Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Does Provenance Add Value Beyond Matched Skill Guidance?

Haocheng Lu

haocheng409@gmail.com

New York University

Research Article

Keywords: LLM agents, agent evaluation, provenance, skill distillation, tool-using agents, controlled evaluation, agent benchmarks, context effects

Posted Date: September 1st, 2026

DOI: https://doi.org/10.21203/rs.3.rs-10846054/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: The authors declare no competing interests.

Does Provenance Improve Terminal Task Success Beyond Matched Skill Guidance?

Anonymous authors
Paper under double-blind review

Abstract

Agent skills often bundle task guidance with source and evidence metadata. We ask whether those fields improve terminal task success once ordinary guidance is held fixed. We compare paired tasks with identical guidance and a separate placebo matched on provider-reported input tokens. Across 250 task-ID-held-out tasks with visible-template overlap, BESD (Evidence-Bound Semantic Distillation) did not outperform SkillGenMatched. The payload-matched difference was -3.2 percentage points (95% task-bootstrap interval [-6.0,-0.4]; 125-cluster interval [-7.2,0.0]). In a separate exploratory 100-task provider-token-matched amendment, the estimate was +2.0 points (task interval [-2.0,+6.0]; cluster interval [-2.0,+7.0]). In a constructed exploratory opportunity cell, a large contrast appeared only when provenance determined the oracle action. The apparent effect therefore changes with the matching regime and task design. Provenance utility must be identified with controlled comparisons, rather than inferred from a bundled skill score.

1 Introduction

Tool-using agents increasingly rely on external skills assembled from traces, source files, documentation and prior runs. These artifacts can include the source event supporting a clause, the condition under which it applies, or an evidence reference that a person can inspect. Such metadata may aid auditing, but its effect on terminal success is usually measured together with guidance, context length and execution protocol.

This coupling creates an attribution problem. A no-skill baseline tests whether the whole package helps; it does not test whether provenance adds value to the same guidance. Extra fields can also change serialization and context length. We therefore ask two linked questions: how does the contrast change as these nuisance factors are controlled, and when does provenance carry information that can change the correct action?

We answer with a paired intervention at the skill-payload slot. For each task, both arms receive the same ordinary guidance, tools, student configuration and execution policy. One payload contains provenance fields; the other contains a fixed-width placeholder. We first compare these payloads, then evaluate a separate control matched on provider-reported input tokens. Responses are captured per arm and replayed by canonical request hash, so a fixture never supplies a response from another request.

The payload-matched estimate was -3.2 points, whereas a separate provider-token-matched cell gave +2.0 points, with intervals spanning zero. A constructed opportunity cell produced a large contrast only when provenance changed the executable oracle action. Semantic controls do not isolate provenance correctness as the source of the original contrast. Family, lifecycle and frozen/live analyses show where estimates vary; they do not identify a single mechanism.

We contribute an evaluation protocol for separating whole-skill utility from provenance utility. The protocol treats context matching as part of the estimand and makes dependence and decision relevance visible to the reader.

2 Related work

Prior work has shown that reusable experience can improve an agent, but it rarely identifies which fields of a structured artifact provide that utility. SkillGen (SkillGen, 2025) and Trace2Skill (Trace2Skill, 2026) extract transferable instructions from trajectories, while AgentDistill (AgentDistill, 2025) and related memory systems package guidance outside the student weights. ReAct, Reflexion and Voyager likewise treat traces, feedback or skills as reusable experience (ReAct (Yao et al., 2022), Reflexion (Shinn et al., 2023), Voyager (Wang et al., 2023)). Our question is narrower: after the ordinary guidance is fixed, what task utility remains in its provenance fields? We do not re-run the reported baselines because their splits and provider protocols are not shared with this study.

Provenance standards such as W3C PROV (W3C, 2013), OpenTelemetry (OpenTelemetry, 2024) and OpenInference (OpenInference, 2024) make source relationships inspectable. Tracing provenance in reasoning distillation (Liu et al., 2025) instead follows where distilled sentences or decisions originated. We study a different endpoint: provenance is an inference-time payload, and utility is measured by executable terminal success. Auditability and task performance are related goals, but one does not imply the other.

The attribution issue also appears in agent evaluation. AI Agents That Matter (Kapoor et al., 2025) shows how bundled harness choices can obscure the source of an apparent improvement. Work on context position and length, including Lost in the Middle (Liu et al., 2024) and Context Length Alone Hurts LLM Performance Despite Perfect Retrieval (Du et al., 2025), motivates measuring tokenization, formatting and lifecycle rather than treating them as fixed background conditions.

Benchmarks such as AgentBench (Liu et al., 2023), WebArena (Zhou et al., 2023), OSWorld (Xie et al., 2024), ToolSandbox (Lu et al., 2024) and τ-bench (Yao et al., 2024) provide increasingly realistic interactive settings. Their realism complements, rather than replaces, an attribution design: a score difference can still combine guidance, metadata, context and harness effects. Our protocol supplies that missing comparison.

