
ForkAudit: From Accuracy Gains to Commit Risk in Meta-Agent Fork Selection

Anonymous Author(s)

Affiliation

Address

email

Abstract

1 Forking an agent is useful only if its supervisor can decide which branch to release.
2 We study this missing *pre-commit* decision: given two candidate events from the
3 same task, state, and execution prefix, can a meta-agent select the better-supported
4 continuation—and does one more quarantined observation help? FORKAUDIT
5 evaluates this question on AFTRAJ-FORK, a controlled set of task-unique strict-
6 prefix pairs whose candidates share the task, state, and execution prefix but differ
7 in outcome construction. Across three fixed open-weight auditors, one matched
8 observation improves the same pairwise gate by a mean of 6.3 percentage points.
9 This improvement is not clean evidence of semantic oversight: task-blind probes
10 and auditors recover the reference label from construction cues without access
11 to the task. Yet a no-shadow absolute scorer remains stronger, and selective
12 dual-order decisions retain high conditional error risk. FORKAUDIT therefore
13 treats the observation as an incremental benchmark gain, not a deployment-ready
14 safety capability, and evaluates commit gates by attacking their interpretation and
15 measuring risk–coverage rather than significance alone.

16 1 Introduction

17 Forking buys reversibility, not judgment. A meta-agent can checkpoint a worker, inspect alternative
18 continuations, and roll one back [Yu et al., 2026]; at the commit barrier, however, it must still decide
19 which branch deserves to survive. That prospective decision is rarely measured. Detecting later that a
20 trajectory failed [Zhang et al., 2026] cannot undo a released side effect.

21 The distinction matters because the supervisor acts before outcomes are known. It may compare two
22 next events directly, abstain, or spend compute to observe both forks inside a quarantine. In related
23 remediation settings, the cost of an incorrect intervention can exceed the cost of abstention, motivating
24 an explicit risk-constrained decision over whether to act [Dai et al., 2026]. Post-hoc attribution asks
25 *who* and *when* after a failed run [Liu et al., 2026a]; counterfactual trace audits compare behavioral
26 effects after paired executions [Zhou et al., 2026]. We instead ask the prospective merge question:
27 *which continuation should be released now?*

28 State-aligned forks provide unusually strong experimental control: both candidates share the task,
29 state, and full prefix. They also create two traps. First, extra evidence may improve a seeded
30 presentation without producing an order-invariant decision. Second, the pipeline that generated
31 reference and error branches may leave label-correlated textual signatures. Controlled state does not
32 control how candidates were constructed. A convincing evaluation must therefore measure the benefit
33 of additional evidence and try to break that benefit with task-blind and exact-order controls.

34 We build precisely this test. FORKAUDIT is a controlled evaluation protocol that compares the same
35 zero-shot gate before and after a bounded, quarantined observation, then challenges the apparent

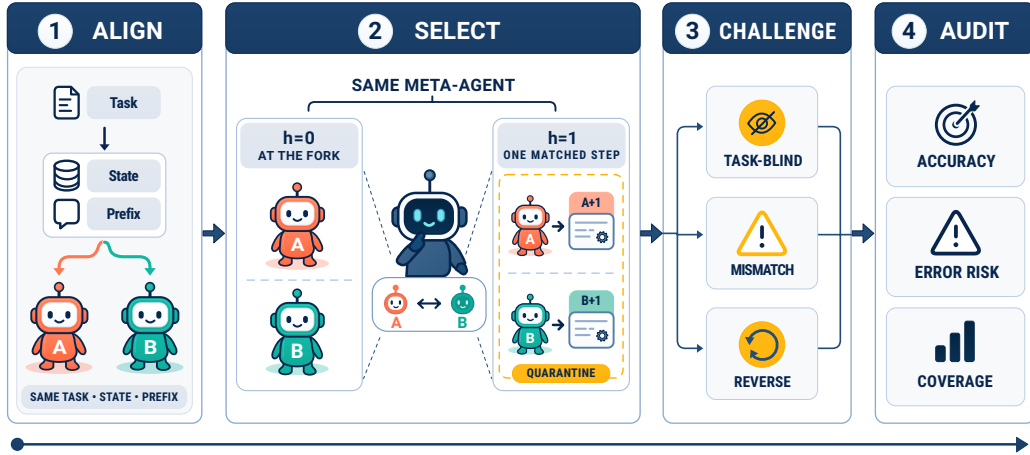


Figure 1: **FORKAUDIT audits a meta-agent’s pre-commit branch selection.** A strict-prefix pair holds task, state, and execution history fixed. The same meta-agent chooses between candidate branches at the fork ($h = 0$) or after one matched observation per branch in quarantine ($h = 1$). Three parallel challenges—task-blind input, mismatched continuation, and exact order reversal—probe alternative explanations for any gain. The resulting policy is judged jointly by accuracy, decisive-error commit risk, and coverage.

gain with task-blind, correspondence-disruption, and exact-order controls. AFTRAJ-FORK supplies 641 strict-prefix pairs across math, coding, and tool-using tasks. The result has a productive tension: one observation improves all three auditors, yet simple absolute scoring is stronger, construction artifacts nearly solve the benchmark, and task-blind auditors improve from another observation. Correspondence controls sharpen rather than erase the ambiguity: matched observations also beat carefully mismatched ones. The paper therefore asks not whether a gain exists, but what that gain licenses us to conclude.

Our contributions are:

- We isolate prospective fork selection as an operational oversight problem and derive AFTRAJ-FORK, 641 strict-prefix pairs that hold task, state, role, and execution history fixed at the commit barrier.
- We introduce FORKAUDIT, a validation envelope that measures a bounded-shadow intervention and then challenges it with stronger baselines, task-blind inputs, correspondence disruption, and exact order reversal.
- We identify a mixed result: matched observations add signal, but construction shortcuts, a stronger no-shadow policy, miscalibration, and high conditional error risk rule out a deployment-ready safety claim.

2 Related work

Reversible execution and meta-agent supervision. Meta-agents build, optimize, or supervise other agents [Xu and Tai, 2026, Lu et al., 2026, Hu et al., 2025]. Reversible runtimes expose checkpoint, replay, intervention, and fork operations [Yu et al., 2026]; exact execution-edit checks specify what permissions and results those operations must preserve [Zheng et al., 2026]. Transactional agent memory similarly stages evidence-bearing beliefs before commit and gates irreversible tool calls on in-flight state [Li et al., 2026a]. Controlled contrasts further show that planner-executor safety effects can conflate operational reframing, planner refusal, and approval-framed delegation [Liu et al.,

2026b]. Branching can also reduce training variance [He et al., 2026], hide speculative latency [Sui et al., 2026], or support backtracking [Wang et al., 2024]. We study the decision these mechanisms leave open: before outcomes are known, which state-aligned continuation should cross the commit barrier?

