BEFORE LLM MEASUREMENTS CAN DISAGREE: AUDITING OUTPUT ADMISSIBILITY ACROSS SYSTEMS

Anonymous authors
Paper under double-blind review

ABSTRACT

Large language models (LLMs) now serve as measurement instruments in social-science and ML pipelines, and their outputs are compared across models and runs to assess reproducibility. Such comparisons presuppose an admissible output from every call. Yet recent work resamples failed calls or drops them before aggregation, so whether each system produced admissible outputs on the same inputs goes untested. We therefore audit output admissibility before agreement with a locally frozen, stop-rule-governed protocol. It treats every failed call as terminal and keeps verified zero extraction, abstention and terminal failure as separate states. The pipeline first extracts atomic stakeholder claims with verbatim evidence and then classifies them into seven frames. Its inputs are 11 masked documents from 8 carbon capture, utilization and storage (CCUS) project clusters. Three open-weight 7–8B checkpoints were each run in three seeded repeats with fixed prompts and decoding settings. The primary estimand contrasts cross-system with within-system Jensen–Shannon divergence of cluster-level frame distributions. The frozen protocol requires every cell of every cluster to be complete and stops at the first terminal failure. The run stopped at its first call. We then executed all remaining calls to characterize the failure flow. Only 23 of 153 extraction calls returned schema-valid output with every claim anchored to verbatim evidence. Common support, all 9 cells complete, held in 0 of 8 clusters, so the primary estimand is not estimable. Most failures were evidence misalignments, and even dropping every non-conforming claim let only 2 of 8 clusters pass extraction. Failures recurred across independently seeded repeats even as responses varied, and their kind differed across systems. Token-level replay traced every unparseable constrained response short of the output-length cap to the constraint adapter. Recoding failures as empty extractions or abstentions gave 7 of 8 clusters apparent common support. The study concerns measurement reproducibility for this finite corpus and these checkpoints. We did not validate outputs against human labels, and the study does not assess semantic accuracy, construct validity or which model is correct. For these checkpoints, prompts and decoder, which inputs yielded an admissible output depended on the system. Admissibility should therefore be audited before cross-system agreement is reported.

1 INTRODUCTION

LLMs can convert unstructured records into variables for social-science and policy analysis (Ziems et al., 2024). Such variables come from a chain of extraction and classification decisions. A project-level distribution of stakeholder frames, for example, is not observed in the source text. A system produces it by identifying relevant claims, assigning each claim to a codebook category and aggregating these assignments. LLMs play the same measurement role inside ML pipelines. As evaluators, they score generations and gate training data (Zheng et al., 2023; Bavaresco et al., 2025; Zhu & Zhang, 2026). They also extract structured records from documents (Tan & D'Souza, 2026). Their labels replace human coding (Gilardi et al., 2023) and feed downstream statistical analyses (Egami et al., 2023).

Work on measurement stability compares outputs across repeated runs and across checkpoints. Neither cross-model agreement nor self-consistency can be assumed (Plaza-del-Arco et al., 2024; Lee et al., 2025). Prompt-sensitivity studies address how outputs change when the task specification

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

changes (Errica et al., 2025; Hua et al., 2025). All of these comparisons presuppose an admissible output from each call. We call an output admissible if it is schema-valid and every claim in it is anchored to verbatim evidence. A recent protocol resamples failed calls (Camuffo et al., 2025), and a study drops them before aggregation (Murin, 2026). A reported agreement then conditions on the inputs each system could process, and these inputs can differ across systems. How invalid outputs and abstentions are handled already determines what an agreement statistic estimates (Rao & Callison-Burch, 2026). Whether every system produces admissible outputs is seldom checked before agreement is computed.

We determine output admissibility before agreement. Throughout, *reproducibility* means agreement across systems and seeded repeats under one fixed protocol. Re-executing a published result with the same code and data is a separate notion (Pineau et al., 2021; Gundersen & Kjensmo, 2018). The audit uses formal records about carbon capture, utilization and storage (CCUS) projects. In this domain, automated discourse measures can serve as inputs to downstream analysis. The eligible corpus contains 8 linked project/asset clusters and 11 documents. Three open-weight checkpoints process the same inputs in three prespecified repeats with fixed prompts. Under the frozen protocol, every failed call is terminal, and verified zero extraction, abstention and terminal failure remain distinct states. A cross-system contrast of cluster-level frame distributions is computed only if every system completes every cell of every cluster. The descriptive audit addresses three questions: whether this contrast is estimable, where and how consistently calls fail, and whether recoding or relaxation restores measurability. In our application, agreement was not measurable.

Contributions. Our main contribution is empirical (Section 5). An executed audit of three open checkpoints on a CCUS corpus found the primary estimand not estimable. Most extraction calls failed the output contract, chiefly on evidence alignment. The failures recurred across repeats and differed in kind across systems. Two secondary contributions accompany this result. For measurement theory and practice, a simple recoding can create apparent completeness at any nonzero failure rate. Treating failed calls as empty extractions or abstentions gave most clusters apparent common support. Under every tested relaxation of the evidence rules, most clusters still failed Stage 1. Verified zero extraction, abstention and terminal failure therefore need separate states, and the resulting missingness could not be assumed ignorable. For audit design, we provide a frozen, stop-rule-governed protocol with call-level diagnostics (Sections 3 and 5). It pairs a cross-system contrast with a within-system repeat reference at the clustered unit. Its state model and stop rule produced the verdict. The contrast was fixed in advance but never computed on common support. Exploratory checks (a validator positive control, a relaxation ladder and a token-level adapter replay) gave a call-level account of the failures. Related designs use stability scores, invalid-output rules, evaluator gates and failure resampling (Barrie et al., 2024; Rao & Callison-Burch, 2026; Zhu & Zhang, 2026; Camuffo et al., 2025). Our design treats a failed call as terminal and conditions the contrast on common support.

We report non-estimability as the result. No part of the protocol was changed after any response was read. Stability, had it been observed, would have described agreement under this protocol, but it would not certify a frame assignment as correct.

2 Related Work

Five lines of work bear on whether LLM outputs can serve as reproducible measurements.

LLM annotation as measurement. Validation practice for LLM annotation remains uneven, and published guidelines call for validating every task against human labels (Desai et al., 2026; Pangakis et al., 2023; Törnberg, 2024). Evaluator quality varies across models and datasets (Bavaresco et al., 2025), and apparent validity depends on how indeterminate ratings are aggregated (Guerdan et al., 2025). Configuration choices and non-random annotation errors can change downstream conclusions (Baumann et al., 2025; Egami et al., 2023; Ashwin et al., 2026). A codebook framework tests basic capabilities before measurement accuracy and finds open-weight models limited in following codebooks zero-shot (Halterman & Keith, 2026). These works target accuracy against human labels or the value of LLMs as assistants to human annotators (Gu et al., 2026). The gap we address is whether every compared system returns an admissible output for the same inputs.

Consistency, prompt variation and nondeterminism. LLM outputs can change across repeated identical calls, equivalent prompts, prompt formatting and inference settings (Reiss, 2023; Liu, 2026; Errica et al., 2025; Hua et al., 2025; Sclar et al., 2024; Atıl et al., 2025; Yuan et al., 2025), and model retirement threatens re-annotation (Cheng et al., 2024). A preregistered audit of hosted evaluators found the instrument unstable although every request ran cleanly (Zhu & Zhang, 2026), and identical calls need not recover identical structure (Sarkar, 2026). Consistency establishes neither accuracy (Tan et al., 2025) nor construct validity (Batzdorfer & Santagiustina, 2026). These studies measure variation among outputs already obtained. Rao & Callison-Burch (2026) show how the handling of invalid outputs changes what agreement estimates. None of them executes a cross-system comparison in which invalid outputs are common and differ by system.

Structured output and constrained decoding. JSON benchmarks document schema violations under unconstrained decoding, including by 7–8B open-weight models (Lu et al., 2025; Shorten et al., 2024). Constrained decoders mask invalid tokens at each step (Geng et al., 2023; Willard & Louf, 2023; Beurer-Kellner et al., 2023; Zheng et al., 2024; Dong et al., 2025), yet compliance varies across engines, schemas, serving layers and model families (Geng et al., 2025; Tang & Zheng, 2026; Kotte, 2026). Constrained decoders can truncate outputs at the token budget or fail to reach acceptance under tokenizer and grammar mismatch (Kato & Tarashima, 2026; Collura et al., 2026). Constraining generation can also distort the model's distribution or lower task quality (Park et al., 2024; Beurer-Kellner et al., 2024; Banerjee et al., 2025; Tam et al., 2024). For small models, constrained decoding removed all structural failures while leaving a semantic gap (Chavan, 2026). Parseability can drive apparent format sensitivity (Mehta, 2026). Format repair can likewise masquerade as self-correction (Chen et al., 2026). These studies report compliance per model, family or engine. The gap is how such failures carry into a cross-system measurement comparison.

Verbatim evidence and attribution. Citation, hallucination and verbatim-quotation studies find frequent gaps between generated claims and their evidence (Gao et al., 2023; Zhang et al., 2025a; Niu et al., 2024; Zhang et al., 2026). Evidence extraction breaks down as relational demands grow (Tan & D'Souza, 2026). Models can be trained to quote verbatim (Zhang et al., 2025b). Whether small open checkpoints meet a strict verbatim contract without such training is not tested in these studies. These studies measure attribution quality, and its bearing on agreement between systems remains open.

Measurement validity and missingness. Measurement theory separates the reliability of a construct's measurement from its validity (Jacobs & Wallach, 2021), and ML evaluation has been recast as a measurement problem (Wallach et al., 2025). Measurement invariance is a logical prerequisite for comparing groups (Vandenberg & Lance, 2000). Restricting a comparison to observed outputs can change its target unless the missingness is ignorable (Rubin, 1976). In our run, missingness was prevalent, system-dependent and recurrent across repeats, so this condition cannot be assumed. Our protocol was frozen locally. It was not preregistered in the sense of van Miltenburg et al. (2021).

Across these lines of work, format compliance is mostly measured per model, and agreement mostly on outputs already obtained. A cross-system agreement is interpretable only if every system returns admissible outputs on the same inputs, so admissibility has to be settled first. The frozen protocol is built around this order.

3 Frozen Audit Protocol

The protocol turns the order of checks motivated above into fixed rules. All definitions in this section are transcribed from the frozen protocol and code. None was chosen after reading any response.

3.1 Design and units

The design first fixes the unit of analysis and uncertainty, the *linked project/asset cluster* $j \in \{1, \ldots, 8\}$, labeled G001–G008. Multiple documents within a cluster contribute to its measurement but do not add independent units. A *system* is one pinned checkpoint with its own tokenizer and chat template. Decoding settings, schemas, constraint adapter and validators are shared by all systems. Every system $s \in \mathcal{S} = \{\text{Qwen}, \text{Mistral}, \text{Granite}\}$ processes each cluster in every repeat

$r \in \{\text{R01}, \text{R02}, \text{R03}\}$, giving 9 cells per cluster. A cluster enters *common support* only if all nine cells are complete. A cell is complete when every required call reaches terminal success and the cell holds classified or verified zero claims. The frozen minimum on common support is 8 clusters, equal to the size of the eligible census, so there is no attrition margin. If any cluster lacks full support, the primary estimate is not computed and the run is reported as insufficient frame. Common support thus gates every comparison.

