ASD: AUDITING SOURCE DEPENDENCE IN LLM JUDGES FOR SCIENTIFIC BENCHMARKING

Anonymous authors
Paper under double-blind review

ABSTRACT

LLM judges are widely used to evaluate AI agents on open-ended scientific benchmarks, where expert grading is prohibitively costly. These benchmarks are often constructed from published papers, which raises the question of whether a verdict reflects the quality of the answer or access to the source paper, since judges can often consult the same literature. We propose Auditing Source Dependence (ASD), a two-stage pipeline that measures source dependence as the change in a judge's verdict when the source content in its evidence is varied. ASD first builds a catalog of relevant and irrelevant papers for each task and splits the source paper into passages (paragraphs and topic blocks) labeled by rhetorical role. It then judges the same answer while swapping the source paper in and out of the judge's evidence. To locate source dependence within the paper, it further applies counterfactual edits to single passages of the source paper. Across BiomniBench, BixBench, and LABBench2, replacing a relevant paper with the source paper raises judge accuracy, and editing a single passage can reverse the verdict on an unchanged answer. Methods and results passages account for over 60% of the most influential passages, and edits to different passages shift verdicts in opposite directions, so small average effects can mask large passage-level effects. These findings motivate reporting the source sensitivity of LLM judges alongside agent performance.

1 INTRODUCTION

AI Agent assistants are now automating work in many industries, including software engineering, finance and recently fundamental sciences (Huang et al., 2026; Anthropic, 2026b; OpenAI, 2026), and there is an increase in agentic benchmarks (Qu et al., 2026; Chen et al., 2025; Bigeard et al., 2025; Starace et al., 2025; Jimenez et al., 2024) to measure their performance. Unlike software engineering, scientific benchmarks are often *open-ended* and scored via rubrics that cannot be programatically verified. It is often *prohibitively expensive* to employ human experts to assess agent outputs across these benchmarks, and as a result, the community has widely adopted LLM judges as automated verifiers. Moreover, LLM judges can be more reliable than human experts in some fields (Bedi et al., 2026; Arora et al., 2025). In biological benchmarks, for instance, LLM judges can be equipped with additional information (such as papers and rubrics) to provide a verdict on the agentic outputs, but questions remain as to how sensitive LLM judges are to additional information.

Recent works have characterized LLM judges' sensitivity to many factors orthogonal to agent response quality. For example, when two answers are given to an LLM judge, their ordering can flip the verdict even when the content is unchanged and this sensitivity grows as the two answers converge in quality (Shi et al., 2025); Zheng et al. (Zheng et al., 2023) showed that when the judge LLM is also the agent LLM, the judge favors its own outputs over comparable answers from other models; and Lee et al. (Lee et al., 2026) showed that when a supplied reference answer contradicts what the judge already believes, the judge follows its own belief rather than the reference.

A critical concern comes from how biology benchmarks are often constructed from source papers (Laurent et al., 2024; Qu et al., 2026; Mitchener et al., 2025). LLM judges are increasingly given retrieval and browsing tools so that they can find additional information to back-up their verdicts (Gou et al., 2025), which elevates them to *Agents-as-a-Judge*. As a result, if an LLM judge is shown source information, its assessment may reflect access to that source material rather than solely the quality of the agent's response. This raises a question previous works do not address: how sensitive

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.

is an LLM judge to source information and which parts of the source drive that sensitivity? We name this as **source-dependence**.

In this paper, we propose Auditing Source Dependence (ASD), a two-stage pipeline that quantifies source dependence in LLM judging of scientific benchmarks. The key observation is that we can choose exactly what evidence the judge sees: by fixing the question, candidate answer, and judging protocol and varying only the source content in that evidence, corresponding shifts in the verdict can be traced to source access rather than to answer quality. In Stage 1, ASD builds a catalog of papers relevant and irrelevant to each task's source paper and, motivated by DeepTrace (Venkit et al., 2026), decomposes the source paper into different types of passages (i.e. paragraphs and topic blocks) each labeled by rhetorical role (e.g. background, methods, results). In Stage 2, ASD swaps the source paper in and out of the judge's evidence, and then applies counterfactual edits to single passages of the source paper, tracing source dependence from the whole paper down to individual passages and, through their role labels, to sections of the paper. Our contributions are:

- **ASD, an interventional audit of source dependence in LLM judges** (Section 3), which estimates paper, passage, and role-level source effects from verdicts on controlled evidence sets that differ in one paper or one edited passage (REMOVE, FILL, or REWRITE).
- **Evidence that LLM judges on scientific benchmarks are source-dependent.** Across different frontier LLM judges on BiomniBench (Qu et al., 2026), BixBench (Mitchener et al., 2025), and LABBench2 (Laurent et al., 2026), verdict accuracy rises consistently from no evidence, to relevant papers, to the source paper, and swapping a single relevant paper for the source paper raises accuracy (Section 5.2).
- **A passage-level account of where this dependence arises.** Methods and results passages account for over 60% of the most influential paragraphs across benchmarks and edit types, yet edits to different passages push verdicts in opposite directions. This shows that small average effects can hide large passage-level effects and a single edited passage can reverse the verdict on an unchanged answer (Section 5.3).

Together, these experiments show that the same answer can pass or fail depending on (1) which paper is given and (2) which parts of those papers the judge is given, so scores from LLM judges on scientific benchmarks are not directly comparable across evaluation setups unless the judge's evidence is reported. We therefore recommend reporting source sensitivity, at both the paper and passage level, alongside agent performance.

2 BACKGROUND

We first formalize scientific benchmarks and LLM judging, and then define source dependence.

Scientific Benchmarks. A scientific benchmark task c consists of a question Q_c and a reference answer A_c^{ref}, both constructed from a source paper P_c^S. Benchmarks such as BiomniBench (Qu et al., 2026) are formulated by human experts and the reference answers are constructed differently across benchmarks. For example, BiomniBench uses an evaluation rubric, whereas BixBench and LABBench2 use curated answers. An AI agent then produces a candidate answer A_c^{cand}. For simplicity we consider only the final answer, though the execution trajectory could equally be included. The judge is also provided with *evidence set* E, obtained via Agent-as-a-Judge (Gou et al., 2025) or curated in advance. In this work, we restrict E to a set of papers, although, in general, it could be any structured or unstructured data. Given E, an LLM judge $f_c \equiv f(Q_c, A_c^{\text{cand}}, \cdot)$ produces

$$f_c(E) \equiv f(Q_c, A_c^{\text{cand}}, E), \qquad f_c \sim p_{f_c}, \tag{1}$$

where p_{f_c} denotes its verdict distribution. We consider $f_c(E) \in \{\text{pass}, \text{fail}, \text{abstain}\}$ and denote the indicator function by $\mathbb{1}[\cdot]$; the framework also extends to continuous rubric-based scores such as those in Qu et al. (2026).

Source Dependence. Suppose E_c^+ is a set of papers that contain passages from P_c^S, and E_c^- is a set that does not. For instance, $P_c \in E_c^+$ could be a paper that contains information from P_c^S (e.g., P_c citing P_c^S), or a modification of P_c^S (e.g., reorganizing the structure of the paper). In addition, $P_c \in E_c^-$ could be a paper relevant to P_c^S or a completely irrelevant paper. Then for judge f, the

