Stage–Audit: Auditable Source-Frontier Discovery for Cross-Wiki Tables

Chen Shen
Megagon Labs
chen_s@megagon.ai

Abstract

LLM-curated tables can appear source-grounded while containing unsupported rows: the curator may recall entries from parametric memory and retroactively attach page-level citations that are not the actual source. We study this hazard in *Seed2Frontier discovery*: the task of finding complement Wikipedia pages from a seed page to assemble a structured table. Stage–Audit addresses it with disjoint curator-auditor write rights, a row-level source-citation gate, and a 12-check audit taxonomy over keys, schema, source roles, cardinality, and scope. On a curated 51-instance Seed2Frontier evaluation set spanning 15 top-level domains, Stage–Audit improves source-frontier precision over a vanilla LLM curator from 0.356 to 0.505 ($+42\%$ relative) and F1 from 0.334 to 0.451 ($+35\%$), while maintaining explicit per-row source traceability. The vanilla-LLM-vs-Stage–Audit comparison isolates the policy contribution rather than LLM-based discovery in general.

1 Introduction

Take a question like "list transboundary rivers longer than 1000 km by continent." No single Wikipedia page carries the answer. An LLM agent that wants to return a structured table, not a paragraph, has to walk a master list, the continent overview pages, and the individual river articles, then stitch them into rows that are keyed and source-traceable. We call this problem *Seed2Frontier discovery*: starting from a seed Wiki page, discover the source frontier needed to assemble a structured table.

The technical hazard is unintentional ungroundedness. An LLM curator can emit rows recalled from parametric memory and retroactively attach plausible page-level citations; the row looks cited even when the citation is not the actual source of the row. Prior work documents self-preference and weak self-correction at evaluation time [Panickssery et al., 2024, Wataoka et al., 2024, Barkan et al., 2025, Huang et al., 2024, Kamoi et al., 2024]; here the failure happens earlier, when the reference table is built.

We introduce *Stage–Audit*, a governance protocol for Seed2Frontier. A curator can stage canonical rows only with an external source URL and locator; an auditor cannot edit the table and instead appends findings under a 12-check taxonomy over row evidence, primary keys, schema, source roles, cardinality, and scope (Figure 1). Existing retrieval systems can attempt Seed2Frontier (Section 2); our contribution is the governance overlay, not a new retrieval method. A table can have individually plausible rows and still be unusable if its partition, key, or exhaustiveness claim is wrong, which is what our table-level audit catches.

We make three contributions. **(1) Task and curated set:** a 51-instance Seed2Frontier evaluation set spanning 15 top-level domains, with Wikipedia seed pages, labelled complement pages, and primary-key ground-truth tables. **(2) Protocol:** Stage–Audit defines role-disjoint write rights, a source-citation gate, a row-witness property, and a 12-check audit taxonomy for structured tables. **(3) Policy ablation:** a four-configuration comparison (memory-only, seed-outlink, vanilla LLM curator,

AI Agents for Discovery in the Wild (AID-Wild), Workshop at ACM CAIS 2026.

Stage–Audit) shows that the source-citation gate plus audit, not LLM-based discovery alone, drives source-frontier precision and F1.

2 Related Work

Source-grounding and judge bias. FActScore, ALCE, SAFE, and PaperTrail evaluate generated text after the fact [Min et al., 2023, Gao et al., 2023b, Wei et al., 2024, Martin-Boyle et al., 2026]; LLM-as-judge work documents bias and self-preference under same-family setups [Zheng et al., 2023, Panickssery et al., 2024, Wataoka et al., 2024]. Stage–Audit places the gate before a row enters the table, so the output is a structured table with keys and scope claims rather than a sequence of atomic facts.