3 Study design

3.1 Estimands and registered contrasts

Let $Y_i(z) \in \{0, 1\}$ denote the terminal oracle outcome for task i under condition z. The confirmatory estimand is the finite-population paired contrast

$$\Delta_{P-M} = \frac{1}{250} \sum_{i=1}^{250} \{Y_i(P) - Y_i(M)\},$$

where P is BESD and M is SkillGenMatched. `Base` provides a whole-package reference, while `EqualLength` `ShuffledHEG` is a nuisance/placebo reference. The confirmatory question is only Δ_{P-M}. A preregistered positive-gain rule is reported separately from the descriptive estimate and its compatibility boundary.

The provider-token-matched amendment defines a separate exploratory estimand, Δ_{P-T}. Here T is a placebo payload matched to the provider-reported first-turn input length of P, without meaningful provenance semantics. It uses 100 task IDs and a declared ± 2-token tolerance. The task subset and capture protocol differ from the primary cell, so the estimate is reported alongside Δ_{P-M} rather than pooled with it.

The nine-arm mechanism cell (100 tasks) is exploratory. `CorrectProvenance` is the pre-specified semantic contrast; the remaining six augmented payloads are nuisance controls. Holm step-down adjustment is reported across the seven non-reference contrasts. No mechanism arm is confirmatory.

Two secondary interactions were fixed before inspecting treatment outcomes: the difference between the retry-family contrast and the mean contrast of the other nine families, and a lifecycle difference-in-differences comparing the same 22 tasks with the retry wrapper enabled and disabled. Neither interaction conditions on a post-treatment retry count. A frozen/live comparison is kept as a separate sensitivity estimand.

3.2 Benchmark, splits and task families

The source-local benchmark contains ten scenario families, spanning workspace writes, issue conflicts, memory re-exposure, cancellation, search, permissions, retries, compaction, handoff and duplicate creation. A deterministic generator creates 834 tasks: 417 support, 167 development and 250 task-ID-held-out tasks. Each task starts from a manifest-defined state and is scored by a versioned executable terminal-action oracle.

The held-out set contains 125 observationally equivalent pairs. They are kept because repeated visible structure is part of the attribution problem, but inference treats the pair as a dependence cluster in sensitivity analysis. The visible-fingerprint audit identified 140 held-out rows whose fingerprints also occur in the support partition. Here, "held out" means task-ID held out, not template novel. Family counts and the complete split are fixed before provider outcomes are inspected.

The oracle-action audit defines the primary benchmark's boundary. None of the 125 retained pairs changes its accepted action as a function of the hidden provenance label. The primary cell can therefore test whether redundant metadata changes behaviour, but not whether an agent can use provenance to choose between otherwise indistinguishable actions. We address that separate question in an opportunity experiment.

3.3 Intervention and manipulation checks

The intervention has a simple contract. Within each task, the two arms receive the same ordinary guidance, tools, skill position, student configuration and execution policy. Only the evidence-bearing slot differs: P carries source/evidence and applicability fields, whereas M carries a fixed-width placeholder. This is payload matching, not provider-token matching.

The underlying representation is an ordered Harness Evidence Graph with source, configuration, runtime, tool, memory, agent, wire and provider-hidden nodes. Edges record event identity, order, applicability and source spans, or an explicit unknown marker. The compiler admits a clause only when its observable evidence is closed; it never infers inaccessible provider context. Provider usage reports mean input lengths of 2,400 tokens for P and 1,424 for M, with P longer on 249 of 250 tasks. The primary contrast therefore includes any residual context-length or serialization effect, which the token-matched amendment examines separately.

As a fixture-local manipulation check, observable graph edges were compared with the executable source oracle. Precision and recall were both 1.0 for the observable edge set. This calibration does not claim real-application provenance accuracy or substitute for a blind human gold standard.