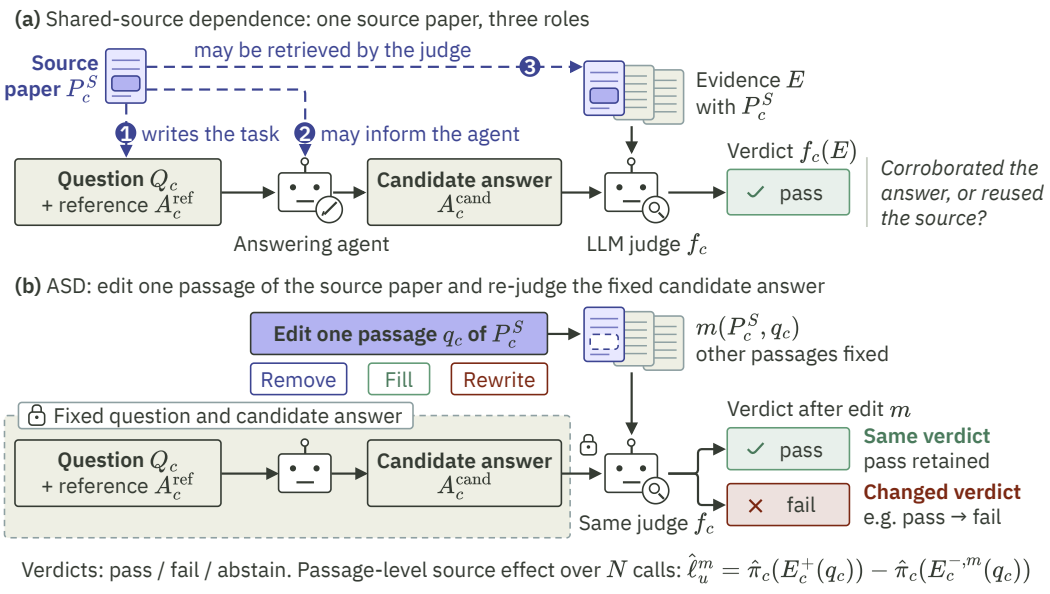


Figure 1: Shared-source dependence and the ASD intervention; symbols as defined in Sections 2–3. (a) One source paper writes the task, may inform the agent, and may be retrieved by the judge, so a pass leaves open whether the judge corroborated the answer or reused the source. (b) ASD holds question, answer, and judge fixed, edits one passage of the source paper (REMOVE, FILL, or REWRITE), and compares paired verdicts; the bottom line is the passage-level effect of Eq. (7).

source effect at the paper level is defined as

$$\ell_p(c, f) = \mathbb{E}_{\substack{E_c^+ \sim p_c^+ \\ f_c \sim p_{f_c}}}\Big[\mathbb{1}\big[f_c(E_c^+) = \text{pass}\big]\Big] - \mathbb{E}_{\substack{E_c^- \sim p_c^- \\ f_c \sim p_{f_c}}}\Big[\mathbb{1}\big[f_c(E_c^-) = \text{pass}\big]\Big], \tag{2}$$

where p_c^+ and p_c^- are distributions over sets of papers containing and not containing information from the source paper, respectively. Computing the source effect allows one to quantify source dependence for task c. Similarly, suppose $q_c \in P_c^S$ is a *passage*, for instance a smaller chunk of the paper. Let $E_c^+(q_c)$ denote a set of papers that contains q_c, and $E_c^-(q_c)$ a set that does not. We can then define the *source effect at the passage level* as

$$\ell_u(c, f, q_c) = \mathbb{E}_{\substack{E_c^+(q_c) \sim p_{q_c}^+ \\ f_c \sim p_{f_c}}}\Big[\mathbb{1}\big[f_c(E_c^+(q_c)) = \text{pass}\big]\Big] - \mathbb{E}_{\substack{E_c^-(q_c) \sim p_{q_c}^- \\ f_c \sim p_{f_c}}}\Big[\mathbb{1}\big[f_c(E_c^-(q_c)) = \text{pass}\big]\Big], \tag{3}$$

where $p_{q_c}^+$ and $p_{q_c}^-$ are distributions over sets of papers containing and not containing q_c.

3 ASD: AUDITING SOURCE DEPENDENCE IN LLM JUDGES

We propose Auditing Source Dependence (ASD), a two-stage pipeline that isolates the source effect by intervening on the judge's evidence set at the paper and passage level and comparing the resulting verdicts (Figures 1 and 2). Ideally, we would estimate the source effects in Eqs. (2) and (3) directly, but this is infeasible since $p^{\pm}$ are defined over a combinatorially intractable space of evidence sets. We therefore restrict $\tilde{p}^{\pm}$ to point masses on matched pairs of constructed evidence sets that differ in a single paper or a single passage of the source paper. Each pair measures the counterfactual sensitivity of the verdict to a judge with retrieval access seeing the source paper instead of any other paper (paper level), or to editing a single passage of the source paper while the rest is kept intact (passage level). In practice, the former indicates whether source access changes verdicts at all, and the latter identifies the parts of the source paper to which the judge is most sensitive. The resulting effects $\tilde{\ell}$ are thus source effects at these constructed evidence sets rather than averages over $p^{\pm}$. Stage 1 (Section 3.1) prepares the materials for these evidence sets, and Stage 2 (Section 3.2) constructs them and estimates paper-level, passage-level, and role-specific source effects.

