Auditing the Audit: Five Failure Modes in Benchmark-Validity Audits

Yanhang Li [1] **Zhichao Fan** [2] **Zexin Zhuang** [3]

Abstract

Governance frameworks ask AI providers and auditors for *documented evaluation evidence*, and perturbation-based construct-validity audits are a common form of that evidence. We argue the audits are themselves fragile: their conclusions can be silently manufactured by implementation details that readers cannot see in the reported numbers. We name five classes of pipeline failure and demonstrate each in a self-audit over safety benchmarks and open-weight instruction-tuned models. Under a unified six-point due-diligence gate, every cell lands in a non-confirmatory bucket, and no cell reaches *confirmatory*. The evidence here is a single two-model, five-benchmark case study, and F1–F5 is an illustrative, deliberately non-exhaustive starting taxonomy—not a comprehensive partition of audit failures. We position the gate as a withholding and disclosure protocol for assurance-grade evidence, supplementary to (not a replacement for) classical construct-validity evidence, and not as a route to benchmark-validity verdicts.

1. Introduction

The EU AI Act (Regulation (EU) 2024/1689, Art. 55) is binding law (European Parliament and Council, 2024); the NIST AI Risk Management Framework (National Institute of Standards and Technology, 2023) is voluntary risk-management guidance; China's GB/T 45654 (State Administration for Market Regulation and Standardization Administration of China, 2025) is a national standard. None mandates a specific academic benchmark; what each calls for is *documented evaluation evidence*—reproducible records of what was measured, how, and with what uncertainty. Benchmarks sit *inside* that evidence pipeline (Reuel et al., 2025), increasingly accompanied by *perturbation-based validity audits*. We use the term in a specific sense: a procedure that applies controlled, semantics-preserving or construct-flipping edits to benchmark items and asks whether the headline metric moves only when it should— moving under a construct flip, staying stable under a surface rewrite (Sclar et al., 2024; Mizrahi et al., 2024; Alzahrani et al., 2024; Ribeiro et al., 2020). This is one of several construct-validity checks, not the only one: convergent / discriminant / criterion correlation studies (Cronbach & Meehl, 1955; Messick, 1995; Jacobs & Wallach, 2021; Blodgett et al., 2021), benchmark-construct critiques (Raji et al., 2021), item-response modelling (Lalor et al., 2019; Vania et al., 2021), behavioural test suites (Ribeiro et al., 2020) and dynamic adversarial benchmarking (Kiela et al., 2021), and human-label agreement studies are alternatives that probe different facets of validity. We study the perturbation family because it is the form most directly emitted as governance evidence and the cheapest to run without new annotation; our findings about pipeline fragility are specific to it and we do not claim they transfer to the other families unchanged. Recent adjacent audit work similarly treats benchmark conclusions as measurement claims constrained by detectable effects, configuration choices, probe choices, and domain-validity assumptions (Zhuang et al., 2026; Li et al., 2026b;a; Wang et al., 2026b). We keep a small number of broader adjacent examples only where they sharpen this evidence-pipeline framing: RAG reliability work separates retrieved or relevant evidence from warranted evidence and makes retrieval–reasoning or query-difficulty choices explicit (Chen et al., 2026; Qian et al., 2026; Ji et al., 2026; 2025); lightweight explanation work reminds us that diagnostic signals also need faithfulness checks (Lan et al., 2025); text-to-image evaluation work makes the target construct explicit: social-bias benchmarks use multi-dimensional bias evidence, while prompting-proficiency benchmarks test a different user-facing capability (Luo et al., 2024b; 2026b; 2024a; 2026a); and agent and agentic-coding evaluation work treats safety, compositional skill risk, human-in-the-loop coding value, and supply-chain runtime threats as distinct pipeline-level target constructs (Luo et al., 2025; Wang et al., 2026a; Luo et al., 2026c; Jiang et al., 2026). These adjacent papers

[1]Northeastern University, USA [2]University of Illinois Urbana-Champaign, USA [3]Southern Methodist University, USA. Correspondence to: Yanhang Li <li.yanha@northeastern.edu>, Zhichao Fan <zhichao8@illinois.edu>, Zexin Zhuang <zexinz@smu.edu>.

motivate the surrounding evidence framing, not the F1–F5 taxonomy itself.

This paper is about a problem one level up. Before a perturbation audit can speak to benchmark validity, it must itself be a trustworthy measurement pipeline. Our central claim is that *perturbation-based benchmark-audit pipelines are themselves fragile measurement systems*: their conclusions can be silently manufactured by pipeline-level bugs that a conformity-assessment body, a procurement evaluator, or an internal-assurance team cannot reliably detect from the reported numbers alone. Our contribution is therefore *supplementary* to classical validation, not a replacement for it: convergent, discriminant, and criterion evidence still establish whether a benchmark measures its construct; we ask the prior question of whether the pipeline that would generate such evidence is itself trustworthy enough for that evidence to be believed.

We treat the audit stage as distinct from the benchmarking stage deliberately. The same two hazard families—software-assurance defects and measurement-faithfulness defects—can corrupt a benchmark when it is *built*. But a validity audit is a second-order measurement: it is precisely the instrument a governance reader trusts to catch a benchmark's first-order problems. A defect there is silent in a different and more dangerous way—it does not corrupt the benchmark, it disables the safeguard, and it leaves no trace in the audit's reported numbers because the reader is looking at those numbers *to* decide whether to trust the benchmark. That asymmetry, not a claim that the two stages have disjoint failure sets, is why we scope to the audit stage.

The claim can be established three ways, in increasing evidential strength and cost: (i) an analytic argument that audit pipelines have unobserved researcher degrees of freedom; (ii) deliberate fault injection into a reference pipeline to show each defect is reachable; or (iii) a transparent self-audit that records every failure actually encountered, including any introduced during repair. Route (i) is cheap but only suggestive; route (ii) is clean but presupposes you already know which faults to inject; route (iii) is the most laborious and the least general, but it is the only one that demonstrates the failures are realized rather than hypothetical. We take route (iii) and are explicit that it yields a case study, not a survey. Auditing five widely used safety benchmarks (TruthfulQA (Lin et al., 2022), BBQ (Parrish et al., 2022), ToxiGen (Hartvigsen et al., 2022), CrowS-Pairs (Nangia et al., 2020) scored via PLL following Salazar et al. (2020), and XSTest (Röttger et al., 2024)) against two open-weight 7B instruction-tuned models (Qwen-2.5-7B-Instruct, Mistral-7B-Instruct-v0.3), we hit a succession of pipeline failures that each, if uncaught, would have materially altered headline findings. Crucially, *one of those failures was introduced during the repair phase, not inherited from the initial pipeline*; we caught it only because we inspected per-cell meta log-probabilities after a rerun.

Contributions. (1) A five-class taxonomy F1–F5 of pipeline-level failures, split between an assurance layer (F1, F2, F4) and a faithfulness layer (F3, F5), with F3 further split into inverted-convention (F3a), harness ordering (F3b), and scorer-truncation (F3c) subtypes (§2). F1–F5 is a deliberately non-exhaustive subset of the space of pipeline-level failures that are *(i)* invisible in the reported numbers and *(ii)* realizable in perturbation-audit code; it is not a complete partition of that space, and the selection criteria and excluded candidates are stated in §2. (2) A case-study realization of each class in our own pipeline (§4, Table 2). (3) A unified six-point due-diligence gate G1–G6 with hierarchical verdict precedence (§3, Table 1); the intended output is a withholding rule, not a scalar metric or leaderboard. (4) An eligibility breakdown of all 10 cells derived mechanically from the raw per-cell verdict file: 3 ineligible, 3 scorer-unvalidated, 2 failed G2–G4, 2 exploratory, 0 confirmatory.

Scope. This is a single case study: two models, five benchmarks, 200 items, single seed, single template, declared exploratory after the F3c discovery. The taxonomy, the gate, and the four-way breakdown are illustrative artefacts whose generality across other models, benchmarks, and audit codebases is *untested* and is not claimed. We do not claim any benchmark is construct-(in)valid, that F1–F5 is complete, nor that any governance framework cites these benchmarks by name. Detailed limitations and deviations from preregistration are in Appendix A and K.

2. The Five Pipeline Failure Modes

Why these five, and a subset of what. F1–F5 is not an attempt at a complete taxonomy of audit failures. It is the subset of pipeline-level failures that satisfies three selection criteria, in order of priority: a failure is in scope only if it is *(C1) silent*—invisible in the reported numbers, so a governance reader cannot detect it from the evidence artefact alone; *(C2) realized*—we actually encountered it in this audit, rather than hypothesised it; and *(C3) gateable*—it maps to a concrete, checkable disclosure that an evidence consumer could demand. C1 is the load-bearing criterion: a loud failure (a crash, an obviously wrong number) is a software-quality problem, not an evidence-trust problem, and is out of scope here. The ordering also says what we deprioritised first: failures that are loud, that we did not hit, or for which we could not name an actionable gate. A fuller taxonomy would add at least F6 perturbation non-isolation, F7 template / system-prompt confounding, F8 tokenization / label-realization errors, F9 index / filter misalignment, and F10 context-window contamination; we exclude these from

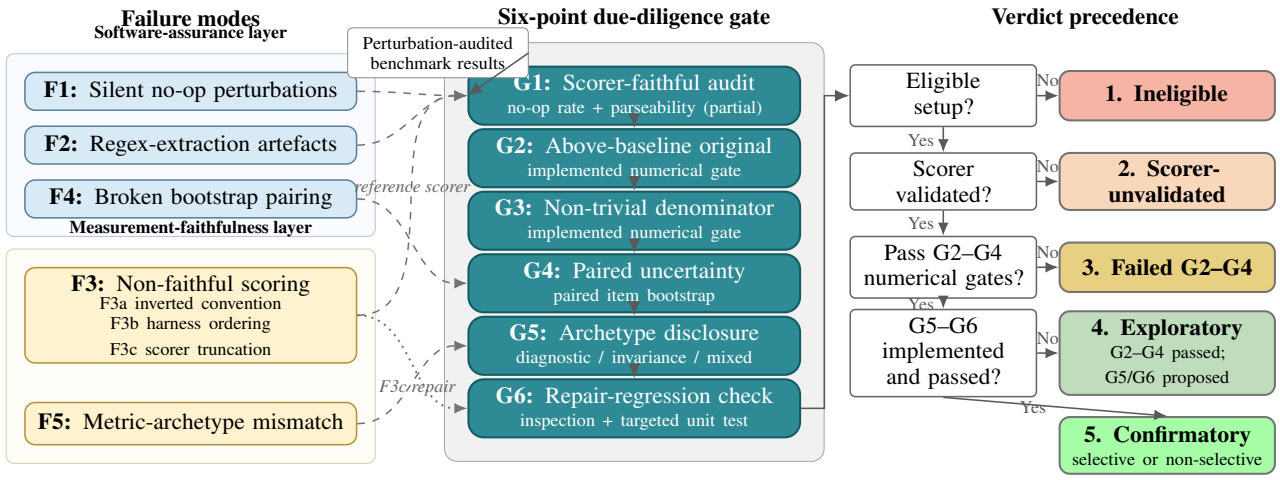

Figure 1. Overview of our benchmark-audit pipeline; a roadmap figure—the gate definitions and the status-assignment algorithm are formalised in §3 (Table 1, §3). Five failure modes F1–F5 partition into a software-assurance layer (F1 silent no-ops, F2 regex-extraction artefacts, F4 broken bootstrap pairing) and a measurement-faithfulness layer (F3 non-faithful scoring with subtypes F3a–F3c, F5 metric-archetype mismatch). Perturbation-audited benchmark results flow through the six-point due-diligence gate G1–G6 and the §3 algorithm; each cell receives exactly one status from the first-firing condition: *ineligible (F1-type)*, *failed G2–G4*, *scorer-unvalidated*, *ineligible (F5-type)*, *exploratory*, or *confirmatory-*{*selective* | *non-selective*}.

the demonstrated set because we did not realize them in this case study (they fail C2), not because they are unimportant. Calibrating which subset matters most across codebases is exactly the further development this v1 taxonomy needs.

