RIGHT VERDICT, WRONG REASON: COUNTERFACTUAL ATTRIBUTION REVEALS SHORTCUT LEARNING IN LLM EVALUATORS

Anonymous authors
Paper under double-blind review

ABSTRACT

As large language models take over the role of evaluator—scoring outputs, modeling rewards, and checking policy compliance—trustworthy evaluation is emerging as a problem as hard as capability itself: an evaluator's verdicts gate agent actions, human review, and training data, yet the evaluator is validated only by the labels it emits. Deployed evaluators are, moreover, continually *updated*: evidence pipelines are corrected, rules are amended, and the scope of a rule is widened or narrowed. When a verdict flips after an update, standard evaluation checks only whether the new label is correct—never whether it changed *for the right reason*. This failure mode, *right verdict, wrong reason*, is invisible to accuracy and is precisely what auditability requires ruling out. We make it measurable. Factoring an evaluator's inputs into typed components (evidence, rules, scope), the 2^3 old/new hybrid configurations form a *counterfactual table* whose inclusion-minimal component swaps define an exactly computable gold attribution for any verdict change; we prove correctness of enumeration, bound attribution multiplicity, establish a worst-case query lower bound for black-box verification, and separate *predicting* an attribution from *verifying* one. We instantiate the framework in REASONBENCH, a benchmark of 19,520 cases with executable rule-based evaluators, exact gold attributions, and paired consistency controls. The findings are cautionary. Fine-tuned models reach 98.4% exact attribution accuracy in distribution, yet merely reordering semantically equivalent input sections changes their attribution about half the time; under compositional shift, verdict accuracy holds at 93.8% while attribution accuracy on changed cases collapses to 7.2%; and a pre-specified hypothesis that supervising the full counterfactual table would improve attribution is rejected—the richer target consistently hurts. Accuracy is a poor proxy for faithful attribution; reliable attribution requires executing counterfactuals rather than trusting model outputs.

1 INTRODUCTION

Large language models are no longer only the systems being evaluated; they are increasingly the systems doing the evaluating. LLM judges score model outputs (Zheng et al., 2023), reward models shape policies during RLHF (Ouyang et al., 2022), and automated policy checkers decide whether an agent's action is released, whether a transaction is escalated, and which examples are recycled into training data. In these roles an evaluator's label is not a passive measurement: it allocates permission and attention.

This creates a dilemma the field increasingly recognizes: building trustworthy evaluation may be as hard as building capable models. A learned judge is typically validated by the labels it emits—so its failure modes are exactly the ones labels cannot see, and every downstream consumer of its verdicts, from release gates to reward signals to training-data filters, silently inherits them. Most responses to this dilemma work from the benchmark side: better test sets, better agreement metrics, better judges of judges. We work from the side of real-world deployment, where a structural fact offers unused leverage: production evaluators are *versioned*.

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.

Between one version and the next, policies are amended, evidence pipelines are corrected, and the scope of a rule—which users, regions, or products it governs—is expanded or narrowed. After such an update, an evaluator's verdict on a case may flip, say from *deny* to *approve*. Three very different causes can produce the same flip: the accepted facts changed, the governing rule changed, or the rule newly gained (or lost) applicability to the case. Benchmarks that score only the final label erase this distinction entirely. So do free-text rationales, which are well documented to be plausible without being faithful to the computation that produced the prediction (Jacovi & Goldberg, 2020; Adebayo et al., 2018; Turpin et al., 2023).

The resulting failure mode is what we call *right verdict, wrong reason*: a model reproduces the correct post-update label while attributing the change to the wrong cause, or to superficial cues such as the mere presence of an edit in the prompt. Label accuracy is blind to this failure. For evaluators whose outputs carry operational consequences, it is precisely the failure that matters for auditability: if two evaluator versions disagree, which declared change was sufficient to produce the new verdict?

This paper makes attribution of verdict changes an exactly measurable quantity, and then uses it to show that high benchmark accuracy can coexist with essentially no attribution ability—a textbook signature of shortcut learning (Geirhos et al., 2020). Concretely:

An executable formalism (§2–3). We factor an evaluator's decision-relevant inputs into three typed components—evidence, rules, and scope. Toggling each component between its old and new version yields a 2^3 factorial of counterfactual configurations, the *counterfactual table*. The gold explanation of a verdict change is the family of all inclusion-minimal component swaps sufficient to reproduce the new verdict, a version-transition analogue of minimal sufficient reasons for classifier decisions (Darwiche & Hirth, 2020). We prove that exhaustive execution recovers this family exactly, bound its size via Sperner's theorem, and show that any exact black-box verification procedure must query all 2^k-2 internal configurations in the worst case. The formalism cleanly separates *prediction* (a model asserts an attribution) from *verification* (the counterfactuals are executed); Figure 1 summarizes the workflow.

A benchmark with exact ground truth (§4). REASONBENCH contains 19,520 cases built from 845 independently split source units: organizational policy rules derived from the public τ^2-Bench environments (Barres et al., 2025) and logical rule worlds generated with the RuleTaker grammar (Clark et al., 2020), each with executable old and new evaluator programs, all eight counterfactual outcomes, and the exact attribution family. Crucially, 7,200 paired controls test *consistency under semantics-preserving transformations*—section reordering, transition reversal, and irrelevant distractors—in the spirit of contrast sets and behavioral testing (Gardner et al., 2020; Ribeiro et al., 2020).

A pre-specified negative result and a diagnosis (§6). With the hypothesis, splits, and analysis frozen (via a cryptographic hash of the protocol and code) before results were inspected, we tested whether supervising the full counterfactual table improves attribution over predicting the attribution directly. It does not: the richer target is reliably *worse* on the held-out test (-1.42 pp, 95% CI $[-2.87, -0.30]$), a result that replicates on a smaller backbone. More importantly, both targets pass the standard held-out test at $\geq$97% while failing the consistency controls at $\approx$50%, and under a train-on-simple/test-on-composed split they preserve verdict accuracy (93.8%) while attribution accuracy on changed cases collapses to 7.2% (0% on the harder stratum). Final-answer accuracy transfers; the causal account does not.

The practical conclusion is architectural: learned evaluators can cheaply *propose* attributions for triage, but trustworthy attribution must be *verified* by executing counterfactuals against versioned, executable components—and explanation benchmarks should report invariance under semantics-preserving transformations alongside accuracy, which alone is nearly uninformative about whether reasons were learned.

2 PROBLEM SETUP

Table 1 collects the paper's terminology; each term is defined where first used.