Cross-page Wiki retrieval and citation-aware generation. SRAG, KARMA, WikiContradict, and InfoGather bear on multi-page Wiki construction [Lin et al., 2025, Lu and Wang, 2025, Hou et al., 2024, Yakout et al., 2012]; HybridQA, OTT-QA, and Open-WikiTable evaluate QA over fixed multi-page table corpora [Chen et al., 2020, 2021, Kweon et al., 2023], whereas Seed2Frontier evaluates *discovery* of the source frontier itself. Our vanilla-LLM-curator baseline stands in for unguarded LLM-based retrieval of this family. RARR retrofits citations to free-text outputs and Self-RAG learns to self-cite during generation [Gao et al., 2023a, Asai et al., 2024]; both place the citation step inside the generator and have no separate audit role over a structured table contract. Stage–Audit instead enforces citation *before* a row is staged and adds a disjoint auditor whose findings cannot rewrite the curator's table; SPADE and MAST motivate the table-contract view [Shankar et al., 2024, Cemri et al., 2025], while generator–critic, debate, Self-Refine, and Constitutional AI explore critique-via-extra-calls without a row-level source gate [Madaan et al., 2023, Du et al., 2024, Bai et al., 2022].

3 Method

3.1 Stage–Audit Oversight Protocol

Let M be a model or agentic framework used to curate source-grounded tables for user questions. Some curated tables may later be used to evaluate or diagnose the same model family. The threat is not adversarial deception but unintentional self-reference: M emits remembered rows and then attaches plausible page-level citations. Because intrinsic self-correction is weak without external feedback [Huang et al., 2024, Kamoi et al., 2024], the protocol makes the source policy, not the model's self-critique, the load-bearing mechanism.

Design tenets. Stage–Audit has four operational rules. *Append-only audit*: the auditor cannot edit the canonical table and records findings separately. *Cited-source-first*: every emitted row must carry a source URL and locator before staging. *Parametric memory is not evidence*: remembered facts may suggest where to search, but cannot justify a row. *Two-axis severity*: every finding is tagged by issue type (factual / structural / scope-related) and severity (blocking / hygiene / suggestion), so repair can distinguish blockers from cosmetics.

Artifact lifecycle. A staged table moves through four states (*proposed*, *staged*, *audited*, *repaired*), iterating until no blocking findings remain. The curator proposes and stages only locator-backed rows; the auditor appends findings but cannot edit canonical rows. In deployment, a human acceptance step gates final sign-off; this paper evaluates the LLM-curator + LLM-auditor sub-pipeline only.

Source-gated acceptance and audit checks. We say that an accepted row has a non-parametric witness with respect to M if every emitted row R has an evidence locator $E(R)$ such that the locator content supports R under an extractor f that operates only on that locator content. Stage–Audit operationalizes this through a source-citation gate: proposed rows without locators are rejected before audit. The gate does not certify truth. It only shifts the failure mode: a row is wrong only if the extractor misread the cited locator. The artifact contract $A = (Q, K, C, P, S, E)$ records the user query, primary-key schema, columns, scope statement, source set, and evidence map. Audit checks span row evidence, key integrity, schema conformance, cardinality, source-role coverage, claim type, normalization, scope match, and temporal knowability (checks: App. I; severity: App. F; proof sketch: App. C). *Factual* findings target row evidence; *structural* findings target K,C; *scope-related* findings target P,S. Figure 1 shows two scope/cardinality findings the citation gate alone would miss.

