CAVEAT: A DETERMINISTIC STRUCTURAL VETO RATHER THAN A SECOND MODEL VIEW LOWERS GUI-AGENT VERIFIER FALSE POSITIVES, AT A RECALL COST

Anonymous authors
Paper under double-blind review

ABSTRACT

Models that read a recorded GUI-agent trajectory and predict whether the task succeeded tend to accept the agent's own narration as evidence. We find that what lowers such a verifier's false positives is a deterministic structural check of the record rather than a second model view, and that the reduction is paid for in recall. CAVEAT (Counterfactual Agent Verification via Evidence-only Auditing of Trajectories) is a selective verifier that reports success only when two reasoning-free views agree and a structural checker finds no contradiction in the record. Because neither view reads the agent's reasoning, rewrites confined to reasoning cannot change a verdict; we prove this rather than measure it. On the expert-false test records the view that reads the agent's answer reports 67.7% as successes and the full rule 22.6%, at 50.0% recall, and the structural check accounts for 78.6% of the drop; the best recorded judge gated by the same check reaches a false-positive rate a paired test does not separate from the rule's. On the untouched test split the rule's false-positive rate is below that of every recorded LLM judge, critic, reward model and single view we measured, and its predeclared risk certificate fails calibration, so the deployed policy abstains rather than overclaim. Part of the remaining gap is coverage, and whether the check selects by expert label is unresolved on this sample. The certificate fails on both counts, a sample below the size the target needs and an error rate no attainable sample would certify. The check that lowers false positives is deterministic, so anyone holding the record can run it.

1 INTRODUCTION

Web and GUI agents are commonly evaluated by a model that reads screenshots, accessibility trees, actions and the agent's own explanation, then predicts task success Lù et al. (2025); Xue et al. (2025). AgentRewardBench reports substantial false-positive rates across judge configurations, and targeted rewrites of an otherwise fixed trajectory can make observational judges accept failed executions Lù et al. (2025); Khalifa et al. (2026); better prompts and larger models do not close this input channel.

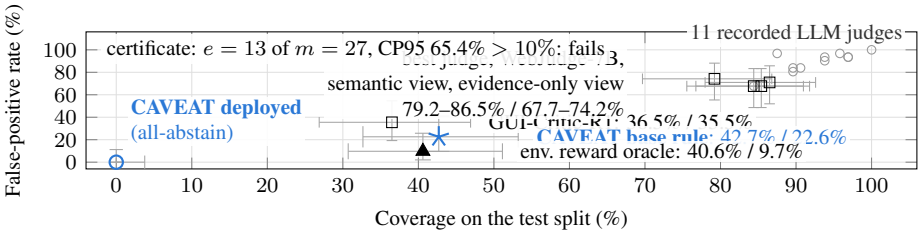

Figure 1: Coverage vs. false-positive rate on the untouched test split (96 rows), with exact two-sided 95% Clopper–Pearson whiskers on every Table 1 row. Recorded LLM judges (grey) sit at 84.4%–100.0% coverage, 67.7% FPR at best; the deployed rule sits at the origin because the certificate failed. Generated from released artifacts.

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.

A verification record proves that code ran and returned a verdict, not that the verdict agrees with the task semantics: our own pilot holds a live counterfactual that confirms a page-state dependency the experts reject. We therefore separate *structural robustness*, provable from which inputs a rule reads, from *semantic risk*, estimable only against held-out labels under an explicit sampling unit.

We present CAVEAT (Counterfactual Agent Verification via Evidence-only Auditing of Trajectories), a three-layer verifier. A live intervention layer restores WebArena/VisualWebArena snapshots and executes step ablation and state mutation; replay and evidence-occlusion checks run offline. Here that layer is a 30-trajectory design pilot rather than a measurement. An audit layer applies a frozen rule to recorded trajectories: a semantic view checks the agent's answer, an evidence-only view withholds it, both omit chain-of-thought, and a deterministic record checker can veto a report. A calibration gate enables the rule only if an exact finite-sample upper bound certifies a predeclared selective risk.

Two scoped guarantees follow. Because agent reasoning is absent from every report-rule input, changing only that reasoning cannot change a CAVEAT verdict under batch-invariant decoding (Theorem 1). Because calibration uses a fixed rule and one preselected representative per task, a one-sided exact binomial bound certifies the task-level selective risk, and a failed certificate disables reporting (Theorem 2). The confirmatory protocol is kept apart from the design pilot (§4): whole tasks go to splits, expert labels join after generation is frozen, and one hash-fixed attempt per task enters the certificate. The frozen 30B auditor is also distilled into a Qwen2.5-VL-7B LoRA student whose certificate is computed on the test split its training never saw (§3.4).

Contributions. (i) An architecture whose executed counterfactuals are a mechanism pilot (30 trajectories; §4, Appendix G) and whose answer-withheld two-view audit with a structural veto is the measured layer (Table 1, Figure 1); the two were never combined on the holdout, their only joint evaluation being the pilot's failed preview gate. (ii) An exact narrative-rewrite invariance theorem for the reasoning channel of the gaming-the-judge threat model (§3.2), proved, not measured. (iii) A selective-risk certificate, the standard binomial-tail bound (Geifman & El-Yaniv, 2017; Angelopoulos et al., 2022) applied at a task-level unit with abstain-on-failure deployment, over one representative per task (§3.3, Figure 4). (iv) A pilot-disjoint, label-blind evaluation against recorded AgentRewardBench judges, GUI-Critic-R1 and WebJudge-7B on 96 test rows (Table 1), charging every removed false positive to the component that removed it (§6, Figure 3), plus a task-cluster noise floor (Table 3). (v) A paired study of four gaming-the-judge attacks on the 30-record pilot, not on the holdout (§5, Figure 6), and a 7B audit-supervised student evaluated without label leakage (§5). The deterministic checker, not the second model view, does most of the work, and a pre-registered certificate still refuses to deploy the rule.

2 Related Work

Environments, judges, critics and reward models. WebArena Zhou et al. (2024), Mind2Web Deng et al. (2023) and OSWorld Xie et al. (2024) provide reproducible tasks with scripted final-state oracles; we use them as the execution substrate for interventions rather than as the evaluation endpoint. AgentRewardBench Lù et al. (2025) audited twelve trajectory judges and found the best below 70% precision, with roughly 30% of failed trajectories marked successful; WebJudge Xue et al. (2025) raised precision on Online-Mind2Web but still reads only the record. Environment-aware judging Shi et al. (2026) and tool-agent judge audits Gurram (2026) extend the paradigm; judges are gameable, with unfaithful chain-of-thought pushing false-positive rates to 90% Khalifa et al. (2026), and reported scores can exceed what the evidence supports Gao & Zhou (2026); outcome-only judges miss silent faults that step rubrics catch Mohammadi (2026), pre-action verification checks a step before it runs Althoubi (2026), and evidence-grounded verification anchors agent safety testing in recorded evidence Feng et al. (2026). Process reward models and critics score steps or visual states, not outcomes Chae et al. (2025); Zhang et al. (2026b); Qian (2026); Li et al. (2026); Song et al. (2026), measured under the CUARewardBench protocol Lin et al. (2025), as does the pre-operative critic GUI-Critic-R1 Wanyan et al. (2025); all remain observational scorers with the same false-positive exposure. Demanding evidence over narration is not GUI-specific: video verification scores a claim only against the segments it depends on Xiao et al. (2026f;d), code agents predict risk before an action runs Xiao et al. (2026c), and our fixed intersection rule is one point in

a surveyed space of training-free channel combination Xu et al. (2026b;a). None reports which part of a rule removes which false positives (§6).

Attribution, intervention, and verification with guarantees. Failure attribution asks why a trajectory failed Zhang et al. (2025b;a); Zhu et al. (2025); Deshpande et al. (2025), with replay and Shapley-style estimators Shah (2026); Bonagiri et al. (2026) or minimal repair of the suspect explanation Xiao & Li (2026); it diagnoses failure rather than certifying success. Verifiers acting on the environment are closest to our live layer: CIVeX Rovai (2026) verifies language agents by causal intervention, agentic reward modelling interacts online Cui et al. (2026), HANSEL Zhang & Nam (2026) re-executes extracted breadcrumbs, DoVer Ma et al. (2025) injects interventions to debug multi-agent runs, Evidence Over Plans Zhou et al. (2026) couples online verification with distillation, and a typed, effect-annotated route makes an artifact safe before it runs Xiao et al. (2026a). All of these re-execute or intervene to test a claim; none audits a recorded trajectory without re-execution, attaches a task-level selective-risk certificate (Geifman & El-Yaniv, 2017; Angelopoulos et al., 2022), or bounds what a reasoning rewrite can change, which is the layer measured here. Statistical reliability for verifiers is newer: E-valuator Sadhuka et al. (2025) wraps verification in sequential testing, computer-use verifier design is surveyed by Rosset et al. (2026), and calibrated selective gates risk-control LLM labelling outside agent evaluation Xiao et al. (2026b), whereas gates thresholded by evolutionary search or adaptive anomaly detection state no finite-sample bound Xiao et al. (2026e; 2025). Step-level diagnosis benchmarks Hong et al. (2026); Fan et al. (2026); Chung et al. (2026); Zhai et al. (2026) and evidence-carrying agents Zhang et al. (2026a) measure process quality rather than certifying reports. CAVEAT adds a reasoning-free two-view audit with a deterministic veto, an abstain-on-failure certificate, and measures what each component removes.

3 METHOD

CAVEAT separates causal evidence collection from semantic risk control: live operators execute interventions on a restorable environment, the auditor evaluates two views of a fixed record, and the deployment gate uses held-out labels to decide whether it may report.

