Refutation-Ledger Values: Auditable Warrant Accounting for Contestable Derived Claims

DeepForge Research

September 18, 2026 · refutation-ledgers

Preprint · Artifact · Not peer reviewed

Abstract

Derived content — hints, explanations, difficulty labels, prerequisite edges — is usually consumed with positive evidence only: a passing check is a boolean with no record of what was attacked and what failed. Refutation-Ledger Values gives a runtime value a warrant that is an append-only, tamper-evident ledger of falsification attempts, a grade that is a total, recomputable function of that ledger (declared dependence-class count, a cap of 3, and the grades of cited values), and demotion that is exact with respect to the cite graph. The Derived-Claim Arena that accompanies it is a constructed instance, not an experiment: ground truth is generated from a declared blind-spot table, and the shipped relation B7 is defined by that same table, so in the correlated regime X B7 coincides with the analysis-only oracle B8 and the pair-win 1.000 against B5's 0.000, the AUC 1.000 against 0.000, and the audit precision AP@12 1.000 against 0.000 are algebraic consequences of the construction rather than measurements of grader quality. What the arena shows is that a dependence-class grade is stable under seed churn and replay — seed-only churn moves the mean grade by 0.000 and flips 0.000 of pairs, which are the figures carrying information — and that the syntactic tuple variant (B6) is killed as pre-declared. The finding it reproduces inside the generator is the published one that count-based fusion collapses when sources are correlated (Dong, Berti-Equille & Srivastava 2009). No individual mechanism is claimed as new — TMS retraction, evidence fusion, audited ledgers, and attack-family scoring are all known — and the contribution is the value-level ledger-to-grade-to-demotion contract with its falsification record, never a truth verdict.

Keywords: warrant, falsification ledger, recomputed grade, declared dependence classes, exact demotion, audit, contestability

1. Introduction & motivation

Derived content is consumed everywhere and carries no negative evidence. A legal research tool can fabricate or misattribute source material on between 17 and 33 percent of pre-registered queries (the Stanford RegLab study, https://arxiv.org/abs/2405.20362). A widely deployed sepsis model reached hundreds of hospitals on vendor-reported performance, and an external evaluation measured an AUC of 0.63 where 0.76 to 0.83 had been reported, with 67 percent of sepsis cases missed and alerts on 18 percent of admissions (https://jamanetwork.com/journals/jamainternalmedicine/fullarticle/2781307). A near-1,000-student field experiment found that access to a generative tutor improved practice performance by 48 percent but left students 17 percent worse on later exams taken without it (https://www.pnas.org/doi/abs/10.1073/pnas.2422633122). In each case the consumed artifact carried a positive signal — a citation, a score, an explanation — and none carried a record of what had been attacked and had failed.

The gap is not that evidence is absent; it is that the negative half is never recorded. A passing test is

a boolean with no ledger of what was tried and failed; an explanation is displayable text with no attempt record; a derived label has a producer and no defeaters. Assurance research argues the point directly: positive arguments are confirmation-bias prone, and defeaters should be recorded, investigated, and retained (Assurance 2.0; Bloomfield et al.). Existing machinery does not transfer: TMS and ATMS maintain consistency rather than graded warrant; provenance semirings are positive-only; survival scores over scientific claims are offline reports, not runtime values with cites. This paper contributes the smallest thing that closes part of that gap: a value-level contract in which a warrant is a ledger, the grade is a recomputable function of the ledger, and demotion is exact through cites.

- A value-level contract only: a value's warrant is an append-only ledger of attempts; the grade is a total, recomputable function of that ledger with a declared dependence-class term, a cap of 3, and the minimum over cited grades; demotion is exact with respect to the cite graph. Every individual mechanism is known.
- A frozen Derived-Claim Arena that gives the declared dependence relation its best shot: 48 generated claims, 8 defect classes, 16 refuters in 4 families with fixed blind spots (regime X adds two families sharing f2's blind spot), 12 matched conflict pairs per regime, and 200 pre-committed seeds.
- A falsification record reported in place: the syntactic tuple variant B6 is killed by the F1 criterion; the shipped relation B7 passes P1-P5 at the 5th percentile; the cheap baselines, the negative arms, and the oracle ceiling are printed beside the headline rather than removed.
- A read-only lab at /warrant that never affects a score, path, rank, or teacher view, with the exact grade definition, the saturation note, and the declared relation visible next to every value.
- An explicit limits list: fabricated-but-consistent ledgers, declared blind spots, absorbing kill, synthetic corpus, and the admission trust root, stated rather than hidden.

> *The claim ceiling — The defensible claim is that a runtime value's warrant is an append-only, tamper-evident ledger of falsification attempts whose grade is a total, recomputable function of that ledger (declared dependence-class term, K-cap, and cited-value grades included) and whose demotion is exact with respect to the cite graph — a value-level ledger->grade->demotion contract that we did not locate in any published work or located open-source system, and which we therefore claim as an absence in the literature we surveyed rather than as a priority; the individual mechanisms (TMS retraction, evidence fusion, audited ledgers, attack-family scoring) are all known. Nothing stronger is claimed anywhere in this paper or in the lab: audit is arithmetic consistency against the ledger plus anchored tamper-evidence at the append point, never truth, never authenticity of a well-formed fabrication, and never a statement about any person. The arena contributes no evidence for the contract; it is an implementation consistency check on a generator whose ground truth and shipped grader are two readings of one table (Section 6).*

2. Related work, stated honestly

The closest work is the record layer, not the value layer. `falsification-ledger` implements a pre-registered falsification contract over an append-only, hash-chained ledger with content-addressed reports and a chain recompute; FalsiFlyer binds kernel, dataset, and decision-rule hashes to verdicts in a signed, hash-chained ledger; `falsifyr` scores the survival of scientific claims from attacks keyed by family and seed. Each makes a claim's standing depend on

recorded attempts. None treats a runtime value as the unit, none feeds a cite graph, and none demotes dependents through cites when an attempt refutes. An earlier version of this table also listed a fourth record-layer project as the closest single match for the grade clause; that repository could not be located when the reference list was re-checked, the row has been withdrawn rather than left standing, and no part of the claim above depends on it. The negative result that survives is the one stated in Section 1: no ledger, ledger-family, or truth-discovery system we located composes ledger-as-warrant, a dependence-class grade, and cite demotion at the value level.

