RE^{2}MATH: BENCHMARKING THEOREM RETRIEVAL IN RESEARCH-LEVEL MATHEMATICS

Anonymous authors
Paper under double-blind review

ABSTRACT

Mathematical proofs often use known theorems. We introduce Re^{2}Math, a benchmark for retrieving theorems needed in unfinished research proofs. Given the global setup and a local proof prefix, a large language model (LLM) uses external search to retrieve a theorem statement with its source. The source lets researchers check the statement and credit prior work. The retrieved theorem must justify the intended next step using the available facts. The task allows suitable alternative theorems and sources. We build a dataset from published papers across five mathematical domains. Across the tested models, most outputs that pass the grounding check do not pass the reference-based sufficiency check. Case studies show that a theorem may assume a fact the proof still needs to establish, while a conclusion weaker than the reference may suffice for the next step. Evaluation must therefore check how the theorem applies to the intended step, including whether the theorem concerns the objects in the proof. We release the dataset and code for construction and evaluation.

1 INTRODUCTION

Problem-solving accuracy is a common measure of progress in mathematical AI. Existing benchmarks cover school and competition problems (Cobbe et al., 2021; Hendrycks et al., 2021; He et al., 2024) as well as research-level questions (Glazer et al., 2024; Zhang et al., 2025). AI has also helped mathematicians identify patterns and formulate conjectures (Davies et al., 2021). We study theorem retrieval as a form of research assistance: using external search to find a theorem and its source for the next step of an unfinished proof.

Different proofs of the same theorem may use different auxiliary results. A proof prefix records choices already made and narrows the retrieval need to results that can advance the current argument. Understanding a proof involves reconstructing its plan (Hamami & Morris, 2024). Following another proof may require learning a new argument without resolving the step at hand. Mathematicians seek help with subproblems and partial arguments (Tausczik et al., 2014), and their information needs depend on goals and prior knowledge (Zhao et al., 2008).

Retrieving one theorem with its source gives researchers a specific statement to check against the literature (Rashkin et al., 2023; Gao et al., 2023). They can then examine whether its assumptions hold and its conclusion can justify the intended step. Access to the theorem's proof also helps them understand and reuse the result, while the citation credits prior work (Tao, 2026).

We construct Re^{2}Math from citations used within published proofs. The model receives the global setup and a local proof prefix. The withheld continuation specifies the intended step, and the cited result provides a reference statement. The model must retrieve a theorem that justifies this step from the available facts. Alternative theorems and sources are allowed, provided they justify the intended step within the current argument.

Our empirical study evaluates seven large language models (LLMs) under the same search budget. Most outputs judged supported by their sources do not pass the reference-based sufficiency check. We examine these outcomes through oracle-source evaluation, separate planning and query diagnostics, and cases from the original proofs.

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.

The cases distinguish problems with theorem application from problems with evaluation. One transfer theorem receives positive judgments despite assuming the bound that the proof still needs to establish. In a graph case, the returned conclusion can establish the required embedding but is rejected for lacking the reference's stronger counting guarantee. In another case, the judge compares results for different equations as though only their time ranges differed.

Our contributions are:

- **Task definition.** We formulate theorem retrieval from an unfinished proof: the model infers the needed result and retrieves a theorem that justifies the intended next step.
- **Benchmark.** We build Re^{2}Math from published papers across five mathematical domains, with proof contexts, reference statements, cited sources, and documented construction checks.
- **Empirical study.** We evaluate seven LLMs and distinguish errors in applying a theorem from limitations of reference-based judging.

2 Task Definition

2.1 Proof context and intended step

Let s be an ordered *proof sequence* for a target theorem. For a retrieval instance i, fix $0 \le k < T$ and write

$$s = (s_1, \dots, s_T), \qquad s_{1:k} = (s_1, \dots, s_k).$$

Here $s_{1:k}$ is the completed proof prefix, and Δ_i is the proposition to be established at the *intended proof step* s_{k+1}. We use $s_{a:b} = (s_a, \dots, s_b)$, with the empty sequence when $a > b$.

The available *proof context* consists of a global setup and a visible suffix of the prefix:

$$\mathcal{C}_i = (\mathcal{C}^i_{\text{global}}, \mathcal{C}^i_{\text{local}}), \qquad \mathcal{C}^i_{\text{local}} = s_{\ell:k}, \quad 1 \le \ell \le k+1.$$

The global setup specifies the mathematical objects, definitions, and target theorem. Let Γ_i be the assumptions and previously established statements available from $\mathcal{C}_i$ at s_{k+1}, with their scopes respected. The unproved target conclusion is excluded.

The continuation $s_{k+1:T}$ and the proposition Δ_i are hidden. The proposition Δ_i is fixed, but need not be uniquely determined by $\mathcal{C}_i$.

2.2 Theorem Retrieval

Let $\mathcal{D}_i$ be a collection of admissible sources and $\mathcal{K}(d)$ the set of theorem statements supported by source d.

A theorem K is sufficient for the intended step when

$$\Gamma_i \cup \{K\} \vdash \Delta_i,$$

where $\vdash$ denotes derivability in a fixed sound background proof system. The derivation may establish intermediate facts and use K through one or more instantiations. The conditions of each application must hold at its point of use; they need not be stated explicitly in $\mathcal{C}_i$.

The set of *suitable theorem–source pairs* is

$$\mathcal{V}_i = \big\{(d, K) : d \in \mathcal{D}_i, \quad K \in \mathcal{K}(d), \quad \Gamma_i \cup \{K\} \vdash \Delta_i\big\}.$$

Given $\mathcal{C}_i$ and access to $\mathcal{D}_i$, the task is to retrieve a source $\hat{d}_i$ and theorem $\hat{K}_i$ such that $(\hat{d}_i, \hat{K}_i) \in \mathcal{V}_i$. Different pairs may justify the same Δ_i.

This is a logical sufficiency criterion: the derivation need not use K, and membership in $\mathcal{V}_i$ does not measure the proof work saved.

3 Benchmark Construction

Following Section 2, each proof block s_t is a sentence, displayed equation, or mathematical environment. Citations identify candidate retrieval steps s_{k+1}. Figure 1 shows the construction and evaluation pipeline; Appendix A gives the full protocol.

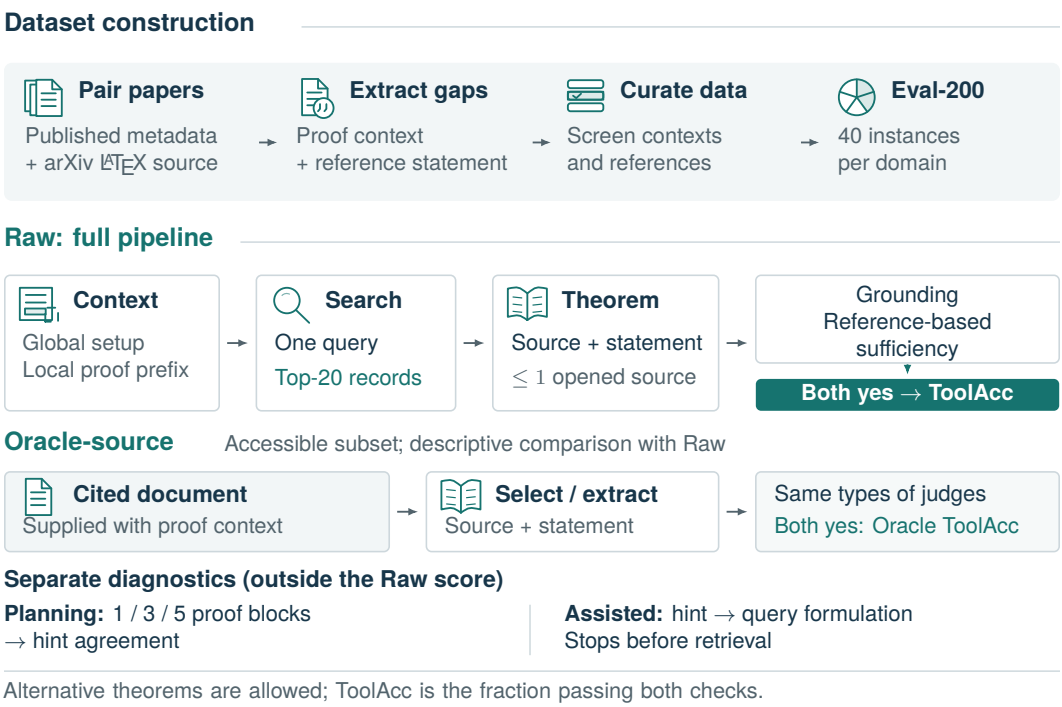


Figure 1: **Overview of Re^{2}Math.** Raw runs the full retrieval pipeline; oracle-source supplies the cited document. Both report ToolAcc, which requires positive grounding and reference-based sufficiency labels. Separate planning and Assisted diagnostics evaluate need identification and query formulation.

3.1 From Published Papers to Retrieval Data

We select published papers whose proofs cite external mathematical results. Using publication metadata, we match each paper to an arXiv version and obtain its LaTeX source. Extraction first locates the setup, target proof, and candidate citations in each paper. For each citation, it then records the preceding context, a hint describing the missing result's role, a reference statement, and citation metadata.

We check for ambiguous paper matches, incorrectly placed proof gaps, incomplete reference statements, and hints that identify sources. We withhold the citation-associated continuation.

Context and withheld information. The **citing paper** contains the proof, and the **cited source** is the work named by its citation. The input presents premises and derived facts within the proof text. The withheld citation-associated continuation fixes Δ_i, while the invoked result provides the reference statement $K_i^\star$. The reference statement, cited-source metadata, and continuation are withheld from the model in both Raw and Assisted conditions. A short **hint**, called an *anchor* in the release, describes the missing result's role. It is intended to omit source names, theorem titles, citation keys, DOIs, and arXiv identifiers; residual leakage is assessed below. Only Assisted gives the model this hint.

Reference and intended step. The reference statement $K_i^\star$ records one proposed justification of Δ_i. A different source and theorem may form a pair in $\mathcal{V}_i$, even if the theorem is weaker than or incomparable to $K_i^\star$. The release stores $K_i^\star$, while Δ_i is specified by the withheld continuation. Section 5.3 gives a case where that continuation supplies information missing from the visible prefix.

Worked example: supplying a missing limit. In a stored instance from Jego (2021), the proof studies random measures constructed from planar Brownian motion. The global setup defines the regularized measures m_ε^γ and the critical approximation μ_ε. The visible prefix fixes $\gamma_n \in [1, 2)$ tending to 2 and establishes a comparison between $(2-\gamma_n)^{-1} m_\varepsilon^{\gamma_n}(A)$ and $2\mu_\varepsilon(A)$, for the Borel set A used in the proof.

The intended next step needs the subcritical limit for each fixed n:

$$m_{\varepsilon}^{\gamma_n}(A) \xrightarrow[\varepsilon \to 0]{\text{in probability}} m^{\gamma_n}(A).$$

The withheld sentence invokes the convergence result of Jego (2020), which applies for every fixed $\gamma \in (0, 2)$. Substituting $\gamma = \gamma_n$ supplies this step and lets the proof use the limit in its comparison. This example paraphrases the stored context and continuation of `2005.14610v2_gap_1`.

3.2 Dataset Statistics and Quality Checks

The release contains 860 proof-gap instances from 631 distinct papers, with arXiv versions counted as one paper. It spans geometry/topology, analysis/PDE, probability/statistics/control, algebra/number theory, and combinatorics/discrete mathematics. Eval-200 contains 40 instances per domain, from 162 distinct citing papers under the same counting rule. Selection requires nonempty local context, a complete reference statement, and a journal-paper cited source. Eligible candidates are ranked by quality, with a per-paper cap. The evaluation set is therefore balanced by domain and filtered for quality. Figure 2 gives the release statistics.

A single annotator audited 120 benchmark instances for construction quality. Citation role passes in 110/120 (91.7%), hint leakage control in 98/120 (81.7%), and citation metadata in 116/120 (96.7%). Context placement, reference sufficiency, and external-source dependence each pass in 120/120. Context placement checks that the prefix precedes the citation; reference sufficiency is a judgment against the supplied context. Alignment with the original continuation is examined separately through the cases in Section 5.4. Appendix A.6 gives the audit criteria and scope. Hints that reveal source information can affect the Assisted diagnostics and the references used to judge planning descriptions. Raw inputs omit the hints.

Released resources. The anonymous release provides the dataset and split, construction and evaluation code, prompts, aggregate results, and a source-access summary. It omits local query caches and per-output judgments (Section 4.3). Changes to evaluated instances require a new release and updated results.