3.1 TYPED CLAIMS AND LIVE COUNTERFACTUAL OPERATORS

A trajectory $\tau = (s_1, a_1, \ldots, s_T, a_T)$ carries a task goal, actions, screenshots, accessibility trees and optional agent reasoning and terminal answer. A claim pairs a proposition (task success, step causality, evidence grounding) with recorded-evidence pointers and an intervention specification. Four operators on a snapshot-replayable environment return CONFIRMED, REFUTED or UNVERIFIABLE with their inputs and observed deltas: *replay* re-executes recorded actions from the restored snapshot; *step ablation* removes a credited action, a persisting effect refuting its necessity; *evidence occlusion* hides cited UI evidence; *state mutation* changes a whitelisted backend value and checks whether the reported fact tracks it. One refutation vetoes a pilot report; unavailable or ambiguous executions abstain. In the design pilot, ablation and mutation ran against restored services, replay and occlusion as deterministic record-level heuristics (Appendix F); mutation returned no decisive verdict (§4). A confirmation record is an audit invariant rather than semantic correctness.

3.2 REASONING-FREE MULTIVIEW AUDIT

For the confirmatory cohort we freeze two auditor views. The *semantic* view receives the goal, recorded actions, sampled screenshots and terminal answer, and checks every task constraint. The *evidence-only* view sees the same goal, actions and screenshots but withholds both the separate terminal answer and terminal-message action payloads. Agent chain-of-thought is absent from both views (Figure 2). A deterministic record checker supplies a hard structural verdict $h(\tau) \in \{\text{CONFIRMED}, \text{REFUTED}, \text{UNVERIFIABLE}\}$ from four record predicates on the actions, their observations and the terminal message (loop tail, terminal execution error, missing action target, answer ungrounded in every observation; Appendix E); it reads the terminal answer only to ground it, and nothing reads the chain-of-thought. Let $v_s, v_e \in \{\text{SUCCESS}, \text{FAILURE}, \text{INSUFFICIENT}\}$ be the two frozen auditor verdicts. The base rule is

$$g(\tau) = \mathbb{1}[v_s = \text{SUCCESS} \ \wedge \ v_e = \text{SUCCESS} \ \wedge \ h(\tau) \neq \text{REFUTED}]. \tag{1}$$

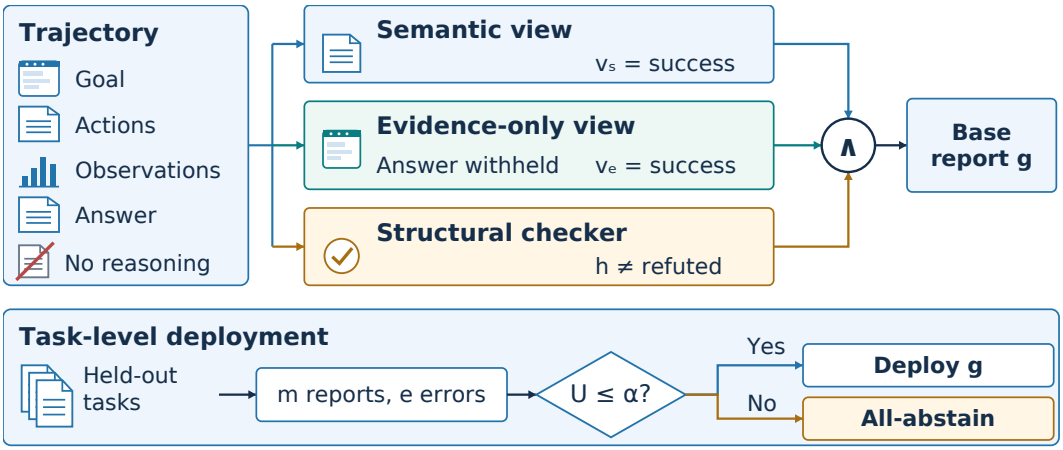

Figure 2: What each consumer of a recorded trajectory reads. The semantic view reads the goal, the actions with sampled screenshots and the terminal answer; the evidence-only view reads the same record with the terminal answer withheld; the checker h reads actions and screenshots, touches the answer only to ground it against the observations, and never reads the goal. No consumer reads the chain-of-thought (Theorem 1); every implemented attack rewrites the terminal message. A report needs all three answers; a refuted h is the veto. Bottom: the deployment gate, decided on one representative per task from the certificate split alone; here $e = 13$ of $m = 27$, so it abstains.

Theorem 1 (Narrative-rewrite invariance)**.** *Let A be any transformation that changes only the agent-reasoning fields of τ while preserving the goal, actions, observations, and terminal answer. If auditor decoding is deterministic and batch-invariant (a prompt's output does not depend on the other prompts in its batch), then $g(A(\tau)) = g(\tau)$.*

Both prompts and h are deterministic functions of the preserved fields, so they are byte-identical before and after A (Appendix A); the exclusion is unit-asserted.

The theorem covers chain-of-thought rewrites, not edits to actions, screenshots, the task or terminal answer. All four attacks of §4 also rewrite the terminal message, a field the rule reads, so their effect on g is outside this theorem, and measured in §5.

3.3 Task-level selective-risk certificate

Repeated attempts on one task share semantics and observations, so they are not independent calibration trials. Before labels are joined we hash-select one representative trajectory per task of a designated *certificate split*: the calibration split for the pilot-frozen rule and, for the distilled student of §3.4, the test split, the only labelled split disjoint from its training. Let m be the number of representatives with $g = 1$ and e the number of those reports not unanimously labelled successful by the experts; mixed or unsure labels count as errors. For target risk α and failure probability δ, define $U(e, m, \delta) = \mathrm{Beta}^{-1}(1 - \delta; e + 1, m - e)$, with no certificate when $m = 0$. The deployed rule is $g_{\mathrm{dep}} = g$ if $U \le \alpha$ and the all-abstain rule otherwise.

Theorem 2 (Abstain-on-failure selective-risk control)**.** *Assume the hash-selected tasks of the certificate split and the deployment tasks are i.i.d. draws from one task distribution and that the reported rule is fixed without using that split's labels. With probability at least $1 - \delta$, either g_{dep} abstains everywhere or its task-level selective risk $\Pr(Y \neq \text{successful} \mid g_{\mathrm{dep}} = 1)$ is at most α.*

Given the fixed rule and m, e is binomial at the selective risk (proof: Appendix A).

The bound is minimised at zero errors, where $U(0, m, \delta) = 1 - \delta^{1/m}$, so $\alpha = 0.10$ at $\delta = 0.05$ is reachable only from $m \ge \lceil \log \delta / \log(1 - \alpha) \rceil = 29$ reports, the pre-registered target the splits were sized for. Attainability is settled before a label is read; failing below it is about m, not the rule's error rate.

3.4 AUDIT-SUPERVISED 7B STUDENT

The frozen Qwen3-VL-30B-A3B auditor labels both views of the teacher-train and calibration tasks, and a Qwen2.5-VL-7B LoRA adapter learns the same verdict schema. Neither calibration nor test expert labels enter training. At evaluation the student reuses the rule, veto and certificate; because its training consumed calibration-split inputs, its certificate is computed on the test-split representatives it never saw, which also carry its reported metrics, so those metrics are in-sample for its deployment decision. On the rows the teacher rule reports, let ε be the teacher's false-report rate, δ_c the fraction where the student disagrees, and FR_S the student's false-report rate; $\mathrm{FR}_S \leq \varepsilon + \delta_c$ is a counting identity, checked in §5, and not a weak-to-strong result: the student is the smaller model.

4 EXPERIMENTAL SETUP

Substrates and design pilot. The live substrate is WebArena/VisualWebArena Zhou et al. (2024), six services as rootless containers restored before every state-bearing run; OpenStreetMap has no hermetic backend and is not replayable. Our 30-trajectory M1 cohort is a *design pilot* on which the operators, prompts, report rule and failure taxonomy were built. Two operators were executed live: the stage reached 22 of 30 trajectories and returned 3 confirmed, 7 refuted and 12 unverifiable step-ablation verdicts, and state mutation returned no decisive verdict: all 22 attempts were unverifiable (Appendix G). Replay and evidence-occlusion checks ran offline as deterministic heuristics (Appendix F) rather than as executed interventions. A pilot claim that step ablation confirmed disagreed with unanimous expert labels (§6), falsifying the idea that an executed confirmation implies semantic truth and motivating the explicit risk certificate.

Label-blind confirmatory cohort. We use AgentRewardBench Lù et al. (2025), pinned at revision `b6d17e6`. Cohort construction takes the WebArena and VisualWebArena trajectories the public `gpt-4o-noscreen` judge accepts, excludes every pilot task, and assigns whole task groups by salted hash to teacher-train, calibration and test splits of 48/96/96 trajectories. Expert annotations join only after the cohort, splits and one certificate representative per task are frozen. No CAVEAT threshold, prompt or deployment decision touches the test split; the one exception, the recorded-judge envelope row of Table 1, favours the baseline family and is disclosed in §5.

Models and baselines. The frozen multiview auditor is Qwen3-VL-30B-A3B-Instruct at commit `9c4b90e`; the student is Qwen2.5-VL-7B-Instruct at `cc59489` with rank-16 LoRA. We compare against every recorded public LLM judge whose verdicts our pinned parser maps on all 240 cohort trajectories (11 of 15 configurations, plus the environment reward; aer, aerv, nnetnav carry no verdict tag and are recorded incomplete rather than dropped; Table 4), the primary judge, the recorded environment reward (an in-benchmark oracle, for reference), the two single views, WebJudge-7B Xue et al. (2025), and GUI-Critic-R1 Wanyan et al. (2025) with the authors' prompt and a last-three-action outcome adapter (a pre-operative step critic, so this row is an adaptation). Text-only AgentRM checkpoints are not image-aware GUI baselines and are omitted.

Research questions and metrics. **RQ1, selective reliability:** precision, false-positive rate, coverage and task-level selective risk of the frozen rule on the untouched test split, and whether calibration certifies $\alpha = 0.10$ at 95% confidence; if not, deployed coverage is zero by protocol. Recorded binary judges are compared at native coverage: their public artifacts carry no thresholdable per-example score. **RQ2, view and operator contributions:** the two views, their intersection, the structural veto and the live pilot operators, with every removed false positive charged to the component that removed it. **RQ3, audit distillation:** the 7B student's teacher-view fidelity, human-label metrics, certificate, latency and coverage. **RQ4, gaming robustness:** following the threat model of Khalifa et al. (2026), four fixed-template attacks (verbosity padding, confidence–authority assertion, evidence stuffing and instruction echo) hold environment-recorded fields fixed while each rewrites the terminal message and appends an unfaithful-confidence line to step reasoning; we report false-positive and flip rates for the observational re-judge and the gate. No implemented attack changes reasoning only, so this study does not exercise Theorem 1, whose batch-invariance assumption we did not verify for our batched vLLM decoding; it runs on the pilot pool, not the holdout.

