When No Results Does Not Mean No Evidence: A Failure-Transparency Protocol for Scientific Tool Use

Davie Chen
Independent Researcher

21 October 2026
Research manuscript, version 2. Not peer reviewed.

Abstract

Scientific assistants increasingly gather evidence through tools that aggregate multiple external sources. A record list alone does not reveal whether an empty result reflects a completed search, incomplete coverage, or total upstream failure. This paper develops a failure-transparency protocol that separates record identity, request coverage, and claim support. We combine three evidence streams: an audit of eighteen dated website examples containing twenty-four displayed records across eighteen source identifiers; an exhaustive analysis of 117 abstract source-state configurations; and a small live metadata-check episode. All displayed records carry a URL and source field, but the shortened examples do not expose a per-source execution envelope, illustrating why record provenance and coverage are distinct audit properties. In the controlled enumeration, record-only and HTTP-aware policies correctly classify 28 and 31 configurations respectively; a coverage-aware reference classifies all 117 by construction. These are contract checks, not language-model performance measurements. During three read-only Crossref requests, two returned HTTP 429 while one returned a retraction-notice record, providing a concrete case in which unavailable metadata must not be interpreted as a missing publication. Fifty-two offline connector checks also passed using previously recorded responses; this does not measure current upstream reliability. Illustrated reference, dataset, literature, and archive examples motivate a three-layer reporting contract and a practical answer-level evaluation rubric. The study offers a reproducible interface-audit method, not a new retrieval model or evidence of a commercial service's superiority.

Keywords: scientific assistants; evidence provenance; tool failure; partial results; citation verification; retrieval evaluation

1 Introduction

An assistant helping with research may resolve a reference, find a dataset, check an editorial notice, retrieve a time series, or combine records from several repositories. The resulting answer depends not only on reasoning over returned text but also on whether the requested evidence was actually gathered. A successful application response can conceal unsuccessful source requests.

Consider a search over two selected sources. One completes with no matches and the other is rate-limited. The merged record list is empty, just as it would be if both searches completed with no matches. A summary saying that the search returned no results may sound innocuous, but without a coverage qualification it can invite an incorrect negative inference. Saying that no relevant work exists is stronger still and is not justified even by complete execution of a finite search.

This ambiguity affects routine research operations. An inaccessible retraction registry cannot establish that a paper has no notice. A failed repository query cannot establish that no dataset exists. Missing observations in an environmental series can reflect measurement coverage rather

than a zero value. These are related information-preservation problems, although their scientific interpretations differ.

The present paper asks how an interface should preserve the distinctions that a downstream assistant needs. **RQ1:** Which evidence properties can be audited in public tool examples? **RQ2:** What information is lost when responses expose records but omit request coverage? **RQ3:** How should real failures and partial results constrain final claims? Our focus is the tool boundary and its observable contract, not proprietary retrieval routing or model prompting.

We contribute a three-layer evidence model; an audit of eighteen historical, publicly presented capability examples; an exhaustive state-enumeration diagnostic with interpretable baselines; and a failure-inclusive live metadata-check vignette. Worked examples and a scoring rubric make the protocol operational. The empirical scope is deliberately bounded: no language-model comparison, user study, or production reliability estimate is claimed.

2 Background and related work

2.1 Scientific retrieval and tool use

OpenScholar studies retrieval-augmented synthesis of scientific literature [1]. ToolUniverse provides an environment for scientific tool use [2]. These efforts address important parts of evidence acquisition and scientific assistance. Our question is narrower: when a retrieval operation is incomplete, what information must survive the tool interface so that the next consumer can avoid overstating the evidence?

A standardized tool name or a common record schema can facilitate integration, but neither alone communicates request completion. Likewise, a high-quality synthesis model cannot observe source outcomes that were discarded before it received the response. The protocol proposed here is intended to complement retrieval and reasoning systems, not replace them.

2.2 Citation quality and hallucination

ALCE separates dimensions of citation-oriented generation evaluation, including correctness and citation quality [3]. RAGTruth provides annotated material for studying unsupported or contradictory generated statements under retrieval augmentation [4]. These lines of work motivate evaluating the relationship between an answer and its evidence rather than merely checking whether an answer contains links.

Our audit distinguishes a preceding interface problem. A URL can identify a real record while saying nothing about whether every selected source completed. A citation can be accurate bibliographically while failing to support an asserted claim. We therefore avoid collapsing record existence, coverage, and entailment into a single "verified" label. No scores from ALCE or RAGTruth are transferred to the present cases.

2.3 Provenance and protocol semantics

W3C PROV models entities, activities, agents, and derivation [5]. It provides a conceptual basis for recording how evidence was produced. Our compact response envelope is not a replacement for PROV and does not claim formal conformance; it identifies application-level observations needed for a bounded audit.

The Model Context Protocol describes tool results, structured content, and error signaling [6]. Protocol-level success does not, by itself, state that all internal upstream requests succeeded. A tool may legitimately return partial records, but that condition must be represented explicitly if consumers are expected to interpret it. We distinguish application coverage semantics from transport semantics rather than attributing the proposed contract to an existing protocol requirement.

Crossref's update documentation also illustrates why evidence relationships matter: a correction or retraction can be a separate record linked to the original work [7]. Finding the original DOI is not equivalent to checking all relevant editorial updates, and failure to retrieve a relationship is not evidence that none exists.