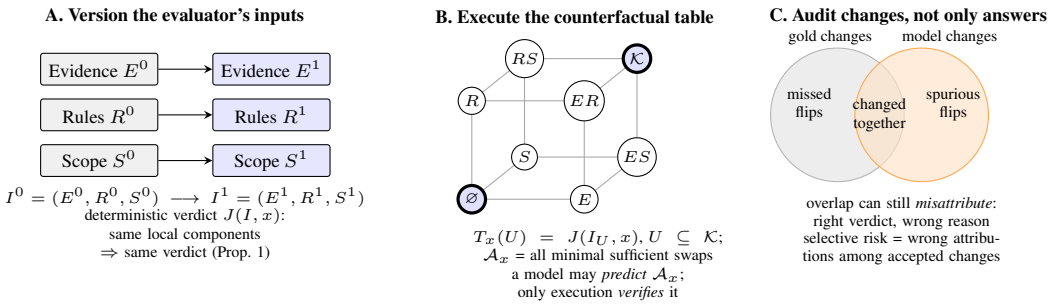


Figure 1: The counterfactual-attribution workflow. (A) The evaluator's inputs are factored into versioned, typed components. (B) Executing all 2^3 old/new hybrids yields the counterfactual table and the exact attribution family $\mathcal{A}_x$; a model-generated table or attribution is a *prediction* of this object, not its verification (§3.3). (C) Predictions are audited against executed ground truth, scoring endpoint errors and misattribution separately.

Table 1: Terminology and notation. $I_t = (E_t, R_t, S_t)$ is the evaluator version at time t and $J(I, x)$ its deterministic verdict on case x. Standard measures—false positives/negatives, selective risk, coverage, recall—are used with their usual meanings (§3.5).

Symbol	Term	Meaning
E, R, S	components	Evidence (accepted facts), rules (policy mapping facts to a decision), scope (which rules apply where).
$\mathcal{K}$	component set	$\mathcal{K} = \{E, R, S\}$; in general $k = \lvert\mathcal{K}\rvert$ components.
I_U	hybrid configuration	*New* components for $U \subseteq \mathcal{K}$, *old* for the rest.
$T_x(U) = J(I_U, x)$	counterfactual table	Verdicts of all 2^k hybrids: a full factorial ablation.
$\mathcal{A}_x$	attribution family	All inclusion-minimal U with $T_x(U) = T_x(\mathcal{K})$: the minimal sufficient causes of the change.
$\lvert\mathcal{A}_x\rvert$	multiplicity	Number of minimal attributions; $\min_{U \in \mathcal{A}_x} \lvert U\rvert$ is the attribution *complexity*.
A; M	gold / predicted changes	Cases whose gold verdict changes; cases whose predicted verdict changes.
$M \setminus A$; $A \setminus M$	spurious / missed flips	False positives / false negatives of change detection ("did the verdict change?").
$\hat{\mathcal{A}}_x \neq \mathcal{A}^\star_x$	misattribution	Wrong attribution on a gold change; possible even with both endpoint verdicts correct.
$\mathrm{IC}(\tau)$	consistency rate	Fraction of paired prompts satisfying the relation implied by semantics-preserving transformation τ.

2.1 Versioned evaluators and typed components

An evaluator version is a triple $I_t = (E_t, R_t, S_t)$: the *evidence* it accepts about a case (observations, retrieved records), the *rules* that map accepted evidence to a disposition (thresholds, prohibitions, exceptions), and the *scope* that records which rules apply to which actors, regions, products, or channels. A deterministic decision function returns a verdict

$$J(I_t, x) \;=\; D\big(E_t(x),\, R_t|_x,\, S_t|_x,\, x\big) \;\in\; \mathcal{Y}, \tag{1}$$

where $\mathcal{Y}$ is a finite set of verdicts and $R_t|_x$ and $S_t|_x$ restrict to the rules and scope relations capable of affecting case x. The tuple $\big(E_t(x), R_t|_x, S_t|_x\big)$ is the *local state* for x.

Scope is deliberately separated from the rules themselves: a rule says what follows *if* it applies; scope says *whether* it applies to this case; collapsing the two turns "out of scope" into a decision on the merits. ReasonBench's decision function makes the separation executable: with $\pi_r(e, x) \in \{\text{approve}, \text{deny}\}$ the merits decision under rule r and $s(r, e, x) \in \{0, 1\}$ the applicability indicator,

$$D(e, r, s, x) \;=\; \begin{cases} \text{refer}, & s(r, e, x) = 0, \\ \pi_r(e, x), & s(r, e, x) = 1. \end{cases} \tag{2}$$

Each component has a different version history and a different failure mode—exactly what attribution must distinguish. The factoring is an interface, not an ontology: a deployment may expose more or fewer typed components, and the construction generalizes to k components at cost exponential in k.

A first, intentionally elementary observation pins down the obligation this interface creates.

Proposition 1 (Local conservation)**.** *If D is deterministic and two versions agree on the local state for x, then $J(I_s, x) = J(I_t, x)$.*

Proof. The two calls to D receive identical arguments; determinism gives identical outputs. □

The contrapositive is the useful direction: a changed deterministic verdict requires at least one changed local component, so if these components are declared to be the locally sufficient inputs of the decision, an unexplained verdict change signals an incomplete interface or implementation. The converse fails—a component can change without affecting the verdict—and such consequence-free edits are important controls, because a model may treat the mere *presence* of an edit as a cue to flip its answer.

3 Counterfactual attribution of verdict changes

3.1 The counterfactual table and the attribution family

Fix an old version I_0 and a new version I_1. For $U \subseteq \mathcal{K}$, let I_U denote the hybrid that uses the new value for every component in U and the old value for the rest. The *counterfactual table* for case x is the full factorial

$$T_x(U) \;=\; J(I_U, x), \qquad U \subseteq \mathcal{K}. \tag{3}$$

The endpoints $T_x(\varnothing)$ and $T_x(\mathcal{K})$ record whether the verdict changed; the $2^k - 2$ internal cells record which single or joint swaps reproduce the new outcome, including interactions that one-at-a-time ablations miss. The table is well defined only when old and new components share a typed contract, so every hybrid is a valid input to the same decision function; ReasonBench guarantees this by construction (a production migration that changes field meanings would first need a compatibility map).