Table 1: The structural checker sets the false-positive rate: the base rule reads 22.6% against 67.7% for the best recorded judge; that judge gated by the same checker reads 19.4% at 54.8% recall, not separated from the rule (McNemar $p = 1.000$); the deployed rule abstains (certificate failed). Brackets: exact two-sided 95% Clopper–Pearson intervals; one preselected representative per task for the certificate. Precision and FPR count the label-consistent reported rows; under the certificate's convention (§3.3, mixed labels are errors) the base rule reads 75.6%/29.4% and row 2 71.6%/67.6% (App. H). Bold: best measured per column. †: tautologies of the accept-set conditioning. ‡: selected on the test split (§4), the recorded family's best case. Recall is over the 62 expert-true rows (95% CI: base rule [37.0, 63.0], row 2 [84.2, 98.3]). "n/a": no certificate; "descriptive": the ungated rule; "—": the row reports on nothing.

Method	Precision ↑	FPR ↓	Recall ↑	Coverage ↑	Certified?
Primary public judge	66.7% [56.1, 76.1]	100.0%†	100.0%†	100.0%†	n/a
Best recorded LLM judge‡	73.4% [62.3, 82.7]	67.7% [48.6, 83.3]	**93.5%**	84.4% [75.5, 91.0]	n/a
GUI-Critic-R1 adapter	68.6% [50.7, 83.1]	35.5% [19.2, 54.6]	38.7%	36.5% [26.9, 46.9]	n/a
WebJudge-7B agentic RM	68.5% [56.6, 78.9]	74.2% [55.4, 88.1]	80.6%	79.2% [69.7, 86.8]	n/a
30B semantic view	73.4% [62.3, 82.7]	67.7% [48.6, 83.3]	**93.5%**	85.4% [76.7, 91.8]	n/a
Evidence-only view	72.5% [61.4, 81.9]	71.0% [52.0, 85.8]	**93.5%**	**86.5%** [78.0, 92.6]	n/a
Structural checker alone	77.1% [62.6, 88.0]	35.5% [19.2, 54.7]	59.7%	53.1%	n/a
Best recorded judge + checker	**85.0%** [70.1, 94.3]	**19.4%** [7.4, 37.5]	54.8%	43.8%	n/a
CAVEAT base rule	81.6% [65.7, 92.3]	22.6% [9.6, 41.1]	50.0%	42.7% [32.7, 53.2]	descriptive
CAVEAT deployed	—	0.0% [0.0, 11.2]	0.0%	0.0% [0.0, 3.8]	no

5 RESULTS

This section draws only on the confirmatory single-writer artifacts; the 30-row pilot output never enters the holdout estimates.

5.1 RQ1: SELECTIVE RELIABILITY

Calibration produces $e = 13$ errors among $m = 27$ reported task representatives, giving the 95% upper bound 65.4% against the predeclared 0.10 target, so the deployed row is all-abstain, untouched by test labels. The failure lies in the error rate rather than the report count: at 48.1% realized risk (95% CI [28.7, 68.1]) no attainable m certifies 0.10, and the smallest target the rule certifies at the observed (m, e) is 65.4%. The count also fell short (27 of 60 representatives against the 29 the target needs; floor 10.5% at $e = 0$), and that alone would have decided it, since the floor already exceeds the target (Figure 4). The two failures are independent: the error rate fails at any m, the count at any e.

Row 2, claude-3.7-sonnet-noaxtree, is the minimum-FPR configuration (precision as tie-break) among those parsing on every cohort row, chosen *on the test split*, so it is the recorded family's best case against us; the calibration split selects gpt-4o-mini-noaxtree (test FPR 93.5%); the envelope is not one over abstainers (Figure 1). The recorded environment reward (92.3% precision, 9.7% FPR, 40.6% coverage) reads the benchmark's own instrumentation: an oracle reference rather than a judge. Against row 2, CAVEAT's advantage comes from the false-positive rate rather than precision; the inputs differ, since h reads the accessibility tree and the execution-error field while row 2 is the no-tree judge configuration, so the information-matched comparison is the judge+h row. Both rules see every expert-unanimous-failure test row, so the comparison can be paired: exact McNemar on the 3 versus 17 discordant rows gives $p = 0.0026$, while the 81.6% versus 73.4% precision gap is within sampling error (Fisher exact $p = 0.366$). We claim only the first. Rows of one task are not independent (56 tasks), so the test is repeated at task level: the cluster-bootstrap FPR difference is -45.2 pp (95% CI [-70.0, -19.4]), the exact sign-flip test gives $p = 0.0019$, and against the judge the calibration split selects (gpt-4o-mini-noaxtree) the difference is -71.0 pp ([-89.7, -48.0]), so choosing row 2 on the test split works against the rule (App. H). **RQ1 takeaway: the audit rule lowers the false-positive rate from 67.7% to 22.6% on the same 31 expert-false rows (McNemar $p = 0.0026$, task-level $p = 0.0019$) at 50.0% recall against 93.5% and unequal coverage; the precision gap is within sampling error; the certificate fails on its error rate, not only on m.**

5.2 RQ2: WHAT EACH VIEW CONTRIBUTES

The intersection of the two views changes precision by $+1.6$ pp and coverage by -7.3 pp against the semantic view alone; the veto then removes 34 of the 75 rows both views accept (split by label in §6, Figure 3). Two rows of Table 1 carry this: the checker alone reaches 35.5% FPR at 53.1% coverage, and the best recorded judge gated by it 85.0% precision, 19.4% FPR and 54.8% recall at 43.8% coverage, which the paired test does not separate from the rule ($p = 1.000$): the two model views are not shown to add anything the checker does not give a single judge. **RQ2 takeaway: the second view removes 3 false positives and moves precision by $+1.6$ pp, which is 0.2$\times$ the noise floor of §6 and is not claimed; the deterministic structural veto is the larger step, taking the false-positive rate from 58.1% to 22.6%, and removes 23 true successes in doing so; a single judge gated by the same checker is not separable from the rule ($p = 1.000$); whether the checker selects by label is unresolved (Fisher $p = 0.188$, odds ratio 2.12, 95% CI [0.62, 7.45]).**

5.3 RQ3: AUDIT-SUPERVISED 7B STUDENT

The student agrees with the teacher on 83.3% of the view pairs that parsed (180/192; 88.9% and 87.8% under two further seeds, App. D), with test precision 87.1% and coverage 35.4%, at 6.42$\times$ the teacher's median latency, where the teacher's figure is its pooled wall time amortised over batches of 4 calls and the student's is per call, so the ratio is descriptive rather than a per-call measurement. The counting bound of §3.4 holds (final false-report rate 10.5% against the bound 40.4%), but the pre-registered fidelity criterion $\delta_c \leq 15\%$ is missed ($\delta_c = 22.0\%$). Its certificate, on the 56 unseen test-task representatives, reports on $m = 15$ with $e = 2$ errors, a 95% upper bound of 36.4% against the 0.10 target (certified: no), so the student deploys as all-abstain; at that m the smallest attainable bound is 18.1%, so no labelling could have certified. Its realized risk is 13.3% (95% CI [1.7, 40.5]) against the teacher's 48.1%: fewer reports rather than better calibration; its metrics are in-sample (§3.4). **RQ3 takeaway: the student reproduces 83.3%–88.9% of the teacher's view verdicts over three seeds, is 6.42$\times$ slower on that descriptive ratio, misses the pre-registered $\delta_c \leq 15\%$ criterion, and its certificate fails on m alone; its 87.1% test precision against the teacher's 81.6% is 0.8$\times$ the noise floor and is not claimed.**

5.4 RQ4: GAMING-THE-JUDGE ROBUSTNESS

Gaming is measured on the 30-claim design-pilot pool against its committed baseline re-judge, and kept out of the holdout estimates (Figure 6, Appendix G). Across the four attack families the pinned observational re-judge flips verdicts on up to 10.0% of records (exact 95% upper bound 23.9%) and its worst-case false-positive rate is 100.0%, also its baseline (100.0%); on three families it falls by one record, which $n = 9$ cannot resolve. The gate here is the pilot's operator gate (§3.1), not Table 1's deployment gate; every attack rewrites the terminal message, which its offline legs read, so its worst-case false-positive rate rises from 22.2% to 66.7%, with up to 10/30 suppressed records flipping to reports; H2 (no flip-to-report) fails under every attack and H3 (gamed FPR within $1/9$ of baseline, one record) on 3 of four. **RQ4 takeaway: rewrites confined to reasoning cannot move a verdict (Theorem 1; proved, not measured here); rewrites touching the terminal message move the operator gate's worst-case FPR from 22.2% to 66.7% on the pilot, and both pre-registered operator-gate hypotheses fail.**

6 ANALYSIS: WHERE THE FALSE POSITIVES GO, AND WHAT THE CERTIFICATE MEASURES

Which component removes which false successes. Figure 3 charges each removed row to the first conjunct of the base rule it fails. Of the 31 expert-false rows of the cohort (the primary judge's accept set) the semantic view removes 10, the evidence-only view 3 more, and the structural veto 11 more; 7 remain. Of the 62 expert-true rows, the two views remove 4 and 4, and the veto removes 23. Both views say success on 75 rows (Table 2) and the veto refutes 34 of those. Read as rates on the same rows, each conjunct is a step of the rule, not of the gap to row 2: the semantic view alone still reports 67.7% of them, both views 58.1%, and the veto 22.6%. Most of the reduction is therefore the deterministic record checker's rather than the second model view's: 35.5 pp of the 45.2 pp (78.6%; the rule's total step equals the gap to row 2 because rows 2 and 5 coincide) against 9.7 pp, whereas

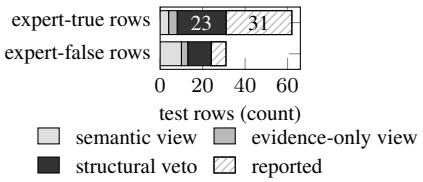

Figure 3: Where the 31 expert-false and 62 expert-true test rows leave the report set, charged to the first conjunct they fail; hatched = reported.