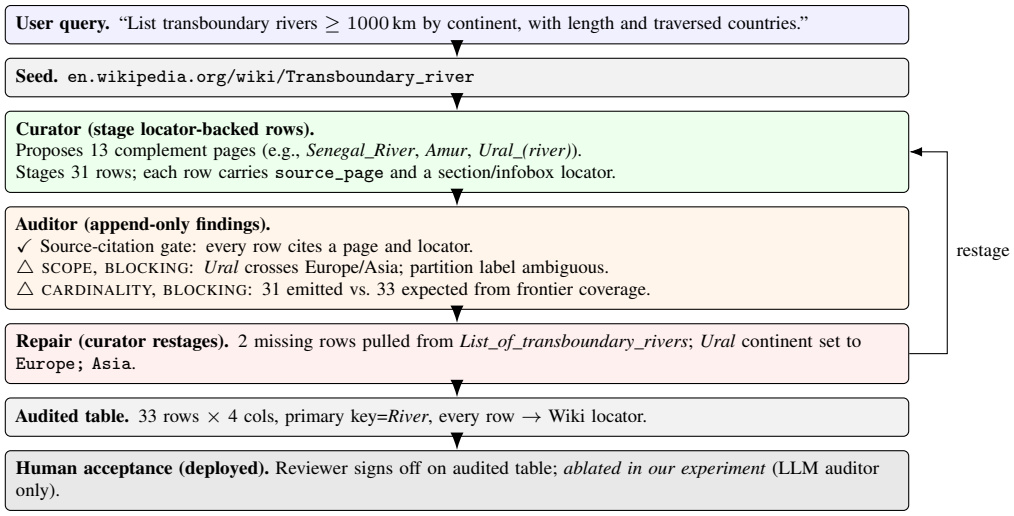

Figure 1: Stage–Audit walkthrough on a curated Seed2Frontier instance. The curator proposes locator-backed rows; the auditor appends findings under the 12-check taxonomy and cannot edit the table; blocking findings trigger a repair-restage loop back to the curator. The dashed final step shows the deployed human-acceptance role; our experiment ablates the human step and runs a single curator-and-auditor pass with no repair-restage iteration, to isolate the single-step value of the policy.

3.2 Seed2Frontier Discovery Procedure

Seed2Frontier takes a user question and a seed Wiki page, then asks for the complement pages needed to assemble a structured table. Stage–Audit bounds search through a logged frontier discipline: every page in S records the operator that admitted it (seed table, sibling list, redirect, entity page, optional Wikidata count or rank check; template in Appendix B). Audit treats missing required operators, unexplained pages, and mismatched source roles as scope failures. Full pseudocode appears in Appendix A. The procedure is a source-discovery policy, not a new retrieval method; existing planners can attempt the same task, and our experiment isolates the value of the governance overlay.

4 Anchor Experiment: Seed2Frontier Discovery

Curated evaluation set. We curate 51 Seed2Frontier instances spanning 15 top-level domains (geography, culture, science, politics, sports, business, technology, and others). 35 instances are independently authored questions; 16 are programmatic derivatives produced by filter, top-K, or year-range modification of multi-page parents (Appendix D). Each instance has a Wikipedia seed page, labelled complement pages (45 of 51, mean 10.1 per instance, range 1–35), and a primary-key ground-truth table.

Configurations. Four conditions evaluated with the same model. *Memory-only*: one prompt for parametric-memory rows, no source frontier produced. *Seed-outlink*: deterministic Wikipedia-API enumeration of all outlinks of the seed; no LLM, no rows. *Vanilla LLM curator*: a single LLM prompt asking for complement pages and rows, no source-citation gate, no audit (a proxy for an unguarded LLM-based retrieval planner). *Stage–Audit-governed*: curator prompt with the Wikipedia discovery hint and source-citation requirement, followed by an auditor prompt that appends findings under the 12-check taxonomy.

Setup. We run the experiment with `gpt-5.4` at $T{=}0$.[1] We manually label the seed Wikipedia URL for each instance as the most natural starting page among the candidate URLs. Two protocol steps are ablated: human acceptance (Figure 1, gray dashed) and any repair-restage iteration. We run a single curator-and-auditor pass to match the single-pass budget the vanilla and memory-only baselines receive (loop ablation in Appendix J). Following WikiTabGen-style scoring (Appendix E), we report

[1] Identifiers `gpt-5.4` and `Llama 3.3 70B` are API release tags, quoted verbatim from the API response.