3.2 Two-stage measurement and the eight-state distribution

Within each cell, two stages and an aggregation step turn text into a frame distribution. Stage 1 extracts explicit atomic stakeholder claims (actor, proposition, exact evidence span) from each masked document chunk. Before Stage 2, claims are deduplicated by a normalized exact key. Stage 2 classifies each unique claim into one of 7 frames (safety and health, place and property, procedural fairness, distrust, climate legitimacy, economic cost, other). It returns a primary frame, an optional secondary frame with an ambiguity flag, a seven-way probability vector and an evidence substring. It may abstain only when the claim is unsupported or is not a stakeholder position.

The state model maps each cell to a frame distribution or a named missing state. For cell (j, s, r), the *primary* representation p_{jsr} is the empirical distribution of schema-valid, non-abstained categorical primary-frame labels. It covers the seven frames plus an eighth no-claim state. A complete cell with a verified empty claim set is represented by the one-hot vector on the no-claim state. A cell whose claims all abstain is *missing* (no classifiable claims). Retrieval failures, invalid outputs and terminal call failures make a cell incomplete and are never mapped to the no-claim state. The primary distribution is conditional on classifiable claims whenever claims exist. The plan requires every cell's state and failure counts to accompany any contrast. A matched probability-vector representation $\tilde{p}_{jsr}$ over the same claims is a sensitivity only. The state model thus keeps empty, abstained and failed cells apart.

3.3 Primary estimand

Computed on complete cells only, the primary estimand contrasts cross-system with within-system variation. With base-2 Jensen–Shannon divergence $\mathrm{JS}(p, q) \in [0, 1]$ (Lin, 1991), define for each cluster

$$D_B(j) = \frac{1}{3} \sum_{\{s,s'\} \subset \mathcal{S}} \frac{1}{9} \sum_{r,r'} \mathrm{JS}(p_{jsr}, p_{js'r'}), \quad D_W(j) = \frac{1}{3} \sum_{s \in \mathcal{S}} \frac{1}{3} \sum_{\{r,r'\}} \mathrm{JS}(p_{jsr}, p_{jsr'}). \tag{1}$$

Here $D_B(j)$ averages 27 comparisons (all 3×3 repeat combinations, equal indices included, for each of three system pairs). $D_W(j)$ averages 9 (three unordered repeat pairs per system). Each component is the mean over its own pairs, so the unequal numbers of pairs do not weight the contrast. The primary estimand gives equal weight to each of the 8 clusters:

$$\Delta_{\mathrm{JS}} = \frac{1}{8} \sum_{j=1}^{8} \{D_B(j) - D_W(j)\}. \tag{2}$$

The contrast is defined only when all 8 clusters are on common support.

3.4 Analysis plan

Before any output, the plan fixed every confirmatory analysis, and any analysis outside it is exploratory. It comprises (a) a finite-corpus description of the estimand, (b) influence diagnostics and (c) a rule under which no uncertainty interval is primary. It adds (d) a matched representation sensitivity with a separate all-valid-record view, (e) stagewise diagnostics with Gwet AC1 (Gwet, 2008) and (f) the full failure-flow matrix. Pairwise system comparisons receive no p-values. Null, negative and not-estimable results are reported without changing systems, prompts, representation or support.

The protocol thus defines in advance what a complete cell is, when a cluster enters the comparison and what is reported when none does. Its use depends on corpus, systems and execution path.

4 EXPERIMENTAL SETUP

The protocol's rules were applied to one corpus and three pinned systems, all fixed before the reported run.

Corpus. Eligibility was fixed before model execution. The gates used project scope, linked-asset membership, source rights and a lifecycle anchor, with a one-year record window censored before any outcome disclosure. Of 13 linked groups in the frozen registry, 8 clusters passed all gates, together comprising 11 openly available government documents. Project, proponent and location names are masked, and documents are given opaque IDs, before any model call. The names of individual officials and of federal agencies are not masked. Informed readers could therefore re-identify some projects.

Systems and decoding. The three checkpoints are Qwen2.5-7B-Instruct (Qwen Team, 2024), Mistral-7B-Instruct-v0.3 (Jiang et al., 2023; Mistral AI, 2025) and Granite-3.1-8B-Instruct (IBM Granite Team, 2025). All are ungated Apache-2.0 models at pinned repository revisions, run in native BF16 without quantization. Pinned open-weight checkpoints keep every call re-executable, whereas hosted models can change or be retired. Prompts are rendered with each model's own chat template (one system and one user message; Mistral's template prepends the system message to the user turn). Decoding is identical across systems (sampling at temperature 0.2, top-p 0.95). Inputs are chunked without truncation, and the 11 documents yield 17 chunks. Chunks are sized so the full rendered Stage-1 prompt fits a common 2,048-token input limit under every tokenizer. Stages 1 and 2 generate at most 1,536 and 512 new tokens, respectively. Seeds are derived per system and call from a SHA-256 hash, so the three systems receive distinct pseudorandom streams. Both stages are constrained at decode time by the same JSON schemas and one adapter shared by all systems, lm-format-enforcer (Gat, 2025). Frozen parse, schema, enum, probability, exact-span and actor-in-span validators then apply.

Failure policy. Only infrastructure failures before a complete response are retried, at most twice with fixed parameters and seed. Invalid JSON, evidence-alignment failures, context overflow and empty or refusal responses are terminal measurement states and trigger no repair prompt. The first terminal failure of a required call makes full common support unattainable. The runner then stops, records the frozen early-stop state and produces no partial comparative output.

Execution path (deviations from the frozen release chain). The protocol specified execution through a containerized, hash-verified release chain (`exact-v`N). After the grammar-constrained amendment, attempts to run the chain stopped for operational reasons before any model call. The reported run used a direct runner, which imported the frozen runner module unchanged (source hash prefix `d905cb64879c`). It called the module's functions for generation, validation, cell construction and early stop. It bypassed the chain's gate reports and frozen-input verification, and no frozen reference hash of the weights existed. To keep a cell running, its parallel workers passed a failed call to the cell function as an empty or abstained placeholder. Frozen cell states, however, used the terminal failure on disk. We executed the calls beyond the frozen stop to report the failure flow of plan item (f). Diagnostic files written automatically on the all-valid-record view were not inspected, reported or distributed. The scientific specification is unchanged.

With these elements fixed, outcomes can be compared across systems, seeds and chunks.

5 RESULTS

Apart from the frozen verdict and plan items (d) and (f), every analysis in this section is exploratory and descriptive.

5.1 THE PRIMARY ESTIMAND IS NOT ESTIMABLE

The frozen runner reached its early-stop state at the first required call in run order (Qwen, R01, Stage 1, D011/C0001), which ended in a schema failure (output-length cap reached). Under the no-attrition rule, this single terminal failure already makes full common support unattainable. By the

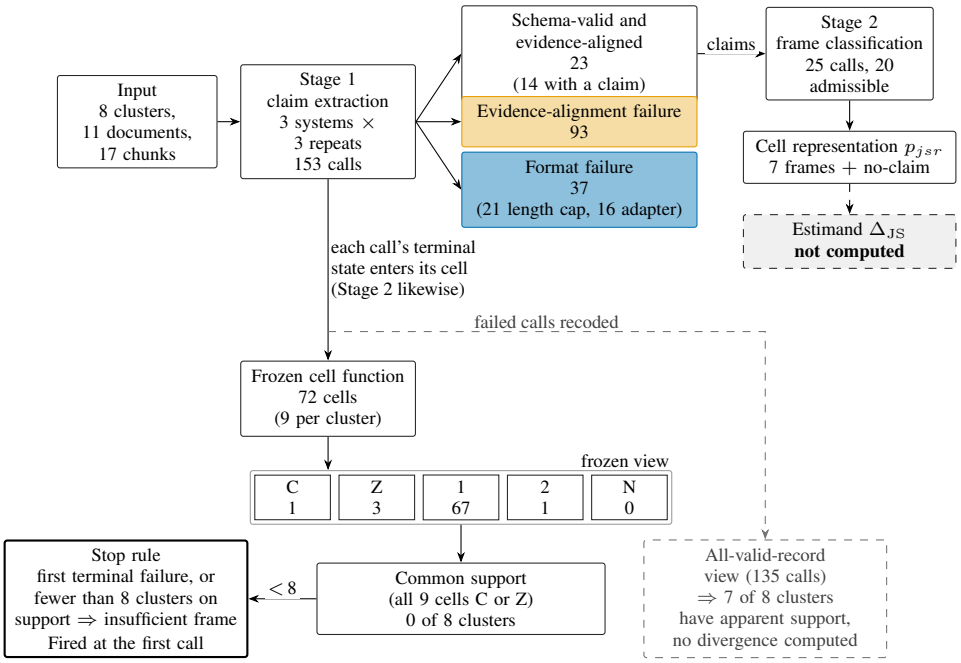

Figure 1: Under the frozen cell function, common support held in 0 of 8 clusters, so Δ_{JS} was not computed. Orange and blue mark the two Stage-1 failure kinds. Cell codes are C (complete, classified), Z (complete, zero claims), 1 and 2 (incomplete at Stage 1 or 2) and N (no classifiable claims).

protocol's stop rules, no comparative result (D_B, D_W or Δ_{JS}, pooled or per cluster) was computed, inspected or reported. To characterize the failure flow required by plan item (f), we nevertheless executed every call of the 3×3 matrix. Figure 1 traces the complete call record through the frozen cell function. Common support held in 0 of 8 clusters. Mistral had no complete cell in any cluster (0 of 24). No cluster could therefore satisfy a rule requiring all three systems, whatever the attrition margin or the order of execution. The verdict thus rests on incomplete cells.

5.2 Most extraction calls fail the frozen output contract

Of the incomplete cells, 67 failed at Stage 1 and 1 at Stage 2, so the verdict traces back to individual extraction calls. Of 153 Stage-1 calls, 23 (15%) returned output both schema-valid and evidence-aligned (Table 1). All 9 successful Qwen calls returned verified empty claim sets. Only 14 of the 153 calls therefore produced at least one admissible claim. Evidence misalignment was the dominant failure kind (93 calls, against 37 format failures). Every alignment-failed call returned parseable JSON with a claim whose span was not verbatim in the chunk or whose actor mention fell outside it. The mix differed by system (Table 1). Qwen mostly returned unresolvable spans, Mistral mostly placed actors outside the span, and Granite split between the two. Under the grammar constraint, 21 responses reached the output-length cap, 10 of them with a repetitive field. Another 16 responses, all from Qwen, ended unparseable without reaching the cap. Most failed calls thus returned parseable JSON with misaligned evidence.