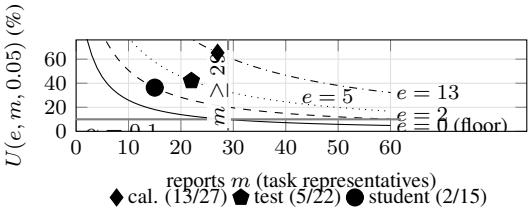

Figure 4: No measured certificate reaches the target: the one-sided CP95 upper bound $U(e, m, 0.05)$ against the report count m for fixed e ($e = 0$ is the attainability floor), the target $\alpha = 0.10$, the $m \geq 29$ line, and the three measured certificates.

Table 3: No Table 1 rate has a task-cluster standard deviation below 5.8 pp, so a gap under about 6 pp is inside resampling noise. Task-cluster bootstrap resampling the 56 test tasks with replacement (2,000 draws, fixed seed): sd in percentage points, band = 2.5–97.5 percentile of the rate itself.

	FPR		Precision	
Rule	sd (pp)	2.5–97.5% band (%)	sd (pp)	2.5–97.5% band (%)
Best recorded judge	8.5	52.0–85.2	5.8	60.8–83.5
GUI-Critic-R1	9.3	16.1–52.8	9.7	48.5–86.5
WebJudge-7B	8.5	55.6–88.9	7.1	53.5–81.3
Semantic view	9.5	50.0–86.5	6.1	60.5–84.0
Evidence-only view	10.2	50.0–89.7	6.2	58.9–83.0
CAVEAT base rule	9.0	7.1–41.4	7.1	66.7–94.6

among the 24 rows removed from the primary judge's accept set the checker's 11 (45.8%) is the largest single share but not a majority. Charged in the six possible orders that share runs 45.8%–83.3% (Shapley 61.1%); the rule's own order, used here, charges the checker least. Share is not selectivity: Fisher exact tests of removed-versus-kept against expert label find the semantic view selective (odds ratio 6.90, 95% CI [1.71, 32.70], $p = 0.0019$) and cannot resolve the evidence-only view ($p = 0.375$) or the veto (2.12, [0.62, 7.45], $p = 0.188$). The checker removes more true successes (23) than false ones (11). Per-agent base-rule FPR spans 11.1%–50.0% over 4–11 rows each; descriptive.

Predicted, then measured. The attainability threshold $m \geq 29$ for $\alpha = 0.10$ at $\delta = 0.05$ was pre-registered with the split sizes, so the 10.5% floor of §5 was fixed before any label was read; the student's certificate sits on the same curve at $m = 15$ (Figure 4). The i.i.d. premise of Theorem 2 has a direct check: the frozen rule realizes 48.1% risk on calibration representatives and 22.7% on test representatives (a task-unit rate, not Table 1's row-level FPR), a 25.4 pp gap (Fisher $p = 0.082$).

Noise floor. The 96 test rows cover 56 tasks (1.71 rows per task) whose repeated attempts share semantics and evidence, so the certificate counts task units and a row-level interval is anti-conservative. Table 3 resamples the 56 test tasks with replacement (2,000 draws, fixed seed) and recomputes every rate. The base rule's FPR has standard deviation 9.0 pp (2.5–97.5% band [7.1, 41.4]), its precision 7.1 pp, its coverage 6.6 pp; the recorded judges' FPR moves up to 23.8 pp between splits (gpt-4o-noaxtree), and the teacher rule's own calibration-versus-test gap is 25.4 pp. Figure 5 measures every gap in units of that floor; we claim a gap only when it is at least twice the floor, and the one-floor line marks resolvability rather than a claim. The headline FPR gap of 45.2 pp is 5.0× the floor and is claimed, as is the veto's own step (4.0×), both at unequal coverage (42.7% against 84.4%): thinning the judge's 81 reports at random to the rule's 41 (20,000 paired draws) leaves a matched-coverage difference, rule minus thinned judge, of -11.7 pp (95% CI [-32.3, 8.9], includes zero), 1.3× the floor and not claimed. Thinning at random carries no information of its own; a k-of-K vote over the 11 recorded judges never reaches the rule's coverage (61.5% at unanimity), and its nearest rung is inside the floor (App. H). Nothing else reaches the threshold (precision gap 1.1×, Fisher $p = 0.366$; intersection 0.2×; evidence-only step 1.1×; critic gap 1.4×; student precision 0.8×), and none is claimed.

Table 2: The two views agree on 75 of the 96 test rows as success; their disagreement is mostly the evidence-only view declining to report (rows: semantic; columns: evidence-only; unp. = non-report).

$v_s \backslash v_e$	succ.	fail.	insuff.	unp.
succ.	75	0	1	6
fail.	4	1	1	0
insuff.	2	0	2	0
unp.	2	0	0	2

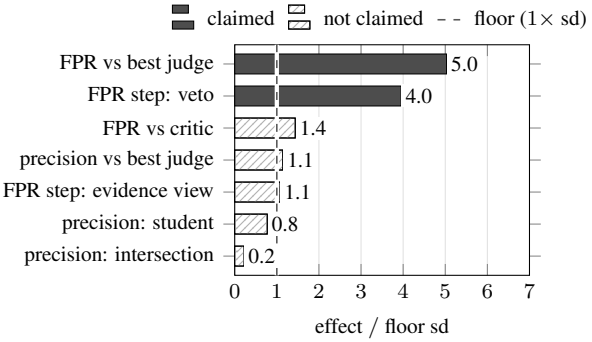

Figure 5: Two gaps clear the task-cluster noise floor and are claimed; five do not. Bar: gap divided by the bootstrap sd of the rate it moves.

Pilot rows and negative results. On visualwebarena.293 / gpt-4o-2024-11-20 replay and step ablation both confirmed the credited action, the re-judge said success, and the expert label is Unsuccessful; occlusion alone refuted it. On webarena.654 / Llama-3.3-70B-Instruct ablation refuted a re-judge success, matching the label. The blinded 32B re-judge confirms all 9 expert-false pilot rows (Appendix G); the student and both operator-gate hypotheses fail (§5).

7 Limitations

The invariance theorem is exact but covers only rewrites of fields the verifier never reads, and no row here measures it; the risk theorem is distributional, controlling the error rate of reports under i.i.d. tasks rather than the truth of an individual report. The student's certificate fails on m alone; the teacher's two fail on the realized error rate (48.1% on calibration, 22.7% on test representatives), which no attainable m certifies at 0.10, so the protocol is shown only in its abstaining regime; a setting that licenses reporting is the experiment this paper lacks, and splitting the other way would not change that (bound 42.0%). The i.i.d. task assumption of the certificate must be monitored, and §6 gives the one check our data allow. *Every row is conditional* on one public judge's accept set, so no number here estimates verifier performance on AgentRewardBench at large, and two Table 1 cells are tautologies of that conditioning. *The executed layer is a pilot that failed its own gate*: it reached 22 of 30 trajectories, decided none of 22 state mutations, and refuted more of the covered true successes (35.7%) than of the false ones (16.7%); the layers are never combined on the holdout. *Breadth*: three auditors (App. C), one benchmark pair, temperature 0, three student seeds spanning 5.6 pp fidelity and 5.3 pp coverage (App. D), no transfer setting. The two prompts agree on 80 of 96 rows, so the intersection is a design choice rather than an independence assumption; labels can be wrong, GUI-Critic-R1 is repurposed from step critique, and live counterfactuals need restorable services. *The floor is one-arm*: §6 divides each gap by one arm's bootstrap sd, not the paired difference's.

8 Conclusion

CAVEAT pairs a reasoning-free two-view audit of the fixed record and a deterministic structural veto with a pre-registered task-level risk certificate that decides whether it may report. On the label-blind holdout the frozen rule has a lower false-positive rate than every LLM-judge, critic, reward-model and single-view baseline at its native coverage (22.6% at 42.7%), and the recorded judge gated by the same checker reaches 19.4% at 54.8% recall; a paired test supports the FPR half (exact McNemar $p = 0.0026$), not the precision half (Fisher $p = 0.366$). The deterministic checker accounts for 78.6% of the false-positive drop at a cost of 23 true successes; the gated judge is not separable from the rule (McNemar $p = 1.000$). The predeclared 0.10 certificate fails calibration (13 errors among 27 reports), so the deployed policy abstains on every test item. The 7B student fails its own certificate.

REPRODUCIBILITY STATEMENT

All verdicts, gates, and reported numerals derive from seeded, pinned artifacts: AgentRewardBench is pinned to an immutable revision (`b6d17e6`), sampled trajectories carry sha256 manifests, and every numeral in this paper is emitted by a macro generator (`experiments/caveat/gen_macros.py`) from the released result files; quantities whose experiments have not completed appear as explicit unfilled-macro markers rather than fabricated values. The verifier, operators, gate, and orchestration runbooks are released with per-step assertions and rollbacks. All released files are in the supplementary zip; start at `supp/README_supp.md`. Appendix B details schemas, thresholds, and compute.