Definition 1 (Attribution family)**.** *When $T_x(\varnothing) \neq T_x(\mathcal{K})$, the* attribution family *is*

$$\mathcal{A}_x \;=\; \big\{\, U \subseteq \mathcal{K} \;:\; T_x(U) = T_x(\mathcal{K}) \;\wedge\; \forall\, V \subsetneq U,\; T_x(V) \neq T_x(\mathcal{K}) \,\big\}. \tag{4}$$

$\mathcal{A}_x$ is a *family*, not a single selected explanation: if swapping the evidence alone or the rules alone each reproduces the new verdict, then $\mathcal{A}_x = \{\{E\}, \{R\}\}$, and recording only one member suppresses an equally minimal account. The definition transports the minimal-sufficiency discipline of prime-implicant explanations for static classifier decisions (Darwiche & Hirth, 2020; Ji & Darwiche, 2023) to a new object: a version-to-version transition over typed, executable component swaps, with scope as a separately executable component. Unlike actual causality (Halpern & Pearl, 2005) or provenance (Green et al., 2007), no causal-model or lineage semantics is assumed: the family is a mechanical property of the declared interface, which is what makes it exactly computable.

3.2 Exact computation and the cost of verification

Proposition 2 (Existence and exact enumeration)**.** *If $T_x(\varnothing) \neq T_x(\mathcal{K})$ then $\mathcal{A}_x \neq \varnothing$, and executing all 2^k hybrids and retaining the sufficient subsets with no sufficient strict subset returns exactly $\mathcal{A}_x$.*

Proposition 3 (Sperner bound on multiplicity)**.** *With k components, $|\mathcal{A}_x| \leq \binom{k}{\lfloor k/2 \rfloor}$; for $k = 3$, at most three minimal attributions coexist.*

Proposition 4 (Worst-case verification lower bound)**.** *Suppose the two endpoint verdicts are known and differ. With no structural assumptions on the black-box hybrid evaluator, any exact deterministic algorithm that outputs the complete family $\mathcal{A}_x$ must query all $2^k - 2$ internal hybrids in the worst case.*

Proofs are short and given in Appendix A; Proposition 3 follows from the antichain structure of $\mathcal{A}_x$ and Sperner's theorem (Engel, 1997), and Proposition 4 from an adversary argument. Three components make exact verification cheap (eight executions); fine-grained decompositions do not, and reducing the cost requires stated structural assumptions such as monotonicity.

3.3 Prediction is not verification

A language model can output an attribution family directly, or all eight table cells from which one is derived. Both are *predictions*; neither certifies its own correctness—consistent with findings that LLM self-generated counterfactual explanations are unreliable (Mayne et al., 2025; Dehghanighobadi et al., 2025; Shi et al., 2025). Writing $\hat{\mathcal{A}}_x$ for a model output and $\mathcal{A}_x^\star$ for the family obtained by executing a trusted reference implementation, exact attribution accuracy is $\mathbf{1}[\hat{\mathcal{A}}_x = \mathcal{A}_x^\star]$, and a *verified* attribution is the executed table together with the deterministic minimality computation and cryptographic identifiers binding the case, the decision function, and both source versions. A predicted table can be internally consistent—its derived family cannot contradict its cells—while its cells are simply wrong. Proposition 4 explains why the full table is the right *audit* object; it says nothing about whether the table is the better *learning target*. Section 6 shows it is not.

3.4 Consistency under semantics-preserving transformations

Some prompt transformations provably preserve the underlying transition; others transform it in a known way. Let $z(x)$ be the structured output under audit (endpoints, attribution family, or table) for a case x in an evaluation set $\mathcal{X}$, and let τ be a transformation with known output relation H_τ. The *consistency rate* is

$$\mathrm{IC}(\tau) \;=\; \tfrac{1}{|\mathcal{X}|}\textstyle\sum_{x\in\mathcal{X}} \mathbf{1}\big[H_\tau\big(\hat{z}(x), \hat{z}(\tau x)\big)\big]. \tag{5}$$

For reordering typed input sections or appending an irrelevant record, H_τ requires identical endpoints and attribution; for reversing the transition ($I_0 \leftrightarrow I_1$), it requires the endpoints to swap. This is the logic of contrast sets and invariance tests (Gardner et al., 2020; Ribeiro et al., 2020) applied to attribution: it makes no claim about internal representations, only checks a necessary behavioral consequence of the task semantics. Failures are directly interpretable:

Proposition 5 (Consistency failures lower-bound error)**.** *If gold outputs satisfy H_τ, then the average of the paired error indicators $\frac{1}{2}\big(\mathbf{1}[\hat{z}(x) \neq z^\star(x)] + \mathbf{1}[\hat{z}(\tau x) \neq z^\star(\tau x)]\big)$ is at least $\big(1 - \mathrm{IC}(\tau)\big)/2$.*

A violated relation certifies that at least one member of the pair is wrong—no error model, no independence assumption. Unpaired accuracy cannot see this structure.

3.5 Error metrics

Let $\hat{y}_x^0, \hat{y}_x^1$ be the predicted old and new verdicts, $A = \{x : T_x^\star(\varnothing) \neq T_x^\star(\mathcal{K})\}$ the set of gold changes, and $M = \{x : \hat{y}_x^0 \neq \hat{y}_x^1\}$ the set of predicted changes. Treating "did the verdict change?" as binary change detection, *spurious flips* ($M \setminus A$) are its false positives and *missed flips* ($A \setminus M$) its false negatives; *misattribution* occurs on a gold change when $\hat{\mathcal{A}}_x \neq \mathcal{A}_x^\star$, and can survive perfect endpoint prediction—so we also report attribution error conditional on correct endpoints. For accepted predicted changes $C \subseteq M$ under a selective threshold, the *selective risk* of attribution (Geifman & El-Yaniv, 2019) is the fraction of C with a wrong attribution; it must be reported jointly with coverage and with recall of gold changes, since abstaining on every hard case yields zero risk while silently freezing the evaluator.

4 The ReasonBench benchmark

ReasonBench is built for intervention rather than retrospective explanation: every case ships with *executable* old and new evaluator states, all eight hybrid verdicts, and the exact attribution family, across two strata with complementary weaknesses.

Organizational policy stratum. 45 audited policy clauses from the public customer-service environments of τ^2-Bench (Barres et al., 2025) (airline, retail, telecom; 15 each) are translated into typed, executable rules. Cases carry structured evidence and a scope over roles, regions, products,

or channels; verdicts are {approve, deny, refer} via the gate in Eq. (2). The translation keeps recognizable policy structure while making every counterfactual executable; it does not claim to simulate a firm.

Logical rule stratum. 800 positive-rule worlds are independently sampled from the public RuleTaker generator and grammar (Clark et al., 2020) and solved by forward chaining (no original labels are reused). Facts serve as evidence, implication rules as rules, and an explicit applicability mask—added by REASONBENCH—as scope. This stratum supplies many independent units and controlled proof depth, at the price of templated language that invites shortcuts—which is why the consistency controls below are part of the test, not optional.

