

# When the Benchmark Is Just a Rubber Stamp: How Scaffolding Alone Scored 27% on Legal Review



*A model-deleted floor and a planted-issue answer key for legal-agent benchmarks*

---

AUTHOR

Aaron Kelly

PUBLISHED

Jul. 05, 2026

AFFILIATION

Litil

# Table of Contents

---

|    |                                                         |
|----|---------------------------------------------------------|
| 1  | 1. The memo nobody wrote                                |
| 2  | 2. How we got here                                      |
| 3  | 3. What a harness is, and what one score can't separate |
| 4  | 4. The experiment                                       |
| 5  | 5. What we found                                        |
|    | 5.1 The empty rig, with the model removed               |
|    | 5.2 Put a top model back, and the rig fights it         |
| 6  | 6. Is the AI grader dumb?                               |
| 7  | 7. The checklist                                        |
| 8  | 8. What we're releasing, and what we're not claiming    |
| 9  | Endnotes                                                |
| 10 | Reproducibility note (frozen evidence)                  |

# 1. The memo nobody wrote

---

Here is the whole paper in one artifact.

An 86-kilobyte legal memorandum. Properly formatted, divided into sections, citing the right authorities, appearing to flag issues, and reading at a glance like the work of a diligent junior associate. It was produced with zero involvement from any AI model, large or small. No model read the documents; no model wrote a sentence. The software *around* the model assembled the whole thing by itself. Then we handed it to an AI grader, which marked it against a 33-point rubric and awarded it 27%.

Twenty-seven percent, for a memo nobody wrote.

Any litigator will recognize the maneuver, because there is a name for it: *papering the file*. It manufactures the appearance of diligence, and here that manufacture was automated. This paper is about that 27%: what it means, why it is so easy to earn, and how to measure a benchmark so that this particular illusion can no longer fool you.

## 2. How we got here

---

We did not set out to study AI graders. We set out to reproduce a result we admired.

Joel Niklaus's *Don't Train the Model, Evolve the Harness* [\[1\]](#) is one of the cleaner demonstrations of a genuinely surprising fact: you can raise an AI agent's benchmark score by +16.7 points (with held-out data, on Harvey's legal-agent benchmark [\[2\]](#)) *without changing the model at all*. The AI itself was frozen. Everything that moved was the software scaffolding wrapped around it. Nothing in this article disputes that result; if anything, our own experiment reproduces its central lesson. It was the inspiration for what follows, and it is exactly the kind of finding we wish we'd had the instruments to fully explain.

So we tried to run the same style of experiment in our own lab, on a legal document-review agent: steer an automated loop that mutates the scaffolding and keeps whatever scores higher.

The luckiest thing that happened to us was an optimization loop that refused to lie to us. Across roughly nineteen rounds it improved on nothing. Not one candidate beat the incumbent, and the loop's own quality gates rejected every single one, so in the ordinary sense, ours failed. But rigor of that kind leaves evidence behind, and the last rejected candidate left something strange sitting in the working directory. Its setup step had been rewired to *write the deliverable*

*itself*: read every source document, draft the issue cards, render a complete memorandum, all before the model took its first turn. The model was left one turn, in which it typically just reported that the work looked done.

Which raised a suspicious question: what does *that* contraption score with the model deleted entirely? The answer was the 86-kilobyte memo above, and the 27%.

The idea sounds too simple to work, and we only trusted it because we stumbled into it. We checked before we let ourselves believe it, and the exciting version, that our optimizer had invented this dodge on its own, quietly died: the drift was steered by us, the human operators; the dodge *lost* to the ordinary baseline under the loop's own independent grader (51% versus 81%); and the loop's safeguards had already caught and rejected every version of it. So we killed the exciting claim and kept the checkable one, the stronger of the two.

### 3. What a harness is, and what one score can't separate

---

Start with two plain definitions, because everything turns on them.