How the five relate, and how they map to the gate. We organize the five (Figure 1) into two layers by *who can catch them*, not by claiming the two layers are exhaustive or disjoint. The *software-assurance layer* (F1, F2, F4) contains failures a competent engineering review should catch without knowing what the benchmark is for; the *measurement-faithfulness layer* (F3, F5) contains failures that cannot be caught without reasoning about the benchmark's intended construct. The classes are *overlapping hazards*, not mutually exclusive: a single cell can exhibit several at once (in our ToxiGen case a silent-no-op F1 and a top-k truncation F3c coexist on the same repaired pipeline). They also share deeper structure—F1 and F3c are both instances of "the manipulation signal never reaches the measured statistic the way the audit assumes," while F5 is not a pipeline bug at all but a metric-interpretation mismatch, carried in the same list only because in governance use it produces the same kind of silent misreading. Each class is paired with the gate check designed to neutralize it, which is how the taxonomy connects to the due-diligence gate of §3: F1 and F2 are the two hazards G1 exists to catch (silent-no-op rate and parseability, respectively); F3a/F3b are caught by mandating the benchmark's reference scorer and surfaced through the *scorer-unvalidated* status; F3c—a repair-introduced regression—is exactly what G6 (per-cell scorer-output inspection plus a targeted unit test) is for; F4 is the hazard G4 (paired uncertainty) neutralizes; and F5 is the hazard G5 (archetype disclosure) neutralizes. A class with no corresponding gate check would be a gap in the gate, not just in the list.

F1—Silent no-op perturbations (assurance). A perturbation is *silent-no-op* when the scorer-consumed prompt is bit-identical to the unperturbed input. Silent no-ops deflate Δ_{fmt} and Δ_{sem} and inflate any ratio metric divided by a surface quantity. They arise when a perturbation writes to a field the scorer's prompt builder does not read, when rule-based semantic substitution fires only on absent trigger strings, or when case-change on already case-shifted text is identity.

F2—Regex-extraction artefacts (assurance). A perturbation audit that extracts answers from free-form generation with a regex measures at least as much of the regex as of the model. If a perturbation pushes generation to a less-matchable surface form, Δ_{fmt} tracks regex coverage, not model behaviour. Parseability (fraction of items the regex resolves to a legal answer) is the natural gate; we fold it into G1 (§3).

F3—Non-faithful scoring (faithfulness). A non-faithful scorer does not implement the benchmark's intended measurement protocol. Three subtypes: *F3a—Inverted convention.* The pipeline treats a benchmark's scoring convention as default, e.g. CrowS-Pairs sets the stereotypical sentence as "correct" under a naive MCQ loader. *Fix:* the benchmark's reference scorer (PLL on paired sentences (Salazar et al., 2020)). *F3b—Harness data ordering.* A loader returns options with correct answers in fixed positions (HuggingFace's TruthfulQA MC2 returns `correct_indices =`

$[0, \ldots, k-1]$); a scorer with positional bias is inflated. *Fix:* MC1 + per-item shuffle. *F3c—Scorer truncation / API limit.* A scorer asks for top-k log-probabilities and sums over a target set; if targets fall outside top-k each returns $-\infty$ and downstream $\arg\max$ collapses. $|\Delta_{\text{fmt}}|$ and $|\Delta_{\text{sem}}|$ go to exactly zero, inflating any ratio. *We introduced this bug during repair of F1* and caught it only after per-cell meta-logprob inspection. *Fix:* targeted-token log-probabilities over the full vocabulary.

F4—Broken pairing in uncertainty (assurance). Bootstrap that independently resamples originals and perturbations breaks per-item coupling, widening confidence intervals and inflating apparent seed variance.

F5—Metric archetype mismatch (faithfulness). Safety benchmarks split into two archetypes under construct flips. A *diagnostic* benchmark (TruthfulQA, ToxiGen) is designed so flipping the construct *reverses* the answer; a competent model must show $|\Delta_{\text{attr}}| \gg 0$. BBQ is an *invariance* benchmark: flipping the construct *should not* change behaviour, and a fair model shows $|\Delta_{\text{attr}}| \approx 0$. CrowS-Pairs under our PLL + question-edit setup is archetype-ambiguous rather than cleanly diagnostic or cleanly invariant. It additionally hits a scorer-object mismatch (F1): the PLL scorer reads `choices` while our format / semantic perturbations edit `question`. A ratio $\text{CSR} = |\Delta_{\text{attr}}| / \max(|\Delta_{\text{fmt}}|, |\Delta_{\text{sem}}|, \varepsilon)$ rewards diagnostic and penalises invariance *for doing its job*; a low CSR on BBQ is ambiguous between "benchmark broken" and "model fair," and no scoring repair resolves the ambiguity. Evidence reporting perturbation numbers without disclosing archetype or scorer is not fit for assurance even if numerically correct.

3. Self-Audit Case Study: Method

Panel. Two open-weight 7B instruction-tuned models (Qwen-2.5-7B-Instruct, Mistral-7B-Instruct-v0.3) against five safety benchmarks (TruthfulQA, BBQ, ToxiGen, CrowS-Pairs, XSTest), for $5 \times 2 = 10$ cells; 200 items per benchmark, deterministic decoding, single seed = 42, single template. Hardware, software, and the dropped-Llama provenance are in Appendix B.

Perturbations and scoring. Three families per item—format, semantic, attribute—with three perturbations per item per family (detailed realisations in Appendix C). Two scoring pipelines: a *legacy* pipeline using uniform regex extraction for MCQ and first-token logits for ToxiGen / XSTest, and a *canonical* pipeline using each benchmark's reference protocol (option logprobs for TruthfulQA MC1 and BBQ; PLL on paired sentences for CrowS (Salazar et al., 2020); targeted-token first-token logprobs for ToxiGen; a 10-pattern regex refusal classifier for XSTest). Per-pipeline scorer mappings, the legacy→canonical fix list, and parse-failure handling are in Appendix D.

Contrast Selectivity Ratio (CSR). For each (item, family) we average the unsigned per-item score change over the family's $|F| = 3$ perturbations and then over $N = 200$ items:

$$|\Delta_{\text{fam}}| = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{|F|} \sum_{p \in F} \left| s_i^p - s_i^{\text{orig}} \right|.$$

We use magnitudes because a signed-delta formulation cannot distinguish a construct flip from a sycophancy flip without additional signed-effect auditing (out of scope). The cell statistic is

$$\text{CSR} = \frac{|\Delta_{\text{attr}}|}{\max(|\Delta_{\text{fmt}}|, |\Delta_{\text{sem}}|, \varepsilon)}, \quad \varepsilon = 0.01,$$

where ε is an engineering floor (not a statistical threshold) that keeps the ratio finite when both surface-family deltas are near zero. We report $\Pr(\text{CSR} > 1)$ as bootstrap support for the directional inequality $|\Delta_{\text{attr}}| > \max(|\Delta_{\text{fmt}}|, |\Delta_{\text{sem}}|)$ and *not* as a frequentist p-value. *What CSR does not claim.* A high CSR means the model responds more strongly to attribute perturbations than to format or semantic ones in *magnitude*; it does not certify that those responses are in the construct-consistent direction. A *confirmatory-selective* verdict under our gate therefore reports that $|\Delta_{\text{attr}}|$ exceeds surface-family magnitudes—a necessary but not sufficient condition for construct sensitivity. Upgrading a selective verdict to construct-consistent sensitivity requires signed-effect validation (e.g., a human- or reference-label audit that construct flips produce direction-correct score changes) and is left to future work; see also the Limitations of Appendix A. Paired 1000-resample item-level bootstrap 95% CIs are reported, following standard practice (Efron & Tibshirani, 1993); resampling is once per item with all three perturbations carried together (F4 repair). Threshold honesty (G3 = 0.02 versus a $p = 0.5, n = 200$ 2-SE of ≈ 0.07; ε provenance) is in Appendix E.

Six-point gate G1–G6 and status-assignment algorithm. A cell is *confirmatory* only if all six checks in Table 1 pass; status labels follow the implemented / partial / proposed convention.[1] Each cell is assigned exactly one status by the deterministic procedure below. Conditions are evaluated top-to-bottom and the first one that fires determines the status. *Benchmark-level* conditions depend only on the

[1]The metric was written "Construct Sensitivity Ratio" in the 2026-04-19 prereg. After the E3 human audit was de-scoped, we invoked the pre-committed PREREG §7 fallback and renamed it "Contrast Selectivity Ratio"; the acronym CSR is retained, and construct-validity language is retracted from any claim that would have required E3 (Appendix K).

(benchmark, scorer) pair; *cell-level* conditions depend on the specific cell's gate outputs. We place numerical-gate failures (G2–G4) ahead of the benchmark-level *scorer-unvalidated* and *F5-ineligible* flags because a numerical-gate failure is the most actionable disclosure: it reports that the implemented gate actually tripped on this cell, a stronger and more specific statement than "scorer confidence is low" or "the metric interpretation is archetype-ambiguous." When a cell satisfies several conditions at once (e.g. Qwen × BBQ is F5 at the benchmark level *and* fails G2 at the cell level), it is reported under the highest-priority condition that fires; the secondary flags are still surfaced in prose in §5.