Trajectory auditing and verification. Execution provenance and evidence tracing connect agent actions, tool outputs, memory, and intermediate evidence to process-level auditability [Wang et al., 2026c]. AgentForesight detects the earliest decisive failure [Zhang et al., 2026]; Who&When Pro localizes the responsible agent and step [Liu et al., 2026a]; and Counterfactual Trace Auditing compares paired executions to characterize behavioral effects [Zhou et al., 2026]. We change the endpoint from explanation or attribution to a live accept/abstain decision. RealMistake’s naturally observed errors provide ecological validity but weaker state alignment [Kamoi et al., 2024]; our injected forks invert that trade-off.

Pairwise judging and selective oversight. Training-free verifiers combine fine-grained scores and repeated evaluation to rank candidate solutions [Kwok et al., 2026]; debate can likewise help a weaker judge choose between answers [Kenton et al., 2024]. We instead isolate a same-state, event-level decision under bounded pre-commit evidence. Pairwise preferences change under order swaps [Shi et al., 2025], and benchmark configuration choices can reverse pairwise model rankings [Li et al., 2026b]. SCOPE combines both orders with selective conformal calibration [Badshah et al., 2026]. Our dual-order rule is intentionally only a diagnostic fail-closed policy: we test whether order agreement actually produces usable commit risk rather than assuming invariance implies safety.

Evaluation shortcuts. Shortcut learning captures benchmark rules that fail under the intended distribution [Geirhos et al., 2020]. Harness search can generalize poorly beyond visible tasks [Wang et al., 2026b], while self-improving harnesses can install guardrails for failures refuted by an exact execution oracle when benign inputs contain familiar rule-shaped patterns [Wang et al., 2026a]. Agents can satisfy tests while violating a specification [Zhao et al., 2026]; and simulators or performance benchmarks can reward procedure-specific artifacts [Cao et al., 2026, Cui et al., 2026]. Monitoring probes trained on synthetic or off-policy outputs can likewise fail under distribution shift [Skapars et al., 2026]. Our complementary warning occurs before release: candidate construction can make the injected distribution easier to identify than the intended semantic error.

3 Problem setup

Let x be a user task and $\tau_{<t} = (e_0, \dots, e_{t-1})$ a worker’s execution trace at a reversible checkpoint. Two candidate events e_t^A and e_t^B define branches A and B from the same state. Exactly one retained event is the *verified-success reference*: it occurs in a recorded trajectory independently verified to complete x ; the other is the annotated earliest decisive error of a failed trajectory. This is a benchmark reference label, not a claim that every stochastic rollout after the event is safe. The meta-agent observes

$$o_0 = (x, \tau_{<t}, e_t^A, e_t^B)$$

and outputs $d \in \{A, B, \perp\}$ with confidence c , where $\perp$ denotes abstention. Ground-truth outcomes and mistake annotations are never rendered to the auditor.

A reversible runtime may additionally advance both candidates in a quarantined environment. At shadow horizon h , the observation o_h includes up to h subsequent events from each branch, but none of their side effects has been released. A policy may select a branch at any $h \leq H$ or exhaust its budget. We measure (i) reference-selection accuracy, (ii) *decisive-error commit risk*, the error rate among non-abstained decisions, (iii) coverage, and (iv) mean shadow steps. Following selective prediction [Geifman and El-Yaniv, 2019], risk–coverage curves prevent a gate from appearing safe by abstaining on every task.

4 Method

FORKAUDIT separates *what differs* from *how much more execution evidence to buy*, then tests whether the measured gain survives controls that remove task context, break continuation correspondence, or swap candidate order. It requires no model updates.

Paired divergence audit. The auditor sees the shared prefix once and the two next events in randomized order, then returns d, c and a short rationale. Unlike two absolute scores, the paired view exposes the only changed event while holding the task and state fixed. We also test a structured variant that names the exact divergent claim, derives an invariant grounded in the task or visible tool evidence, and predicts the immediate consequence of each choice. This separates the value of paired evidence from the value of a particular reasoning scaffold.

Bounded shadow fork. Before committing, the runtime may advance both branches by up to H events in quarantined copies of the state. The same gate then evaluates o_H , with the future observations explicitly marked as unreleased. We use fixed horizons so the gain can be attributed to a known evidence budget even when verbal confidence is poorly calibrated. Branch order is fixed within a run and exactly reversed in a paired robustness replicate. When either recorded branch ends early, we expose an equal matched horizon $\tilde{h} = \min(H, |\tau_{>t}^A|, |\tau_{>t}^B|)$ from both forks. This prevents continuation length alone from leaking the label. We report both the operational “up to H ” policy and an eligible-only estimate on pairs that contain the full budget.

Validation envelope. The intervention is credible only if it survives three attempts to explain the gain without semantic consequence reasoning. A *task-blind* condition removes the task, tools, and prefix; a *mismatched* condition preserves the construction side, domain, role, and event length but breaks task–shadow correspondence; and an *order* condition exactly swaps A/B. These tests separate input effect, correspondence sensitivity, and presentation stability. We then compare the surviving gate with a no-shadow absolute scorer and report coverage together with conditional decisive-error commit risk.

5 Experimental setup

Branch pairs. We derive pairs from AFraj-2K [Zhang et al., 2026]. For an injected-error trajectory with mistake index t , we search for a verified successful trajectory with the same domain, task, and answer. We retain the pair iff (i) the match is unique, (ii) both contain event t , (iii) their serialized prefixes through $t - 1$ are exactly equal, and (iv) their events at t differ while retaining the same responsible-agent role. This deterministic filter yields 641 unique tasks: 332 math, 243 coding, and 66 agentic. The source injection stream rejects recovered or unchanged errors and labels t by construction. A SHA-256 hash of the task assigns disjoint 15% calibration, 15% development, and 70% test partitions, leaving 442 locked-test tasks. Corpus and horizon counts appear in Table 1.

Conditions and models. We compare random choice, two-call absolute scoring, pairwise judging, a structured invariant audit, and one matched shadow step; a development-only sweep freezes the unstructured one-step gate before locked test. Auditors are 4-bit Qwen3-8B, Gemma-3-12B, and Llama-3.1-8B [Yang et al., 2025, Gemma Team, 2025, Grattafiori et al., 2024]. Decoding is greedy, model thinking is disabled, and all seeded A/B presentations receive an exact reversed-order replicate. Missing choices or confidences abstain. Model revisions, hardware, prompts, generation counts, latency, token cost, and the 10 truncated event fields across 8 primary pairs are documented in Appendix A.