Table 1: Stage–Audit on the 51-instance Seed2Frontier evaluation set with gpt-5.4 at $T{=}0$. Frontier metrics (primary axis) on the 45 instances with complement-page labels; PK metrics (secondary axis, see body) on all 51. Closed-loop scoring: when the auditor returns no accepted rows or pages (8/51 for rows, 4/51 for pages), Stage–Audit emits 0 for that instance rather than passing curator output through.

	Frontier (page set)				Primary key (row set)		
Config	Recall	Prec.	F1	Size	Recall	Prec.	F1
Memory-only	n/a	n/a	n/a	0.0	**0.644**	**0.645**	**0.619**
Seed-outlink	**0.580**	0.020	0.034	612.3	n/a	n/a	n/a
Vanilla LLM curator	0.384	0.356	0.334	15.1	0.598	0.611	0.588
Stage–Audit-governed	0.491	**0.505**	**0.451**	11.4	0.544	0.599	0.534

recall, precision, and F1 over two axes: predicted Wikipedia pages against the labelled complement set, and emitted rows against the labelled primary keys.

Frontier comparison. Stage–Audit improves source-frontier precision (0.505 vs. 0.356), F1 (0.451 vs. 0.334), and complement recall (0.491 vs. 0.384) over the vanilla LLM curator on the 45 instances with complement labels. We compute paired-bootstrap 95% CIs by resampling the 45 per-instance Stage–Audit−vanilla differences with replacement (5000 draws) and taking the central 95% range of the resulting means: $[+0.05, +0.25]$ for precision (two-sided sign test $p{=}0.006$), $[+0.03, +0.22]$ for F1 ($p{=}0.029$), and $[+0.02, +0.21]$ for recall. The precision and F1 intervals lie entirely above zero, so the gains are robust to instance resampling. The seed-outlink-vs-vanilla gap (precision 0.020 vs. 0.356) shows that LLM-based discovery does most of the precision lift over deterministic outlink enumeration; Stage–Audit then adds the source-citation gate and the audit step, raising precision a further +0.149 absolute (+42% relative) while producing per-row locator output the unguarded curator does not.

PK rows (secondary axis). By construction Stage–Audit emits only source-grounded rows, so unguarded baselines that emit every remembered row rank higher on PK overlap by design (memory-only $0.619 >$ vanilla $0.588 >$ Stage–Audit 0.534); the primary axis is frontier quality with per-row source traceability, and a memory-only table cannot satisfy a source-grounded consumer.

Where the policy helps. Stage–Audit's frontier-F1 gain over vanilla is largest on small-frontier instances (1–3 complement pages, $+0.25$, $n{=}17$); the gain shrinks on larger frontiers (4–9 pages, $+0.07$, $n{=}10$; 10+ pages, $+0.02$, $n{=}18$). The largest per-domain gains are in *culture* ($+0.32$, $n{=}7$), *sports* ($+0.25$, $n{=}4$), and *science* ($+0.18$, $n{=}5$); $n{\leq}3$ buckets are deferred to the supplement.

Audit-finding density. 182 findings across 51 instances (mean 3.6, range 1–5; issue-type×severity in Figure 2, Appendix H); the auditor returned no accepted rows on 8/51 instances and no accepted complement pages on 4/51, contributing 0 to the Stage–Audit means under closed-loop scoring. Stage–Audit's frontier averages 11.4 pages per instance, $54\times$ smaller than seed-outlink's 612.3.

Cross-model. Repeating the four-configuration evaluation on Llama 3.3 70B (open-source; Appendix G) yields a small directional gain ($+0.008$ frontier F1 vs. vanilla, $+0.012$ precision, $+0.033$ recall); the policy contribution is most pronounced on stronger instruction-following models.

