Method Bench: Evaluating Path-to-Feasibility Reasoning in the Sciences

Anonymous ACL submission

Abstract

Large language models (LLMs) have shown promise as scientific research tools, but key research capabilities remain largely untested by existing evaluation techniques. Real-world research programs are often goal-driven and seek methodologies that enable a desired research finding. Scientists use this path-to-feasibility reasoning to focus their efforts on explorations likely to produce useful outcomes. We present Method Bench, a technique for benchmarking path-to-feasibility reasoning in the sciences and demonstrate[1] it by mining 1755 reported results and 3346 methodological contributions from 639 materials science papers. LLMs are prompted to generate ranked hypotheses about the non-trivial methodological innovations required to enable each reported result, and we measure recall against the contributions in the source papers. Mokken scale analysis shows that these performance measures form a strong, unidimensional cumulative scale ($H = 0.625$). We find that the GPT-5 family forms the cost-performance Pareto frontier, with its mid-tier model competitive with both frontier models in the Gemini family at a fraction of the cost. Overall performance is limited, with the best model achieving only 62.5% overall recall@5 and 23.7% of contributions remaining unrecovered by any evaluated system. Finally, we show that technically-trained human judges broadly agree with our automated hit judgments.

1 Introduction

Methodological reasoning is a fundamental meta-skill for researchers in the sciences. Identifying promising research approaches, designing good experiments, and developing descriptive theories are all important skills that efficient researchers develop during their careers. As large language models are incorporated into scientific workflows, they are being explored both as support tools for human scientists and as components of more autonomous research systems. Success in either setting will depend on how well LLM-based systems can suggest research approaches that merit the commitment of resources.

We focus on a capability that we call path-to-feasibility reasoning: the ability to propose methodological contributions that are likely to enable a prespecified scientific outcome. In many cases, the desired outcome of a course of research is not arbitrary. A scientist may have a target material property, performance measure, computational result, or theoretical demonstration in mind when designing an approach. The difficult task is identifying the methods, assumptions, design choices, and practical technical details that make a given target result attainable.

Evaluating this capability is generally difficult because the feasibility of a new research methodology is typically unknown until it has been attempted. An idea may sound plausible but fail for practical reasons, or appear unlikely but succeed because of a previously unknown phenomenon. This makes it hard to score arbitrary model-generated proposals directly. Instead, we use published papers as records of research paths that have already been executed with successful outcomes. In this work, we prompt systems to hypothesize the methodological contributions required to produce a publicly reported scientific result. If two systems rediscover the same methodological contributions reported by human authors at different rates, then we argue that this is evidence of a difference in their path-to-feasibility reasoning capabilities. This approach does not penalize unmatched hypotheses because the truth of their feasibility is unknown.

We present an implementation of this path-to-feasibility evaluation technique and apply it to materials science papers from December 2024. We show that our specific staged question construction

[1] Anonymized code and data is available here: https://anonymous.4open.science/r/method-bench-517F

enables a fair and itemized evaluation of the contributions presented in a paper. We evaluate several language models from the Gemini and GPT families and find significant capability differences between them. We show that the performance metrics produced with our approach are strongly uni-modal, greatly simplifying capability comparisons. Finally, we argue the validity of our results by showing that ground truth items in our dataset are judged to be substantiated by the source paper at a very high rate, and that technically trained human annotators agree with the automated judgments made by our LLM-as-judge framework at a high rate. These results suggest that current systems, given detailed descriptions of a scientific result, can often rediscover some contributions reported by human authors. However, even in this simplified setting, performance on this task is far from saturated, indicating that there is substantial room for capability improvement.

2 Prior Work

There are many opportunities for LLM-based tools to assist with the scientific research process, and existing evaluation techniques already measure many essential scientific competencies.

The application of LLMs to procedural parts of scientific workflows, such as literature review, is already well studied in existing work. The LitQA2 benchmark evaluates the ability of LLMs to solve obscure scientific question-answering problems requiring retrieval of a single specific source (Skarlinski et al., 2024), while ScholarQABench extends this idea to expert-written problems requiring retrieval and synthesis of multiple academic sources (Asai et al., 2026). Other benchmarks such as SciEval and SciKnowEval assess core capabilities such as subject knowledge, scientific reading comprehension, numerical reasoning, and applied science question answering (Sun et al., 2024; Feng et al., 2025). Evaluations like LAB-Bench and LAB-Bench 2 complement broad capability assessments with targeted evaluations designed to test subject-specific skills like accessing and navigating biology databases (Laurent et al., 2024, 2026). While these evaluations successfully assess basic scientific competency, we argue that growing interest in automating the research process necessitates new evaluation techniques that measure the higher-order scientific reasoning skills exercised during research planning and execution.

The substantial increase in language model capabilities in the past two years has enabled new systems that accelerate increasingly high-level research tasks. Human-in-the-loop ideation systems such as IRIS, Acceleron, and IdeaSynth help researchers identify, critique, plan, and ground research proposals using existing literature (Garikaparthi et al., 2025; Nigam et al., 2024; Pu et al., 2025). More automated hypothesis development systems like the AI Co-Scientist demonstrate that test-time compute scaling of multi-agent systems can produce novel and well supported research hypotheses using existing literature (Gottweis et al., 2025). Other systems such as Agent Laboratory and AI Scientist attempt to fully automate the entire research process from initial ideation to research product and even peer review (Schmidgall et al., 2025; Lu et al., 2024).

In each of these settings, the utility and efficiency of the overall system depends heavily on the scientific feasibility of the ideas that it produces. Indeed, the authors of Matter-of-Fact made the same observation- that the cost imbalance between producing ideas and testing ideas critically ties the efficiency of a system to its rate of feasible idea production (Jansen et al., 2025). The Matter-of-Fact benchmark tests the feasibility assessment capabilities of AI systems in materials science, reasoning that feasibility assessment is the core skill required to boost feasible idea production under a hypothesize-and-filter paradigm. IdeaBench and TOMATO-Chem both measure hypothesis generation by grounding LLMs with cited works from a target paper and judging generated ideas against the true paper with quality, feasibility, and similarity metrics (Guo et al., 2025; Yang et al., 2025). A large expert-judged blind-comparison between LLM generated and human expert generated ideas in natural language processing found that humans preferred LLM generated ideas but judged them to be less feasible than those produced by humans (Si et al., 2025).

Concurrent with our work, Chamoun et al. (2026) introduced SciPaths, a benchmark for discovery pathway forecasting in the machine learning and natural language processing domains. Method Bench and SciPaths both attempt to measure how well a system can predict the antecedent ideas that enabled a target result. Method Bench is designed to measure the prediction of novel contributions within the scope of a single paper, awarding no credit to ideas taken from prior work. In con-

trast, SciPaths evaluates the prediction of both established prior art and new, ungrounded contributions, requiring models to identify enabling ideas and retrieve appropriate citations for those that already exist in the literature. This introduces a retrieval and knowledge attribution task to the evaluation that Method Bench does not measure. SciPaths also aggregates ground truth contributions into higher-level canonical descriptions, judging similarity at a broader level of abstraction, rather than isolating granular methodological decisions like Method Bench does. The use of question staging in Method Bench enables fair evaluation against multiple compounding contributions from a single paper, whereas the requirement that named ideas must exhibit re-use elsewhere in the literature renders many dependent procedural and analytical contributions out-of-scope for evaluation in SciPaths.

3 Methodology

3.1 Task Definition

We evaluate *path-to-feasibility reasoning*: the ability to propose the non-obvious methodological choices that make a desired scientific result attainable. Benchmark problems describe a result that was actually reported in a scientific paper and ask an evaluated system to generate hypotheses about the methodological path that enabled it. The ground truth items extracted from each paper in the benchmark are specifically scoped to target core methodological contributions: the technical decisions and innovations that are claimed to have materially contributed to achieving the stated result.

3.2 Source Corpus

The reported benchmark dataset was constructed from materials science papers submitted to arXiv in December 2024 under the `cond-mat.mtrl-sci` category. Rich markdown OCR-ed full-text of these papers generated using Marker (Datalab, 2024) was already available from a prior effort. We further filtered this corpus using Gemini 2.5 Flash to classify suitability from each paper's abstract. The model was instructed to select or reject papers based on whether they appeared to report new scientific results and methodological contributions that could feasibly be re-framed into a path-to-feasibility problem. This step was designed to remove survey papers, position papers, and other works whose abstracts did not indicate that they contained a suitable methodological path. The final benchmark corpus contained 639 papers.

3.3 Problem Stages

Scientific papers often contain multiple substantive contributions that build on one another to enable a headline result. This poses a challenge for designing evaluation questions: what happens when one contribution is only motivated in the context of another contribution? For example, a paper might present a new material synthesis method and then introduce a new analysis method to deduce the importance of different parameters in the synthesis. This introduces a compounding difficulty problem. A system would have to first predict the synthesis approach correctly before it could realistically predict the analysis method.