3 Materials, evidence boundaries, and methods

3.1 Historical website examples

We extracted the static examples used by ScholarMuse's capability pages from the available site source [9]. The catalog contains twenty examples. Two describe only a figure-generation response shape and are explicitly marked as such; they are excluded from empirical record counts. The remaining eighteen are described by the site's maintained source as shortened real calls retrieved on 13 October 2026.

These eighteen examples contain twenty-four displayed records and eighteen distinct source identifiers. A displayed record is the unit for auditing URL and source-field presence; a capability example is the unit for auditing the shown response envelope. The examples intentionally display only a subset of records, so a summary count need not equal the displayed array length. We preserve that distinction throughout the analysis.

The examples are vendor-selected demonstrations, not independent test queries. Their retrieval date is historical, not the date of this audit. Extracting them does not rerun a production tool or verify every current record field. We retain the request, shown response, example slug, tool name, and date to make the interpretation reproducible without copying private queries or credentials.

3.2 Offline connector checks

We ran five existing offline connector test files covering literature, grants/datasets, statistics, life sciences, and earth/astronomy/humanities. Their harness replaces network fetches with previously recorded upstream responses and substitutes a nonfunctional database stand-in. Fifty-two checks passed. They exercise parsing and mapping against recorded material rather than measuring live availability.

This evidence stream is separate from the website examples and from the abstract policy diagnostic. The tests are not fifty-two user questions or fifty-two independently verified scientific answers. Their purpose here is to establish a reproducible record-transformation check and illustrate the distinction between replay correctness and present-day retrieval success. The published analysis does not require production prompts or routing code.

3.3 Controlled enumeration and live vignette

The controlled study enumerates all source-state configurations for two, three, and four selected sources. It compares deterministic policies with differing observable information. No language model or paid inference service is called. The experiment is a specification diagnostic, not an evaluation of a deployed assistant.

Separately, on 21 October 2026 we made three read-only requests to Crossref's public work-metadata endpoint for identifiers appearing in the reference/retraction examples. HTTP outcomes, returned metadata where available, and response digests were retained. This small convenience episode is not an availability benchmark. We did not turn rate-limit responses into empty successful records, bypass limits, or run repeated requests until a successful result could replace the failure evidence.

Evidence stream	Units	Date	What it can establish
Historical website cases	18 examples / 24 records	13 Oct	Shown provenance and response structure
Offline replay checks	52 checks	21 Oct run	Mapping under recorded responses
Controlled enumeration	117 configurations	21 Oct	Information sufficiency under definitions
Live metadata vignette	3 requests	21 Oct	These request outcomes at this time

Table 1: Evidence streams are not pooled into a single sample size or product-accuracy estimate. Two response-shape-only examples are excluded from the historical record audit.

4 A three-layer evidence model

4.1 Identity, coverage, and support

The *record layer* identifies returned items: title, stable identifier, supplying source, retrieval time, and source-specific fields. It answers what the response contains. The *coverage layer* records intended and completed requests, including failures and not-attempted sources. It answers which parts of the selected search were actually executed. The *support layer* associates final claims with the records and fields or passages that justify them.

These layers are complementary. A record may have a resolving DOI but be irrelevant to the question. A response may have full selected-source completion but low retrieval recall. A claim may be appropriately supported by one source even when other sources failed, provided that its wording does not imply broader coverage. An audit should make each relation visible rather than declare an entire answer simply "grounded."

4.2 Source states and bounded negative conclusions

Let $S = \{s_1, \ldots, s_k\}$ be the sources selected for the current request. Each source has simplified state $x_i \in \{R, E, F\}$: completed with records, completed empty, or failed. Let u count completed sources and r count sources returning records. Define

$$y = \begin{cases} \text{failed}, & u = 0, \\ \text{partial}, & 0 < u < k, \\ \text{full-positive}, & u = k,\ r > 0, \\ \text{full-empty}, & u = k,\ r = 0. \end{cases} \tag{1}$$

Here "full" means completion of the selected requests, not exhaustive coverage of scientific literature. Each selected source may itself have indexing gaps, pagination limits, access restrictions, or query-translation errors. The model does not make those sources complete.

A full-empty outcome supports only a bounded statement: no matching records were returned by these queries, from these selected sources, at this time. Partial-empty means that the completed

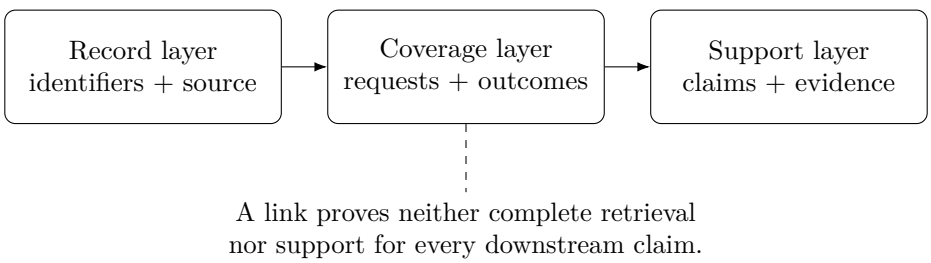

Figure 1: Three audit layers. This is a reporting model, not a disclosure of any service's private execution architecture.

Field group	Example information	Audit question
Request scope	Sources, query digest, time	What was intended?
Execution outcomes	Completed, failed, not attempted	What actually ran?
Result provenance	Source and identifier per item	Where did this record come from?
Coverage summary	Complete, partial, failed	Is the prose qualified?
Claim support	Record/field supporting each claim	Does evidence entail the statement?
Recovery status	Retryable/unknown, next action	Is uncertainty preserved?