TMS and ATMS give exact dependency-directed retraction for beliefs and assumption sets, but a belief is alive or dead, not graded by a ledger; we borrow the retraction obligation and restrict our claim to it (the demotion sanity check exercises TMS-style retraction and is reported as a check, not as the novelty). Provenance semirings are positive-only: a refuted attempt has no semiring value, so demotion is not derivable from the algebra. Defeasible argumentation (Verheij's accrual; weighted gradual semantics) makes the quality and count of independent reasons a first-class grading input, but has no executable ledger, no syntactic refuter identity, and no exact cite demotion. Truth discovery detects copied sources and penalizes dependent votes, but produces claim confidence rather than runtime warrants. Subjective logic and Dempster-Shafer require source separation for fusion and document how hard that assumption is; the truth-discovery literature makes the same point in the grading setting, that counts over dependent sources are not counts over independent ones, and that is the standing counter-pressure that regime X operationalizes. Knight and Leveson is cited above as the multiversion-programming origin of the independence assumption, but that paper was not read for this table, so no claim here rests on its content. Assurance 2.0 records defeaters and warns that counts are gameable; EviBound promotes claims only with machine-checkable evidence; the evidence-ledger line audits traceability. T1 states the exact delta row by row, and nothing in that table supports a claim that ledgers, retraction, or attack-family scoring are new here.

Work	URL	Implements	Clause matched	Not implemented
falsification-ledger (OSS, PyPI 0.1.1, 2026)	https://github.com/foolproof-labs/falsification-ledger	pre-registered falsification contract; append-only hash-chained ledger; content-addressed reports; chain recompute on demand	RLV(i); RLV(iii) tamper-audit	runtime values; declared dependence-class count; cite demotion; grade semantics
falsifyr (CRAN/GitHub 1.0.0, 2025-26)	https://github.com/msaule/falsifyr/	attack leaderboard; attack families; deterministic per seed; smallest-kill; 0-100 survival score	RLV(i); partly RLV(ii)	runtime values; declared dependence-class grade; cites and demotion; audit by recompute (the score is a heuristic)
falsification-ledger hit-rate report	https://pypi.org/project/falsification-ledger/	hit-rate over adjudicated claims with per-source-type buckets; verdict_ready gate	RLV(iii) audit of a derived statistic	per-value grades; declared dependence accounting; demotion

Work	URL	Implements	Clause matched	Not implemented
FalsiFlyer AUDIT_LEDGER_SPEC (OSS, retrieved 2026-09-18)	https://github.com/subvurs/FalsiFlyer/blob/main/docs/AUDIT_LEDGER_SPEC.md	hash-chained and signed ledger binding kernel, dataset, and decision-rule hashes to verdicts; chain check; eight adversarial baselines; truncation caveat	RLV(i); RLV(iii)	runtime values; grade function; dependence classes; cite demotion
Doyle 1979 TMS / de Kleer 1986 ATMS	https://dspace.mit.edu/handle/1721.1/5733	justifications as warrant; exact dependency-directed retraction and label recomputation; nogoods	RLV(iv)	falsification ledger; dependence accounting; graded warrant
provenance semirings (Green et al. 2007)	https://dl.acm.org/doi/10.1145/1265530.1265535	positive provenance polynomials for how a result derives	RLV(i) flavor only	negative evidence; refutation; demotion
Verheij accrual of arguments (1995/1999)	https://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.41.4458	independent arguments reinforce; more reasons imply more cogency; compound defeat	RLV(ii) concept	combinatorial dependence set; machine audit; demotion of values
weighted gradual semantics (Amgoud et al. IJCAI 2017)	https://doi.org/10.24963/ijcai.2017/9	acceptability degree from attacker strength; reinforcement principles	RLV(ii)/(iii) concept	ledger warrant; syntactic refuter identity; exact demotion
truth discovery with dependent sources (SIGKDD Explor. 2016)	https://doi.org/10.1145/2897350.2897352	detects copied sources; penalizes dependent votes; claim confidence against source reliability	RLV(ii) concept	runtime warrants; declared dependence-class count; demotion; audit recompute
subjective logic / EBSL (Josang 2001)	https://doi.org/10.1016/s0218-4885(01)00083-1	grades as functions of positive and negative evidence; fusion assumes source separation; discounting	RLV(ii)/(iii) concept	append-only ledger; syntactic ids; cite demotion; audit

Work	URL	Implements	Clause matched	Not implemented
Knight & Leveson (IEEE TSE 1986)	https://www.csc.kth.se/utbildning/kth/kurser/DA2210/vettig13/Seminarier/KnightLeveson.pdf	cited as the multiversion-programming origin of the independence assumption; the record is live but was not read for this table, so the specific attribution below rests on the truth-discovery literature rather than on this paper	RLV(ii) counter-pressure	no system; everything else
Assurance 2.0 defeaters (2024)	https://www.csl.sri.com/~rushby/papers/defeaters24.pdf	defeaters recorded, investigated, retained; confidence from doubts; warns that counts are gameable	RLV(i)/(ii) caution	runtime values; exact grade function; cite demotion
EviBound (arXiv 2511.05524, 2025)	https://arxiv.org/abs/2511.05524	claims promoted only with machine-checkable evidence; run id, artifacts, status; refusal and blocking	RLV(i)/(iii); the promotion-with-evidence thesis is prior art and is not claimed here	ledger of falsification attempts; dependence; grade; demotion
SV-COMP witnesses 2.0 (SPIN 2024)	https://www.sosy-lab.org/research/pub/2024-SPIN.Software_Verification_Witnesses_2.0.pdf	violation and correctness witnesses as exchangeable, independently checked artifacts with explicit assumption scopes	adjacent candidate; FK(i)	subsumption; difference witness; composition gate
checked subsumption certificates in DL (PAAR 2020)	https://ceurspt.wikidata.dbis.rwth-aachen.de/Vol-2663/paper-5.pdf	checked certificates for computed subsumptions; non-entailment countermodels proposed as future work	adjacent; partial FK(ii)	subsumption between refutations; scope typing; gate
mutant subsumption graphs (ICSTW 2014)	https://dl.acm.org/doi/10.1109/ICSTW.2014.20	kill-set inclusion between mutants; true subsumption undecidable; approximations	adjacent; partial FK(ii)	refutation values; scope types; difference witnesses; gate
ORCHESTRA deletion and derivability	https://repository.upenn.edu/cis_papers/655	provenance decides when a deletion removes derivability and when it does not	RLV(iv) adjacent	runtime values; grade; falsification ledger

T1. Closest work and the exact delta (static; no arena numbers). Clause key: (i) ledger of attempts as the warrant; (ii) dependence accounting; (iii) grade as a total, recomputable function of the ledger; (iv) exact dependency-directed demotion through cites. Every row states what is not implemented there; no row is a comparison against a strawman, and none of these works was measured in our arena. One row present in an earlier version of this table was

withdrawn because the repository it named could not be located when the reference list was re-checked, so the "closest single match' it was labelled is not claimed here; the table is a survey of located works, not an exhaustive one, and the Section 1 ceiling is phrased as an absence in the literature we surveyed for that reason.