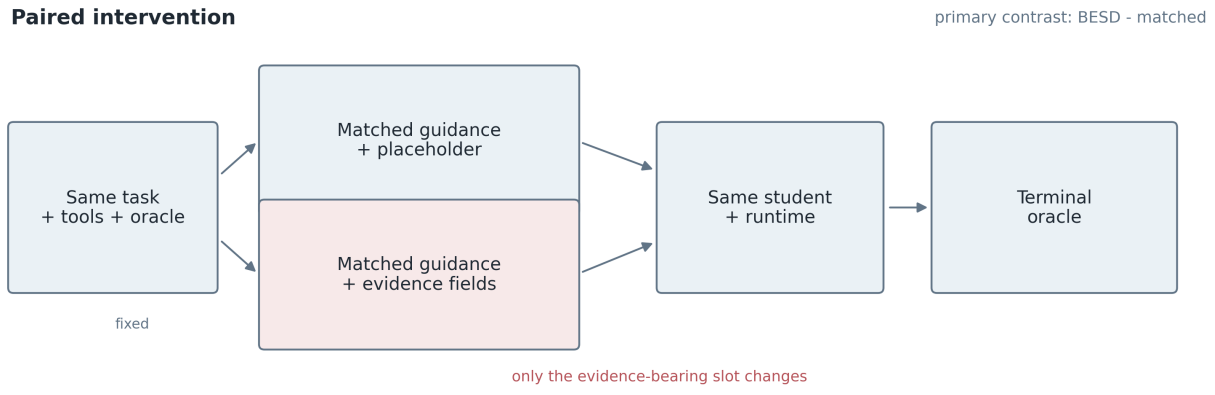


Figure 1: Paired identification design. Each task uses the same ordinary guidance, tools, student runtime and terminal oracle in both arms; the evidence-bearing slot is the planned difference.

3.4 Capture, replay and provider boundary

The adapter records the client request boundary, canonicalizes structured fields, removes credentials and binds each response to a request hash. Replay fails closed on a missing fixture, schema mismatch or endpoint/model mismatch. Each arm has its own captured response fixture under the same recorded configuration; a frozen fixture is not a shared counterfactual response. The teacher, verifier and student are identified by the role labels `sol`, `terra` and `luna`. Exact provider identifiers and hidden server context remain in private hash-bound records.

The estimand is consequently fixture-conditioned and source-local. Live calls are analysed only in the predeclared drift subset and are kept separate from replay outcomes. Server-side tokenization, hidden prompts, dynamic routing and provider rewrites are outside the observation boundary.

3.5 Statistical analysis

All assigned task-arm rows remain in the intention-to-treat denominator. An assigned provider failure counts as a failure. We report absolute rates, paired differences, repairs and regressions, and exact two-sided McNemar tests. Task-level intervals use 10,000 PCG64 bootstrap draws. A dependence sensitivity resamples the 125 equivalence clusters, while family intervals use equal-family resampling and a ten-family label randomization test.

The lifecycle interaction uses an exact sign-flip test over non-zero paired differences and a task-cluster bootstrap interval. Mechanism p-values are Holm-adjusted and exploratory. We also report a fixture-specific effect-size boundary, including the one-sided upper endpoint and compatibility of 0, 2, 5 and 10 percentage-point gains. The token-matched amendment applies the same paired outcome analysis after its provider-reported length check.

4 Results

4.1 Primary payload-matched comparison

In the primary cell, BESD succeeded on 38.8% of tasks versus 42.0% for `SkillGenMatched`; `Base` and the shuffled reference succeeded on 36.8% and 41.6%.

The paired estimate was

$$\widehat{\Delta}_{P-M} = -0.032 \text{ (-3.2 percentage points)},$$

with a 95% task-bootstrap interval of [-0.060,-0.004]. There were 3 repairs and 11 regressions. The exact two-sided McNemar test gave $p = 0.0574$; the directional loss probability ($p = 0.0287$) is exploratory.

The cluster-aware estimate was also -3.2 points, with a 95% interval of [-0.072,0.000] over 125 equivalence clusters. Removing template-overlapping rows left 110 task IDs and a descriptive -1.8-point estimate (bootstrap interval [-4.5,0.0]). Because the cluster interval includes zero, this fixture provides neither a robust positive-gain result nor evidence of a general negative effect.

Table 1: Primary terminal success by arm in the frozen paired cell (n=250 task IDs).

Arm	Success	Interpretation
Base	36.8%	no-skill reference
SkillGenMatched	42.0%	primary comparator
BESD	38.8%	provenance-bearing treatment
EqualLengthShuffledHEG	41.6%	nuisance/placebo reference