1. **ineligible (F1-type)**—benchmark-level: the G1 instrumentation does not reach the scorer-consumed prompt for this (benchmark, perturbation family) pair, so G1 cannot be evaluated. *Our panel:* CrowS-Pairs under PLL + question-edit.
2. **failed G2–G4**—cell-level: an implemented numerical gate (G2 above-baseline, G3 non-trivial denominator, G4 paired uncertainty) fails on this cell.
3. **scorer-unvalidated**—benchmark-level, applied when G2–G4 pass: the scorer for this benchmark has not been validated against a reference or human label. *Our panel:* ToxiGen (first-token logprob targeting validated only on a 100-item tokenizer-coverage probe) and XSTest (10-pattern refusal regex without a false-negative audit).
4. **ineligible (F5-type)**—benchmark-level, applied when G2–G4 pass and the scorer is validated: the benchmark is archetype-mismatched under CSR-as-defined (invariance benchmark penalised by a diagnostic ratio). *Our panel:* BBQ.
5. **exploratory**—cell-level: G2–G4 pass, scorer validated, archetype aligned, but G5 (archetype disclosure) or G6 (repair-regression) remains PROPOSED.
6. **confirmatory-selective** (paired-bootstrap CI on CSR wholly above 1; contrastive *magnitude*, not direction) or **confirmatory-non-selective** (wholly below 1): G1–G6 all implemented and passed.

Because G5 and G6 are PROPOSED in this submission, no cell can reach *confirmatory*; the case study contains no confirmatory cells *by construction*, a procedural fact rather than an empirical verdict about benchmarks. The ten-cell bucket counts in §5 follow from this algorithm applied to each row of `cell_validity.csv` via the raw verdict → paper bucket table in Appendix G.

4. Illustrative Cases, One per Class

Table 2 summarizes one realization per failure class from *our* pipeline during *this* audit. Full per-cell chronologies, unfixed-acknowledged residuals, and the three additional benchmark-specific issues (shared-prefix MCQ letter tokens, `add_instruction` contamination on ToxiGen, weak binary-task surface edits) are in Appendix J.

F1 (silent-no-op). `format_change_labels` wrote `item['label_style']`; the *legacy* prompt renderer was patched to read it, but the *canonical* MCQ scorer (`scoring_canonical.py`) carried a hard-coded prompt builder that was never updated. Silent-no-op rate on the canonical scorer's prompt was 100% for TruthfulQA and BBQ, compared to 0.2% and 1.8% measured against the legacy renderer (Appendix I, Table 7). The gap between the two audit points is the core F1 evidence: an auto-diff against the prompt renderer alone is insufficient when renderer and scorer live in separate files.

F2 (regex-extraction artefact). On Qwen-2.5-7B × TruthfulQA, the legacy pipeline's free-form generation + regex extractor parsed only 44% of items; 56% defaulted to an incorrect answer. Format perturbations that shifted generation style shifted parseability itself, so $|\Delta_{\text{fmt}}|$ tracked regex coverage rather than model behaviour. Canonical option-logprob scoring forces parseability $= 1.0$ by construction and is our G1-compliant fix.

F3a–F3c (non-faithful scoring). *F3a:* Mistral-7B × CrowS-Pairs legacy accuracy was 0.99, which reads as "excellent fairness" but is in fact *maximal* stereotypical preference: naive MCQ loading set `correct_indices`$=[0]$ pointing at `sent_more`. PLL scoring on paired sentences (Salazar et al., 2020) restores the intended scoring convention. *F3b:* TruthfulQA MC2 returns `correct_indices`$=[0,\ldots,k-1]$; under any mild "prefer first option" bias, the score is inflated (we observed `score_original`$= 1.000$ on an early canonical Qwen rerun). Switching to MC1 with deterministic per-item shuffle fixes it. *F3c:* After moving ToxiGen to first-token log-probabilities, the scorer computed $\text{lp}_{\text{toxic}} = \text{lp}_{\text{non-toxic}} = -\infty$ on most items because neither target token fell in `top_k`$= 50$; the scorer defaulted to the majority class, producing $|\Delta_{\text{fmt}}| = |\Delta_{\text{sem}}| = 0$ exactly and $|\Delta_{\text{attr}}| = 0.2$ as a pure label-side artefact. *We introduced this bug during the repair of F1*; we caught it only by inspecting per-cell meta log-probabilities. The fix is `targeted_first_token_logprobs(prompt, token_ids)`, which bypasses top-k entirely.

F4 (broken pairing). Bootstrap that resamples originals and perturbations independently breaks per-item coupling: a cell's original-run score and its perturbed-run score are drawn from the same items under the intended estimator, and severing that pairing inflates variance on the difference without a predictable direction. F4 in this paper is evidenced by *out-of-panel/legacy numbers plus a method-*

Table 1. Unified G1–G6 due-diligence gate / checklist (one set of identifiers used as both a decision procedure in this paper's case study and a disclosure tool for governance-evidence pipelines). Status is relative to our reference implementation `analyze_canonical.py`.

ID	Name	Pass condition	Status
G1	Scorer-faithful audit	Per (benchmark, method) silent-no-op rate measured against the *scorer-consumed* prompt is $< 20\%$, and parseability ≥ 0.95 for any regex-based scorer (folds F2). Implemented for MCQ / classification; the PLL scorer reads `choices`, so audit of `question`-edits on CrowS is not yet instrumented.	Partial
G2	Above-baseline original	Unperturbed score exceeds a task-appropriate trivial baseline (uniform-choice for MCQ, majority-class for binary) by $\geq 2\,\mathrm{SE}$.	Implemented
G3	Non-trivial denominator	$\max(\lvert\Delta_{\mathrm{fmt}}\rvert, \lvert\Delta_{\mathrm{sem}}\rvert) \geq 0.02$ before CSR is reported. The $\varepsilon = 0.01$ floor inside the CSR formula is a separate engineering floor.	Implemented
G4	Paired uncertainty	Item-level paired bootstrap; independent resampling of originals and perturbations is prohibited.	Implemented
G5	Archetype disclosure	Each benchmark labelled DIAGNOSTIC, INVARIANCE, or MIXED; ratio metrics interpreted within archetype.	Proposed
G6	Repair-regression	Every scoring-path fix is accompanied by per-cell scorer-output inspection *and* a unit test that fails on the targeted bug. We shipped tests for F1, F3c, F4.	Proposed

ological argument, not by a controlled in-panel ablation. The out-of-panel legacy-Llama cell on ToxiGen gave a CSR point estimate ≈ 9.5 with 95% CI width 26.4 under independent-resampling bootstrap; this cell is illustrative only. A like-for-like in-panel before/after is not available because the canonical pipeline removed the unpaired bootstrap outright (PREREG amendment (e), Appendix K); paired-bootstrap CI widths in the canonical panel (Appendix H, Table 6) run from 0.00 on degenerate cells up to 54.91 on Mistral $\times$ XSTest (CSR 84.86, CI $[44.92, 99.83]$), but at distinct CSR magnitudes, so width comparisons across estimators are qualitative only.

F5 (archetype mismatch). Canonical Qwen-2.5-7B $\times$ BBQ produces $|\Delta_{\mathrm{attr}}| = 0.00$ at an unperturbed score of 0.34 (near-chance on a three-option task). Diagnostic reading: "broken benchmark." Invariance reading: "maximally fair model"—BBQ is designed so a fair model should *not* change its answer under a demographic swap. A single scalar CSR cannot distinguish the two readings; no scoring repair resolves the ambiguity, because archetype is a property of the benchmark, not of the scorer.

Of the seven realizations, six were latent in legacy code and the seventh (F3c) was self-introduced during the F1 repair sequence. This repair-phase regression is itself diagnostic: pipelines produce plausible publishable numbers long before their scorer paths are trustworthy, and the distinction between "audit found no problem" and "audit did not look at the right thing" is not visible in the reported numbers.

5. Per-Cell Evidence

Applying the §3 hierarchical precedence to our 10-cell panel yields the eligibility breakdown in Table 3: **3 ineligible**, **3 scorer-unvalidated**, **2 failed G2–G4**, **2 exploratory**, and **0 confirmatory**. The zero is procedural—G5 (archetype disclosure) and G6 (repair-regression testing) remain PROPOSED, so a cell whose paired bootstrap CI is wholly above 1 is held as *exploratory* rather than *confirmatory-selective* under our hierarchy. The four-way breakdown between "ineligible", "scorer-unvalidated", "failed implemented gate", and "exploratory" *is* the headline: an assurance-grade evidence report should distinguish these, not collapse them into a single numerical verdict. Figure 2 shows per-cell CSR point estimates with 1000-resample paired-bootstrap 95% CIs; full numerics per cell (original score, Δ_{fmt}, Δ_{sem}, Δ_{attr}, CSR, CI, raw gate verdict) are in Appendix H, Table 6. The paper-bucket assignments in Table 3 follow mechanically from the raw per-cell verdict column of `cell_validity.csv` via the mapping in Appendix G.

How the counts read. The **3 ineligible** cells expose F1 and F5. CrowS-Pairs ($\times 2$) is archetype-ambiguous under our PLL + question-edit setup and hits a scorer-object mismatch (F1): the PLL scorer reads `choices` while format / semantic perturbations edit `question`, so $|\Delta_{\mathrm{sem}}| = 0$ is consistent with F1. Mistral $\times$ BBQ passes all implemented gates but its attribute family swaps demographic tokens while the answer key stays fixed, so Δ_{attr} is not a clean construct flip under CSR-as-defined (F5). The **3 scorer-unvalidated** cells (Qwen $\times$ XSTest plus both ToxiGen cells) pass all *implemented* gates but rely on scorers for which we supplied no external validation:

Table 2. Five failure classes, one realization per class in our canonical Qwen-2.5-7B / Mistral-7B panel. "Verdict" uses the short labels of §3.