To address this issue, we organize the benchmark questions for each paper into stages. In Stage 0, a system is tasked with predicting the primary methodological contributions that enabled the headline results of a paper. In Stage 1, a system is shown the question and ground-truth methodology from Stage 0 and asked a follow-up question about a secondary result from the paper. In Stage 2, the questions and answers from Stages 0 & 1 are provided as context. In each stage, additional ground-truth context is revealed, enabling direct evaluation of a system's ability to predict contributions that are conditional on earlier methodology choices.

3.4 Problem and Ground-Truth Construction

To construct a set of staged problems from each paper in the selected corpus, we designed a structured DSPy (Khattab et al., 2024) program to map the OCR-ed text of each paper into a schema-conforming set of problems and ground-truth contributions. We first developed prompts to do this process well on a development set of papers, and then we manually selected high-quality results from this process as bootstrapped few-shot examples to improve the quality of the full benchmark generation.

For each problem stage, the generating model compiled four fields based on the source paper. The `question` re-states a significant result from the source paper, asking a model to predict the enabling contributions; it follows the template: "[*Problem-specific context*]. Researchers have shown, {computationally | experimentally | theoretically}, [*key, non-trivial result from the paper*]. Consider the challenges of demonstrating [*specific re-statement of the result*] and attempt to re-discover the pri-

mary contributions that enabled it. Be as detailed as possible in your independent attempts. Your responses will be graded against the contributions in the actual paper."

A `prior art` field documents background methods and ideas that the authors used but that they do not claim as their own. This context is used during scoring to ensure that hypotheses are not awarded hits for predicting these existing ideas. A `contribution narrative` field describes the actual methodological contributions that enabled the result described in the corresponding question. This prose description may contain multiple substantive ideas. The `itemized contributions` field contains a list of discrete ideas that will be used as scoring units. These are short descriptions of the specific non-obvious methodological choices that appear in the contribution narrative.

GPT-5.2 was used to generate the final evaluation dataset. 1755 staged questions and 3383 itemized contributions were generated from the 639 source papers. Only 3% of papers had more than 3 question stages, so we limit our evaluation to stages 0-2. 42% of problems are self-identified as experimental, 41% as computational, 15% as theoretical, and 2% do not identify a category.

3.5 Evaluated Models

We evaluated seven models spanning each cost tier in the Gemini 2.5, Gemini 3.1, and GPT-5 families. These models were selected because their knowledge cutoff dates fell before or contemporaneous with the publication of the source papers to arXiv. The GPT-5 models have knowledge cutoff dates prior to December 2024, and the Gemini models all share a knowledge cutoff date of January 2025. Although data contamination is a concern for the Gemini models, we reason that the methodological contributions found in the problem corpus are unlikely to have become widely represented in the scientific literature by the Gemini knowledge cutoff date. Moreover, our results suggest that any benefits of data contamination to Gemini's performance only strengthen the interpretation that the GPT-5 family exhibits better path-to-feasibility reasoning.

3.6 Evaluation Procedure

For each question stage in the benchmark, the evaluated system is prompted to produce five independent methodological hypotheses, ordered from most to least likely. Systems are instructed to produce self-contained narrative hypotheses that describe, in as much detail as possible, the non-obvious methodological steps required to achieve the described result. Systems are provided with questions and ground-truth narratives from all prior question stages of a paper as context.

3.7 Scoring

We apply an automated LLM-as-judge protocol to evaluate semantic agreement between the hypotheses proposed by each system and the itemized ground truth contributions in the dataset. For each question in the benchmark, a judge LLM is presented with the question, prior art, contribution narrative, a ground truth item, and the set of predicted hypotheses from a candidate system. The judge is prompted to produce a list rationales and binary hit or miss decisions indicating whether each presented hypothesis contains the specific non-obvious methodological choice described in the ground truth item. The judge is instructed to award hits only to hypotheses that describe what procedures to perform and how to perform them. Motivations, expected experiment outcomes, and other non-methodological details are allowed to differ. Extraneous ideas not present in the ground-truth contribution narrative are not penalized. The binary hit decisions for each hypothesis compared with each ground-truth item are aggregated by stage, and we report recall@k for each model.

4 Results

4.1 Path-to-Feasibility Recall

GPT-5.2 was used to score the hypotheses from each studied model against the ground-truth, and Table 1 reports the measured overall and stage-stratified recall@{1,3,5} for each model on the 3346 ground-truth methodological contributions from the benchmark. GPT-5.1 is the strongest evaluated system across every recall level and question stage, reaching 62.5% recall@5 overall. GPT-5 Mini measured the next strongest, with 54.5% overall recall@5. Among the Gemini systems, Gemini-3.1 Pro performs best, followed by Gemini-2.5 Pro. The weakest systems in the current comparison are Gemini-2.5 Flash-Lite and GPT-5 Nano. We find that the reported performance order holds across almost all recall levels and stages. The only exception to the overall performance order is in Stage 0, where GPT-5 Nano achieves a 2.9% higher recall@1 than Gemini-2.5 Flash, which has an 8.5% higher overall recall@1.

Table 1: Recall at k (R@k) judged by GPT-5.2 for all systems across question stages. There are 3,346 possible hits overall, and the number of hit opportunities in each stage is listed in parentheses. The highest recall values in each column are listed in **bold**.

	Overall (3346)			**Stage 0** (1242)			**Stage 1** (1231)			**Stage 2** (873)		
System	R@1	R@3	R@5	R@1	R@3	R@5	R@1	R@3	R@5	R@1	R@3	R@5
Gemini-2.5 Flash-Lite	0.165	0.253	0.278	0.143	0.235	0.266	0.168	0.253	0.275	0.191	0.277	0.300
GPT-5 Nano	0.235	0.319	0.353	0.246	0.309	0.336	0.219	0.314	0.353	0.244	0.340	0.379
Gemini-2.5 Flash	0.255	0.356	0.389	0.239	0.331	0.367	0.258	0.356	0.386	0.273	0.393	0.423
Gemini-2.5 Pro	0.294	0.392	0.435	0.279	0.369	0.407	0.293	0.385	0.424	0.316	0.434	0.491
Gemini-3.1 Pro	0.334	0.447	0.488	0.291	0.395	0.435	0.344	0.467	0.500	0.383	0.493	0.546
GPT-5 Mini	0.438	0.509	0.545	0.393	0.464	0.494	0.447	0.515	0.554	0.489	0.564	0.604
GPT-5.1	**0.497**	**0.588**	**0.625**	**0.444**	**0.537**	**0.570**	**0.501**	**0.592**	**0.630**	**0.565**	**0.653**	**0.695**

Table 2: Overall Recall at k (R@k) estimates for the top four models when judged by Gemini 3.1 Pro. The absolute change relative to the corresponding recall score when judged by GPT-5.2 (Table 1) is indicated below each value.

System	**R@1**	**R@3**	**R@5**
Gemini 2.5 Pro	0.228 −0.066	0.309 −0.083	0.347 −0.088
GPT-5 Mini	0.295 −0.143	0.341 −0.167	0.365 −0.180
Gemini 3.1 Pro	0.295 −0.040	0.380 −0.066	0.410 −0.077
GPT-5.1	**0.371** −0.125	**0.448** −0.139	**0.484** −0.141

We additionally find that recall rate increases monotonically with question stage. Predictions in later stages benefit from the additional context provided by prior stage questions and ground-truth. This added context prevents the compounding difficulty problem, but it also constrains the hypothesis space, making the prediction problem progressively easier. We observe an average improvement in recall@5 of 18.7% between Stage 0 and Stage 2.

4.2 Robustness to judge selection

It is well documented that LLMs exhibit various biases, including self-preference, when making open-ended judgments in an LLM-as-judge framework (Zheng et al., 2023). To evaluate the stability of the measured performance ordering under an alternative judge, we re-scored the top four models using Gemini 3.1 Pro. The overall recall values are recorded in Table 2. We found that the Gemini model was universally less likely to award hits than GPT-5.2, assigning 44% fewer hits overall. The relative reduction in recall was disproportionately larger for the GPT family than the Gemini family when switching to the Gemini judge. Regardless of the underlying cause for this discrepancy, both judges agree that GPT-5.1 is the highest performing model and that GPT-5 Mini out-performs Gemini 2.5 Pro. When judged by Gemini 3.1 Pro, the performance difference between GPT-5 Mini and Gemini 3.1 Pro narrows, with both models achieving the same recall@1 and Gemini 3.1 Pro outperforming the cheaper model in recall@5. These results validate the observed performance trends, but they suggest that the exact performance order of closely competing models from different families is likely to change subject to the judge.

4.3 Cost-performance tradeoffs

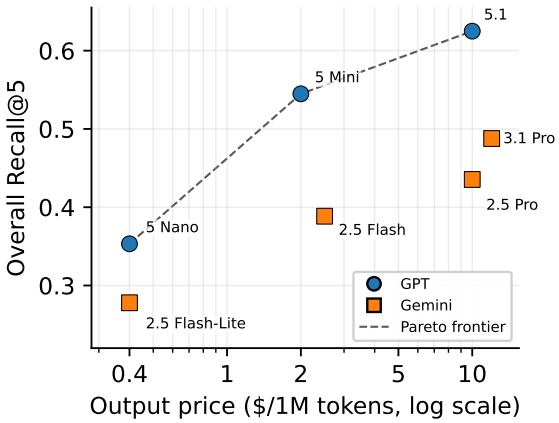

Figure 1: Overall recall@5 versus output-token price in USD per million tokens for each studied model. The dashed line illustrates the Pareto frontier formed by the models in the GPT-5 family. Full price-normalized values are listed in Appendix A.