5.3 Failures are reproducible within systems and differ across systems

Because each repeat uses an independently derived seed, the repeats probe whether these failures are sampling noise. For a given system and chunk, failures mostly recurred across repeats (Figure 2). For 14, 12 and 14 of the 17 chunks, all three repeats of Qwen, Mistral and Granite, respectively, failed. Mixed outcomes occurred on 0, 4 and 2 chunks, respectively. All three repeats agreed in 45 of 51 (system, chunk) groups, against 31.2 expected if outcomes were independent of the chunk. Recurrence did not come from identical text. Only 6, 2 and 1 of the 17 chunks produced byte-

Table 1: Most Stage-1 calls fail the frozen output contract, and most failures are evidence-alignment failures. The table gives the terminal outcome of every executed call under the frozen contract, by system. Stage 1 covers 17 chunks × 3 repeats, and Stage 2 runs only on claims from successful Stage-1 calls. Italic headings are frozen states. A format failure is a schema failure with invalid JSON, and an alignment failure is an evidence-alignment failure. A probability-sum failure is a schema failure with valid JSON. Rows beneath the headings are post-hoc subcategories coded by script from each call record. Early termination and control character together are the 16 adapter failures. A length-cap failure uses at least 1,520 of 1,536 new tokens after re-tokenization. In an early termination, the response ends before the JSON object closes. A span does not resolve if it is absent from the chunk or its occurrence index exceeds its count.

Terminal outcome	Qwen	Mistral	Granite
Stage 1 (claim extraction)			
Calls	51	51	51
Schema-valid and evidence-aligned	9	8	6
Format failures (invalid JSON)			
length cap reached	4	5	12
early termination	13	0	0
unescaped control character	3	0	0
Evidence-alignment failures			
span does not resolve in chunk	20	7	15
actor mention not inside span	2	31	18
Stage 2 (frame classification)			
Calls	0	10	15
Schema-valid and evidence-aligned	0	7	13
Failures			
evidence words not in claim or span	0	3	1
probabilities do not sum to one	0	0	1

identical responses across the three repeats. At temperature 0.2, seed variation thus appeared to explain little of the failure pattern. Across systems, by contrast, binary outcomes (success or failure) agreed in 33 of 51 (chunk, repeat) pairs, close to the 31.4 expected. All of these were joint failures. Joint successes across systems concentrated on a few inputs (Figure 2).

5.4 Neither the validator nor the adapter explains the verdict

Stable failures could still originate in the validator or the adapter. A positive control tested whether the validator would reject conforming claims. In this control, each of 71 synthetic claims takes a chunk sentence verbatim as its span. Its actor mention is the span's first two words. The frozen validator accepted 71 of the 71. The Stage-1 prompt states every rule the validator checks, so the failures cannot be traced to unstated requirements. Superficial character mismatches did not explain the alignment failures either. We re-matched every claim in the 93 alignment-failed calls after Unicode normalization, case-folding, quote unification and whitespace collapse. This rescued 1 call. Every span still absent after normalization shared at least 90% of its words with the chunk, consistent with paraphrase or recombination. Normalization rescued so few calls partly because a single failing claim fails the whole call. At the claim level, 76 of 343 claims in parseable Stage-1 outputs (22%) passed the frozen rules.

The 16 adapter failures had an identifiable mechanism. Token-by-token replay through the frozen adapter showed its parser state collapsing at a span-text value in all 13 early terminations. These terminations fell on 5 chunks, and on 4 of them all three repeats terminated early. In the 3 control-character cases, the adapter admitted a single token comprising a quote, a comma and a raw newline. It followed a token ending in a backslash. All of these failures occurred with Qwen. Part of the cross-system difference in failure kind thus reflected an interaction of the shared adapter with one system. It likely arose partly through Qwen's tokenizer. Counting all 16 adapter failures as successful calls still left 0 of 8 clusters on common support under the frozen evidence rules. Counting all 37 format failures as successes also left 0, so format failures did not drive the verdict.

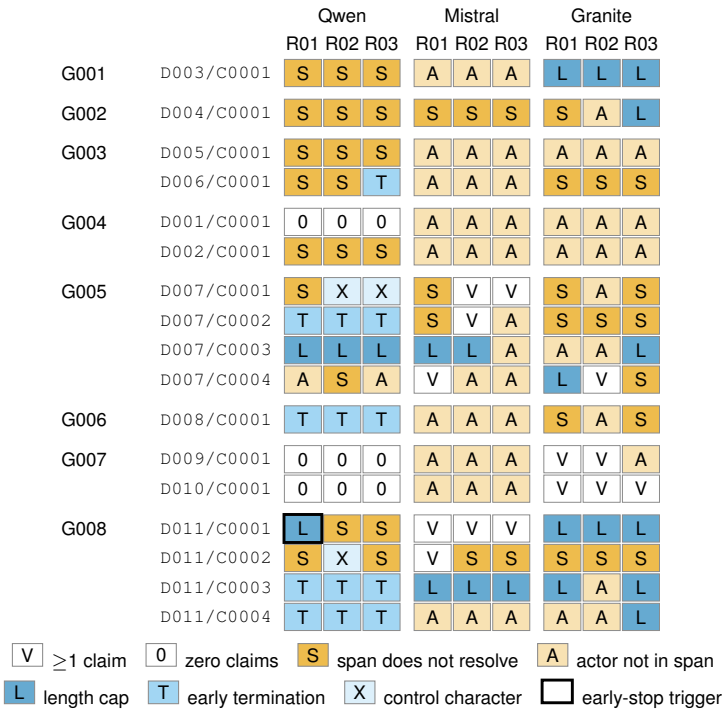

Figure 2: Outcome codes recur across a system's repeats on the same chunk but rarely across systems. One code holds in all 3 repeats for 33 of 51 (system, chunk) groups. Across all 3 systems, the same code holds for only 3 of 51 (chunk, repeat) pairs. The bold cell marks the early-stop call.

5.5 An all-valid-record view does not restore the comparison

A looser reading of the records might still restore the comparison. Plan item (d) specifies an all-valid-record view. It keeps successful outputs from cells later rendered incomplete, for failure transparency only. Keeping these outputs requires deciding what each failed call contributes, so we recoded the 135 failed calls. Each failed Stage-1 call contributed no claims, and each failed Stage-2 call became an abstention. Under this view, 62 of 72 cells became zero-claim cells, 9 contained a classified claim and 1 had no classifiable claim. In 7 of 8 clusters, all nine cells would be non-missing. Frame distributions here would be dominated by the no-claim state produced by failures. The state model forbids this conflation, so no divergence is computed on the view. This view documents the failures without restoring an admissible comparison.

5.6 No tested relaxation restores full common support

The evidence contract can also be relaxed on the recorded outputs. Outside the frozen plan, we re-validated the recorded Stage-1 outputs under progressively weaker rules. Relaxed claims were never classified. The first rung uses normalized span matching. The second rung also ignores the occurrence index, and the third also drops the actor-in-span requirement. The last rung drops every non-conforming claim and keeps the call, so only parseability is required. Passing calls went from 23 under the frozen rule to 24, 24, 70 and 116. Clusters whose nine cells pass Stage 1 went from 0 to 0, 0, 1 and 2. Format failures, which no relaxation of the evidence rules can rescue, blocked the remaining clusters. With adapter failures also treated as parseable at the last rung, 4 of 8 clusters would pass Stage 1. The protocol excludes fuzzy and semantic matching, and no agreement statistic is computed on relaxed outputs.

Taken together, the results locate the verdict in individual calls. Most extraction calls missed the evidence contract. The failures were mostly stable within systems and different across them, and no tested relaxation restored full common support. Only recoding failures as empty extractions or abstentions gave most clusters apparent common support.

6 DISCUSSION

Admissibility precedes agreement. Repairing or re-prompting inadmissible outputs, then dropping the calls still failing, can conflate disagreement about frames with disagreement about which evidence exists. For these checkpoints, prompts and decoder, this outcome has three implications. First, admissibility appears to be a first-order component of LLM measurement reproducibility, since its failure here precluded the estimand. Like measurement invariance, admissibility is a prerequisite for comparing systems (Vandenberg & Lance, 2000). Zhu & Zhang (2026) likewise halted at instrument validation, but with clean execution and unstable readouts. Here, admissibility failed before any agreement estimate existed. Second, restricting agreement to admissible outputs assumes ignorable missingness (Rubin, 1976), and failure tied to system and input means this assumption cannot be made. After failed cases are excluded, an evaluator's accuracy over all cases is bounded only within a worst-case interval as wide as the excluded fraction (Rao & Callison-Burch, 2026). Agreement computed on admissible calls alone faces a similar limit. Third, admissibility should be reported before and alongside agreement statistics. In clinical extraction, cross-prompt disagreement concentrated on the difference between a documented *no* and the absence of documentation (Murin, 2026). Agreement statistics can thus depend on how empty readings are coded. Our state model therefore keeps verified zero extraction apart from abstention and failure. Recoding failed calls can create apparent support at any nonzero failure rate, so the third point does not rest on the rates observed here.

Constrained decoding did not remove format failures, and a syntax check would have missed alignment failures. In one benchmark of smaller models, constrained decoding with other engines removed all structural failures (Chavan, 2026). Here, by contrast, some constrained responses still failed to parse, consistent with finite token budgets and tokenizer and grammar mismatch (Kato & Tarashima, 2026; Collura et al., 2026). A syntax-only check would have passed most failed extraction calls, since their JSON parsed.

Protocol history and execution. The protocol was frozen locally (SHA-256 manifests) before its first model output. An earlier unconstrained run failed on invalid JSON and is not pooled. A later amendment, written without reading the run's responses, added grammar-constrained decoding and automatic enforcement of the existing stop. The deviations from the release chain (Section 4) do not change the frozen verdict, which the frozen code recomputes from the call records.

The evidence therefore supports a statement about admissibility under this protocol. It leaves agreement on frames unmeasured.

7 CONCLUSION

The records of all extraction calls and their terminal states support one central finding. Before LLM measurements can disagree, every system must produce admissible outputs, and for these checkpoints most calls did not. Which inputs yielded an admissible output depended on the system, and recoding the failed calls would have hidden this dependence behind apparent completeness.

Limitations. The most important limitation is the strict verbatim-evidence contract applied to small open checkpoints. Larger models, other prompts, another decoding backend or a looser contract might yield more admissible outputs. The failure rates and system differences may therefore not transfer beyond this setting. The central finding still holds within it, since non-estimability needs only one unsupported cluster and the contract was fixed before any output. Frame assignments were not validated against human labels.

Within these limits, an admissibility audit should precede any agreement statistic in LLM measurement. Recording every call's terminal state before aggregation shows whether cross-system agreement can be measured at all. In this audit, it could not.

Ethics Statement

The corpus consists of publicly issued government records about infrastructure projects. No human subjects were recruited, and no new annotation was collected. Project, proponent and location names were masked before model processing, but names of officials and federal agencies were not. The study does not rank models by task quality or rank projects and makes no claim about the correctness of any frame assignment. Agreement among LLMs should not serve as evidence of validity in policy assessments. The Reproducibility Statement lists each withheld item and why it is withheld.