Statistics. The task is paired by construction. Accuracy differences use paired task bootstrap intervals; proportions use Wilson intervals; and condition changes use exact McNemar tests. The frozen primary contrast holds the prompt fixed and compares pairwise judging with and without one matched observation. The pooled primary estimate is the mean effect across the three fixed auditors. Its interval resamples shared task clusters and averages auditors within task, so it quantifies task-sampling uncertainty for these auditors rather than model-family uncertainty. Confidence thresholds are chosen on calibration for at least 80% coverage and transferred unchanged to test. Reviewer-motivated task-blind, correspondence, and full-order controls were separately frozen before extension generation and remain post-hoc robustness analyses. All mechanism slices are descriptive and unadjusted for multiplicity.

Task-blind construction audit. Structural and character n -gram probes train on the 199 calibration and development pairs and evaluate once on all 442 test pairs. They see only the seeded divergent-event pair or only the first shadow-event pair—never the task, prefix, tools, continuation history,

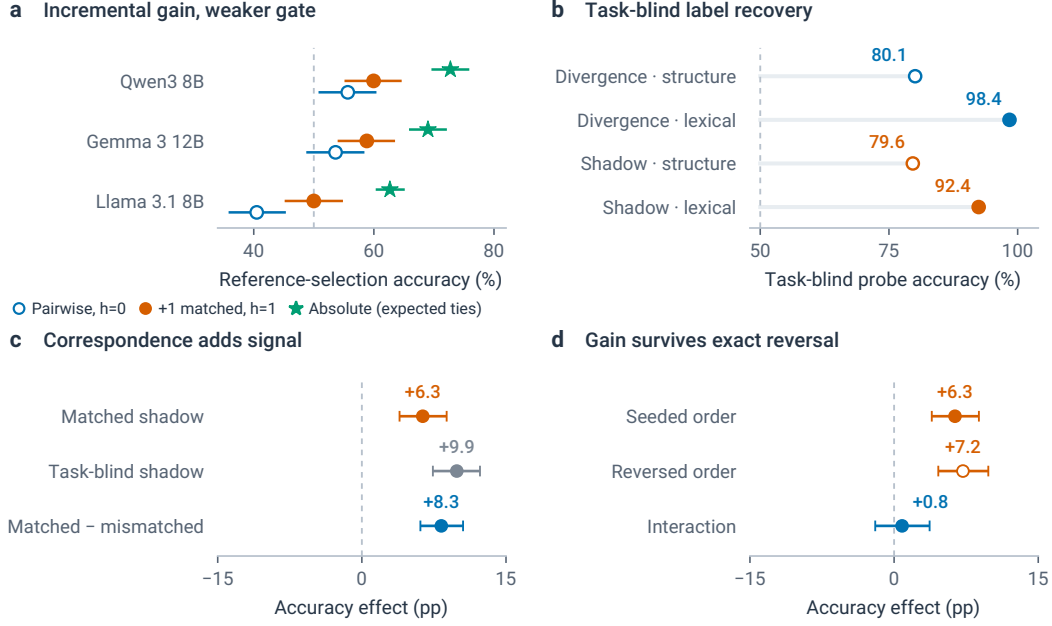


Figure 2: **The complete validation envelope separates an incremental benchmark gain from a trustworthy release policy.** (a) Pairwise judging, one matched observation, and the stronger no-shadow absolute baseline. (b) Task-blind recovery of labels from divergent and shadow-event text. (c) Matched, task-blind, and correspondence-disruption contrasts. (d) The one-observation effect under complete seeded and exactly reversed order, plus their interaction. Intervals are 95%; effect intervals resample shared task clusters while averaging the three auditors.

or annotation. The mismatch control selects cross-task donors within the same domain, role, and reference/error construction side by nearest length, then attaches them to the corresponding candidate. Exact features, matching diagnostics, and bootstrap procedure appear in Appendix A.

6 Results

The complete audit returns a mixture rather than a single success or failure. An added matched observation improves the frozen gate; construction cues can produce an even larger task-blind gain; and post-hoc correspondence controls provide evidence for additional signal beyond those cues. None of these results rescues the gate as an operational release policy.

6.1 Matched observations improve a weak gate

Across 442 locked-test tasks, one event raises pairwise accuracy by 6.3 points (task-cluster 95% CI: [3.9, 8.8]). The per-auditor gains are 4.3, 5.2, and 9.5 points, each with a positive supportive interval (Figure 2a). On the 383 pairs with a full event, the gain is 7.3 points ([4.5, 10.2]); on all 59 pairs without one, raw outputs, choices, confidences, and correctness are byte-identical. Added text therefore causes a real within-gate improvement. It does not make the gate competitive: exact uniform resolution of absolute-score ties reaches 62.7–72.7% expected accuracy, versus 50.0–60.0% for pairwise judging with one observation (Figure 2a).

6.2 Construction shortcuts coexist with correspondence signal

The verified branch is recoverable without the task. A character n -gram probe reaches 98.4% from the divergent events and 92.4% from the first shadow events alone (Figure 2b). More directly, removing task, tools, and prefix from the zero-shot auditors makes the one-observation gain larger, not smaller: 9.9 points ([7.4, 12.3]). The original improvement therefore cannot be interpreted as task-consequence reasoning alone.

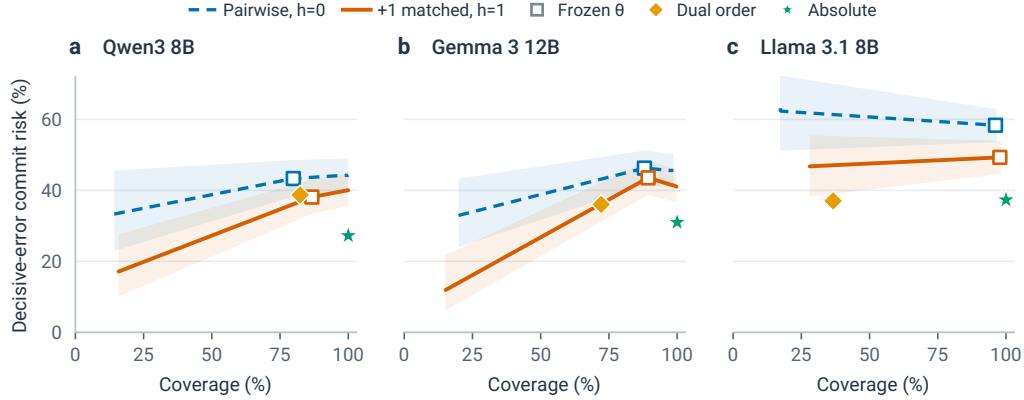


Figure 3: **Full risk-coverage curves expose the limits of selective filtering.** Each panel sweeps all integer verbal-confidence thresholds from 0 to 100 for pairwise judging at the fork ($h = 0$, blue) and after one matched observation ($h = 1$, vermillion). Shading shows pointwise Wilson 95% intervals for conditional decisive-error risk. Squares mark the calibration-frozen threshold on each curve; diamonds mark the $h = 1$ dual-order fail-closed policy; stars mark the no-shadow absolute scorer with exact expected tie-breaking. Curves use all 442 locked-test tasks and retain every distinct selected set; lower and farther right is better.