AI USE STATEMENT

In this work we disclose generative AI use under the categories of the ICLR 2027 AI policy. Within the required-disclosure categories, we used generative AI tools to help develop the conceptual framework, propose and refine hypotheses, formulate the paper's formal claims, provide critical ingredients for proving them and assist in the writing of proofs, give feedback on research methodology and experiment design, implement methods (the experiment, analysis and figure-generating code released with this work), clean and reformat datasets, and interpret results. We have *not* used generative AI tools to generate synthetic data sets, assist with translation, or support qualitative or thematic data analysis; the remaining required-disclosure categories are not applicable to this work, which has no human-subject component and no data collection beyond public benchmarks and the outputs of open-weight models. Under the recommended categories, generative AI was additionally used for software code, literature search and analysis, brainstorming, figure preparation, and editing the text of this paper. No experimental result was produced, selected or filtered by a generative AI tool: every experimental numeral in this paper is emitted by a committed single-writer artifact and reaches the text through an auto-generated macro bridge that admits no hand-typed value; design constants, protocol settings and figures quoted from prior work are of course written by hand and cited. The research questions, the claims and the conclusions are the authors'. We have reviewed all AI-assisted work and take responsibility for the final content, including every claim, number and artifact.

ETHICS STATEMENT

This work builds a verifier for GUI agents, and the questions it raises are about what such a verifier licenses rather than about the data it uses: the benchmark is public and no human subject is involved. The live results in §4 were produced against self-hosted, rootless WebArena/VisualWebArena services (local replicas of shopping, Wikipedia, Postmill, GitLab and classifieds sites) and not against any third-party service or real user account. OpenStreetMap, the one site with no distributable hermetic backend, is marked not replayable rather than evaluated against the public service, which is the rule we would want applied generally: an agent-verification study should not send its interventions to somebody else's production system. Three scoping limits belong here because a reader could otherwise take the certificate for more than it is. The system measures whether an agent's trajectory did what the task asked; it does not detect or bound harm, and no claim in the paper says otherwise. The task-level certificate bounds a risk under a pre-registered calibration protocol, so it speaks only about the distribution it was calibrated on; on this paper's own holdout it fails calibration and the deployed policy abstains on every item, which is the behaviour we argue a verifier owes its users and not an incidental result. And the counterfactual layer executes interventions on restorable environments; deploying that layer anywhere it can touch state that is not restorable would break the premise it rests on. Every state-bearing run here begins from a pristine restore or the benchmark's reset endpoint, and we release no credentials, sessions or captured pages.

REFERENCES

Asaad Althoubi. Look before you leap: Pre-action verification for llm agents. arXiv:2609.11957, 2026.

Anastasios N. Angelopoulos, Stephen Bates, Emmanuel J. Candès, Michael I. Jordan, and Lihua Lei. Learn then test: Calibrating predictive algorithms to achieve risk control. arXiv:2110.01052, 2022.

Akash Bonagiri, Devang Borkar, Gerard Janno Anderias, et al. Causalflow: Causal attribution and counterfactual repair for llm agent failures. arXiv:2605.25338, 2026.

Hyungjoo Chae, Sunghwan Kim, Junhee Cho, et al. Web-shepherd: Advancing prms for reinforcing web agents. arXiv:2505.15277, 2025.

Jiwan Chung, JiHyuk Byun, Vibhav Vineet, and Seon Joo Kim. Where did it go wrong? process-level evaluation of web agents with semantic state tracking. arXiv:2606.15673, 2026.

Chaoqun Cui, Jing Huang, Shijing Wang, et al. Agentic reward modeling: Verifying gui agent via progressive trajectory-grounded interaction. arXiv:2602.00575, 2026.

Xiang Deng, Yu Gu, Boyuan Zheng, et al. Mind2web: Towards a generalist agent for the web. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023. arXiv:2306.06070.

Darshan Deshpande, Varun Gangal, Hersh Mehta, et al. Trail: Trace reasoning and agentic issue localization. arXiv:2505.08638, 2025.

Shengda Fan, Xuyan Ye, Yupeng Huo, Zhi-Yuan Chen, et al. Agentprocessbench: Diagnosing step-level process quality in tool-using agents. arXiv:2603.14465, 2026.

Yunhao Feng, Ruixiao Lin, Ming Wen, Qinqin He, Yanming Guo, Yifan Ding, Yutao Wu, and Jialuo Chen. Safety testing llm agents at scale: From risk discovery to evidence-grounded verification. arXiv:2607.01793, 2026.

Shanshan Gao and Liyi Zhou. Can agent benchmarks support their scores? evidence-supported bounds for interactive-agent evaluation. arXiv:2605.10448, 2026.

Yonatan Geifman and Ran El-Yaniv. Selective classification for deep neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

Bhaskar Gurram. Auditing automated evaluation, error propagation, and runtime mitigation in tool-using language agents. arXiv:2604.16706, 2026.

Sirui Hong, Zhijie Liu, Tengfei Li, Wei Tao, Yifan Wu, and Chenglin Wu. Diageval: Trajectory-conditioned diagnosis for reliable software evaluation with gui agents. arXiv:2605.17439, 2026.

Muhammad Khalifa, Lajanugen Logeswaran, Jaekyeom Kim, et al. Gaming the judge: Unfaithful chain-of-thought can undermine agent evaluation. arXiv:2601.14691, 2026.

Zehao Li, Zhenyu Wu, Yibo Zhao, et al. Os-themis: A scalable critic framework for generalist gui rewards. arXiv:2603.19191, 2026.

Haojia Lin, Xiaoyu Tan, Yulei Qin, et al. Cuarewardbench: A benchmark for evaluating reward models on computer-using agent. arXiv:2510.18596, 2025.

Xing Han Lù, Amirhossein Kazemnejad, Nicholas Meade, et al. Agentrewardbench: Evaluating automatic evaluations of web agent trajectories. arXiv:2504.08942, 2025.

Ming Ma, Jue Zhang, Fangkai Yang, et al. Dover: Intervention-driven auto debugging for llm multi-agent systems. arXiv:2512.06749, 2025.

Hadi Mohammadi. trajectory-judge: What outcome-only llm judges miss on agent trajectories. arXiv:2609.00038, 2026.

Jiachen Qian. Viscritic: Visual state comparison as process reward for gui agents. arXiv:2606.24525, 2026.

Corby Rosset, Pratyusha Sharma, Andrew Zhao, Miguel Gonzalez-Fernandez, and Ahmed Awadallah. The art of building verifiers for computer use agents. arXiv:2604.06240, 2026.

Fabio Rovai. Civex: Causal intervention verification for language agents. arXiv:2605.09168, 2026.

Shuvom Sadhuka, Drew Prinster, Clara Fannjiang, et al. E-valuator: Reliable agent verifiers with sequential hypothesis testing. arXiv:2512.03109, 2025.

Jaineet Shah. Causal agent replay: Counterfactual attribution for llm-agent failures. arXiv:2606.08275, 2026.

Wentao Shi, Yu Wang, Yuyang Zhao, et al. Aj-bench: Benchmarking agent-as-a-judge for environment-aware evaluation. arXiv:2604.18240, 2026.

Linxin Song, Jieyu Zhang, Huanxin Sheng, et al. Video-based reward modeling for computer-use agents. arXiv:2603.10178, 2026.

Yuyang Wanyan, Xi Zhang, Haiyang Xu, et al. Look before you leap: A gui-critic-r1 model for pre-operative error diagnosis in gui automation. arXiv:2506.04614, 2025.

Yifan Xiao and Shijie Li. ARGUS: Argumentation-based minimal-change repair for verifiable LLM self-explanations. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence (UAI)*, 2026.

Yifan Xiao, Zhongjun Wang, Tiantian Yan, Weiwei Li, Dazhong Wu, and Jie Ma. Time series anomaly detection: A hybrid deep learning framework with adaptive thresholding for electricity industry data. In *IEEE International Conference on Cyber Security and Cloud Computing (CSCloud)*, 2025. doi: 10.1109/CSCLOUD66326.2025.00046.

Yifan Xiao, Shijie Li, and Yuhao Ge. RunbookFX: Type- and effect-safe LLM synthesis for executable incident diagnosis and mitigation. *Proceedings of the ACM on Programming Languages (ICFP)*, 2026a. doi: 10.1145/3828675.

Yifan Xiao, Shijie Li, and Yu Huang. CALM-TS: Risk-controlled LLM labeling for time-series via calibrated selective gating. In *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, 2026b. doi: 10.1145/3770855.3817859.

Yifan Xiao, Shijie Li, and Yu Huang. CodeFault: Predicting fault risks from code changes via multi-modal learning. In *Proceedings of the IEEE/ACM International Conference on Automated Software Engineering (ASE)*, 2026c.

Yifan Xiao, Shijie Li, and Yu Huang. ProgV-Trace: An LLM-driven verification framework for evidence-grounded video misinformation attribution. In *Proceedings of the ACM International Conference on Multimedia (ACM MM)*, 2026d. doi: 10.1145/3767308.3836284.

Yifan Xiao, Dexuan Xu, Shijie Li, Weiping Ding, and Yu Huang. Fuzzy-gated verification of LLM agents for industrial fault diagnosis: An evolutionary-optimized closed-loop framework. *Applied Soft Computing*, 2026e. In press.

Yifan Xiao, Dexuan Xu, Shijie Li, and Yu Huang. Plan-retrieve-verify: An LLM-driven framework for long-video audio-visual evidence reasoning. In *Proceedings of the ACM International Conference on Multimedia (ACM MM)*, 2026f. doi: 10.1145/3767308.3836283.

Tianbao Xie, Danyang Zhang, Jixuan Chen, et al. Osworld: Benchmarking multimodal agents for open-ended tasks in real computer environments. In *Advances in Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks Track*, 2024. arXiv:2404.07972.

Dexuan Xu, Yanyuan Chen, Zhongyan Chai, Yifan Xiao, Yandong Yan, Weiping Ding, Hanpin Wang, Zhi Jin, Wenpin Jiao, Weihua Yue, Hang Li, and Yu Huang. Knowledge fusion in deep learning-based medical vision-language models: A review. *Information Fusion*, 125:103455, 2026a.

Dexuan Xu, Yifan Xiao, Shijie Li, Xingyu Yang, Siqi Hou, Hanpin Wang, and Yu Huang. Beyond one granularity: Spectral ensemble and consensus expansion for training-free composed image retrieval. In *Proceedings of the ACM International Conference on Multimedia (ACM MM)*, 2026b. doi: 10.1145/3767308.3835648.