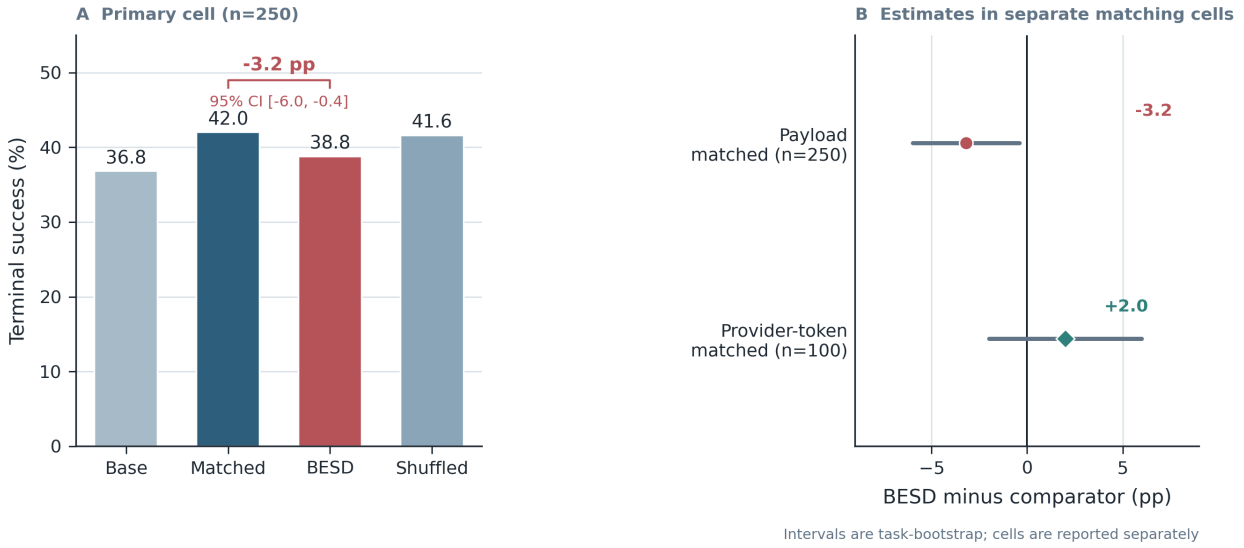


Figure 2: Primary terminal success in the 250-task paired cell. The highlighted bracket marks the paired BESD minus SkillGenMatched contrast; `Base` and `Shuffled` are reference conditions.

Figure 2 displays the arm-level rates and paired contrast. The one-sided 95% task-bootstrap upper endpoint for $P - M$ is -0.8 points. A stable +5-point gain is incompatible with this frozen fixture's interval and its family-cluster sensitivity interval. That compatibility boundary is local to this task population and response regime.

BESD was 2.0 points above `Base` but 3.2 points below `SkillGenMatched`. The first contrast measures the whole skill package; the second targets the incremental contrast. A separate amendment tests the latter after matching provider-reported input length.

4.2 Provider-token-matched comparison

The provider-token-matched amendment was exploratory. It compared BESD with a placebo whose first-turn input was matched to the BESD target. Across 100 task IDs, the declared target averaged 825.8 tokens (median 802.0), and the provider-reported input averaged 825.4 tokens (median 801.0). All tasks were within the declared ±2-token tolerance.

The paired estimate was +2.0 points, with 3 repairs and 1 regression. Its task-bootstrap interval was [-2.0,+6.0], its equivalence-cluster interval was [-2.0,+7.0], and the exact two-sided McNemar test gave $p = 0.625$. The interval includes zero, so this cell does not establish a positive benefit. In this separately sampled, provider-token-matched cell, the estimate was positive but imprecise rather than negative.

Table 2: Provider-token matching and exploratory token-matched control (n=100 task IDs).

Measure	Value
Task IDs (n)	100
Target first-turn input (mean / median)	825.8 / 802.0 tokens
Actual first-turn input (mean / median)	825.4 / 801.0 tokens
Exact match	30.0%
Within ±2 tokens (declared)	100.0%
Within ±5 tokens	100.0%
BESD minus ProviderTokenMatchedPlacebo	+2.0 pp
Task-bootstrap 95% interval	[-2.0,+6.0] pp
Equivalence-cluster 95% interval	[-2.0,+7.0] pp
Repairs / regressions (BESD-only / placebo-only)	3 / 1
Exact McNemar two-sided p	0.625

Token matching addresses reported input length only. Hidden context, tokenization, serialization and response randomness remain unobserved. We therefore treat this as an exploratory, fixture-conditioned comparison rather than a causal comparison between cells.

4.3 Mechanism controls

The 100-task mechanism amendment compares several provenance payloads with `SkillGenMatched`. All contrasts are exploratory; the table reports their paired differences:

Table 3: Exploratory mechanism contrasts relative to SkillGenMatched (n=100 tasks).

Payload	Difference (pp)	95% task interval
CorrectProvenance	-4.0	[-10.0, 1.0]
EvidenceOnly	-4.0	[-9.0, 0.0]
MetadataOnly	-6.0	[-12.0, -1.0]
WrongProvenance	-4.0	[-10.0, 1.0]
ShuffledProvenance	-4.0	[-10.0, 1.0]
IrrelevantProvenance	-4.0	[-10.0, 2.0]
CorrectNoFormatting	-3.0	[-9.0, 3.0]

The Holm-adjusted p-value for `CorrectProvenance` was 1.0. Correct, wrong, shuffled and irrelevant payloads moved similarly in this cell. The controls do not isolate provenance semantics as the source of the primary contrast; context length, serialization and lifecycle remain viable alternatives. This result does not imply that correct and incorrect provenance are equivalent in an unconstrained application.