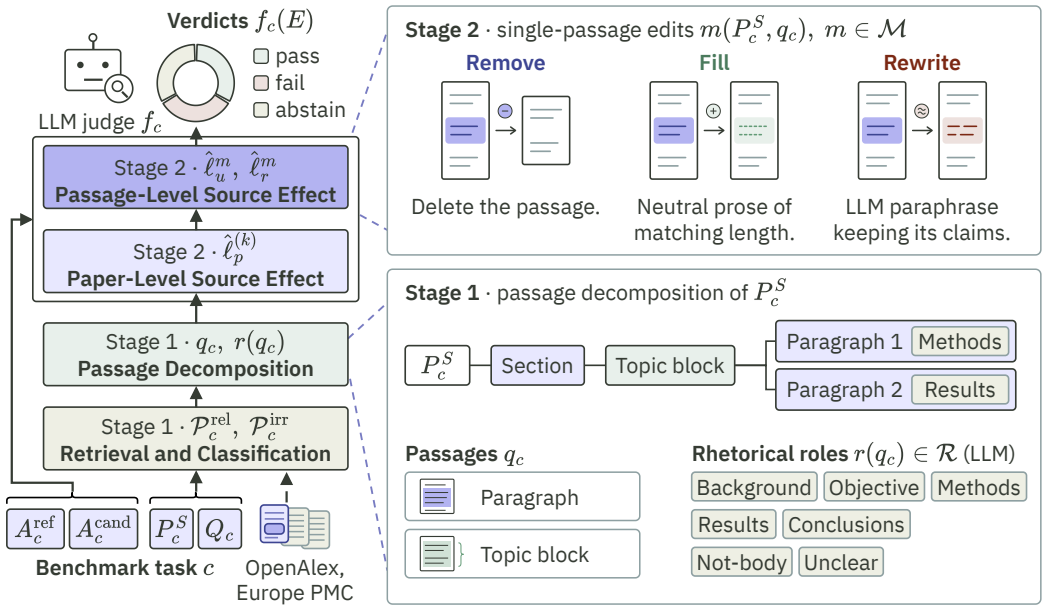


Figure 2: Overview of ASD. **Stage 1** retrieves the source paper and relevant and irrelevant papers for each task, decomposes the source paper into paragraphs and topic blocks, and labels each passage with a rhetorical role. **Stage 2** judges a candidate answer with and without the source paper and under single-passage edits, yielding paper-level ($\hat{\ell}_p^{(k)}$), passage-level ($\hat{\ell}_u^m$), and role-specific ($\hat{\ell}_r^m$) source effects.

3.1 ASD: Stage 1

Retrieval and Classification. For each task c, we build a paper catalog (details in Appendix A). We acquire the full text of the source paper P_c^S from benchmark metadata and retrieve candidate papers from OpenAlex and Europe PMC using queries derived from Q_c and P_c^S. An LLM then classifies the candidates by title and abstract into relevant papers $\mathcal{P}_c^{\text{rel}}$ and irrelevant papers $\mathcal{P}_c^{\text{irr}}$, and we call the result the evidence pool of that LLM (e.g., the DeepSeek pool for a DeepSeek LLM judge). We exclude P_c^S, its other versions (e.g., preprints), and papers citing it from $\mathcal{P}_c^{\text{rel}}$, since they may reproduce passages of the source paper and would therefore belong to E_c^+ instead of E_c^-.

Passage Decomposition. We decompose the body of P_c^S, excluding the abstract, figure captions, and tables, into passages at two resolutions: paragraphs of the parsed article, and topic blocks, i.e., contiguous spans of paragraphs within a section obtained with TextTiling (Hearst, 1997), with boundaries aligned to paragraph breaks. At either resolution, we write the passages as $\{q_{c,1}, \ldots, q_{c,K_c}\}$ and omit the resolution from the notation. An LLM then labels each passage q_c with a rhetorical role $r(q_c) \in \mathcal{R} = \{\text{Background, Objective, Methods, Results, Conclusions, Not-body, Unclear}\}$, used to aggregate passage-level effects in Stage 2.

3.2 ASD: Stage 2

For each task, the question Q_c, candidate answer A_c^{cand}, and judge f (model and prompt) are fixed, and the evidence set E is assigned by ASD. We write $Y_c(E) = \mathbb{1}[f_c(E) = \text{pass}]$ for the verdict, where $f_c \sim p_{f_c}$ captures the residual stochasticity of the judge's decoding at temperature 0, and $\pi_c(E) = \mathbb{E}_{f_c \sim p_{f_c}}[Y_c(E)]$ for the pass probability. Since E is assigned, $\pi_c(E) = \mathbb{E}_{f_c \sim p_{f_c}}[Y_c \mid do(E)]$ (Pearl, 2009), and the source effects for task c and judge f are interventional contrasts

$$\hat{\ell} = \mathbb{E}_{E_c^+ \sim \tilde{p}^+}\left[\pi_c(E_c^+)\right] - \mathbb{E}_{E_c^- \sim \tilde{p}^-}\left[\pi_c(E_c^-)\right]. \tag{4}$$

