Macrodeep Research

Keeping a Benchmark Honest — Contamination, Held-Out Splits, and Decay in GTM-Bench

Evaluation · Jul 2026 · 13 min read

Macrodeep Research Team

Abstract

Third in a series on GTM-Bench. The design note covered the taxonomy and scoring; the ORCHESTRATE deep-dive covered agentic grading. This piece covers the least glamorous and most important part of any benchmark: the machinery that keeps a number from meaning less than it claims. No model scores here — this is about trust.

Contributions

- Three distinct threats hide under “contamination”: overlap, distributional leakage, and decay
- A public development set for reproducibility, a private held-out split for authoritative ranking
- Contamination policy published as a first-class artifact — lineage, per-category status, held-out-only scoring
- Versioning, rotation, and canary probes to fight decay over time

The question that decides whether a benchmark is real

Every benchmark faces one question that matters more than its task design, its grading harness, or its leaderboard: can the reported number be trusted, or did the model already see the answers?

For a benchmark authored by a lab that also builds a model on the same domain, this is not a hypothetical — it is the central risk. GTM-Bench draws its instances from real go-to-market operations, and a GTM model may be trained on data from the same world. If the training set and the test set overlap, a high score measures memorization, not capability — and a benchmark that measures memorization is worse than no benchmark, because it launders a null result into a marketing chart. This piece is about the machinery built to prevent that. It is deliberately the least exciting article in the series. Rigor usually is.

Three distinct threats, often conflated

“Contamination” gets used as a catch-all, but three separate failures hide under the word, and each needs its own defense.

Train/test overlap. The model was trained on data that contains, or closely paraphrases, specific test

instances or their answers. The classic contamination case and the most damaging. Distributional leakage. No specific instance leaked, but the model was trained on so much data from the same source that it has effectively internalized the answer distribution — it can reconstruct the oracle from priors rather than from capability. Benchmark decay. Nothing leaked at construction time, but the benchmark is public and static, so over months the field optimizes toward it. The number stays the same; what it measures erodes.

A serious methodology addresses all three. Defending against only the first is the common mistake.

Defense one: held-out construction and a private authoritative split

GTM-Bench is split into two sets with different roles. A public development set is released openly — instances, harness, graders, everything. Its purpose is reproducibility: anyone can run their own model, inspect the grading, and verify the mechanics. It is also, by definition, contaminable — once public, it can enter training corpora, and scores on it should be read as directional.

A private held-out set is the authoritative leaderboard split. It is never published, drawn from the same distribution and construction pipeline as the public set, and used to produce the numbers that actually rank models. Because it is unpublished, it cannot enter a training corpus by osmosis, and a model cannot have memorized instances it has never seen. A score is only reported as authoritative if it comes from the held-out split, run under a controlled submission process.

Defense two: contamination policy as a disclosed, first-class artifact

Held-out splits handle instances a model couldn't have seen. They do not, by themselves, handle the case where the benchmark author is also the model author and the two share a data lineage. That case requires an explicit, disclosed policy — not a reassurance. GTM-Bench's policy has three parts, and all three are published alongside the results rather than kept internal.

Lineage disclosure. For any model the benchmark author also built, the overlap between the model's training sources and the benchmark's instance sources is stated plainly — the actual relationship, so a reader can judge it. Per-category contamination status. Contamination risk is not uniform across the taxonomy; each category carries a status flag, and any category where overlap cannot be ruled out is reported with that caveat attached, not folded into a clean headline number. Held-out-only authoritative scoring. The ranking number comes exclusively from the private split, and for the author's own model, from held-out instances constructed after the training cutoff wherever possible.

A contamination policy that lives in a drawer is worthless. The only version that builds trust is one published next to the scores, specific enough that a skeptic can check it.

Defense three: fighting decay with versioning and rotation

Even a clean benchmark rots if it is static and public. The mitigations are structural, not one-time. Versioned releases — every reported score names the version it was run against; a number without a version is uninterpretable. Rotating instances — new instances are added and stale ones retired on a cadence, so the target a model would optimize toward is moving. A refreshed private split per authoritative run — the held-out set is periodically reconstructed from new operational data, so it postdates prior training runs by construction. Contamination probes — before an authoritative run, canary checks look for near-verbatim reconstruction of oracle values or suspiciously low variance on hard instances; a model that trips the probes is flagged, not silently ranked.

What this buys, and what it doesn't

Stated plainly, so the claims stay bounded: these defenses make a reported GTM-Bench number mean what it says — a model's capability on held-out, version-pinned instances it did not train on, with the author's own data lineage disclosed and per-category contamination risk visible. That is a strong, checkable claim.

They do not make the benchmark incorruptible. No public benchmark is. A determined actor can still overfit to the public distribution; a static set still decays between rotations; distributional leakage is hard to fully rule out even with held-out splits. The honest position is that these mechanisms raise the cost of gaming above the cost of just being good at the task — which is the most any benchmark can actually promise. A benchmark that claims to be uncontaminable is lying; one that publishes its defenses and their limits is doing the real thing.

Why this article exists

A benchmark's credibility does not come from its task design being clever. It comes from the reader being able to answer, for themselves, “why should I believe this number?” For a lab-authored benchmark in a domain the lab also models, that question is sharper, not softer — and hand-waving it is exactly what turns a research artifact into a marketing prop.

So the methodology is published before the leaderboard, on purpose. When the numbers come, they will arrive against a documented contamination policy, a versioned held-out split, and a disclosed data lineage — so that the first question a skeptic asks has an answer already written down. A benchmark earns trust in the order it releases things, and rigor comes first.

References

1. Jimenez et al. SWE-bench: Can Language Models Resolve Real-World GitHub Issues? 2024.

2. Rein et al. GPQA: A Graduate-Level Google-Proof Q&A Benchmark. 2023.

3. Zhou et al. Don't Make Your LLM an Evaluation Benchmark Cheater. 2023.