The harness is the software rig around the model: its tools, the documents it can open, the checklist it follows, the code that formats and files the final deliverable. The model is the reasoning engine; the harness is everything else in the room. When people say "an AI agent," they mean the two bolted together.

The judge is an AI grader: a second model that reads the finished deliverable and marks it against a written rubric, criterion by criterion, the way a bar examiner works down a model answer.

That raises the question. When an agent's benchmark score goes up, three completely different things could have happened, and the score cannot tell you which:

- Capability: the model got smarter. It actually found more of the issues.
- Conformance: the paperwork got tidier. The same analysis, delivered in a cleaner shape the grader rewards.
- Appeasement: the grader got charmed. The deliverable *looks* like good work, so it is marked as good work.

Did the model get smarter, did the paperwork get tidier, or did the grader get charmed? A benchmark number folds all three into a single figure, because the judge only ever sees one

finished document and marks it. The problem is structural, not specific to any one benchmark. In single-score judged-output benchmarks, including everything our own lab had built, these three cannot be separated, because the judge's score is the only measurement in the building. There is nothing else to check it against.

And yet the fix is embarrassingly simple: delete the model and run the harness anyway. Whatever the harness scores on its own is the harness's score; only the lift above that is the model's. You cannot see the shape of the scaffolding while the model is still wearing it — the only way to discover your collar became an exoskeleton is to remove the model and watch.

## 4. The experiment

---

To tell the three apart you need something the AI grader does not have: an answer key you never optimized against.

So we built one, the way a law professor builds a final. We took five realistic legal matters and hid 51 issues in them for the agent to find. Twelve came with the benchmark's original data-privacy task. The other thirty-nine were freshly authored and planted by a practicing general counsel across four new matters: a master-services-agreement markup, a share-purchase indemnity and escrow redline, a credit-agreement covenant review, and a diligence pass over a target's contract portfolio. Independent reviewers then confirmed that every one of the 51 is findable from the documents alone. It is exactly the law-school-exam setup: the professor hides specific issues in the fact pattern and grades you on how many you spot.

Three rules kept us honest.

First, we committed the whole exam before running anything: every document, every planted issue, every scoring rule, cryptographically fingerprinted and time-stamped. You cannot move the goalposts after you have seen where the ball landed.

Second, we graded every memo twice, with two very different graders. One is the AI judge, marking against the rubric. The other is a deliberately dumb, mechanical checklist that does one thing: count the planted issues the memo *actually spotted*, stated as a finding, not merely quoted from the source. The dumb checklist has no opinions about polish. It cannot be charmed. We even verified it isn't a stingy grader: run against gold-standard memos that do flag every planted issue, it credits 39 of 39 with zero false alarms. Those 39 are the newly authored issues across the four new matters; the benchmark's original twelve were validated separately — the checklist reproduced the known historical result on an archived memo (3 of 12) and

agreed with the AI judge’s issue-level verdicts on the memos we cross-checked. So when it reports a low number, that is a fact about the memo, not a defect in the checklist.

Third, for the headline numbers we deliberately used one task we had verified was clean. Its answer is not partly baked into the scaffolding code, its documents are unmodified, and it never appeared anywhere in our optimization loop. (We did find, and the [technical companion](#) discloses, that some scaffolds elsewhere hard-code parts of an answer; we simply steered around that by choosing a task where they don’t.)

## 5. What we found

---

The individual controls are familiar. What was missing was a legal-agent audit that measured all three on the same artifacts: the model-deleted floor, the AI-judge score, and issue-level ground truth.

### The empty rig, with the model removed

A few terms recur in the tables below. The canonical harness is the upstream runtime we build on, an open-source legal harness by Duy Nguyen and Leon Reilly [\[3\]](#); canonical + 2 minor fixes is that same runtime with two small, benign corrections applied. The left-behind rig is our optimization loop’s modified copy of that runtime, left sitting in the working directory. Its bypass is a memo-writing shortcut wired into the setup step: deterministic code reads every source document, fills issue-card templates with verbatim quotes, and renders a finished memorandum (document assembly, not writing) before the model takes a turn, and “bypass on/off” is simply whether that assembly step runs. The refine step is an extra pass bolted onto that rig that reworks the assembled memo before it is filed.