Cases, splits, and leakage control. Source units (clauses and worlds) are assigned to splits before any prompts are generated, so no prompt derived from a unit crosses splits. The main set has 19,520 cases: 11,712 train, 2,096 calibration, 2,400 test, and 3,312 shift-only. The held-out test split—frozen, together with the analysis protocol, before any model was run—draws on 102 independent units. 10,888 cases have a gold verdict change (complexity 1/2/3 for 7,366/2,693/829 of them; 3,470 have multiplicity > 1); all eight intervention masks occur. An independent re-execution audit found zero label errors, zero unit leakage, and zero prompt overlap across splits (Appendix B).

Paired consistency controls. Each test example generates three paired transformations (7,200 total): (i) *transition reversal*, whose endpoints must swap; (ii) *section reordering* of the semantically typed input blocks, whose endpoints and attribution must be unchanged; (iii) an *irrelevant distractor* record, likewise attribution-preserving. These are transformations of test examples, not additional independent samples.

Distribution shifts. Frozen shift splits vary the operator/clause mixture, the prompt template, and the logical proof depth. One originally planned split was discovered (before any learned results were inspected) not to withhold intervention combinations; it was relabeled *reserved structural* and is not used as evidence of compositional generalization. A corrected compositional split—train on 0/1-component changes, test on 2/3-component changes, with disjoint source units—was added and is labeled secondary throughout.

5 EXPERIMENTAL SETUP

Supervision targets. All methods receive identical descriptions of the old and new evidence, rules, and scope, and differ only in the serialized target: *direct verdict* (old and new verdicts), *direct attribution* (verdicts plus the complete attribution family), *full table* (all eight cell verdicts in fixed order; endpoints and attribution derived mechanically), and a one-seed *rationale* control that prefixes a deterministic one-sentence explanation. Table outputs are internally coherent by construction—a formatting property, not a correctness guarantee (§3.3).

Pre-specified hypothesis. Before any result was inspected, the protocol and analysis code were frozen under a SHA-256 content hash, together with the primary hypothesis: full-table supervision improves verdict and exact-attribution accuracy over direct attribution, via denser counterfactual supervision, with the gain (a) concentrated on multi-component and multi-attribution cases and (b) surviving equivalent presentations. These implications make the hypothesis falsifiable; the freeze is an internal cryptographic hash of the protocol and analysis code, not a third-party registry deposit.

Models and optimization. The primary backbone is a pinned Qwen3-1.7B release (Yang et al., 2025) with LoRA (Hu et al., 2022) (rank 16, α=32), five seeds, 500 steps, greedy decoding, and a strict full-string parser; malformed generations count as errors. A three-seed Qwen3-0.6B replication repeats the main comparison; character TF-IDF and majority baselines serve as leakage diagnostics; full hyperparameters and compute are in Appendix C.

Outcomes and inference. The two primary metrics are new-verdict accuracy and exact attribution-family accuracy (also reported on gold changes only, since stable cases have an empty family). Uncertainty is clustered by source unit; paired differences use 10,000 hierarchical bootstrap replicates over seeds and units, with Holm adjustment for the two primary test-set comparisons; selective thresholds are fit on calibration data only.

Table 2: Held-out test accuracy (mean over 5 seeds; seed SD in parentheses, percentage points). The rationale row is a one-seed descriptive control. Paired effects use hierarchical bootstrap over seeds and source units.

Supervision target	New-verdict acc.	Exact attribution acc.	Seeds
Direct verdict	99.10 (0.30)	—	5
Direct attribution	**99.27 (0.05)**	**98.41 (0.06)**	5
Full counterfactual table	97.42 (0.33)	96.99 (0.44)	5
Rationale + attribution	98.29	97.75	1
Table − direct (95% CI)	-1.85 pp $[-4.01, -0.23]$	-1.42 pp $[-2.87, -0.30]$	

6 RESULTS

6.1 SANITY CHECKS: LABEL IMBALANCE AND LEXICAL LEAKAGE

Majority prediction reaches 53.8% new-verdict / 44.2% exact-attribution accuracy on the held-out test; character TF-IDF reaches 90.0% / 61.7%; analytic chance for attribution is 22.6% (train-prior) and 7.7% (uniform over observed support). The split thus contains exploitable lexical signal, though far below the learned results. These diagnostics rule out a label-frequency account; they do not establish that reasons were learned—that requires the paired controls below.

6.2 SUPERVISING THE FULL COUNTERFACTUAL TABLE HURTS ACCURACY

Table 2 gives the frozen primary comparison. Direct attribution prediction is the strongest learned target; full-table supervision is significantly worse on both primary metrics (Holm-adjusted $p{=}.002$ and $p{=}.0036$), and the deficit does not shrink on the cases the hypothesis said it should help: direct prediction remains stronger at attribution complexity 1 (-2.04 pp) and 2 (-1.55 pp), the two methods tie at complexity 3, and there is no table advantage on multi-attribution cases (-0.67 pp $[-1.87, 0.29]$; full subgroup table in Appendix D). The three-seed 0.6B replication reproduces the ordering (attribution: -1.26 pp $[-3.08, -0.04]$; Appendix E). The pre-specified hypothesis is rejected in full: no overall gain, no gain where the mechanism predicted one, and (below) no survival under equivalent presentations.

The result separates two senses of "stronger." The executed table is stronger *evidence* for an attribution: it supports replay and minimality checks and is worst-case query-optimal for verification (Proposition 4). It is not the stronger *statistical target*: emitting eight cells is a larger joint prediction problem in which any cell error can corrupt the derived family, while direct prediction compresses the target to the object the metric scores. Output length does not explain the gap (table serializations average 6.7 generated tokens vs. 14.0), and format validity exceeds 99.9% for both. Structured output is not structured understanding.

6.3 RIGHT VERDICT, WRONG REASON, IN DISTRIBUTION

Even in distribution, correct endpoints do not entail correct attribution: on changed cases where *both* endpoint verdicts are predicted correctly, direct prediction still outputs a wrong attribution on 1.66% $[0.26, 3.61]$ of eligible seed-example observations, the table target on 1.85% $[0.47, 3.57]$. Change-detection errors are small (direct: 0.08% spurious, 0.31% missed flips; table: 0.52% / 1.07%), so in distribution the dissociation is a thin margin. Out of distribution, it becomes the dominant phenomenon.