4 Experiments

We first evaluate complete retrieval runs and then examine individual parts of the task. Oracle-source evaluation supplies the cited document to examine theorem selection. Planning and query diagnostics examine how the model identifies and expresses the needed result (Ribeiro et al., 2020). Case analysis then checks the theorem's application and the judge's decision against the original proof.

4.1 Evaluation Design

End-to-end evaluation. In **Raw**, the model receives the global setup and up to five preceding local proof blocks (shorter prefixes as available). It states the needed result in a **planning description**, writes one Scholar query, inspects Top-20 metadata, opens at most one source, and returns $(\hat{d}_i, \hat{K}_i)$. Here, *theorem* includes lemmas, propositions, corollaries, estimates, and criteria; the statement may be extracted verbatim or faithfully restated. The query prompt asks for 3–5 keywords within five words. The context remains available throughout. Appendix A.2 gives the window indexing; Appendix B.1 specifies stage inputs.

Oracle-source evaluation. The Top-20 results depend on the model's query and the search engine's coverage and ranking, and may contain no source with a suitable theorem. Our **oracle-source** condition supplies the cited document directly and asks the model to select a theorem from it, bypassing document search (Liu et al., 2024). Cited sources were retrieved and parsed for 67–74 instances per run. Section 5.2 reports the source-access rate, *Oracle ToolAcc* on the accessible subsets, and the percentage-point differences from Raw on all 200 instances.

Planning and query diagnostics. Before searching, the model must decide what mathematical conclusion the current proof step needs. A query can match the topic yet miss that conclusion. Separate planning runs elicit a description without bibliographic identifiers. We compare its mathematical

objects, intended use, and required conclusion with the stored hint; agreement is an indirect check of whether the model has identified the needed result. We vary the preceding text over one, three, or five blocks to examine dependence on the visible proof context. **Assisted** supplies the hint and evaluates the resulting query without running search. Appendix C reports full diagnostics, domain results, and uncertainty estimates.

Case analysis. We inspect historical outputs against their original proof continuations to examine theorem application, reference alignment, and judge decisions.

4.2 Evaluation Protocol and Metrics

We use LLM judges (Zheng et al., 2023) to score source support and reference-based sufficiency separately, following KILT's joint evaluation of provenance and task performance (Petroni et al., 2021). The grounding judge checks whether the source supports the statement (Rashkin et al., 2023). The sufficiency judge asks whether the returned statement is equivalent to or stronger than the reference and whether the returned theorem's assumptions hold in the proof context. Their labels are J_i^G and J_i^S. For a model evaluated on M instances, ToolAcc is the fraction receiving positive labels from both judges:

$$\mathrm{ToolAcc} = \frac{1}{M}\sum_{i=1}^{M} \mathbf{1}[J_i^G = J_i^S = \texttt{yes}].$$

Uncertain, missing, non-evaluable, and abstained outcomes stay in the denominator and receive no credit. Appendix B.1 gives the diagnostic metric definitions.

LLM-based evaluation and task success. ToolAcc summarizes LLM judgments; $\mathcal{V}_i$ specifies mathematical task success. The sufficiency judge uses $K_i^\star$ as its comparison target. This comparison can reject a weaker or incomparable theorem that establishes Δ_i. Grounding assesses support in source excerpts; source admissibility requires a separate check. Reference misalignment and judge errors can also produce false positives. We therefore interpret ToolAcc as a score under this evaluation protocol. Appendices B and D.8 give prompts and human checks. We also report:

- $\mathrm{GroundRate}$: the fraction receiving positive grounding judgments, regardless of sufficiency.
- $\mathrm{CiteRecall@}N$: the fraction whose first N retrieved records contain the original cited source, after source-identity matching.
- $\mathrm{AnchorAcc}$: agreement between the planning description and the stored hint, over instances with a hint.

AnchorAcc uses separate planning runs; GroundRate and CiteRecall use all end-to-end instances.

4.3 Common Setting

Eval-200 gives the five domains equal weight. The remaining 660 instances are unused for training, prompt tuning, or model selection. Future training and development splits must separate citing papers from Eval-200 (Appendix A).

Models. We evaluate seven models: GPT-5.2, Gemini 3.1 Pro, Claude Opus 4.5, DeepSeek V3.2, Qwen3-235B Thinking, Kimi K2 Thinking, and Grok 4. Appendix B.2 lists the exact released model identifiers. The configuration sets reasoning effort to `medium`.

Retrieval and source text. Google Scholar searches across publication types (Appendix B.3), with no lower year bound and the citing paper's year as the upper bound. All models use the same query and source budget and shared text extraction. We cache Top-20 metadata, ranks, query times, and access outcomes to replay the observed searches. The cache records the candidates returned for each model's queries at the time of execution.

Source admissibility. The admissible collection $\mathcal{D}_i$ is constrained by the search rules above. For the citing paper and its published or arXiv variants, the selected text must independently state or prove the invoked result outside the held-out argument.

Judging and human checks. The 120-instance construction audit checks input and reference quality (Section 3.2); output audits inspect evaluation errors. The main tables reproduce public aggregate scores; the public configuration specifies GPT-5.4 (`openai/gpt-5.4`) as judge. The 175-output audit and case studies use earlier predictions, without a released per-output link to the main scoring run. One annotator confirms AI-assisted judgments on outputs sampled by model and outcome. These records document historical evaluation errors; calibration of the configured judge requires human judgments on the same outputs. Appendices B.2 and D.8 give provenance and findings.

5 Results and Analysis

Table 1: **Planning and end-to-end retrieval on Eval-200.** AnchorAcc uses separate planning runs; the other columns use Raw outputs. All four score columns use 200 instances, with counts in parentheses. The final ratio is the fraction of outputs passing the grounding check that also pass the reference-based sufficiency check (Section 4.2).

	Planning		**Raw end-to-end retrieval**						
Model	**AnchorAcc**		**CiteRecall @20**		**GroundRate**		**ToolAcc**		**ToolAcc / GroundRate**
Claude Opus 4.5	24.0%	(48)	10.5%	(21)	24.0%	(48)	7.0%	(14)	29.2%
Grok 4	26.5%	(53)	4.5%	(9)	41.5%	(83)	3.5%	(7)	8.4%
Kimi K2 Thinking	43.5%	(87)	6.0%	(12)	24.0%	(48)	3.5%	(7)	14.6%
GPT-5.2	5.5%	(11)	8.5%	(17)	25.0%	(50)	3.0%	(6)	12.0%
DeepSeek V3.2	8.0%	(16)	5.5%	(11)	29.0%	(58)	2.5%	(5)	8.6%
Gemini 3.1 Pro	2.0%	(4)	5.5%	(11)	12.0%	(24)	2.0%	(4)	16.7%
Qwen3-235B Thinking	40.5%	(81)	6.0%	(12)	17.0%	(34)	1.0%	(2)	5.9%

5.1 Source Grounding and Reference-Based Sufficiency

Across the seven models, only 5.9%–29.2% of outputs that pass the grounding check also pass the reference-based sufficiency check (Table 1). In the recorded Raw runs, Grok receives more positive grounding labels than Claude (83 versus 48 of 200), yet fewer outputs pass both checks (7 versus 14). Section 5.4 examines theorem applications and evaluation decisions in selected historical outputs.

The highest $\mathrm{ToolAcc}$ is 7.0% (14/200); the other runs score 1.0%–3.5%. The Wilson 95% interval (Wilson, 1927) for 14/200 is [4.2%, 11.4%], and the paper-cluster interval is [3.6%, 10.7%] (Table 6). The small counts yield broad intervals, with substantial overlap for several model pairs.

Cited-source matches. Between 75% and 100% of each model's outputs that pass both checks select a source not matched to the cited source (Table 8). Matching the cited source measures citation recovery, while the two judges assess the returned statement. Requiring a match to the original citation would exclude most outputs that pass both current checks. For unmatched sources, admissibility remains a separate check.

The archived Tulenbaev case illustrates retrieval without a selected source (Appendix D.1). Several queries name Tulenbaev and Steinberg groups; Claude queries "Steinberg group Tulenbaev lifting property." In all seven Raw runs, the cited source is absent from Top-20 and no source is selected. Grounding and sufficiency labels are missing or non-evaluable. These runs end before theorem extraction, even though the queries contain specific mathematical terminology. The oracle-source condition below examines theorem selection once a cited document is supplied.

5.2 Oracle-Source Evaluation

With the cited document supplied, Oracle ToolAcc ranges from 1.4% to 12.7% (Table 2). A negative or uncertain judgment may reflect an unsuitable theorem, a mismatch with the reference, or a judge error.

Table 2: **Oracle-source versus Raw.** Raw uses all 200 instances; Oracle uses instances with retrieved and parsed cited documents. The access rate is relative to Eval-200. Difference is Oracle minus Raw in percentage points, computed from unrounded fractions.

Model	Source-access rate		Raw ToolAcc	Oracle ToolAcc		Difference (pp)
Claude Opus 4.5	35.5%	(71/200)	7.0%	12.7%	(9/71)	+5.7
Grok 4	37.0%	(74/200)	3.5%	2.7%	(2/74)	−0.8
Kimi K2 Thinking	37.0%	(74/200)	3.5%	1.4%	(1/74)	−2.1
GPT-5.2	35.5%	(71/200)	3.0%	7.0%	(5/71)	+4.0
DeepSeek V3.2	37.0%	(74/200)	2.5%	2.7%	(2/74)	+0.2
Gemini 3.1 Pro	33.5%	(67/200)	2.0%	7.5%	(5/67)	+5.5
Qwen3-235B Thinking	37.0%	(74/200)	1.0%	2.7%	(2/74)	+1.7

Oracle ToolAcc exceeds Raw ToolAcc for five models; the differences range from -2.1 to $+5.7$ percentage points. Cited sources were retrieved and parsed for 74 instances in at least one run and for 66 in every run; the remaining 126 were unavailable in all completed runs (Table 9). The comparison is descriptive: Raw and Oracle use different evaluation sets and prompts, and the released aggregates do not support a matched-subset comparison. The cited document may also help the model infer the intended step.

5.3 Identifying and Expressing the Needed Result

More preceding proof text did not consistently improve agreement with the stored hint in our runs. Separate planning runs retain the global setup and vary the local window over $m \in \{1, 3, 5\}$ blocks on the same 200 instances per model (Figure 3). With one, three, and five blocks, Kimi's descriptions match 89, 104, and 87 hints, while Qwen's match 83, 66, and 81. For every model, at least one shorter window scores higher than the five-block window. Each setting uses one run, scored for agreement with the stored hints.

Many queries fail the alignment check even when the hint is supplied: Assisted $\mathrm{QueryAcc}$ is 11.5%–55.6% (Table 10). The judge assesses query alignment using the reference statement and cited-source metadata. Negative labels may reflect errors in the query, hint, or judgment.

Ambiguity in the proof context. The archived compactification instance shows that a prefix can identify a topic without specifying the result needed next (Appendix D.2). Its Raw local prefix ends with an unfinished sentence about monodromy. The needed matroidal-locus identity is not specified by the global setup and becomes explicit only in the withheld continuation. GPT and Claude extract it when given the cited source, while their Raw queries can follow the visible monodromy discussion without targeting that identity.

The window experiment varies the amount of preceding text. Here, the information identifying the intended step lies in the withheld continuation. Supplying the cited document may help identify the needed result as well as provide its statement. Oracle-source evaluation can therefore change both document access and the information available for interpreting the proof context.

5.4 Theorem Application and Reference Comparison

We examine theorem applications and evaluation decisions in selected historical outputs, using the original proofs to identify the intended steps.

Theorem application: missing premises. A theorem may depend on the very fact being sought. In the Boolean-to-tropical case, the proof needs a monotone Boolean lower bound for the logical permanent (Appendix D.4). The returned transfer theorem describes how to use that bound:

$$\underbrace{\text{Boolean lower bound}}_{\text{missing premise}} \xrightarrow{\text{transfer theorem}} \text{tropical lower bound.}$$

Both archived judges give this transfer theorem positive labels; the historical audit identifies it as a false positive. The transfer theorem belongs after the missing lower bound in this argument: it explains what follows from that bound, whereas the retrieved result must justify that bound.

In the sphere instance, the proof reduces an energy bound to $p = 2$ and introduces positive definite functions (Appendix D.3). The cited source appears in all seven Raw candidate lists. Qwen returns a theorem requiring smoothness, convexity, and boundary conditions not established in the displayed proof; the grounding label is positive and the sufficiency label negative. Claude returns the Schoenberg characterization through nonnegative Gegenbauer coefficients and receives both positive labels. The two theorems require different hypotheses for use in the same proof.