Figure 1 is the whole thing with nobody home. We ran each rig with the model deleted, verified at zero tokens in and out, and read the score.

## Harness floor on the clean data-privacy task, model removed

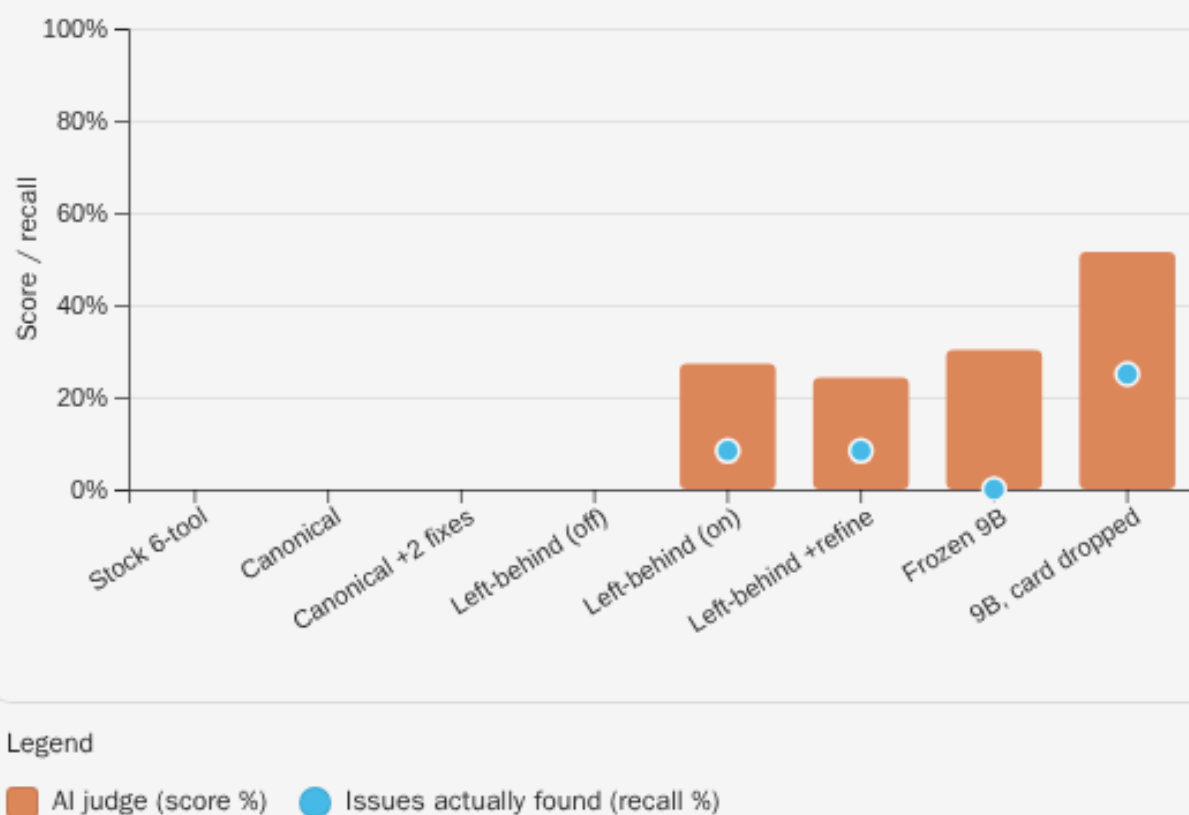


Figure 1 · Bars: the AI judge's score (%) for each rig run with the model deleted (0 tokens in/out). Dots: issues actually found (recall %), shown only for rigs that emit a memo. The two rightmost bars restore a real frozen small model on the canonical harness for comparison. Hover for per-rig detail. Data: `fig1_data.json`.