API pricing generally determines which commercial models are direct competitors of each other, and it can serve as a rough indication of the performance and inference costs expected by a model

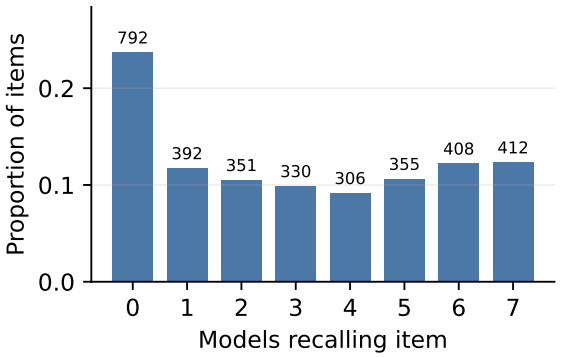

Figure 2: Distribution of ground-truth contribution difficulty, as measured by the number of unique models that recall each item at any recall level.

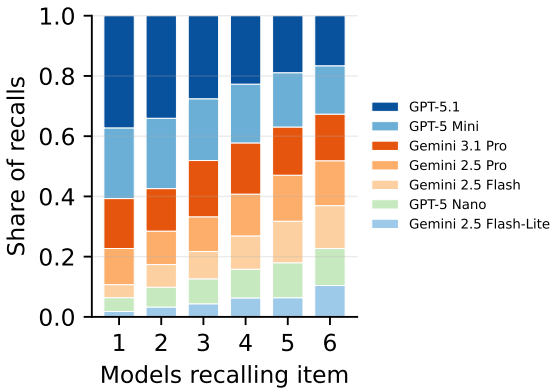

Figure 3: Share of overall hits@5 attributable to each model, broken down by difficulty (as measured by number of models recalling). Stronger models claim larger proportions of difficult hits.

provider. In Figure 1, we compare the overall recall@5 for each studied model with its output price per million tokens (Appendix A). The studied models cluster into three price strata. Across the three cost tiers, we observe that the GPT-5 family substantially out-performs the Gemini models, forming the Pareto frontier of the benchmark. The standout performer is GPT-5 Mini which outperforms Gemini 2.5 Pro at 1/5th the price and competes closely with Gemini 3.1 Pro, beating it when judged by GPT-5.2 and matching it in recall@1 when judged by Gemini 3.1 Pro.

4.4 Contribution difficulty

The measured recall rates in Table 1 suggest that benchmark performance is far from saturated by the studied models. Across all stages, 792 ground truth contributions (23.7% of the total) were not predicted by any model. To investigate the overall problem difficulty distribution, we use the number of models that correctly predict each ground truth contribution as a proxy for contribution difficulty. Figure 2 depicts the distribution of contributions by the number of models that identified them. We find a broadly even distribution, with unpredicted contributions comprising the largest category.

In Figure 3, we further examine which models account for the recalls in each non-trivial difficulty bin. We find that the strongest models account for a larger share of the uniquely predicted contributions, with GPT-5.1 accounting for 37.2% of the unique solves, followed by GPT-5 Mini with 23.5%. The share of solves attributable to GPT-5.1 grew substantially with problem difficulty, while the share attributable to GPT-5 Mini grew only mildly. The Gemini Pro models contributed a consistent proportion of solves across each bucket, while the weakest models contributed progressively decreasing proportions of solves. These patterns show that higher recall models capture larger shares of hard problem solves, but that Guttman errors are still common, with lower-capability models frequently predicting contributions that higher-capability models miss. This suggests that there is still a high degree of stochasticity to the evaluation.

4.5 Scalability

Figure 3 exhibits trends consistent with a unidimensional item response structure, where stronger models tend to recall supersets of the contributions recalled by weaker models. Borrowing from the field of psychometrics, we tested this statistically by constructing the binary hit matrix for the studied models and computing Loevinger's H at each recall level (Loevinger, 1948). We found $H = 0.545$ for R@1, $H = 0.602$ for R@3, and $H = 0.625$ for R@5. Under conventional interpretations of this coefficient, $H > 0.5$ indicates strong Mokken scalability (Sijtsma and Molenaar, 2002). Practically, this means that pairwise upsets, cases where a weaker model recalled a contribution missed by a stronger model, were substantially less common than would be expected if model hits were independent conditional on their marginal recall rates.

This evidence supports the interpretation that the benchmark is largely unidimensional. Contributions appear to have characteristic empirical difficulty levels, and models appear to have characteristic empirical capability levels. More capable models are more likely to recall harder contribu-

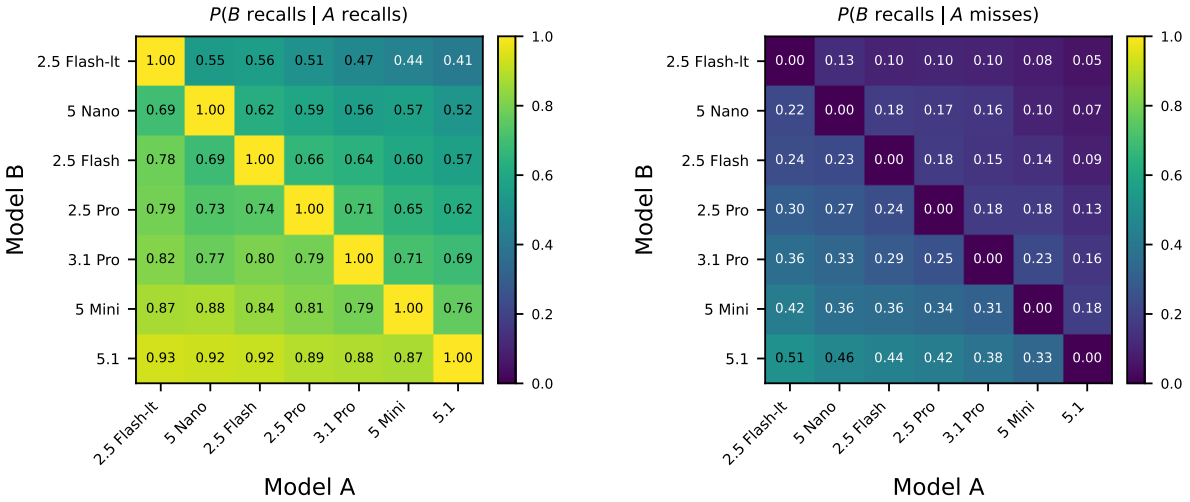

Figure 4: Conditional recall structure at $k = 5$. Left: $P(B \text{ recalls} \mid A \text{ recalls})$. Right: $P(B \text{ recalls} \mid A \text{ misses})$. In both panels, model A is shown on the x-axis and model B on the y-axis; models are ordered from weakest to strongest overall recall.

tions. Figure 4 visualizes these same effects by plotting the measured conditional probability that any model recalls a contribution given that another model recalls, or fails to recall, the same contribution. For example, when GPT-5.1 fails to predict a contribution, GPT-5 Mini only recalls that contribution 18% of the time. As predicted by the Mokken scalability result, this rate decreases monotonically with model ranking. The next-best model, Gemini 3.1 Pro, only recalls GPT-5.1's missed contributions 16% of the time, and the weakest model, Gemini 2.5 Flash-Lite, only 5% of the time.

4.6 Observed generation failures

All models were instructed to generate five hypotheses per staged problem, but we found that around 5% of hypothesis sets contained generation failures that artificially reduced the number of predicted hypotheses. Examination suggests that missing hypotheses were primarily attributable to parsing and structured generation issues terminating hypotheses early or interpreting redundant delimiters as hypotheses. To assess the impact of these failures, we performed two data filtering analyses. First, we excluded question stages with generation failures on an individual model basis and recomputed the overall recall@5 on the remaining problems. The largest measured change in overall recall@5 was from Gemini 2.5 Flash (+1.11 pp), followed by Gemini 2.5 Flash-Lite (+0.72 pp) and GPT-5 Mini (+0.38 pp). In the second filtering analysis, we constructed a common evaluation set containing the intersection of all unaffected problems across models. Again, we found that Gemini 2.5 Flash was the most impacted (+2.61 pp) in overall recall@5. The next largest changes were Gemini 2.5 Pro (-1.25 pp), and Gemini 2.5 Flash-Lite (+0.80 pp). Gemini 2.5 Pro was unaffected by generation failures, so its performance loss on the sliced data is an artifact of the test. The full set of results for both filtering methods is reported in Tables 6 and 7 of Appendix F.

The nature of the failures meant that recall@1 measurements were unaffected, and neither filtering technique changed the relative performance order of the studied models. Our results suggest that Gemini 2.5 Flash was the most affected, and that its true recall rate is closer to that of Gemini 2.5 Pro than reflected in Table 1.