Taking each $\tilde{p}^{\pm}$ as a point mass (defined later), $\tilde{\ell}$ reduces to $\pi_c(E_c^+) - \pi_c(E_c^-)$. Since both evidence sets are evaluated for the same task with everything else fixed, and judge calls are independent, both terms are observed directly, so $\tilde{\ell}$ can be estimated by comparing their verdicts without adjusting for

confounders. We run the judge once per evidence set at temperature 0 and use the recorded verdict $\hat{\pi}_c(E) = Y_c(E)$ as the estimate of $\pi_c(E)$. When the judge exposes token probabilities, we also use the probability it assigns to pass, $\hat{\pi}_c^{\text{prob}}(E) = s_c(E)$, as a confidence-weighted alternative.

Source Effect at the Paper Level. From the catalog, we construct four evidence sets for each task:

$$E_c^{3+} = \{P_{c,1}^{\text{rel}}, P_c^S, P_c^{\text{irr}}\}, \quad E_c^{3-} = \{P_{c,1}^{\text{rel}}, P_{c,2}^{\text{rel}}, P_c^{\text{irr}}\}, \quad E_c^{1+} = \{P_c^S\}, \quad E_c^{1-} = \{P_{c,1}^{\text{rel}}\}, \tag{5}$$

where $P_{c,1}^{\text{rel}}, P_{c,2}^{\text{rel}}$ are the two top-ranked relevant papers and P_c^{irr} is an irrelevant paper, selected after source and citation filtering. For $k \in \{1, 3\}$, P_c^S replaces $P_{c,2}^{\text{rel}}$ ($k = 3$) or $P_{c,1}^{\text{rel}}$ ($k = 1$) in the same prompt position, so the number of papers and all other evidence are held fixed. We also evaluate the judge without evidence, $E_c^0 = \emptyset$, as a baseline. Taking $\tilde{p}_c^{k\pm}$ as point masses on $E_c^{k\pm}$, we estimate the paper-level source effect $\tilde{\ell}_p^{(k)}$ as

$$\hat{\ell}_p^{(k)}(c, f) = \hat{\pi}_c(E_c^{k+}) - \hat{\pi}_c(E_c^{k-}), \quad k \in \{1, 3\}, \tag{6}$$

i.e., how the verdict changes when the judge *sees the source paper instead of a comparable relevant paper* among k papers, taking values 1, 0, or -1 when the verdict changes from fail to pass, is unchanged, or changes from pass to fail. Over a task set $\mathcal{C}$, we report the average $\hat{\ell}_p^{(k)}(f) = |\mathcal{C}|^{-1} \sum_{c \in \mathcal{C}} \hat{\ell}_p^{(k)}(c, f)$ with 95% percentile confidence intervals from 30K bootstrap resamples of tasks.

Source Effect at the Passage Level. For a passage q_c and an edit $m \in \mathcal{M} = \{\text{REMOVE}, \text{FILL}, \text{REWRITE}\}$, let $m(P_c^S, q_c)$ denote the source paper with q_c edited by m and all other passages unchanged. REMOVE deletes the passage; FILL replaces it with neutral prose of matching length, separating the loss of content from the loss of length; and REWRITE replaces it with a paraphrase from a separate model that keeps claims, quantities, and qualifications, serving as a wording-only control (edit validation in Appendix D). Some examples of passage edits are in Appendix H. Taking $\tilde{p}_{q_c}^{+}$ and $\tilde{p}_{q_c}^{-,m}$ as point masses on $E_c^+(q_c) = \{P_c^S\}$ and $E_c^{-,m}(q_c) = \{m(P_c^S, q_c)\}$, we estimate the passage-level source effect $\tilde{\ell}_u^m$ as

$$\hat{\ell}_u^m(c, f, q_c) = \hat{\pi}_c\big(E_c^+(q_c)\big) - \hat{\pi}_c\big(E_c^{-,m}(q_c)\big). \tag{7}$$