Table 2: Minimal application-level contract. Fields can be represented differently; their meanings should remain explicit.

subset returned no records while the uncompleted subset remains unknown. All-source failure supports no completed-search conclusion. In each case the wording should reflect observed evidence rather than an assumed global absence.

Proposition 1 *A response exposing only a merged record list cannot generally distinguish full-empty, partial-empty, and failed multi-source outcomes.*

Proof. For two sources, (E, E), (E, F), and (F, F) all yield an empty list, while their reference outcomes differ. A deterministic classifier of that list must return one value for all three and cannot correctly distinguish them. □

This is an information-loss observation, not a claim that a new classifier outperforms existing assistants. A model can infer prior probabilities of failure, but a probability inferred from context is not an observation of the present request's execution.

4.3 A minimal failure-transparent envelope

An auditable response should include selected sources, per-source status, timestamps, returned-record provenance, and a summary consistent with those fields. More detailed error categories can distinguish rate limit, timeout, access denied, malformed content, and upstream service failure. Credentials, internal hostnames, private queries, and stack traces should not be exposed simply to make errors appear detailed.

Machine-readable and natural-language representations must agree. A partial flag combined with a confident complete-search summary remains misleading for a consumer who reads only the summary. Conversely, prose warnings without structured coverage complicate automated audits. We treat contradictions as a separate defect rather than accepting either representation as sufficient.

5 Audit of the historical website examples

5.1 Coverage of example types

Table 3 lists the eighteen eligible cases. They cover reference verification, editorial status, literature, journals, grants, datasets, statistics, biological records, clinical-trial metadata, compounds, environmental data, astronomy, and archives. This diversity illustrates varied provenance needs; it does not establish comprehensive coverage of those disciplines.

Every displayed record has a URL and a source field: 24/24 for each presence check. These are field-completeness observations, not current link-resolution or scientific-correctness rates. The eighteen shown replies have only `summary` and `records` as top-level keys. A per-source execution envelope is not visible in this shortened display.

The last observation must not be generalized into a claim that the production service omits coverage. The source file explicitly presents shortened examples. Documentation examples

No.	Capability example	Shown records	Shown source identifiers
01	verify references	2	crossref
02	check retractions	2	crossref
03	search papers	1	openalex
04	journal checker	1	openalex
05	find grants	1	nih-reporter
06	find datasets	1	dryad
07	statistics	1	fred
08	gene protein lookup	1	ncbi-gene
09	protein structures	1	pdb
10	clinical trials	1	clinicaltrials
11	drugs targets	2	chembl, openfda
12	compounds materials	1	pubchem
13	weather history	1	nasa-power
14	air quality	2	openaq
15	earthquake water	2	usgs
16	species occurrence	1	gbif
17	exoplanets stars	1	nasa-exoplanet
18	historical archives	2	europeana, loc

Table 3: Historical examples dated 13 October 2026. The displayed arrays are shortened; a source named only in a full-result summary need not appear among the shown records. Twenty-four displayed records span eighteen source identifiers.

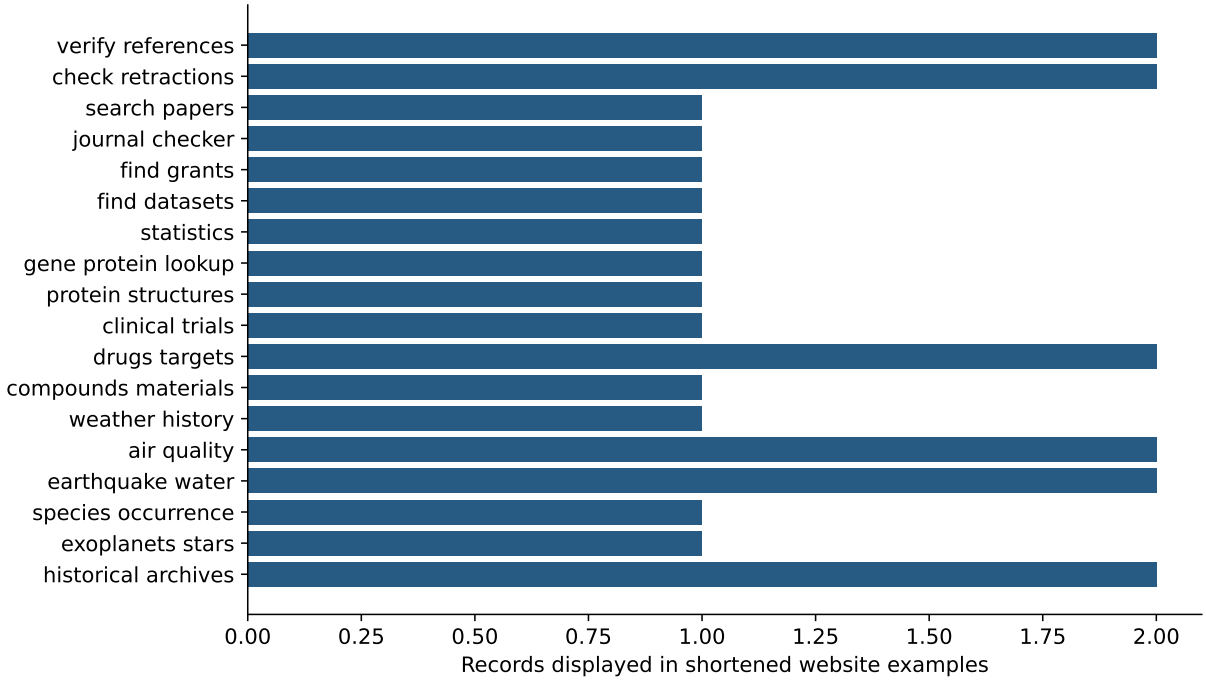

Figure 2: Displayed record counts, not returned totals or query success. Unequal bar heights reflect shortening choices in website examples.

can omit fields that actual responses include. The justified conclusion is that a reader cannot reconstruct request coverage from the displayed record arrays alone.