Reproducibility Statement

Appendix A gives the full protocol, Appendix B the verbatim prompts, and Appendix C the pinned models, decoding settings and seeds. Appendix D details the claim-level failures and adapter replay, and Appendix E gives background. The anonymous supplement holds the runner, the analysis code (entry point `make_results.py`), the launch scripts and one redacted record per executed call. A README and a datasheet describe the package. A standard-library script recomputes the reported values, table cells and figure cells from these records and names each value it cannot recompute. Raw responses and parsed claims are withheld because they quote the source documents. The masked source texts are withheld because their republishing conditions conflict with masking. Three files with unmasked names are withheld to preserve masking. The protocol document holds internal project records, and the diagnostic files were never inspected, so neither is shipped.

AI Use Statement

The three LLMs studied are objects of measurement only. We used LLM-based coding agents to assist with study design, code, analysis and writing. Scripts they wrote produced one synthetic smoke-test passage and the synthetic control data. No proofs or translation were involved. We reviewed all AI-assisted work and take responsibility for the final content.

References

Julian Ashwin, Aditya Chhabra, and Vijayendra Rao. Using large language models for qualitative analysis can introduce serious bias. *Sociological Methods & Research*, 55(3):795–839, 2026. doi: 10.1177/00491241251338246.

Berk Atıl, Sarp Aykent, Alexa Chittams, Lisheng Fu, Rebecca J. Passonneau, Evan Radcliffe, Guru Rajan Rajagopal, Adam Sloan, Tomasz Tudrej, Ferhan Ture, Zhe Wu, Lixinyu Xu, and Breck Baldwin. Non-determinism of "deterministic" LLM system settings in hosted environments. In *Proceedings of the 5th Workshop on Evaluation and Comparison of NLP Systems*, pp. 135–148. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.eval4nlp-1.12.

Debangshu Banerjee, Tarun Suresh, Shubham Ugare, Sasa Misailovic, and Gagandeep Singh. CRANE: Reasoning with constrained LLM generation. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 2836–2857, 2025.

Christopher Barrie, Elli Palaiologou, and Petter Törnberg. Prompt stability scoring for text annotation with large language models. arXiv preprint arXiv:2407.02039, 2024.

Veronika Batzdorfer and Carlo Romano Marcello Alessandro Santagiustina. Reproducibility is not construct validity: LLM measurement of institutionally situated communication. arXiv preprint arXiv:2609.19866, 2026.

Joachim Baumann, Paul Röttger, Aleksandra Urman, Albert Wendsjö, Flor Miriam Plaza-del-Arco, Johannes B. Gruber, and Dirk Hovy. Large language model hacking: Quantifying the hidden risks of using LLMs for text annotation. arXiv preprint arXiv:2509.08825, 2025.

Anna Bavaresco, Raffaella Bernardi, Leonardo Bertolazzi, Desmond Elliott, Raquel Fernández, Albert Gatt, Esam Ghaleb, Mario Giulianelli, Michael Hanna, Alexander Koller, Andre Martins, Philipp Mondorf, Vera Neplenbroek, Sandro Pezzelle, Barbara Plank, David Schlangen, Alessandro Suglia, Aditya K Surikuchi, Ece Takmaz, and Alberto Testoni. LLMs instead of human judges? A large scale empirical study across 20 NLP evaluation tasks. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 238–255. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.acl-short.20.

Andrew M. Bean, Ryan Othniel Kearns, Angelika Romanou, Franziska Sofia Hafner, Harry Mayne, Jan Batzner, Negar Foroutan Eghlidi, Chris Schmitz, Karolina Korgul, Hunar Batra, Oishi Deb, Emma Beharry, Cornelius Emde, Thomas Foster, Anna Gausen, María Grandury, Sophia Han, Valentin Hofmann, Lujain Ibrahim, Hazel Kim, Hannah Rose Kirk, Fangru Lin, Gabrielle Liu, Lennart Luettgau, Jabez Magomere, Jonathan Rystrøm, Anna Sotnikova, Yushi Yang, Yilun Zhao, Adel Bibi, Antoine Bosselut, Ronald Clark, Arman Cohan, Jakob Foerster, Yarin Gal, Scott Hale, Deborah Raji, Christopher Summerfield, Philip Torr, Cozmin Ududec, Luc Rocher, and Adam Mahdi. Measuring what matters: Construct validity in large language model benchmarks. In *Advances in Neural Information Processing Systems*, volume 38, pp. 19868–19949, 2025. doi: 10.52202/085713-0590.

Luca Beurer-Kellner, Marc Fischer, and Martin Vechev. Prompting is programming: A query language for large language models. *Proceedings of the ACM on Programming Languages*, 7(PLDI): 1946–1969, 2023. doi: 10.1145/3591300.

Luca Beurer-Kellner, Marc Fischer, and Martin Vechev. Guiding LLMs the right way: Fast, non-invasive constrained generation. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 3658–3673, 2024. URL https://proceedings.mlr.press/v235/beurer-kellner24a.html.

Samuel R. Bowman and George Dahl. What will it take to fix benchmarking in natural language understanding? In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 4843–4855, 2021. doi: 10.18653/v1/2021.naacl-main.385.

Arnaldo Camuffo, Alfonso Gambardella, Saeid Kazemi, Jakub Malachowski, and Abhinav Pandey. Variance-aware LLM annotation for strategy research: Sources, diagnostics, and a protocol for reliable measurement. arXiv preprint arXiv:2601.02370, 2025.

Akash Chavan. Constrained decoding eliminates structural failures in small LLMs but reveals a scale-dependent semantic gap. arXiv preprint arXiv:2609.23742, 2026.

Lingjiao Chen, Matei Zaharia, and James Zou. How is ChatGPT's behavior changing over time? *Harvard Data Science Review*, 6(2), 2024. doi: 10.1162/99608f92.5317da47.

Mingguang Chen, Bo Qu, and Licheng Wang. The calibration floor: Format repair can masquerade as self-correction at small-to-mid scale. arXiv preprint arXiv:2608.04355, 2026.

Xiang Cheng, Raveesh Mayya, and João Sedoc. To err is human; to annotate, SILICON? Toward robust reproducibility in LLM annotation. arXiv preprint arXiv:2412.14461, 2024.

Vincenzo Collura, Karim Tit, Eleonora Giunchiglia, Mike Papadakis, and Maxime Cordy. Stay within your bounds: Distance-guided decoding for guaranteed context-free grammar compliance. arXiv preprint arXiv:2608.28229, 2026. To appear in Findings of EMNLP 2026.

Meera Desai, Dallas Card, and Abigail Z. Jacobs. Validating LLMs in social science: Epistemic threats and emerging norms. arXiv preprint arXiv:2607.07915, 2026.

Yixin Dong, Charlie F. Ruan, Yaxing Cai, Ziyi Xu, Yilong Zhao, Ruihang Lai, and Tianqi Chen. XGrammar: Flexible and efficient structured generation engine for large language models. In *Proceedings of Machine Learning and Systems*, volume 7, 2025. URL https://proceedings.mlsys.org/paper_files/paper/2025/hash/5c20ca4b0b20b0bd2f1d839dc605e70f-Abstract-Conference.html.

Naoki Egami, Musashi Hinck, Brandon M. Stewart, and Hanying Wei. Using imperfect surrogates for downstream inference: Design-based supervised learning for social science applications of large language models. In *Advances in Neural Information Processing Systems*, volume 36, pp. 68589–68601, 2023. doi: 10.52202/075280-3000.

Federico Errica, Davide Sanvito, Giuseppe Siracusano, and Roberto Bifulco. What did I do wrong? Quantifying LLMs' sensitivity and consistency to prompt engineering. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 1543–1558, 2025. doi: 10.18653/v1/2025.naacl-long.73.

Tianyu Gao, Howard Yen, Jiatong Yu, and Danqi Chen. Enabling large language models to generate text with citations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 6465–6488, 2023. doi: 10.18653/v1/2023.emnlp-main.398.

Noam Gat. lm-format-enforcer. Software, version 0.11.3, https://github.com/noamgat/lm-format-enforcer, 2025.

Saibo Geng, Martin Josifoski, Maxime Peyrard, and Robert West. Grammar-constrained decoding for structured NLP tasks without finetuning. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 10932–10952, 2023. doi: 10.18653/v1/2023.emnlp-main.674.

Saibo Geng, Hudson Cooper, Michał Moskal, Samuel Jenkins, Julian Berman, Nathan Ranchin, Robert West, Eric Horvitz, and Harsha Nori. JSONSchemaBench: A rigorous benchmark of structured outputs for language models. arXiv preprint arXiv:2501.10868, 2025.

Fabrizio Gilardi, Meysam Alizadeh, and Maël Kubli. ChatGPT outperforms crowd workers for text-annotation tasks. *Proceedings of the National Academy of Sciences*, 120(30):e2305016120, 2023. doi: 10.1073/pnas.2305016120.

Robert M. Groves. Nonresponse rates and nonresponse bias in household surveys. *Public Opinion Quarterly*, 70(5):646–675, 2006. doi: 10.1093/poq/nfl033.

Feng Gu, Zongxia Li, Carlos R. Colon, Benjamin Evans, Ishani Mondal, and Jordan Lee Boyd-Graber. Large language models are effective human annotation assistants, but not good independent annotators. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 71–89, 2026. doi: 10.18653/v1/2026.findings-acl.4.

Luke Guerdan, Solon Barocas, Kenneth Holstein, Hanna Wallach, Zhiwei Steven Wu, and Alexandra Chouldechova. Validating LLM-as-a-judge systems under rating indeterminacy. In *Advances in Neural Information Processing Systems*, volume 38, pp. 124475–124543, 2025. doi: 10.52202/085713-3751.

Odd Erik Gundersen and Sigbjørn Kjensmo. State of the art: Reproducibility in artificial intelligence. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018. doi: 10.1609/aaai.v32i1.11503.

Kilem Li Gwet. Computing inter-rater reliability and its variance in the presence of high agreement. *British Journal of Mathematical and Statistical Psychology*, 61(1):29–48, 2008. doi: 10.1348/000711006X126600.

Andrew Halterman and Katherine A. Keith. Codebook LLMs: Evaluating LLMs as measurement tools for political science concepts. *Political Analysis*, 34(2):188–204, 2026. doi: 10.1017/pan.2025.10017.

Andong Hua, Kenan Tang, Chenhe Gu, Jindong Gu, Eric Wong, and Yao Qin. Flaw or artifact? Rethinking prompt sensitivity in evaluating LLMs. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 19889–19899, 2025. doi: 10.18653/v1/2025.emnlp-main.1006.

Ben Hutchinson, Negar Rostamzadeh, Christina Greer, Katherine Heller, and Vinodkumar Prabhakaran. Evaluation gaps in machine learning practice. In *Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency (FAccT)*, pp. 1859–1876, 2022. doi: 10.1145/3531146.3533233.