5 Human Validation

To validate the presented benchmarking protocol, we conducted a human annotation study designed to assess the quality of the ground truth extractions and the automated LLM-as-judge verdicts. Sixteen non-specialist human annotators were recruited for this study. All annotators had graduate degrees, or were current graduate students in a technical field such as computer science or electrical engineering. Annotators were asked to complete two annotation tasks. The first measured the rate at which automatically extracted ground truth contributions were substantiated by their source papers. The second tasked annotators with independently

judging whether model predictions contained the same core methodological idea itemized in a problem's ground truth.

In the data substantiation task (Appendix G.1), annotators were shown ground-truth contribution narratives and asked to judge whether or not an itemized contribution from that narrative accurately reflected the content of the source paper. Evidence spans that were extracted during the dataset construction process where presented to the annotators to facilitate this process. Links to the full text of each paper were also provided for cases when the automatically extracted spans were insufficient to determine substantiation. Annotators reviewed and marked 152 of 160 ground-truth contributions as substantiated by their source papers, yielding a 95.0% substantiation rate (95% Wilson CI: 90.4-97.4%). This suggested that the LLM-extracted contributions reliably represented the claimed methodological contributions of authors in the source papers.

In the scoring task (Appendix G.2), annotators were shown pairs of hypotheses and itemized contributions and asked to judge whether the presented hypothesis contained the same concrete methodological idea as the itemized ground-truth contribution. Annotators were instructed to disregard non-methodological details such as expected outcomes, motivations, stated intuition, or any other explanatory framing or reasoning. Problem assignments were intentionally balanced, such that each annotator saw a roughly even sample of judged-hit and judged-miss pairs from each studied model.

In total, 289 judgments were recorded. 34 items received triplicate judgments from multiple annotators, while 8 items received duplicate judgments. 7 judgments were found to correspond to trivial misses- generation failures where a model produced fewer than ten characters for a hypothesis. Without specific guidance on how to assess such hypotheses, some annotators judged them as misses and others flagged them as "too vague to judge". We normalize these judgments as misses. Across all annotations, $234/289 = 81.0\%$ (95% Wilson CI: 76.0-85.1%) agreed with the automated LLM-as-judge. Comparing the human majority-vote to the automated judge (excluding the single tie), $175/212 = 82.5\%$ (95% Wilson CI: 76.9-87.1%) of verdicts agreed. We found that the annotators agreed with hit verdicts at a rate of $92/107 = 86.0\%$ (95% Wilson CI: 78.2-91.3%), while they only agreed with miss verdicts at a rate of $84/106 = 79.2\%$ (95% Wilson CI: 70.6-85.9%). This suggested that the non-expert annotators tended to be more lenient when attributing hits than the LLM-as-judge. Pairwise inter-annotator agreement among the triplicate and duplicate judgments was found to be moderate, with a Cohen's $\kappa = 0.436$ and $79/110 = 71.8\%$ (95% Wilson CI: 62.8-79.4%) of annotation pairs agreeing.

6 Conclusions

We introduced Method Bench, an automated empirical method for measuring path-to-feasibility reasoning in scientific domains. The scientific reasoning skill that Method Bench measures differs from other capabilities tested by existing techniques, aligning more closely with the goal-oriented planning that scientists often perform in the early stages of a research project. We showed that problems can be mined with high fidelity from scientific papers and that automatic evaluation against the ground truth is feasible and broadly aligns with human judgment. Our staged evaluation structure allowed us to measure recall against secondary and tertiary contributions in a fair way.

Our findings suggest that the proposed evaluation technique produces unsaturated measurements of the studied capability in recent models while revealing substantial performance differences across model families. Even accounting for judge biases, we found that the flagship Gemini 3.1 Pro model performed similarly to the far cheaper GPT-5 Mini, indicating that path-to-feasibility reasoning *efficiency* can differ substantially between architectures and training recipes. Additionally, our performance measures were consistent with a strong item-response hypothesis, suggesting that Method Bench tests a single dominant axis of capability rather than many competing orthogonal capabilities. This finding makes Method Bench a good candidate for evaluating future techniques that improve path-to-feasibility reasoning.

The published software documents how Method Bench can be reproduced for papers in other subject areas and time frames. While the dataset that we release demonstrates an instantiation of the Method Bench protocol for materials science, we view the methodology itself as our most durable contribution. It enables a practical on-demand evaluation of path-to-feasibility reasoning across scientific subject areas and can be reproduced on publications contemporary with future model releases.

7 Limitations

Several limitations should guide interpretation of the presented results. First, the human validation study of the LLM-as-judge framework was conducted with sixteen technical non-expert annotators. The collected judgments are broadly aligned with those of the LLM-as-judge; however, they may not align with the judgment of expert materials scientists. Second, the knowledge cutoff dates of the Gemini models overlapped with the publication of the source corpus by a month. Data contamination is likely, although any benefits received by the Gemini family only strengthen the interpretation that the GPT-5 family performs better on our benchmark. Third, we found that structured generation and response parsing errors disproportionately affected Gemini 2.5 Flash, likely reducing its measured recall performance artificially. Finally, we observed that absolute recall levels can vary significantly based on the LLM-as-judge model, and that inter-family differences were unstable when changing judge models. We found that our primary performance trends were consistent between judge models; however, our observations are consistent with known limitations of LLM-as-judge methodologies.

References

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, Jenna Sparks, Jena D. Hwang, Varsha Kishore, Minyang Tian, Pan Ji, Shengyan Liu, Hao Tong, and 9 others. 2026. Synthesizing scientific literature with retrieval-augmented language models. *Nature*, 650(8103):857–863.

Eric Chamoun, Yizhou Chi, Yulong Chen, Rui Cao, Zifeng Ding, Michalis Korakakis, and Andreas Vlachos. 2026. Scipaths: Forecasting pathways to scientific discovery. *Preprint*, arXiv:2605.14600.

Datalab. 2024. Marker: Convert pdf to markdown and json. GitHub repository.

Kehua Feng, Xinyi Shen, Weijie Wang, Xiang Zhuang, Yuqi Tang, Qiang Zhang, and Keyan Ding. 2025. Sciknoweval: Evaluating multi-level scientific knowledge of large language models. *Preprint*, arXiv:2406.09098.

Aniketh Garikaparthi, Manasi Patwardhan, Lovekesh Vig, and Arman Cohan. 2025. IRIS: Interactive research ideation system for accelerating scientific discovery. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 592–603, Vienna, Austria. Association for Computational Linguistics.

Juraj Gottweis, Wei-Hung Weng, Alexander Daryin, Tao Tu, Anil Palepu, Petar Sirkovic, Artiom Myaskovsky, Felix Weissenberger, Keran Rong, Ryutaro Tanno, Khaled Saab, Dan Popovici, Jacob Blum, Fan Zhang, Katherine Chou, Avinatan Hassidim, Burak Gokturk, Amin Vahdat, Pushmeet Kohli, and 15 others. 2025. Towards an ai co-scientist. *Preprint*, arXiv:2502.18864.

Sikun Guo, Amir Hassan Shariatmadari, Guangzhi Xiong, Albert Huang, Myles Kim, Corey M. Williams, Stefan Bekiranov, and Aidong Zhang. 2025. Ideabench: Benchmarking large language models for research idea generation. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2*, KDD '25, page 5888–5899, New York, NY, USA. Association for Computing Machinery.

Peter Jansen, Samiah Hassan, and Ruoyao Wang. 2025. Matter-of-fact: A benchmark for verifying the feasibility of literature-supported claims in materials science. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 4090–4102, Suzhou, China. Association for Computational Linguistics.

Jina AI. 2024. Reader: Convert any url to an llm-friendly input with a simple prefix. `https://github.com/jina-ai/reader`.

Saleh Ahmed Khan, Ahmed Ibreljic, Stephen Margiotta, and A. F. M. Anhar Uddin Bhuiyan. 2025. Low-pressure cvd grown si-doped -ga2o3 films with promising electron mobilities and high growth rates. *Applied Physics Letters*, 126(1).

Omar Khattab, Arnav Singhvi, Paridhi Maheshwari, Zhiyuan Zhang, Keshav Santhanam, Sri Vardhamanan, Saiful Haq, Ashutosh Sharma, Thomas T. Joshi, Hanna Moazam, Heather Miller, Matei Zaharia, and Christopher Potts. 2024. Dspy: Compiling declarative language model calls into self-improving pipelines.

Jon M Laurent, Albert Bou, Michael Pieler, Conor Igoe, Alex Andonian, Siddharth Narayanan, James Braza, Alexandros Sanchez Vassopoulos, Jacob L Steenwyk, Blake Lash, Andrew D White, and Samuel G Rodriques. 2026. Labbench2: An improved benchmark for ai systems performing biology research. *Preprint*, arXiv:2604.09554.

Jon M. Laurent, Joseph D. Janizek, Michael Ruzo, Michaela M. Hinks, Michael J. Hammerling, Siddharth Narayanan, Manvitha Ponnapati, Andrew D. White, and Samuel G. Rodriques. 2024. Lab-bench: Measuring capabilities of language models for biology research. *Preprint*, arXiv:2407.10362.