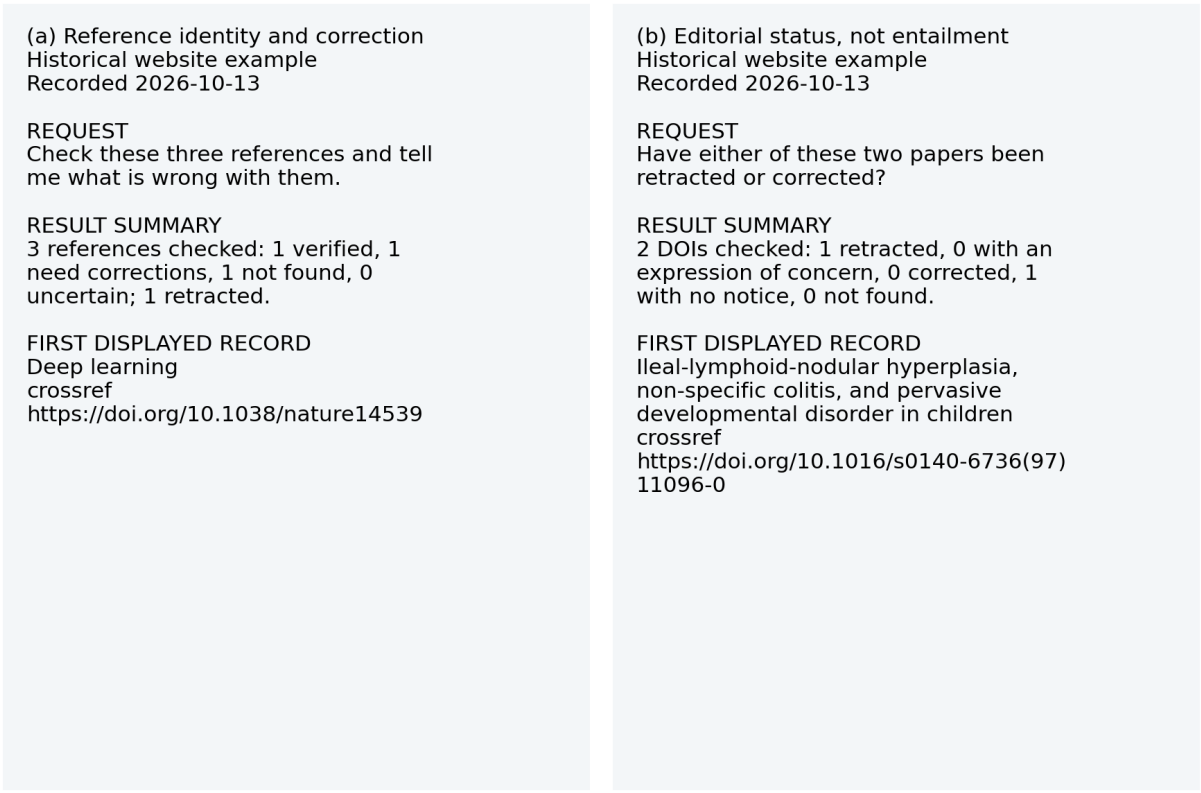

Figure 3: Evidence cards composed from actual dated website requests and shortened responses, not screenshots of new live calls. They illustrate bibliographic correction and editorial-status distinctions. No medical efficacy conclusion is drawn from these records.

5.2 Reference identity and editorial status

Figure 3 presents two actual historical examples as evidence cards. The reference-verification request includes an intentionally incorrect publication year and a deliberately nonexistent citation as diagnostic inputs. The example summary distinguishes a verified reference, a correction, and a not-found item. The fictitious input is not included in this paper's bibliography or treated as a real publication.

The retraction example separately identifies editorial status. This distinction matters: a bibliographically recognizable article can also be retracted. Crossref documents how an update notice can be registered as a separate work and linked to the original [7]. A retraction check should preserve the notice's identity and relationship rather than treating successful DOI resolution as an all-purpose validation.

The historical example reports a correction from 2014 to 2015 for the reference to *Deep learning*. Our fresh metadata request for that DOI was rate-limited, so this revision does not claim to have revalidated its current record through that request. The dated example and the failed current check remain distinct observations. This is the kind of temporal boundary a transparent assistant should preserve.

5.3 Dataset discovery and literature discovery

Figure 4 contrasts dataset and literature examples. The dataset summary reports five results across several repositories, while the shortened record array displays one Dryad item. The paper-search summary likewise reports five results while the shown array contains one item from OpenAlex. Counting the arrays as the total search results would incorrectly infer omissions or failures.