4.4 Family and execution sensitivity

The family audit asked whether the primary contrast was distributed across scenarios or concentrated in one workflow. The largest difference was in `retry_wrapper`: its 22 tasks had 22.7% success with BESD and 59.1% with `SkillGenMatched`, a -36.4-point contrast. The other nine families had point contrast zero. Omitting this family removed the observed primary loss; omitting any other family left a negative overall contrast.

The task-weighted family interval was [-63.6,-4.5] points. The equal-family ten-cluster interval was [-10.9,0.0], and the ten-family label randomization test gave $p = 0.10$. We interpret this as family-associated heterogeneity, not evidence for a retry mechanism.

Table 4 gives the complete family audit. The success rates are included to make the concentration visible rather than relying on a selected subgroup.

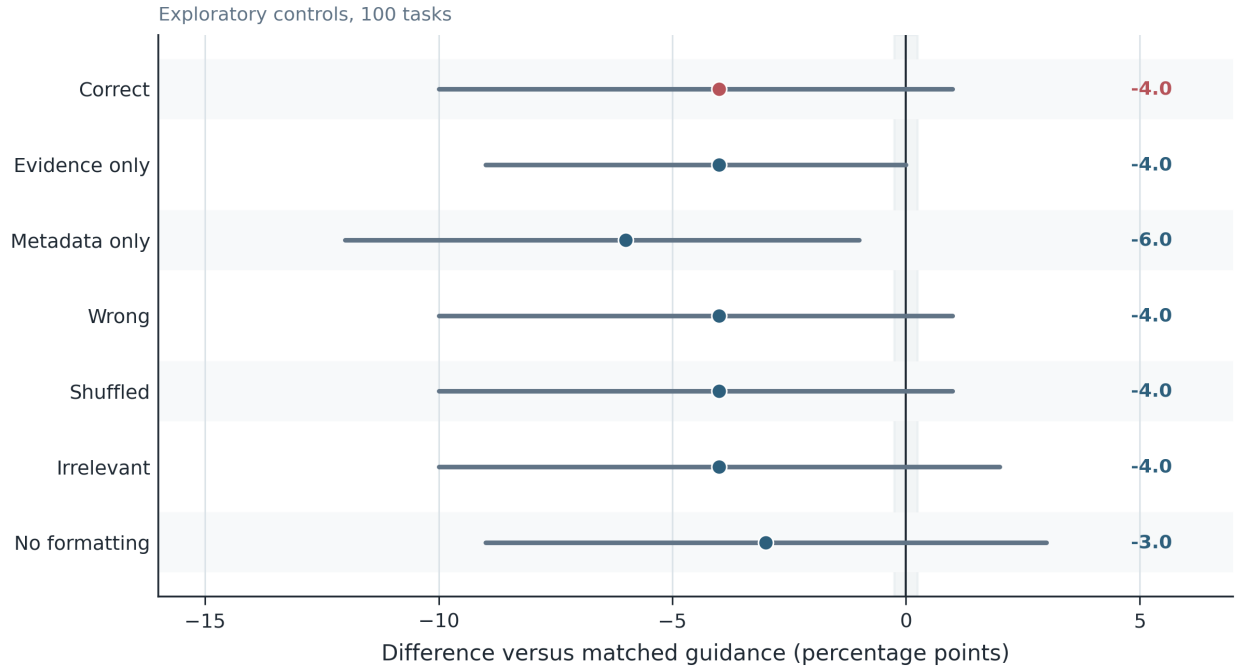


Figure 3: Exploratory mechanism contrasts relative to SkillGenMatched in 100 tasks. Points are paired estimates and horizontal lines are 95% task-bootstrap intervals.

Table 4: Family-stratified terminal success and paired contrasts in the primary cell.

Scenario family	n	SkillGenMatched	BESD	Difference
async_cancel	28	100.0%	100.0%	+0.0 pp
compaction_restore	28	0.0%	0.0%	+0.0 pp
duplicate_issue	18	0.0%	0.0%	+0.0 pp
empty_search	30	100.0%	100.0%	+0.0 pp
handoff	16	0.0%	0.0%	+0.0 pp
issue_conflict	24	0.0%	0.0%	+0.0 pp
memory_reexposure	28	0.0%	0.0%	+0.0 pp
permission_denied	22	0.0%	0.0%	+0.0 pp
retry_wrapper	22	59.1%	22.7%	-36.4 pp
workspace_write	34	100.0%	100.0%	+0.0 pp

The direct lifecycle intervention did not reproduce that family pattern. With the wrapper disabled, the contrast was -31.8 points; with it enabled, the contrast was -13.6 points. The interaction was +18.2 points (95% task-cluster interval [-27.3,+63.6], exact sign-flip $p = 0.5773$). The interval was wide and the direction opposed the proposed mechanism. This intervention does not establish a lifecycle effect or mediation.