ID	Symptom in our pipeline	Fix	Governance lesson
F1	`change_labels` was a 100% silent-no-op on the canonical MCQ scoring path for TruthfulQA and BBQ (legacy renderer was patched; canonical scorer file `scoring_canonical.py` was not).	Parametrise the canonical scorer's prompt builder.	Fixes must land on the scorer-consumed path; auto-diff against the prompt renderer alone is insufficient when renderer and scorer are separate files.
F2	Qwen×TruthfulQA legacy parseability 0.44; the regex defaulted 56% of items to incorrect, and perturbations that shifted generation style shifted parseability itself, so $\lvert\Delta_{\mathrm{fmt}}\rvert$ tracked regex coverage.	Switch to option logprobs (parseability $= 1.0$ by construction).	Sensitivity numbers below parseability 0.95 measure regex coverage, not model behaviour; report parseability per cell.
F3a	Mistral×CrowS legacy accuracy 0.99 reads as "excellent fairness" but is in fact *maximal* stereotypical preference under naive MCQ loading (`correct_indices` $= [0]$ pointing at `sent_more`).	Pseudo-log-likelihood over `choices[0]` and `choices[1]` (Salazar et al., 2020).	Benchmark conventions are load-bearing; a number reported without its convention statement inherits a construct inversion invisible in the number.
F3b	TruthfulQA MC2 loader returns `correct_indices` $= [0, \ldots, k-1]$; under any mild "prefer first option" positional bias the score is unfairly inflated (we observed `score_original` $= 1.000$ on a first canonical Qwen rerun).	Switch to MC1 with deterministic per-item shuffle.	Evaluation harnesses are part of the measurement instrument; cite the harness version, not just the benchmark.
F3c	After moving ToxiGen to first-token-logprob scoring, $\mathrm{lp}_{\text{toxic}} = \mathrm{lp}_{\text{non-toxic}} = -\infty$ on most items because neither target token fell in `top_k`=50; the scorer defaulted to majority class, producing $\lvert\Delta_{\mathrm{fmt}}\rvert = \lvert\Delta_{\mathrm{sem}}\rvert = 0$ exactly and $\lvert\Delta_{\mathrm{attr}}\rvert = 0.2$ as pure label-side artefact. *Introduced during repair of F1.*	`targeted_first_token_logprobs` that bypasses top-k entirely.	Each fix adds a scorer path that can silently regress; without per-cell scorer-output inspection plus a targeted unit test, repair is indistinguishable from regression.
F4	Independent-resampling bootstrap on legacy out-of-panel Llama gave CI width 26.4 at CSR ≈ 9.5 (Appendix J); in the canonical two-model panel the largest CI width on any cell with non-trivial denominator is roughly one third of that.	Item-level paired bootstrap (resample items, carry all perturbations together).	Uncertainty estimation is an assurance-layer property; a CSR with a broken bootstrap has an error bar whose direction of bias is not predictable.
F5	Canonical Qwen×BBQ produces $\lvert\Delta_{\mathrm{attr}}\rvert = 0.00$ at original score 0.34 (near-chance on a 3-option task). Diagnostic reading: "broken benchmark." Invariance reading: "maximally fair model." A single CSR cannot distinguish.	Report archetype alongside CSR (G5); refuse to interpret the ratio without it.	Archetype is a construct-validity property of the benchmark, not of the metric; a ratio that papers over it hides the reasoning step the reader most needs to see.

a 10-pattern refusal regex for XSTest, and a first-token-logprob targeting for ToxiGen whose tokenizer-coverage is audited only on a 100-item subset (Appendix B; mean target-token coverage 0.853 and 0.857, $p_5 = 0.64$ and 0.74). The **2 failed G2–G4** cells are Qwen × BBQ (fails G2: unperturbed score 0.340 is only $0.7\,\mathrm{pp}$ above the $1/3$ uniform baseline, under the $2\,\mathrm{SE} \approx 6.7\,\mathrm{pp}$ margin) and Mistral × XSTest (fails G2 *and* G3: unperturbed 0.565 vs. $1/2$ uniform, and $\max(\lvert\Delta_{\mathrm{fmt}}\rvert, \lvert\Delta_{\mathrm{sem}}\rvert) = 0.012 < 0.02$, the one truly denominator-dominated cell in the panel). The **2 exploratory** cells (Qwen × TruthfulQA and Mistral × TruthfulQA) pass all implemented gates with paired-bootstrap CIs wholly above 1 (CSR 13.04 with CI $[10.53, 17.65]$ and CSR 8.33 with CI $[6.67, 10.34]$); under paper precedence (G5, G6 PROPOSED), they are held as exploratory rather than confirmatory-selective. None of these four non-confirmatory statuses supports a construct-(in)validity conclusion about the underlying benchmark.

Table 3. Per-cell eligibility in the canonical panel. The primary bucket is the first-firing condition under §3; secondary flags mark non-primary F5 or scorer-unvalidated concerns. Raw verdicts and full numerics are in Appendix H.

Model	Benchmark	Primary bucket	Flags
Qwen	CrowS-Pairs	ineligible (F1)	—
Mistral	CrowS-Pairs	ineligible (F1)	—
Mistral	BBQ	ineligible (F5)	—
Qwen	XSTest	scorer-unvalidated	—
Qwen	ToxiGen	scorer-unvalidated	—
Mistral	ToxiGen	scorer-unvalidated	—
Qwen	BBQ	failed G2–G4	F5
Mistral	XSTest	failed G2–G4	SU
Qwen	TruthfulQA	exploratory	—
Mistral	TruthfulQA	exploratory	—

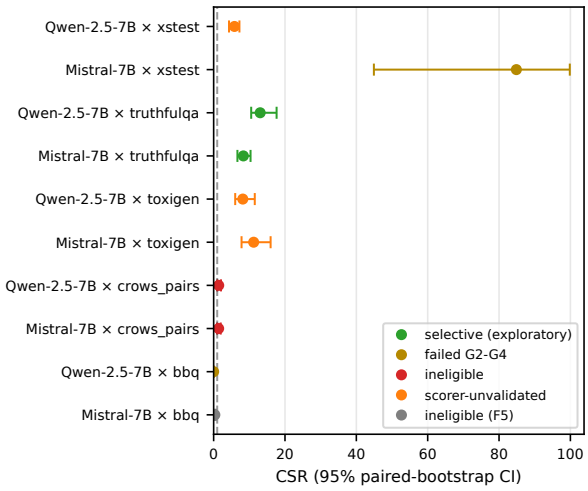

Figure 2. Canonical-pipeline CSR per cell with 1000-resample paired-bootstrap 95% CI. Colour encodes the gate verdict (§3). Mistral × XSTest fails G3 ($\max(|\Delta_{\text{fmt}}|, |\Delta_{\text{sem}}|) = 0.012 < 0.02$); no cell reaches *confirmatory-selective* under the full G1–G6 gate because G5, G6 remain PROPOSED. CSR values for *ineligible* and *scorer-unvalidated* cells are shown descriptively only and are not interpretable as benchmark-validity evidence (see §5). Full per-cell numbers are in Appendix H.

6. Discussion

Everything below is read off a *single* two-model, five-benchmark panel. We state the takeaways as what this case study illustrates, not as comprehensive findings; their generality is untested.

Why a self-audit chronology must be published. F3b (TruthfulQA MC2 correct-first ordering) and F3c (ToxiGen top-k truncation) had already produced plausible publishable numbers before per-cell scorer-output inspection exposed them. Any assurance-grade benchmark audit should therefore publish a self-audit chronology, not only cleaned final numbers. Box 1 gives the minimal fields; Appendix J is our worked instance. Without it, repair silently absorbed into a clean narrative is indistinguishable from a pipeline where the bugs were never caught. Appendix G maps the raw `cell_validity.csv` verdicts to Table 3, making the eligibility counts mechanical.

Box 1: Minimal self-audit chronology template

Per failure encountered, record: **(1)** a stable bug ID and its failure class (e.g. F1–F5 or a project-local code); **(2)** discovery date and *how* it was found (which inspection or test surfaced it—not just "code review"); **(3)** affected cells (every model × benchmark the defect touched); **(4)** the symptom *as it appeared in the reported numbers* (what a reader would have seen and believed); **(5)** provenance—*inherited* from the initial pipeline vs. *introduced during repair*; **(6)** the fix (the specific code change) *and* a regression test that fails on the pre-fix code; **(7)** the before/after delta on the headline statistic for each affected cell; **(8)** residual status: FIXED, DISCLOSED-UNFIXED, or OUT-OF-SCOPE, with the reason. *Per chronology as a whole*, also state: total entries, how many were repair-introduced, and which remain unfixed and why. An entry missing field (4) or (7) is not auditable: the reader cannot tell what the wrong number was or how much it moved.

Why we report four buckets, not one verdict. The four non-confirmatory statuses in §5 separate different consumer actions: *ineligible* calls for scope revision, *scorer-unvalidated* for scorer validation, *failed G2–G4* for threshold or evidence revision, and *exploratory* for disclosed, non-confirmatory release. Collapsing these into a single "inconclusive" verdict would hide the reasoning step that a governance reviewer most needs.

Zero confirmatory cells is procedural, not an empirical null. The zero confirmatory verdict is not a claim about the five benchmarks. Under §3, even a cell whose CI is wholly above 1 remains *exploratory* while G5 or G6 is PROPOSED; the two TruthfulQA cells would upgrade only after those gates are implemented. The contribution is the gate and taxonomy, not a benchmark-validity verdict. *No benchmark-validity claim is issued for any cell.*

7. Conclusion

Perturbation-based benchmark-validity audits can be silently wrong in at least five pipeline-level ways. We documented all five in a single 10-cell case study, and caught one (F3c) only after introducing it ourselves during repair. F1–F5 is an illustrative, non-exhaustive starting list whose calibration across codebases is open work.

The reusable artefact is the F1–F5 taxonomy, the §3 status-assignment algorithm, the G1–G6 gate, and the self-audit chronology of Box 1. This is a withholding and disclosure

protocol, not a scalar metric or benchmark leaderboard: all 10 cells in our panel remain non-confirmatory, and even a future *confirmatory-selective* cell would certify contrast magnitude, not signed direction (Appendix A).

Our ask is one sentence. *Any audit producing benchmark-based evidence for governance use should publish its self-audit chronology—at minimum the eight fields of Box 1—alongside its numbers.* Without it, a clean number and a silently broken pipeline look the same.

References

Alzahrani, N., Alyahya, H., Alnumay, Y., AlRashed, S., Alsubaie, S., Almushayqih, Y., Mirza, F., Alotaibi, N., Al-Twairesh, N., Alowisheq, A., Bari, M. S., and Khan, H. When benchmarks are targets: Revealing the sensitivity of large language model leaderboards. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 13787–13805. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.acl-long.744. URL https://aclanthology.org/2024.acl-long.744.

Blodgett, S. L., Lopez, G., Olteanu, A., Sim, R., and Wallach, H. Stereotyping Norwegian salmon: An inventory of pitfalls in fairness benchmark datasets. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 1004–1015. Association for Computational Linguistics, 2021. URL https://aclanthology.org/2021.acl-long.81.

Chen, Y., Qian, P., Wang, S., Zhang, S., Xu, H., Lin, S., and Wei, X. Does RAG know when retrieval is wrong? diagnosing context compliance under knowledge conflict, 2026. URL https://arxiv.org/abs/2605.14473.

Cronbach, L. J. and Meehl, P. E. Construct validity in psychological tests. *Psychological Bulletin*, 52(4):281–302, 1955.