Dataset discovery also has two licensing layers. Permission to reuse metadata does not

Figure 4: Historical dataset and literature examples. Full-result summaries and shortened displayed arrays have different denominators. These cards preserve that distinction instead of reporting the display count as retrieval recall.

automatically establish permission to reuse the underlying dataset. The example summary explicitly distinguishes record-level licensing from a dataset's own license when stated. Similarly, an open-access indicator is useful metadata but does not establish that every desired full text can currently be fetched or redistributed.

5.4 Temporal coverage and rights in other domains

The air-quality example contains station provenance, units, daily statistics, and observation coverage percentages. A daily value with incomplete measurement coverage is not equivalent to a complete daily measurement, and a missing observation is not a zero concentration. This is a within-source coverage issue, distinct from whether a source request completed.

Figure 5 plots the actual numbers displayed for two London stations. The shortened example contains three daily observations per station even though the request spans seven days. The remaining days must be treated as not displayed, not imputed zeros and not evidence of failed source execution. For the first station, the displayed coverage is 88, 100, and 96 percent; the second station's three displayed values each have 100 percent coverage. This worked example separates request-window length, display truncation, and measurement coverage without inventing additional observations.

The archive example identifies records from the Library of Congress and Europeana and warns readers to inspect item rights. A link to an image record is not a license to reproduce the image. We therefore do not copy the historical photographs into this manuscript; the evidence audit concerns the record and its provenance. Domain-specific caution remains necessary even when every record is traceable.

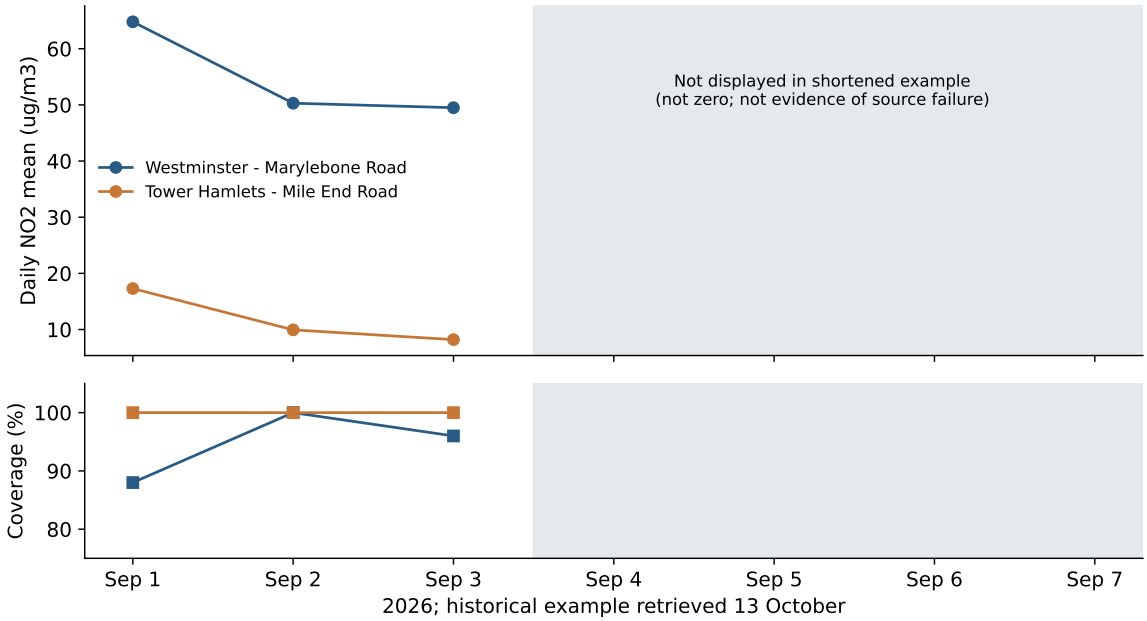

Figure 5: Actual values and observation coverage from the historical air-quality example, not newly measured concentrations. Shading denotes days absent from the shortened display, with no inference about the underlying complete response. Source records identify OpenAQ locations 154 and 151.

6 Exhaustive controlled analysis

6.1 Design and analytic expectations

For $k \in \{2, 3, 4\}$, the study enumerates every $\{R, E, F\}^k$ vector. This produces $3^2 + 3^3 + 3^4 = 117$ configurations. Each R source supplies one synthetic record; none represents a real paper. The reference class counts are twenty-five full-positive, three full-empty, eighty-six partial, and three failed.

For a fixed k, full-positive configurations number $2^k - 1$, because every source must complete and at least one must return a record. There is one full-empty configuration and one all-failed configuration. The remainder, $3^k - 2^k - 1$, are partial. These expressions provide an independent check on the enumerator. They also explain why partial states occupy a growing portion of the enumerated space.