3. Model and definitions

A value is an artifact with a payload, a kind, a context, and frozen cites. Attempts arrive as append records over the value's chain, and nothing rewrites the chain. The model below is the whole contract; the engine, the audit, and the lab render it and nothing else.

```
Fm1. Attempt, value, ledger, cite graph
Attempt = { refuter: RootId, family: RootId, seed: u64, outcome:
survived | refuted | inconclusive, cost: Nat, witness: Digest,
submittedBy: Identity }; Append = { prev: Digest, attempt: Attempt,
admittedBy: VerifierId, sig: Digest }; Value = { id: ValueId, op:
assert | derive, kind, payload, context, cites: frozen ValueId[],
chain: Append[] }; ValueId = Digest(op, kind, payload, cites, context,
genesis).
```

Roots are registered lineages with a declared author and declared coverage; a new code digest under an existing root is a version, not a new root. Cites are frozen at creation and every cite names an earlier value, so the cite graph is acyclic by construction and cycles are unrepresentable.

```
Fm2. Grade and declared dependence
gamma(v) = dead if any admitted attempt in chain(v) has outcome
refuted; else gamma(v) = min(K, classes(v), min over c in cites(v) of
gamma(c)), where classes(v) is the number of distinct declared
dependence classes among v's survived attempts; Dependent(a, b) :=
sameFamily(a, b) or declaredCoverageOverlap(a, b) >= 1/2; K = 3; seeds
are repetitions and never create a class.
```

Two survived attempts share a dependence class when their families match, or when the declared coverage of their families overlaps by at least one half; the class count is the number of connected components under that relation, so equal families and overlapping coverage both collapse to a single class. dead is absorbing and sits below 0.

> *Instance vs claim semantics — An attempt instance is evidence about a value in a context. It is not the claim itself, not a truth value, and never a statement about a person. The same payload can exist as a second instance in a different context with its own chain, so a warrant answers about an instance unless canonical context identity is adopted; refuting an explanation demotes the values that cite it and changes no learner score. Admission is the append point: whoever controls it controls what enters the ledger, and that trust root is disclosed rather than hidden.*

- I1' Auditable grade: any published grade equals the read-time recompute, transitively from leaves over the value's admitted chain; audit reports mismatches, malformed attempts, broken prev links, and anchor mismatches. Scope: arithmetic consistency and anchored tamper-evidence only, not truth and not authenticity of a well-formed fabrication.
- I2' No bookkeeping inflation: the class count counts distinct registered (refuter root, family root) pairs; identical attempts, replays, seed sweeps, re-signings, and cosmetic code versions add 0.

Declared residual: distinct roots controlled by one author still count separately.

- I3' Exact demotion: with frozen cites and unique ids, a value is dead iff a directed path through cites reaches a value with an admitted refuted attempt; precision and recall are 1 by construction, and derive on dead cites yields a dead value.
- I4' Append-only, monotone, absorbing: chains append only and outcomes are immutable; appends with no refuted attempt are monotone non-decreasing on every affected grade; any append containing a refuted attempt is absorbing-dead for the value and all its dependents, and no operation resurrects a value.

4. The grade algebra

The class count is a union over survived attempts. Surviving pairs are deduplicated by (refuter root, family root); two pairs share a declared dependence class when they share a family or when their families' declared coverage overlaps by at least one half, and the classes are the connected components of that relation. Seeds and cosmetic code versions change nothing: a replay adds no class. A refuter registered as a new lineage still counts as a new root, which is the declared residual that the registry makes visible through authorRoot.

K = 3. The cap is a product decision, not a measurement: it bounds the blast radius of a single value and keeps the grade readable, and it saturates. Above three declared dependence classes the grade stops moving, so diminishing returns are hidden and must be read from the ledger rather than from the number.

A derived value cannot out-warrant its weakest cite: the grade is also the minimum over cited grades, and any cited value that is dead makes the dependent dead. The closure is not code that propagates demotion; it is a read-time recursion over frozen cites, which is why over-demotion and under-demotion are impossible by construction once every cite names an earlier value.

Grades are pull-based. lambda(v) is a pure structural recursion memoized per call; there is no stored demotion state and no propagation pass. audit recomputes the closure transitively from leaves, compares the published grades against the recompute, and reports mismatches, malformed attempts, broken prev links, and anchor mismatches. Scope: arithmetic consistency against the ledger plus anchored tamper-evidence at the append point — not truth, not authenticity of a well-formed fabrication, and not a statement about evidence quality.

Frozen API (public engine surface)

```
export function D(chain: readonly Append[], registry: Registry): number;
export function lambda(store: Store, registry: Registry, id: ValueId): Grade;
export function why(store: Store, id: ValueId): { readonly chain: readonly
Append[]; readonly cites: readonly ValueId[] };
export function audit(
  id: ValueId,
  store: Store,
  registry: Registry,
  published: PublishedGrades,
  anchor: HeadAnchor | null,
): Report;
export function recomputeGrades(id: ValueId, store: Store, registry: Registry):
ReadonlyMap<ValueId, Grade>;
```

```
// Grade = "dead" | 0 | 1 | 2 | 3; K = WARRANT_MAX_GRADE = 3.
// D counts declared dependence classes; lambda is the read-time closure.
```

> *dead != 0 — The grade domain is dead plus 0 through K. dead is terminal, ordered below every k-warranted value, never serialized or rendered as 0, and never counted as survival by any grader. The churn metric maps dead to -1 inside its proxy only so that a terminal state can never look like a live grade.*

5. The Derived-Claim Arena

The arena is built to give the declared dependence relation its best shot and to make cheap counters look strong. There are 48 claims across 4 kinds (hint, explanation, difficulty label, prerequisite edge); each claim has a hidden boolean defective flag, and a defective claim carries exactly one of 8 defect classes. Sixteen refuters sit in 4 families with fixed blind spots — f0 is blind to classes {0, 1}, f1 to {2, 3}, f2 to {4, 5}, f3 to {6, 7} — and an attempt refutes a defective claim exactly when its class is outside its family's blind spot; correct claims always survive. A fifth oracle family with an empty blind spot exists only for scoring.

Five regimes run over the same claims, so every comparison is paired: R redundant, D diverse, C churn (R plus extra seeds in the same family), P replay (identical attempts repeated), and X correlated, where f2' and f3' share f2's blind spot so a family count sees three modes where only one exists. Each regime holds 12 conflict pairs whose two sides are matched on raw survivor count, so the raw-count baseline B1 ties by construction. The 200 seeds are frozen before the run and regenerate the corpus, the manifests, and the pairs.