Jane Loevinger. 1948. The technic of homogeneous tests compared with some aspects of scale analysis

and factor analysis. *Psychological bulletin*, 45 6:507–29.

Chris Lu, Cong Lu, Robert Tjarko Lange, Jakob Foerster, Jeff Clune, and David Ha. 2024. The ai scientist: Towards fully automated open-ended scientific discovery. *Preprint*, arXiv:2408.06292.

Harshit Nigam, Manasi Patwardhan, Lovekesh Vig, and Gautam Shroff. 2024. An interactive co-pilot for accelerated research ideation. In *Proceedings of the Third Workshop on Bridging Human–Computer Interaction and Natural Language Processing*, pages 60–73, Mexico City, Mexico. Association for Computational Linguistics.

Kevin Pu, K. J. Kevin Feng, Tovi Grossman, Tom Hope, Bhavana Dalvi Mishra, Matt Latzke, Jonathan Bragg, Joseph Chee Chang, and Pao Siangliulue. 2025. Ideasynth: Iterative research idea development through evolving and composing idea facets with literature-grounded feedback. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, CHI '25, New York, NY, USA. Association for Computing Machinery.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Samuel Schmidgall, Yusheng Su, Ze Wang, Ximeng Sun, Jialian Wu, Xiaodong Yu, Jiang Liu, Michael Moor, Zicheng Liu, and Emad Barsoum. 2025. Agent laboratory: Using LLM agents as research assistants. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 5977–6043, Suzhou, China. Association for Computational Linguistics.

Chenglei Si, Diyi Yang, and Tatsunori Hashimoto. 2025. Can LLMs generate novel research ideas? a large-scale human study with 100+ NLP researchers. In *The Thirteenth International Conference on Learning Representations*.

K. Sijtsma and I.W. Molenaar. 2002. *Introduction to Nonparametric Item Response Theory*. Measurement Methods for the Social Science. SAGE Publications.

Michael D. Skarlinski, Sam Cox, Jon M. Laurent, James D. Braza, Michaela Hinks, Michael J. Hammerling, Manvitha Ponnapati, Samuel G. Rodriques, and Andrew D. White. 2024. Language agents achieve superhuman synthesis of scientific knowledge. *Preprint*, arXiv:2409.13740.

Liangtai Sun, Yang Han, Zihan Zhao, Da Ma, Zhennan Shen, Baocai Chen, Lu Chen, and Kai Yu. 2024. Scieval: A multi-level large language model evaluation benchmark for scientific research. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(17):19053–19061.

Zonglin Yang, Wanhao Liu, Ben Gao, Tong Xie, Yuqiang Li, Wanli Ouyang, Soujanya Poria, Erik Cambria, and Dongzhan Zhou. 2025. Moose-chem: Large language models for rediscovering unseen chemistry scientific hypotheses. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

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. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA. Curran Associates Inc.

A API Pricing Details

All model costs were retrieved from their providers' API documentation on May 10, 2026. Wherever a provider distinguishes between token prices for prompts greater than or less than a certain threshold in length, we take the lower price. Table 3 reports the assumed input and output token prices for each model, alongside price normalized overall recall@5 measurements.

B Retrieval-Augmented Search Agent

To assess whether retrieval augmentation improves recall on Method Bench, we evaluate a search-augmented agent on a randomly held-out 25% subsample of the benchmark problems (852 scored contribution items across stages 0-2). The agent is implemented as a DSPy ReAct loop built on GPT-5.1 and equipped with two MCP tools: `search_arxiv`, which performs semantic search against a local vector database of arXiv abstracts indexed using `sentence-transformers/all-mpnet-base-v2` (Reimers and Gurevych, 2019); and `read_url`, which fetches the full content of any URL in Markdown format via the Jina reader API (Jina AI, 2024). To prevent temporal information leakage into the December 2024 evaluation set, a knowledge cutoff is enforced server-side on `search_arxiv`, excluding papers submitted on or after that month. The GPT-5.1 baseline is reported on the same subsample to enable a controlled comparison, and scoring for both systems uses the same LLM-as-judge protocol described in Section 3.7 with GPT-5.2 as the judge model.

Table 4 reports the results of the comparison. The search agent improves over the GPT-5.1 baseline on every recall level in aggregate, reaching

Table 3: Overall recall@5 normalized by input and output token cost in USD per million tokens.

System	R@5	Input $/MTok	R@5/Input	Output $/MTok	R@5/Output
GPT-5 Nano	0.353	0.05	7.07	0.40	0.883
Gemini-2.5 Flash-Lite	0.278	0.10	2.78	0.40	0.695
GPT-5 Mini	0.545	0.25	2.18	2.00	0.272
Gemini-2.5 Flash	0.389	0.30	1.30	2.50	0.155
GPT-5.1	0.625	1.25	0.50	10.00	0.062
Gemini-2.5 Pro	0.435	1.25	0.35	10.00	0.044
Gemini-3.1 Pro	0.488	2.00	0.24	12.00	0.041

Table 4: Comparison: GPT-5.1 vs Search Agent

	Overall (852)			**Stage 0** (312)			**Stage 1** (324)			**Stage 2** (216)		
System	R@1	R@3	R@5	R@1	R@3	R@5	R@1	R@3	R@5	R@1	R@3	R@5
GPT-5.1	0.4836	0.5857	0.6256	0.4423	0.5321	0.5641	0.4691	0.5895	**0.6265**	0.5648	**0.6574**	**0.7130**
Search Agent	**0.5188**	**0.6138**	**0.6385**	**0.4904**	**0.6026**	**0.6314**	**0.5123**	**0.6049**	**0.6265**	**0.5694**	0.6435	0.6667

51.9% recall@1 versus 48.4% for the base model (+3.5 pp), and 63.9% recall@5 versus 62.6% (+1.3 pp). The gains are largest at earlier stages: at stage 0, the search agent improves by +4.8 pp recall@1 (0.490 vs. 0.442) and +6.7 pp recall@3 (0.603 vs. 0.532). At stage 1, recall@1 improves by +4.3 pp while recall@5 is unchanged (0.627 in both systems). At stage 2, the search agent matches the baseline at recall@1 (0.569 vs. 0.565) but falls below it at recall@3 (0.644 vs. 0.657) and recall@5 (0.667 vs. 0.713).

The gains are largest at early stages, where problem context is sparse and access to prior literature can supply relevant domain vocabulary and methodological priors that help the model generate more targeted hypotheses. As more ground-truth context is revealed in later stages, this advantage diminishes. The problem itself already encodes the domain framing, and unfocused retrieval may introduce noise rather than useful context. Importantly, even with retrieval augmentation the absolute recall improvements are modest. This evidence supports the interpretation that retrieval from literature is helpful for path-to-feasibility reasoning, but it does not dramatically transform it.

C Category-Stratified Recall

To examine whether path-to-feasibility reasoning capabilities vary by research methodology type, we stratify the 3,346 ground-truth contribution items by the problem category assigned during dataset construction: *Experimental*, *Computational*, *Theoretical*, and *Other*. The LLM judge determines the category labels by the stage-0 prompt text, where the judge determines whether the researchers demonstrated their result experimentally, computationally, theoretically or in another way. The *Other* category consists of problems that do not belong to a single category and are primarily hybrid methodologies. The benchmark contains 1,064 experimental items, 1,242 computational items, 339 theoretical items, and 703 unknown items. We do not find statistically significant differences in overall difficulty across the three main categories using problem-clustered bootstrap confidence intervals, though descriptive patterns are worth examining.

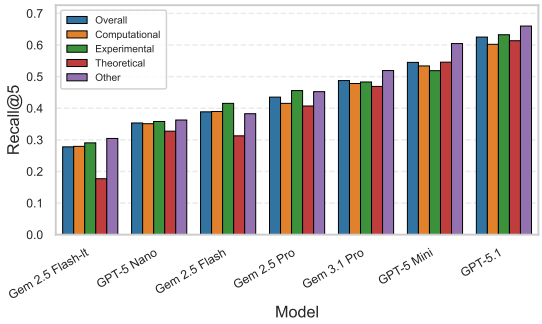

Figure 5: Recall@5 for each evaluated model stratified by problem category (Overall, Computational, Experimental, Theoretical, Other).

C.1 Recall@5 by Category and Model

Figure 5 and the corresponding heatmap in Figure 6 show recall@5 for each model broken down by problem category. The most consistent pattern across all models is that *Theoretical* items are the hardest category. For the weakest evaluated model, Gemini-2.5 Flash-Lite, theoretical recall@5 is 0.18, compared to 0.28-0.30 for the

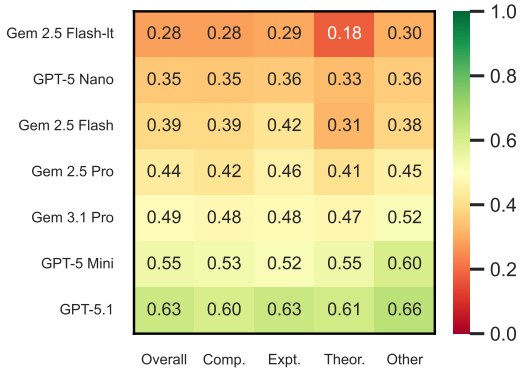

Figure 6: Recall@5 heatmap by model and problem category. Models are ordered from weakest (top) to strongest (bottom) overall recall.