5 Conclusion

Stage–Audit governs Seed2Frontier with disjoint curator-auditor write rights, a row-level source-citation gate, and a 12-check audit taxonomy. On 51 instances using gpt-5.4 it improves source-frontier precision over an unguarded LLM curator from 0.356 to 0.505 (paired 95% CI $[+0.05, +0.25]$, $p{=}0.006$) and F1 from 0.334 to 0.451. The present evaluation ablates the human-acceptance step and the repair-restage loop, and covers one closed-source and one open-source model. Within those scope limits, Stage–Audit serves as a source-grounding integrity overlay for cross-Wiki table construction.

References

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. Self-RAG: Learning to retrieve, generate, and critique through self-reflection. *International Conference on Learning Representations (ICLR)*, 2024.

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, et al. Constitutional ai: Harmlessness from ai feedback. *arXiv preprint arXiv:2212.08073*, 2022.

Casey O. Barkan, Sid Black, and Oliver Sourbut. Do large language models know what they are capable of? *arXiv preprint arXiv:2512.24661*, 2025.

Mert Cemri, Melissa Z. Pan, Shuyi Yang, Lakshya A. Agrawal, Bhavya Chopra, Rishabh Tiwari, Kurt Keutzer, Aditya Parameswaran, Dan Klein, Kannan Ramchandran, Matei Zaharia, Joseph E. Gonzalez, and Ion Stoica. Why do multi-agent llm systems fail? *arXiv preprint arXiv:2503.13657*, 2025.

Wenhu Chen, Hanwen Zha, Zhiyu Chen, Wenhan Xiong, Hong Wang, and William Yang Wang. HybridQA: A dataset of multi-hop question answering over tabular and textual data. In *Findings of the Association for Computational Linguistics: EMNLP*, 2020.

Wenhu Chen, Ming-Wei Chang, Eva Schlinger, William Yang Wang, and William W. Cohen. Open question answering over tables and text. In *International Conference on Learning Representations (ICLR)*, 2021.

Yilun Du, Shuang Li, Antonio Torralba, Joshua B. Tenenbaum, and Igor Mordatch. Improving factuality and reasoning in language models through multiagent debate. In *International Conference on Machine Learning*, 2024.

Luyu Gao, Zhuyun Dai, Panupong Pasupat, Anthony Chen, Arun Tejasvi Chaganty, Yicheng Fan, Vincent Y. Zhao, Ni Lao, Hongrae Lee, Da-Cheng Juan, and Kelvin Guu. RARR: Researching and revising what language models say, using language models. *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2023a.

Tianyu Gao, Howard Yen, Jiatong Yu, and Danqi Chen. Enabling large language models to generate text with citations. *arXiv preprint arXiv:2305.14627*, 2023b.

Yufang Hou, Alessandra Pascale, Javier Carnerero-Cano, Tigran T. Tchrakian, Radu Marinescu, Elizabeth Daly, Inkit Padhi, and Prasanna Sattigeri. Wikicontradict: A benchmark for evaluating llms on real-world knowledge conflicts from wikipedia. In *Advances in Neural Information Processing Systems*, 2024.

Jie Huang, Xinyun Chen, Swaroop Mishra, Huaixiu Steven Zheng, Adams Wei Yu, Xinying Song, and Denny Zhou. Large language models cannot self-correct reasoning yet. *arXiv preprint arXiv:2310.01798*, 2024.

Ryo Kamoi, Yusen Zhang, Nan Zhang, Jiawei Han, and Rui Zhang. When can llms actually correct their own mistakes? a critical survey of self-correction of llms. *Transactions of the Association for Computational Linguistics*, 2024.

Sunjun Kweon, Yeonsu Kwon, Seonhee Cho, Yohan Jo, and Edward Choi. Open-WikiTable: Dataset for open domain question answering with complex reasoning over table. In *Findings of the Association for Computational Linguistics: ACL*, 2023.

Teng Lin, Yizhang Zhu, Yuyu Luo, and Nan Tang. Srag: Structured retrieval-augmented generation for multi-entity question answering over wikipedia graph. *arXiv preprint arXiv:2503.01346*, 2025.