Item	Value as frozen
Claims	48 = 4 kinds x 12 (hint, explanation, difficulty label, prerequisite edge)
Defect classes	8; a defective claim carries exactly one
Refuters	16 in 4 families; the family is the semantic unit and seeds are repetitions
Blind spots	f0 {0, 1}; f1 {2, 3}; f2 {4, 5}; f3 {6, 7}; an attempt refutes iff the class is outside its family's blind spot
Correlated families	regime X adds f2' and f3', both sharing f2's blind spot
Regimes	R redundant; D diverse-label control (coincides with R under the binary blind-spot model, since a defective claim can only survive its one blind family); C churn (R plus extra seeds in one family); P replay (identical attempts repeated); X correlated

Item	Value as frozen
Conflict pairs	12 per regime, both sides matched on raw survivor count
Seeds	200 frozen seeds; s <- (1664525 * s + 1013904223) mod 2^32
Graders	B0 survivor present; B1 raw survivor count; B2 survival share; B3 pinned survivor count; B4 alive/dead belief; B5 distinct surviving families (the strongest cheap adversary we constructed, capped at K = 3 - a cap that matters in regime X, where a defective claim can survive three families); B6 distinct attempt tuples (kill-criterion bait); B7 declared dependence classes (shipped); B8 oracle blind-spot classes (analysis only)
Oracle	one family with an empty blind spot, used for scoring only; never implemented in the engine or the lab

T2. Corpus, regimes, and graders (generator constants). The arena is pure TypeScript with zero dependencies. Ground truth comes from the generator's blind-spot table; graders never define it. Regimes run over the same claims, so every comparison is paired, and each regime holds 12 conflict pairs whose two sides are matched on raw survivor count so the raw-count baseline ties by construction.

```
                    Fm3. Frozen seed LCG
    s <- (1664525 * s + 1013904223) mod 2^32; every regime, claim
assignment, and conflict pair is regenerated from the frozen seed list,
            and no other randomness exists in the arena.
```

The same generator and seed list produce the corpus, the manifests, and the pairs on every run. The digest ca0cda0f562b8c10 pins the 200-seed sweep.

> *Anti-gaming of the experiment — Ground truth comes from the generator's blind-spot table, never from a grader. The conflict pairs are matched on raw survivor count, so B1 ties instead of losing to a construction artifact, and B5 — distinct surviving families, capped at the same K = 3 as the shipped grade — is the adversary, not a strawman; that cap is a design choice and it matters in regime X, where a defective claim can survive three families. B8 is an explicit upper bound so that 'the corpus is too easy' cannot rescue a failure, and in regime X it coincides with B7 by construction rather than confirming it. Thresholds are stated at the 5th percentile over 200 seeds frozen before the run, and the seed list and the thresholds are part of the reviewed commit; where the per-seed value does not vary, the percentile threshold is non-binding, which is the case for every pair-win and AUC cell. B6 is the pre-declared bait: if the syntactic tuple variant were the shipped relation, P1 and P2 would be its only path to significance, and F1 predicts its kill.*

6. Experiments

Before any number: the arena's ground truth and the shipped grader's dependence relation are two readings of the same declared table. In regime X, families F2P and F3P carry F2's blind spot and overlap its declared coverage, so B7 merges exactly what B8 treats as one mode, and B7 = B8 by construction. Every comparison involving B7 and B8 in T3-T5 is therefore a consistency check on the implementation, not a measurement of grader quality, and the +1.000 pair-win, +1.000 AUC, and +1.000 AP@12 gaps over B5 are consequences of the declared table rather than findings about evidence fusion. What that arena does reproduce is the published result that naive count fusion

collapses when sources are correlated (Dong, Berti-Equille & Srivastava 2009; Knight & Leveson 1986), which is a reason to discount correlated counts and not a new result. The two informative contrasts in the arena are B5 versus B6 (dedup by family versus dedup by attempt tuple) and B7 versus B5 under seed churn, where the claim is invariance rather than superiority.

Every table and figure below is transcribed from the arena artifact (digest ca0cda0f562b8c10, 200 seeds), and the permanent gate re-runs the arena and pins the headline cells. Means and 5th percentiles are over the seed sweep; criteria apply to the 5th percentile except where a criterion names churn or demotion. One property of these cells needs stating because it makes the threshold machinery weaker than it looks: for pair-win and AUC the per-seed value is identical across all 200 seeds, so the mean and the 5th percentile coincide and the percentile threshold is satisfied exactly when the mean is. Those cells carry no sampling variability, and the seed sweep is a determinism check, not a stability check. Only the AP@12 cells for the count baselines show spread (mean 0.5175, 5th percentile 0.250).

Grader	R	D	C	P	X
B0	0.500 / 0.500	0.500 / 0.500	0.500 / 0.500	0.500 / 0.500	0.500 / 0.500
B1	0.500 / 0.500	0.500 / 0.500	0.500 / 0.500	0.500 / 0.500	0.500 / 0.500
B2	0.500 / 0.500	0.500 / 0.500	0.500 / 0.500	0.500 / 0.500	0.500 / 0.500
B3	0.500 / 0.500	0.500 / 0.500	0.500 / 0.500	0.500 / 0.500	0.500 / 0.500
B4	0.500 / 0.500	0.500 / 0.500	0.500 / 0.500	0.500 / 0.500	0.500 / 0.500
B5	1.000 / 1.000	1.000 / 1.000	1.000 / 1.000	1.000 / 1.000	0.000 / 0.000
B6	0.500 / 0.500	0.500 / 0.500	0.500 / 0.500	0.500 / 0.500	0.500 / 0.500
B7	1.000 / 1.000	1.000 / 1.000	1.000 / 1.000	1.000 / 1.000	1.000 / 1.000
B8	1.000 / 1.000	1.000 / 1.000	1.000 / 1.000	1.000 / 1.000	1.000 / 1.000

T3. Pair-win rate by grader and regime (mean / 5th percentile over 200 seeds). Fraction of the 12 matched conflict pairs per seed where the grader ranks the non-defective side above the defective side; ties count 0.5. B0-B4 and B6 sit at 0.500 in every regime. B5, the strongest cheap count baseline we constructed (distinct surviving families, capped at K = 3, and that cap is load-bearing in X where a defective claim can survive three families), separates in R/D/C/P and collapses to 0.000 in the correlated regime X, where family-dedup sees three modes where only one exists. B7 keeps 1.000 and the analysis-only B8 matches it, because in X they are the same function of the same table (Section 6). The mean and the 5th percentile are identical in every cell here: the per-seed value does not vary, so the seed sweep is a determinism check and not a stability check.