In the predeclared live-drift subset, $P - M$ was +1.0 point (task-bootstrap interval [-4.0,+7.0]). The same-task live-minus-frozen change was +5.0 points (task interval [-5.0,+16.0], family-cluster interval [0.0,+14.0]). Live and frozen regimes consequently represent different estimands.

The pre-treatment difficulty proxy used prompt length, required action size and initial-state leaves, and was frozen before outcome analysis. Its tertile contrasts were 0.0, -9.6 and 0.0 points ($n = 84, 83, 83$), with a middle-tertile interval of approximately [-18.1,-1.2] points. These are descriptive strata. An observed request/tool-count proxy gave 0.0, -6.0 and -3.6 points, but it is post-treatment and exploratory. The audits do not support reducing the family pattern to a monotone difficulty effect.

Across the paired primary rows, 142 tasks failed under both arms, 94 succeeded under both, 11 were regressions and 3 were repairs. `no_valid_tool_trace` accounted for 145 SkillGenMatched and 153 BESD failures; request-cap failures occurred only in the no-skill reference. These transition counts, rather than a post-treatment failure subset, define the primary paired comparison.

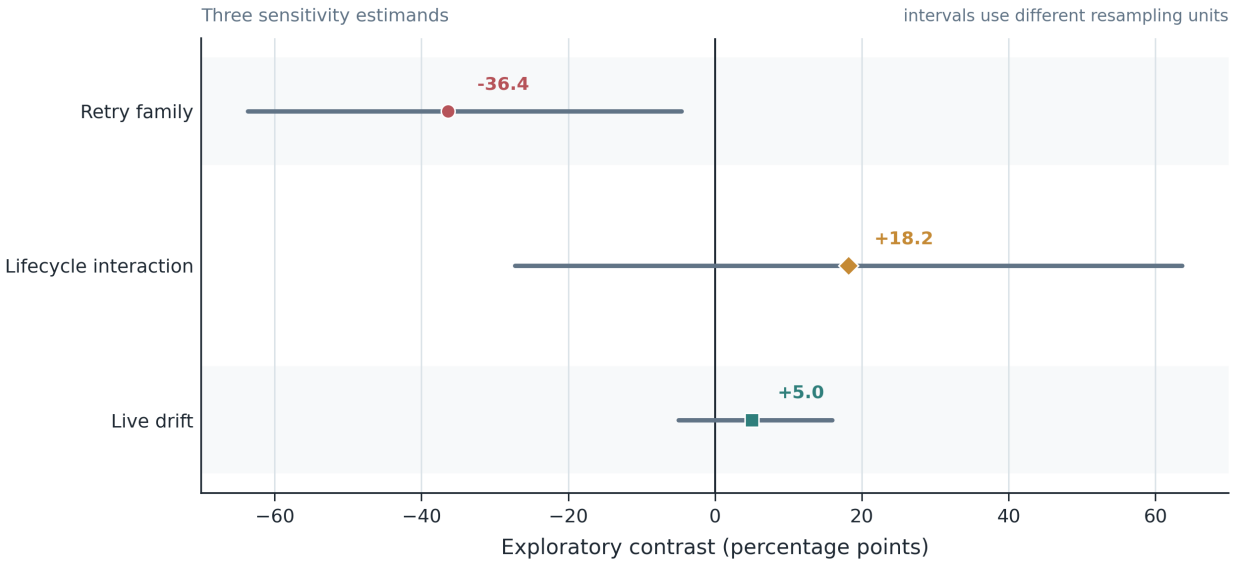


Figure 4: Evaluation-sensitive diagnostics. Points are paired estimates in percentage points; intervals use the family-stratified, task-cluster and task-bootstrap units stated in the adjacent text.

4.5 Decision-relevant opportunity boundary

The opportunity design asks a different question: can provenance help when it changes the action accepted by the oracle? The design-level audit contains 100 pairs with identical visible state, ordinary guidance and tools within each pair, but different provenance labels and oracle actions.

The later amendment selected 25 pairs and froze an action-only response contract. Outcome-blind selection was not demonstrated, so the result is exploratory. In archived replay, `CorrectProvenance` succeeded on 50/50 tasks, `SkillGenMatched` on 25/50, and both shuffled and wrong provenance on 0/50. The paired contrasts were +50 and +100 points, respectively. Because the correct payload supplied the oracle action in this constructed cell, the result marks a construct boundary rather than general student behaviour. `Wrong Provenance` is a decision-value placebo whose donor value conflicts with the target task; replay uses parsed actions and outcomes, not byte-for-byte provider responses.