| Rig, model removed                                 | Memo produced?   | AI grader     | Issue-s |
|----------------------------------------------------|------------------|---------------|---------|
| Stock 6-tool loop                                  | none             | 0             | —       |
| Canonical harness (upstream)                       | none             | 0             | —       |
| Canonical + 2 minor fixes                          | none             | 0             | —       |
| The left-behind rig, bypass off                    | none             | 0             | —       |
| The left-behind rig, bypass on                     | 86,497-byte memo | 9/33 (27.3%)  | 0/15    |
| The left-behind rig + refine step                  | 86,597-byte memo | 8/33 (24.2%)  | 0/15    |
| — for comparison, with a real model —              |                  |               |         |
| A real frozen small model, canonical rig, 30 turns | real run         | 10/33 (30.3%) | 0/15    |
| The same model, one malformed input card dropped   | real run         | 17/33 (51.5%) | 3/15    |

\* *The dumb checklist credits one issue here, but it is a documented over-credit: the memo merely quotes a source sentence without ever flagging the problem. The true count is arguably zero. Throughout, “the left-behind rig” is our optimization experiment’s modified copy of the upstream runtime, left in the working directory; its behavior is a property of that specific modified artifact, never of the model, and never of the well-designed upstream harness it was forked from.*

Notice the top four rows. Remove the model, and the ordinary harnesses produce nothing at all: a blank, scored zero. That is the correct behavior, and it is worth pausing on. The canonical stack we build on [3], Nguyen and Reilly’s system, is a well-designed control whose finalize gate does exactly what a benchmark’s plumbing should do: when no model does any work, it emits no deliverable and scores zero. A benchmark that floors at zero is doing its job.

The left-behind rig is the exception, and it is the whole story.

(i) The empty rig is worth about 27%; a real small model adds about 3. This number did not exist before we measured it: what a legal benchmark scores with the model deleted entirely, read through the benchmark’s own AI judge. It comes to 27.3%. With the model gone, the left-behind rig emits its 86-kilobyte memo and the strict grader marks it there. Put a real, frozen small model back in and let it take thirty genuine turns through the *canonical* harness, and it scores 30.3%, a lift of +3.0 points over the empty rig. And here is the part that matters most: that model passes zero of the fifteen issue-spotting criteria, exactly like the empty rig. Its contribution is real but tiny, and it buys none of the thing the exam is actually about.

This is the midnight partner sign-off. The memo cites the right authorities and covers the required sections, so it gets signed — nobody pulled the cases. The score rewards the shape of diligence, and the shape is most of the score.

(ii) One formatting fix was worth seven times the model’s entire visible contribution. Nobody had ever put a price on tidiness alone, isolated from analysis; here it is. We changed *no analysis whatsoever*; we dropped a single malformed input card that had been jamming the pipeline. The same model’s score jumped from 30.3% to 51.5%, a +21.2-point swing, and for the first time, the memo actually spotted planted issues: recall went from 0 to 3, and three real issue-spotting criteria flipped to pass. A pure tidiness fix, authoring not one new sentence of legal analysis, moved the number roughly seven times as far as the model’s whole visible effort did.

Laid out as a ledger — an accounting across measured configurations, not a causal ladder on one rig — the points line up like this:

| Where the points come from                  | Value |
|---------------------------------------------|-------|
| The rig alone, no model                     | 27.3% |
| + the model actually running                | +3.0  |
| + one formatting fix (zero new analysis)    | +21.2 |
| = score reached                             | 51.5% |
| still on the table (real analysis, unfound) | 48.5  |

One caveat on reading it: these three numbers come from three different measured configurations — the bypass rig’s zero-model floor, a frozen small model on the *canonical* harness, and that same canonical run with one malformed card dropped — not from adding parts to a single rig. What stays constant across all three is the measurement: one task and one judge protocol throughout. With that caveat, of the 51.5 points reached at the top, 27.3 were there before any model existed, 3.0 came from the model reasoning, and 21.2 came from tidiness. The largest movable lever in the entire ledger is conformance, not capability. The 48.5 points still on the table, mostly issue-spotting, are the only part that measures the thing you are paying for.