Yuxing Lu and Jinzhuo Wang. Karma: Leveraging multi-agent llms for automated knowledge graph enrichment. *arXiv preprint arXiv:2502.06472*, 2025.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, et al. Self-refine: Iterative refinement with self-feedback. *arXiv preprint arXiv:2303.17651*, 2023.

Anna Martin-Boyle, Cara A. C. Leckey, Martha C. Brown, and Harmanpreet Kaur. Papertrail: A claim-evidence interface for grounding provenance in llm-based scholarly q&a. *arXiv preprint arXiv:2602.21045*, 2026.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Wei Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. Factscore: Fine-grained atomic evaluation of factual precision in long form text generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 2023.

Arjun Panickssery, Samuel R. Bowman, and Shi Feng. Llm evaluators recognize and favor their own generations. In *Advances in Neural Information Processing Systems*, 2024.

Shreya Shankar, Haotian Li, Parth Asawa, Madelon Hulsebos, Yiming Lin, J. D. Zamfirescu-Pereira, Harrison Chase, Will Fu-Hinthorn, Aditya G. Parameswaran, and Eugene Wu. Spade: Synthesizing data quality assertions for large language model pipelines. *Proceedings of the VLDB Endowment*, 17(12), 2024.

Koki Wataoka, Tsubasa Takahashi, and Ryokan Ri. Self-preference bias in llm-as-a-judge. *arXiv preprint arXiv:2410.21819*, 2024.

Jerry Wei, Chengrun Yang, Xinying Song, Yifeng Lu, Nathan Hu, Jie Huang, Dustin Tran, Daiyi Peng, Ruibo Liu, Da Huang, Cosmo Du, and Quoc V. Le. Long-form factuality in large language models. In *Advances in Neural Information Processing Systems*, 2024.

Mohamed Yakout, Kris Ganjam, Kaushik Chakrabarti, and Surajit Chaudhuri. Infogather: Entity augmentation and attribute discovery by holistic matching with web tables. In *Proceedings of the ACM SIGMOD International Conference on Management of Data*, pages 97–108. ACM, 2012. doi: 10.1145/2213836.2213848.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging llm-as-a-judge with mt-bench and chatbot arena. In *Advances in Neural Information Processing Systems Datasets and Benchmarks Track*, 2023.

A Seed2Frontier Discovery Algorithm

Algorithm 1 Seed2Frontier discovery procedure (curator's propose step in Figure 1)

Require: natural-language table query Q, seed Wiki page s
Ensure: candidate source set S

$P \leftarrow \textsc{InferScope}(Q)$ ▷ partition/scope statement extracted from Q (e.g., "by continent" $\Rightarrow$ continents enumeration)
$S \leftarrow \{s\}$ with source role and retrieval metadata
if a single page covers all partitions in P **then**
    return S
end if
expand through sibling list pages, redirects, and list/table links
for each under-covered partition or tail entity in P **do**
    fetch the relevant entity page and add supporting pages to S
end for
if Q implies a count, rank, or partition-size constraint **then**
    run a Wikidata SPARQL count or rank sanity check
end if
log each URL, locator, retrieval time, and content hash for audit
return S

B Wikidata Sanity-Check Template

When a user question implies a count, rank, membership, or partition-size constraint, Wikidata can provide a weak sanity check on the source frontier. The query is not the row witness: accepted rows

still require source locators in E. In practice, the curator records the property and class choices, the query time, and the returned count or rank signal so an auditor can replay whether the frontier is plausibly complete.

```
# Q_CLASS, P_FILTER, Q_FILTER are placeholders for instance-specific
# Wikidata classes and properties; substitute before issuing the query.
PREFIX wd: <http://www.wikidata.org/entity/>
PREFIX wdt: <http://www.wikidata.org/prop/direct/>
SELECT (COUNT(DISTINCT ?entity) AS ?n) WHERE {
  ?entity wdt:P31/wdt:P279* wd:Q_CLASS .
  ?entity wdt:P_FILTER wd:Q_FILTER .
}
```