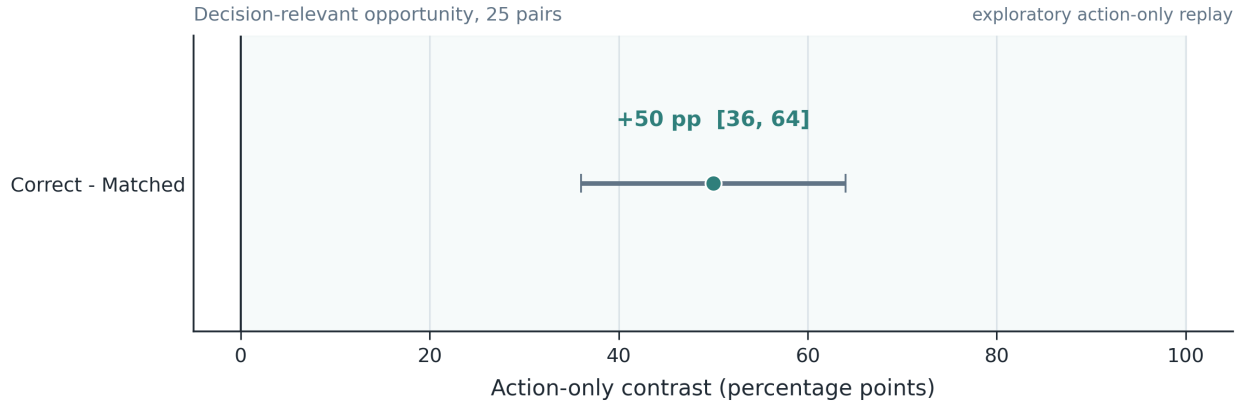


Figure 5: Decision-relevant opportunity design. The action-only replay reports the CorrectProvenance minus SkillGenMatched contrast for 25 pairs (50 tasks).

4.6 Scope diagnostics

The real-terminal TUA-Bench amendment was attempted as an external check, but 72 of 200 assigned cells lacked provider rows. The strict intersection contains only six task pairs (difference 0.0, interval [-50.0,+50.0] points). We treat this as infrastructure evidence rather than external validation.

A 12-task secondary-model cell has 4/12 successes in each core arm and is underpowered. Two 200-task secondary-runtime captures reuse the same task IDs; 30 of 400 terminal outcomes changed between captures, and only 84/400 logical-request hash lists were identical. Their stored runtime revisions cannot be reanalysed by the current analyzer, so they remain run-sensitivity diagnostics.

Table 5: Status and estimand of secondary diagnostic regimes.

Diagnostic regime	Contrast	Interpretation
Frozen primary (250 tasks)	-3.2 pp	confirmatory payload-matched estimate
Provider-token-matched amendment (100 tasks)	+2.0 pp	exploratory context-matched estimate
Mechanism amendment (100 tasks)	Correct vs matched -4.0 pp	exploratory semantic/control contrasts
Live drift subset (50 tasks, two replicates)	live-minus-frozen +5.0 pp	secondary regime sensitivity
Opportunity amendment (50 tasks)	Correct vs matched +50.0 pp	separate action-only construct boundary

5 Discussion

The main lesson is methodological: the apparent value of provenance changes as the comparison becomes more controlled. BESD was 2.0 points above `Base`, 3.2 points below `SkillGenMatched`, and 2.0 points above the provider-token-matched placebo in a separate cell. The last interval spans zero. This sign change does not establish a benefit, but it shows why bundled skill comparisons cannot identify provenance utility.

These comparators answer different questions. `Base` versus BESD measures whether the complete skill package helps. `SkillGenMatched` versus BESD measures the incremental contrast under payload matching. The semantic controls moved similarly in the observed cell. Together with the token imbalance, this leaves context length, serialization and provider regime as viable explanations for the original negative contrast.

The constructed action-contract cell defines a boundary condition: provenance can matter when the oracle action depends on it. `CorrectProvenance` succeeded on 50/50 tasks, versus 25/50 for matched guidance and 0/50 for shuffled or wrong provenance. This exploratory result does not establish general model utility.

The family, lifecycle and live-drift analyses reinforce the same methodological lesson. Effects varied by scenario family and response regime, while the direct lifecycle intervention did not support the proposed wrapper mechanism. These diagnostics do not select a single explanation; they identify conditions in which an attribution estimate deserves stress testing.

For practitioners, provenance may still improve audit, debugging, ownership and human review when terminal success is unchanged. For evaluators, the study supports a simple practice: publish a guidance-matched comparator, report provider-token balance, predeclare dependence-aware sensitivity analyses, and state whether the task oracle makes provenance decision-relevant.