Reference requirement: a stronger conclusion than needed. In the graph instance (Appendix D.5), the global target asks for polynomially many cliques. The visible prefix assumes a spectral threshold on G and observes that a fixed graph F, with $\chi(F) = k + 1$, lies in $K_{k+1}[t]$ for some fixed t. This is the complete $(k + 1)$-partite graph with t vertices per part. The hidden continuation asks to find a copy of F in G, so Δ_i is $F \subseteq G$:

$$F \subseteq K_{k+1}[t] \subseteq G \quad \Longrightarrow \quad F \subseteq G.$$

The reference demands a stronger clique-counting guarantee. An archived GPT output returns a blow-up theorem with logarithmically growing parts, but receives a negative sufficiency label.

Let $V_1, \ldots, V_{k+1}$ be the independent classes of a proper coloring of fixed F. Mapping each class to a different part gives

$$a_j \geq |V_j| \quad (j = 1, \ldots, k + 1) \quad \Longrightarrow \quad F \subseteq K_{a_1,\ldots,a_{k+1}}.$$

Every edge of F is present between the corresponding parts. Since F is fixed, logarithmically growing parts suffice for large n.

The blow-up containment alone does not imply the counting guarantee. For fixed $k \geq 1$, let $r = \lfloor \log n \rfloor$ and pad $K_{r,\ldots,r}$, with $k + 1$ parts, to n vertices using isolated vertices. It contains fixed F for large n, but only $r^{k+1} = o(n^{k+1})$ copies of K_{k+1}. This example compares the conclusions alone; the padded graph need not satisfy the spectral threshold.

The returned conclusion supplies the second containment without the counting guarantee. A judgment can follow the reference rule correctly yet reject a conclusion sufficient for the local step. The theorem's hypotheses remain to be checked for this application.

Judge decision: comparing different mathematical objects. In the equation case (Appendix D.6), the global target concerns fractional heat, while the local context and reference concern a Kolmogorov equation. A retrieved Kolmogorov theorem covers the required small-time non-controllability result. The archived judge nevertheless accepts a fractional-heat theorem as stronger because it covers a longer time range. A longer time range for fractional heat does not establish the required Kolmogorov result.

Checking which equation each result concerns would expose the judge error in the equation case. A judge that follows the reference rule correctly would still reject the graph conclusion for lacking the counting guarantee; accepting that conclusion requires evaluating the local step itself. Evaluating an explicit theorem application would require identifying the mathematical objects, checking the hypotheses, and verifying that the relevant consequence establishes the intended step.

Source admissibility. When a model selects the citing paper, the selected passage must independently state or prove the result outside the held-out argument (Appendix B.4). Source admissibility remains unresolved for the historical outputs discussed here (Appendix D.8).

6 Related Work

Theorem retrieval and premise selection. NaturalProofs retrieves proof references for natural-language theorems (Welleck et al., 2021). In formal mathematics, DeepMath learns to rank premises, while LeanDojo, LeanSearch, and Magnushammer retrieve statements from proof-assistant libraries (Alemi et al., 2016; Yang et al., 2023; Gao et al., 2024; Mikuła et al., 2023). MIRB combines

natural-language and formal retrieval and identifies useful unannotated premises as a source of false negatives (Ju & Dong, 2025).

TheoremSearch searches nine million research-level mathematical results from a description of a desired theorem (Alexander et al., 2026). TheoremQA supplies theorem-dependent questions; BRIGHT's TheoremQA-T subset uses them to evaluate reasoning-dependent retrieval (Chen et al., 2023; Su et al., 2025). In Re^{2}Math, the model infers the needed result from a proof prefix and returns a statement that must be supported by its source and justify the next step.

Document retrieval and citation recommendation. ReAct interleaves reasoning with actions that gather information from external sources (Yao et al., 2023). BrowseComp and BrowseComp-Plus study web fact finding and fixed-corpus search (Wei et al., 2025; Chen et al., 2026). BEIR evaluates retrieval across heterogeneous text collections (Thakur et al., 2021); SPECTER and SciRepEval study scientific document representations across downstream tasks (Cohan et al., 2020; Singh et al., 2023). Context-aware citation recommendation retrieves papers for a passage (He et al., 2010; Färber & Jatowt, 2020). After selecting a paper, the model must still find a result sufficient to derive the intended step from the current proof context.

Scientific literature assistance and proof benchmarks. Scientific claim verification requires finding evidence for a stated claim, as in SciFact and SciFact-Open (Wadden et al., 2020; 2022). PaperQA and OpenScholar use retrieved literature to answer questions (Lála et al., 2023; Asai et al., 2024), while KILT and ALCE evaluate retrieval and attribution (Petroni et al., 2021; Gao et al., 2023). Source support also matters in mathematical research: Aletheia reports fabricated references without search and misstatements of real results despite tool use (Feng et al., 2026). Formal proof completion and formalization are assessed by miniF2F, ProofNet, and PutnamBench (Zheng et al., 2021; Azerbayev et al., 2023; Tsoukalas et al., 2024).

7 Limitations

Evaluation and calibration. To assess the reliability of these scores, independent experts should review theorem–source pairs that pass both checks and a stratified sample of the remaining pairs, including borderline cases and alternative theorems. Public per-output records would let others inspect these judgments. A matched human baseline would require multiple domain experts, common time and search budgets, and separate records of expertise and literature familiarity.

Coverage and access. The benchmark favors published proofs with explicit citations and recoverable arXiv source. Eval-200 is a curated, domain-balanced sample. Construction checks identify citation-role errors and residual hint leakage. The one-query, one-source budget is narrower than normal research browsing; search coverage, ranking, text access, and parsing all affect the available evidence.

Local scope. The task fixes the intended step through the continuation. The compactification case shows that the visible prefix can leave this step ambiguous. Natural-language judging errors and published-source memorization remain concerns; evaluation on newly published papers would help assess memorization.

8 Conclusion

Re^{2}Math studies theorem retrieval for the next step of an unfinished proof. ToolAcc remains low under the tested protocols. Applying a retrieved theorem requires establishing its conditions and showing that its conclusion justifies the intended step. Comparison with the reference can reject a conclusion that suffices for the local step. Future work should pair retrieved theorems with explicit justifications of how they establish the intended step, and evaluate these justifications while checking source support.

AI USE STATEMENT

AI tools assisted with polishing the manuscript, including improving clarity, grammar, and consistency of expression. We also used LLMs for dataset construction and evaluation, as detailed in Appendices A and B. The authors are responsible for the final manuscript and the reported results.

REPRODUCIBILITY STATEMENT

Section 4 specifies the common search and source-access budget. Appendices A and B describe the dataset construction, prompts, source matching, and scoring protocol. Appendix C reports supplementary experiments; Appendix D gives cases and evaluation audits. Comparisons require the same dataset release and recorded retrieval and scoring artifacts; new queries can require a new search snapshot.

REFERENCES

Alex A. Alemi, François Chollet, Niklas Een, Geoffrey Irving, Christian Szegedy, and Josef Urban. DeepMath – deep sequence models for premise selection. *arXiv preprint arXiv:1606.04442*, 2016.

Luke Alexander, Eric Leonen, Sophie Szeto, Artemii Remizov, Ignacio Tejeda, Jarod Alper, Giovanni Inchiostro, and Vasily Ilin. Semantic search over 9 million mathematical theorems. *arXiv preprint arXiv:2602.05216*, 2026. URL https://arxiv.org/abs/2602.05216v2. Version 2.

Akari Asai, Jacqueline He, Rulin Shao, Weijia Shi, Amanpreet Singh, Joseph Chee Chang, Kyle Lo, Luca Soldaini, Sergey Feldman, Mike D'Arcy, David Wadden, Matt Latzke, Minyang Tian, Pan Ji, Shengyan Liu, Hao Tong, Bohao Wu, Yanyu Xiong, Luke Zettlemoyer, Graham Neubig, Dan Weld, Doug Downey, Wen-tau Yih, Pang Wei Koh, and Hannaneh Hajishirzi. OpenScholar: Synthesizing scientific literature with retrieval-augmented LMs. *arXiv preprint arXiv:2411.14199*, 2024. URL https://arxiv.org/abs/2411.14199.

Zhangir Azerbayev, Bartosz Piotrowski, Hailey Schoelkopf, Edward W. Ayers, Dragomir Radev, and Jeremy Avigad. ProofNet: Autoformalizing and formally proving undergraduate-level mathematics. *arXiv preprint arXiv:2302.12433*, 2023.

Wenhu Chen, Ming Yin, Max Ku, Pan Lu, Yixin Wan, Xueguang Ma, Jianyu Xu, Xinyi Wang, and Tony Xia. TheoremQA: A theorem-driven question answering dataset. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 7889–7901, 2023. doi: 10.18653/v1/2023.emnlp-main.489. URL https://aclanthology.org/2023.emnlp-main.489/.

Zijian Chen, Xueguang Ma, Shengyao Zhuang, Ping Nie, Kai Zou, Sahel Sharifymoghaddam, Andrew Liu, Joshua Green, Kshama Patel, Ruoxi Meng, Mingyi Su, Yanxi Li, Haoran Hong, Xinyu Shi, Xuye Liu, Hosna Oyarhoseini, Nandan Thakur, Crystina Zhang, Luyu Gao, Wenhu Chen, and Jimmy Lin. BrowseComp-Plus: A fair and disentangled evaluation benchmark for deep search agents. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 22349–22370, 2026. doi: 10.18653/v1/2026.acl-long.1023. URL https://aclanthology.org/2026.acl-long.1023/.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Arman Cohan, Sergey Feldman, Iz Beltagy, Doug Downey, and Daniel S. Weld. SPECTER: Document-level representation learning using citation-informed transformers. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 2270–2282, 2020. doi: 10.18653/v1/2020.acl-main.207. URL https://aclanthology.org/2020.acl-main.207/.

Alex Davies, Petar Veličković, Lars Buesing, Sam Blackwell, Daniel Zheng, Nenad Tomašev, Richard Tanburn, Peter Battaglia, Charles Blundell, András Juhász, Marc Lackenby, Geordie Williamson, Demis Hassabis, and Pushmeet Kohli. Advancing mathematics by guiding human intuition with AI. *Nature*, 600:70–74, 2021. doi: 10.1038/s41586-021-04086-x. URL https://www.nature.com/articles/s41586-021-04086-x.

Michael Färber and Adam Jatowt. Citation recommendation: Approaches and datasets. *International Journal on Digital Libraries*, 21(4):375–405, 2020. doi: 10.1007/s00799-020-00288-2.

Tony Feng, Trieu H. Trinh, Garrett Bingham, Dawsen Hwang, Yuri Chervonyi, Junehyuk Jung, Joonkyung Lee, Carlo Pagano, Sang-hyun Kim, Federico Pasqualotto, Sergei Gukov, Jonathan N. Lee, Junsu Kim, Kaiying Hou, Golnaz Ghiasi, Yi Tay, YaGuang Li, Chenkai Kuang, Yuan Liu, Hanzhao Lin, Evan Zheran Liu, Nigamaa Nayakanti, Xiaomeng Yang, Heng-Tze Cheng, Demis Hassabis, Koray Kavukcuoglu, Quoc V. Le, and Thang Luong. Towards autonomous mathematics research. *arXiv preprint arXiv:2602.10177*, 2026. URL https://arxiv.org/abs/2602.10177v3. Version 3.

FutureSearch, Nikos I. Bosse, Jon Evans, Robert G. Gambee, Daniel Hnyk, Peter Mühlbacher, Lawrence Phillips, Dan Schwarz, and Jack Wildman. Deep research bench: Evaluating AI web research agents. *arXiv preprint arXiv:2506.06287*, 2025. URL https://arxiv.org/abs/2506.06287.

Guoxiong Gao, Haocheng Ju, Jiedong Jiang, Zihan Qin, and Bin Dong. A semantic search engine for Mathlib4. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 8001–8013, 2024. doi: 10.18653/v1/2024.findings-emnlp.470. URL https://aclanthology.org/2024.findings-emnlp.470/.

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. URL https://aclanthology.org/2023.emnlp-main.398/.

Elliot Glazer, Ege Erdil, Tamay Besiroglu, Diego Chicharro, Evan Chen, Alex Gunning, Caroline Falkman Olsson, Jean-Stanislas Denain, Anson Ho, Emily de Oliveira Santos, et al. FrontierMath: A benchmark for evaluating advanced mathematical reasoning in AI. *arXiv preprint arXiv:2411.04872*, 2024.