One objection deserves a direct answer. *If the benchmark evaluates the whole agent system, then harness output is part of the system.* Correct. But then the score should not be read as model capability. The harness floor tells us how much of the end-to-end score exists before model reasoning enters.

Put a top model back, and the rig fights it

Figure 2 puts a strong frontier model into every rig and runs the full exam: six rigs, five matters, three attempts each, ninety runs in total. Both graders report, side by side.

### A strong frontier model across six rigs: judge score vs. issues actually found

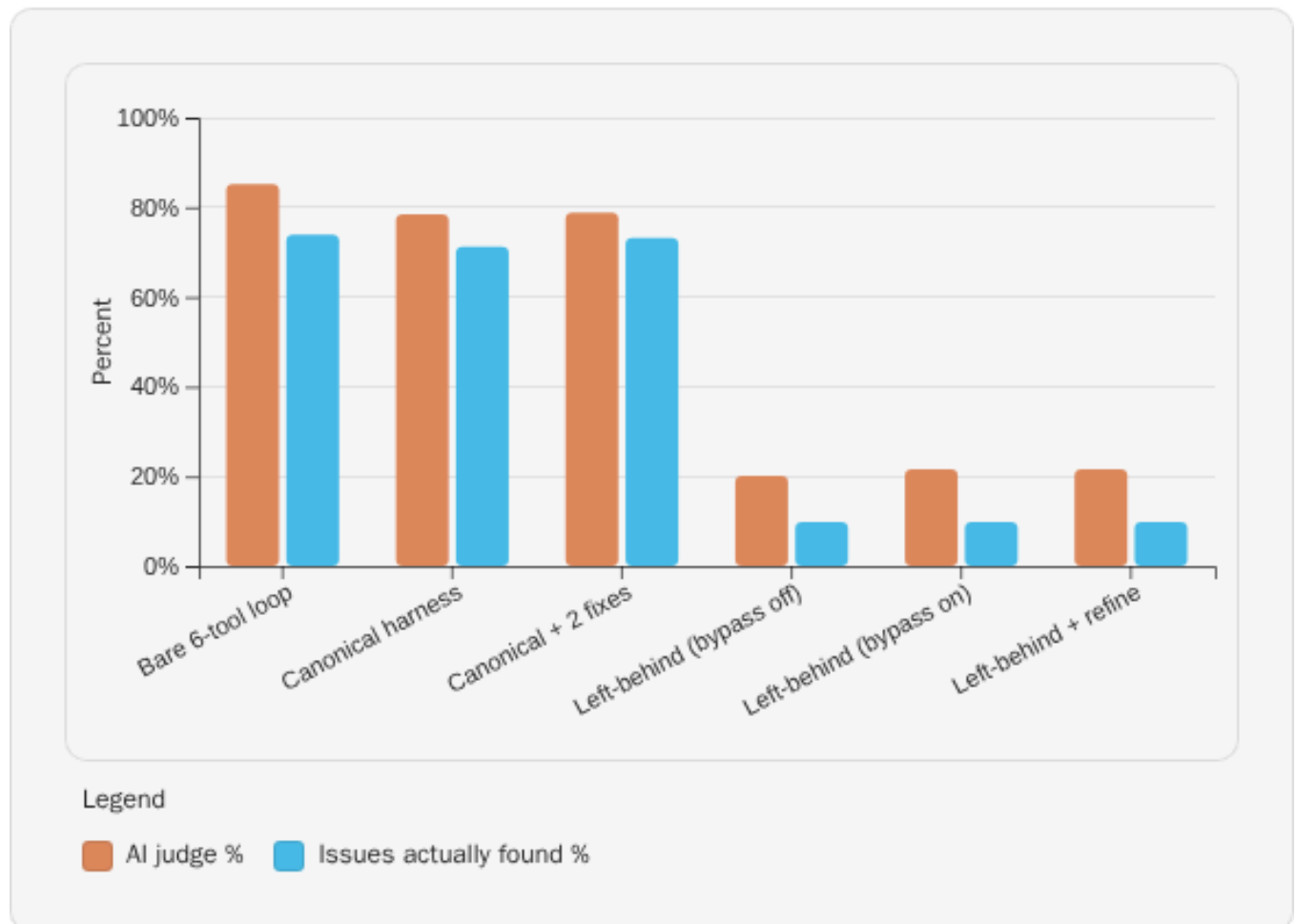


Figure 2 · Grouped bars per rig: the AI judge's score (%) beside issues actually found (recall %), aggregated over five matters and ninety runs. The distance between the two bars is the form-over-substance gap; note the cliff at the left-behind rigs. Hover for the gap and token cost. Data: `fig2_data.json`.

| Rig (with a strong model)         | AI grader | Issues actually found | Gap   |
|-----------------------------------|-----------|-----------------------|-------|
| Bare 6-tool loop                  | 85.2%     | 73.9%                 | +11.3 |
| Canonical harness                 | 78.4%     | 71.2%                 | +7.2  |
| Canonical + 2 minor fixes         | 78.8%     | 73.2%                 | +5.6  |
| The left-behind rig, bypass off   | 20.1%     | 9.8%                  | +10.3 |
| The left-behind rig, bypass on    | 21.6%     | 9.8%                  | +11.8 |
| The left-behind rig + refine step | 21.6%     | 9.8%                  | +11.8 |

(iii) The over-automated rig made a top model worse, and this time the loss is verifiable. The direction of this finding belongs to Nguyen and Reilly, whose own write-up reports that the same structured harness that dramatically lifts a small model can cost a frontier model points [\[3\]](#). What nobody could show without an answer key is what those lost points were made of, real capability or grader taste, and how far the effect can run. Our answer key settles both: the loss is real, and in the extreme case it is total. Through