Grader	R	D	C	P	X
B0	0.500 / 0.500	0.500 / 0.500	0.500 / 0.500	0.500 / 0.500	0.500 / 0.500
B1	0.500 / 0.500	0.500 / 0.500	0.500 / 0.500	0.500 / 0.500	0.500 / 0.500
B2	0.500 / 0.500	0.500 / 0.500	0.500 / 0.500	0.500 / 0.500	0.500 / 0.500
B3	0.500 / 0.500	0.500 / 0.500	0.500 / 0.500	0.500 / 0.500	0.500 / 0.500
B4	0.500 / 0.500	0.500 / 0.500	0.500 / 0.500	0.500 / 0.500	0.500 / 0.500
B5	1.000 / 1.000	1.000 / 1.000	1.000 / 1.000	1.000 / 1.000	0.000 / 0.000
B6	0.500 / 0.500	0.500 / 0.500	0.500 / 0.500	0.500 / 0.500	0.500 / 0.500
B7	1.000 / 1.000	1.000 / 1.000	1.000 / 1.000	1.000 / 1.000	1.000 / 1.000

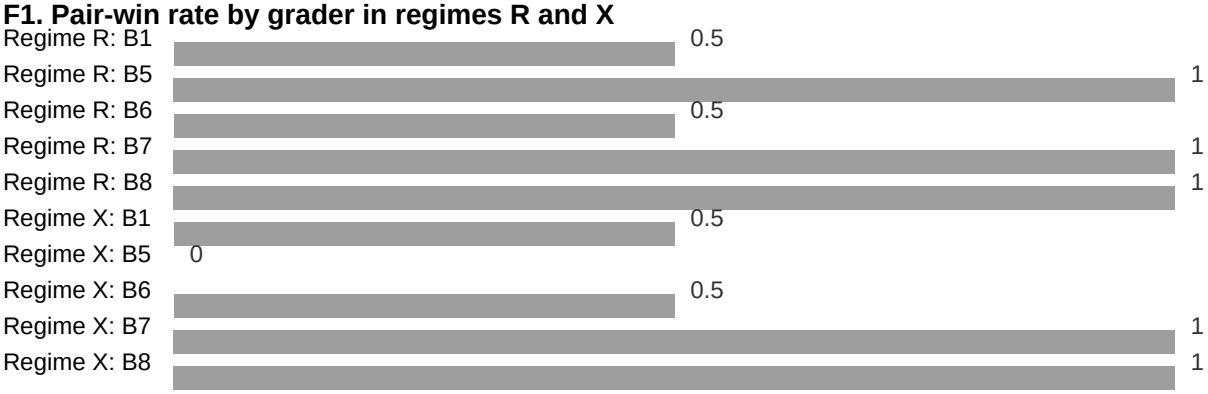
Grader	R	D	C	P	X
B8	1.000 / 1.000	1.000 / 1.000	1.000 / 1.000	1.000 / 1.000	1.000 / 1.000

T4. Calibration AUC by grader and regime (mean / 5th percentile over 200 seeds). Tie-aware AUC of each grader's score against the hidden defective label over all 48 claims, per regime. Label 1 is the non-defective side. The same split as T3 appears: B5 reads 0.000 in X, B7 reads 1.000, and the analysis-only B8 oracle also reads 1.000 in X because the three overlapping roots share one blind spot - that equality is circular rather than corroborating, since B7 and B8 read the same declared table (Section 6). B8 bounds the signal the corpus carries and is never implemented by the system. Mean and 5th percentile coincide in every cell, as in T3.

Metric	Scope	Measured	Reading
AP@12, B0	regime X (mean / p5)	0.5175 / 0.250	chance-like on the matched pairs
AP@12, B1	regime X (mean / p5)	0.5175 / 0.250	chance-like on the matched pairs
AP@12, B2	regime X (mean / p5)	0.5175 / 0.250	chance-like on the matched pairs
AP@12, B3	regime X (mean / p5)	0.5175 / 0.250	chance-like on the matched pairs
AP@12, B4	regime X (mean / p5)	0.5175 / 0.250	chance-like on the matched pairs
AP@12, B5	regime X (mean / p5)	0.000 / 0.000	strongest cheap count baseline we constructed (distinct surviving families, K cap load-bearing in X)
AP@12, B6	regime X (mean / p5)	0.5175 / 0.250	chance-like on the matched pairs
AP@12, B7	regime X (mean / p5)	1.000 / 1.000	shipped declared-dependence relation
AP@12, B8	regime X (mean / p5)	1.000 / 1.000	analysis-only oracle ceiling
Seed churn, mean change in grade	R to C	0.000	seeds never create a dependence class
Seed churn, pair flips	R to C	0.000	no matched pair flips toward the churned side
Demotion precision	reachable-set check	1.000	TMS retraction check, not the novelty
Demotion recall	reachable-set check	1.000	kill propagates to every dependent through cites

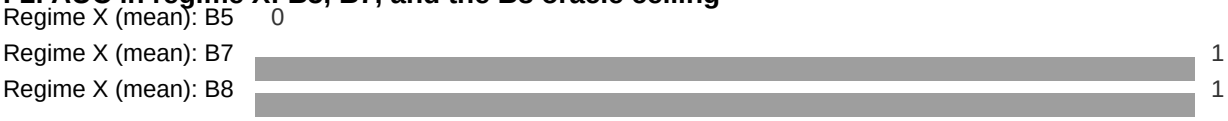
T5. Decision metrics: AP@12 in regime X, seed churn, and demotion sanity. AP@12 ranks values by suspicion (1 divided by grade, original claim index as the tie-break), replays the top 12 with the oracle, and reports the share that is actually defective; values are mean / 5th percentile over 200 seeds in regime X. AP@12 is a replay of the top 12 against an oracle that reads the same declared table the shipped relation B7 reads, so in regime X the B7 1.000 is a consistency check on the implementation and not evidence that the grade ranks real defects: with 48 claims and a ranking signal that is the ground-truth rule, this metric cannot fail in the way a held-out metric could. It is reported here and in F3 rather than headlined. Churn is measured from R to the seed-swamped regime C. Demotion precision and recall compare the read-time closure against the reachable-set oracle; that check exercises TMS-style retraction and is reported as such, never as the novelty.