183 Construction cues are not sufficient to explain the entire effect. On 383 eligible tasks, matched
 184 observations beat cross-task observations from the same construction side and matched by domain,
 185 role, and length by 8.3 points ([6.1, 10.5]); all three auditor intervals are positive (Figure 2c; Table 4).
 186 This post-hoc result supports correspondence sensitivity beyond a strong construction shortcut, but
 187 does not causally identify semantic reasoning.

188 6.3 Aggregate robustness does not yield a release policy

189 The aggregate gain survives complete reversal: 6.3 points in the seeded order and 7.2 after swapping
 190 A/B, with an interaction of 0.8 points ([-2.0, 3.7]; Figure 2d).

191 The original 132-task attenuation does not generalize. Choice-level stability is much weaker: dual-
 192 order agreement is 82.4% for Qwen, 72.2% for Gemma, and 36.7% for Llama, while 10-bin ECE
 193 remains 35–42 points (Table 5).

194 At calibration-frozen thresholds, one observation lowers conditional decisive-error risk, but accepted
 195 decisions still fail on 38–49% of tasks. Requiring both orders to select the same semantic branch
 196 leaves 36.1–38.7% error risk at only 36.7–82.4% coverage. The no-shadow absolute scorer instead
 197 delivers 62.7–72.7% expected accuracy and 27.3–37.3% error risk at full coverage (Table 3). The
 198 intervention improves one gate; it does not produce the strongest or an operationally credible policy.

199 **Secondary diagnostics do not expand the claim.** A structured invariant prompt is unstable, and a
 200 second shadow step remains development-only (Appendix B). Non-multiplicity-adjusted domain,
 201 role, evidence-type, and probe-confidence slices remain available in the artifact rather than being
 202 promoted to manuscript evidence.

203 7 Discussion

204 The central distinction is among an *input effect*, a *correspondence effect*, and a *release capability*.
 205 The first two are supported: another matched event improves the same gate, and matching it to the
 206 correct task matters. The third is not. Construction alone carries strong label information, absolute
 207 scoring is stronger, and neither confidence nor order agreement produces low conditional error risk.
 208 Calling the intervention “useful evidence” is therefore defensible only inside this benchmark; calling
 209 it reliable semantic oversight is not.

210 This distinction changes how reversible-agent evaluations should be built. Extra execution is not
211 automatically better evidence: it can amplify whatever signal the data-generation process made
212 easiest to recognize. A convincing commit-gate benchmark should therefore treat task-blind probes,
213 correspondence-breaking controls, stronger no-execution baselines, exact order reversal, and risk-
214 coverage as one validation envelope. The next experimental step is not another slice of this injected
215 corpus, but origin-matched natural alternatives produced inside a live sandbox.

216 8 Limitations

217 **Construction shift and external validity.** Our controlled forks are derived from an existing corpus
218 rather than produced inside a live reversible runtime. Shadow events therefore replay recorded
219 continuations; they estimate selection quality and evidence cost, not runtime latency or sandbox
220 fidelity. More fundamentally, verified trajectories and injected-error trajectories have sharply different
221 textual signatures: a task-blind divergent-event probe nearly saturates the benchmark, and a shadow-
222 event probe reaches 92.4%. The task-blind zero-shot auditors also gain 9.9 points from an added
223 observation. The strict-prefix filter improves state alignment but does not remove this construction
224 shift. We expose the shortcut and restrict our conclusions to paired within-benchmark effects; a
225 deployment benchmark needs naturally occurring alternatives or interventions that match the origin
226 and style of both branch classes. The filter also reduces the agentic-domain sample and inherits the
227 source benchmark’s task and error distributions. A recorded verified-success trajectory is not a proof
228 that every future continuation from its first event will succeed or avoid harm.

229 **Evidence and auditor scope.** Many coding forks expose explicit test feedback within one shadow
230 step. We stratify that evidence, but the benchmark does not establish equal gains when only subtle
231 behavioral cues are available. We evaluate reproducible 4-bit 8–12B auditors rather than proprietary
232 frontier models, so absolute levels may not transfer even though the comparisons remain paired within
233 each model. The task-blind, correspondence-disruption, and full-order controls were motivated by
234 reviewer feedback after the original locked study. They were independently frozen before extension
235 generation and are reported regardless of outcome, but remain post-hoc robustness evidence rather
236 than new confirmatory tests. The positive matched-minus-mismatched result provides evidence of
237 sensitivity to task correspondence under our donor construction; it does not isolate semantic reasoning,
238 because mismatching may also alter discourse coherence or other features not captured by domain,
239 role, class side, and nearest length.

240 **Policy safety.** Pairwise judging also underperforms exact expected uniform-tie absolute scoring, so
241 our evidence supports the incremental value of an observation within one gate, not the superiority
242 of direct comparison. Dual-order fail-closed decisions retain 36–39% conditional error risk, and
243 verbal confidence is severely miscalibrated; neither order agreement, a score threshold, nor recorded
244 sandbox evidence is a safety guarantee.

245 9 Responsible-use statement

246 Branch selection can reduce the chance that a supervising agent releases a decisive worker error, but
247 it can also create unwarranted confidence in automated oversight. False accepts may expose users
248 or third parties to the effects of a decisive-error branch; false rejects can suppress legitimate work,
249 and systematic differences across tasks or languages could distribute both harms unevenly. Shadow
250 execution additionally consumes compute and must not be confused with containment unless tool
251 side effects are technically blocked. We therefore recommend fail-closed sandboxing, explicit human
252 approval for irreversible or high-impact actions, retention of both branch traces and audit rationales,
253 domain-specific calibration, task-blind construction probes, exact order-reversal tests (or dual-order
254 ensembles), and periodic testing for disparate error rates. Monitors trained on injected failures should
255 be revalidated on naturally occurring, origin-matched branches before deployment. The released
256 artifact contains recorded research trajectories and prompts, not credentials or live-action tools.