C Proof Sketch for the Non-Parametric Witness Claim

The source-citation gate gives a simple invariant over staged rows. Let R_A be the rows in artifact A and let $\mathrm{Block}(A)$ be its unresolved blocking findings. For row r, define the operational acceptance predicate:

$$\mathrm{RowOK}(r, A) \equiv \exists e \in E(r) : e \in S \wedge \mathrm{supports}_f(e, r).$$

The acceptance condition is then:

$$\mathrm{Accept}(A) \equiv \forall r \in R_A,\ \mathrm{RowOK}(r, A) \wedge \mathrm{Block}(A) = \emptyset.$$

Here $\mathrm{supports}_f(e, r)$ means that extractor f reads locator content e, rather than model memory, and recovers support for the row. For any accepted row r, the staging rule requires such an evidence locator in the logged source set S. The audit rule then permits validation only from the locator content, not from the model's remembered facts. By induction over staged row updates, every row that passes the gate has such a witness, modulo extractor faithfulness and locator drift.

D Curated Evaluation Set

We curate 51 Seed2Frontier instances spanning 15 top-level domains, with the largest concentrations in geography (13 instances), culture (8), science (6), politics (4), sports (4), and business (3); the remaining nine domains (technology, religion, food, education, environment, health, history, law, society) contribute one to three instances each. Each instance is paired with a Wikipedia seed page, the labelled set of cited Wikipedia URLs (mean 10.1 pages per instance, median 6, range 1–35), and a primary-key ground-truth table (mean 68.2 rows per instance, median 24, range 3–526). We manually label the seed page for each instance as the most natural starting Wikipedia page for the query; the remaining labelled Wikipedia URLs form the complement set.

E Scoring Notes

Complement-page recall compares predicted source pages to the labelled complement set. Source-frontier precision and F1 use the same page set after Wikipedia-title normalization. Cardinality error is the absolute difference between emitted row count and labelled row count. Table-extraction F1 is computed over normalized primary-key overlap when a configuration emits rows.

F Severity Taxonomy Reference

Issue types are *factual* (a source-supported value is wrong or unsupported), *structural* (schema, primary key, type, duplicate, or artifact-contract problem), and *scope-related* (row-universe, partition, rank, cardinality, or exhaustiveness problem). Severity levels are *blocking*, *hygiene*, and *suggestion*. The crossing is deliberate: a scope-related blocker can make a table unusable even if every emitted row is factually correct, while a structural hygiene issue can be repairable without changing the row set.

G Cross-Model Generalization

We pick one closed-source model (gpt-5.4) and one open-source model (Llama 3.3 70B) as the two model families M, to test whether the policy contribution generalizes across release types and training pipelines.

Table 2: Vanilla LLM curator vs. Stage–Audit on the same 51-instance Seed2Frontier evaluation set with picked seeds, across one closed-source and one open-source model. Frontier metrics on the 45 instances with complement labels. Δ is Stage–Audit minus vanilla; sign-test p on per-instance F1 deltas.

	Vanilla LLM curator			Stage–Audit-governed			
Model	Rec.	Prec.	F1	Rec.	Prec.	F1	ΔF1 (sign-p)
gpt-5.4	0.384	0.356	0.334	**0.491**	**0.505**	**0.451**	+0.118 (0.014)
Llama 3.3 70B	0.329	0.344	0.311	0.362	0.356	0.319	+0.008 (0.21)

Stage–Audit's frontier-F1 improvement over the vanilla LLM curator is large and paired-significant on gpt-5.4 (+0.118, two-sided sign test p=0.029) and small but directionally consistent on Llama 3.3 70B (+0.008 F1, with positive deltas on all three frontier metrics). The pattern is consistent with the policy contribution being most effective on top of strong-instruction-following models, where the curator emits citation-tagged rows the auditor can then filter cleanly.