6.4 CONSISTENCY CONTROLS EXPOSE SHORTCUT LEARNING

Ordinary distribution shifts (Table 3, top) degrade attribution accuracy gradually and never favor the table target. The paired controls (bottom) tell a qualitatively different story. Both models ignore an appended irrelevant record ($\geq$93%), but merely *reordering* the typed input sections—which changes nothing about the transition—yields a different attribution in roughly half of all pairs (54.8% / 49.2% consistency), and reversing the transition direction fails more than half the time. By Proposition 5,

Table 3: Top: exact attribution accuracy under distribution shift (per-seed case counts; paired effects with 95% CIs). Bottom: consistency rates on the paired semantics-preserving controls (2,400 pairs per seed). Near-ceiling test accuracy coexists with ≈50% consistency.

Shift split	n/seed	Direct	Table	Table − direct (95% CI)
Held-out test (i.i.d.)	2,400	98.41	96.99	-1.42 $[-2.87, -0.30]$
Mixed operator	288	81.81	78.61	-3.19 $[-8.61, 2.36]$
Prompt template	1,200	80.50	68.40	-12.10 $[-16.35, -7.50]$
Reserved structural	912	99.41	96.27	-3.14 $[-5.92, -0.86]$
Proof depth	912	96.73	82.19	-14.54 $[-19.34, -8.00]$

Paired consistency control	Direct	Table
Transition reversal (endpoints must swap)	47.22	42.97
Section reordering (attribution must be unchanged)	54.77	49.18
Irrelevant distractor (attribution must be unchanged)	98.63	93.43

Table 4: Compositional holdout (train on 0/1-component changes, test on 2/3-component changes; disjoint source units; 3 seeds; seed SD in parentheses). "Changes only" uses the conditional denominator of gold-changed cases.

Split / metric	n/seed	Direct	Table	Table − direct (95% CI)
Unseen comp.: new verdict	1,440	93.75 (0.91)	66.94 (0.50)	-26.81 $[-30.95, -22.70]$
Unseen comp.: attribution, all cases	1,440	36.62 (0.11)	37.22 (0.57)	$+0.60$ $[-0.37, 1.61]$
Unseen comp.: attribution, changes only	968	7.16 (0.06)	9.57 (0.70)	$+2.41$ $[0.35, 4.50]$
+ greater depth: new verdict	576	99.83 (0.30)	59.20 (3.25)	-40.63 $[-45.78, -36.57]$
+ greater depth: attribution, changes only	384	0.00 (0.00)	0.00 (0.00)	0.00

the reordering result alone implies an average error rate of at least 22.6% on these pairs for the direct target, against its 98.4% unpaired test accuracy. The models have learned template-specific correlations between section position, edit form, and output labels—the signature of shortcut learning (Geirhos et al., 2020), familiar from annotation artifacts in NLU (Gururangan et al., 2018; Poliak et al., 2018; McCoy et al., 2019) and order sensitivity in LLMs (Zheng et al., 2023). Held-out accuracy under a fixed prompt template measures predictability of that template, not a presentation-independent grasp of the task.

6.5 Verdicts generalize compositionally; attributions collapse

The compositional holdout (Table 4) is the paper's starkest dissociation. Direct prediction transfers its *verdicts* almost intact—93.75% on unseen multi-component changes, 99.83% on the deeper split—while its *attribution* on genuinely changed cases collapses to 7.16%, and to exactly 0% on the 768 logical-stratum changes and the deeper split (stable cases, whose gold family is empty, account for most of the all-case score). The table target's small changed-only advantage ($+2.41$ pp) is confined to 200 organizational changes from six clauses (stratum effect $+11.67$ pp $[3.76, 16.84]$) and coexists with a 26.8-point verdict loss: structured supervision reallocates errors between verdicts and attributions rather than enabling compositional attribution. The correct disposition survives when the account of *why* it changed does not—right verdict, wrong reason, now at scale.

6.6 Selective prediction trades coverage for attribution risk

Calibrated selective prediction (Geifman & El-Yaniv, 2019) concentrates attribution errors: at a nominal 80% operating point, direct prediction realizes 0.27% selective risk at 81.97% accepted-change coverage; near full coverage, risk rises to 2.26% (table: 3.48%; all operating points in Appendix F). Confidence identifies uncertain cases—useful for routing which cases get verified

first—but not the shortcut failure mode itself; selective risk must therefore always be read jointly with coverage and recall (§3.5).

7 Related work

Faithfulness of explanations. Plausible explanations need not be faithful (Jacovi & Goldberg, 2020): saliency maps can fail sensitivity sanity checks (Adebayo et al., 2018), and chain-of-thought can misstate the true drivers of a prediction (Turpin et al., 2023). ERASER operationalizes sufficiency for extractive rationales (DeYoung et al., 2020), and right-for-the-right-reasons training constrains explanations during learning (Ross et al., 2017). Our unit of analysis differs: the explained object is a *transition between evaluator versions*, candidate causes are typed executable component swaps rather than token subsets, and the complete minimal family is retained—contrastive in the sense of Miller (2019) (why the new verdict rather than the old) but selected by mechanical minimality, not salience.

Formal explanations and causality. Minimal sufficient reasons for classifier decisions correspond to prime implicants (Darwiche & Hirth, 2020), extended beyond Boolean features by Ji & Darwiche (2023); we inherit the minimal-sufficiency discipline and change the object. The counterfactual table is related in spirit to actual causality (Halpern & Pearl, 2005) and provenance (Green et al., 2007) but makes no causal-model or lineage claim: it is a replayable property of a declared interface.

Shortcut learning and behavioral testing. High benchmark accuracy from spurious cues is extensively documented (Geirhos et al., 2020; Gururangan et al., 2018; Poliak et al., 2018; McCoy et al., 2019); contrast sets and CheckList advocate paired, semantics-controlled evaluation (Gardner et al., 2020; Ribeiro et al., 2020). We contribute a setting where the paired relation is *provably* entailed by executable semantics, so violations lower-bound error (Proposition 5) with no judgment call about label preservation.

LLM evaluators and model updates. LLM-as-judge pipelines exhibit position and formatting biases (Zheng et al., 2023), and self-generated counterfactual explanations are unreliable (Mayne et al., 2025; Dehghanighobadi et al., 2025; Shi et al., 2025). Model editing separates edit success from locality (Mitchell et al., 2022; Ma et al., 2024); spurious/missed flips give the analogous change-detection decomposition, and attribution adds what editing metrics omit: *which* declared change an update is recorded under. Explanation stability under shift (Meyer et al., 2023), selective prediction (Geifman & El-Yaniv, 2019), and audit frameworks (Kroll et al., 2017; Raji et al., 2020) are complementary; an executed attribution is one narrow machine-checkable artifact within that program.

8 Limitations

ReasonBench's evaluators are executable programs by construction—what makes its counterfactuals exact is also an external-validity limit: real organizations contain tacit practice, conflicting records, and contested scope, and the declared interface must be audited rather than assumed (Selbst et al., 2019). The theory assumes determinism and typed hybrid compatibility, and exact verification is exponential in the number of components. The organizational test stratum rests on six independent clauses; experiments use one open-weight model family at two scales; and no human study tests the interface (extended discussion in Appendix G).