IBM Granite Team. granite-3.1-8b-instruct model card. https://huggingface.co/ibm-granite/granite-3.1-8b-instruct/tree/4009206d5fc95d2e65a7b7633e159d6e97e25d35, 2025. Hugging Face repository, revision 4009206d.

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

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mistral 7B. arXiv preprint arXiv:2310.06825, 2023.

Yoshio Kato and Shuhei Tarashima. TruncProof: A guardrail for LLM-based JSON generation under token-length constraints. arXiv preprint arXiv:2605.13076, 2026. Accepted at IJCNN 2026.

Varun Kotte. PromptPort: A reliability layer for cross-model structured extraction. arXiv preprint arXiv:2601.06151, 2026.

Noah Lee, Jiwoo Hong, and James Thorne. Evaluating the consistency of LLM evaluators. In *Proceedings of the 31st International Conference on Computational Linguistics*, pp. 10650–10659, 2025. URL https://aclanthology.org/2025.coling-main.710/.

Jianhua Lin. Divergence measures based on the Shannon entropy. *IEEE Transactions on Information Theory*, 37(1):145–151, 1991. doi: 10.1109/18.61115.

Jingyuan Liu. What is actually being annotated? Inter-Prompt Reliability as a measurement problem in LLM-based social science labeling. arXiv preprint arXiv:2604.16413, 2026.

Yaxi Lu, Haolun Li, Xin Cong, Zhong Zhang, Yesai Wu, Yankai Lin, Zhiyuan Liu, Fangming Liu, and Maosong Sun. Learning to generate structured output with schema reinforcement learning. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 4905–4918. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.acl-long.243.

Deep Pankajbhai Mehta. Format Sensitivity Index: Token-controlled prompt wrapper robustness and schema compliance in LLM benchmarking. arXiv preprint arXiv:2607.09665, 2026.

Samuel Messick. 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.

Mistral AI. Mistral-7B-Instruct-v0.3 model card. https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.3/tree/c170c708c41dac9275d15a8fff4eca08d52bab71, 2025. Hugging Face repository, revision c170c708.

Martin Murin. Measuring the sensitivity of LLM-based structured extraction to prompt, model, and schema choices in clinical discharge summaries. arXiv preprint arXiv:2606.05970, 2026.

Cheng Niu, Yuanhao Wu, Juno Zhu, Siliang Xu, KaShun Shum, Randy Zhong, Juntong Song, and Tong Zhang. RAGTruth: A hallucination corpus for developing trustworthy retrieval-augmented language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 10862–10878, 2024. doi: 10.18653/v1/2024.acl-long.585.

Nicholas Pangakis, Samuel Wolken, and Neil Fasching. Automated annotation with generative AI requires validation. arXiv preprint arXiv:2306.00176, 2023.

Kanghee Park, Jiayu Wang, Taylor Berg-Kirkpatrick, Nadia Polikarpova, and Loris D'Antoni. Grammar-aligned decoding. In *Advances in Neural Information Processing Systems*, volume 37, pp. 24547–24568, 2024. doi: 10.52202/079017-0774.

Joelle Pineau, Philippe Vincent-Lamarre, Koustuv Sinha, Vincent Larivière, Alina Beygelzimer, Florence d'Alché-Buc, Emily Fox, and Hugo Larochelle. Improving reproducibility in machine learning research (a report from the NeurIPS 2019 Reproducibility Program). *Journal of Machine Learning Research*, 22(164):1–20, 2021. URL https://jmlr.org/papers/v22/20-303.html.

Flor Miriam Plaza-del-Arco, Debora Nozza, and Dirk Hovy. Wisdom of instruction-tuned language model crowds. Exploring model label variation. In *Proceedings of the 3rd Workshop on Perspectivist Approaches to NLP (NLPerspectives) @ LREC-COLING 2024*, pp. 19–30, 2024. URL https://aclanthology.org/2024.nlperspectives-1.2/.

Qwen Team. Qwen2.5 technical report. arXiv preprint arXiv:2412.15115, 2024.

Delip Rao and Chris Callison-Burch. Agreement metrics for LLM-as-judge evaluation: What to report and why. arXiv preprint arXiv:2606.00093, 2026.

Michael V. Reiss. Testing the reliability of ChatGPT for text annotation and classification: A cautionary remark. arXiv preprint arXiv:2304.11085, 2023.

Donald B. Rubin. Inference and missing data. *Biometrika*, 63(3):581–592, 1976. doi: 10.1093/biomet/63.3.581.

Dipankar Sarkar. How reproducible are evaluation conclusions? A self-audit of LLM-inferred prompt structure. arXiv preprint arXiv:2609.30074, 2026.

Melanie Sclar, Yejin Choi, Yulia Tsvetkov, and Alane Suhr. 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*, 2024.

Connor Shorten, Charles Pierse, Thomas Benjamin Smith, Erika Cardenas, Akanksha Sharma, John Trengrove, and Bob van Luijt. StructuredRAG: JSON response formatting with large language models. arXiv preprint arXiv:2408.11061, 2024.

Arjun Subramonian, Xingdi Yuan, Hal Daumé III, and Su Lin Blodgett. It takes two to tango: Navigating conceptualizations of NLP tasks and measurements of performance. In *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 3234–3279, 2023. doi: 10.18653/v1/2023.findings-acl.202.

Zhi Rui Tam, Cheng-Kuang Wu, Yi-Lin Tsai, Chieh-Yen Lin, Hung-yi Lee, and Yun-Nung Chen. Let me speak freely? A study on the impact of format restrictions on large language model performance. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pp. 1218–1236. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.emnlp-industry.91.

Hexiang Tan, Fei Sun, Sha Liu, Du Su, Qi Cao, Xin Chen, Jingang Wang, Xunliang Cai, Yuanzhuo Wang, Huawei Shen, and Xueqi Cheng. Too consistent to detect: A study of self-consistent errors in LLMs. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 4755–4765, 2025. doi: 10.18653/v1/2025.emnlp-main.238.

Zhiyin Tan and Jennifer D'Souza. Diagnosing structural failures in LLM-based evidence extraction for meta-analysis. In *Proceedings of the 22nd Conference on Information and Research Science Connecting to Digital and Library Science (IRCDL 2026)*, volume 4240 of *CEUR Workshop Proceedings*, 2026. URL https://ceur-ws.org/Vol-4240/paper52.pdf.

Lijuan Tang and Yuemeng Zheng. Measuring the serving stack instead of the model: Hidden confounds in local tool-use evaluation. arXiv preprint arXiv:2609.26693, 2026. Accepted at the 2nd REALM Workshop at EMNLP 2026.

Petter Törnberg. Best practices for text annotation with large language models. *Sociologica*, 18(2): 67–85, 2024. doi: 10.6092/issn.1971-8853/19461.

Emiel van Miltenburg, Chris van der Lee, and Emiel Krahmer. Preregistering NLP research. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 613–623, 2021. doi: 10.18653/v1/2021.naacl-main.51.

Robert J. Vandenberg and Charles E. Lance. A review and synthesis of the measurement invariance literature: Suggestions, practices, and recommendations for organizational research. *Organizational Research Methods*, 3(1):4–70, 2000. doi: 10.1177/109442810031002.

Hanna Wallach, Meera Desai, A. Feder Cooper, Angelina Wang, Chad Atalla, Solon Barocas, Su Lin Blodgett, Alexandra Chouldechova, Emily Corvi, P. Alex Dow, Jean Garcia-Gathright, Alexandra Olteanu, Nicholas J. Pangakis, Stefanie Reed, Emily Sheng, Dan Vann, Jennifer Wortman Vaughan, Matthew Vogel, Hannah Washington, and Abigail Z. Jacobs. Position: Evaluating generative AI systems is a social science measurement challenge. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 82232–82251, 2025. URL https://proceedings.mlr.press/v267/wallach25a.html.

Bingbing Wen, Jihan Yao, Shangbin Feng, Chenjun Xu, Yulia Tsvetkov, Bill Howe, and Lucy Lu Wang. Know your limits: A survey of abstention in large language models. *Transactions of the Association for Computational Linguistics*, 13:529–556, 2025. doi: 10.1162/tacl_a_00754.

Brandon T. Willard and Rémi Louf. Efficient guided generation for large language models. arXiv preprint arXiv:2307.09702, 2023.

Jiayi Yuan, Hao Li, Xinheng Ding, Wenya Xie, Yu-Jhe Li, Wentian Zhao, Kun Wan, Jing Shi, Xia Hu, and Zirui Liu. Understanding and mitigating numerical sources of nondeterminism in LLM inference. In *Advances in Neural Information Processing Systems*, volume 38, pp. 188045–188077, 2025. doi: 10.52202/085713-5653.

Jiajie Zhang, Yushi Bai, Xin Lv, Wanjun Gu, Danqing Liu, Minhao Zou, Shulin Cao, Lei Hou, Yuxiao Dong, Ling Feng, and Juanzi Li. LongCite: Enabling LLMs to generate fine-grained citations in long-context QA. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 5098–5122, 2025a. doi: 10.18653/v1/2025.findings-acl.264.

Jiashuo Zhang, Yuling Chen, Yvonne Commodore-Mensah, and Michael Oberst. Verifiable by construction: Claim-level evaluation of verbatim citation in clinical question answering. arXiv preprint arXiv:2609.15964, 2026.

Jingyu Zhang, Marc Marone, Tianjian Li, Benjamin Van Durme, and Daniel Khashabi. Verifiable by design: Aligning language models to quote from pre-training data. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 3748–3768, 2025b. doi: 10.18653/v1/2025.naacl-long.191.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, Hao Zhang, Joseph Gonzalez, and Ion Stoica. Judging LLM-as-a-judge with MT-Bench and Chatbot Arena. In *Advances in Neural Information Processing Systems*, volume 36, pp. 46595–46623, 2023. doi: 10.52202/075280-2020.

Lianmin Zheng, Liangsheng Yin, Zhiqiang Xie, Chuyue Sun, Jeff Huang, Cody Hao Yu, Shiyi Cao, Christos Kozyrakis, Ion Stoica, Joseph Gonzalez, Clark Barrett, and Ying Sheng. SGLang: Efficient execution of structured language model programs. In *Advances in Neural Information Processing Systems*, volume 37, pp. 62557–62583, 2024. doi: 10.52202/079017-2000.

Haoyuan Zhu and Jie Zhang. Clean engineering, unstable measurement: A preregistered reliability failure of black-box LLM observers on shared endpoints. arXiv preprint arXiv:2609.04198, 2026.

Caleb Ziems, William Held, Omar Shaikh, Jiaao Chen, Zhehao Zhang, and Diyi Yang. Can large language models transform computational social science? *Computational Linguistics*, 50(1): 237–291, 2024. doi: 10.1162/coli_a_00502.

A Full Protocol Details

This appendix gives the frozen specification in full, followed by the operational record.