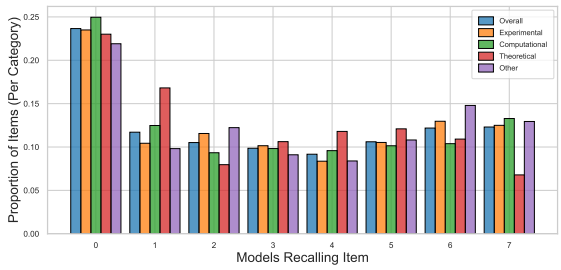

Figure 7: Distribution of per-item difficulty by problem category, measured as the number of models (out of 7) that recalled each ground-truth contribution at recall@5.

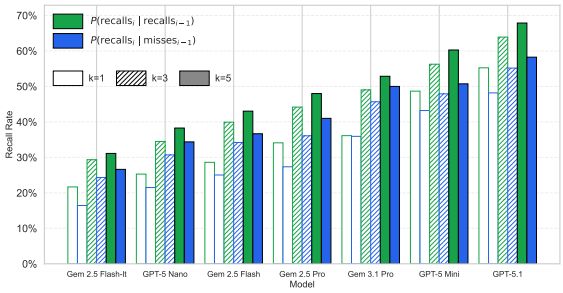

Figure 8: Stage-conditional recall rates at $k \in \{1, 3, 5\}$. For each model and recall level, bars show the probability that the model recalls a stage given that it recalls successfully or misses in immediately preceding stage for the same paper.

remaining categories—a gap of roughly 10 percentage points. This gap is largest for the weaker models and narrows as the models' capabilities grow. The two strongest models, GPT-5 Mini and GPT-5.1 do not have theoretical as their weakest category by a few percentage points while all the other models do. This convergence suggests that the difficulty penalty associated with theoretical reasoning is not a ceiling effect but rather a capability gap that stronger models partially close.

Computational and experimental items perform comparably across models, with experimental items scoring slightly higher than computational items at several capability tiers (e.g., 0.46 vs. 0.42 for Gemini-2.5 Pro; 0.63 vs. 0.60 for GPT-5.1). This aligns with the intuition that experimental methods may be easier to describe at the level of specificity the benchmark rewards, while computational contributions sometimes involve subtle algorithmic choices that require deeper domain knowledge to reconstruct.

C.2 Item Difficulty Distribution by Category

The distribution of per-item difficulty—measured as the number of models that recalled each contribution—is roughly uniform across categories (Figure 7), consistent with the absence of a statistically significant category effect on overall difficulty. The zero-recall proportions are nearly identical across categories: 25.0% of computational items, 23.5% of experimental items, and 23.0% of theoretical items were recalled by no model. This suggests that theoretical problems are not simply harder to get off the ground—the floor is shared across categories. The divergence appears instead at the ceiling: only 6.8% of theoretical items were recalled by all seven models, compared to 13.3% of computational items and 12.5% of experimental items. In other words, theoretical contributions are roughly half as likely to be consensus-easy compared to other categories. This asymmetry reinforces the interpretation that theoretical methodology poses a qualitatively harder prediction problem: not more items that stump everyone, but substantially fewer items that every capable model can reliably reconstruct.

D Recall Evolves over Stages

Our multi-stage benchmark design (Section 3.3 reveals not only that recall evolves across stages but also whether a model's performance for one stage can predict performance for the following stages of the same paper. To quantify this, we compute the probability that the model correctly recalls at stage *i* given it was correct (recalled successfully) or incorrect (a miss) in stage *i-1*.

Figure 8 illustrates that across every model and recall level $k \in \{1, 3, 5\}$, successful recall at a given stage is somewhat more likely $(5{-}10\%)$ if

the model was successful at the immediately preceding stage. This gap is consistent with two reinforcing mechanisms. First, some papers may be intrinsically easier across all stages, so a model that predicts one contribution well is likely operating on a problem where other contributions are also accessible. Second, models that construct internally coherent methodological narratives in earlier stages may produce hypotheses in later stages that are better aligned with the actual paper methodology. These two mechanisms are difficult to disentangle, but both imply that early-stage recall is a useful predictor of later-stage recall within a paper.

E Problem Generation Criteria

The prompt used to extract Method Bench problems from source papers established strict definitions and rejection tests for identifying and recording contributions. Rather than broadly summarizing a paper's methodology, the generation model was instructed to isolate only the most non-obvious, fundamentally predictable methodological decisions. We summarize the primary criteria outlined in the prompt below. The full prompts are included in the accompanying software.

Rejection Tests for Contributions: Before finalizing any ground-truth contribution, the generation model was instructed to apply five strict rejection tests to better exclude standard practices and incidental methodological details from the benchmark:

1. **Counterfactual (Standard Practice):** Would any competent scientist with the same goal and prior-stage context be very likely to make the same choice? If yes, the idea is likely standard practice and not a contribution. Routine tool application, standard characterization, and obvious candidate enumeration fail this test.

2. **Verification (Robustness):** Does the idea merely confirm, validate, or demonstrate the robustness of something already captured by another itemized contribution? Testing and verification of prior contributions is standard scientific practice and not evidence of a standalone contribution.

3. **Separability (Dependency):** Could someone with knowledge of the previous contributions accomplish this without additional non-obvious insight? Immediate algebraic consequences and obvious next steps should be merged or removed.

4. **Discovery vs. Design (Methodology):** Is this an observation about nature or a choice the authors made? Physical mechanisms and unexpected properties are results, not methodology, unless the analytical design revealing them was itself non-obvious.

5. **Attribution (Novelty):** Is the idea new to this paper? Established practice or methods from the authors' own prior work are excluded as prior art.

Defining True Contributions: The prompt strictly penalizes false positives, instructing the model to err toward undercounting to protect the sensitivity of the evaluation. Contributions must be non-obvious choices that the authors made, where many alternatives existed, and that materially contributed to the result. Furthermore, contributions must be fundamentally predictable; incidental discoveries about the natural world are explicitly excluded.

Question Formulation and Staging: To prevent data leakage and handle logical dependencies, the generation of questions and ground-truth items is subject to additional structural constraints dictated by the generation prompt:

- **Preventing Method Leakage:** The question must concretely describe the target achieved result without leaking any non-trivial information about the methods used. Incidental specific measurements should be replaced with generic ranges where appropriate.

- **Sequential Dependency:** Because secondary innovations are impossible to predict without context from preceding decisions, contributions are broken into sequential stages. Questions are ordered so that contributions dependent on previous decisions or results are presented only after the context they depend on is established.

- **The Scientist's Perspective:** Questions are scoped to mirror real-world research planning, reflecting the level of detail that a scientist would possess when envisioning a target result at the beginning of a project.

F Investigating Generation Failures

Although all models were instructed to predict five hypotheses per staged problem, some responses failed to satisfy this format. Across all evaluated

Table 5: **Prevalence of generation failures by model.** The most commonly observed failure mode was generation of fewer than five hypotheses (< 5 hyp.). The secondary observed failure mode was generation of very short hypotheses containing only whitespace or punctuation. We generalize short hypotheses to be all generations with fewer than ten characters. Possibly missed hits are ground-truth items that were not recalled by a model and corresponded to a hypothesis set with generation failures.

System	Short hyp. stages	< 5 hyp. stages	Any-failure stages	Possibly missed hits
Gemini-2.5 Flash	31/1,755 (1.77%)	350/1,755 (19.94%)	380/1,755 (21.65%)	470/3,346 (14.05%)
Gemini-2.5 Flash-Lite	10/1,755 (0.57%)	134/1,755 (7.64%)	143/1,755 (8.15%)	215/3,346 (6.43%)
Gemini-2.5 Pro	0/1,755 (0.00%)	0/1,755 (0.00%)	0/1,755 (0.00%)	0/3,346 (0.00%)
Gemini-3.1 Pro	0/1,755 (0.00%)	2/1,755 (0.11%)	2/1,755 (0.11%)	2/3,346 (0.06%)
GPT-5 Mini	93/1,755 (5.30%)	20/1,755 (1.14%)	113/1,755 (6.44%)	113/3,346 (3.38%)
GPT-5 Nano	15/1,755 (0.85%)	16/1,755 (0.91%)	31/1,755 (1.77%)	41/3,346 (1.23%)
GPT-5.1	248/1,755 (14.13%)	25/1,755 (1.42%)	261/1,755 (14.87%)	188/3,346 (5.62%)

Table 6: **Per-model filtered recall performance.** For each model, question stages with observed generation failures were excluded and overall recall@5 was recomputed. Δ pp indicates the percentage point change in overall recall@5 caused by excluding stages with generation failures.