F1. Pair-win rate by grader in regimes R and X



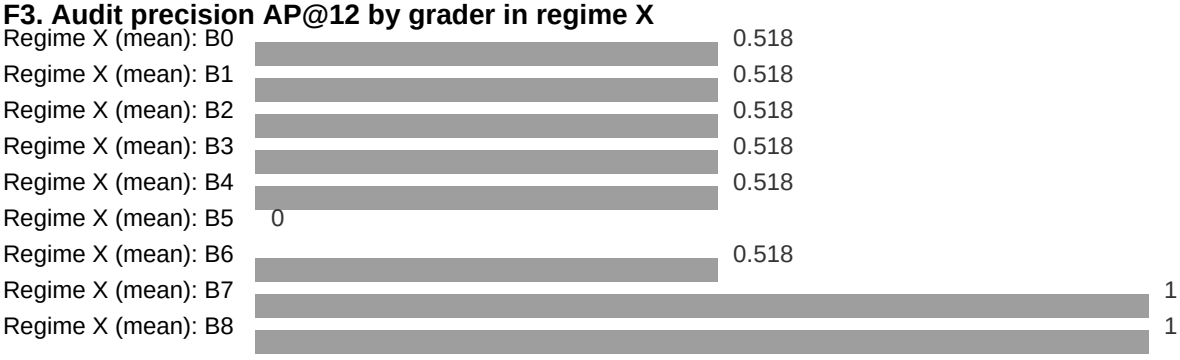
B1, B5, B6, B7, and B8, means over 200 frozen seeds. Regime R is redundant; regime X is correlated, where f2' and f3' share f2's blind spot. B5 falls to 0.000 in X while B7 holds 1.000, and in X B7 and the analysis-only B8 are the same function of the same declared table, so the gap is a property of the generator. The dashed line is the 0.75 decision gate from the significance attack, shown for reference; it is not a criterion the arena applies to this figure, and it is not met in X by any arm.

F2. AUC in regime X: B5, B7, and the B8 oracle ceiling



Mean tie-aware AUC in the correlated regime, from T4's column X. The measured headroom B8 minus B5 is 1.000. The dashed line marks the 0.05 headroom threshold from F3: a gap that small would have killed the idea, not just the witness. B8 is analysis only and is never implemented in the engine or the lab. In X, B8 and B7 are the same function of the same declared table, so this figure shows that the shipped relation reaches the generator's own ceiling there, not that either is better than a count.

F3. Audit precision AP@12 by grader in regime X



Mean AP@12 over 200 seeds. Values are ranked by suspicion (1 divided by grade, original claim index as the tie-break), the top 12 are replayed with the oracle, and the bar is the share actually defective. The dashed line is the 0.75 gate from P4. B5 sits at 0.000; B7 and the analysis-only B8 sit at 1.000. That equality is circular, not corroborating: the oracle replayed here reads the same declared blind-spot table that B7 merges, and with 48 claims the ranking signal is the ground-truth rule, so this figure cannot fail the way a held-out metric could. It is a consistency check on the implementation and is reported here rather than headlined.

ID	Measured	Threshold	Status	Reading
P1	0.5	>= 0.15 (p5, R: B7-B1)	pass	declared dependence beats raw count under redundancy
P2	1	>= 0.10 (p5, X: B7-B5)	pass	AUC gap of 1 in the correlated regime

ID	Measured	Threshold	Status	Reading
P3	0	seed churn mean <= 0.1 and flips <= 0.05	pass	flip rate 0; seeds are repetitions
P4	1	AP@12 >= 0.75 (p5, X)	pass	gap over B5 of 1; the ranking changes audit outcomes
P5	0	>= -0.02 (p5, every regime)	pass	no calibration regression against B5
F1	true	B6 does not beat B1 in R and is dominated by B7 in X	fail	expected kill: the syntactic tuple variant is a dedup count
F2	false	seed-only churn must not raise gamma	pass	seeds never add a dependence class
F3	1	> 0.05 headroom over B5 (p5, X, B8-B5)	pass	the oracle ceiling confirms remaining signal
F4	1	reported: collapsed-manifest X advantage >= 0.10	pass	X pair sides already hold one attempt per family, so the collapse is a no-op

T6. Kill-criterion status: P1-P5 and F1-F4 as the arena reports them. Readings are the arena's own recorded criterion table. P1-P5 are the pass criteria; F1-F4 are the pre-declared kill criteria. F1's status is 'fail' because the criterion describes B6: measured true means the predicted kill of the syntactic tuple variant is confirmed. B7, the shipped relation, passes P1-P5. F2-F4 pass, which means none of their kills fired.

> *Outcome space: P1-P5, F1-F4 — The pinned thresholds are: P1, pair-win of B6 or B7 minus B1 is at least 0.15 at the 5th percentile in regime R. P2, pair-win of B7 minus B5 is at least 0.10 and AUC of B7 minus B5 is at least 0.10 at the 5th percentile in regime X. P3, seed-churn mean change in grade is at most 0.1 and the pair-flip share is at most 0.05. P4, AP@12 is at least 0.75 and at least B5 plus 0.10 at the 5th percentile in regime X. P5, B7 is at least B5 minus 0.02 at the 5th percentile in every regime. F1, B6 does not beat B1 in R and is dominated by B7 in X (measured true: the syntactic tuple variant is killed as predicted). F2, P3 fails for the shipped relation (seed-only churn raises the grade; measured false). F3, AUC(B8) minus AUC(B5) at most 0.05 in X (measured 1.000, so the oracle ceiling confirms remaining headroom). F4, the reported duplicate-suppression ablation: the collapsed-manifest X advantage of B7 over B5 stays at least 0.10 (measured 1.000; the X pair sides already carry one attempt per family, so the collapse is a no-op and the X advantage is not duplicate suppression). Measured: P1-P5 pass; F2, F3, and F4 do not fire. The honest path if F1 or F2 had held for the shipped relation would have been to publish the negative result and to say so in the product copy; B7 is headlined only because P2-P4 pass at the 5th percentile.*

Reading the tables. Ties count 0.5 in pair-win; thresholds are stated at the 5th percentile (nearest rank over the 200 frozen seeds) and both the mean and the percentile are printed. For the pair-win and AUC cells the per-seed value is identical across all 200 seeds, so mean and 5th percentile coincide and the percentile threshold is satisfied exactly when the mean is; those cells carry no sampling variability and the seed sweep is a determinism check, not a stability check. Only the AP@12 cells for the count baselines show spread (mean 0.5175, 5th percentile 0.250). B8 is analysis-only: it is the oracle family's blind-spot-class count, present for the ceiling argument and never implementable by the system; in regime X it coincides with B7 by construction, so their agreement is a consistency check. B5 is the strongest cheap adversary we constructed (distinct surviving families, capped at K = 3 - a cap that is load-bearing in regime X, where a defective claim can survive three families), not a strawman, but it is also a capped counterfactual rather than an

adversarial optimum. B0-B4 and B6 are constant or chance-like on the matched pairs, which is what count-matching is for. The aggregates for regimes R, D, C, and P coincide at the reported precision; X is the discriminating regime, and only the declared dependence relation stays at 1.000 there.