Efron, B. and Tibshirani, R. J. *An Introduction to the Bootstrap*. Chapman and Hall / CRC, 1993.

European Parliament and Council. Regulation (eu) 2024/1689 laying down harmonised rules on artificial intelligence (artificial intelligence act), 2024. URL https://eur-lex.europa.eu/eli/reg/2024/1689/oj/eng. Official Journal of the European Union; see in particular Art. 55 on obligations of providers of general-purpose AI models with systemic risk.

Hartvigsen, T., Gabriel, S., Palangi, H., Sap, M., Ray, D., and Kamar, E. ToxiGen: A large-scale machine-generated dataset for adversarial and implicit hate speech detection. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3309–3326. Association for Computational Linguistics, 2022. URL https://aclanthology.org/2022.acl-long.234.

Jacobs, A. Z. and Wallach, H. Measurement and fairness. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency (FAccT)*, pp. 375–385. ACM, 2021. doi: 10.1145/3442188.3445901.

Ji, Y., Lan, W., and Ng, P. MRAG-Suite: A diagnostic evaluation platform for visual retrieval-augmented generation. *arXiv preprint arXiv:2509.24253*, 2025. doi: 10.48550/arXiv.2509.24253. URL https://arxiv.org/abs/2509.24253.

Ji, Y., Li, Z., Meng, R., and He, D. Retrieval–reasoning processes for multi-hop question answering: A four-axis design framework and empirical trends. *arXiv preprint arXiv:2601.00536*, 2026. doi: 10.48550/arXiv.2601.00536. URL https://arxiv.org/abs/2601.00536.

Jiang, X., Yang, S., Yang, W., Liu, Y., and Ji, C. SOK: A taxonomy of attack vectors and defense strategies for agentic supply chain runtime. *arXiv preprint arXiv:2602.19555*, 2026. doi: 10.48550/arXiv.2602.19555. URL https://arxiv.org/abs/2602.19555.

Kiela, D., Bartolo, M., Nie, Y., Kaushik, D., Geiger, A., Wu, Z., Vidgen, B., Prasad, G., Singh, A., Ringshia, P., Ma, Z., Thrush, T., Riedel, S., Waseem, Z., Stenetorp, P., Jia, R., Bansal, M., Potts, C., and Williams, A. Dynabench: Rethinking benchmarking in NLP. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pp. 4110–4124. Association for Computational Linguistics, 2021. URL https://aclanthology.org/2021.naacl-main.324.

Lalor, J. P., Wu, H., and Yu, H. Learning latent parameters without human response patterns: Item response theory with artificial crowds. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 4249–4259. Association for Computational Linguistics, 2019. URL https://aclanthology.org/D19-1434.

Lan, T., Xu, J., He, X., Hwang, J.-N., and Li, L. Attention consistency for LLMs explanation. In *Findings of the Association for Computational Linguistics:*

EMNLP 2025, pp. 1736–1750. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.findings-emnlp.91. URL https://aclanthology.org/2025.findings-emnlp.91.

Li, Y., Fan, Z., and Zhuang, Z. Auditing reasoning-trace memorization claims after unlearning with head-conditioned canaries. *arXiv preprint arXiv:2605.18891*, 2026a. URL https://arxiv.org/abs/2605.18891.

Li, Y., Fan, Z., and Zhuang, Z. SafetyRepro: Configuration-conditional rank instability on alignment benchmarks. *arXiv preprint arXiv:2605.25492*, 2026b. URL https://arxiv.org/abs/2605.25492.

Lin, S., Hilton, J., and Evans, O. TruthfulQA: Measuring how models mimic human falsehoods. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3214–3252. Association for Computational Linguistics, 2022. URL https://aclanthology.org/2022.acl-long.229.

Luo, H., Deng, Z., Chen, R., and Liu, Z. FAIntbench: A holistic and precise benchmark for bias evaluation in text-to-image models. *arXiv preprint arXiv:2405.17814*, 2024a. URL https://arxiv.org/abs/2405.17814.

Luo, H., Huang, H., Deng, Z., Li, X., Wang, H., Jin, Y., Liu, Y., Xu, W., and Liu, Z. BIGbench: A unified benchmark for evaluating multi-dimensional social biases in text-to-image models. *arXiv preprint arXiv:2407.15240*, 2024b. URL https://arxiv.org/abs/2407.15240.

Luo, H., Dai, S., Ni, C., Li, X., Zhang, G., Wang, K., Liu, T., and Salam, H. AgentAuditor: Human-level safety and security evaluation for LLM agents. *arXiv preprint arXiv:2506.00641*, 2025. URL https://arxiv.org/abs/2506.00641. Accepted to NeurIPS 2025.

Luo, H., Huang, Z., Chung, S., Wang, Y., Jin, Y., Li, J., Li, J., Li, X., and Salam, H. AtelierEval: Agentic evaluation of humans & LLMs as text-to-image prompters. In *Proceedings of the 43rd International Conference on Machine Learning (ICML)*, 2026a. URL https://openreview.net/forum?id=Q8LOMDdr2y. arXiv:2605.22645.

Luo, H., Huang, Z., Huang, H., Deng, Z., Chen, R., Li, X., Liu, Z., and Salam, H. BiasIG: Benchmarking multi-dimensional social biases in text-to-image models. *arXiv preprint arXiv:2604.11934*, 2026b. doi: 10.48550/arXiv.2604.11934. URL https://arxiv.org/abs/2604.11934. Accepted to IJCNN 2026.

Luo, H., Ni, C., Wen, J., Huang, Z., Wang, Y., Liao, B., Chung, S., Jin, Y., Li, X., Xu, W., Wang, X., and Salam, H. CentaurEval: Benchmarking human-in-the-loop value in agentic coding. In *Proceedings of the 43rd International Conference on Machine Learning (ICML)*, 2026c. URL https://openreview.net/forum?id=GR335S9IgZ. arXiv:2512.04111.

Messick, S. J. Validity of psychological assessment: Validation of inferences from persons' responses and performances as scientific inquiry into score meaning. *American Psychologist*, 50(9):741–749, 1995. doi: 10.1037/0003-066X.50.9.741.

Mizrahi, M., Kaplan, G., Malkin, D., Dror, R., Shahaf, D., and Stanovsky, G. State of what art? a call for multi-prompt LLM evaluation. *Transactions of the Association for Computational Linguistics*, 12:933–949, 2024. doi: 10.1162/tacl_a_00681. URL https://aclanthology.org/2024.tacl-1.52. arXiv:2401.00595.

Nangia, N., Vania, C., Bhalerao, R., and Bowman, S. R. CrowS-Pairs: A challenge dataset for measuring social biases in masked language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1953–1967. Association for Computational Linguistics, 2020. URL https://aclanthology.org/2020.emnlp-main.154.

National Institute of Standards and Technology. Artificial intelligence risk management framework (AI RMF 1.0). NIST AI 100-1, 2023. URL https://www.nist.gov/itl/ai-risk-management-framework.

Parrish, A., Chen, A., Nangia, N., Padmakumar, V., Phang, J., Thompson, J., Htut, P. M., and Bowman, S. R. BBQ: A hand-built bias benchmark for question answering. In *Findings of the Association for Computational Linguistics: ACL 2022*, pp. 2086–2105. Association for Computational Linguistics, 2022. URL https://aclanthology.org/2022.findings-acl.165.

Qian, P., Wang, S., Wang, X., Chen, Y., Xu, W., Yu, Q., Lin, S., Zhang, S., You, J., and Wei, X. Relevant is not warranted: Evidence-force calibration for cited RAG, 2026. URL https://arxiv.org/abs/2605.28044.

Raji, I. D., Bender, E. M., Paullada, A., Denton, E., and Hanna, A. AI and the everything in the whole wide world benchmark. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, 2021.

Reuel, A., Bucknall, B., Casper, S., Fist, T., Soder, L., Aarne, O., Hammond, L., Ibrahim, L., Chan, A., Wills,

P., Anderljung, M., Garfinkel, B., Heim, L., Trask, A., Mukobi, G., Schaeffer, R., Baker, M., Hooker, S., Solaiman, I., Luccioni, A. S., Rajkumar, N., Moës, N., Ladish, J., Bau, D., Bricman, P., Guha, N., Newman, J., Bengio, Y., South, T., Pentland, A., Koyejo, S., Kochenderfer, M. J., and Trager, R. Open problems in technical AI governance. *Transactions on Machine Learning Research*, 2025. URL https://arxiv.org/abs/2407.14981. arXiv:2407.14981.

Ribeiro, M. T., Wu, T., Guestrin, C., and Singh, S. 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. Association for Computational Linguistics, 2020. URL https://aclanthology.org/2020.acl-main.442.

Röttger, P., Kirk, H. R., Vidgen, B., Attanasio, G., Bianchi, F., and Hovy, D. XSTest: A test suite for identifying exaggerated safety behaviours in large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 5377–5400. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.naacl-long.301. URL https://aclanthology.org/2024.naacl-long.301.

Salazar, J., Liang, D., Nguyen, T. Q., and Kirchhoff, K. Masked language model scoring. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 2699–2712. Association for Computational Linguistics, 2020. URL https://aclanthology.org/2020.acl-main.240.

Sclar, M., Choi, Y., Tsvetkov, Y., and Suhr, A. Quantifying language models' sensitivity to spurious features in prompt design, or: How i learned to start worrying about prompt formatting. In *The Twelfth International Conference on Learning Representations (ICLR)*, 2024. URL https://openreview.net/forum?id=RIu5lyNXjT.

State Administration for Market Regulation and Standardization Administration of China. GB/T 45654–2025, cybersecurity technology—basic security requirements for generative artificial intelligence service. National Standard of the People's Republic of China, 2025. URL https://openstd.samr.gov.cn/bzgk/std/newGbInfo?hcno=F67D3F376E0A0A0FF5317FB36B32A30A. Issued 2025-04-25; implemented 2025-11-01.

Vania, C., Htut, P. M., Huang, W., Mungra, D., Pang, R. Y., Phang, J., Liu, H., Cho, K., and Bowman, S. R. Comparing test sets with item response theory. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 1141–1158. Association for Computational Linguistics, 2021. URL https://aclanthology.org/2021.acl-long.92.

Wang, S., Qian, P., Chen, Y., You, J., Wang, X., Jiang, X., Liu, L., Yu, H., and Xu, J. When safe skills collide: Measuring compositional risk in agent skill ecosystems, 2026a. URL https://arxiv.org/abs/2606.00448.

Wang, Y., Sun, X., Li, Y., Fan, Z., and Zhuang, Z. Auditing and fixing economic validity in tabular foundation models for discrete choice. *arXiv preprint arXiv:2605.26559*, 2026b. URL https://arxiv.org/abs/2605.26559.