A.1 Population, eligibility and inputs

The population rules determine which documents could enter any cell. The primary population comprises CCUS projects selected without regard to controversy and passing the scope, linked-asset, source-rights and anchor gates. The projects are identified from publicly issued official records. The lifecycle rule (`formal_development_commitment_v1`) permits three anchor subtypes: regulatory entry, binding/authorized commitment, and formal proponent deployment commitment. Known earlier records, scope conflict, dates not resolved to the day, retrospective-only evidence and missing sources trigger a hold state. In this state the anchor stays unresolved, and the record is treated as missing. No imputation or outcome-specific substitute applies. Anchor subtype is recorded as metadata. A government page covering several explicitly titled projects may contribute a project-specific subsection. The subsection is included only if its start and end markers are unique and frozen before execution. It must also pass a check for text from other projects. General multi-project prose is excluded. Missing, title-only, structural-zero and thin-discourse documents are recorded as distinct states. A missing cluster or document is not backfilled from another source layer or a later document. A rights-driven amendment made before any real model output retained the original nine groups. It added four projects from the original candidate list under unchanged rules, and exactly 8 clusters pass all document gates. Clusters comprise 1 to 4 chunks. The frozen minimum of 8 passed the protocol's predeclared synthetic null-calibration criterion (2,000 repetitions per simulation condition). This criterion did not concern power. The lifecycle anchor is the earliest contemporaneous formal record, dated to the day and specific to the project. It marks entry from concept or feasibility into public development commitment. Eligible records date from the anchor to the day before its first anniversary and are censored before any same-scope outcome disclosure. Each record window must contain at least 90 observable days. Retrieval used frozen aliases and neutral technology, location, source and date terms. Outcome and stance terms were prohibited. For uncertainty, a cluster-t interval may be shown only as a working-independence sensitivity. A pre-output synthetic stress audit (20,000 repetitions) found coverage of 0.78775 for a nominal 95% interval under pairwise cluster correlation 0.20. This coverage shortfall is reported with any such interval. The 8 clusters are the full set of eligible clusters in the frozen registry, so the corpus is a census of eligible projects. These rules fixed the corpus before any real model output.

Relation to an earlier pilot. The study was motivated by defects found in an earlier, unpublished and unsubmitted pilot. The pilot also used LLMs, including Qwen, to measure project-level frames in CCUS discourse, and it examined cross-model disagreement. The candidate project list was inherited from the pilot. One of the 8 eligible projects also appeared in the pilot's empirical sample. The eligibility gates, source layer, masked inputs, three-checkpoint and three-repeat design, failure taxonomy and estimand are new. No text passage analyzed in the pilot occurred verbatim in the 11 current documents, and no pilot result, table or figure was reused.

A.2 Cell states and support

Cell states record what happened to every required call. A cell is complete only if every required Stage-1 and Stage-2 call reaches terminal success and the cell holds classified or verified zero claims. Distinct states are retained for retrieval missingness, source-rights exclusion and infrastructure failure. Distinct states are also retained for schema failure (including non-finite or invalid probability vectors), evidence-alignment failure, context-policy failure, model refusal or empty, verified zero claims (the no-claim state) and no classifiable claims. None of these states is mapped to another. The primary analysis uses common support. The failure policy also defines an all-attempted sensitivity analysis, which treats terminal failures as explicit pipeline outcomes. This sensitivity analysis is not an item of the analysis plan and was not computed. Only infrastructure failures before a complete response are retried (worker crash, out-of-memory, interruption, transient read failure). At most two retries are allowed, after waits of 30 s and 120 s. Each retry keeps the prompt, parameters, seed, run-order (batch) position and revision. Figure 1 summarizes the flow from calls to cell states and common support. Its counts come from the complete 3×3 matrix executed beyond the early

stop, and the frozen run is a prefix of this matrix. Cell codes follow Table 2. C is complete and classified, and Z is complete with verified zero claims (one-hot on the no-claim state). Codes 1 and 2 are incomplete at Stage 1 and Stage 2, and N has no classifiable claims. The frozen function never maps a failure to the no-claim state. Only the all-valid-record view recodes the 135 failed calls, and no divergence is computed in it. This view, dashed in Figure 1, is analysis-plan item (d), kept for failure transparency only.

A.3 Analysis plan in full

The plan below is exhaustive for confirmatory analyses, and any analysis outside it is exploratory. **(a) Finite-corpus description.** Δ_{JS}, all eight $D_B(j)$, $D_W(j)$ and contrasts, their median and range, the three system-pair means and three within-system means. **(b) Influence.** Leave-one-cluster-out means, sign crossings and a dominance diagnostic. **(c) Uncertainty.** No design-based or super-population interval is primary. **(d) Representation.** Δ_{JS} recomputed on $\tilde{p}$ with the same support, with near-tie, entropy and ambiguity diagnostics. A separate all-valid-record view retains successful outputs from cells later rendered incomplete. It serves failure transparency only. **(e) Stagewise localization.** Strict normalized-key diagnostics cover extraction, classification and aggregation over the same 27 cross-system and 9 within-system cell pairs per cluster. Classification uses the aligned denominator of mutually classified claims and seven-category Gwet AC1. No fuzzy, embedding or LLM matcher is used, and the diagnostics are not an additive or causal decomposition. **(f) Failure flow.** The full missingness and terminal-state matrix by cluster, system and repeat. Pairwise system comparisons receive no p-values, so no multiplicity correction applies. Null, negative and not-estimable results are reported without changing systems, prompts, representation or support. Only the all-valid-record view of (d) and item (f) were computed, because common support was empty.

A.4 Stop rules

Comparative results stay uninspected in five cases. Two concern inputs: a model revision, prompt-bundle hash or repeat registry differing from the frozen manifest, and drift in source or masking hashes. Two concern execution, as failing deterministic validators or a systematic runtime error affecting every cell of a system or stage. The fifth is common support in fewer than 8 clusters. Because the minimum equals the census, the first terminal failure of a required call already implies the fifth case, so the runner stops there. A fatal implementation bug found after output inspection must be recorded in a dated deviation. The code must then be corrected, and all systems and repeats rerun in full. Selective reruns and best-version selection are prohibited. Comparative results were not inspected in the reported run.

A.5 Interpretation of the estimand and the matched representation

The interpretation of the estimand was fixed before any output. Both components of Δ_{JS} are invariant to relabeling repeats within a system. If the three systems produced identically distributed outputs, the two components would have the same expectation. A positive Δ_{JS} means the average cross-system divergence exceeds the average within-system repeat divergence in this frozen corpus and configuration. A value near zero does not imply high agreement, because both components may be large. The plan therefore reports both components whenever the estimand is computed. The sign of Δ_{JS} does not identify a correct system or a population effect. Nor does it identify a model-family effect, since one checkpoint represents each upstream family. The matched sensitivity representation $\tilde{p}_{jsr}$ is the equal-weight mean of the probability vectors of exactly the same non-abstained claims, on identical support. It is not primary, because reported probabilities are not calibrated across architectures. None of this reading was applied, since the estimand was not computed.

A.6 Influence and representation diagnostics

Two further diagnostics would have qualified the primary estimate. Plan item (b) reports all eight leave-one-cluster-out (LOO) means, the maximum absolute LOO shift, any sign crossing and a dominance diagnostic. With $\Delta_j = D_B(j) - D_W(j)$, cluster j has the share $|\Delta_j - \Delta_{\mathrm{JS}}| / \sum_k |\Delta_k - \Delta_{\mathrm{JS}}|$. The dominance diagnostic reports the largest single-cluster share and the combined share of

Table 2: Under the frozen protocol, common support held in 0 of 8 clusters. Mistral completed 0 of its 24 cells, so no cluster has 9 complete cells. The table gives frozen cell states for all 72 system×repeat×cluster cells, recomputed with the frozen cell function from the call records. C is complete and classified, and Z is complete with verified zero claims (one-hot on no-claim). Cells coded 1 and 2 are incomplete at Stage 1 and Stage 2, respectively, because a required call reached a terminal failure. N has no classifiable claims. A cluster enters common support only if all 9 of its cells are C or Z. This matrix shows the frozen-protocol view only.

	Qwen			Mistral			Granite		
Cluster	R01	R02	R03	R01	R02	R03	R01	R02	R03
G001	1	1	1	1	1	1	1	1	1
G002	1	1	1	1	1	1	1	1	1
G003	1	1	1	1	1	1	1	1	1
G004	1	1	1	1	1	1	1	1	1
G005	1	1	1	1	1	1	1	1	1
G006	1	1	1	1	1	1	1	1	1
G007	Z	Z	Z	1	1	1	C	2	1
G008	1	1	1	1	1	1	1	1	1

the two largest clusters. It flags dominance when the largest single-cluster share is at least 0.50. Plan item (d) reports near-tie shares (top-1 minus top-2 probability margin $\leq 0.01, 0.05, 0.10$), claim-level entropy and the ambiguity rate, without deleting claims. Neither item (b) nor item (d) was computed on common support, which was empty.

A.7 Stagewise formulas

The stagewise diagnostics localize divergence to extraction, classification or aggregation. Each of the span text, actor mention and atomic claim fields is normalized separately by NFKC, case-folding and collapse of non-word runs and whitespace to single spaces. The three normalized fields are joined into one claim key. Among claims with the same key, the first in source order is kept. Consider cells a, b with verified claim-key sets A, B, where $n_a = |A|$ and $n_b = |B|$. Jaccard is $|A \cap B|/|A \cup B|$, set to 1 if both are verified empty. Dispersion is $|n_a - n_b|/(n_a + n_b)$ and the unmatched rate is $|A \triangle B|/(n_a + n_b)$. Each is set to 0 if both sets are verified empty. A missing cell never becomes an empty set. Take the n keys in $A \cap B$ with valid non-abstained primary frames in both cells. Exact agreement p_o is their matching share. Gwet AC1 is $(p_o - p_e)/(1 - p_e)$, with $p_e = \sum_{k=1}^{7} \pi_k(1 - \pi_k)/6$ and π_k the pooled share of category k over both cells. If $n = 0$, the pair is not estimable. Complements $1-$Jaccard and $1 - p_o$ are display transforms. The aggregation diagnostics are the primary categorical D_B, D_W and Δ_{JS} of Equations (1)–(2), computed on the same pairs. Optional exploratory slices do not change support. Language, source, volume or anchor-subtype slices may be reported only where the eight-cluster support is nondegenerate. Failure-rate slices need no such condition. These formulas were fixed but never applied on common support.

A.8 Analyses excluded by the protocol

Some analyses were excluded before execution. The protocol excludes project-score ICC/Spearman, top-k capacity analyses, project-level discordance thresholds, selective coverage or project abstention, and pairwise p-values. It also excludes semantic or fuzzy claim matching, bootstrap intervals from three repeats, human-gold accuracy and cancellation-cause analysis. Recorded project status may be used only for descriptive stratification of discordance, after the primary analysis is locked. Two alternatives to the estimand are also prohibited. They are an all-pairs median and a D_B restricted to equal repeat indices. Equal indices are not coupled across systems, because seeds are derived per system.

A.9 Operational history

The operational record documents how the reported run was executed.