10 Conclusion

Forking is not an oversight capability until the release decision is tested. Holding the pairwise prompt fixed, one matched observation improves all three auditors by 6.3 points, and the effect survives complete exact reversal. What that gain means is less simple. Task-blind auditors improve even more, and lexical probes recover both divergent and shadow labels, showing that construction cues can create an apparent benefit without task reasoning. Yet matched observations also beat carefully mismatched ones by 8.3 points; this post-hoc contrast is consistent with correspondence contributing information beyond those cues. The evidence is mixed, not absent: the current benchmark cannot cleanly identify semantic consequence reasoning, and the resulting policy is weak. A no-shadow absolute scorer beats every paired shadow gate; dual-order abstention still leaves 36–39% conditional error risk. Future meta-agent evaluations should therefore treat task-blind shadow controls, correspondence disruption, stronger operational baselines, no-evidence identity checks, complete order reversal, and risk–coverage reporting as part of the intervention itself—not as optional diagnostics added after a positive headline result.

References

- Sher Badshah, Ali Emami, and Hassan Sajjad. SCOPE: Selective conformal optimized pairwise LLM judging, 2026. URL <https://arxiv.org/abs/2602.13110>.
- Hongliu Cao, Ilias Driouich, and Eoin Thomas. Beyond task completion: Revealing corrupt success in LLM agents through procedure-aware evaluation, 2026. URL <https://arxiv.org/abs/2603.03116>.
- Yingyun Cui, Yi Xie, Piaohong Wang, Jiawei Ma, Bo Liu, and Liangliang Cao. PERFOPT-Bench: Evaluating coding agents on software performance optimization, 2026. URL <https://arxiv.org/abs/2607.07744>.
- Chengxiao Dai, Zhaokun Yan, Chenjun Lei, Qiao Li, and Luyan Zhang. Safe remediation as risk-constrained intervention decision in microservice systems, 2026. URL <https://arxiv.org/abs/2607.20005>.
- Yonatan Geifman and Ran El-Yaniv. SelectiveNet: A deep neural network with an integrated reject option. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 2151–2159. PMLR, 2019. URL <https://proceedings.mlr.press/v97/geifman19a.html>.
- Robert Geirhos, Jørn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A. Wichmann. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2:665–673, 2020. doi: 10.1038/s42256-020-00257-z. URL <https://doi.org/10.1038/s42256-020-00257-z>.
- Gemma Team. Gemma 3 technical report, 2025. URL <https://arxiv.org/abs/2503.19786>.
- Aaron Grattafiori et al. The Llama 3 herd of models, 2024. URL <https://arxiv.org/abs/2407.21783>.
- Bowei He, Yankai Chen, Xiaokun Zhang, and Xue Liu. Branching policy optimization: Sandbox-native language agent reinforcement learning, 2026. URL <https://arxiv.org/abs/2607.14171>.
- Shengran Hu, Cong Lu, and Jeff Clune. Automated design of agentic systems, 2025. URL <https://arxiv.org/abs/2408.08435>.
- Ryo Kamoi, Sarkar Snigdha Sarathi Das, Renze Lou, Jihyun Janice Ahn, Yilun Zhao, Xiaoxin Lu, Nan Zhang, Yusen Zhang, Ranran Haoran Zhang, Sujeeth Reddy Vummanthala, Salika Dave, Shaobo Qin, Arman Cohan, Wenpeng Yin, and Rui Zhang. Evaluating LLMs at detecting errors in LLM responses. In *Conference on Language Modeling*, 2024. URL <https://arxiv.org/abs/2404.03602>.

304 Zachary Kenton, Noah Y. Siegel, János Kramár, Jonah Brown-Cohen, Samuel Albanie, Jannis Bulian,
305 Rishabh Agarwal, David Lindner, Yunhao Tang, Noah D. Goodman, and Rohin Shah. On scalable
306 oversight with weak LLMs judging strong LLMs, 2024. URL <https://arxiv.org/abs/2407.04622>.
307

308 Jacky Kwok, Shulu Li, Pranav Atreya, Yuejiang Liu, Yixing Jiang, Chelsea Finn, Marco Pavone,
309 Ion Stoica, and Azalia Mirhoseini. LLM-as-a-Verifier: A general-purpose verification framework,
310 2026. URL <https://arxiv.org/abs/2607.05391>.

311 Xiaoyang Li, Yiqi Wang, Haohui Lu, Zhi Chen, Mo Li, Pingan Song, Mingkai Zheng, and Taotao
312 Cai. MemTX: Transactional belief commit for stateful agent memory, 2026a. URL <https://arxiv.org/abs/2607.23929>.
313

314 Yanhang Li, Zhichao Fan, and Zexin Zhuang. SafetyRepro: Configuration-conditional rank instability
315 on alignment benchmarks, 2026b. URL <https://arxiv.org/abs/2605.25492>.

316 Jiale Liu, Huajun Xi, Shaokun Zhang, Yifan Zeng, Tianwei Yue, Chi Wang, Jian Kang, Qingyun Wu,
317 and Huazheng Wang. Who&When Pro: Can LLMs really attribute failures in AI agents?, 2026a.
318 URL <https://arxiv.org/abs/2607.09996>.

319 Lifei Liu, Haoran Yu, Xiaochong Jiang, Su Wang, Pin Qian, and Yihang Chen. Operational reframing
320 and approval-framed delegation in multi-agent LLM safety, 2026b. URL <https://arxiv.org/abs/2607.07097>.
321

322 Xinyu Lu, Tianshu Wang, Pengbo Wang, Zujie Wen, Zhiqiang Zhang, Jun Zhou, Boxi Cao, Yaojie
323 Lu, Hongyu Lin, Xianpei Han, and Le Sun. The meta-agent challenge: Are current agents capable
324 of autonomous agent development?, 2026. URL <https://arxiv.org/abs/2606.04455>.

325 Lin Shi, Chiyu Ma, Wenhua Liang, Xingjian Diao, Weicheng Ma, and Soroush Vosoughi. Judging
326 the judges: A systematic study of position bias in LLM-as-a-Judge. In *Proceedings of the 14th
327 International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-
328 Pacific Chapter of the Association for Computational Linguistics*, pages 292–314. Asian Federation
329 of Natural Language Processing and Association for Computational Linguistics, 2025. doi: 10.
330 18653/v1/2025.ijcnlp-long.18. URL <https://aclanthology.org/2025.ijcnlp-long.18/>.