the plain harnesses, the strong model finds roughly 37 of the 51 planted issues (73.9% at the bare loop); drop the very same model into the left-behind rig, and it finds about 5 of 51 (9.8%), and it makes no difference whether the memo-writing bypass is switched on or off. The cause is mechanical: the rig hands the model a finished-looking deliverable and invites it to sign. Its recall through that rig collapses to *exactly* the score the rig gets with no model at all: the frontier model contributes nothing the scaffolding had not already faked. A heavyweight, over-eager harness can *subtract* capability, and now there is an instrument that shows when it does.

And note the plain rigs at the top: the bare six-tool loop, the canonical harness, and the canonical-plus-fixes are essentially flat (85 / 78 / 79 by the grader; 74 / 71 / 73 by the checklist). The elaborate scaffolding buys a strong model no extra issues over a bare loop, at nearly double the token cost. Scaffolding has to be matched to the model; past a point, more of it is just more surface for the illusion to live on.

## 6. Is the AI grader dumb?

---

No. And this matters, because the easy conclusion (“AI graders are useless”) is the wrong one.

The grader ranks correctly. Watch what happens at the cliff between the plain rigs and the left-behind rig: the grader’s score falls 59 points and the real recall falls 63 points, together, at the

same step. A grader that plunges when the substance plunges is not blind. As an *ordinal* instrument, telling you which memo beats which, it is serviceable.

In this study, the two failure modes were both about *level*, not *rank*.

The first is the floor: it never scores an empty-shaped-but-hollow deliverable at zero. The second is a consistent form-over-substance bonus: at every rung it sits between +5.6 and +11.8 points above the truth, six to twelve points of credit for looking right. Every one of those points comes from presence, citation, and structure criteria (“cites the framework,” “attaches section references,” “mentions the deadline”), never from a judgment criterion. It is grading the letterhead.