Since $E_c^+(q_c) = E_c^{1+}$, its verdict is recorded once per task and shared across passages, edits, and the paper level. Since all other passages are held fixed, $\hat{\ell}_u^m(c, f, q_c)$ is a *leave-one-out effect* of the content and length of q_c under REMOVE, of its content alone under FILL, and of its wording under REWRITE. We rank passages within each benchmark by $\hat{\ell}_u^{\text{REMOVE}}(c, f, q_c)$ to identify the most influential ones, and obtain role-specific source effects by averaging over all passages with role r in a benchmark with task set $\mathcal{C}$:

$$\hat{\ell}_r^m(f) = \frac{1}{|\mathcal{Q}_r|} \sum_{(c,q_c) \in \mathcal{Q}_r} \hat{\ell}_u^m(c, f, q_c), \qquad \mathcal{Q}_r = \{(c, q_c) : c \in \mathcal{C},\ q_c \in P_c^S,\ r(q_c) = r\}. \tag{8}$$

Here, $\hat{\ell}_r^m(f)$ pools passages across tasks, so tasks with more passages of role r receive more weight, and comparisons between roles are descriptive, since roles are not assigned by intervention. Confidence intervals come from the same bootstrap over tasks, keeping all passages of each resampled task.

4 Related Work

Scientific Agents. Agentic AI assistants such as ChatGPT and Claude have accelerated research, from coding to literature review. Recently, there is an interest to build agents specialized for scientific tasks, such as Coscientist (Boiko et al., 2023) for chemical research, Biomni (Huang et al., 2026) for biomedical analysis, The AI Scientist (Lu et al., 2026) for end-to-end AI research, and Claude Science (Anthropic, 2026b) for general scientific analysis. As a result, there is also an increase in benchmarks for evaluating scientific agents, including BiomniBench (Qu et al., 2026), BioMysteryBench (Anthropic, 2026a), GeneBench (Li & Ho, 2026), BixBench (Mitchener et al., 2025), and LABBench2 (Laurent et al., 2026), several of which use LLM judges. For instance, BiomniBench builds each task from a journal papers such as Nature and scores agent trajectories against expert-designed rubrics with an LLM judge. We apply ASD to BiomniBench, BixBench, and LABBench2 to study source dependence in their judging stage.

LLM Judges. LLM judges are sensitive to factors unrelated to the quality of agent answers. Judges favor answers generated by themselves (Zheng et al., 2023), which Panickssery et al. (2024) linked to their ability to recognize their own generations. When two candidate answers are compared, their ordering alone can flip the verdict (Wang et al., 2024), and this position bias grows as the answers become closer in quality (Shi et al., 2025). Moreover, when a reference answer contradicts the judge's own belief, the judge follows its belief (Lee et al., 2026). More recently, LLM judges have been equipped with tools to retrieve and browse evidence, elevating them to Agents-as-a-Judge (Zhuge et al., 2025; Gou et al., 2025), which also exposes them to the source material used to construct the benchmark task. ASD addresses this gap by estimating how much verdicts depend on the source material and which parts of it drive this dependence.

Context Decomposition and Attribution. Previous works also decompose text into fine-grained passages to measure how language models rely on evidence. FActScore (Min et al., 2023) evaluates long-form generations at the level of atomic facts, and DeepTRACE (Venkit et al., 2026) decomposes the answers of deep research agents into statements to audit how they cite evidence. Context attribution methods such as ContextCite (Cohen-Wang et al., 2024) identify which parts of the context cause a generated statement. More broadly, interventional approaches estimate causal effects via Pearl's do-calculus (Pearl, 1995; 2009), either on model internals, such as neurons and attention heads (Vig et al., 2020) or hidden states (Meng et al., 2022), or on model inputs, by editing a single concept in the input text (Abraham et al., 2022). ASD intervenes on model inputs, replacing a relevant paper with the source paper or editing passages of the source paper, and measures the resulting change in verdicts.

5 Experiments

We evaluate ASD at two hierarchical levels: the paper level and the passage level. The paper-level comparisons assess whether use of the source paper alters judgments of a fixed answer beyond the influence of independent, task-relevant paper. The passage-level comparisons investigate how materials of source paper impact judgments and identify which types of passages are more sensitive.

5.1 Experimental Setup

Benchmarks and judgments. We keep only tasks with an explicit source paper: 21 from BiomniBench, 26 from BixBench, and 309 from LABBench2. For each task, GPT-5.6-Luna (GPT), GLM-5.3-flash (GLM), and DeepSeek-V4-flash (DeepSeek) each run Stage 1 evidence retrieval, and each retrieval is judged in Stage 2 by the other two models to avoid self-evaluation, giving 6 judge-retrieval configurations per benchmark. We use Qwen-3.8 27B-INT4 to generate the REWRITE edits. it neither retrieves nor judges, so no judge evaluates its own rewrite. Each judge returns *pass*, *fail*, or *abstain*. The correctness of its judgment depends on whether it correctly assesses A^{cand}, whose correctness is determined by comparison with A^{ref}. A verdict flip between the original and edited evidence counts as a gain if it goes from incorrect to correct, and as a loss if the reverse. Over a task set $\mathcal{C}$, we report the average effects with pointwise 95% percentile confidence intervals from 30K bootstrap resamples of tasks.