Tilmann Gneiting. Strictly and non-strictly positive definite functions on spheres. *Bernoulli*, 19(4): 1327–1349, 2013. doi: 10.3150/12-BEJSP06. URL https://arxiv.org/abs/1111.7077.

Google AI for Developers. Gemini Deep Research agent, 2026. URL https://ai.google.dev/gemini-api/docs/deep-research. Accessed September 18, 2026.

Yacin Hamami and Rebecca Lea Morris. Understanding in mathematics: The case of mathematical proofs. *Noûs*, 58(4):1073–1106, 2024. doi: 10.1111/nous.12489. URL https://onlinelibrary.wiley.com/doi/10.1111/nous.12489.

Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, et al. OlympiadBench: A challenging benchmark for promoting AGI with olympiad-level bilingual multimodal scientific problems. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3828–3850, 2024. doi: 10.18653/v1/2024.acl-long.211. URL https://aclanthology.org/2024.acl-long.211/.

Qi He, Jian Pei, Daniel Kifer, Prasenjit Mitra, and C. Lee Giles. Context-aware citation recommendation. In *Proceedings of the 19th International Conference on World Wide Web*, pp. 421–430, 2010. doi: 10.1145/1772690.1772734.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the MATH dataset. *arXiv preprint arXiv:2103.03874*, 2021.

Antoine Jego. Planar Brownian motion and Gaussian multiplicative chaos. *The Annals of Probability*, 48(4):1597–1643, 2020. URL https://arxiv.org/abs/1812.01903.

Antoine Jego. Critical Brownian multiplicative chaos. *Probability Theory and Related Fields*, 180 (1–2):495–552, 2021. doi: 10.1007/s00440-021-01051-7. URL https://arxiv.org/abs/2005.14610.

Haocheng Ju and Bin Dong. MIRB: Mathematical information retrieval benchmark. *arXiv preprint arXiv:2505.15585*, 2025. URL https://arxiv.org/abs/2505.15585v1. Version 1.

Jakub Lála, Odhran O'Donoghue, Aleksandar Shtedritski, Sam Cox, Samuel G. Rodriques, and Andrew D. White. PaperQA: Retrieval-augmented generative agent for scientific research. *arXiv preprint arXiv:2312.07559*, 2023. URL https://arxiv.org/abs/2312.07559.

Minghao Li, Ying Zeng, Zhihao Cheng, Cong Ma, and Kai Jia. ReportBench: Evaluating deep research agents via academic survey tasks. *arXiv preprint arXiv:2508.15804*, 2025. URL https://arxiv.org/abs/2508.15804.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173, 2024. doi: 10.1162/tacl_a_00638. URL https://aclanthology.org/2024.tacl-1.9/.

Maciej Mikuła, Szymon Tworkowski, Szymon Antoniak, Bartosz Piotrowski, Albert Qiaochu Jiang, Jin Peng Zhou, Christian Szegedy, Łukasz Kuciński, Piotr Miłoś, and Yuhuai Wu. Magnushammer: A transformer-based approach to premise selection. *arXiv preprint arXiv:2303.04488*, 2023.

OpenAI. Introducing deep research, 2025. URL https://openai.com/index/introducing-deep-research/. Accessed September 18, 2026.

Fabio Petroni, Aleksandra Piktus, Angela Fan, Patrick Lewis, Majid Yazdani, Nicola De Cao, James Thorne, Yacine Jernite, Vladimir Karpukhin, Jean Maillard, Vassilis Plachouras, Tim Rocktäschel, and Sebastian Riedel. KILT: a benchmark for knowledge intensive language tasks. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 2523–2544, 2021. doi: 10.18653/v1/2021.naacl-main.200. URL https://aclanthology.org/2021.naacl-main.200/.

Hannah Rashkin, Vitaly Nikolaev, Matthew Lamm, Lora Aroyo, Michael Collins, Dipanjan Das, Slav Petrov, Gaurav Singh Tomar, Iulia Turc, and David Reitter. Measuring attribution in natural language generation models. *Computational Linguistics*, 49(4):777–840, 2023. doi: 10.1162/coli_a_00486. URL https://aclanthology.org/2023.cl-4.2/.

Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. Beyond accuracy: Behavioral testing of NLP models with CheckList. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 4902–4912, 2020. doi: 10.18653/v1/2020.acl-main.442. URL https://aclanthology.org/2020.acl-main.442/.

Mark Sanderson. Test collection based evaluation of information retrieval systems. *Foundations and Trends in Information Retrieval*, 4(4):247–375, 2010. doi: 10.1561/1500000009.

Amanpreet Singh, Mike D'Arcy, Arman Cohan, Doug Downey, and Sergey Feldman. SciRepEval: A multi-format benchmark for scientific document representations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 5548–5566, 2023. doi: 10.18653/v1/2023.emnlp-main.338. URL https://aclanthology.org/2023.emnlp-main.338/.

Hongjin Su, Howard Yen, Mengzhou Xia, Weijia Shi, Niklas Muennighoff, Han-yu Wang, Haisu Liu, Quan Shi, Zachary S. Siegel, Michael Tang, Ruoxi Sun, Jinsung Yoon, Sercan Ö. Arık, Danqi Chen, and Tao Yu. BRIGHT: A realistic and challenging benchmark for reasoning-intensive retrieval. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/7a0f8055c838df8e62329a76c7c6403d-Paper-Conference.pdf.

Terence Tao. Mathematics in the age of AI. *arXiv preprint arXiv:2608.16753*, 2026. doi: 10.48550/arXiv.2608.16753. URL https://arxiv.org/abs/2608.16753.

Yla R. Tausczik, Aniket Kittur, and Robert E. Kraut. Collaborative problem solving: A study of MathOverflow. In *Proceedings of the 17th ACM Conference on Computer Supported Cooperative Work & Social Computing*, pp. 355–367, 2014. doi: 10.1145/2531602.2531690. URL https://ylataus.github.io/files/TausczikKitturKraut2014.pdf.

Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. BEIR: A heterogeneous benchmark for zero-shot evaluation of information retrieval models. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1, 2021. URL https://datasets-benchmarks-proceedings.neurips.cc/paper/2021/hash/65b9eea6e1cc6bb9f0cd2a47751a186f-Abstract-round2.html.

George Tsoukalas, Jasper Lee, John Jennings, Jimmy Xin, Michelle Ding, Michael Jennings, Amitayush Thakur, and Swarat Chaudhuri. PutnamBench: Evaluating neural theorem-provers on the Putnam mathematical competition. In *Advances in Neural Information Processing Systems*, volume 37, pp. 11545–11569, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/1582eaf9e0cf349e1e5a6ee453100aa1-Abstract-Datasets_and_Benchmarks_Track.html.

David Wadden, Shanchuan Lin, Kyle Lo, Lucy Lu Wang, Madeleine van Zuylen, Arman Cohan, and Hannaneh Hajishirzi. Fact or fiction: Verifying scientific claims. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 7534–7550, 2020. doi: 10.18653/v1/2020.emnlp-main.609. URL https://aclanthology.org/2020.emnlp-main.609/.

David Wadden, Kyle Lo, Bailey Kuehl, Arman Cohan, Iz Beltagy, Lucy Lu Wang, and Hannaneh Hajishirzi. SciFact-Open: Towards open-domain scientific claim verification. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pp. 4719–4734, 2022. doi: 10.18653/v1/2022.findings-emnlp.347. URL https://aclanthology.org/2022.findings-emnlp.347/.

Jason Wei, Zhiqing Sun, Spencer Papay, Scott McKinney, Jeffrey Han, Isa Fulford, Hyung Won Chung, Alex Tachard Passos, William Fedus, and Amelia Glaese. BrowseComp: A simple yet challenging benchmark for browsing agents. *arXiv preprint arXiv:2504.12516*, 2025. URL https://arxiv.org/abs/2504.12516.

Sean Welleck, Jiacheng Liu, Ronan Le Bras, Hannaneh Hajishirzi, Yejin Choi, and Kyunghyun Cho. NaturalProofs: Mathematical theorem proving in natural language. *arXiv preprint arXiv:2104.01112*, 2021.

Edwin B. Wilson. Probable inference, the law of succession, and statistical inference. *Journal of the American Statistical Association*, 22(158):209–212, 1927. doi: 10.1080/01621459.1927.10502953.

Kaiyu Yang, Aidan Swope, Alex Gu, Rahul Chalamala, Peiyang Song, Shixing Yu, Saad Godil, Ryan J Prenger, and Animashree Anandkumar. LeanDojo: Theorem proving with retrieval-augmented language models. In *Advances in Neural Information Processing Systems*, volume 36, pp. 21573–21612, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/4441469427094f8873d0fecb0c4e1cee-Abstract-Datasets_and_Benchmarks.html.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. ReAct: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations*, 2023. URL https://arxiv.org/abs/2210.03629.

Jie Zhang, Cezara Petrui, Kristina Nikolić, and Florian Tramèr. RealMath: A continuous benchmark for evaluating language models on research-level mathematics. In *Advances in Neural Information Processing Systems*, volume 38, pp. 88960–88991, 2025. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/734abb86d3caa949f44da8a093717f61-Abstract-Datasets_and_Benchmarks_Track.html.

Jin Zhao, Min-Yen Kan, and Yin Leng Theng. Math information retrieval: User requirements and prototype implementation. In *Proceedings of the 8th ACM/IEEE-CS Joint Conference on Digital Libraries*, pp. 187–196, 2008. doi: 10.1145/1378889.1378921. URL https://www.comp.nus.edu.sg/~kanmy/papers/fp096-zhao.pdf.

Kunhao Zheng, Jesse Michael Han, and Stanislas Polu. miniF2F: a cross-system benchmark for formal Olympiad-level mathematics. *arXiv preprint arXiv:2109.00110*, 2021.

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*, volume 36, pp. 46595–46623, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/91f18a1287b398d378ef22505bf41832-Abstract-Datasets_and_Benchmarks.html.

A Dataset Construction Protocol

This appendix describes paper selection, extraction of proof contexts, preparation of the release, and construction checks. Appendix B gives the evaluation protocol.

A.1 Published-Paper Selection and arXiv Pairing

Published metadata identifies each paper and its venue; the matched source provides theorem boundaries, proof text, and citations without relying on OCR of publisher PDFs.

The manifest records two linked identities for every accepted paper. The *published identity* stores title, authors, venue, year, publication date, DOI, and related bibliographic metadata. The *extraction identity* stores the arXiv identifier, source URL, matched title, match type, and match confidence. We first try an exact DOI match, then a high-confidence match on title, authors, and year, followed by manual verification. Ambiguous pairs go to the review queue rather than into the release.

A.2 Two-Stage Mining Pipeline

Stage I: paper-level proof scouting. Given the matched arXiv source, a compact bibliography map, and a target-theorem hint, the scout extracts the global setup, the target theorem's proof span, and citations that appear to supply external results within that proof. The scout is instructed to exclude historical remarks, background citations, definitions, and citations outside the proof.

Stage II: citation-level instance extraction. The second stage processes one retained citation occurrence at a time. It instantiates $s = (s_1, \dots, s_T)$ from Section 2.1 as ordered proof blocks, each a sentence, displayed equation, or mathematical environment. A citation in block s_c fixes $k = c - 1$; Δ_i is the citation-associated inference at s_{k+1}, not necessarily the entire block. For a window of at most m blocks, choose

$$\ell = \max\{1, k - m + 1\}, \qquad \mathcal{C}^i_{\mathrm{local}}(m) = s_{\ell:k}, \qquad \mathcal{C}_i = (\mathcal{C}^i_{\mathrm{global}}, \mathcal{C}^i_{\mathrm{local}}(m)).$$

The local context contains only available preceding blocks and is empty when $k = 0$. The main evaluation uses $m = 5$, and window diagnostics also use $m \in \{1, 3\}$. Write a_i for the stored hint and d_i^{cite} for the cited-source metadata. The extraction produces the release triple

$$(\mathsf{x}_i, \mathsf{y}_i, \mathsf{z}_i) = \big((\mathcal{C}^i_{\mathrm{global}}, \mathcal{C}^i_{\mathrm{local}}, a_i), K_i^\star, d_i^{\mathrm{cite}}\big)$$

for the task in Section 2; the hint is omitted from Raw inputs. If the citation cannot be localized reliably, the candidate is dropped. We do not backfill missing local context from unrelated parts of the proof.