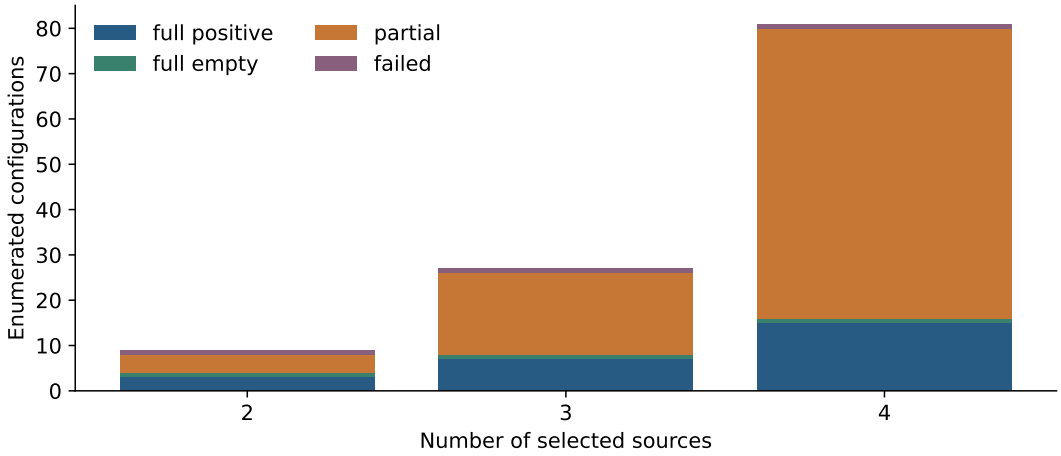

Figure 6: Reference outcomes across exhaustive state spaces. Frequencies are combinatorial counts under equal enumeration, not observed provider-failure probabilities.

Policy	Correct	Accuracy (%)	False empty	Partial disclosed
recordOnly	28/117	23.9	25	0/86
httpAware	31/117	26.5	22	0/86
coverageAware	117/117	100.0	0	86/86

Table 4: 117 abstract configurations. False-empty predictions label partial or failed states as full-empty. The coverage-aware row is correct by definition, not evidence of trained-model performance.

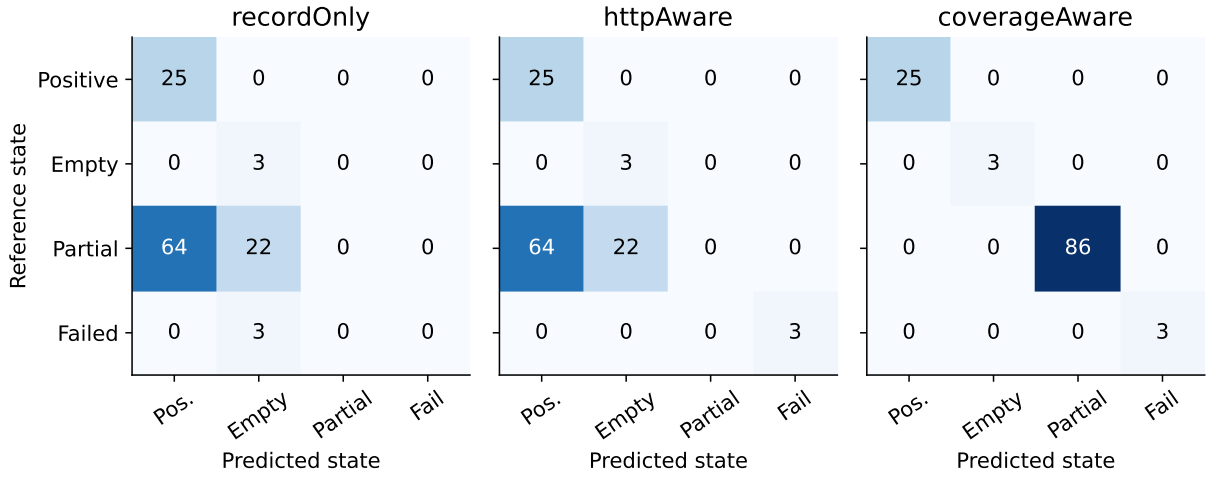

Figure 7: Exact confusion counts. Rows are reference states; columns are policy predictions. Coverage-aware classification reproduces the reference rule, while record-only and HTTP-aware policies collapse partial states.

Three deterministic policies receive different observations. *Record-only* predicts full-positive from a nonempty list and full-empty from an empty list. *HTTP-aware* additionally recognizes all-source failure under an experimental status convention, but receives no partial-coverage detail. *Coverage-aware* implements the reference rule from selected and completed-source counts. This status convention is part of the diagnostic, not a universal property of HTTP or MCP.

6.2 Results and confusion structure

Record-only predicts twenty-five false-empty outcomes and discloses none of the eighty-six partial states. HTTP awareness corrects the three all-source failures but leaves twenty-two partial-empty cases mislabeled. The remaining partial configurations can contain records and are consequently mislabeled full-positive. Looking only for false-empty statements would therefore miss a large class of incomplete positive answers.

The coverage-aware policy classifies all 117 cases correctly because it directly implements the specification. Its apparent advantage should not be read as a production accuracy gain. The study does not measure whether a language model reads the envelope, whether summaries remain consistent under generation, or whether users make better decisions. Those questions require a separate experiment.

Equal weighting over configurations is also not a model of operational traffic. Provider failures may be correlated, queries may select different source sets, and successful sources may return many records rather than one. Statistical significance tests on this exhaustive deterministic enumeration would not turn it into evidence of real-world effect size.

States	Reference outcome	Appropriate interpretation
E, E	Full-empty	Both selected searches completed; no matches returned.
E, F	Partial-empty	One search returned no matches; another failed.
F, F	Failed	No selected source completed; existence remains unknown.
R, F	Partial-positive	Records available from one source; other coverage incomplete.
R, E	Full-positive	Selected requests completed; returned records still need relevance checks.

Table 5: Illustrative interpretations. Even full selected-source completion is not an exhaustive literature review.