9 Conclusion

When an LLM evaluator's verdict changes across versions, the change has an exactly computable account: the minimal swaps of evidence, rules, or scope that reproduce it. Measuring that account reveals what label accuracy hides: near-perfect attribution under a fixed prompt template, lost under reordering and almost entirely under compositional shift while verdicts remain correct—with the audit-optimal full-table target making prediction worse. The standard this supports is two-part: treat learned outputs as *predictions* whose invariance, coverage, and selective risk are measured against executed counterfactuals, and reserve *verification* for the counterfactuals themselves. Accuracy says the evaluator got the right verdict; only executed attribution says it changed for the right reason.

REPRODUCIBILITY STATEMENT

All theoretical claims are stated with assumptions in §2–3 and proved in Appendix A. Benchmark construction, split assignment at the source-unit level, audit procedures, and control generation are described in §4 and Appendix B; hyperparameters, seeds, decoding, parsing, and compute in §5 and Appendix C; and the statistical protocol (clustering, hierarchical bootstrap, Holm adjustment, freeze procedure) in §5. The benchmark generators build on public assets (Barres et al., 2025; Clark et al., 2020); data, code, and the frozen protocol hash will be released upon publication.

ETHICS STATEMENT

This work aims to improve the auditability of automated evaluators whose outputs carry operational consequences. An executed attribution shows that a declared change was sufficient inside an executable evaluator; it does not show that a rule is just, that evidence is true, or that a scope assignment is legitimate, and it should complement rather than replace human appeal processes. The benchmark contains no personal data; it is derived from public synthetic environments.

AI USE STATEMENT

Generative AI tools were used to edit this manuscript for readability and typographical consistency, to check compliance with the ICLR 2027 formatting requirements, and to verify the consistency and accuracy of the bibliographic references. We did not use generative AI tools for any task requiring mandatory disclosure under the ICLR 2027 AI policy, including generating data, developing the theoretical framework, formulating or proving mathematical claims, designing the methodology or experiments, implementing methods, or interpreting results. We have reviewed all AI-assisted output and take responsibility for the final content of this work.

REFERENCES

Julius Adebayo, Justin Gilmer, Michael Muelly, Ian Goodfellow, Moritz Hardt, and Been Kim. Sanity checks for saliency maps. In *Advances in Neural Information Processing Systems*, volume 31, 2018.

Victor Barres, Honghua Dong, Soham Ray, Xujie Si, and Karthik Narasimhan. τ^2-Bench: Evaluating conversational agents in a dual-control environment. *arXiv preprint arXiv:2506.07982*, 2025.

Peter Clark, Oyvind Tafjord, and Kyle Richardson. Transformers as soft reasoners over language. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 3882–3890, 2020.

Adnan Darwiche and Auguste Hirth. On the reasons behind decisions. In *Proceedings of the 24th European Conference on Artificial Intelligence (ECAI)*, pp. 712–720, 2020.

Zahra Dehghanighobadi, Asja Fischer, and Muhammad Bilal Zafar. Can LLMs explain themselves counterfactually? In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 7787–7815, 2025.

Jay DeYoung, Sarthak Jain, Nazneen Fatema Rajani, Eric Lehman, Caiming Xiong, Richard Socher, and Byron C. Wallace. ERASER: A benchmark to evaluate rationalized NLP models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 4443–4458, 2020.

Konrad Engel. *Sperner Theory*. Cambridge University Press, 1997.

Matt Gardner, Yoav Artzi, Victoria Basmov, Jonathan Berant, Ben Bogin, Sihao Chen, Pradeep Dasigi, Dheeru Dua, Yanai Elazar, Ananth Gottumukkala, et al. Evaluating models' local decision boundaries via contrast sets. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pp. 1307–1323, 2020.

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 (ICML)*, pp. 2151–2159, 2019.

Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A. Wichmann. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673, 2020.

Todd J. Green, Grigoris Karvounarakis, and Val Tannen. Provenance semirings. In *Proceedings of the 26th ACM Symposium on Principles of Database Systems (PODS)*, pp. 31–40, 2007.

Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel R. Bowman, and Noah A. Smith. Annotation artifacts in natural language inference data. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL), Volume 2 (Short Papers)*, pp. 107–112, 2018.

Joseph Y. Halpern and Judea Pearl. Causes and explanations: A structural-model approach. part I: Causes. *British Journal for the Philosophy of Science*, 56(4):843–887, 2005.

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

Alon Jacovi and Yoav Goldberg. Towards faithfully interpretable NLP systems: How should we define and evaluate faithfulness? In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 4198–4205, 2020.

Chunxi Ji and Adnan Darwiche. A new class of explanations for classifiers with non-binary features. In *Logics in Artificial Intelligence (JELIA)*, volume 14281 of *Lecture Notes in Computer Science*, pp. 106–122, 2023.

Joshua A. Kroll, Joanna Huey, Solon Barocas, Edward W. Felten, Joel R. Reidenberg, David G. Robinson, and Harlan Yu. Accountable algorithms. *University of Pennsylvania Law Review*, 165 (3):633–705, 2017.

Jun-Yu Ma, Zhen-Hua Ling, Ningyu Zhang, and Jia-Chen Gu. Neighboring perturbations of knowledge editing on large language models. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, pp. 33839–33854, 2024.

Harry Mayne, Ryan Othniel Kearns, Yushi Yang, Andrew M. Bean, Eoin D. Delaney, Chris Russell, and Adam Mahdi. LLMs don't know their own decision boundaries: The unreliability of self-generated counterfactual explanations. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 24161–24186, 2025.

R. Thomas McCoy, Ellie Pavlick, and Tal Linzen. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 3428–3448, 2019.

Anna P. Meyer, Dan Ley, Suraj Srinivas, and Himabindu Lakkaraju. On minimizing the impact of dataset shifts on actionable explanations. In *Proceedings of the 39th Conference on Uncertainty in Artificial Intelligence (UAI)*, pp. 1434–1444, 2023.

Tim Miller. Explanation in artificial intelligence: Insights from the social sciences. *Artificial Intelligence*, 267:1–38, 2019.

Eric Mitchell, Charles Lin, Antoine Bosselut, Christopher D. Manning, and Chelsea Finn. Memory-based model editing at scale. In *Proceedings of the 39th International Conference on Machine Learning (ICML)*, pp. 15817–15831, 2022.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*, volume 35, pp. 27730–27744, 2022.