A.3 Hint Construction and Leakage Controls

The hint a_i is produced by combining an LLM-generated planning description, citation-local heuristics, implication-style rewrites from the locator snippet, and local-context heuristics. The hint should describe the mathematical role of the missing result without including theorem titles, author names, citation keys, source titles, DOI strings, arXiv identifiers, or other bibliographic shortcuts. Manual repairs are applied only to remove leakage, repair malformed text, or make the intended proof move grammatically clear.

A.4 Construction Prompt Templates

The summaries below specify the construction prompts' roles, inputs, rules, and outputs. The released code can invoke fallback variants after parsing failures or incomplete extractions; these variants retain the same requirement to recover cited mathematical content without inventing it.

C1. Stage-1 structure and citation scouting

Role. Extract one benchmark instance from one mathematical paper. Return JSON only.
Input payload. A compact bibliography map and a compact paper view constructed from the matched arXiv source.
Required fields.

- `global_context.setup`: concise setup needed for the paper's main theorem.
- `global_context.target_theorem`: the main theorem only; return an empty string if uncertain.
- `proof_span.start_snippet` and `proof_span.end_snippet`: verbatim 20–40 word snippets delimiting the proof of the target theorem when recoverable.
- `proof_citations`: at most six citations used as external mathematical tools inside that proof. Each item contains a citation key, locator snippet, and short usage reason.

Rules. Use only citation keys from the bibliography map. Exclude citations used only for background, setup, history, definitions, or non-proof discussion. Prefer empty fields over guessing. Keep the setup and target theorem compact.

Stage-1 fallbacks. Malformed or truncated outputs trigger a compact JSON retry with the C1 keys and at most four citations. Missing proof spans or citations trigger targeted recall from the target-theorem hint, setup, bibliography, and paper view; when the proof span is known, recall is restricted to that proof text. Recall prompts still exclude background, historical, definitional, and non-proof citations. The JSON repair prompt asks for syntax corrections without changing or inventing mathematical content.

C2. Stage-2 citation-local reference extraction

Role. Mathematical Logic Expert.
Input payload. Target theorem, target citation key, pre-citation local proof blocks, citation-focus snippet, short tool-usage reason, and bibliography entry.
Task. Recover only the grounded external tool for the given citation occurrence. Focus on the proof step immediately around the citation. Return the minimal theorem-like statement, estimate, implication, or criterion needed at this step.
Rules.

- Do not rewrite the local context blocks.
- Do not output a theorem title alone.
- Prefer an explicit implication, estimate, criterion, or theorem-like statement over vague prose.
- Reject discourse wrappers such as "Recall that," "In addition," case-analysis narration, future-work text, examples, and remarks.
- Set `restated_in_citing_paper=true` only when the citing text explicitly states the mathematical implication or estimate itself.
- The extracted reference statement must not include theorem labels or source-reporting wrappers such as "Proposition 1.5 states that."

Output. JSON with `reference_tool_latex`, `reference_tool_type`, and `restated_in_citing_paper`.

Stage-2 fallbacks. Empty, generic, or narrative outputs trigger focused extraction of the cited result. If only the consequence used in the proof is recoverable, the retry asks for that consequence as a faithful theorem-like paraphrase. An implication pass rewrites citation-local reasoning into an explicit "If ... then ..." statement; an empty output is allowed when no theorem-like content is recoverable. Optional batch extraction applies the same fields to multiple citation candidates, retaining each citation key and restatement flag.

A.5 Curation, Schema, Release, and Split Policy

After mining, we remove duplicate rows and low-confidence alignments, repair broken core fields, and check that all required task fields are present. The release fields x_i, $\mathsf{y}_i = K_i^\star$, and $\mathsf{z}_i = d_i^{\mathrm{cite}}$ store the input, reference statement, and citation metadata, respectively.

The release stores at most five pre-citation proof blocks. Window diagnostics take the last $m \in \{1, 3, 5\}$ blocks of this list, following the shorter-prefix and empty-context rules in Appendix A.2.

Table 3: **Release and leaderboard contract.** This table lists the artifacts needed to compare results from the same release.

Artifact	Frozen contents	Comparison rule
Dataset release	Instance set, schema, split, domain labels, construction manifest	Results are tied to a named release.
Retrieval artifact	Retrieval provider and externally controlled settings, evaluated queries, Top-20 candidate metadata, timestamps, aggregation rules	New query logs require a new frozen artifact or re-evaluation.
Source-access log	Resolved source identifiers, full-text access status, parsing status	The source-access rate records retrieved and parsed cited documents, not model accuracy.
Scoring artifact	Judge version, rubric prompts, source matcher, aggregation script	Scores should be recomputed if the judge or matcher changes.

Eval-200 is the held-out split for the main experiments. The remaining instances are unused for training, prompt tuning, or model selection. We provide no training split and do not train or tune on Eval-200. Future training and development partitions must exclude its citing papers, since one paper can contribute multiple related gaps.

Eval-200 selection. The held-out evaluation split is quality-filtered rather than randomly sampled. The selection script requires a journal-paper cited source, nonempty locator, a nonempty hint passing the automated selection filters, nonempty complete reference statement, and nonempty local context. It then ranks candidates by a quality score, enforces a per-paper cap, and fills a balanced number of instances in each benchmark domain.

Leaderboard and release policy. Comparisons must identify the evaluation date and the frozen artifacts in Table 3. A new model's queries may require a new query-result artifact under the same provider and externally controlled settings. Results from different releases or retrieval artifacts are comparable only after re-evaluating older systems under the same artifact.

Anonymous release access. The anonymous release link is `https://anonymous.4open.science/r/Re-2Math-4763/`. The release contains the dataset and splits, schema and protocol documentation, core construction and evaluation code, aggregate results, and a source-access summary. It excludes local query caches, raw search downloads, and per-output judgments. The repository will be de-anonymized by the camera-ready deadline if accepted.

A.6 Construction Quality Audit

A single annotator audited a fixed subset of 120 retained instances. The audit checks the citation's proof role, whether the stored context precedes it, the reference statement's sufficiency under that context, and the hint and metadata. Table 4 summarizes the results.

Ten rows fail the citation-role check, 22 fail hint leakage control, and four fail metadata correctness. A reference statement judged sufficient can accompany a citation that does not serve the intended proof role. Passing reference-sufficiency and external-source checks therefore does not resolve the other errors.

Context placement checks where the visible proof stops. Whether the visible context identifies the intended step, and whether the reference justifies that step, require examining the original continuation. Reference sufficiency is an annotator judgment against the supplied context. Section 5.4 examines cases where checking the original continuation changes the interpretation of the reference.

The Raw end-to-end condition never exposes the hint. Hint leakage can nevertheless affect Assisted results and planning agreement against the stored hint. Construction corrections that change evaluated rows require a new release and recomputation of the affected results.

Table 4: **Single-annotator construction audit on 120 retained instances.** These checks describe the audited instances. They do not measure how many eligible citations the pipeline misses or establish quality across the full dataset.

Check	Pass condition	Positive / scored	Rate
Citation role	Citation supplies a result for the intended proof step	110/120	91.7%
Context placement	Stored context precedes the target citation	120/120	100.0%
Reference sufficiency	Reference statement is judged sufficient under the context	120/120	100.0%
Hint leakage control	Hint omits identifying source information	98/120	81.7%
Citation metadata	Metadata matches the source cited in the proof	116/120	96.7%
External-source dependence	Step is judged to require an external result	120/120	100.0%

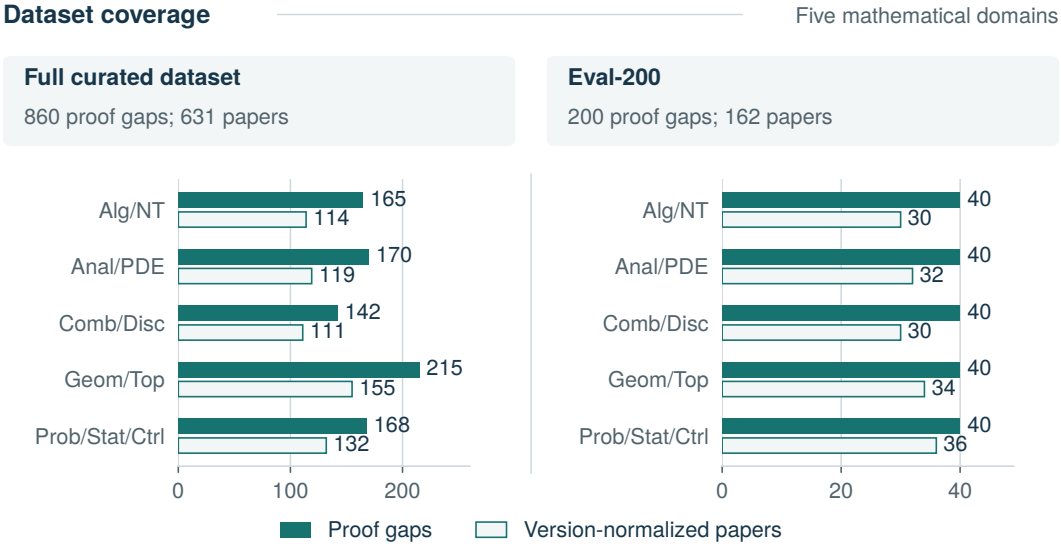


Figure 2: **Dataset statistics.** Overview of the curated benchmark and the Eval-200 subset. The full curated dataset contains 860 proof-gap instances from 631 version-normalized citing papers. Eval-200 contains 200 held-out proof-gap instances from 167 raw citing-paper identifiers (162 after arXiv-version normalization), balanced across five domains.

A.7 Snapshot Statistics

Figure 2 summarizes instance and version-normalized paper counts by domain for the full release and Eval-200.

B Shared Evaluation Protocol

The primary condition disables model-native web-search, browser, file-search, and deep-research tools. A common backend executes model-authored queries and retrieves and parses source text. This follows shared-test-collection practice (Sanderson, 2010): models use the same instances, access budget, and judging policy, although candidate lists depend on their queries. The judges use the reference-based evaluation rule in Section 4.2; they do not directly test mathematical task success.

Scope of the comparison. Integrated deep-research systems search iteratively and synthesize reports (OpenAI, 2025; Google AI for Developers, 2026; FutureSearch et al., 2025; Li et al., 2025). Their results reflect both the model and its search and document-reading backend. Our common

interface controls these operations; it does not measure the full capabilities of deployed browsing agents.

B.1 Stage Inputs and Diagnostic Metrics

Stage inputs. The Raw input is $\mathcal{C}_i$; the separate Assisted query diagnostic adds the stored hint a_i. The planning description $\hat{a}_i$ is predicted from $\mathcal{C}_i$ in Raw and set to a_i in Assisted. Query generation uses $(\mathcal{C}_i, \hat{a}_i)$. The reported Assisted evaluation stops here. In the full Raw run, the model is then asked to select a source from the ranked candidates and extract a statement from its text. The proof context remains available throughout. The full Raw output is the theorem–source pair $(\hat{d}_i, \hat{K}_i)$. Oracle-source receives the same proof context and the retrieved cited document, skipping query generation and candidate selection.

Derivability and its scope. In Section 2.2, $\Gamma \vdash \varphi$ means that a finite derivation of φ exists in the fixed background proof system, with all undischarged assumptions in Γ. The system fixes the language, background axioms, and inference rules. Intermediate deductions and subproofs must respect variable scope and assumption discharge. Adding K makes that theorem available as a premise; access to $\mathcal{D}_i$ does not add every source statement as a premise. The context Γ_i is interpreted from the visible $\mathcal{C}_i$, without importing omitted proof blocks or the hidden continuation.

If $\Gamma_i \vdash \Delta_i$, any admissible, source-supported K meets the derivability condition, even if the derivation does not use it. The released inputs are proof text, not formal contexts; the model returns a theorem–source pair, not a derivation. The judges in Section 4.2 use a reference-based criterion rather than implementing the derivability test.

Planning and retrieval diagnostics. Let $\mathcal{T}$ denote the evaluation set. For the Raw planning diagnostic, write $\hat{a}_i^{\mathrm{raw}}(m)$ for the description generated from the global setup and the last m available local proof blocks. With $\mathcal{T}_a = \{i \in \mathcal{T} : a_i \neq \bot\}$,

$$\mathrm{AnchorAcc}_m = \frac{1}{|\mathcal{T}_a|} \sum_{i \in \mathcal{T}_a} \mathbf{1}[\mathsf{AnchorMatch}_i(\hat{a}_i^{\mathrm{raw}}(m), a_i) = 1].$$

The main setting uses $m = 5$; the window ablation uses $m \in \{1, 3, 5\}$. Shorter or empty stored prefixes are used as available (Appendix A.5). Appendix C.3.1 specifies the matching rubric.