331 Adrians Skapars, Nathalie Maria Kirch, Samuel Dower, Ekdeep Singh Lubana, and Dmitrii
332 Krasheninnikov. The impact of off-policy training data on probe generalisation. In *Findings
333 of the Association for Computational Linguistics: ACL 2026*, pages 22673–22729. Association
334 for Computational Linguistics, 2026. doi: 10.18653/v1/2026.findings-acl.1139. URL
335 <https://aclanthology.org/2026.findings-acl.1139/>.

336 Yifan Sui, Han Zhao, Rui Ma, Zhiyuan He, Hao Wang, Jianxun Li, Kaiqiang Xu, Kai Chen, and
337 Yuqing Yang. Parallelizing tool execution and LLM generation for low-latency agent serving, 2026.
338 URL <https://arxiv.org/abs/2603.18897>.

339 Haoyu Wang, Tao Li, Zhiwei Deng, Dan Roth, and Yang Li. Devil’s advocate: Anticipatory reflection
340 for LLM agents, 2024. URL <https://arxiv.org/abs/2405.16334>.

341 Su Wang, Pin Qian, Yifan Lin, Jingzhou Xu, Yihang Chen, Xiaochong Jiang, Lifei Liu, and Haoran
342 Yu. Phantom guardrails: When self-improving agent harnesses fix failures that never happened,
343 2026a. URL <https://arxiv.org/abs/2607.13083>.

344 Yike Wang, Huaisheng Zhu, Zhengyu Hu, Yige Yuan, Zhengyu Chen, Shakti Senthil, Hannaneh
345 Hajishirzi, Yulia Tsvetkov, Pradeep Dasigi, and Teng Xiao. Rethinking the evaluation of harness
346 evolution for agents, 2026b. URL <https://arxiv.org/abs/2607.12227>.

347 Yiqi Wang, Jiaqi Zhang, Taotao Cai, Zirui Liu, Qingqiang Sun, Zequn Sun, Zhangkai Wu, Manqing
348 Dong, Mingkai Zheng, Xuefei Yin, and Yanming Zhu. From agent traces to trust: A survey of
349 evidence tracing and execution provenance in LLM agents, 2026c. URL <https://arxiv.org/abs/2606.04990>.
350

351 Andy Xu and Yu-Wing Tai. Meta-agent: From task descriptions to verified multi-agent systems,
352 2026. URL <https://arxiv.org/abs/2605.25233>.

- 353 An Yang et al. Qwen3 technical report, 2025. URL <https://arxiv.org/abs/2505.09388>.
- 354 Simon Yu, Derek Chong, Ananjan Nandi, Dilara Soylu, Jiuding Sun, Christopher D. Manning, and
 355 Weiyan Shi. Shepherd: Enabling programmable meta-agents via reversible agentic execution
 356 traces, 2026. URL <https://arxiv.org/abs/2605.10913>.
- 357 Boxuan Zhang, Jianing Zhu, Zeru Shi, Dongfang Liu, and Ruixiang Tang. AgentForesight: Online
 358 auditing for early failure prediction in multi-agent systems, 2026. URL <https://arxiv.org/abs/2605.08715>.
- 360 Bingchen Zhao, Dhruv Srikanth, Yuxiang Wu, and Zhengyao Jiang. SpecBench: Measuring reward
 361 hacking in long-horizon coding agents, 2026. URL <https://arxiv.org/abs/2605.21384>.
- 362 Yusheng Zheng, Xiaoyu Song, Yanpeng Hu, Lebin Cheng, Yuxi Huang, and Wei Zhang. When can
 363 agents safely checkpoint, fork, restore, and merge? exact checking for execution edits, 2026. URL
 364 <https://arxiv.org/abs/2608.22928>.
- 365 Xiaolin Zhou, Jinbo Liu, Li Li, Ryan A. Rossi, and Xiyang Hu. Counterfactual trace auditing of
 366 LLM agent skills, 2026. URL <https://arxiv.org/abs/2605.11946>.

367 A Reproducibility details

368 **Pair construction.** The released builder indexes verified-success trajectories by the tuple (domain,
 369 task text, gold answer). For every injected-error trajectory it keeps the example only when exactly
 370 one indexed trajectory has the same sequence of event dictionaries before the annotated step and a
 371 different event at that step. It then asserts uniqueness of both task strings and pair identifiers. The
 372 split is the first eight hexadecimal digits of SHA-256(task), modulo 100: values 0–14 are calibration,
 373 15–29 development, and 30–99 test. Unit tests recompute these invariants for all 641 released pairs.

Table 1: AFTRAJ-FORK strict-prefix pairs. $h \geq k$ counts pairs for which both recorded forks contain at least k post-divergence events.

Domain	All	Cal.	Dev.	Test	$h \geq 1$ / $h \geq 2$
Math	332	48	50	234	332 / 264
Coding	243	38	41	164	158 / 67
Agentic	66	11	11	44	66 / 61
Total	641	97	102	442	556 / 392

374 **No label access.** The derived JSONL retains the gold answer, source trajectory IDs, responsible
 375 agent, and mistake rationale as source and evaluation metadata. Prompt construction uses only the
 376 task, listed tools, shared events, candidate events, and the matched number of shadow events. Tests
 377 assert that source IDs and mistake rationales never enter a rendered prompt. A/B order depends only
 378 on pair ID and the public seed; replicate 1 is the exact reversal of replicate 0. The original 132-task
 379 order diagnostic selects 44 pairs per domain by sorting the SHA-256 hash of `seed: :pair_id`; this
 380 label-blind subset and rule were fixed before the original locked evaluation. The post-hoc extension
 381 adds the exact complement, yielding both orders for all 442 test tasks.

382 **Shortcut-probe details.** The shorter-event rule counts the same serialized role and field headers
 383 used by the learned probes. The structural probe standardizes the difference between the two events’
 384 character, whitespace-token, digit, newline, and nonempty-field counts. The lexical probe fits a 3–5
 385 character `char_wb` TF-IDF vocabulary on the calibration/development events only, with minimum
 386 document frequency 2 and at most 50,000 features, then subtracts branch B from A. Both use scikit-
 387 learn logistic regression with $C = 1$, 2,000 maximum iterations, and seed 20260828. The responsible
 388 role is exactly matched within every pair. A representation stress test refits the same lexical probe
 389 after deleting role names and serialization headers. It reaches 96.4% from action/content values alone,
 390 whereas a thought-only probe reaches 57.5%, localizing most of the shortcut to externally visible
 391 event text.

Table 2: Task-blind lexical-probe representation stress test on all 442 test pairs. Intervals are task-bootstrap 95% CIs. Every row refits on the same 199 calibration/development tasks.