Evidence conditions. Across both experiments, we hold the question, candidate answer, judge, and judging protocol fixed, varying only the evidence presented to the judge. At the paper level, we construct four evidence conditions, E_c^{1+}, E_c^{1-}, E_c^{3+}, and E_c^{3-}, which form two matched pairs as defined in Section C, together with a no-evidence reference condition E_c^0. At the passage level, we take E_c^{3+} as the intact condition and, for each source passage q_c, construct a corresponding modified condition by applying text modification m only to q_c in P_c^S, while keeping all other source passages and the remaining papers in E_c^{3+} fixed.

5.2 Paper-level: Source Dependence under Controlled Evidence Conditions

Source paper improves recorded-verdict accuracy. Figure 3 compares verdict accuracy for fixed candidate answers A^{cand} under five controlled evidence conditions. Across benchmarks, judge models, and both paragraph and topic-block decompositions, the accuracy profiles share a consistent

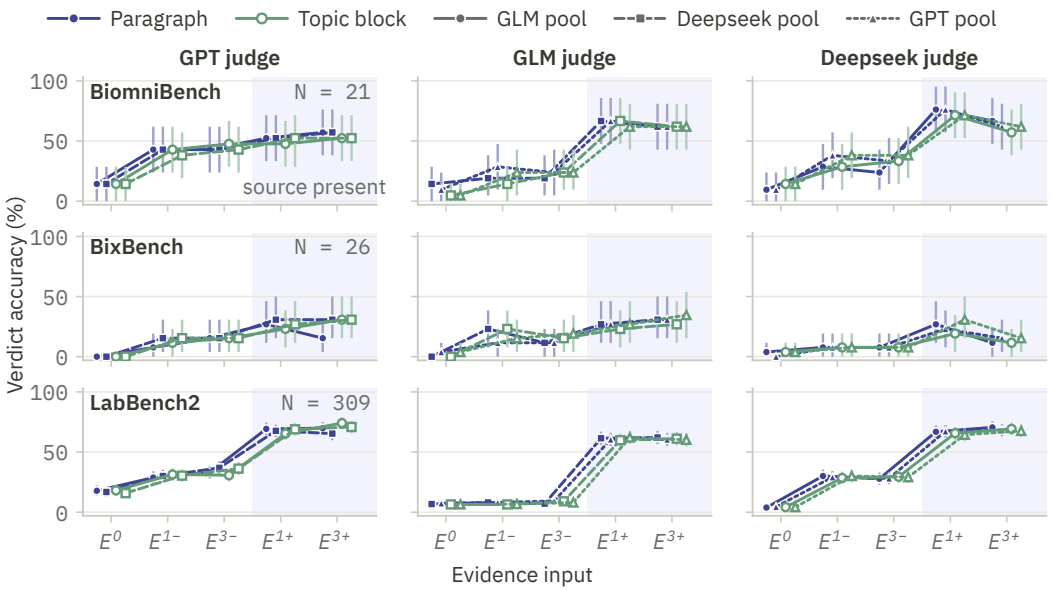


Figure 3: **Verdict accuracy under five evidence inputs for paragraph and topic blocks.** Columns correspond to judges and rows to benchmarks; benchmark sample sizes N are shown in each row. Colors distinguish paragraph and topic blocks, and line styles distinguish the GLM, DeepSeek, and GPT retrieval. The shaded region marks evidence inputs that contain the source paper. Each point shows question-averaged accuracy of the recorded verdicts, with pointwise 95% percentile intervals.

qualitative pattern:

$$E^0 < \{E^{1-}, E^{3-}\} < \{E^{1+}, E^{3+}\},$$

where this ordering summarizes the broad separation among the no evidence, source absent, and source present conditions. Specifically, source present conditions generally achieve higher accuracy than their matched source absent counterparts, with E^{1+} outperforming E^{1-} and E^{3+} outperforming E^{3-}. This pattern is consistently observed across across all three benchmarks and judge models, supporting the conclusion that access to the source paper facilitates correct verdicts beyond what can be achieved with relevant papers alone.