Zhuang, Z., Li, Y., and Fan, Z. Pre-registering the detectable effect: A paired-MDE budget for 4-bit quantization benchmarks, with a pilot audit. *arXiv preprint arXiv:2605.28873*, 2026. URL https://arxiv.org/abs/2605.28873.

A. Detailed Limitations

We place limitations here so the reader can calibrate every number in the main text against what this evidence cannot support.

1. **Two models, five benchmarks, 200 items per cell, one seed, one prompt template.** Our 10-cell grid (Qwen-2.5-7B and Mistral-7B-v0.3 $\times$ five benchmarks) cannot support any claim that a class of 7–8B open-weight models exhibits any pattern.

2. **Post-prereg exploratory status.** Our preregistration (§K) was committed before we discovered failures F3b and F3c; analyses after those discoveries are formally exploratory.

3. **Structurally-ineligible cells.** CrowS-Pairs is archetype-ambiguous under our PLL + demographic-swap configuration and additionally hits F1 scorer-path mismatch; BBQ is an invariance benchmark penalised by our diagnostic ratio metric (F5). Their numbers illustrate the two failure classes but are not benchmark-validity claims.

4. **Unvalidated scorers for ToxiGen and XSTest.** ToxiGen scoring uses targeted first-token log-probabilities over a fixed token set (after the F3c repair); XSTest refusal classification uses a ten-pattern regex over generations that we did not separately audit for false-negatives against human judgement. Neither scorer is a human-judged label; both cells are reported in the *scorer-unvalidated* bucket.

5. **CSR is a contrastive-magnitude metric, not a signed-direction metric.** CSR uses unsigned family-deltas, so a paired-bootstrap CI wholly above 1 only certifies that $|\Delta_{\text{attr}}|$ exceeds surface-family magnitudes. It does *not* certify that attribute-flip responses are in the construct-consistent direction: a large $|\Delta_{\text{attr}}|$ is consistent with any of construct-sensitive flipping, sycophancy flipping, lexical-salience noise, or an uncaught scorer artefact. Upgrading a *confirmatory-selective* reading to *construct-sensitive* requires signed-effect validation (the E3 human audit, deferred) and is future work.

6. **Canonical scorers are audit-hardened, not reference-faithful across the board.** TruthfulQA MC1 with deterministic per-item shuffle, ToxiGen targeted-token first-token logprobs, and XSTest's 10-pattern refusal regex are audit-robust substitutes rather than reference-faithful implementations; see Appendix D for the reference / audit / validation split. CrowS-Pairs PLL on paired sentences (Salazar et al., 2020) and BBQ option log-probabilities split by `context_condition` are reference-faithful.

7. **The six-point checklist is a proposed tool, not a standard.** It does not validate thresholds against external reference data, does not assign institutional ownership, and does not include anti-gaming provisions. Three further benchmark-specific limitations (shared-prefix MCQ tokens, `add_instruction` contamination, weak surface edits on binary tasks) are listed in Appendix J.

B. Panel and Compute

Compute: a single NVIDIA RTX 4080 (16 GB) with HuggingFace Transformers 4.57 in bfloat16; approximately 26 GPU-hours total across the legacy and canonical pipelines. The panel originally included Llama-3.1-8B; per author decision, Llama was dropped after the legacy runs and before the canonical rerun. Only the two-model canonical panel is the case study of this paper; legacy-Llama numbers appear only in this appendix and the case-level audit trail (Appendix J). Software versions: Python 3.13, PyTorch 2.10, `transformers` 4.57.6; model revisions pinned at load time (`Qwen/Qwen2.5-7B-Instruct` and `mistralai/Mistral-7B-Instruct-v0.3`; tokenizer revision recorded in results JSON).

C. Perturbation-Family Realisations

Three families per item; three perturbations per item per family. Parse failures (legacy regex only) are symmetric: if either the original or the perturbed item fails parsing, the item-level paired difference for that family is dropped. *Format:* case changes, whitespace / punctuation noise, label-format changes, choice reordering, and—for text-classification benchmarks only—instruction prepend. We retain `add_instruction` in the format family despite its task-boundary contamination on ToxiGen (Appendix J marks this as unfixed-acknowledged). *Semantic:* paraphrase templates, synonym substitution, rule-based sentence restructure. *Attribute:* benchmark-specific construct flips: opposite-construct stem rewrite (TruthfulQA), demographic swap (BBQ, CrowS-Pairs), toxicity-flip rewrite (ToxiGen), safe↔unsafe pair lookup (XSTest).

D. Scoring Pipelines

The *legacy* pipeline uses a uniform regex-extraction scorer over free-form generation for all multiple-choice benchmarks and a first-token logit comparison for ToxiGen and XSTest. The *canonical* pipeline implements each benchmark's intended scoring protocol: option log-probabilities for TruthfulQA MC1, option log-probabilities split by `context_condition` for BBQ, pseudo-log-likelihood over the paired sentences for CrowS-Pairs (following Salazar et al. (2020)), targeted-token first-token log-probabilities for ToxiGen, and a ten-pattern regex refusal classifier over generations for XSTest.

The legacy→canonical fix list: read `label_format` when rendering MCQ prompts (F1 fix); replace `flip_truthfulqa` label inversion with opposite-construct stem rewrite; switch CrowS scoring to PLL on paired sentences (F3a); switch TruthfulQA loading from MC2 to MC1 to eliminate correct-first ordering (F3b); use targeted-token first-token log-probabilities for ToxiGen to bypass top-k truncation (F3c); switch the bootstrap to an item-level paired resampler (F4).

Reference vs. audit vs. validation evidence. "Canonical" collapses three distinct things that should be kept apart when reading downstream claims. Table 4 splits them.

Table 4. Scorer fidelity split. *Reference scorer* = the benchmark's published scoring protocol. *Audit scorer used here* = the scorer our canonical pipeline actually runs. *Validation evidence* = what we have to back the audit scorer; NONE means the cell ships as *scorer-unvalidated* under §3.

Benchmark	Reference scorer	Audit scorer used here	Validation evidence
TruthfulQA	MC1 over options, accuracy	MC1 over options with deterministic per-item shuffle (F3b fix)	Reference-faithful up to the shuffle, which does not change the population metric.
BBQ	Option log-probabilities split by `context_condition`	Same	Reference-faithful.
CrowS-Pairs	PLL on paired sentences (Salazar et al., 2020)	Same, reading the `choices` field	Reference-faithful on the scored object; G1 instrumentation against the *question*-edit path is not yet in place, which is why both cells ship as ineligible-F1.
ToxiGen	First-token logprob of `toxic`/`non-` tokens (post-F3c fix)	Same, with targeted-token bypass of top-k	Tokenizer-coverage probe on a 100-item subset only; NONE human-label. Ships as scorer-unvalidated.
XSTest	Refusal classification	10-pattern regex refusal classifier over generations	NONE; no false-negative audit against human labels. Ships as scorer-unvalidated.

E. Threshold Honesty

The 0.02 threshold in G3 and the 0.01 floor of ε are *engineering choices*, not statistical thresholds. The intent is to avoid ratio explosion when both surface-family deltas are near zero; the specific numeric choice is tied to our 200-item subset size but we do not claim it is optimal or that 0.02 equals "two standard errors" of anything (for a proportion at $p = 0.5$ with $n = 200$, $2\,\mathrm{SE} \approx 0.07$, so 0.02 is substantially more permissive than a strict statistical-significance threshold). The 20% silent-no-op cap in G1 is likewise pragmatic, motivated by our observed 0–83% range (Appendix I). A fuller treatment would derive each threshold from a target sensitivity level in the downstream evidence consumer. Satisfying G1–G6 does not establish that any benchmark is construct-valid; it substantially reduces the risk that non-confirmation is driven by known preventable pipeline failures. A production due-diligence instrument would additionally require institutional ownership, externally-validated thresholds, anti-gaming provisions, and legal-compatibility review.

Threshold sensitivity of the 10-cell breakdown. We resweep the three threshold knobs and re-derive the five-bucket counts of §5 under each setting; the qualitative breakdown is stable over the grid we considered. **G3** in $\{0.01, 0.02, 0.05\}$: the only boundary cell is Mistral × XSTest with $\max(|\Delta_{\mathrm{fmt}}|, |\Delta_{\mathrm{sem}}|) = 0.012$. At 0.01 that cell passes G3 but still fails G2 (0.565 vs. 0.500, under $2\,\mathrm{SE}$), so it remains *failed G2–G4*; at 0.05 it fails both G2 and G3 and remains *failed G2–G4*. Every other in-panel cell with non-degenerate denominator has $\max(|\Delta_{\mathrm{fmt}}|, |\Delta_{\mathrm{sem}}|) \geq 0.050$, and the F1-degenerate CrowS cells are pre-filtered. ε in $\{0.001, 0.01, 0.1\}$: ε only floors the CSR denominator; it shifts CSR point estimates on denominator-dominated

cells but does not change any G2 or G3 outcome, so bucket assignment is unchanged. **G1 silent-no-op cap** in $\{5, 10, 20\}\%$: the observed canonical-scorer no-op rates are $\{0.0, 0.0, 100, 100, 83\}\%$ for XSTest/ToxiGen/TruthfulQA/BBQ/CrowS; any cap $\leq 83\%$ leaves CrowS F1-ineligible and TruthfulQA/BBQ F1-passing (their no-op rates are 0% on the canonical path after the F1b repair). The $3/3/2/2/0$ breakdown therefore holds over this grid. Thresholds outside the grid (e.g., G3 $= 0.1$, which would flag denominators up to half the $p = 0.5, n = 200$ 2-SE of 0.07) would reclassify TruthfulQA's exploratory cells; we report the grid above as the regime where the thresholds are defensible as conservative engineering floors.