6.3 Five worked interpretation cases

Table 5 makes the state distinction concrete. The proposed wording is illustrative protocol language, not output attributed to a production assistant. Partial-positive answers should identify the records they do have without implying complete retrieval. Full-empty answers should qualify the query scope rather than assert that a scientific phenomenon or literature does not exist.

7 A live metadata-check failure vignette

7.1 Observed requests

The three fresh Crossref requests produced the outcomes in Table 6. The requests for the original reference examples returned HTTP 429 with no metadata body. The request for the retraction notice returned HTTP 200 and a notice record whose `update-to` relationship identified the original article as a retraction target. These are actual request observations, not injected failures.

7.2 Why the failure must remain visible

If the two 429 outcomes were normalized to successful empty results, the resulting table would appear to say that the requested records were not found. That conclusion is unsupported: rate limiting prevented the metadata check. The successful notice request also demonstrates that separate related records can have different retrieval outcomes within the same short episode.

We do not report "two-thirds of Crossref requests fail" as a reliability estimate. The sample is tiny, selected, concurrent, and affected by the requesting environment. Nor do we claim that ScholarMuse generated an incorrect answer in this episode; the requests were direct public metadata checks outside its production service. The vignette establishes the practical possibility of the ambiguity modeled by the protocol.

The preserved HTTP status is sufficient to reject a completed-empty interpretation, even though the rate-limit response contains no record metadata. This is useful negative evidence about the execution, not negative evidence about publication existence. A careful follow-up would wait or use an appropriate permitted retrieval route, with any later result logged as a new observation rather than silently replacing the failed attempt.

Requested DOI	HTTP	Observation	Permissible conclusion
`10.1038/nature14539`	429	No metadata	This request could not verify the record.
`10.1016/S0140-6736(97)11096-0`	429	No metadata	Original-work metadata unavailable in this request.
`10.1016/S0140-6736(10)60175-4`	200	Notice record	Returned metadata links a retraction notice to the original DOI.

Table 6: Read-only public metadata requests on 21 October 2026. This three-request convenience episode does not estimate Crossref availability or a service error rate.

8 Evaluating downstream scientific answers

8.1 A claim-level rubric

The response contract is necessary for truthful coverage reporting but not sufficient for a correct scientific answer. We propose evaluating four separately scored dimensions: identity, coverage, support, and consistency. Table 7 gives concrete questions. A high score on identifier resolution cannot compensate for an unsupported substantive claim.

For support judgments, distinguish metadata assertions from scientific assertions. A retrieved year can support a statement about a record's year. An abstract snippet may not support an effect-size claim requiring full-text methods and results. A retraction notice supports an editorial-status observation, not the validity of every surrounding medical assertion. These distinctions are especially important when a tool combines heterogeneous records under one schema.

8.2 A paired fault-injection design

A future assistant study can freeze a set of independently checkable tasks and present matched successful and faulty tool responses to the same model. Relevant faults include rate limits, timeouts, malformed payloads, partial pagination, stale caches, and contradictory fields. The query, model version, prompt, tool budget, response ordering, and scoring procedure should be controlled or explicitly randomized.

Measure unsupported negative claims, partial-failure disclosure, correct bounded abstention, evidence support, latency, and cost separately. Disclosure should not be rewarded when it merely repeats boilerplate while making an unsupported claim elsewhere. Conversely, a qualified partial answer should not be penalized as though it confidently asserted a complete result. Independent reviewers and a disagreement process are needed for nontrivial support judgments.

Repeated conditions within a task are paired observations, not independent questions. If several tasks use one template or one source record, the analysis should retain that clustering. Failure rates from an injected distribution should be reported as conditional sensitivity, not operational prevalence. The proposed design is a future empirical extension, not a study already completed in this manuscript.

Dimension	Scoring question	Failure example
Identity	Is the cited record the intended item?	Correct title, wrong DOI
Coverage	Does wording reflect completed requests?	Timeout called no results
Support	Does cited evidence justify this claim?	Real paper cited for unrelated conclusion
Consistency	Do prose and structured fields agree?	Partial flag, complete-search summary
Temporal scope	Is the observation date preserved?	Historical status presented as freshly checked

Table 7: Proposed answer-level rubric. No human annotation or language-model benchmark under this rubric is claimed in the present study.

9 Discussion

9.1 Provenance is visible more often than coverage

The website examples show that ordinary record provenance can be compact: all twenty-four displayed records carry a source and URL. Yet the shown arrays do not identify every selected source's execution outcome. These findings are compatible, not contradictory. Provenance describes where present records came from; coverage also describes what was attempted but did not contribute a record.

This distinction explains why adding more links is not a complete remedy. An answer can cite every returned item accurately and still conceal that a major selected repository failed. Conversely, a partial response can contain enough support for a narrow claim. The important question is whether the answer's scope matches the evidence, not whether it contains an arbitrary number of citations.

9.2 Replay, documentation, and live observation

The three non-identical empirical views each have value. Historical examples make behavior inspectable and explainable. Offline replay makes mapping checks repeatable without paid calls or unstable upstream access. Live requests expose present execution conditions, including failures. None can replace the others.

A replay pass cannot prove that a provider is currently reachable. A live successful request cannot validate all mappings. A shortened documentation example cannot establish the full production response contract. Reporting these boundaries prevents a study from assembling an impressive-looking but incoherent aggregate success count.