7. Failure cases & limitations

> *Limitations that apply to every number in this paper — Four of them, stated once so they cannot be lost in a citation. (1) No human-participant data, and no real content of any kind: the arena's 48 claims, 8 defect classes, 16 refuters, and 12 conflict pairs are generated, and no real hint, explanation, difficulty label, or prerequisite edge is audited anywhere in this paper. (2) No external replication. The arena is one generator with one defect-class model and one frozen seed schedule, and nobody outside this project has attempted to reproduce it; nothing here is evidence about any deployed content pipeline. (3) Replayed against recomputed. The gate recomputes the arena from source and compares the result with a digest and with pinned cells, which proves that the code is deterministic and that the recorded run has not drifted; it does not prove that the generator is a faithful model of anything, because the ground truth and the shipped grader B7 are both read off one declared table (Section 6). Correctness for the contract itself rests on the Section 4 algebra and on the demotion sanity check, not on the arena. Only the Philox and FIPS known-answer vectors elsewhere in this portfolio are external oracles, and no claim in this paper rests on one. (4) No experiment here measures a generative-AI tutor. The three motivating studies in Section 1 concern a legal research tool, a sepsis model, and a generative-tutor field experiment; they motivate the problem and they are other people's evidence about other systems, and no experiment in Sections 5 to 9 bears on how a generative tutor behaves or how its output should be graded.*

The contract has a visible envelope. Every item below is a limit of the shipped system or of the experiment, not a caveat added after the fact.

- dead is not 0: the grade domain is dead plus 0 through K; dead is terminal, ordered below every k-warranted value, and never rendered or counted as 0.
- Declared dependence has blind spots: D is syntactic (same declared family, or declared coverage overlap of at least one half). Shared method, shared input domain, and shared blind spot survive it; Knight and Leveson's correlated-failure result is the standing counter-pressure, and regime X is built to show a family-dedup count being fooled by it.
- Fabricated but consistent ledgers are out of scope: a producer that controls the append point can fabricate a well-formed, internally consistent ledger, and audit-by-recompute will recompute from it. The admission trust root is the append point; hash chains are tamper evidence, not signatures.
- Absence of attempts is not survival: 'no challenge recorded' is not a positive warrant, and the empty ledger and the all-inconclusive ledger both grade 0.
- Saturation at K: the cap is 3; above it, more surviving classes do not move the grade and diminishing returns are hidden.
- Refuter cost is recorded, never weighed: a cheap attempt and an expensive attempt count alike.
- Instance versus claim semantics: values are artifacts, and the same payload can exist as a different instance in a different context, so a warrant answers about an instance unless canonical context identity is adopted.
- B8 is not implementable: it is a generator-side oracle used for scoring only, never in the engine or the lab.

- Demotion sanity tests retraction, not significance: precision and recall are 1 by construction against the reachable-set oracle; it checks TMS-style retraction and must not be sold as the novelty.
- The corpus is synthetic: ground truth is generated from the blind-spot table, never inferred from a grader; 48 claims and 12 pairs per regime are small, and no real content is audited.
- The arena is circular where it looks strongest: in regime X the shipped relation B7 and the analysis-only oracle B8 are the same function of the same declared table, so the +1.000 pair-win, AUC, and AP@12 gaps over B5 are consequences of the generator. What the arena supports is stability under seed churn and the pre-declared kill of B6; the collapse of count fusion under correlated sources is the published result it reproduces, not a discovery (Dong, Berti-Equille & Srivastava 2009; Knight & Leveson 1986).
- The 200 seeds are regenerations of a deterministic generator, not 200 observations: for pair-win and AUC the per-seed value is identical, so mean and 5th percentile coincide and the percentile threshold is non-binding. Only the AP@12 count-baseline cells show spread.
- No truth guarantee: a survived attempt does not make a claim true; a refuted attempt is taken on the refuter's authority, and a false refutation is permanent because kill is absorbing.
- One author can register many roots and families; the registry makes author concentration visible, but semantic dependence is undecidable and remains a declared residual that K caps rather than fixes.

8. Product: the Warrant Lab

The product is a read-only contestability lab at /warrant. It renders one value at a time: kind, payload, warrant line, chain, cites, the exact grade definition with the cap and saturation note, the declared dependence relation, and a demotion preview that shows the exact affected set through cites. It is browser-local: no storage, no network, no server call, no learner state, and no clock reads — challenges are logical indices in a local fixture, never Date.now(). The route imports the pure engine and local fixtures only, never grading, sync, storage, review, or curriculum modules.

- Ships: the ledger, why, and audit views over local fixture values; the grade with its definition, cap, and saturation note; the declared dependence relation; the demotion preview through cites; logical challenge indices; and honest copy that names 'no challenge recorded' as an absence, not a warrant.
- Ships: an always-visible caveat — audit is arithmetic consistency plus anchored tamper-evidence at the append point; survived attempts are recorded attacks, not truth.
- Cut: any write path, storage, sync, or network call; any coupling to grading, curriculum, review, or learner state; any effect on a score, path, rank, or teacher view; any percent, score, or rating rendering of a grade; any learner-facing consequence of a refuted learner-authored explanation.
- Cut: the B8 oracle, which is analysis-only, and any rehabilitation or re-mint path — kill is absorbing and a refuted claim cannot be re-asserted in its context.

> *Honest copy (the three allowed strings) — "Survived 3 check-families; 0 refutations; 1 challenge on record." · "No challenge recorded." · "Warrant: 2 of 3 independent checks (cap 3)."*

- Never attach any assessment word from the copy deny-list to a claim's grade; the lab shows counts of declared check-families and the recomputed state, nothing resembling a quality score.
- Never use a grade for assessment or grading of learners, mastery, readiness, XP, streaks, ranks,

path gating, recommendations, or teacher reporting. Grades rank content contestability, never people.
- Never imply that counts of survived attempts establish failure independence; say declared check-families and disclose blind spots (Knight-Leveson).
- Never imply that audit is truth; audit checks arithmetic against the ledger, and hash chains are tamper evidence, not authenticity.
- Never sell 'never challenged' as a positive warrant; absence of attempts is not survival.
- Never let a refuted learner-authored explanation count against the learner; refute the claim, never the person, and never change learner scores.
- No claim of improved learning outcomes from the ledger itself; the problem evidence in section 1 is about the problem, not this solution.