H Severity Taxonomy in the 182 Auditor Findings

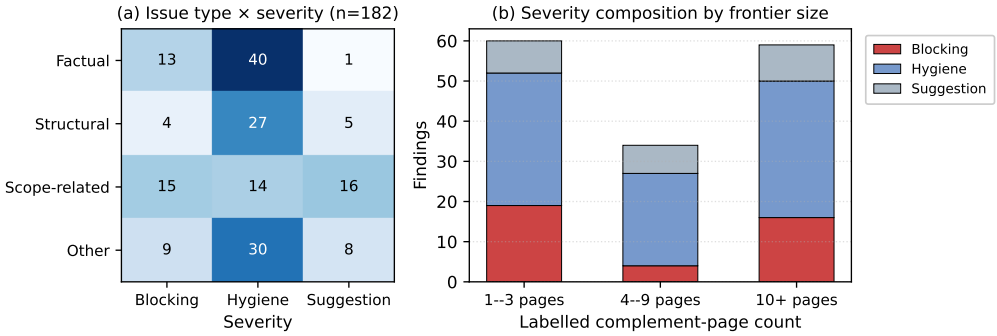

Figure 2: Severity taxonomy applied to the 182 Stage–Audit auditor findings on the 51-instance set with gpt-5.4. (a) Issue-type × severity heatmap: factual findings are dominated by the source-citation gate (n=40 hygiene, 13 blocking) confirming the gate fires on essentially every instance; scope-related findings are spread across blocking, hygiene, and suggestion, reflecting the harder-to-automate cardinality and partition checks. (b) Severity composition stratified by labelled complement-page count: every frontier-size bucket produces blocking findings, and the smallest-frontier bucket carries the highest blocking share, consistent with the Section 4 finding that the gate provides the most lift on small-frontier instances. Issue-type assignments use a conservative keyword classifier over the auditor's free-text `check` labels; the small "Other" bucket reflects labels that did not match the keyword set.

I Audit Check Enumeration

The 12-check audit taxonomy used in Section 3. Each check has a stable identifier; the auditor assigns one identifier per finding.

Factual. `F1` row-evidence (locator content supports row); `F2` source-URL well-formed and reachable; `F3` locator format and section/table reference valid.

Structural. `S1` primary-key uniqueness; `S2` primary-key non-null; `S3` column type conformance; `S4` column completeness against the artifact contract.

Scope-related. `P1` cardinality match against query expectation; `P2` partition coverage; `P3` source-role coverage (each page tagged with its admitting operator); `P4` temporal knowability; `P5` normalization consistency across rows.

J Repair-Loop Ablation

The body experiment caps Stage–Audit at a single curator-and-auditor pass to match the per-instance call budget of the unguarded baselines. To check whether additional repair-restage iterations would change the picture, we run gpt-5.4 with $k{=}0, \ldots, 5$ repair iterations on a 15-instance domain-stratified subset (one instance per top-level domain). Each iteration after $k{=}0$ feeds the auditor's findings back to the curator with a restage prompt and re-audits the new output; an instance converges if the auditor returns no blocking findings.

Table 3: Stage–Audit performance vs. repair-loop count k on a 15-instance domain-stratified subset (gpt-5.4 at $T{=}0$); $k{=}0$ is the body experiment. PK-F1 gains $+0.02$ at $k{=}1$ and plateaus.

k	Comp. recall	Frontier prec.	Frontier F1	PK F1	Frontier size
0	0.403	0.404	0.375	0.544	10.1
1	0.403	0.401	0.373	**0.565**	10.5
2	0.403	0.401	0.373	0.560	11.9
3	0.403	0.401	0.373	0.560	10.8

Stage–Audit's frontier metrics are stable through $k{=}3$ and show no improvement past the initial pass; PK-F1 receives a small one-shot lift at $k{=}1$ and then plateaus. Productive iteration would require either external retrieval or auditor signals naming candidate replacement pages, both deferred to future work.