Divergent-event representation	Accuracy	95% CI
Full serialized event	98.4	[97.1, 99.5]
Values only (thought/action/content)	96.4	[94.6, 98.0]
Action/content values only	96.4	[94.6, 98.0]
Thought values only	57.5	[52.7, 62.2]

Conditions. Absolute judging produces a 0–100 risk and accept/reject verdict for each candidate in a separate model call; the lower-risk candidate wins and equal scores abstain. The random-tie baseline reuses these exact calls and resolves only equal valid scores with a seeded coin. Pairwise judging sees both events in one prompt. The invariant variant additionally requests the divergent claim/action, a task-grounded invariant, and the immediate consequence. Shadow conditions append equal-length recorded events from quarantined forks. No condition receives the source outcome or annotation.

Parsing and failures. The requested control fields are choice and confidence (or risk and verdict). A deterministic parser first decodes a JSON object and then, if a rationale contains an illegal escape, recovers only those named scalar control fields. We report malformed-schema and missing-control rates separately. A missing control field is an abstention and is incorrect for overall accuracy. Exact prompts and parser tests are included in the artifact.

Evidence taxonomy. For analysis only, the first matched shadow step is assigned a label-blind coarse type. Events containing tool strings for tests or answer verification are “explicit verifier/test”; another environment event is “other environment”; the remainder are “agent action.” The full 641-pair corpus contains 233, 8, and 315 such pairs respectively, plus 85 pairs with no matched future event. This breakdown tests whether a gain is confined to explicit verification feedback.

Artifact and licensing. The preprocessing, prompts, resumable MLX runner, analysis, figures, and table generator are included. Our code is MIT-licensed. AFTraj-2K and the derived trajectory text are attributed to the source authors and distributed under CC BY 4.0. Downloaded model weights retain their upstream licenses and are not redistributed. The experiments require no networked action tools and replay recorded research trajectories only.

Frozen inference artifacts. Before the first LLM auditor generation on the test split, we froze the pair file, prompts, parser, runner, output auditor, threshold rule, model revisions, and environment lockfile in a machine-readable SHA-256 manifest. The exact MLX model revisions are 545dc425 (Qwen3-8B), 86cc6a8d (Gemma-3-12B), and 90215b22 (Llama-3.1-8B). No prompt or condition is selected from auditor test performance.

Inference and extension accounting. All auditors use MLX-LM 0.31.3 on an Apple M2 Pro with 32 GB unified memory, greedy decoding, disabled model thinking, and a 512-token output cap. Tasks and event fields are deterministically capped at 2,400 and 1,800 characters; the primary locked-test prompt truncates 0 task strings and 10 event fields across 8 pairs. The locked study uses 2,652 generations per auditor; the original order diagnostic uses 528; and the frozen reviewer extension uses 1,887. Mismatched donor events have median and 90th-percentile absolute length gaps of 0 and 4 characters. All three model reports predate the source corpus release. Across auditors, the locked study required 3.35–6.87 recorded generation hours and the original reversal diagnostic required 0.56–1.24 hours. Per decision, stored prompts and outputs contain 1,541–1,603 tokens for two-call absolute scoring, 854–892 for pairwise scoring, and 961–1,016 for pairwise scoring with one shadow event.

B Development-only horizon selection

The final development sweep uses the same 51 task-hash-selected pairs for all three auditors. The first matched event improves the cross-model pairwise gate by 14.4 points (task-cluster 95% CI [7.2,

22.9)]; every per-model change is positive. A second event adds a smaller 4.6 points [0.7, 8.5] while mean used evidence rises from 0.84 to 1.51 events, and no individual one-to-two-step McNemar test is significant. Two future events are also available for only 263/442 test pairs, versus 383/442 for one event. Following the predeclared preference for the smallest operational intervention, we froze one step before test and report no two-step test result.

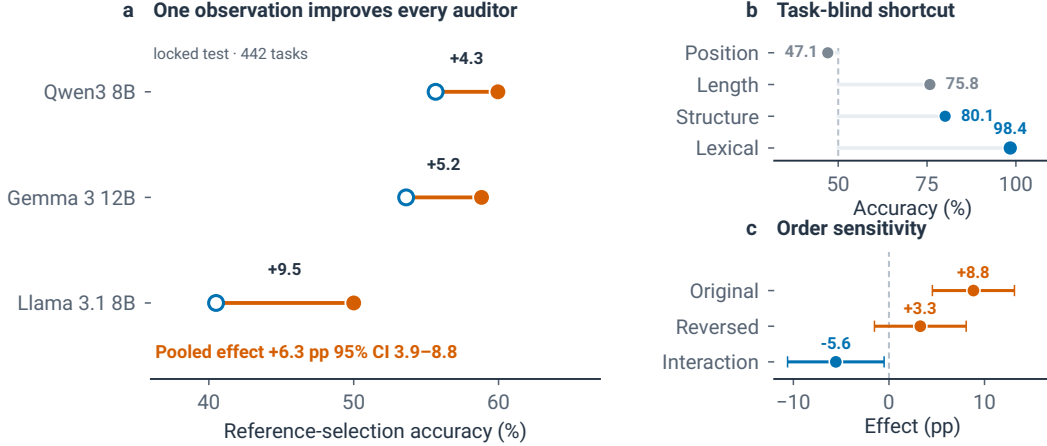


Figure 4: **Original locked result and initial stress tests.** (a) One matched observation improves all three auditors. (b) Task-blind divergent- event probes identify the reference label without task context. (c) The original 132-task audit shows attenuation under exact reversal. The main-text validation envelope adds the stronger baseline, shadow-only probe, correspondence disruption, and complete-test reversal.

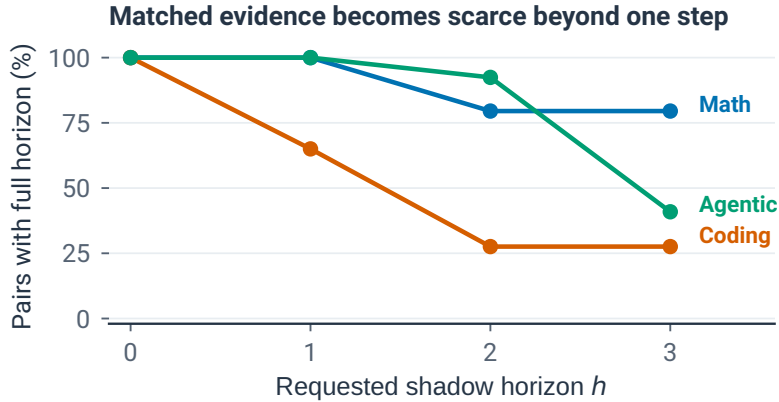


Figure 5: **Why the evidence budget stops at one.** Fraction of strict pairs that support each matched horizon. Coding continuations become scarce after the first observation, while a second step also adds a smaller development-only gain.