Tianci Xue, Weijian Qi, Tianneng Shi, et al. An illusion of progress? assessing the current state of web agents. In *Conference on Language Modeling (COLM)*, 2025. doi: 10.48550/arXiv.2504.01382. arXiv:2504.01382.

Yuwen Zhai, Runze Li, Liang Wang, et al. Guide: Interpretable gui agent evaluation via hierarchical diagnosis. arXiv:2604.04399, 2026.

Guibin Zhang, Junhao Wang, Junjie Chen, Wangchunshu Zhou, Kun Wang, and Shuicheng Yan. Agentracer: Who is inducing failure in the llm agentic systems? arXiv:2509.03312, 2025a.

Guijia Zhang, Hao Zheng, and Harry Yang. Hallucination as exploit: Evidence-carrying multimodal agents. arXiv:2605.19192, 2026a.

Shaokun Zhang, Ming Yin, Jieyu Zhang, et al. Which agent causes task failures and when? on automated failure attribution of llm multi-agent systems. In *International Conference on Machine Learning (ICML)*, 2025b. Spotlight; arXiv:2505.00212.

Yao Zhang, Shijie Tang, Zeyu Li, et al. Webarbiter: A principle-guided reasoning process reward model for web agents. arXiv:2601.21872, 2026b.

Yujin Zhang and Daye Nam. Hansel: Extracting breadcrumbs from web agent trajectories for interactive verification. arXiv:2606.18671, 2026.

Shuyan Zhou, Frank F. Xu, Hao Zhu, et al. Webarena: A realistic web environment for building autonomous agents. In *International Conference on Learning Representations (ICLR)*, 2024. arXiv:2307.13854.

Yang Zhou, Zihan Dong, Zhenting Wang, et al. Evidence over plans: Online trajectory verification for skill distillation. arXiv:2605.09192, 2026.

Kunlun Zhu, Zijia Liu, Bingxuan Li, et al. Where llm agents fail and how they can learn from failures. arXiv:2509.25370, 2025.

A Proofs

Proof of Theorem 1. Both auditor prompts are deterministic functions of the preserved fields, and h also ignores agent reasoning. Thus the two generated prompts and the structural input are byte-identical before and after A, so under the assumed deterministic, batch-invariant decoding the outputs and g are identical. The exclusion is unit-asserted: a sentinel injected into agent reasoning never reaches either view prompt. □

Proof of Theorem 2. Conditional on the fixed selection rule and its number of reports, the reports are i.i.d. draws, so the error count is binomial at the selective risk. The one-sided Clopper–Pearson bound covers that risk with probability at least $1 - \delta$. Deployment is enabled only on the event $U \leq \alpha$; otherwise coverage is zero. □

B Reproducibility Details

Compute and reproducibility. All model and dataset revisions, split hashes, prompts and manifests are pinned; raw generations are append-only; only aggregation reads expert labels. Of the 480 confirmatory teacher view calls, 25 (5.2%) return no parseable verdict and default to non-report, an abstention-conservative failure on 21 of the 240 rows. This appendix states what the released artifacts do and do not measure about compute (the campaign was budgeted under a 72 GPU-hour ceiling).

Data pins. AgentRewardBench Lù et al. (2025) is pinned to revision `b6d17e646009d6cb63d5dd7be78807b680693f61` (dataset id verified via the Hugging Face API against the `arxiv:2504.08942` tag; the similarly named `MultimodalAgent/Agent-RewardBench` is a different paper's dataset and is explicitly excluded by an id assertion). The both-branches sample is a seeded draw (seed 20260717) of 50 core trajectories from WebArena+VisualWebArena plus a deterministic 8-trajectory top-up to 30 judge-accepted under the primary judge (`gpt-4o-noscreen`), 58 manifest rows total, each carrying the file's sha256. Day-1 schema probes assert file counts (1,302 cleaned trajectories, 19,530 judgment files, 1,408 annotation rows), per-benchmark splits, label domains, and bijective annotation–trajectory joins.

Claim schema and gate. Claim types, verdict domains, and the gate rule are exactly §3: report iff ≥ 1 CONFIRMED and 0 REFUTED, UNVERIFIABLE suppressed, the invariant asserted on every emitted claim.

Compute: what is measured and what is not. The body's campaign was budgeted under a 72 GPU-hour ceiling on a single 80 GB GPU node, where the student seeds of Appendix D also ran; the auditor-scale legs of Appendix C ran on 24 GB RTX 3090 cards. Every stage records the facts that determine its cost: model and revision pins, dtype, context and screenshot budgets, and per-batch generation wall times inside the append-only raw logs (batch size itself is not recorded). We do *not* release a per-stage GPU-hour ledger: the raws carry generation wall time, not exclusive device occupancy, and reconstructing hours from them would be an estimate presented as a measurement. The corresponding macro is therefore left visibly unfilled in the released macro file (`paper/results/macros.tex` in the supplementary zip) rather than filled with a plausible number, which is the same convention this paper applies to every quantity whose experiment has not produced it.

Recorded-judge grid. Table 4 gives the complete 15-configuration $\times$ three-split grid behind Table 1's recorded-judge rows, including the 3 configurations the pinned parser does not map. Rows 2 and 5 of Table 1 print identical cells because their four confusion counts coincide, although the two rules disagree on 23 of the 96 test rows; it is a coincidence, not a duplicated cell.

Table 4: All 15 recorded AgentRewardBench judge configurations on the confirmatory cohort, per split, in percent. "Parses" is the label-free, split-blind completeness filter: a configuration is complete iff the pinned parser maps its recorded verdict on all 240 rows. Incomplete configurations show dashes rather than rates: with no mapped verdict their coverage is 0 and their FPR a vacuous 0.0, which is the degenerate optimum the filter exists to exclude. ‡ primary (cohort-defining) judge; ⋆ the test-selected envelope row of Table 1; § environment reward, an oracle reference outside the judge family. Auto-generated from the released artifact.

		Teacher-train			Calibration			Test		
Judge config	Parses	Cov	Prec	FPR	Cov	Prec	FPR	Cov	Prec	FPR
aer	no	—	—	—	—	—	—	—	—	—
aerv	no	—	—	—	—	—	—	—	—	—
claude-3.7-sonnet-noaxtree⋆	yes	85.4	75.6	66.7	80.2	66.7	73.5	84.4	73.4	67.7
claude-3.7-sonnet-noscreen	yes	85.4	73.2	73.3	90.6	64.7	88.2	89.6	69.0	83.9
functional§	yes	47.9	91.3	13.3	46.9	84.1	20.6	40.6	92.3	9.7
gpt-4o-mini	yes	95.8	71.7	86.7	84.4	68.4	73.5	95.8	66.3	96.8
gpt-4o-mini-noaxtree	yes	91.7	72.7	80.0	79.2	67.6	70.6	93.8	66.7	93.5
gpt-4o-mini-noscreen	yes	95.8	69.6	93.3	88.5	65.1	85.3	93.8	67.8	90.3
gpt-4o-mini-noscreen-noaxtree	yes	81.3	71.8	73.3	83.3	64.1	82.4	87.5	63.0	96.8
gpt-4o-noaxtree	yes	89.6	74.4	73.3	95.8	63.3	97.1	96.9	67.8	93.5
gpt-4o-noscreen‡	yes	100.0	68.8	100.0	100.0	63.8	100.0	100.0	66.7	100.0
llama-3.3-70b-noscreen	yes	85.4	68.3	86.7	88.5	65.1	85.3	89.6	69.9	80.6
nnetnav	no	—	—	—	—	—	—	—	—	—
qwen-2.5-vl-noaxtree	yes	100.0	68.8	100.0	90.6	65.9	85.3	90.6	69.0	83.9
qwen-2.5-vl-noscreen	yes	93.8	71.1	86.7	93.8	65.9	88.2	96.9	67.8	93.5

The same points, with the exact intervals of every Table 1 row, are Figure 1; the separation from the oracle is not resolved at this sample size: its precision interval is [79.1, 98.4] against CAVEAT's [65.7, 92.3], and its FPR interval [2.0, 25.8] against [9.6, 41.1].

Substrate orchestration. Services, port plan, acquisition manifest (exact byte sizes and published checksums), bring-up order, health probes, reset procedure, and the performance acceptance thresholds (20-action trajectory median $\leq$ 5 s / p95 $\leq$ 15 s / total $\leq$ 5 min; warm TTFP $\leq$ 5 s, GitLab $\leq$ 10 s; reset $\leq$ 5 min) are released as a cmd+assert+rollback runbook with the orchestration and measurement scripts. Nothing about the substrate is asserted from documentation alone; unmeasured items are marked as such in the runbook.

C Two More Auditors: Does the Failed Certificate Move with Auditor Size?

Every confirmatory number in the body comes from one auditor. A reviewer can read the failed 10% certificate as a property of that auditor rather than of the protocol. We therefore reran the confirmatory campaign, unchanged, with two other auditors in its place: Qwen2.5-VL-7B-Instruct (revision `cc594898`, the student's base model) and Qwen2.5-VL-32B-Instruct-AWQ (revision `66c370b7`, int4 weights, float16 activations). Everything else is the body's: the same 240-row task-isolated holdout and splits, the same two frozen views and prompts, the same fixed report rule, temperature 0, the same context and screenshot budgets. Each leg was registered before its first row existed (`PREREG_B3_AUDITOR_SCALE.md`, `PREREG_B3_LEG2_32B.md`); Table 5 gives the read-out.

Table 5: The confirmatory protocol under three auditors: Qwen2.5-VL-7B-Instruct, Qwen3-VL-30B-A3B-Instruct (the body's auditor) and Qwen2.5-VL-32B-Instruct-AWQ. The axis confounds size with generation (Qwen2.5-VL against Qwen3-VL), with activation (3B active in the 30B mixture) and with quantisation (AWQ int4 for the 32B); three points are reported as three points. Calibration rows are the certificate's 60 hash-fixed task units and the 94 labelled calibration rows; test rows are the untouched test split.