These are separate Raw planning runs. Assisted QueryAcc is the fraction of evaluable queries judged to target the reference statement or essentially the same result. The query judge receives the query, reference statement, and cited-source metadata; the model receives the proof context and stored hint. No search is run. The local-only planning and Assisted query diagnostics use the released evaluable-case denominators in Table 10; unavailable diagnostic judgments are excluded.

GroundRate and CiteRecall@N both use $|\mathcal{T}|$ as the denominator. GroundRate counts positive grounding judgments. CiteRecall@N counts instances with at least one candidate matching the cited source among the first N ranked records; Appendix B.4 specifies source matching. Neither diagnostic requires positive labels from both judges.

B.2 Model Configuration and Score Provenance

The public evaluation configuration specifies `openai/gpt-5.4` as the fixed judge, with reasoning effort set to `medium`. Table 5 lists the seven solver identifiers. These settings were checked against code revision `fe6fe43`; code and artifacts are provided in the anonymized release.

Release scores and auxiliary artifacts. The main results reproduce the public aggregated tables. Older local prediction-level outputs used for auxiliary checks record GPT-5.2 as judge for six runs and Claude Opus 4.5 for the Claude run. The public package provides neither per-output judgments nor records linking the older outputs to GPT-5.4 rescoring. These older labels and the human-confirmed audit cannot be attributed to GPT-5.4 and do not independently validate it. The main ToolAcc/GroundRate ratios use only the public aggregate counts.

The labels in the release are named `Ground` and `Suff`; they correspond to J_i^G and J_i^S in Section 4.2. The source-access rate is documented in Appendix C.2.

Table 5: **Model identifiers in the public evaluation configuration.** GPT-5.4 is the configured judge; the seven rows are the evaluated solvers.

Paper name	API model identifier
GPT-5.2	`openai/gpt-5.2`
Gemini 3.1 Pro	`google/gemini-3.1-pro-preview`
Claude Opus 4.5	`anthropic/claude-opus-4.5`
DeepSeek V3.2	`deepseek/deepseek-v3.2`
Qwen3-235B Thinking	`qwen/qwen3-235b-a22b-thinking-2507`
Kimi K2 Thinking	`moonshotai/kimi-k2-thinking`
Grok 4	`x-ai/grok-4`

Some archived outputs pass grounding but have an uncertain Suff label (Gemini: 2; Claude, DeepSeek, Kimi: 1 each).

B.3 Input Rendering and Retrieval Artifact

Input rendering. The evaluator builds inputs from the dataset rows. The reported end-to-end runs use Raw inputs with up to five local proof blocks.

Choice of Google Scholar. Google Scholar covers journal and proceedings articles, preprints, books, surveys, and bibliographic records linking to sources. This breadth matters because a needed theorem may lie outside arXiv or any one publisher's index. Its coverage and ranking are not controlled by us, so we freeze observed query results. Candidate type alone does not establish source admissibility or mathematical validity.

Retrieval execution and temporal controls. Each model-authored Google Scholar query is executed once, and its Top-20 results are cached under these settings:

Setting	Value
interface/result language	English
region or domain	not additionally controlled
ranking mode	default relevance ranking
date-range policy	no lower-bound year; upper year = citing-paper publication year
patent results	excluded

The cache records the query string, execution timestamp, returned rank order, candidate metadata, the externally controlled Google Scholar settings, and source-access outcomes when source access is attempted. This frozen cache fixes the retrieval evidence for the reported query-based runs; scoring also uses the corresponding prediction and judgment records.

Temporal interpretation. The publication-year cutoff restricts the dates of candidate records; freezing the artifact makes the observed results replayable. Neither reconstructs the exact historical search state available to the original authors. CiteRecall@20 measures citation recovery under this publication-year filter using the cached search results.

Ranking. Candidates retain Google Scholar's cached rank order. We neither promote the cited source nor rerank with the reference statement.

Source access and text extraction. After inspecting Top-20 metadata, the model may select at most one candidate to open. If the source can be retrieved and parsed, the model receives its text and is asked to extract or faithfully restate a theorem-like statement. All models use the same source-access and text-extraction procedure. The source-access log records resolution, retrieval, and parsing outcomes for selected and cited sources.

B.4 Source Identity Matching

We match each candidate's metadata to the stored citation metadata using $\mathsf{Match}(d, \mathsf{z}_i)$. The matcher prioritizes DOI and arXiv identifiers when available; otherwise it applies normalized title–author matching, with ambiguous journal/preprint variants handled by conservative or manual rules. The matcher supports $\mathrm{CiteRecall@}N$ and oracle-source diagnostics. A source-identity match does not establish membership in $\mathcal{V}_i$, which also requires source admissibility, statement support, and derivability of the intended proof step.

Admissibility of the citing paper. The citing paper and its variants are admissible only when the selected text independently states or proves the invoked result outside the held-out argument. Source support for the local proof sentence alone is insufficient.

B.5 Raw and Judging Prompt Templates

The following summaries describe the prompt roles, inputs, constraints, and outputs. E5–E6 report the released judging behavior; its differences from the intended task criterion are stated below.

E1. Planning prompt

Role. Mathematical Research Assistant.
Input. Global setup, target theorem, and pre-citation local proof prefix.
Task. Produce a planning description before retrieval: describe the result needed to justify the intended proof step.
Constraint. Return a planning description without bibliographic identifiers. Do not mention theorem titles, author names, citation keys, source titles, DOI strings, arXiv identifiers, or bibliographic metadata.
Output. JSON with field `planning_anchor`.

E2. Query-generation prompt

Role. Mathematical literature search assistant.
Input. Global setup, target theorem, local proof prefix, and the planning description (predicted in Raw; the stored hint in Assisted).
Task. Write a concise scholarly search query aimed at finding a theorem, estimate, lemma, criterion, or standard result that can justify the next proof step. Prefer mathematical objects, canonical terminology, named properties, theorem families, and strength indicators over generic application words.
Rules. Avoid copying unique prose from the citing paper. Do not use author names, citation keys, DOI strings, arXiv identifiers, or bibliographic metadata from the withheld reference records. The archived prompt instructs the model to use 3–5 keywords with a five-word maximum. This is a prompt instruction, not a verified hard limit on every emitted query. It appears in 1,398 of the 1,400 archived end-to-end prompt records; the two remaining records do not establish compliance.
Output. JSON with field `search_query` in the archived prompt.

E3. Metadata shortlisting prompt

Role. Careful mathematical retrieval shortlister.
Input. Proof context, planning description, search query, and Top-k candidate metadata.
Task. Select the single candidate source most worth opening in full text. Decide from title, authors, venue, snippets, and metadata whether the candidate is likely to contain a theorem-like tool for the proof gap.
Rules. Use the candidate metadata for selection without solving the problem. Prefer papers likely to contain the needed theorem or technique, and keep the shortlist small. The shortlisting prompt in the inspected release does not include an explicit citing-paper exclusion instruction.
Output. JSON with selected candidate indices and a brief reason; the reported condition opens at most one source.

E4. Full-text solver prompt

Role. Mathematical assistant for the gap-filling task.
Input. Proof context and the full text of one selected candidate paper, truncated to a fixed prompt budget.
Task. Decide whether the selected paper contains a theorem-like statement that bridges the proof gap. If so, extract or faithfully restate the minimal relevant theorem, lemma, proposition, corollary, criterion, estimate, or standard result.
Rules. Judge whether the candidate paper contains the needed theorem or lemma, assign a relevance score from 0 to 10, and extract the result as faithfully as possible when the score exceeds 5. The prompt in the

E4. Full-text solver prompt (continued)

inspected release does not explicitly prohibit extracting the local citing proof sentence; source admissibility is a separate task requirement.
Output. JSON with `relevance_score`, short reason, and `extracted_theorem`.

E5. Grounding-judge prompt

Input. Selected source identifier, returned statement, and selected source excerpts. The implementation first takes windows around the first 36 occurrences of theorem-like keywords, including *remark* and *definition*. It ranks these windows by overlap and text similarity to the prediction, selects up to six, and passes at most 12,000 characters to the judge. A relevant result later in the document can fall outside this candidate pool.
Task. Return `yes`, `no`, or `uncertain` depending on whether the returned statement is verbatim or a mathematically faithful restatement of a theorem-like statement supported by the selected source.
Scope. The grounding judge checks whether the source excerpts support the statement, not whether the theorem is mathematically correct. A statement can be grounded even if it is not sufficient for the proof gap.
Output. Label, short reason, confidence, and evidence span.

E6. Sufficiency-judge prompt

Input. Rendered proof context, reference statement $K^\star$, and returned statement $\hat{K}$.
Task. Return `yes` when $\hat{K}$ is mathematically equivalent to the reference statement up to benign notation changes, or is strictly stronger with assumptions satisfied by the proof context. Return `no` for a weaker or mismatched statement, or one requiring unsupported assumptions; use `uncertain` when the context is insufficient to decide.
Comparison target. The released prompt uses $K^\star$; it has no separate field for the evaluator-only intended proof step Δ_i.
Output. Label, short reason, and confidence.

Scope of LLM-based evaluation. E5 checks whether source excerpts support the returned statement; these excerpts may omit supporting passages. E6 compares the statement with the reference and may reject a weaker result that suffices for the local step. Their joint label assesses source support and reference-based sufficiency. Establishing membership in the set of suitable theorem–source pairs defined in Section 2.2 also requires checking source admissibility and whether the theorem justifies the intended step. These checks include independence from the held-out argument and the conditions needed to apply the theorem.

C Supplementary Experimental Results

C.1 Raw End-to-End Results

The following tables expand the Raw results in Table 1, reporting uncertainty, domain variation, and outputs from unmatched sources that pass both checks.

C.1.1 Statistical Uncertainty

Because Eval-200 contains multiple gaps from some citing papers, we report both ordinary binomial intervals and paper-cluster bootstrap intervals. Table 6 gives uncertainty estimates for ToolAcc. These intervals describe sampling or resampling variation conditional on the recorded labels, not systematic error from target misalignment, source admissibility, or judge decisions. Full metric-level intervals are included in the public aggregation artifact. The bootstrap resamples groups defined by the stored paper identifiers, with defaults of 1,000 draws and seed 0; the intervals below reproduce the released values rather than the auxiliary reanalysis of older local outputs.

C.1.2 Domain Stratification

Eval-200 gives equal weight to five broad areas, with 40 instances per domain. Table 7 separates their ToolAcc scores to show variation hidden by the aggregate. With only 40 instances per cell, these are diagnostics rather than stable domain rankings.

Table 6: **Uncertainty for** ToolAcc**.** Wilson intervals use instance-level binomial uncertainty. Paper-cluster intervals reproduce the public aggregation artifact, whose grouping uses 167 stored paper identifiers rather than the 162 version-normalized papers in the dataset summary.

Model	ToolAcc		Wilson 95% CI	Paper-cluster 95% CI
Claude Opus 4.5	7.0%	(14/200)	[4.2%, 11.4%]	[3.6%, 10.7%]
Grok 4	3.5%	(7/200)	[1.7%, 7.0%]	[1.0%, 6.6%]
Kimi K2 Thinking	3.5%	(7/200)	[1.7%, 7.0%]	[1.0%, 6.4%]
GPT-5.2	3.0%	(6/200)	[1.4%, 6.4%]	[1.0%, 5.5%]
DeepSeek V3.2	2.5%	(5/200)	[1.1%, 5.7%]	[0.5%, 5.0%]
Gemini 3.1 Pro	2.0%	(4/200)	[0.8%, 5.0%]	[0.5%, 4.0%]
Qwen3-235B Thinking	1.0%	(2/200)	[0.3%, 3.6%]	[0.0%, 2.6%]

Table 7: **Domain-level** ToolAcc **on Eval-200.** Each domain has 40 instances.

