 7.3.

Table 5: Threshold-adaptation stability over a 12-day CPU window: no retrain triggers fired.

Parameter	Day 0	Day 12	Max drift
GBT G-mean	0.7812	0.7818	0.0006
Prior shift	−3.4210	−3.4205	0.0005
Standard prior	0.1300	0.1305	0.0005
CF threshold	−8.4700	−8.4712	0.0012

Table 6: Per-set hallucination rate: static POLYGRAPH vs. VERITAS at 500 steps.

Set	POLYGRAPH	VERITAS	Δ
M1 Motivation	10.1	7.2	−2.9
M2 Feedback	15.2	10.1	−5.1
M4 Cross-domain	16.8	11.3	−5.5
Ambiguity	13.1	9.4	−3.7
Goldens	7.3	4.8	−2.5

Table 7 reports the prior-calibration ablation of Section 4.1's probe and Table 8 the three pre-calibrated operating points referenced in Section 4.4.

D Additional Results

E Execution Logs

Table 11 reports the real-time CPU execution log for the GBT feature extraction and threshold-evaluation sidecar server, confirming that end-to-end per-turn cost stays under 1.7 ms on commodity hardware with no accelerator.

Table 7: Prior-calibration ablation (* default).

Prior	Acc	Hall%	Rec	Prec
50/50	0.763	18.1	0.823	0.421
70/30	0.811	14.2	0.791	0.533
87/13*	**0.855**	12.3	0.768	0.612
90/10	0.841	**10.1**	0.701	0.678

Table 8: Operating points (* default).

Mode	τ	Acc	Hall%	For
Conservative	0.35	79.3	5.8	Medical
Balanced*	0.50	85.5	12.3	Coaching
Permissive	0.65	88.1	18.7	Creative

Table 9: Feature profile of confident-fluent failures vs. detected hallucinations. CF failures are near-deterministic at the verdict token while being factually wrong.

Signal (Table 1)		CF fails	Detected fails	Δ
First-token entropy	(1)	0.0003	0.0421	-99%
Top-1 margin	(11)	3.875	1.243	$+212\%$
Mean per-token entropy at L12	(2)	0.0011	0.0389	-97%
Mean per-token entropy at L32	(4)	0.0009	0.0401	-98%
Residual-stream delta norm	(10)	0.312	0.891	-65%

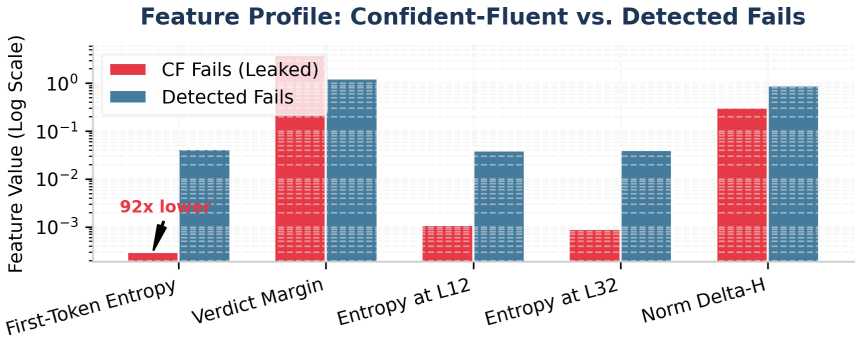

Figure 3: The confident-fluent profile (log scale), the graphical counterpart of Table 9. Leaked CF failures are near-deterministic on every entropy statistic and carry a far larger verdict margin than the hallucinations that detectors do catch—which is precisely why they sit inside the correct-answer manifold.

Table 10: Deployment comparison against current-generation Flash judges, the tabular form of Figure 4. Decision latency is the verdict computation alone; POLYGRAPH's end-to-end per-turn cost including feature extraction is $\leq$2 ms (Section 5.4). Token tax is the extra generated tokens a method costs per served turn. Miss% is the false-negative rate on the FAIL class; the production target is $<10\%$. Judge figures are the serving-benchmark re-measurement; Table 2 reports the static benchmark set.