	7B	30B-A3B (body)	32B-AWQ
Calibration reports / errors (of 60 units)	23 / 11	27 / 13	30 / 15
Clopper–Pearson 95% upper bound on selective risk	66.5%	65.4%	66.2%
Certified at 10%	no	no	no
Calibration coverage, combined rule	39.6%	44.8%	45.8%
Calibration FPR, semantic view	85.3%	67.6%	91.2%
Calibration FPR, evidence-only view	64.7%	73.5%	70.6%
Test precision, combined rule	80.0%	81.6%	78.0%
Test FPR, combined rule	22.6%	22.6%	29.0%
Test coverage, combined rule	38.5%	42.7%	45.8%

Read-out against the registrations. *The certificate does not turn on auditor size (S1, T1): holds.* All three upper bounds sit between 65.4% and 66.5% against the 10% target, and all three certificates fail. The failure is a property of the fixed rule on this holdout, not of any one auditor. *Coverage (S2, T2): holds.* The 7B auditor covers less of the calibration split than the 30B (39.6% against 44.8%), and the 32B at least as much as the 7B (45.8%), both as registered. The 7B registration also predicted that each of its views would have a false-positive rate at or above the 30B's; that is true of the semantic view and false of the evidence-only view, and the direction is reported as measured. *Ordering (S3, T3): no single ordering.* The 32B registration named two size orderings, total parameters (7B $<$ 30B-A3B $<$ 32B) and active parameters (30B-A3B $<$ 7B $<$ 32B), and asked which each quantity follows. Calibration coverage is monotone in total parameters; the semantic view's false-positive rate is monotone in active parameters; the evidence-only view's rate follows neither. No scale claim is made. What the three points share is the intersection rule's effect: each auditor's combined-rule false-positive rate on the calibration split (44.1%, 95% CI [27.1, 62.2], on 34 expert-false rows) is below either of its views' (67.6% and 73.5%), and each auditor's certificate fails by a similar margin.

Cost. The 7B leg ran on one 24 GB RTX 3090 in 27 minutes of wall-clock time for both views over the 240 rows (auditor memory fraction 0.95); the 32B leg ran on two such cards with tensor parallelism in 152 minutes (fraction 0.85, the engine's admission cap set to 8 sequences, the runner submitting 4 prompts per call as in the body). No protocol setting was reduced to fit. Raw generations and aggregations are in `results_b3_7b/` and `results_b3_32b/` beside the body's `results/`, never over it.

D The Student under Three Seeds

The shipped student is one training run. Registered before any second-seed row existed (`PREREG_B5_STUDENT_SEEDS.md`), the same recipe (LoRA rank 16 on the same base, the same

SFT rows, three epochs) was run twice more with the config seed set to two new values; the config seed governs the per-epoch example order and the blinded evaluation order, while the LoRA initialisation is unseeded in every run, the shipped one included, so the spread below is the recipe's launch-to-launch variability. Table 6 gives the read-out.

Table 6: The audit-supervised student under three seeds: view-verdict fidelity on the parsed test view pairs, standalone test coverage and precision, latency ratio against the teacher, and the abstain-on-failure certificate on the 56 unseen test-task representatives (reports m, errors e, one-sided 95% Clopper–Pearson upper bound against the 0.10 target).

seed	fidelity	coverage	precision	cost	m / e	upper bound
shipped	83.3%	35.4%	87.1%	6.42×	15 / 2	36.4%
second	88.9%	40.6%	83.3%	6.04×	19 / 4	42.0%
third	87.8%	39.6%	88.6%	6.07×	19 / 3	36.0%

Read-out against the registration. *U1 (fidelity within 3 pp of the shipped seed): fails.* Both new seeds sit above the shipped run, by more than the registered tolerance, so the abstract and the RQ3 takeaway print the three-seed range 83.3%–88.9% and this appendix names the seeds; the shipped run is the lowest of the three. *U2 (the certificate fails at every seed): holds.* Certified under any seed: no; the largest upper bound is 42.0%. *U3 (spread below the paired-CI resolution): fails.* Fidelity spans 5.6 pp and coverage 5.3 pp over three seeds, both above the registered 3 pp and 5 pp, and the limitations paragraph carries them as the student's noise floor. Nothing in Table 1 moves; the shipped student remains the reported one.

E The Record Checker h

h is the offline replay operator of Appendix F applied to the primary judge's success claim on the record. It reads the recorded actions, the observation that follows each (accessibility tree, URL and the environment's execution-error field) and the terminal message, and nothing else. Four predicates, each a fixed function of those fields, decide the verdict: (1) *loop tail*: the trajectory ends in three or more consecutive identical (action, URL) pairs; (2) *terminal execution error*: the environment recorded an error for one of the last three actions (an earlier error the agent recovered from does not count); (3) *missing target*: one of the last three actions names an element absent from the accessibility tree it acted on; (4) *ungrounded answer*: no content token of the terminal message (four or more characters) appears in any recorded observation. Any predicate firing gives REFUTED; CONFIRMED needs a grounded answer and no firing; a record with no terminal message and no firing is UNVERIFIABLE, which the base rule does not treat as a refutation. The predicates and the two thresholds (three repeats, four characters) were fixed on the 30-trajectory design pilot before the confirmatory cohort was drawn and were not revisited; no confirmatory label was read while building them. Because predicate (4) reads the terminal message, an attack that rewrites the message can change h, and the four gaming attacks do (§5).

F Offline Skeleton Sanity Check (Machine Heuristics, Not Measurements)

Before any GPU or live-environment work, the offline layer ran end-to-end on the 30 judge-accepted trajectories as an engineering falsify-first skeleton. **These are deterministic record-level heuristics, not false-positive measurements**: every row is labelled `machine_heuristic` and this artifact's M1 pass/fail field is null by design. The false-positive *measurements* are the separate double-blind VLM re-judge stage and the live executed-operator leg of Appendix G. Expert labels (Table 7) are shown only as a sanity crosscheck and play no role in candidate selection.

The skeleton flags 15 false-success *candidates* (2 occlusion cases had no citable evidence); applying the strict gate to the judge's 30 success claims would report 4 and suppress 26. Reference-only crosscheck: 11 of the 30 judge-accepted trajectories carry an expert *Unsuccessful* label, 4 of the 15 candidates do. Every skeleton row is labelled `machine_heuristic` in the released artifact.

Table 7: Machine-heuristic skeleton output on 30 judge-accepted trajectories (offline layer, 2026-07-18). Engineering signal only; the paper's FP numbers come from the double-blind re-judge stage (Appendix G) and the live gate.

Operator (record-level)	confirmed	refuted	unverifiable
Replay check	10	9	11
Occlusion check	0	9	21

G M1 Measurement Legs: Double-Blind VLM Re-Judge and Live Executed Operators

The re-judge stage turns the skeleton's candidates into a measurement. A strong open-weight VLM (Qwen2.5-VL-32B-Instruct, AWQ, vLLM, temperature 0, fixed seed) re-judges *all* 30 judge-accepted trajectories: the 15 machine-flagged candidates and the 15 unflagged controls. Blinding is structural: the prompt builder cannot receive machine flags, expert labels, or operator verdicts (whitelisted arguments, unit-asserted), and the evaluation order is a seeded hash shuffle independent of group membership. Each trajectory is rendered as evidence: up to 12 evenly-spaced screenshots (always including the first and last available), the task goal, the recorded action sequence with any recorded execution errors, and the agent's final message, plus the constant fact that an automated judge accepted the trajectory. The rubric forces one of three verdicts (`true_success` / `false_success` / `insufficient_evidence`) and names both error directions as equally serious; unparseable outputs are counted, never coerced. An output-quality gate aborts the run if more than 5% of the first 20 outputs match known quantization-overflow signatures.

Re-judged false-success rates: 0.0% (0 of 15) on flagged vs. 6.7% (1 of 15) on controls; overall, 1 of 30 judge-accepted trajectories (3.3%) are re-judged false successes, with 0 insufficient-evidence and 0 unparseable outputs. Against the expert reference (unanimous annotator labels; used only at aggregation, never shown to the judge), the re-judge refutes 0.0% of judge false positives at a 0.0% false-refutation rate on expert-verified true successes; the CIRCA-style gate metrics are verifier recall 1.000, false-confirm 1.000, and reports-wrong 0.321 against the pre-registered thresholds ($\geq$ 0.9 / $\leq$ 0.1 / $\leq$ 0.02), giving a scope-tagged verdict of fail on this (re-judge) leg alone.

What the re-judge result means. The re-judge confirms 29 of 30 trajectories, including *every one* of the 9 expert-unanimous false successes (false-confirm 1.000). A strong VLM, blinded, at temperature 0, with the full screenshot record in front of it, does not recover the false successes a weaker judge accepted; machine-flagged candidates are confirmed at essentially the control rate (0.0% vs. 6.7% re-judged false). This is the motivating failure mode of the paper observed directly on the evaluation pool: false successes survive stronger passive judging, so the gate must intervene rather than re-read.

Cross-analysis. Table 8 joins, read-only at the aggregation layer so the blind is never broken, expert consensus, the machine flags, the offline intervention-operator verdicts (deterministic replay checking + evidence occlusion, the executed record-level operators of Appendix F), and the re-judge verdicts; Table 9 lists the expert-unanimous false successes row by row. The offline operators refute 4 of the 9 expert false successes that the passive re-judge confirmed (44.4%; 5 are caught by neither), but they also refute 11 of the 19 expert true successes (57.9%); record-level projections cannot distinguish genuine failure from record incompleteness. Under the same CIRCA metrics, the offline-operator leg scores recall 0.105, false-confirm 0.222, reports-wrong 0.500 (verdict fail); a combined preview gate over all three record types scores 0.421 / 0.556 / 0.385 (verdict fail). No record-level leg meets the thresholds: passive re-judging is blind to the false successes and offline record checks over-refute; live executed interventions are the designed instrument for closing exactly that gap.