Model	Alg./NT	Anal./PDE	Comb.	Geom./Top.	Prob./Stat. /Ctrl.
GPT-5.2	0.0%	5.0%	5.0%	2.5%	2.5%
Gemini 3.1 Pro	0.0%	5.0%	0.0%	2.5%	2.5%
Claude Opus 4.5	5.0%	2.5%	10.0%	7.5%	10.0%
DeepSeek V3.2	0.0%	0.0%	5.0%	0.0%	7.5%
Qwen3-235B Thinking	0.0%	0.0%	0.0%	2.5%	2.5%
Kimi K2 Thinking	0.0%	0.0%	7.5%	2.5%	7.5%
Grok 4	0.0%	5.0%	5.0%	2.5%	5.0%

C.1.3 Source Matches among Outputs Passing Both Checks

Table 8: **Outputs from unmatched sources.** The first score repeats end-to-end ToolAcc for context. The other columns summarize outputs that pass both checks but whose selected source was not matched to the cited source; a mismatch does not establish source admissibility.

Model	ToolAcc		Unmatched source ToolAcc		Unmatched share among passing outputs	
Claude Opus 4.5	7.0%	(14/200)	6.5%	(13/200)	92.9%	(13/14)
Grok 4	3.5%	(7/200)	3.5%	(7/200)	100.0%	(7/7)
Kimi K2 Thinking	3.5%	(7/200)	3.5%	(7/200)	100.0%	(7/7)
GPT-5.2	3.0%	(6/200)	2.5%	(5/200)	83.3%	(5/6)
DeepSeek V3.2	2.5%	(5/200)	2.5%	(5/200)	100.0%	(5/5)
Gemini 3.1 Pro	2.0%	(4/200)	1.5%	(3/200)	75.0%	(3/4)
Qwen3-235B Thinking	1.0%	(2/200)	1.0%	(2/200)	100.0%	(2/2)

Remark. Main values reproduce the released aggregate tables. Appendix B.2 describes the older output records used for auxiliary checks. A single-annotator audit identifies three false positives among outputs from unmatched sources that passed both checks (Appendix D.8). Applying those audit labels would lower the counts in the older records; the public aggregate scores remain unchanged.

C.2 Oracle-Source Evaluation

Table 2 reports the source-access rate and Oracle ToolAcc. The Oracle-source condition, called source-provided in prompt E7, includes only instances whose cited source was resolved, retrieved, and parsed. Table 9 reports the 74 instances accessible in at least one run, those accessible in every run, and run-specific differences. Raw and Oracle use different evaluation sets and prompts, so their score differences do not isolate the effect of bypassing search.

Table 9: **Source-access summary.** Counts describe the release-protocol source-access records, not model capability.

Source-access status	Count	Interpretation
Accessible in at least one completed run	74/200	Cited source was resolved, retrieved, and parsed in at least one Oracle-source run.
Retrieved and parsed in every completed run	66/200	Subset shared across all completed Oracle-source runs.
Unavailable in every completed run	126/200	Source could not be retrieved and parsed under the current release protocol.
Run-specific source-access differences	8/200	Cases with source-access/logging differences across completed Oracle-source runs.

E7. Source-provided extraction prompt

Role. Mathematical assistant in the source-provided condition.
Input. Proof context and the retrieved and parsed cited source content.
Task. Given the source identified by the recorded author citation, identify and extract the theorem-like statement that solves the gap. Source retrieval alone does not guarantee that the record is correctly matched or that the required statement is present.
Output. JSON with short reason and `extracted_theorem`.

C.3 Proof Context and Query Formulation

Planning diagnostics compare agreement with the stored hint across proof-context windows; the local-only condition omits the global setup. Assisted query diagnostics assess query formulation given the hint.

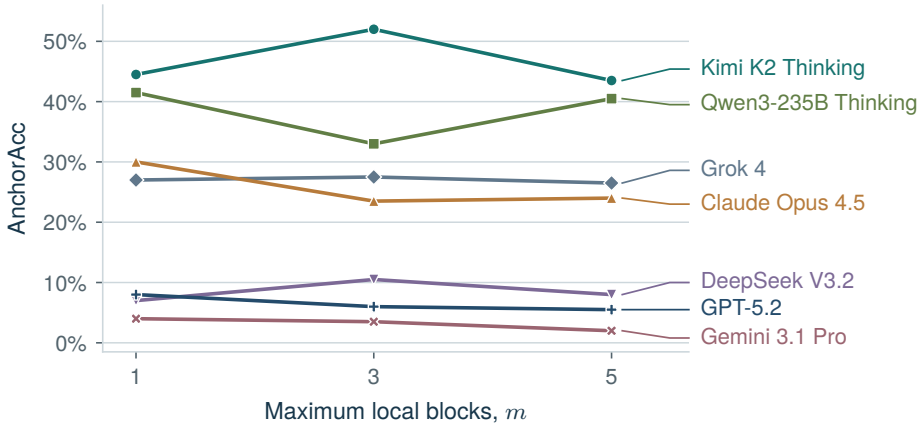


Figure 3: **Planning with different local windows.** Raw inputs contain the global setup and up to m preceding proof blocks. Each point uses 200 instances and measures agreement with the stored hint. Full counts and other input conditions appear in Table 10.

C.3.1 Planning Agreement: AnchorMatch Rubric

AnchorMatch compares the Raw planning description with the stored hint, not with a source or a complete theorem statement. Appendix A.6 reports hints that still reveal source information.

Rubric. A planning description matches the stored hint when it identifies the same needed result: which objects it concerns, how it will be used, and what it must establish for the proof to continue. Exact wording is not required. Paraphrases, equivalent terminology, or a different theorem family can match if they describe a result that serves the same purpose in the proof. Broad agreement on the

topic is insufficient. A description that only says "use compactness," "apply a comparison theorem," or "find a result about positive definite functions" does not match unless it specifies the objects and the proof obligation tightly enough to guide retrieval. Bibliographic shortcuts are also insufficient: mentioning an author, theorem name, source title, citation key, DOI, or arXiv identifier does not by itself establish a hint match.

Examples. A description of a compactness theorem that turns uniform bounds and time-translation control into the strong convergence needed for a nonlinear limit passage can match a hint with that proof role. Merely saying "use compactness to get a subsequence" is uncertain or negative if it omits the required objects, topology, or conclusion. Likewise, "use harmonic analysis on spheres" is too broad to match a hint asking for a characterization of positive definite functions via expansion coefficients. Only clear positive matches contribute to AnchorAcc; agreement neither establishes successful retrieval nor rules out a different theorem that justifies the step.

C.3.2 Context Windows and Assisted Queries

Scope of comparison. Section 5.3 discusses the global-plus-local results. Local-only inputs remove the global setup; a change in score alone does not show whether that setup helps or distracts the model. Assisted QueryAcc and Raw ToolAcc come from different runs with different inputs and judging criteria. Locating an error after query generation would require linking query judgments to final retrieval outcomes within the same run. Assisted $\mathrm{QueryAcc}$ uses imperfect hints (Appendix A.6) and is not a leaderboard metric. Raw end-to-end $\mathrm{ToolAcc}$ never exposes hint text.

Table 10: **Planning-window and assisted-query diagnostics.** Values are percentages with counts in parentheses. Separate Raw planning runs vary the visible context. Assisted queries are generated with the stored hint and judged against the reference statement and cited-source metadata.

	Raw planning: AnchorAcc				Assisted query
Model	**Local only** $m = 5$	**Global+local** $m = 1$	**Global+local** $m = 3$	**Global+local** $m = 5$	**QueryAcc** $m = 5$
GPT-5.2	10.5% (21/200)	8.0% (16/200)	6.0% (12/200)	5.5% (11/200)	18.5% (37/200)
Gemini 3.1 Pro	6.5% (13/200)	4.0% (8/200)	3.5% (7/200)	2.0% (4/200)	11.5% (23/200)
Claude Opus 4.5	23.0% (46/200)	30.0% (60/200)	23.5% (47/200)	24.0% (48/200)	41.0% (82/200)
DeepSeek V3.2	11.5% (23/200)	7.0% (14/200)	10.5% (21/200)	8.0% (16/200)	14.0% (28/200)
Qwen3-235B Thinking	41.5% (83/200)	41.5% (83/200)	33.0% (66/200)	40.5% (81/200)	30.0% (60/200)
Kimi K2 Thinking	44.5% (89/200)	44.5% (89/200)	**52.0%** (104/200)	43.5% (87/200)	48.0% (96/200)
Grok 4	24.4% (48/197)	27.0% (54/200)	27.5% (55/200)	26.5% (53/200)	**55.6%** (110/198)

Note. Denominators differ for two Grok diagnostic cells because a small number of diagnostic judgments were unavailable.

D Case Studies and Evaluation Audit

These cases use archived outputs from seven models under earlier judge configurations (Appendix B.2). They examine retrieval, oracle-source extraction, and theorem application, followed by an audit of the recorded judgments. These selected records do not measure failure rates or rank models.

Legend. Y/N indicate positive/negative labels, and $\perp$ indicates a missing or non-evaluable recorded judgment. **Tool** indicates positive labels for both Ground and Suff; **Cite@20** records whether the Top-20 candidates contain the original cited source. **Search query** is the query from the Raw end-to-end run. **Plan** is the separate five-block Raw planning diagnostic; **Query** is the Assisted query

diagnostic. These diagnostic labels are not successive gates in the displayed retrieval run. Titles are shortened for readability.

D.1 Tulenbaev Lifting: Queries without a Selected Source

Instance. `2110.11087v2_gap_1`. **Paper.** *On the $\mathbb{A}^1$-invariance of K_2 modeled on linear and even orthogonal groups*. **Domain.** Algebra / number theory.
Proof situation. The proof verifies axioms CFC–HIF for $K_2(\Phi, -)$. CFC and DP have been handled; the remaining step uses a Tulenbaev lifting/gluing property for Steinberg groups.
Reference statement. For a regular ring R containing a field k and $\Phi = A_\ell$, $\ell \geq 4$, the Steinberg group $\mathrm{St}(\Phi, R[t])$ satisfies the polynomial gluing property.
Construction citation. Tulenbaev, *The Steinberg group of a polynomial ring*, 1983.

Model behavior. All seven runs miss the cited source in Top-20 and select no source. Several queries explicitly name Tulenbaev and Steinberg groups; for example, Claude queries "Steinberg group Tulenbaev lifting property." Ground, Suff, and Tool are therefore missing or non-evaluable in all seven records.

The records do not isolate why the cited source was absent from Top-20 or why no candidate was selected; the proof prefix may still point to other useful results.

Contrast with Muskat compactness. For `1905.05370_gap_0`, from *Weak Solutions to the Muskat Problem with Surface Tension Via Optimal Transport*, the proof needs strong L^1 compactness of approximate densities. All seven planning labels are positive; six outputs pass grounding, but none passes sufficiency. Most queries use broad compactness terminology, such as Claude's "Aubin-Lions compactness Wasserstein gradient flow." Neither case isolates query wording as the cause of failure.

D.2 Extracting the Matroidal-Locus Identity from a Supplied Source

Instance. `1510.08891_gap_1`. **Paper.** *Complete moduli of cubic threefolds and their intermediate Jacobians*. **Domain.** Algebra / number theory.
Proof situation. The proof needs a toroidal-compactification fact to show that the extended intermediate-Jacobian map lands in the correct compactification locus.
Reference statement. The matroidal locus $\mathrm{Matr}[5]$ is the maximal partial compactification contained in both the second Voronoi and perfect-cone compactifications; equivalently, $\Sigma^V \cap \Sigma^P = \Sigma^{\mathrm{mat}}$.
Construction citation. Melo–Viviani, *Comparing perfect and 2nd Voronoi decompositions: the matroidal locus*.

Model behavior: Matroidal-locus instance.

Model	Search query	Plan	Query	Cite@20	Ground	Suff	Tool	Source
GPT-5.2	cubic threefold monodromy cones	N	N	N	N	N	N	S1
Gemini 3.1 Pro	local monodromy boundary divisors	N	N	N	N	N	N	S2
Claude Opus 4.5	monodromy intermediate Jacobian cubic threefold	N	N	N	⊥	⊥	⊥	–
DeepSeek V3.2	monodromy intermediate Jacobian cubic threefolds	N	N	N	N	N	N	S1
Qwen3-235B Thinking	monodromy GIT boundary Voronoi compatibility	N	N	N	⊥	⊥	⊥	–
Kimi K2 Thinking	monodromy intermediate Jacobian GIT boundary	N	N	N	N	N	N	S1
Grok 4	monodromy intermediate Jacobian cubic threefolds	N	N	N	Y	N	N	S1

Sources. **S1**: Moduli space via degenerations. **S2**: A p-adic local monodromy theorem. A dash denotes no selected source.

A related but insufficient statement. Grok receives a positive Ground label for a statement about the boundary geometry of intermediate Jacobians. The statement is related to the citing paper, but it is not the fan-theoretic equality $\Sigma^V \cap \Sigma^P = \Sigma^{\mathrm{mat}}$ and does not imply the maximality of the matroidal partial compactification.