9.3 Confidentiality without methodological opacity

A black-box study need not release private prompts, model routing, account tokens, proprietary ranking, or customer queries. It should still release the tested inputs or safe fixtures, observation dates, response contract, scoring rules, and failure-inclusive counts. Synthetic faults should be clearly marked, and public metadata should retain source attribution and applicable rights information.

This manuscript uses source-visible examples, public identifiers, synthetic configurations, and offline recorded-response checks. It does not access customer records or mutate production services. Public release of accompanying artifacts remains separate from preparing this local manuscript; no DOI, repository release, or third-party endorsement is asserted.

10 Limitations and threats to validity

Selected examples. The eighteen website cases were curated by an author-affiliated service. They may emphasize successful operations and are not representative of users or a field. The twenty-four records are shortened displays, not a random retrieval sample, and not every field was revalidated live.

Simplified states. The $R/E/F$ model omits not-attempted sources, partially completed pagination, cache freshness, schema drift, and within-source observation gaps. These should be represented in a deployed envelope. Adding them would enlarge the state space but not remove the underlying ambiguity of record-only responses.

No behavioral effect estimate. The reference policy is correct by construction. No experiment shows that a particular language model uses the coverage information correctly, and no human study establishes improved research decisions. Those are necessary next tests, not implied consequences of a clean schema.

Live-request scope. Three public metadata requests are a vignette, not a reliability survey. Shared infrastructure and concurrency may affect rate limits. The manuscript reports outcomes without assigning a population frequency or attributing them to a particular commercial product.

Authorship and affiliation. The author develops ScholarMuse and has a commercial interest in the study object. The source audit is not independent. The paper therefore limits its conclusions to inspectable observations and protocol diagnostics rather than product effectiveness or superiority.

11 Conclusion

Scientific tool responses should preserve the difference between returned evidence, completed retrieval, and claim support. The historical example audit demonstrates compact record provenance while exposing the limits of shortened displays. Exhaustive enumeration explains why record-only and simple failure-status policies collapse partial outcomes. The live rate-limit episode shows why a failed verification must not become an absence claim. Together these observations support a failure-transparent reporting protocol whose practical effect still requires independent answer-level evaluation.

Declarations

This research manuscript has not undergone peer review. The author develops the case-study service and discloses that commercial interest. AI assistance supported drafting, coding, and figure preparation; the author remains responsible for verification and interpretation. No human participants, customer data, paid model inference, or production modifications were used. The reference/retraction cases concern metadata and editorial relationships, not medical advice or an assessment of clinical efficacy. No indexing, acceptance, or public artifact deposit is claimed.

A Reproduction and evidence files

The supplied enumeration script generates every source-state vector, policy prediction, aggregate result, and diagnostic table. The website-example extraction preserves the static request/response examples and their historical dates. The figure-generation script audits field presence, identifies response-shape-only cases, and produces the example cards and charts. Cards are compositions of existing example text, not fabricated screenshots or newly executed production calls.

The public-check script records three metadata request outcomes and response digests. It does not retry away failure evidence. The offline replay command covers the five named connector-test groups and passes fifty-two checks using recorded data. The seven independent controlled-study assertions verify enumeration counts, expected formulas, denominator rules, and evidence hashes. These checks are distinct from scientific claim-support annotation.

B An audit record template

A future audit should retain a task identifier, safe query representation, selected-source list, start/end times, source-specific outcome, record identifiers, response version, and summary. It should record whether the response was live, cached, replayed, shortened, or illustrative. An answer annotation should link each material claim to a record and a field or passage, then classify support as sufficient, insufficient, contradicted, or unresolved.

When a check fails, the audit should retain the failure category without exposing secrets. If a later request succeeds, it becomes a separate observation with its own date. Summaries should report completed, failed, and not-attempted denominators explicitly. This template is intended to make uncertainty reviewable without requiring access to private execution internals.

References

[1] A. Asai et al. OpenScholar: Synthesizing Scientific Literature with Retrieval-augmented LMs. 2024. arXiv:2411.14199.

[2] S. Gao et al. ToolUniverse: An open platform for democratizing AI scientists. 2025, revised 2026. arXiv:2509.23426.

[3] T. Gao, H. Yen, J. Yu, and D. Chen. Enabling Large Language Models to Generate Text with Citations. EMNLP, 2023. arXiv:2305.14627.

[4] C. Niu, Y. Wu, J. Zhu, S. Xu, K. Shum, R. Zhong, J. Song, and T. Zhang. RAGTruth: A Hallucination Corpus for Developing Trustworthy Retrieval-Augmented Language Models. 2024. arXiv:2401.00396.

[5] L. Moreau and P. Missier, editors. PROV-DM: The PROV Data Model. W3C Recommendation, 30 April 2013. `https://www.w3.org/TR/prov-dm/`.

[6] Model Context Protocol. Server features: Tools. Specification dated 25 November 2025. `https://modelcontextprotocol.io/specification/2025-11-25/server/tools`. Accessed 21 October 2026.

[7] Crossref. Registering updates. `https://www.crossref.org/documentation/register-maintain-records/maintaining-your-metadata/registering-updates/`. Accessed 21 October 2026.

[8] Crossref. REST API, work metadata. `https://api.crossref.org/`. Three request observations preserved in the accompanying evidence files, 21 October 2026.

[9] ScholarMuse. Public scientific-tool capability examples. `https://scholarmuse.ai/`. Historical example retrieval date 13 October 2026; source audit 21 October 2026. Author-affiliated material, not independent effectiveness evidence.