System	Retained targets	Excluded targets	Excluded R@5	Filtered R@5	Δ pp
Gemini-2.5 Flash	2,625	721 (21.55%)	0.3481	0.3996	+1.11
Gemini-2.5 Flash-Lite	3,079	267 (7.98%)	0.1948	0.2852	+0.72
Gemini-2.5 Pro	3,346	0 (0.00%)	–	0.4354	+0.00
Gemini-3.1 Pro	3,342	4 (0.12%)	0.5000	0.4877	-0.00
GPT-5 Mini	3,124	222 (6.63%)	0.4910	0.5487	+0.38
GPT-5 Nano	3,286	60 (1.79%)	0.3167	0.3539	+0.07
GPT-5.1	2,845	501 (14.97%)	0.6248	0.6250	+0.00

systems, 547 of 12,285 hypothesis lists (4.45%) contained fewer than five hypotheses, and 743 of 60,006 parsed hypotheses (1.24%) contained fewer than 10 non-whitespace characters. Investigation suggests that these failures were primarily caused by parsing and structured generation issues. For instance, in most of GPT-5.1's failed responses, it produced three real hypotheses in positions 1,3,5 and only a comma in positions 2,4. We found that models exhibited these random generation failures at different rates, prompting a follow-up evaluation to determine whether these effects explain the measured performance differences between models.

Upon investigation, we found that all models consistently generated their first hypothesis correctly, meaning that recall@1 measurements were unaffected by either generation failure mode. We also found that neither Gemini Pro model was affected at a meaningful rate. Generation failures in batches of hypotheses that already correctly recall their target contributions have no impact on recall@5. We computed the number of missed hit opportunities that correspond to hypothesis sets with generation failures, and we found that Gemini 2.5 Flash was the most affected with 470/3346 (14.0%) of its judged-misses@5 potentially affected, followed by Gemini 2.5 Flash-Lite at 215/3346 (6.4%) and GPT-5.1 at 188/3346 (5.6%). Table 5 contains the full breakdown.

To rule out generation failures as a primary driver of the observed performance differences between models, we performed two separate data filtering analyses. First, for each model individually, we excluded question stages with generation failures and recomputed the overall recall@5 on the remaining problem sets. We found that the largest measured change in performance was from Gemini 2.5 Flash, which gained (+1.11 pp) recall@5, followed by Gemini 2.5 Flash-Lite at (+0.72 pp) and GPT-5 Mini at (+0.38 pp). The full set of results is found in Table 6.

In the second data filtering experiment, we constructed a common evaluation set composed from the intersection of all unaffected problems across models. Again, we found that the largest measured change in performance on this data subset was from Gemini 2.5 Flash which gained (2.61 pp) in overall recall@5. The next largest changes were Gemini 2.5 Pro which lost (1.25 pp), and Gemini 2.5 Flash-Lite which gained (0.80 pp). Gemini 2.5 Pro was unaffected by generation failures, so the measured performance loss was an artifact of the re-sliced dataset. The full set of results is found in Table 7.

Neither filtering technique changed the relative

Table 7: Shared-clean intersection robustness analysis. We restricted evaluation to the common subset of stage 0–2 problems for which every model produced five non-short hypotheses. This retained 998/1,736 stages (57.49%) and 1,920/3,346 item-level hit opportunities (57.38%). Recall@5 values changed by at most 2.61 percentage points and the model ranking was unchanged.

System	Original R@5	Excluded R@5	Shared-clean R@5	Δ pp
Gemini-2.5 Flash	0.3885	0.3534	0.4146	+2.61
Gemini-2.5 Flash-Lite	0.2779	0.2672	0.2859	+0.80
Gemini-2.5 Pro	0.4354	0.4523	0.4229	-1.25
Gemini-3.1 Pro	0.4877	0.4902	0.4859	-0.18
GPT-5 Mini	0.5448	0.5449	0.5448	-0.00
GPT-5 Nano	0.3533	0.3457	0.3589	+0.56
GPT-5.1	0.6249	0.6234	0.6260	+0.11

performance ordering of the studied models. We conclude that the most likely interpretation of these results is that the particularly high rate of generation failures exhibited by Gemini 2.5 Flash artificially depressed its recall performance, and that its true rate is closer to that of Gemini 2.5 Pro than suggested by the raw results.

G Human Annotation Details

The sixteen annotators used in the human annotation study were recruited from the respective labs of the authors' institutions. Annotators completed two validation tasks using a browser-based annotation interface. They were first shown two 10 min training videos that explained the annotation interface, the structure of the two tasks, and the intended purpose of their annotations. Figure 10 is a screenshot of the annotation interface and the options presented to the annotators in the first task. In both tasks, annotators were given access to an "export context" button that would copy the full text of the problem, the full text of the source paper, and a set of instructions to their clipboard. This provided a controlled mechanism for non-expert annotators to seek assistance with understanding unfamiliar terminology from an LLM assistant. The copied context included instructions asserting that the LLM should not attempt to render judgments for the annotator and only explain unfamiliar terminology and techniques documented in the problem. The following text-instructions were provided by the web annotation interface to the annotator.

G.1 Provenance Verification

Annotators were first asked to verify that itemized ground-truth contributions were substantiated by their source papers. The interface displayed the contribution narrative, the itemized contribution under review, extracted evidence spans, an evidence summary, and optional paper context.

Task description shown to annotators: The *contribution narrative* is a summary of methodological choices reported in a research paper. The *itemized contribution* below it identifies a specific part of the narrative. Your task is to verify that the source paper contains this specific contribution.

Read the contribution narrative first to understand the full contribution, then use the itemized contribution to identify the specific procedure, design choice, or technical approach under review. Consult the full paper via the arXiv link as needed. The *Export Context* button copies the full context of this annotation task to your clipboard for use with an AI assistant if you need help understanding the technical content.

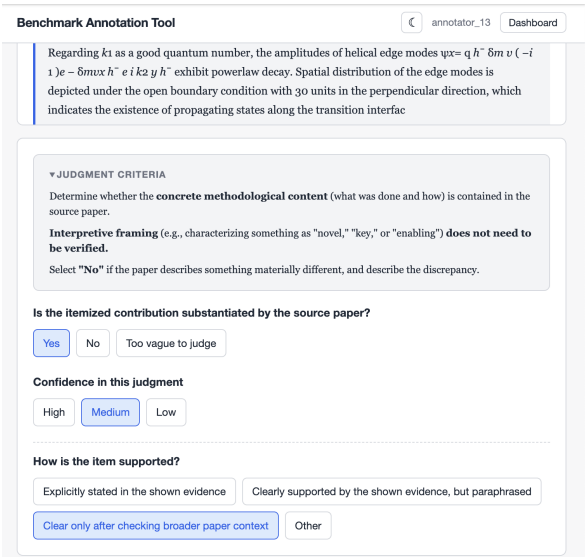

Figure 9: Screenshot of the annotation interface and options provided for the data provenance task.

G.2 Judge Calibration

Annotators were also asked to determine whether model-generated predictions contained the same methodological idea as an itemized ground-truth contribution. The interface displayed the research question, optional prior-art context, the contribution narrative, the itemized contribution under review, and the model prediction.

Task description shown to annotators: The *contribution narrative* is a summary of methodological choices reported in a research paper. The *itemized contribution* below it identifies a specific part of the narrative. The *prediction* is a hypothesis about what methodology the authors used. Your task is to judge whether the prediction contains the same core methodological idea as the itemized contribution.

Read the contribution narrative first to understand the full contribution, then use the itemized contribution to identify the specific procedure, design choice, or technical approach under review. Consult the full paper via the arXiv link as needed. The *Export Context* button copies the full context of this annotation task to your clipboard for use with an AI assistant if you need help understanding the technical content.

Judgment criteria shown to annotators: Determine whether the prediction describes the *same concrete methodology* as the itemized contribution (what was done and how). The methodology need only be *present somewhere* in the prediction. It does not need to be the prediction's central idea. Allow a small tolerance for minor, recoverable procedural differences as long as the core idea of "what to try" is preserved to a high degree.

When judging similarity, disregard any non-methodological details of the prediction, such as:

- Stated results or expected outcomes
- Motivations, physical intuitions, or explanatory framing
- Other details tangential to "what concrete steps should be attempted"

What matters is whether any part of the prediction contains the same procedure described in the itemized contribution, regardless of differences in framing or reasoning.

Select *Too vague to judge* if the prediction is consistent with the itemized contribution but lacks enough methodological detail to confirm it describes the same idea.

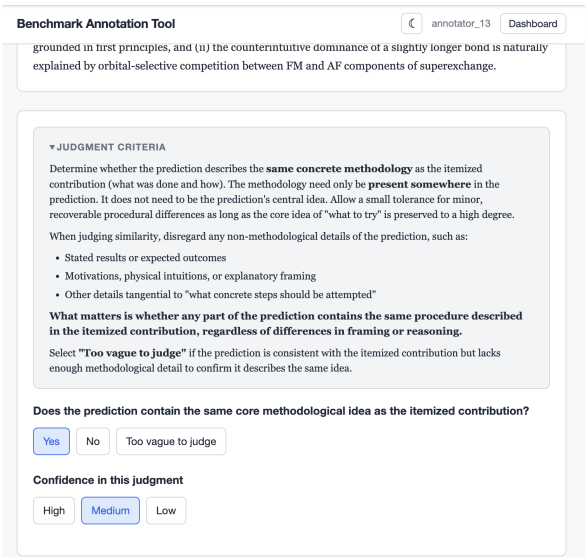

Figure 10: Screenshot of the annotation interface and options provided for the judge calibration task.