Execution path in detail. The protocol specified execution through a containerized, hash-verified release chain (`exact-v`N). The reported run used a direct runner. It imported the frozen runner module unchanged (source hash prefix `d905cb64879c`). It called the module's own functions for chunking, prompt rendering, grammar-constrained generation, seed derivation, infrastructure retries, validation, cell construction and early stop. Its analysis block was executed verbatim from the frozen source. The direct runner bypassed the chain's gate reports and frozen-input verification. The model snapshots were downloaded at the pinned revisions, and every weight file was hashed and recorded. No frozen reference hash of the weights existed for comparison. The runner retains BF16 weights, eager attention, deterministic algorithms and the pinned package versions. This path has two further features. First, parallel workers executed the calls, and each wrote one resumable terminal record per call with the frozen call fingerprint. To keep populating a cell after a failed call, the workers passed the failure to the cell function as an empty or abstained placeholder. The terminal record on disk kept the failure, and frozen cell states used only this record. The frozen runner then read these records and reached its early-stop state without generating any new call. Run order and per-call seeds were fixed in advance, so the frozen run is a prefix of the executed matrix. Second, the frozen runner would have stopped after its first terminal failure. We executed the remaining calls to report the failure flow of plan item (f), and no divergence was computed. On the all-valid-record view, the frozen analysis code also wrote its automatic stagewise and near-tie diagnostic files despite the insufficient frame. We did not inspect these files, and they are not reported or distributed. The scientific specification (systems, prompts, corpus, repeats, seeds, representation, estimand and diagnostics) is unchanged.

Run history. The reported run used 3×NVIDIA A100-SXM4-80GB on 2026-09-25 with PyTorch 2.6.0+cu124, Transformers 4.51.3 and lm-format-enforcer 0.11.3. Each call was generated as a single unbatched sequence (batch size one). One worker ran per system and repeat. The remaining Granite work was later redistributed to workers covering disjoint pairs of projects, to shorten wall-clock time. An earlier run under the original protocol used unconstrained generation. It stopped under the failure policy in the first system's Stage 1 after consecutive terminal invalid-JSON responses. It produced no schema-valid claim set, classification or estimate. A dated post-output amendment followed. It used only sealed status and syntax counts, which carry no response content. Raw responses and candidate claim or frame content were not read. The amendment added the system-neutral grammar-constrained JSON interface and automatic enforcement of the existing eight-cluster stop. No response was repaired, resumed or selectively rerun. Subsequent containerized attempts stopped at operational gates before any model call. The gates were prerequisite tests, a launch-permission check and image registration at the compute provider. None of these attempts produced a scientific result, and none was pooled with the reported run. During the reported run, the first Granite workers were stopped deliberately to redistribute their work. This interrupted 3 calls, which were regenerated under the infrastructure-retry rule, each with its original prompt, parameters and seed. Before the run, a smoke test on synthetic text (not the corpus) confirmed each checkpoint loaded under the frozen settings. It also confirmed each checkpoint returned byte-identical output when the same seeded call was run twice at each stage. The direct runner then replaced the containerized chain. It produced the run reported here, the first execution to reach model calls on the corpus under the grammar-constrained interface. Together, these details document the deviations behind the reported run.

B Prompts and Codebook

All three systems received the frozen prompt templates below, and the last subsection summarizes the codebook.

B.1 Stage-1 claim-extraction prompt (verbatim, frozen `claim_extraction_v1.0`)

```
SYSTEM
You are a measurement annotator. Use only the supplied identity-masked, pre-decision text. Extract explicit
stakeholder positions; do not decide whether they are correct. Do not infer project outcome, cancellation
↪  cause,
social-license status, intent, or facts from world knowledge. Return one valid JSON object and no prose.

USER TEMPLATE
ACTOR TAXONOMY
```

```
{{ACTOR_TAXONOMY_JSON}}

DOCUMENT
document_id: {{DOCUMENT_ID}}
chunk_id: {{CHUNK_ID}}
text:
<<<TEXT
{{TEXT}}
TEXT

TASK
Extract every explicit atomic stakeholder position about the proposed infrastructure project in this chunk.
One item must contain one actor and one proposition. Preserve supportive, opposing, neutral, conditional, and
uncertain positions. Do not search only for conflict. Do not classify the social-license frame in this stage.

OUTPUT SCHEMA
{
  "document_id": "exact supplied document_id",
  "chunk_id": "exact supplied chunk_id",
  "claims": [
    {
      "span_text": "exact contiguous supporting substring from TEXT",
      "span_occurrence": 1,
      "atomic_claim": "concise faithful proposition with no added premise",
      "actor_mention": "exact actor mention or unambiguous pronoun from span_text",
      "actor_type": "one frozen actor label",
      "stance": "support|oppose|neutral|conditional|unclear",
      "abstain_reason": null
    }
  ]
}

VALIDATION RULES
1. span_text must be an exact contiguous substring of TEXT after newline normalization only.
2. span_occurrence is the one-indexed occurrence of that exact span in TEXT.
3. Do not combine different actors or non-contiguous evidence.
4. Split a sentence into multiple items when it contains separable positions or actors; the same exact span
↪  may
   support more than one atomic item when necessary.
5. Do not output factual background, project description, or a journalist's narration unless it attributes an
   evaluative position to an explicit stakeholder.
6. If actor or proposition evidence is insufficient, omit the item. Do not manufacture an abstention item.
7. If there are no qualifying positions, return an empty claims array.
8. Keys, enums, and JSON types must match the schema exactly. No markdown fence.
```

B.2 STAGE-2 FRAME-CLASSIFICATION PROMPT (VERBATIM, FROZEN `frame_classification_v1.0`)

```
SYSTEM
Classify one extracted atomic stakeholder claim using only the supplied claim and frozen codebook. Project
identity and recorded outcome are masked. High agreement is not validity. Return one valid JSON object and no
prose.

USER TEMPLATE
FRAME CODEBOOK
{{FRAME_CODEBOOK_JSON}}

CLAIM
{{ATOMIC_CLAIM_JSON}}

TASK
Assign the principal frame carried by the claim's reason or evaluative warrant. Abstain only when the supplied
claim is not supported by its exact span or is not a stakeholder position. Ambiguity between two supported
frames is represented by a primary and secondary frame; it is not an abstention.

OUTPUT SCHEMA
{
  "primary_frame":
  ↪  "safety_health|place_property|procedural_fairness|distrust|climate_legitimacy|economic_cost|other|null",
  "secondary_frame":
  ↪  "safety_health|place_property|procedural_fairness|distrust|climate_legitimacy|economic_cost|other|null",
  "ambiguous": true,
  "frame_probabilities": {
    "safety_health": 0.0,
    "place_property": 0.0,
    "procedural_fairness": 0.0,
    "distrust": 0.0,
    "climate_legitimacy": 0.0,
    "economic_cost": 0.0,
    "other": 0.0
  },
  "evidence_words": "minimal exact contiguous substring from atomic_claim or span_text",
  "abstain_reason": null
}

VALIDATION RULES
1. Include every probability key exactly once; values must be finite numbers in [0,1] summing to 1 within
   absolute tolerance 0.001.
2. If primary_frame is not null, it must have the largest probability (ties allowed).
3. secondary_frame is null unless ambiguous=true; it must differ from primary_frame.
4. For a supported classification, evidence_words must be an exact contiguous substring of atomic_claim or
   span_text.
5. For abstention, set primary_frame and secondary_frame to null, ambiguous=false, evidence_words="", and give
```

Table 3: Frame codebook summary, with the principal warrant of each frame.

Frame	Include (principal warrant)
`safety_health`	accident, leakage, exposure, emergency response, physical safety, public or worker health, environmental pathways tied to health or safety
`place_property`	land, property rights or value, eminent domain, landscape, place attachment, route siting, local amenity, disruption to farms and homes
`procedural_fairness`	consultation, consent, notice, process transparency, participation, due process, decision authority, fair hearing
`distrust`	honesty, credibility, motives, competence, independence, conflicts of interest, greenwashing, capture of an institution or proponent
`climate_legitimacy`	emissions reduction, mitigation, decarbonization, fossil lock-in, carbon accounting, whether the project is a credible climate solution
`economic_cost`	jobs, taxes, rates, subsidy, compensation, cost, affordability, financial risk, local development, economic benefit or burden
`other`	an explicit relevant stakeholder position not faithfully assignable to the six substantive frames (never a pipeline state)

Table 4: Pinned repositories and revisions of the three systems.

System	Repository	Pinned revision
Qwen	`Qwen/Qwen2.5-7B-Instruct`	`a09a35458c702b33eeacc393d103063234e8bc28`
Mistral	`mistralai/Mistral-7B-Instruct-v0.3`	`c170c708c41dac9275d15a8fff4eca08d52bab71`
Granite	`ibm-granite/granite-3.1-8b-instruct`	`4009206d5fc95d2e65a7b7633e159d6e97e25d35`

```
   a specific abstain_reason. The probability object is still required.
6. Do not infer cancellation cause, adoption, project success, social-license status, or which model is
↪  correct.
7. Keys, enums, and JSON types must match the schema exactly. No markdown fence.
```

(Lines marked ↪ are wrapped for typesetting only; the prompt text is reproduced byte for byte.)

B.3 Frame codebook summary (`frame_codebook_v1.0`)

Table 3 lists each frame with its principal warrant.

The priority rule selects the frame carrying the claim's principal reason or evaluative warrant, and a merely mentioned noun does not decide the frame. If two warrants are inseparable, the classifier gives a primary and a secondary label and sets the ambiguity flag. The codebook forbids inferences about project success, cancellation cause, social-license status, adoption or correctness. It also forbids use of project identity, model family, recorded outcome or outside knowledge. The no-claim state is assigned only by deterministic aggregation. The Stage-1 actor taxonomy has ten labels: government/regulator; project proponent/operator; investor/financier/insurer; employee/labor union; resident/landowner/community; Indigenous/tribal; NGO/advocacy; expert/academic/professional; industry/business/customer; and other. The label describes the speaker or attributed source, and the publisher is not classified.

C Systems, Decoding and Seeds

Table 4 lists the pinned model revisions, and the decoding settings and seeds follow.

The context limit is 4,096 tokens. Stage 1 allows 2,048 input and 1,536 new tokens, and Stage 2 allows 2,048 input and 512 new tokens. Token counts use the maximum over the three tokenizers. Paragraphs are accumulated in source order, and overlong paragraphs are split at frozen sentence boundaries with no overlap. Decoding uses sampling at temperature 0.2 and top-p 0.95, with top-k disabled, one beam, repetition penalty 1.0, KV cache on and native EOS stop. The protocol records no separate rationale for the repetition penalty or the output limits, which were fixed with the other settings before any output. Within a system and repeat, calls run over projects in sorted order of internal project ID. The first project is in cluster G008, so the cluster-ordered rows of Figure 2 do not follow run order. Within a project, calls follow masked docu-

ment ID, chunk ID and claim index. The model manifest lists the document, chunk and claim-index keys but not the project ID. For repeat base seed b_r, each call takes the first eight bytes of $\mathrm{SHA256}(b_r \mid \text{system} \mid \text{stage} \mid \text{document} \mid \text{chunk} \mid \text{claim index})$. These bytes are read as an unsigned big-endian integer, and the call seed is this integer modulo 2147483647. The seed registry is `repeat_registry_v2.0` (system-keyed seeds). A dated amendment designated it as the controlling registry before any model call, and the deployment, model-freeze and analysis-freeze manifests name it. The earlier model manifest still names v1. These settings fix every generation input except hardware, which the run record documents.