Live executed-operator leg (five-way). Executed step ablation and state mutation ran against the live rootless substrate on the 22-trajectory replayable subset (73.3% of the pool; a trajectory is replayable iff every site it recorded is a healthy live service). Discipline: intervention points, canary spans, and mutation targets pre-registered from the records and sha-frozen before any run; one pristine-snapshot restore before every state-bearing run (a failed restore voids the trajectory's runs); a broken baseline replay is UNVERIFIABLE, never a refutation; only a clean re-execution that

fails to reproduce the claimed end state refutes. Write-style claims are checked against *persisted* state (pristine vs. post-run probes of whitelisted database text columns), so unsaved form residue can never fake persistence; on substrates without a whitelisted column map (the postgres-backed forum and GitLab services) write claims fall back to the end-page rule and are explicitly flagged `residue_risk` in the artifact, a weaker probe, carried as such and never upgraded silently. Deterministically undecidable end states (7 rows) go to a blind end-state VLM stage (same pinned judge and blinding discipline as the re-judge leg; 0 resolved, 0 refuted under the frozen replay semantics: a satisfied or undecidable end state stays suppressed, never confirmed). Verdict counts on the subset, ablation: 3 confirmed / 7 refuted / 12 unverifiable; mutation: 0 / 0 / 22. Two forces drive the large unverifiable count, both conservative by design: (i) the record-driven replayer cannot yet faithfully reproduce Magento admin multi-widget forms (attribute grids, dynamically-numbered controls), so several baselines break mid-replay and are suppressed rather than counted as refutations; (ii) every state-mutation attempt on this subset is unverifiable: the Wikipedia mirror is read-only, several claims have no span in the whitelisted DB text columns to mutate, and the write-task baselines that would anchor a mutation did not first reach the claimed state. Against expert consensus the live leg refutes 1 of 6/9 covered expert-unanimous false successes (16.7%), 0 of the 3/5 covered rows missed by *both* record-level legs, while falsely refuting 5/14/19 covered true successes (35.7%). Scope-tagged gate metrics (replayable subset only): recall 0.143, false-confirm 0.167, reports-wrong 0.333 (verdict fail); the strict five-way gate over all records scores 0.071 / 0.500 / 0.750 (verdict fail). This is a proof of mechanism at low coverage rather than a resolved gate: the executed-intervention pipeline runs end to end with pre-registered points, per-run snapshot restores, and an audited evidence trail, but the decisive quantity, coverage, is bounded here by which services are online and by replay fidelity on complex forms, not by the method. The remaining $30-22$ trajectories join as their services come up and as the replayer's form coverage grows.

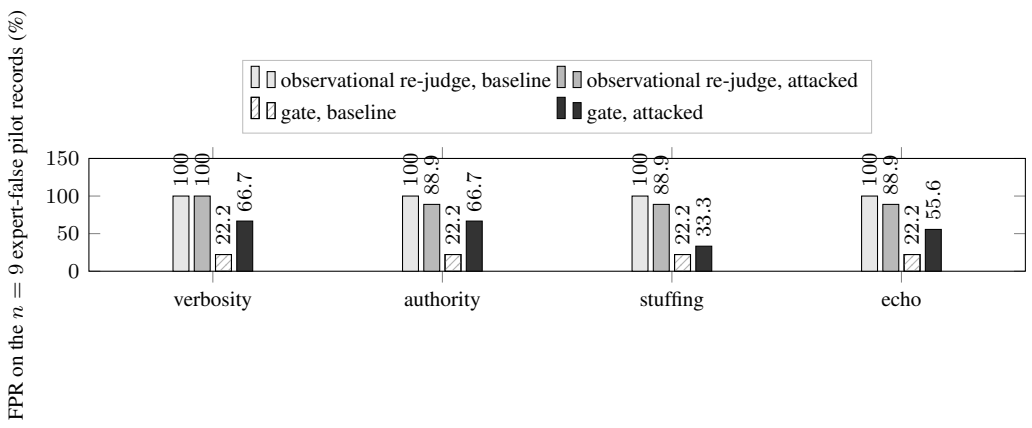

Figure 6: RQ4 on the 30-record design pilot: false-positive rate on its 9 expert-false records before and after each attack, for the pinned observational re-judge and the gate. The re-judge starts at 100.0%, so its bars are a ceiling; the gate starts at 22.2% and rises because every attack rewrites the terminal message, which the gate reads.

Table 8: Record-level cross-analysis on the 30 judge-accepted trajectories: expert consensus (unanimous annotator labels) vs. offline intervention operators (refuted = replay or occlusion) vs. double-blind VLM re-judge; the live executed-operator leg is reported in the text (its scope is the 22-trajectory replayable subset). All cells auto-generated from the released artifact.

Expert consensus	n	Operators refute	Re-judge says false
Successful	19	11	0
Unsuccessful	9	4	0
Mixed / unsure	2	0	1

Table 9: The 9 expert-unanimous false successes the primary judge accepted, row by row (auto-generated from the artifact). †: the accepting judge cited no checkable evidence span, so the occlusion operator is undecidable by design.

Trajectory	Agent	Replay	Occlusion	Re-judge
webarena.698	claude-3.7-sonnet	confirmed	refuted	true_success
visualwebarena.resized.843	claude-3.7-sonnet	confirmed	unverifiable	true_success
visualwebarena.293	gpt-4o-2024-11-20	confirmed	refuted	true_success
webarena.654	Llama-3.3-70B-Instruct	confirmed	refuted	true_success
webarena.377	gpt-4o-2024-11-20	confirmed	refuted	true_success
webarena.769	Qwen2.5-VL-72B-Instruct	unverifiable	unverifiable	true_success
visualwebarena.resized.314	Qwen2.5-VL-72B-Instruct	unverifiable	unverifiable	true_success
webarena.370	claude-3.7-sonnet	unverifiable	unverifiable	true_success
webarena.763	claude-3.7-sonnet	confirmed	unverifiable†	true_success

H POST-HOC ANALYSES ADDED AFTER INTERNAL REVIEW

Every number here is post hoc, computed from the committed single-writer artifacts by `review_additions.py`; none is claimed.

Label convention of Table 1. Precision counts the 38 label-consistent reported rows of 41 reports and FPR the 31 consistent expert-false rows; 3 reported rows carry a mixed label. Under the certificate's convention (§3.3), which counts them as errors, the base rule reads 75.6% precision (95% CI [59.6, 87.7]) and 29.4% FPR ([15.0, 47.5]), row 2 71.6% ([60.4, 81.1]) and 67.6% ([49.4, 82.7]); the gaps are +4.0 pp and +38.2 pp.

Task-level paired tests. The 96 test rows come from 56 tasks. A task-cluster bootstrap (2,000 draws) of the base rule's FPR minus row 2's gives -45.2 pp (95% CI [-70.0, -19.4], one-sided $p < 0.0005$); the exact sign-flip test over the 25 tasks holding a discordant row (19 rows) gives $p = 0.0019$. Against gpt-4o-mini-noaxtree, the judge the calibration split selects (test FPR 93.5%), the discordant rows are 1 rule-only and 23 judge-only (McNemar $p = 3.0 \times 10^{-6}$), the cluster-bootstrap difference -71.0 pp ([-89.7, -48.0], $p < 0.0005$) and the sign-flip $p = 7.6 \times 10^{-5}$.

The checker alone, and a judge gated by it. The checker h alone reports on 53.1% of test rows at 77.1% precision ([62.6, 88.0]), 35.5% FPR ([19.2, 54.7]) and 59.7% recall; McNemar against the base rule $p = 0.125$. Gating each Table 1 judge by h: row 2 reaches 85.0% ([70.1, 94.3]), 19.4% ([7.4, 37.5]), 54.8% recall, 43.8% coverage, $p = 1.000$ against the base rule; the calibration-selected judge 76.7%, 32.3% ([16.6, 51.4]), 53.2%, 47.9%, $p = 0.375$; the semantic view 81.0%, 25.8% ([11.8, 44.7]), $p = 1.000$; the evidence-only view 79.1%, 29.0% ([14.2, 48.1]), $p = 0.500$.

Order of charging the conjuncts. Of the 24 expert-false rows the base rule removes from the primary judge's accept set, the checker's share is 45.8% under the rule's own order (semantic, evidence-only, checker) and 83.3% when the checker is charged first; its Shapley value is 61.1%, the semantic view's 21.5%, the evidence-only view's 17.4%. Among the removed expert-true rows the checker's share runs 74.2%–80.6% (Shapley 77.4%).

A k-of-K vote at matched coverage. The artifact holds 11 complete recorded LLM judge configurations. Reporting when at least k agree, coverage falls monotonically in k from 100% ($k \leq 6$) to 75.0% ($k = 10$) and 61.5% at unanimity, and never reaches the rule's 42.7%. Unanimity has 71.9% precision and 51.6% FPR; rule minus vote is -29.0 pp (95% CI [-55.6, +0.0], McNemar $p = 0.064$). At $k = 10$: 72.9%, 61.3%, -38.7 pp ([-64.0, -13.1], $p = 0.0075$).

Unparsed teacher calls. On the test split 12 teacher calls did not parse, touching 10 rows (6 expert-false, 4 expert-true); the rule treats them as non-reports. Resolving all of them in favour of reporting gives 29.0% FPR ([14.2, 48.1]), 79.1% precision and 47.9% coverage, still +38.7 pp below row 2.

Recall intervals and calibration-split rates. Exact 95% recall intervals for Table 1: primary judge [94.2, 100.0], row 2 [84.2, 98.3], critic [26.6, 52.0], reward model [68.6, 89.6], semantic view

[84.2, 98.3], evidence-only view [84.2, 98.3], base rule [37.0, 63.0], deployed rule [0.0, 5.8]. On the 34 expert-false calibration rows the semantic view reports 67.6% ([49.4, 82.7]), the evidence-only view 73.5% ([55.6, 87.2]) and the base rule 44.1% ([27.1, 62.2]).

Latency. The teacher's recorded latency is its pooled wall time amortised over batches of 4 calls (60 calls, two views of each 30-trajectory pilot-pool record, median 1.48 s); the student's is per call (252 calls: two views of each test-split record plus the same pilot pool, median 9.50 s). The two call sets differ. Batch size, admission cap and GPU model are not recorded in the artifacts, so the ratio is descriptive.