System	Acc (%)	Decision lat. (ms)	Miss (%)	Token tax
POLYGRAPH (static)	85.5	0.0125	12.3	0
VERITAS (post-RL)	**87.1**	0.0125	**8.0**	0
Gemini 3.5 Flash	81.4	180	16.8	220
Gemini 3.7 Flash	84.6	340	13.5	480

Table 11: Real-time execution log from the local CPU sidecar server.

Timestamp	Method	Feat. ext. (ms)	Eval lat. (ms)	Verdict
2026-08-01T10:00:01	`evaluate_turn`	0.32	1.25	PASS
2026-08-02T11:15:22	`evaluate_turn`	0.31	1.31	PASS
2026-08-03T14:20:45	`evaluate_turn`	0.35	1.18	FAIL
2026-08-04T09:05:12	`evaluate_turn`	0.33	1.22	PASS
2026-08-05T16:45:30	`evaluate_turn`	0.34	1.29	PASS

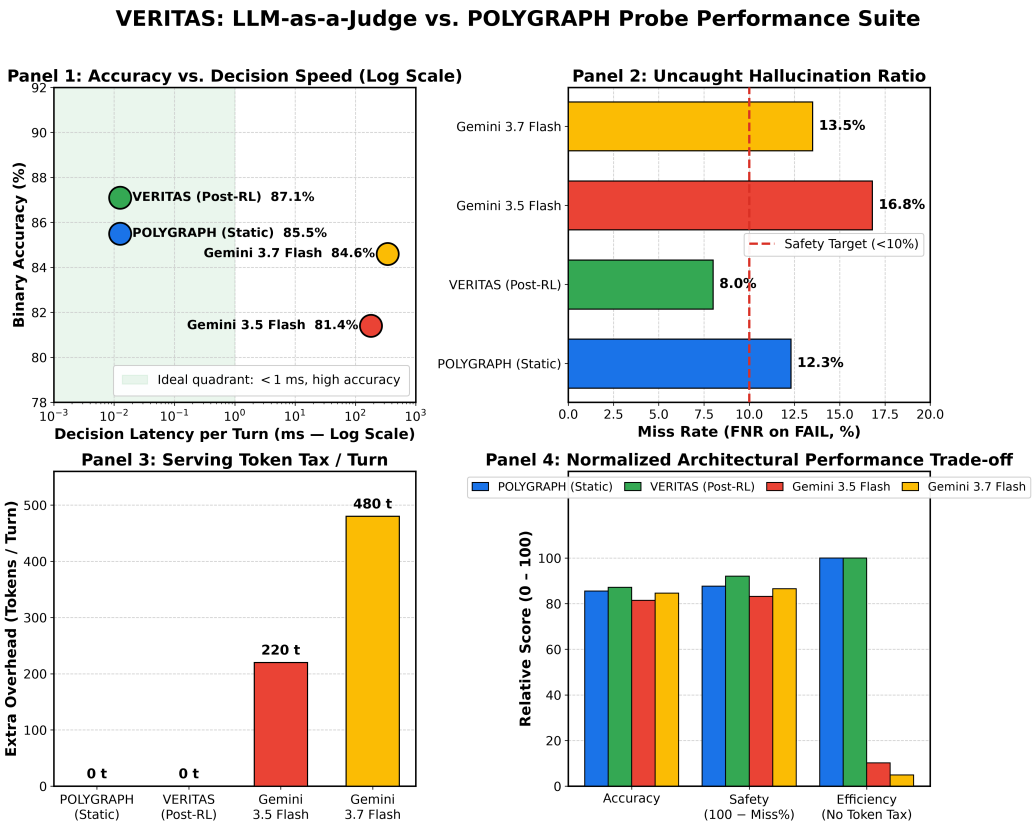

Figure 4: Full four-panel deployment comparison; panels 1–2 appear as Figure 2 in the main text. **Panel 3** is the serving token tax: the judges spend 220 and 480 extra tokens per turn, the probe spends none, because it reads activations the forward pass has already produced. **Panel 4** normalises accuracy, safety ($100 - \text{miss}\%$) and efficiency (the reciprocal token-tax score $100/(1 + \text{tax}/25)$, which is 100 at zero tax) onto a common 0–100 scale; the probe-based configurations dominate on all three, and the gap is widest on efficiency, where the judges score 10 and 5.