F. Per-Benchmark Eligibility Notes

The main-body eligibility breakdown (§5, Table 3, Figure 2) summarises per-cell status under the §3 precedence. Additional per-benchmark notes:

TruthfulQA. Both cells score above the $1/4$ uniform baseline (G2 passes: Qwen 0.485, Mistral 0.520) with $\max(|\Delta_{\text{fmt}}|, |\Delta_{\text{sem}}|) \geq 0.0767$ (G3 passes) and paired bootstrap CIs wholly above 1 (Qwen CSR 13.04, CI $[10.53, 17.65]$; Mistral CSR 8.33, CI $[6.67, 10.34]$). Both cells therefore land in the *exploratory* bucket pending G5 and G6 enforcement. **BBQ.** Mistral × BBQ passes all implemented gates but is *ineligible (F5)* because the attribute family swaps demographic tokens while the answer key stays fixed, so Δ_{attr} is not a clean construct flip. Qwen × BBQ fails G2 (unperturbed score 0.340 vs. $1/3$ baseline is $0.7\,\text{pp}$ above, well under the $2\,\text{SE} \approx 6.7\,\text{pp}$ margin) and additionally inherits the F5 concern; it falls in the *failed G2–G4* bucket under first-match precedence. **CrowS-Pairs.** Both cells are *not_applicable* (mapped to *ineligible* in paper): the canonical PLL scorer reads `choices`, while the format / semantic perturbations edit `question`; the perturbation does not reach the scored object (F1), and archetype is ambiguous under PLL + question-edit. $|\Delta_{\text{sem}}| = 0$ on both cells is consistent with F1. **ToxiGen.** Both cells pass G1–G4 but the scorer (first-token logprob over `toxic`/`non-` tokens) is validated only on a 100-item tokenizer-coverage probe (Appendix B; mean coverage 0.853 and 0.857, $p_5 = 0.64$ and 0.74). Absent a held-out human-judged validation, both ToxiGen cells are labelled *scorer-unvalidated*. **XSTest.** The 10-pattern refusal regex has not been separately audited for false-negative rate. Qwen × XSTest passes all implemented gates (*scorer-unvalidated*); Mistral × XSTest fails G2 above-baseline *and* is denominator-dominated at $\max(|\Delta_{\text{fmt}}|, |\Delta_{\text{sem}}|) = 0.012 < 0.02$ (G3 fails), falling in *failed G2–G4*.

The canonical 200-item ToxiGen slice is class-balanced, so the uniform and majority baselines coincide at 0.5. Gate decisions use full precision throughout; displayed values may round to the threshold.

Seed stability and placebo specificity (PREREG E5, E6). Of the 9 cells with non-degenerate CSR (Qwen-2.5-7B × BBQ fails G2 above-baseline, so its seed CV is undefined), per-cell CSR coefficient of variation across seeds $\{42, 123, 2024\}$ is ≤ 0.27 on all 9 (smallest 0.021 on Qwen × TruthfulQA; largest 0.266 on Mistral × XSTest). Seed instability does not drive any verdict. The placebo specificity ratio $\text{CSR}_{\text{real}}/\text{CSR}_{\text{placebo}}$ is ≥ 4 on 4 of the 8 cells with $\text{CSR}_{\text{real}} > 0$ (Qwen × XSTest 5.00, Qwen × ToxiGen 4.47, Mistral × XSTest 74.25, Mistral × ToxiGen 5.61; median across all 8 cells is 3.98). Excluding the two BBQ cells where $\text{CSR}_{\text{real}} \approx 0$ so the ratio is denominator-noise dominated (raw ratios 0.017 and 0.165), the ratio is ≥ 4 on 4 of 6 cells, median 4.74. This is positive but non-universal support for contrast specificity. Source data: `results/canonical/seed_variance.csv` and `results/canonical/csr_placebo.csv`.

G. Verdict-Mapping Rule

The raw `cell_validity.csv verdict` column encodes, per cell, the first condition that fires under the status-assignment algorithm of §3. The raw-verdict tokens and the algorithm steps that emit them are: `not_applicable` (G1 uninstrumented, step 1), `inconclusive` (at least one of G2/G3/G4 fails, step 2), `uninterpretable_scorer` (gates pass, scorer not validated, step 3), `attribute_family_mismatch` (gates pass and scorer validated, but archetype-mismatched under CSR-as-defined, step 4), and `selective` / `non_selective` (all of the above pass; CI wholly above / below 1; step 5 or 6). Per §3, the raw-verdict tokens are emitted in the order they are checked; a reader with the raw CSV can reproduce every count in the paper by applying the table below mechanically.

Worked example: Mistral × XSTest. Raw verdict `inconclusive`; algorithm trace: step 1 fires G1? No—XSTest's 10-pattern refusal regex reads the generated text and our format perturbations do reach that path (G1 is instrumented, though the scorer itself is unvalidated; that is step 3's concern, not step 1's). Step 2: G2 fails (unperturbed $0.565 < 0.500 + 2\,\text{SE}$) *and* G3 fails ($\max(|\Delta_{\text{fmt}}|, |\Delta_{\text{sem}}|) = 0.012 < 0.02$). Step 2 fires; raw verdict `inconclusive`; paper bucket *failed G2–G4*. Step 3 (scorer-unvalidated) would have fired had G2–G4 passed, but it did not. The scorer-unvalidated flag is still

Table 5. Raw `cell_validity.csv` `verdict` → paper bucket. The raw-verdict emission order matches §3 exactly, so no first-match contention occurs at the mapping step. A cell with `verdict=selective` is downgraded to *exploratory* in this submission because G5 and G6 remain PROPOSED; it would upgrade to *confirmatory-selective* once both gates are IMPLEMENTED.

Raw `verdict`	Paper bucket	§3 step
`not_applicable`	ineligible (F1)	1
`inconclusive`	failed G2–G4	2
`uninterpretable_scorer`	scorer-unvalidated	3
`attribute_family_mismatch`	ineligible (F5)	4
`selective` (G5 / G6 PROPOSED)	exploratory	5
`non_selective` (G5 / G6 PROPOSED)	exploratory	5
`selective` (G5 / G6 IMPLEMENTED)	confirmatory-selective	6
`non_selective` (G5 / G6 IMPLEMENTED)	confirmatory-non-selective	6

surfaced in the appendix eligibility notes as a secondary concern.

Worked example: Qwen × BBQ. Raw verdict `inconclusive`; step 1 does not fire (BBQ option-logprob scorer is fully instrumented); step 2 fires because G2 fails ($0.340 < 0.333 + 2\,\mathrm{SE}$); bucket *failed G2–G4*. The F5 archetype-mismatch concern is benchmark-level and would have fired at step 4 had G2–G4 passed (as on Mistral × BBQ).

Cell-level `gate_*` columns vs. paper G1–G6. The raw `cell_validity.csv` gate columns (`gate_parseability`, `gate_above_baseline`, `gate_denom_not_eps`, `gate_ci_estimable`, `gate_n_items_ok`, `gate_seed_cv_lt_03`) are the empirical per-cell instantiation of the methodology-level G1–G6 in §3, not a separate gate set. `gate_parseability` realises the regex-parseability half of G1; the silent-no-op half of G1 is audited offline against the canonical scorer's prompt builder (Table 7), which is why G1 ships as PARTIAL in Table 1. `gate_above_baseline` realises G2; `gate_denom_not_eps` realises G3; `gate_ci_estimable` + `gate_n_items_ok` + `gate_seed_cv_lt_03` jointly realise G4 (paired-bootstrap feasibility, sample-size guard, seed-stability guard). The CSV carries no direct column for G5 (archetype disclosure) or G6 (repair-regression); their PROPOSED status is what downgrades a raw-verdict `selective` to paper-bucket *exploratory* under §3.

H. Case-Study Table: 10 Cells, Canonical Pipeline

Table 6 is the per-cell evidence behind the counts in §6 / Table 3. All values are taken directly from `results/canonical/csr_canonical.csv` (per-family deltas, CSR point estimate) and `results/canonical/cell_validity.csv` (paired-bootstrap CI, raw verdict). Gate decisions use full precision throughout.

Table 6. Canonical-pipeline case-study numbers, Qwen-2.5-7B and Mistral-7B-v0.3. Raw verdicts are the literal `cell_validity.csv` values; the paper-bucket mapping is Appendix G. Baselines: TruthfulQA MC1 $1/4$, BBQ $1/3$, ToxiGen 0.5, XSTest $1/2$.

Model	Benchmark	s_{orig}	Δ_{fmt}	Δ_{sem}	Δ_{attr}	CSR	95% CI	Raw verdict
Qwen-2.5-7B	CrowS-Pairs	0.655	0.095	0.000	0.135	1.42	[1.01, 2.06]	`not_applicable`
Qwen-2.5-7B	BBQ	0.340	0.050	0.000	0.000	0.00	[0.00, 0.00]	`inconclusive`
Qwen-2.5-7B	XSTest	0.620	0.140	0.140	0.817	5.83	[4.31, 7.23]	`uninterpretable_scorer`
Qwen-2.5-7B	TruthfulQA	0.485	0.077	0.000	1.000	13.04	[10.53, 17.65]	`selective`
Qwen-2.5-7B	ToxiGen	0.785	0.058	0.087	0.708	8.17	[6.04, 11.53]	`uninterpretable_scorer`
Mistral-7B	CrowS-Pairs	0.670	0.095	0.000	0.133	1.40	[0.99, 1.98]	`not_applicable`
Mistral-7B	BBQ	0.425	0.113	0.088	0.038	0.34	[0.20, 0.51]	`attribute_family_mismatch`
Mistral-7B	XSTest	0.565	0.012	0.008	0.990	84.86	[44.92, 99.83]	`inconclusive`
Mistral-7B	TruthfulQA	0.520	0.120	0.105	1.000	8.33	[6.67, 10.34]	`selective`
Mistral-7B	ToxiGen	0.765	0.062	0.053	0.692	11.22	[7.82, 15.97]	`uninterpretable_scorer`