438 C Core audit details

439 To keep the appendix focused, we retain only results that directly support the paper’s operational
 440 comparison, per-auditor robustness claims, or calibration diagnosis. Precomputed exploratory slices
 441 and implementation diagnostics remain in the artifact for reproducibility.

Table 3: Operational baselines on all 442 locked-test tasks. Accuracy counts abstentions as incorrect; error risk is conditional on committing. Absolute ties are resolved in exact expectation by a uniform coin. The dual-order policy commits only when both presentations select the same semantic branch. Calls count auditor generations; shadow is the mean quarantined observation depth per branch.

Auditor	Policy	Acc.	Cov.	Error risk	Calls	Shadow/branch
Qwen3 8B	Absolute, uniform ties	72.7	100.0	27.3	2	0
	Pairwise + 1 shadow	60.0	100.0	40.0	1	0.87
	Dual-order fail-closed	50.5	82.4	38.7	2	0.87
Gemma 3 12B	Absolute, uniform ties	69.0	100.0	31.0	2	0
	Pairwise + 1 shadow	58.8	99.8	41.0	1	0.87
	Dual-order fail-closed	46.2	72.2	36.1	2	0.87
Llama 3.1 8B	Absolute, uniform ties	62.7	100.0	37.3	2	0
	Pairwise + 1 shadow	50.0	98.2	49.1	1	0.87
	Dual-order fail-closed	23.1	36.7	37.0	2	0.87

Table 4: Reviewer-control contrasts by auditor. These per-model intervals resample tasks and are supportive rather than multiplicity-adjusted confirmatory tests.

Auditor	Contrast	Effect (pp) [task-bootstrap 95% CI]
Qwen3 8B	Task-blind: $h=0 \rightarrow 1$	+11.8 [+7.9, +15.8]
	Matched minus mismatched	+9.4 [+6.0, +12.8]
	Seeded: $h=0 \rightarrow 1$	+4.3 [+0.5, +8.1]
	Reversed: $h=0 \rightarrow 1$	+3.4 [-0.7, +7.5]
Gemma 3 12B	Task-blind: $h=0 \rightarrow 1$	+10.0 [+6.1, +13.6]
	Matched minus mismatched	+11.2 [+7.8, +14.9]
	Seeded: $h=0 \rightarrow 1$	+5.2 [+1.4, +9.0]
	Reversed: $h=0 \rightarrow 1$	+7.0 [+3.4, +10.6]
Llama 3.1 8B	Task-blind: $h=0 \rightarrow 1$	+7.9 [+4.5, +11.3]
	Matched minus mismatched	+4.2 [+1.0, +7.3]
	Seeded: $h=0 \rightarrow 1$	+9.5 [+5.9, +13.3]
	Reversed: $h=0 \rightarrow 1$	+11.1 [+7.0, +15.2]

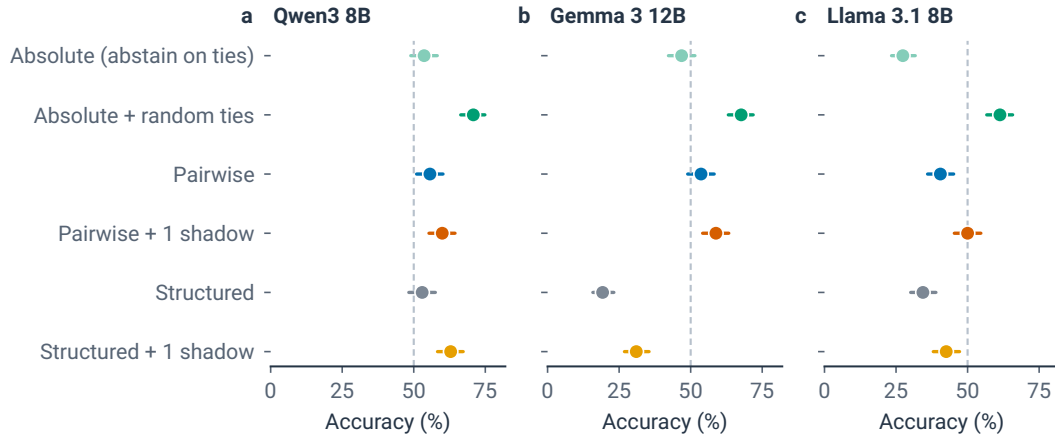


Figure 6: **All locked-test conditions.** Points show reference-selection accuracy and horizontal bars show Wilson 95% intervals. The dashed line is random choice. Accuracy counts abstentions as incorrect; vermillion marks the frozen one-shadow condition.

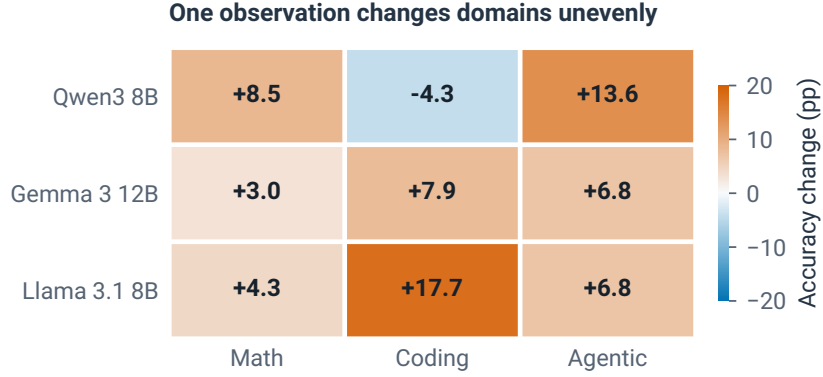


Figure 7: **Where one observation changes accuracy.** Each cell is the one-shadow minus zero-shadow pairwise difference for one auditor and domain. Aggregated domain effects are positive, but Qwen’s coding slice is a visible exception.

Table 5: Confidence calibration among decisions with a parseable choice. ECE is reported in percentage points; Brier is on the unit scale.

Auditor	Gate	n	Conf.	Sel. acc.	ECE	Brier
Qwen3 8B	Pairwise	442	94.45	55.66	38.79	0.40
	Pairwise + 1 shadow	442	95.10	59.95	35.15	0.36
Gemma 3 12B	Pairwise	436	95.42	54.36	41.07	0.42
	Pairwise + 1 shadow	441	95.22	58.96	36.39	0.37
Llama 3.1 8B	Pairwise	427	83.44	41.92	41.99	0.43
	Pairwise + 1 shadow	434	85.46	50.92	35.23	0.38