Adam Poliak, Jason Naradowsky, Aparajita Haldar, Rachel Rudinger, and Benjamin Van Durme. Hypothesis only baselines in natural language inference. In *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics (*SEM)*, pp. 180–191, 2018.

Inioluwa Deborah Raji, Andrew Smart, Rebecca N. White, Margaret Mitchell, Timnit Gebru, Ben Hutchinson, Jamila Smith-Loud, Daniel Theron, and Parker Barnes. Closing the AI accountability gap: Defining an end-to-end framework for internal algorithmic auditing. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency (FAccT)*, pp. 33–44, 2020.

Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. Beyond accuracy: Behavioral testing of NLP models with CheckList. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 4902–4912, 2020.

Andrew Slavin Ross, Michael C. Hughes, and Finale Doshi-Velez. Right for the right reasons: Training differentiable models by constraining their explanations. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 2662–2670, 2017.

Andrew D. Selbst, Danah Boyd, Sorelle A. Friedler, Suresh Venkatasubramanian, and Janet Vertesi. Fairness and abstraction in sociotechnical systems. In *Proceedings of the Conference on Fairness, Accountability, and Transparency (FAT*)*, pp. 59–68, 2019.

Haochen Shi, Shaobo Li, Guoqing Chao, Xiaoliang Shi, Wentao Chen, and Zhenzhou Ji. Do LLMs behave as claimed? investigating how LLMs follow their own claims using counterfactual questions. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 29043–29056, 2025.

Miles Turpin, Julian Michael, Ethan Perez, and Samuel R. Bowman. Language models don't always say what they think: Unfaithful explanations in chain-of-thought prompting. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging LLM-as-a-judge with MT-Bench and Chatbot Arena. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

A PROOFS

Proposition 2. The full set $\mathcal{K}$ is sufficient by definition of the endpoints, so the finite family of sufficient subsets is nonempty and contains a minimal member; hence $\mathcal{A}_x \neq \varnothing$. The enumeration retains a subset iff it is sufficient and has no sufficient strict subset, which is precisely membership in $\mathcal{A}_x$; every subset is queried, so no member is missed. □

Proposition 3. No two members $U, V \in \mathcal{A}_x$ satisfy $U \subsetneq V$, since U would be a sufficient strict subset of V. The family is therefore an antichain in the Boolean lattice over $\mathcal{K}$, and Sperner's theorem (Engel, 1997) bounds its size by $\binom{k}{\lfloor k/2 \rfloor}$. □

Proposition 4. Let the endpoint verdicts be $y_0 \neq y_1$. An adversary answers y_0 to every internal query. Suppose the algorithm halts with some internal subset U unqueried, and consider two completions consistent with the transcript: (i) every internal cell equals y_0, so the unique minimal sufficient subset is $\mathcal{K}$; (ii) among internal cells, exactly U is assigned y_1, so U is the unique minimal member. The two completions require different outputs, so the algorithm errs on one of them. Hence all $2^k - 2$ internal cells must be queried in the worst case. □

Proposition 5. If the predicted pair violates H_τ while the gold pair satisfies it, the two predictions cannot both equal their gold values, so $\mathbf{1}[\hat{z}(x) \neq z^\star(x)] + \mathbf{1}[\hat{z}(\tau x) \neq z^\star(\tau x)] \geq \mathbf{1}[\neg H_\tau(\hat{z}(x), \hat{z}(\tau x))]$ pointwise. Averaging over $\mathcal{X}$ and dividing by two gives the bound. □

B BENCHMARK DETAILS

Composition. Table 5 gives the split composition. Source clauses and logical worlds are assigned to splits before prompts are generated; individual prompts are never treated as independent observations, and all clustered inference in the paper clusters at the source-unit level.

Table 5: REASONBENCH composition. Units are independent source clauses or logical worlds; splits are assigned at the unit level before prompts are generated.

Stratum	Units	Main cases	Train	Calibration	Test	Shift-only
Organizational policy	45	4,320	2,592	576	576	576
Logical rules	800	15,200	9,120	1,520	1,824	2,736
Total	845	19,520	11,712	2,096	2,400	3,312

Among main cases, 10,888 have a gold endpoint change and 8,632 are stable. Minimum attribution size is 1 for 7,366 changed cases, 2 for 2,693, and 3 for 829; 3,470 changed cases have $|\mathcal{A}_x| > 1$. All eight intervention masks over $\{E, R, S\}$ occur. The 45-clause inventory and its clause-level split assignment were reused from an earlier public-data construction and predate all model results; all states, scopes, tables, attributions, prompts, and labels were constructed anew for this study.

Audit. Before the primary freeze, an independent executor recomputed every counterfactual table, attribution family, target string, and intervention label, finding zero label errors, zero source-unit leakage, zero exact prompt overlap, and zero normalized prompt overlap across splits; all 7,200 control transformations were recomputed without error. All sixteen primary runs (five seeds $\times$ three learned targets, plus the one-seed rationale control) completed; a separate audit reconstructed aggregates from saved prediction files and rejected missing, duplicated, malformed, or non-canonical artifacts. Two post-run aggregation bugs (a conditional denominator incremented only after successful parsing, and a heterogeneous CSV read using first-row fields) were logged, fixed, covered by regression tests, and followed by complete re-aggregation from raw predictions; malformed generations count as errors throughout. The final test suite contains 25 passing tests.

Shift splits. *Mixed operator* mixes operator and clause changes; its genuinely held-out exception-rule subset contains 192 cases from only two organizational clauses (direct 80.52%, table 79.58%; $-0.94\,\mathrm{pp}\ [-6.88, 3.75]$, descriptive because two units cannot support a rule-family claim). *Prompt template* uses an alternate surface form of the same content. *Proof depth* increases logical proof depth. *Reserved structural* is the originally planned compositional split that was found (pre-result) not to withhold intervention combinations; it is retained as a shift split only. The corrected compositional holdout (§6.5) trains on intervention cardinality ≤ 1 and tests on cardinality ≥ 2 with no source-unit overlap; its changed-only and per-stratum intervals were added after the combined metrics were inspected and are labeled descriptive extensions.

C TRAINING AND COMPUTE DETAILS

Table 6: Primary configuration.

Setting	Value
Backbone	Qwen3-1.7B, pinned revision
Adaptation	LoRA rank 16, α 32, dropout 0.05 on attention and MLP projections
Unique examples per primary run	5,184 after equalizing the two strata
Optimization	500 steps; batch 2; grad. accumulation 16; cosine schedule; 3% warmup
LR / weight decay	10^{-4} / 10^{-2}
Context / precision	1,024 tokens; bfloat16
Seeds	11, 23, 37, 53, 71 (replication: 3 seeds, Qwen3-0.6B)
Decoding	Greedy; strict full-string parser; malformed outputs count as errors

The full unbalanced training split (11,712 examples) is used by the character TF-IDF and majority diagnostics, which are leakage/surface-predictability probes rather than compute-matched competitors. A 100-step format-only calibration run before the freeze produced 256 valid outputs per serialization; no held-out semantic outcome was opened during calibration.