Choice of three open 7–8B checkpoints. The protocol requires every call to be re-executable from pinned weights, tokenizer, decoding code and seed. Hosted frontier models cannot be pinned in this sense. A hosted model's behavior can change without notice (Chen et al., 2024), and identical requests need not return identical outputs (Zhu & Zhang, 2026; Atıl et al., 2025). One instruction checkpoint per upstream family at a similar scale keeps the three systems comparable and executable on one node. The results therefore describe this contract applied to small open checkpoints.

D Claim-Level Alignment Failures, Format-Failure Anatomy and Adapter Replay

This appendix details the claim-level failures and the adapter replay behind the main results. In Figure 2, rows are chunks grouped by cluster, and columns are systems $\times$ repeats. Each cell is the terminal outcome of one of the 153 executed Stage-1 calls under the frozen contract. Outcomes are identical in the all-valid-record view. V and 0 mark schema-valid, evidence-aligned output with at least one claim and with zero claims, respectively. S and A mark evidence-alignment failures. L, T and X mark format failures, and T and X are the 16 adapter failures. Letter counts per system equal the Stage-1 rows of Table 1. The figure is exploratory and descriptive.

The 93 alignment-failed calls contained 318 claims. Of these, 88 spans occurred nowhere in the chunk even after normalization, and 7 would have matched only after normalization. Another 30 occurred verbatim but with an occurrence index larger than the number of verbatim matches. Among claims whose span resolves, 25 actor mentions contained appended parenthetical annotations. A further 61 actor mentions occurred in the chunk but outside the span, 56 occurred nowhere in the chunk, and 51 claims passed. For each of the 88 absent spans, at least 90% of the words occurred in the chunk. The 56 actor mentions found nowhere in the chunk included actor-type labels written in place of a mention and composites joining several entities.

Of the 21 responses cut off at the output limit, 10 contained a field in which fewer than half of the word 4-grams are distinct. Of the rest, 5 contained a single field of at least 1,000 characters. The remaining 6 had no field of this length. The other 16 unparseable responses, all from Qwen, were generated under the JSON constraint without reaching the limit. They comprise 13 early terminations and 3 control-character cases. We replayed the recorded outputs through the frozen adapter on CPU. The adapter admitted every token of 13 of the 13 re-tokenized outputs, including the premature end-of-sequence token. This replay located the mechanism of the early terminations. In all 13 early terminations, the adapter's parser state collapsed at a span-text value. The collapse happened at the closing quote of a one-space string (10) or at a typographic opening quote (3). In each case, the end-of-sequence token first became allowed at the very next position. Generation then ended after at most 12 whitespace characters, the adapter's configured limit on consecutive whitespace. These terminations fell on 5 chunks. Of these chunks, 4 terminated early in all three repeats. For 3 of these 4, the three responses were byte-identical. Replay reproduced 3 of the 3 control-character cases. In each, a token ending in a backslash came first. The adapter then admitted a token made of a quote, a comma and a raw newline. The quote was read as escaped, so the string stayed open. The adapter's own character-level parser rejected the newline at this position, yet its token-level check admitted it. Counting the adapter failures as parseable at the last rung of the relaxation ladder would let 132 calls pass Stage 1. The mean pairwise character similarity of repeated responses (`difflib` ratio) was 0.69 for Qwen, 0.72 for Mistral and 0.55 for Granite. Table 5 gives the repeat and cross-system counts behind Section 5. Qwen and Granite succeeded together more often than expected under within-system permutation (5 versus 0.9). All of these joint successes fell on 2 chunks of cluster G007, where Qwen returned verified empty claim sets. The

Table 5: Stage-1 outcomes recur across repeats and depend on both the system and the chunk. The table gives frozen-contract call outcomes. The left panel sorts each system's 17 chunks by repeat pattern. The right panel sorts the 51 (chunk, repeat) pairs by which systems succeeded. It compares them with the count expected if each system's outcomes were independent of the chunk. This expectation is exact under random permutation of each system's outcomes over the (chunk, repeat) pairs. It holds each system's success count fixed and treats the systems as independent and the repeats of one chunk as separate pairs. The within-system expectation in the text (31.2) uses the same null. All 9 Qwen successes are verified empty claim sets.

Chunks by repeat pattern (per system)				Pairs by systems succeeding		
Repeat pattern	Qwen	Mistral	Granite	Systems succeeding	Observed	Expected
All 3 repeats succeed	3	1	1	none	33	31.2
Mixed	0	4	2	Mistral	8	5.8
All 3 repeats fail	14	12	14	Granite + Qwen	5	0.9
				Qwen	4	6.7
				Granite	1	4.2
				Mistral + Qwen	0	1.2
				Granite + Mistral	0	0.8
				Granite + Mistral + Qwen	0	0.2

excess thus traces to these inputs alone. Mistral's successes never coincided with another system's (8 alone versus 5.8 expected).

The chance expectations in Section 5 and Table 5 assume independent random permutations of each system's k_s successes over the $n = 51$ (chunk, repeat) pairs. A success set S is then expected on $n \prod_{s \in S}(k_s/n) \prod_{s \notin S}(1 - k_s/n)$ pairs. Three agreeing repeats are expected on $17 \sum_s \left[\binom{k_s}{3} + \binom{n-k_s}{3}\right] / \binom{n}{3}$ (system, chunk) groups. Figure 2 shows the outcome of every Stage-1 call. These details support the call-level account in the main text.

E Background on Measurement Validity

Further measurement-validity background frames the audit. Measurement theory treats validity as a property of the inferences a measure supports (Messick, 1995). Conceptualization and operationalization must be distinguished (Subramonian et al., 2023), and different failure modes should not be treated as equivalent (Hutchinson et al., 2022). Reliable annotation is a precondition for useful benchmarks (Bowman & Dahl, 2021). A review of 445 LLM benchmarks finds recurring construct-validity problems (Bean et al., 2025). Measurement invariance (Vandenberg & Lance, 2000) motivates an earlier requirement here, an admissible output from every system before systems are compared. Abstention by LLMs is usually studied as a capability (Wen et al., 2025). In this audit it is one of several states the state model keeps apart. Nonresponse rates alone predict nonresponse bias poorly, and the bias depends on whether nonresponse relates to the measured quantity (Groves, 2006). Here, nonresponse was tied to system and input, the case in which it can bias a cross-system comparison. These principles motivate treating admissibility as a precondition of agreement.

Related work in detail. The four paragraphs below give the studies behind the first four paragraphs of related work.

Detail: LLM annotation as measurement. Validation practice in published work remains uneven (Desai et al., 2026). A call to validate every annotation task against human labels (Pangakis et al., 2023) came with best-practice standards including rigorous validation (Törnberg, 2024). A large-scale study of LLM evaluators reports substantial variance across models and datasets (Bavaresco et al., 2025). Validating an evaluator depends on how indeterminate human ratings are aggregated (Guerdan et al., 2025). Configuration choices alone can change downstream conclusions (Baumann et al., 2025). LLM annotation errors can be non-random and bias downstream estimates unless corrected (Egami et al., 2023; Ashwin et al., 2026). The closest precedent for sequencing checks is the codebook framework of Halterman & Keith (2026). It tests an LLM's basic capabilities on a codebook before testing measurement accuracy, and it finds open-weight models limited in

following codebooks zero-shot. These works target accuracy against human labels or the value of LLMs as assistants to human annotators (Gu et al., 2026). The gap we address is cross-system admissibility, namely whether every compared system returns an admissible output for the same inputs.

Detail: Consistency, prompt variation and nondeterminism. Identical inputs can return different labels (Reiss, 2023). Stability varies across semantically equivalent prompts (Liu, 2026), and model retirement threatens re-annotation (Cheng et al., 2024). Prompt-sensitivity work separates variation caused by the model from variation introduced by prompt wording, formatting or evaluation procedure (Errica et al., 2025; Hua et al., 2025; Sclar et al., 2024). Outputs also vary under nominally deterministic settings, with inference configuration and hardware (Atıl et al., 2025; Yuan et al., 2025). A preregistered audit of hosted LLM evaluators found the instrument unstable although every request was executed cleanly (Zhu & Zhang, 2026). Identical calls need not recover identical structure (Sarkar, 2026). Consistency does not establish accuracy. Self-consistent errors have been documented in LLM generation (Tan et al., 2025), and highly reproducible LLM measures can diverge from the construct they target (Batzdorfer & Santagiustina, 2026). These studies measure variation among outputs already obtained. Rao & Callison-Burch (2026) show how the handling of invalid outputs changes what agreement estimates. None of them executes a cross-system comparison in which invalid outputs are common and differ by system.

Detail: Structured output and constrained decoding. Benchmarks of JSON generation document schema violations without constraints, including for 7–8B open models (Lu et al., 2025; Shorten et al., 2024). Constrained decoders mask invalid tokens at each step (Geng et al., 2023; Willard & Louf, 2023; Beurer-Kellner et al., 2023; Zheng et al., 2024; Dong et al., 2025). Their compliance in practice is uneven across engines and schemas (Geng et al., 2025). A grammar constraint alone does not respect a token budget, so outputs can be truncated at the length limit (Kato & Tarashima, 2026). Measured format compliance can also reflect the serving layer, and unrecorded failures can be misread as model behavior (Tang & Zheng, 2026). Constraining generation can distort the model's distribution or lower task quality (Park et al., 2024; Beurer-Kellner et al., 2024; Banerjee et al., 2025; Tam et al., 2024). For small models, constrained decoding removed all structural failures while leaving a semantic gap (Chavan, 2026). Decoders enforcing only local prefix feasibility can fail to reach acceptance when tokenizer and grammar are misaligned or token budgets are finite (Collura et al., 2026). Format compliance also varies sharply across model families (Kotte, 2026). Parseability can drive apparent format sensitivity (Mehta, 2026), and format repair can masquerade as self-correction (Chen et al., 2026). These studies report compliance per model, family or engine. The gap is how such failures carry into a cross-system measurement comparison.

Detail: Verbatim evidence and attribution. Citation benchmarks find substantial gaps between generated statements and the evidence cited for them (Gao et al., 2023; Zhang et al., 2025a). In retrieval-augmented generation, a majority of responses from 7B chat and instruct models contain at least one hallucinated span (Niu et al., 2024). Models can be trained to quote verbatim (Zhang et al., 2025b). On clinical questions, most of the twelve evaluated models attach verbatim quotes to over 90% of their claims when prompted. The quotes, however, often fail to substantiate every detail (Zhang et al., 2026). Evidence extraction for meta-analysis breaks down as relational demands grow (Tan & D'Souza, 2026).