None of this is one vendor’s quirk. We re-graded the same memos with four independent AI graders from four different companies. They agreed to within two or three criteria on every rung, and all four inflate the hollow rig above its true recall, with the *weaker* graders inflating slightly more:

| Rig                        | Claude | MiniMax | DeepSeek | GLM |
|----------------------------|--------|---------|----------|-----|
| Left-behind rig, bypass on | 9      | 10      | 10       | 11  |
| Left-behind rig + refine   | 8      | 10      | 10       | 10  |
| Frozen small model         | 10     | 12      | 11       | 13  |
| + one card dropped         | 17     | 17      | 18       | 19  |

(criteria passed, out of 33)

Across the graders we tested, the floor and the bonus both recur, not as any one grader’s quirk. That is the good news and the bad news in one table: you can trust these graders to rank, but not to tell you *how much* of a high score is real.

Is this just hallucination? No, and the distinction is the whole point. Hallucination is a model fabricating content: a case that never existed, a quote nobody said. Here the opposite holds. Every quote in the 86-kilobyte memo is real and every citation accurate. It is the *Mata v. Avianca* fact pattern run in reverse. Where those lawyers filed confident argument resting on invented cases, our rig filed a binder of perfectly accurate photocopies behind a tidy cover page: nothing false anywhere in it, and no judgment anywhere in it either. So the failure is not in the generation; it is in the measurement. The grader awarded analysis-credit to the *appearance* of analysis. Different disease, different cure: for hallucination you verify the claims, but for this you measure substance independently of form, exactly what the planted answer key does. The one true hallucination in our whole dataset is smaller and quieter: the model’s single turn inside our

rig, confidently reporting that “the deliverables have been finalized and validated,” work it never did.

## 7. The checklist

---

Three tests, in order, that anyone running an agent benchmark can apply. Each maps to something a lawyer already does.

1. Measure the harness floor. Remove the model and watch. Run the whole rig with the model deleted and read the score. If the floor is high, most of your benchmark number was never about the model. *(The lawyer’s version: read the memo the rig produces with nobody home. Whatever it earns, the associate didn’t.)*
2. Plant ground truth you did not optimize against. Seed known issues into the matter, keep some tempting non-issues as traps, and score how many were actually spotted, mechanically, with no room for charm. *(The lawyer’s version: the law-school exam. Hide the issues; grade the spotting.)*
3. Decompose conformance before you claim capability. Before crediting any score gain to a smarter model, separate “a well-formed deliverable landed on the desk” from “the issues were actually found.” *(The lawyer’s version: distinguish the memo that was filed on time and looks right from the memo where somebody actually pulled the cases.)*

## 8. What we’re releasing, and what we’re not claiming

---

We are releasing the exam: five legal matters, 51 planted issues, the traps, and the mechanical checklist. As far as we know, it is the first legal document-review benchmark where the grade is “issues actually spotted,” not an AI’s opinion of the write-up. Anyone optimizing a legal agent can now watch the two graders disagree in real time, which is the entire point.

The honest limits are worth stating plainly. Five matters and one benchmark family is a start, not a field; the four new tasks are freshly authored and have not yet been stress-tested by outside parties. The rigs we probe are a *probe set*, not a step-by-step replay of an evolution; three of them share one underlying build, so no single row-to-row jump isolates one clean mechanism. The big grid runs a single frontier model on a single provider. And the cautionary tale in §2 is exactly that, a caution: the memo-writing dodge our loop drifted toward *lost* to the

ordinary baseline and was *caught* by the loop's own safeguards. It is evidence *for* audit-style guardrails, not that shortcuts win. We claim no legal-evaluation firsts; our contribution is the instruments and the answer key, offered as tools we wish we'd had. We are not claiming nobody knew harnesses matter. We are showing how to measure how much they matter, in a legal benchmark, with a model-deleted floor and a planted-issue answer key.