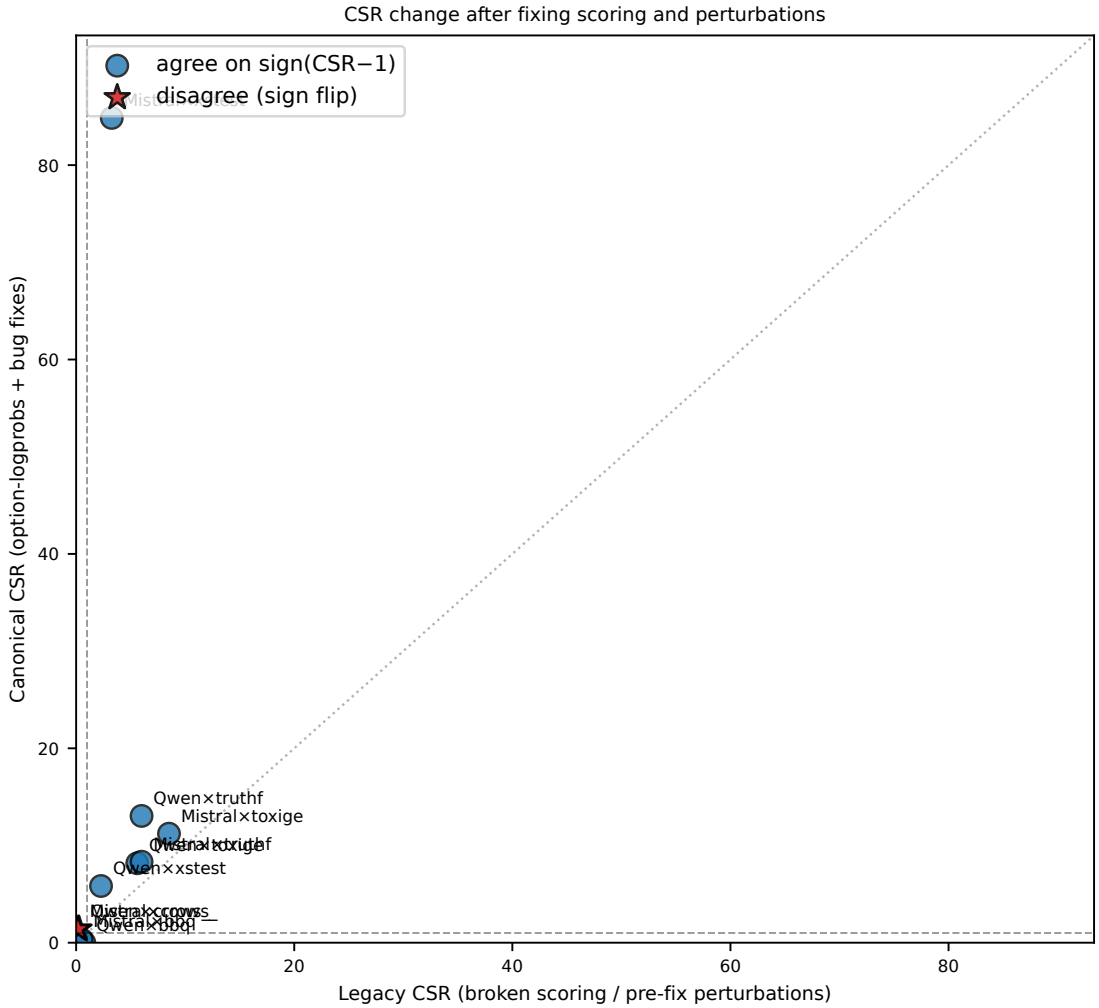

Figure 3. Legacy-pipeline CSR (x-axis) versus canonical-pipeline CSR (y-axis) per cell, on *symlog* axes (linear in $[-1, 1]$, log outside) so both the near-zero BBQ cells and the Mistral × XSTest outlier at CSR ≈ 85 are readable on one panel. Points on the dotted identity line are unchanged by repair; points far from it reflect the joint effect of F1–F3 fixes. Seven of ten cells change verdict (selective ↔ non-selective or denominator-dominated) under the repaired pipeline. Cell codes: Q = Qwen-2.5-7B, M = Mistral-7B-v0.3; TQ = TruthfulQA, BB = BBQ, TX = ToxiGen, CP = CrowS-Pairs, XS = XSTest.

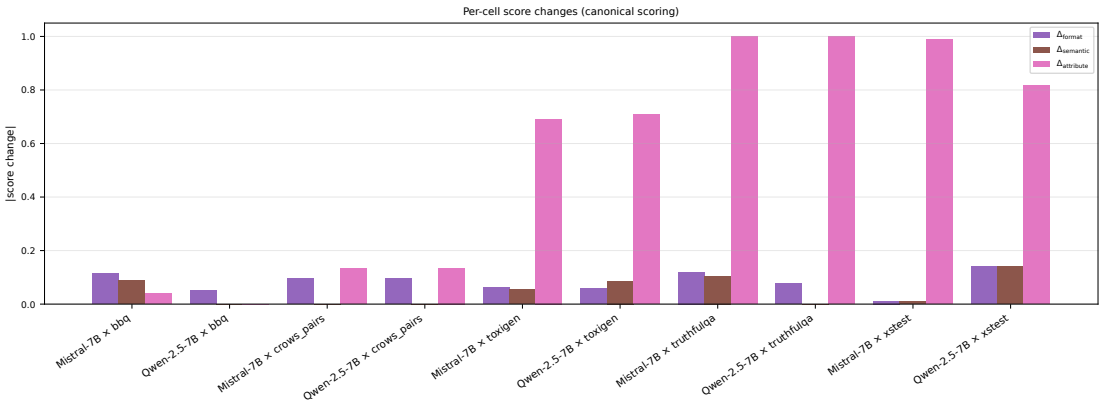

Figure 4. Per-cell $|\Delta_{\text{fmt}}|$, $|\Delta_{\text{sem}}|$, $|\Delta_{\text{attr}}|$ under the canonical pipeline. Cells where $|\Delta_{\text{fmt}}|$ or $|\Delta_{\text{sem}}| < 0.02$ drive the denominator below the ε floor and trigger G3. $|\Delta_{\text{sem}}| = 0$ on CrowS-Pairs reflects F1-type silent no-ops on the question field, which the PLL scorer does not read.

I. Auto-Diff Audit Numbers

Silent-no-op rates of format perturbations per benchmark, measured (i) against the legacy renderer (`evaluate.format_mcq_prompt`) and (ii) against the canonical scorer's prompt builder (`scoring_canonical._format_mcq_prompt`). The gap between the two columns is the core F1 evidence.

Table 7. Silent-no-op rate (%) for the format family. Column L = against the legacy renderer; column C = against the canonical scorer's prompt builder. The F1 regression shows in the C column for MCQ benchmarks (100% no-op).

Benchmark	L (legacy)	C (canonical MCQ)
TruthfulQA	0.2	100
BBQ	1.8	100
CrowS-Pairs	11.0	83 †
ToxiGen	0.0	–
XSTest	0.0	–

† CrowS under PLL: 5 of 6 perturbation methods do not reach the scorer because the scorer reads `choices` and the perturbations edit `question`. The 83% figure is the effective scorer-consumed no-op rate.

J. Case-Level Audit Trail

This appendix is the worked instance of the minimal chronology template (Box 1): each entry below carries a bug ID and failure class (field 1), provenance (field 5: F3c is flagged *self-introduced*), the fix and its regression test (field 6: tests shipped for F1, F3c, F4), and residual status (field 8: FIXED / UNFIXED, ACKNOWLEDGED). For space, the per-cell discovery date, affected-cell list, and before/after deltas (fields 2–4, 7) are condensed here and given in full in the release bundle's running audit log (Appendix K); the F4 entry quotes its before/after delta inline as an example of field 7. All items listed as FIXED are repaired in the canonical pipeline that produced every number in this paper; the trail exists so a reader can audit our audit, not because the canonical pipeline ships with unresolved bugs. Three items remain UNFIXED, ACKNOWLEDGED and are disclosed as active limitations.

Entries are chronological, matched to the failure-mode cases in §4.

1. **F1a (fixed)**: `format_change_labels` wrote `item['label_style']`; prompt builder did not read it. Fix: parametrise `format_mcq_prompt(label_style)`.
2. **F1b (fixed)**: Fix landed only in `evaluate.py`; `scoring_canonical.py::_format_mcq_prompt` remained hard-coded. Fix: parametrise the canonical builder.
3. **F1c (fixed)**: Rule-based `semantic_sentence_restructure` had no-op rates of 57–93% due to trigger-string mismatch on Q&A benchmarks. Fix: add `paraphrase_template`, `synonym_substitution`.
4. **F2a (partially fixed)**: Qwen × TruthfulQA legacy parseability 0.44. Fix: canonical option-logprob scoring sets parseability to 1.0; legacy numbers are withdrawn.
5. **F3a (disclosed, not repaired in data)**: CrowS-Pairs legacy accuracy inverts fairness interpretation. Canonical pipeline switches to PLL; CrowS legacy numbers are withdrawn.
6. **F3b (fixed)**: TruthfulQA MC2 correct-first ordering artefact. Fix: MC1 + per-item shuffle.
7. **F3c (fixed, self-introduced)**: Top-k truncation caused majority-class predictions on ToxiGen. Fix: targeted-token first-token log-probabilities.
8. **F4 (fixed)**: Broken bootstrap pairing. Fix: item-level paired bootstrap. Out-of-panel legacy-Llama at CSR ≈ 9.5 showed CI width 26.4 under unpaired resampling.
9. **F5 (disclosed)**: BBQ / CrowS archetype mismatch under CSR. Not fixed; G3 gate flags denominator-dominated cells.
10. **Shared-prefix collapse on MCQ letter tokens** (UNFIXED, ACKNOWLEDGED): first token of "`(A)`", "`(B)`" can be identical.
11. **TEXTCLF `add_instruction` content contamination on ToxiGen** (UNFIXED, ACKNOWLEDGED).

Containment of unfixed issues. The remaining unfixed items do not alter any *confirmatory* verdict in this paper because no confirmatory verdict is issued: all affected cells remain non-confirmatory under the hierarchical precedence of §3. The unfixed items therefore further motivate—rather than undercut—the paper's withholding rule.

K. Preregistration and Release

Code and data. A release bundle will accompany the de-anonymised version of the paper, containing the canonical pipeline (all F1–F4 fixes landed), per-item predictions for all 10 cells, the preregistration `PREREG.md` (commit `e6e686b8c0b02dd78d1495eb9c4a94882cce5cce`, amended 2026-04-22), running audit logs, and regression tests for F1, F3c, and F4. The bundle is not part of this submission and the URL is withheld for blind review. Hardware and software versions are in Appendix B.

Deviations from preregistration. The 2026-04-19 prereg was amended on 2026-04-22 (PREREG §10). The substantive deviations:

(a) Llama-3.1-8B was dropped between prereg and canonical rerun, reducing the panel to two models.

(b) F3b and F3c were discovered after the prereg SHA; all analyses after their discovery are formally exploratory.

(c) The E3 human audit was de-scoped; the metric was renamed "Construct Sensitivity Ratio" → "Contrast Selectivity Ratio" under the pre-committed PREREG §7 fallback. Construct-validity language is retracted from any claim that would have required E3 validation; artifacts `results/audit/`, `ETHICS.md`, and `annotator_consent.md` are removed from the PREREG §9 manifest.

(d) CrowS-Pairs is marked *not_applicable* at the cell level (the PLL scorer reads `choices` while our perturbations edit `question`); no selective/non_selective verdict is issued for CrowS. See Appendix F.

(e) The legacy unpaired bootstrap (`src/ablations.py::bootstrap_csr`) was removed; claim-bearing CIs come from `src/analyze_canonical.py::paired_bootstrap_csr` (item-level paired, PREREG §6.1). See §4 (F4).

(f) Template robustness (PREREG §6.3) was scoped to TruthfulQA and ToxiGen and reported as secondary, not as a gate for the other three benchmarks whose scoring paths have no prompt-template axis.

(g) The `cell_validity.csv` gate set was tightened from 3 columns (2026-04-21) to the full 6-column G1–G4 realisation; mapping to paper G1–G6 is in Appendix G.

Seed stability and placebo specificity (E5, E6). Reported in Appendix F.