Oracle-source extraction. Given the cited source by Melo–Viviani, GPT and Claude recover the reference statement. Claude extracts the fan identity: a cone belongs to both the second Voronoi and perfect-cone decompositions iff it belongs to the matroidal decomposition.

Visible context and intended step. The global setup describes cubic threefolds and the extended intermediate-Jacobian map, without specifying the perfect-cone or matroidal target. The withheld continuation introduces rank-one quadrics and the matroidal locus before invoking the cited result. All seven queries in the table concern monodromy or boundary geometry rather than the fan identity. As discussed in Section 5.3, this mismatch alone does not establish poor query formulation: the Raw input may not identify the intended proof step.

D.3 Positive Definite Functions on Spheres

Instance. `1901.06096v3_gap_1`. **Paper.** *Repeated minimizers of p-frame energies*. **Domain.** Combinatorics / discrete mathematics.
Proof situation. The proof reduces a p-frame energy bound to the case $p = 2$, where it invokes positive definite functions on the unit sphere.
Reference statement, with conventions explicit. For $d \geq 2$ and continuous $f : [-1, 1] \to \mathbb{R}$, the kernel $K(x, y) = f(\langle x, y\rangle)$ on $\mathbb{S}^{d-1}$ is positive semidefinite iff $f(t) = \sum_{k\geq 0} a_k C_k^{(d-2)/2}(t)$, with $a_k \geq 0$ and $\sum_k a_k C_k^{(d-2)/2}(1) < \infty$. For $d = 2$, use $C_k^0(\cos\theta) = \cos(k\theta)$ (Gneiting, 2013).
Construction citation. Schoenberg, *Positive definite functions on spheres*, 1942 (the stored citation gives 1941).

Model behavior: Positive-definite-function instance.

Model	Search query	Plan	Query	Cite@20	Ground	Suff	Tool	Source
GPT-5.2	positive definite polynomial sphere	N	Y	Y	⊥	⊥	⊥	S1
Gemini 3.1 Pro	positive definite function coefficients one	N	Y	Y	N	N	N	S2
Claude Opus 4.5	positive definite functions spheres Gegenbauer	Y	Y	Y	Y	Y	Y	S3
DeepSeek V3.2	positive definite functions unit sphere	Y	Y	Y	⊥	⊥	⊥	–
Qwen3-235B Thinking	spherical positive definite sum squares	Y	N	Y	Y	N	N	S4
Kimi K2 Thinking	positive definite sphere quadratic forms	N	Y	Y	N	N	N	S5
Grok 4	positive definite functions Gegenbauer expansion	Y	Y	Y	N	N	N	S5

Sources. **S1**: Strictly positive definite functions on spheres. **S2**: Metric spaces and positive definite functions. **S3**: Positive definite functions in distance geometry. **S4**: Positive definite functions on the unit sphere and integrals of Jacobi polynomials. **S5**: Optimal measures for p-frame energies. A dash denotes no selected source.

Output passing both checks. Claude's recorded output invokes the Schoenberg characterization of the zonal kernel $K(x, y) = f(\langle x, y\rangle)$ through nonnegative Gegenbauer coefficients. This is the theorem family used in the $p = 2$ energy argument; the positive Suff label records the judge's comparison with the reference statement.

Additional hypotheses. Qwen receives a positive Ground label for a sufficient-condition theorem involving smoothness, convexity of a signed derivative, and boundary conditions. The recorded

context does not establish these hypotheses; whether they can be derived from it remains to be checked.

The cited source appears among the Top-20 candidates in all seven runs, but only Claude receives a positive Suff label.

D.4 A Transfer Theorem without the Required Lower Bound

Instance. `2012.12831_gap_0`. **Paper.** *Approximation Limitations of Pure Dynamic Programming*. **Domain.** Probability / statistics / control.

Proof situation. The proof needs a monotone Boolean lower bound for the assignment/permanent predicate before applying a Boolean-to-tropical transfer principle.

Reference statement. The logical permanent requires monotone Boolean circuits of size at least $n^{\Omega(\log n)}$.

Construction citation. Razborov, *Lower bounds on monotone complexity of the logical permanent*, 1985.

Model behavior: Boolean-to-tropical instance.

Model	Search query	Plan	Query	Cite@20	Ground	Suff	Tool	Source
GPT-5.2	tropical circuit boolean reduction	N	N	N	Y	Y	Y	S1
Gemini 3.1 Pro	Jerrum Snir monotone perfect matching	N	N	N	⊥	⊥	⊥	–
Claude Opus 4.5	tropical circuit monotone boolean simulation	Y	Y	N	N	N	N	S2
DeepSeek V3.2	boolean assignment problem lower bound	Y	Y	N	⊥	⊥	⊥	–
Qwen3-235B Thinking	tropical circuit monotone boolean approximation	Y	N	N	Y	N	N	S3
Kimi K2 Thinking	Jerrum Snir permanent monotone circuit	Y	Y	N	⊥	⊥	⊥	–
Grok 4	monotone Boolean circuit lower bound	Y	Y	N	N	N	N	S3

Sources. **S1**: Approximation Limitations of Pure Dynamic Programming. **S2**: Reciprocal inputs in arithmetic and tropical circuits. **S3**: Monotone circuit complexity of Boolean functions. A dash denotes no selected source.

Missing lower-bound premise. The recorded judge accepts GPT-5.2's output, but the single-annotator audit flags this as a false positive among outputs from unmatched sources: GPT recovers the Boolean-to-tropical transfer theorem from the citing paper, not the monotone lower bound for the logical permanent needed to instantiate the transfer. This is one of the sensitivity cases reported in Appendix D.8.

Mentioning transfer principles in a query does not show that the model has identified the missing Boolean bound.

D.5 A Weaker Conclusion Suffices for a Fixed Graph Embedding

In instance `2507.02263v2_gap_1`, the reference requires $\Omega_{k,\varepsilon}(n^{k+1})$ copies of K_{k+1} under a signless-Laplacian spectral threshold. The local proof instead embeds a fixed graph F, with $\chi(F) = k + 1$, into a complete multipartite blow-up; the continuation concludes that G contains F.

The stored global target also asks for the clique-counting conclusion. The visible prefix ends with "then for sufficiently large n, we can", and the withheld locator begins "find a copy of F in G". The stored citation occurs later in that locator, in a comparison of proof techniques, after the containment statement. Its position therefore does not by itself establish an external justification missing at the cutoff. Section 5.4 uses this record to illustrate the mismatch between the intended step and the reference statement, not as a verified successful retrieval instance.

GPT extracts the logarithmic-part-size blow-up theorem; Claude and Kimi extract the clique-counting theorem (Theorems 2.4 and 2.2 in the selected PDF). Section 5.4 gives the containment argument and an example separating it from the counting guarantee. GPT's negative sufficiency label therefore does not establish that its conclusion is too weak for the intended step. The theorem's applicability remains a separate check.

All three selected PDFs resolve to the citing paper, *Some Turán-type results for the signless Laplacian spectral radius*, arXiv:2507.02263v2. Their source admissibility remains unresolved, and the aggregate labels are unchanged. The case establishes neither model failure nor an admissible external-source success.

D.6 Fractional Heat and Kolmogorov Equations

Instance `1804.10581v2_gap_12` has a target theorem about the fractional heat equation, but its local context, locator, and reference statement concern small-time non-controllability of a Kolmogorov equation. The source paper, *Lack of null-controllability for the fractional heat equation and related equations*, states these as separate results. Theorem 1.3 addresses fractional heat; Theorem 1.6 addresses Kolmogorov dynamics and implies the local special case $\omega = \mathbb{T} \times (a, b)$, $0 < a < b$, with non-controllability for $T < a^2/2$.

The archived GPT, Gemini, and Grok predictions extract this Kolmogorov theorem. DeepSeek extracts the fractional-heat theorem and receives a positive sufficiency judgment that describes it as stronger. That inference is invalid: a broader time range for a different equation does not establish the required Kolmogorov conclusion.

The four cached selected PDFs are identical and resolve to the citing paper, arXiv:1804.10581v2; a SIAM landing URL does not make one a different external source. This case combines a mismatch between global setup and intended step with unresolved source admissibility. The archived labels remain unchanged.

D.7 Source Admissibility: Brownian Interlacements

In the Brownian-interlacement instance `1804.05967v2_gap_0`, Claude, DeepSeek, Qwen, Kimi, and Grok receive positive historical Ground, Suff, and Tool labels despite zero citation recall. Their returned statements identify conditioned Brownian motion with a Doob h-transform, matching the reference statement. However, their selected source has the same title as the citing paper, *Two-dimensional Brownian random interlacements*. The records do not show that the selected passage is independent of the held-out argument. Unlike the missing premise in the Boolean-to-tropical audit, this is an unresolved admissibility question, not an established mathematical mismatch.

D.8 Human-Confirmed Prediction Audit and Outcome Analysis

A single human annotator confirmed assistant-prepared judgments for 175 outputs on April 26, 2026. The audit covers seven models and multiple predicted outcomes. For 123 outputs, the prepared labels were initialized from existing model judgments; the other 52 record assistant case review. The final human-confirmed fields were populated from these prepared labels. Table 11 reports agreement with the confirmed labels. This assisted procedure audits the recorded outputs; it is not a blind or independent reliability estimate. Ground and Suff each have 156 eligible outputs; Tool, their joint label used in ToolAcc, has 155.

Of 42 outputs that passed both checks and whose sources did not match the stored citation, 39 retain positive audit labels and three receive negative labels. All three outputs from the cited source that passed both checks retain positive audit labels.

Two of the 39 positive audit labels for unmatched-source outputs also carry unresolved citing-paper exclusion flags: `1909.11981v3_gap_3` for Claude and `2109.07887v3_gap_1` for Qwen. Thus 39/42 is a stored-label rate, not an independently established admissibility rate. Table 13 shows the three negative-label cases; resolving those alone does not resolve the two additional flags. These older audit labels have not been used to correct the public aggregate tables.

Table 11: **Agreement of stored judge and confirmed audit labels.** The audit covers 175 sampled outputs stratified by model and predicted outcome. Eligible labels exclude abstentions and cases without an evaluable prediction; here abstentions are recorded as `uncertain` by both the recorded judges and the human-confirmed audit label. κ is judge-vs-human-confirmed kappa, not inter-annotator kappa.

Label	Eligible	Agreement	Precision	Recall	F1	κ
GROUND	156	100.0%	100.0%	100.0%	100.0%	1.000
SUFF	156	98.1%	93.3%	100.0%	96.6%	0.952
TOOL	155	98.1%	93.3%	100.0%	96.6%	0.952

Table 12: **Judge-audit sample composition.** The 175 audited outputs are stratified across the stored success, grounded-but-insufficient, ungrounded, and judge-abstained buckets and the seven models. These are the original sampling labels; some outputs outside the success bucket have unresolved judgments.

Bucket	GPT	Gemini	Claude	DeepSeek	Qwen	Kimi	Grok	Total
Success	6	4	14	5	2	7	7	45
Grounded but insufficient	10	4	8	12	9	9	18	70
Ungrounded	5	7	6	5	8	5	4	40
Judge abstained	3	3	2	3	4	3	2	20
Total	24	18	30	25	23	24	31	175

Subsequent source-level checks of the signless-Laplacian and fractional-heat cases identify additional target-alignment and citing-paper-source problems (Section 5.4). In particular, the positive Claude and Kimi labels for `2507.02263v2_gap_1` and the positive DeepSeek label for `1804.10581v2_gap_12` are not established external-source successes. These findings are separate from the three negative labels in the earlier audit.

Table 13: **Audit rejections among outputs that passed both checks.** All three outputs use unmatched sources. The last column gives changes to the counts passing both checks if these historical audit labels were applied. No correction is applied to the public scores.

Model	Instance	Audit reason	Count change
GPT-5.2	`2012.12831_gap_0`	Recovered the Boolean-to-tropical transfer theorem, but not the monotone lower bound needed to instantiate it.	All sources: -1; unmatched sources: -1
GPT-5.2	`2401.14380v1_gap_0`	Retrieved the paper's own target theorem rather than the external spline formula used for the proof step.	All sources: -1; unmatched sources: -1
Claude Opus 4.5	`2010.01779v3_gap_0`	Retrieved only the heavy-tail trap assumption; the audited reference statement is a scaling-limit theorem.	All sources: -1; unmatched sources: -1