Table 7: Compute and generation cost (primary rows average five seeds; rationale has one seed). Wall time is descriptive given hardware concurrency.

Target	Trainable params	Mean train time	Peak reserved mem.	Gen. tokens/example
Direct verdict	17.43M	53.8 min	12.05 GiB	7.00
Direct attribution	17.43M	53.6 min	12.57 GiB	14.02
Full table	17.43M	53.2 min	12.60 GiB	6.70
Rationale + attribution	17.43M	52.7 min	11.43 GiB	22.22

D STRUCTURAL SUBGROUPS ON THE HELD-OUT TEST

Table 8: Exact attribution accuracy by structure on the held-out test. Rows differ in both structure and source composition and should not be read as isolated causal effects. Intervals shown for the pre-specified paired subgroup comparisons.

Test subgroup	n/seed	Direct	Table	Table − direct
Min. attribution size 1	904	98.21	96.17	-2.04 pp $[-4.79, -0.27]$
Min. attribution size 2	335	95.40	93.85	-1.55 pp $[-4.06, 0.24]$
Min. attribution size 3	101	95.45	95.45	0.00 pp $[-0.99, 1.19]$
Single attribution (incl. stable)	1,980	98.38	96.81	—
Multiple minimal attributions	420	98.52	97.86	-0.67 pp $[-1.87, 0.29]$

On the 1,340 gold changes per seed on the held-out test, direct attribution accuracy is 97.30% and its new-verdict accuracy is 1.54 points higher $[0.24, 3.42]$; for the table target the gap is 0.90 points $[-0.62, 2.45]$. The conditional misattribution rates given both endpoints correct are reported in §6 (1.66% and 1.85%, over 6,609 and 6,337 eligible seed-example observations respectively).

E SMALLER-BACKBONE REPLICATION

Table 9: Qwen3-0.6B replication (3 seeds; seed SD in parentheses; paired source-unit bootstrap CIs). "Changes only" rows use conditional denominators.

Split / metric	n/seed	Direct	Table	Table − direct (95% CI)
Held-out test: new verdict	2,400	99.15 (0.25)	96.76 (0.95)	-2.39 $[-4.75, -0.49]$
Held-out test: attribution, all cases	2,400	98.06 (0.06)	96.79 (1.01)	-1.26 $[-3.08, -0.04]$
Held-out test: attribution, changes only	1,340	96.79 (0.22)	95.12 (1.24)	-1.67 $[-4.36, 0.15]$
Template shift: new verdict	1,200	95.44 (1.18)	92.97 (3.00)	-2.47 $[-8.02, 2.66]$
Template shift: attribution, all	1,200	78.64 (2.68)	70.97 (1.64)	-7.67 $[-11.09, -3.71]$
Template shift: attribution, changes	662	63.24 (3.98)	50.15 (5.39)	-13.09 $[-18.80, -4.96]$
Greater depth: new verdict	912	99.05 (1.27)	99.63 (0.54)	$+0.58$ $[-0.77, 2.63]$
Greater depth: attribution, all	912	74.78 (3.24)	71.97 (0.64)	-2.81 $[-6.58, 0.29]$
Greater depth: attribution, changes	528	56.50 (5.54)	52.15 (0.67)	-4.36 $[-10.80, 0.95]$

The held-out test ordering replicates; the template shift strengthens the direct-target advantage; the greater-depth interval crosses zero while both methods lose substantial attribution accuracy relative to the 1.7B runs, so capacity matters for deeper rule worlds but the tested reduction does not reveal a table advantage. Format validity is 100% for both methods on the test split, 99.97% on the template

shift, and 99.05% (direct) versus 100% (table) at greater depth, so formatting does not account for the gaps.

F SELECTIVE PREDICTION OPERATING POINTS

Table 10: Held-out test selective-prediction operating points (thresholds fit on calibration data; seed means). Selective risk must be read jointly with realized coverage and recall of gold changes.

Method, coverage target	Realized accepted-change coverage	Selective risk	Recall of gold changes
Direct, 80%	81.97%	0.27%	81.61%
Table, 80%	85.54%	0.16%	84.30%
Direct, 100%	99.94%	2.26%	98.60%
Table, 100%	99.98%	3.48%	94.58%

Thresholds are also fit at 20%, 40%, and 60% targets with the same qualitative pattern: selective risk falls monotonically as coverage is withheld, i.e., confidence concentrates observed attribution errors rather than recognizing the causal failure mode.

G EXTENDED LIMITATIONS

Programmatic evaluators. REASONBENCH provides exact counterfactuals because its evaluators are executable programs. Real deployments contain tacit practices, discretionary exceptions, and contested scope assignments that no three-component interface captures automatically; if policy is hidden in model weights, if scope changes the meaning of evidence, or if retrieval and decision-making are entangled, the declared components may omit a relevant cause, and an unexplained verdict change is then evidence of an incomplete interface (Proposition 1) rather than a certified attribution.

Determinism and hybrid compatibility. The formal results assume a deterministic decision function and a shared typed contract between old and new components. Stochastic evaluators would require distributional attributions or a coupling that defines when a swap "reproduces" an outcome; interface migrations that rename fields or change units require an explicit compatibility map, or the transition should be recorded as unverifiable under the old interface.

Exponential verification. Exact unrestricted verification executes 2^k hybrids. Three components are practical; fine-grained decompositions are not. Monotonicity, sparsity, or verified program structure could reduce query cost, but such assumptions must be stated and tested.

Benchmark composition and model coverage. The organizational test stratum contains six independent clauses, so its estimates are reported with intervals but not treated as representative of organizational policy in general; the logical stratum is larger but templated. Experiments cover one open-weight family at 1.7B and 0.6B parameters with LoRA; larger models, constrained decoding, permutation augmentation, and architectures that execute structured programs remain open comparisons that the frozen consistency and compositional controls are designed to score.

No human-validity or systems study. We do not test whether practitioners find the component interface complete, whether auditors prefer the full minimal family, or whether attribution improves appeal outcomes; nor do we measure the latency or labor savings of a deployed prediction-plus-verification pipeline. An executed attribution shows a declared change was sufficient inside an executable evaluator—it does not establish that the rule is just, the evidence true, or the scope legitimate.