Nothing here is a verdict on harness optimization, on Harvey's benchmark, or on Niklaus's +16.7, a result we still admire and, on our own grid, reproduce. What is new is the instrument. The floor is now a number you can measure, conformance is a lever you can weigh before you credit capability, and the answer key is out in the open for anyone to grade against — three tests any lab can run, on any agent, the next time a score goes up and someone has to say which of the three things (capability, conformance, or appeasement) the number actually bought.

## Endnotes

---

1. Joel Niklaus, *Don't Train the Model, Evolve the Harness* — the frozen-weights harness-optimization result (+16.7 held-out) that inspired this work.
2. Harvey, *LAB* — the legal-agent benchmark and evaluation stack that is our substrate.
3. Duy Nguyen and Leon Reilly, `yudduy/larness-release` — the upstream “H\_legal” harness we build on; its finalize gate correctly floors at zero with no model.
4. *Darwin Gödel Machine* ([arXiv:2505.22954](#)) — a harness-evolution loop whose Appendix H documents a candidate scoring perfectly by deleting hallucination markers; the nearest prior art to our failure mode.
5. *ABC* ([arXiv:2507.02825](#)) — a do-nothing agent clears 38% of the  $\tau$ -bench agent benchmark, the canonical warning that trivial baselines are dangerous.
6. *AI Agents That Matter* ([arXiv:2407.01502](#)) — the case for measuring what the scaffolding alone earns before crediting the model.
7. Zheng et al. ([arXiv:2410.07137](#)) — constant, content-free outputs that nonetheless score against LLM-judged benchmarks; the spirit of our empty-rig control.
8. *CriticGPT* ([arXiv:2407.00215](#)) — planted bugs used to score a critic model; the planted-defect method applied to AI review.

9. *Meta-Harness* ([arXiv:2603.28052](#)) — recent scaffold-optimization results of the kind our audit kit is designed to measure.
10. *GEPA* ([arXiv:2507.19457](#)) — one of the optimization methods that makes harness evolution practical.
11. *CLAUSE* ([arXiv:2511.00340](#)) — planted-issue contract review, the nearest legal-evaluation neighbor to our answer key.
12. DeMillo, Lipton, and Sayward, *Mutation testing* (1978) — the 48-year-old idea of seeding known defects to score a detector, of which our planted legal issues are a direct descendant.

*A fully-cited [technical companion](#) with the complete related-work treatment, the measurement-chain audit, and per-criterion data accompanies this article.*

## Reproducibility note (frozen evidence)

---

Pre-registration is committed at `1a83ec5` (the data-privacy task's matchers, match criteria, and rubric) and `c525389` (the four new tasks, with a per-file sha256 fingerprint for every model-facing document in `FREEZE_MANIFEST.md`, and matcher amendments `cda98b7` / `316c976`, each made and re-verified before any measurement). The empty-rig evidence pack is at `ada30f7`; all figures run on the canonical evaluator checkout `c8a83ed`. Measured units: the empty-rig study spent \$0 on metered API calls (the policy model ran at 0 tokens in / 0 out, verified per transcript); the 90-run grid used a metered frontier lane ( $\approx 32.3\text{M}$  input /  $\approx 930\text{K}$  output tokens) and the four-grader panel used token-plan lanes; dollar figures for those are read from the provider console, never modeled here. Every rollout ran through the lab runner under a standing isolation rule, never through a coding assistant's own harness.

---

### Citation

For attribution in academic contexts, please cite this work as

Aaron Kelly (2026). "When the Benchmark Is Just a Rubber Stamp: How Scaffolding Alone Scored 27% on Legal Review".

BibTeX citation

```
@misc{kelly2026_when_the_benchmark_is_just_a_rubber_stamp_how_scaffolding_alone_scored_27_on_legal_review,  
  title={When the Benchmark Is Just a Rubber Stamp: How Scaffolding Alone Scored 27% on Legal Review},  
  author={Aaron Kelly},  
  year={2026},  
}
```

Reuse

Text and figures are licensed under [CC-BY 4.0](#) unless noted otherwise.

Made with  with [research article template](#)