9. Reproducibility & the evidence gate

Everything is offline, deterministic, and dependency-free. The evidence script runs the 200-seed arena and writes a canonical artifact; the permanent gate re-runs the arena twice, checks that both runs produce one digest equal to the pinned constant, recomputes the headline cells, and scans the engine for forbidden runtime facilities. The artifact embeds the arena digest ca0cda0f562b8c10 (the evidence script adds its own outer digest over the artifact), and this paper transcribes that artifact's arena tables.

Reproducing the reported results

```
# evidence: regenerate the Derived-Claim Arena artifact (200 seeds)
bun run scripts/warrant-evidence.ts

# permanent gate: two full arena runs, pinned aggregates, P/F statuses,
# purity scan, runtime budget (wired in CI)
bun run verify:warrant

# contract, arena, and safety tests
bun test tests/warrant*.test.ts
```

- Determinism: two full arena runs produce one digest equal to the pinned ca0cda0f562b8c10.
- Criteria statuses: P1-P5 pass, F1 fails as predicted (the syntactic tuple variant is killed), and F2-F4 pass.
- Pinned aggregates: headline pair-win, AUC, AP@12, churn, and demotion equal the reviewed run.
- Decisive margins: the P2/P4 margins are recomputed from the pinned aggregates (pair-win and AUC gaps of at least 0.10 in X, AP@12 at least 0.75, churn 0).
- F1 diagnosis: B6 does not beat B1 in R and is dominated by B7 in X.
- Purity: no clock, randomness, network, process, file-system, or console tokens in the engine; the oracle grader is hidden from the barrel.
- Runtime budget: two full arena runs inside 60 seconds.
- Summary: the machine-readable final line reports 8 passed, 0 failed, and the pinned digest.

> *No auto-update path — There is no flag or environment variable that rewrites the pinned digest or any pinned aggregate. Drift fails the gate: a changed count must be updated in the gate constant and in*

> *this paper in one reviewed commit, and the gate itself never writes the artifact.*

10. Future work

- Refuter libraries per claim kind: hints, explanations, difficulty labels, and prerequisite edges need different refuters; the arena models families and blind spots, not real refuters for real content.
- Semantic coverage dependence: replace declared coverage overlap with a measured notion of method or input-domain overlap, and test whether it beats the declared relation on the same matched pairs.
- Signed ledgers for third-party re-audit: admission signatures and a transparency log would move the trust root beyond the append point; today hash chains are tamper evidence, not signatures.
- Attempt aging: attempts currently never age; a decay or epoch rule would let stale survival evidence be reconsidered without silent repair.
- Combination with behavioral deltas: BDL-style behavioral signatures could witness that two attempts cover different behavior, sharpening dependence beyond declared families.
- An explicitly scoped rehabilitation operation: kill is absorbing, so a refuter bug permanently kills a cone; a governed supersede operation carrying its own challengeable ledger is the one repair path to specify before shipping any.

References

1. [withdrawn] mmnto-ai/totem capability falsification (commit fc3f4114) — the repository could not be located when the reference list was re-checked; the T1 row and the Section 2 sentence that relied on it were rewritten rather than left standing, and this entry is not to be cited. https://github.com/mmnto-ai/totem/blob/fc3f4114/packages/core/src/capability/falsification.ts
2. falsifyr attack leaderboard keyed by family and seed. https://github.com/msaule/falsifyr/
3. falsification-ledger: append-only hash-chained ledger. https://github.com/foolproof-labs/falsification-ledger
4. falsification-ledger hit-rate report. https://pypi.org/project/falsification-ledger/
5. FalsiFlyer AUDIT_LEDGER_SPEC. https://github.com/subvurs/FalsiFlyer/blob/main/docs/AUDIT_LEDGER_SPEC.md
6. Doyle 1979, A Truth Maintenance System. https://dspace.mit.edu/handle/1721.1/5733
7. de Kleer 1986, An Assumption-Based TMS (Artificial Intelligence 28). https://doi.org/10.1016/0004-3702(86)90080-9
8. Green, Karvounarakis & Tannen 2007, Provenance Semirings. https://dl.acm.org/doi/10.1145/1265530.1265535
9. Buneman, Khanna & Tan 2001, Why and Where: A Characterization of Data Provenance. https://dl.acm.org/doi/10.5555/645504.656274
10. Verheij 1995/1999, Accrual of arguments. https://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.41.4458
11. Amgoud, Ben-Naim et al. 2017, Weighted gradual semantics (IJCAI). https://doi.org/10.24963/ijcai.2017/9
12. Truth discovery with dependent sources (SIGKDD Explorations 2016). https://doi.org/10.1145/2897350.2897352
13. Josang 2001, Subjective logic / evidence-based security logic. https://doi.org/10.1016/s0218-4885(01)00083-1
14. Dempster-Shafer dependence caveat. https://arxiv.org/abs/1303.1518

15. Knight & Leveson 1986, An experimental evaluation of the assumption of independence in multiversion programming. https://www.csc.kth.se/utbildning/kth/kurser/DA2210/vettig13/Seminarier/KnightLeveson.pdf
16. Assurance 2.0 defeaters (Rushby et al., 2024). https://www.csl.sri.com/~rushby/papers/defeaters24.pdf
17. EviBound: machine-checkable evidence gates for claims (arXiv 2511.05524). https://arxiv.org/abs/2511.05524
18. Evidence-ledger adjudication: claim, evidence packet, and route. https://arxiv.org/html/2607.26512v1
19. SV-COMP 2024 software verification witnesses 2.0. https://www.sosy-lab.org/research/pub/2024-SPIN.Software_Verification_Witnesses_2.0.pdf
20. Checked subsumption certificates in description logics (PAAR 2020). https://ceurspt.wikidata.dbis.rwth-aachen.de/Vol-2663/paper-5.pdf
21. Mutant subsumption graphs (ICSTW 2014). https://dl.acm.org/doi/10.1109/ICSTW.2014.20
22. ORCHESTRA deletion and derivability. https://repository.upenn.edu/cis_papers/655
23. falsifyr survival vignette: interpreting survival scores. https://mirrors.linux.iu.edu/CRAN/web/packages/falsifyr/vignettes/interpreting-survival-scores.html
24. Bloomfield, Netkachova & Rushby, defeaters (arXiv 2405.15800). https://arxiv.org/abs/2405.15800
25. Magesh, Surani, Dahl, Suzgun, Manning & Ho 2024, RegLab evaluation of leading AI legal research tools (arXiv 2405.20362). https://arxiv.org/abs/2405.20362
26. Wong et al. 2021, External evaluation of the Epic Sepsis Model, JAMA Internal Medicine. https://jamanetwork.com/journals/jamainternalmedicine/fullarticle/2781307
27. Bastani et al. 2024, Generative AI tutoring field experiment, PNAS. https://www.pnas.org/doi/abs/10.1073/pnas.2422633122