H Example Benchmark Problem

The following example is a representative benchmark problem generated from (Khan et al., 2025). It includes all problem stages for the selected paper, with the model-facing question, prior-art context, generated contribution narrative, and itemized ground-truth contributions for each stage. The problem has been modified from its original markdown format for LaTeX, but the text is otherwise presented in full.

H.1 Example Stage 0

Question prompt: Ultra-wide-bandgap semiconductors like beta-Ga2O3 are attractive for vertical power devices, but they require multi-micron-thick epitaxial drift layers that combine (i) fast growth, (ii) good surface morphology suitable for device processing, and (iii) high electron mobility at low n-type doping—three requirements that often trade off against each other (e.g., high growth rate or heteroepitaxy can introduce roughness/defects that suppress mobility).

Researchers have shown, experimentally, that low-pressure CVD growth can produce Si-doped beta-Ga2O3 films that are several microns thick with growth rates on the order of several to tens of microns/hr, while still achieving room-temperature Hall mobilities around 150–160 cm2/V-s at carrier densities near 10^17 cm^-3 on both native

beta-Ga2O3 and miscut sapphire (heteroepitaxy). Consider the challenges of simultaneously keeping growth fast and surfaces/crystallinity good enough for high mobility (especially in heteroepitaxy), and attempt to re-discover the key growth-process design choices that enabled this combination. Be as detailed as possible in your independent attempts. Your responses will be graded against the contributions in the actual paper.

Prior-art context: LPCVD growth of beta-Ga2O3 from a Ga source and oxidant under low pressure is an established epitaxy route that can deliver higher growth rates than many MOCVD/MBE processes. It is also standard that film morphology and crystalline quality depend strongly on mass-transport/supersaturation conditions (set by pressure, temperature, flows, and reactor geometry) and that heteroepitaxy on sapphire commonly uses off-axis/miscut substrates to encourage step-flow and suppress parasitic orientations.

Using XRD rocking curves and surface microscopy (SEM/AFM) to verify whether a chosen growth window is delivering single-phase, step-flow-like growth is also standard. The authors build within this established LPCVD + miscut-step-flow playbook, but they contribute a specific process-window/geometry optimization that simultaneously preserves fast growth and high mobility-quality microstructure for both homo- and hetero-epitaxy.

Contribution narrative: The core enabling step was not a new characterization method, but a growth-process optimization strategy aimed at the usual LPCVD bottleneck: when the Ga arrival rate (and local supersaturation) is pushed high enough to achieve multi-micron thickness quickly, beta-Ga2O3 surfaces tend to roughen and accumulate mobility-limiting extended defects; when the flux is too low, growth becomes impractically slow for thick drift layers.

The authors' key methodological move was to treat reactor geometry—specifically the Ga source-to-substrate spacing—as a primary control knob for Ga delivery (and thus growth rate vs. surface stability), and to deliberately tune that spacing within a fixed low-pressure/high-temperature LPCVD regime to land in a step/terrace growth window that remains smooth enough for high mobility while still producing multi-micron thickness on practical timescales.

In parallel, for the heteroepitaxial route they selected an off-axis (miscut) sapphire configuration that enforces step-flow along the miscut direction, reducing the likelihood of the kinds of surface breakdown and orientation mixing that would otherwise dominate transport. Together, these choices create a growth environment where high-rate deposition does not automatically translate into high roughness/defect density, enabling the reported high mobilities at low carrier density in both homo- and hetero-epitaxial films.

Ground-truth contribution items:

1. Elevated reactor-geometry control (Ga source-to-substrate distance) to a central process variable and tuned it to control Ga delivery/supersaturation, allowing the authors to reach a regime that simultaneously supports multi-micron thickness with high growth rate and preserves step/terrace morphology needed for high electron mobility.

2. For heteroepitaxial growth, adopted an off-axis (miscut) sapphire approach specifically to promote step-flow-driven epitaxy (rather than islanded/rotated growth), enabling hetero-films with sufficiently low extended-defect impact to retain high RT mobility at low doping.

H.2 Example Stage 1

Question prompt: Precisely tuning n-type doping in beta-Ga2O3 drift layers is crucial for vertical device design, but in thick, fast-grown halide/oxide CVD processes it is easy to end up with (i) non-uniform dopant incorporation vs depth, (ii) unintended impurity incorporation from the dopant chemistry, or (iii) a large gap between the chemically incorporated dopant density and the electrically active free-carrier density due to compensation/trapping.

Researchers have shown, experimentally, that Si-doped beta-Ga2O3 films grown by LPCVD can be tuned across roughly 10^17–10^19 cm^-3 by adjusting the dopant feed, while maintaining a nearly depth-uniform net donor profile in micron-thick layers and keeping common light-element/halogen contaminants below detection—yet still exhibiting a noticeable shortfall between incorporated Si and measured free carriers consistent with incomplete activation/compensation.

Consider the challenges of (a) calibrating a controllable dopant "knob" for thick LPCVD films, (b) proving the doping is uniform and not an artifact of a single measurement modality, and (c)

diagnosing whether discrepancies between incorporated dopant and carriers come from activation/compensation rather than contamination. Attempt to re-discover the key experimental/analysis strategy that enabled the authors to make these claims. Be as detailed as possible in your independent attempts. Your responses will be graded against the contributions in the actual paper.

Prior-art context: Using Si-containing precursors (including chlorosilanes) to dope oxide semiconductors is established, and varying dopant precursor flow to tune carrier concentration is standard practice. Hall measurements are the standard for extracting free-carrier density and mobility; capacitance-based profiling on Schottky diodes is a standard way to extract depth-resolved net ionized dopant density in a drift layer; SIMS is the standard for depth profiling of chemical dopant incorporation and for checking common impurities (e.g., C/H/Cl).

Interpreting discrepancies between SIMS (total dopant) and Hall/C–V (electrically active dopant) in terms of incomplete activation and/or compensating defects is a known concept in semiconductor doping studies. The authors' contribution is in the way they combined these modalities on the specific thick LPCVD drift-layer context to simultaneously (i) calibrate dopant control, (ii) demonstrate depth-uniformity, and (iii) bound impurity-driven explanations tightly enough to support an activation/compensation interpretation.

Contribution narrative: A single measurement (e.g., Hall alone) cannot distinguish "true dopant control" from confounding effects like depth gradients, parallel conduction, or impurity-driven compensation. The authors' enabling strategy was to build a triangulated calibration/validation loop that ties the dopant feed setting to three distinct but complementary quantities: (1) electrically measured free-carrier density (transport), (2) depth-resolved net ionized donor density in an actual diode-like drift-layer geometry (electrical profiling), and (3) the chemically incorporated Si concentration together with quantitative checks for candidate contaminants from the dopant chemistry and environment (chemical profiling).

Implementationally, this required designing the experiments around sufficiently thick drift layers to make depth profiling meaningful and fabricating a vertical Schottky structure specifically to extract a net-doping vs depth profile that can be compared across dopant-feed conditions. With that structure in place, they could show that increasing dopant feed monotonically increases electrical doping and that the net doping stays approximately flat through the film thickness (supporting uniform incorporation rather than a surface-skewed profile).

They then used quantitative SIMS on a representative high-mobility layer to (i) verify that Si is uniformly present through the epilayer, (ii) show that key impurity candidates (notably light elements and halogen residue from the dopant source) are below detection, and (iii) quantify that total incorporated Si can exceed the free-carrier density. With impurity incorporation strongly bounded by SIMS and uniformity cross-checked by independent electrical profiling, the remaining consistent explanation for the Si(carbon-free, halogen-free) vs carrier mismatch is incomplete activation/compensation (i.e., electrically inactive donors and/or acceptor-like traps), rather than uncontrolled contamination or gross non-uniform incorporation.

Ground-truth contribution items:

1. Constructed a multi-modal doping-calibration workflow for thick LPCVD beta-Ga2O3 that intentionally cross-links dopant-feed settings to (i) transport-extracted free carriers, (ii) depth-resolved electrical net-doping profiles from purpose-built Schottky drift-layer structures, and (iii) chemical depth profiles—so that monotonic tunability and depth-uniformity are supported by independent electrical evidence rather than Hall-only inference.

2. Used quantitative chemical depth profiling not just to measure Si incorporation but to explicitly bound impurity incorporation from the dopant chemistry (and thus rule out "contamination explains the carriers" scenarios), enabling a defensible interpretation of SIMS-vs-Hall discrepancies in terms of incomplete activation/compensation in the LPCVD-grown layers.

I Statement on the Use of AI Assistants

LLM assistants were used at various stages of this project to assist with code generation and drafting placeholder language for the final paper. All work included in the final submission is our own.