6 Limitations, ethics and reproducibility

The study observes the client request boundary, not provider-hidden context, routing, server-side rewrites or tokenizer internals. The primary design has a 69% mean input-token imbalance; the provider-token-matched amendment narrows that nuisance but uses a different task subset and capture protocol. Each task-arm has one canonical capture, so within-cell stochastic variance is not separately estimable. The task-ID split contains repeated visible templates, and the primary pairs contain no provenance-dependent oracle action. The opportunity cell is intentionally constructed, small and not demonstrably outcome-blind in its final selection. Lifecycle, external-terminal and secondary-model amendments are diagnostic. No independent blind human annotation was collected; edge precision and recall are fixture-local checks.

The accompanying behavioral projection bundle is locally table-replayable. It contains redacted outcome projections, task manifests, hash-bound source code, generated table checks and a detached manifest digest. The verifier runs the reproducer in a platform-specific OS sandbox that denies network and file writes, then materializes only a fixed output allowlist after validating the child's envelope. On platforms without the required sandbox it fails closed. This is a platform-scoped containment measure, not external certification. Because the platform profile deliberately retains broad read and execute access needed by the interpreter, it is not a confidentiality boundary against a malicious bundle. Verification should therefore run in a disposable account or container, with no secrets mounted. Raw provider responses, hidden reasoning, credentials and host paths are excluded under the provider/data-release contract. The public artifact is a behavioral projection, not a raw-response distribution; its detached digest, source hashes and network-denied replay status are reported separately from any future provider-term or authorization decision. This restriction limits raw response reproduction, but it does not prevent regeneration of the published behavioral tables.

The preregistration and deviation records distinguish confirmatory from exploratory analyses. All assigned failures remain in the intention-to-treat denominator. Model roles, request settings and seeds are reported to the extent permitted by the provider contract. Provenance can improve accountability, but it can also expose source identifiers or induce misplaced trust; the release redacts such material and makes no claim that more metadata is universally safer.

7 Conclusion

Provenance utility cannot be read from a bundled skill score. In the payload-matched primary cell, BESD was -3.2 points relative to `SkillGenMatched`; in a separate provider-token-matched cell, it was +2.0 points with uncertainty spanning zero. A constructed opportunity cell showed a large contrast only when provenance changed the oracle action. Evaluations should therefore match ordinary guidance, audit provider-reported tokens, and report dependence-aware uncertainty. These recommendations remain bounded to the source-local, fixture-conditioned setting studied here.

References

AgentDistill. Agentdistill, 2025. URL https://arxiv.org/abs/2506.14728.

Y. Du et al. Context length alone hurts llm performance despite perfect retrieval. In *Findings of the Association for Computational Linguistics: EMNLP*, 2025. doi: 10.18653/v1/2025.findings-emnlp.1264.

Sayash Kapoor et al. Ai agents that matter. *Transactions on Machine Learning Research*, 2025. URL https://mlanthology.org/tmlr/2025/kapoor2025tmlr-ai/.

K. Liu et al. Where did this sentence come from? tracing provenance in llm reasoning distillation, 2025. URL https://openreview.net/pdf?id=Eem0IYVORm.

Nelson F. Liu et al. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173, 2024. doi: 10.1162/tacl_a_00638.

Xiao Liu et al. Agentbench: Evaluating llms as agents, 2023. URL https://arxiv.org/abs/2308.03688.

Y. Lu et al. Toolsandbox: A stateful, conversational, interactive evaluation benchmark for llm tool use capabilities, 2024. URL https://arxiv.org/abs/2408.04682.

OpenInference. Openinference specification, 2024. URL https://github.com/Arize-ai/openinference.

OpenTelemetry. Opentelemetry documentation, 2024. URL https://opentelemetry.io/docs/.

Noah Shinn et al. Reflexion: Language agents with verbal reinforcement learning, 2023. URL https://arxiv.org/abs/2303.11366.

SkillGen. Skill generation for language-model agents. Project repository, 2025. URL https://github.com/yccm/SkillGen.

Trace2Skill. Trajectory-derived skills for tool-using agents, 2026. URL https://arxiv.org/abs/2603.25158.

W3C. Prov-dm: The prov data model, 2013. URL https://www.w3.org/TR/prov-overview/. W3C Recommendation.

Guanzhi Wang et al. Voyager: An open-ended embodied agent with large language models, 2023. URL https://arxiv.org/abs/2305.16291.

Tianbao Xie et al. Osworld: Benchmarking multimodal agents for open-ended tasks in real computer environments, 2024. URL https://arxiv.org/abs/2404.07972.

Shunyu Yao et al. React: Synergizing reasoning and acting in language models, 2022. URL https://arxiv.org/abs/2210.03629. arXiv preprint arXiv:2210.03629.

Shunyu Yao et al. τ-bench: A benchmark for tool-agent-user interaction in real-world domains, 2024. URL https://arxiv.org/abs/2406.12045.

Shuyan Zhou et al. Webarena: A realistic web environment for building autonomous agents, 2023. URL https://arxiv.org/abs/2307.13854.