Adding papers can reduce accuracy when the source paper is present. When the source paper is already present, adding further non-source papers can reduce recorded-verdict accuracy. This pattern is evident from the comparison between the source-only condition E^{1+} and the full condition E^{3+}, where adding relevant non-source papers leads to lower accuracy in several configurations. For example, on BiomniBench with the DeepSeek judge and GLM evidence retrieval, accuracy decreases from 76.19% under E^{1+} to 66.67% under E^{3+} for paragraph decomposition, and from 71.43% to 57.14% for topic-block decomposition. These results demonstrate that more evidence is not necessarily beneficial once the source paper is available, as additional non-source context can reduce verdict accuracy relative to using the source paper alone.

Increasing evidence reduces abstention. Table 1 reports abstention and grouped verdict transitions for both paragraph and topic-block decompositions. Under the no-evidence condition E^0, abstention is high across all benchmarks. As additional evidence becomes available, abstained runs increasingly transition to substantive verdicts rather than remaining unresolved. This pattern is already evident when moving from E^0 to source-free evidence ($\{E^{1-}, E^{3-}\}$) and becomes more pronounced with source-containing evidence ($\{E^{1+}, E^{3+}\}$). Across these transitions, abstained runs are more likely to resolve to correct than to wrong verdicts, while transitions from correct or wrong verdicts back to abstention remain comparatively infrequent. The same pattern holds for both paragraph and topic-block decompositions. Overall, the results indicate that increasing the available evidence reduces abstention and makes the judges more likely to commit to a verdict, consistent with greater behavioral decision confidence.

Table 1: Abstention and verdict transitions across evidence conditions for paragraph and topic-block decompositions. Values before/after the slash correspond to paragraph/topic-block decomposition.

Benchmark	Comparison	$A \to C$	$A \to W$	$C/W \to A$
BioMNIBench	E^0 (87.3/86.5%) $\to \{E^{1-}, E^{3-}\}$	23.0/21.8	7.9/6.0	2.8/0.8
	$\{E^{1-}, E^{3-}\} \to \{E^{1+}, E^{3+}\}$	31.3/31.0	19.0/21.2	2.0/3.0
BixBench	E^0 (98.7/97.4%) $\to \{E^{1-}, E^{3-}\}$	10.6/11.5	3.5/3.5	0.0/0.6
	$\{E^{1-}, E^{3-}\} \to \{E^{1+}, E^{3+}\}$	14.4/14.9	8.5/7.9	3.4/3.8
LABBench2	E^0 (82.3/82.6%) $\to \{E^{1-}, E^{3-}\}$	15.5/15.7	9.5/9.3	5.7/5.8
	$\{E^{1-}, E^{3-}\} \to \{E^{1+}, E^{3+}\}$	36.7/36.7	6.8/6.3	1.8/2.0

Note. Percentages after E^0 report the abstention rate under the no-evidence condition. A, C, and W denote abstain, correct, and wrong verdicts. Transition columns report the percentage of instances whose verdict changes between the evidence conditions shown in the Comparison columns: $A \to C$ and $A \to W$ denote transitions from abstention to a correct or wrong verdict, and $C/W \to A$ denotes transitions from either a correct or wrong verdict to abstention. Values before/after the slash correspond to paragraph/topic-block decomposition.

5.3 Passage-level: Sensitivity Analysis based on Interventions

Methods and results consistently emerge as the top two contributors. Figure 4 shows that Methods and Results account for the largest shares of the absolute intervention effect across the three benchmarks. At the paragraph decomposition, this pattern is particularly consistent across intervention types: Methods and Results jointly account for approximately 60–65% of the intervention effect, while the remaining sections contribute substantially smaller shares. The same overall pattern persists at the topic block decomposition, where Methods and Results generally remain the two largest contributors, although their relative shares vary more across benchmarks and intervention types. In particular, the topic block distributions assign larger shares to other sections, including Background, Objective, Conclusions, and Not body, than their paragraph decomposition counterparts. Overall, these results identify Methods and Results as the primary sources of intervention effects across both granularities, with the top two pattern being substantially more consistent for paragraphs.

Rewriting produces smaller accuracy changes across benchmarks. Figure 5 shows a consistent aggregate pattern across benchmarks and intervention granularities: Rewrite produces smaller changes in question-level accuracy than Remove or Fill. Averaged across the six judge-retrieval configurations, Rewrite has the smallest absolute accuracy change on every benchmark at both the paragraph and topic-block levels. At the paragraph level, its average changes range from 7.1 to 8.2

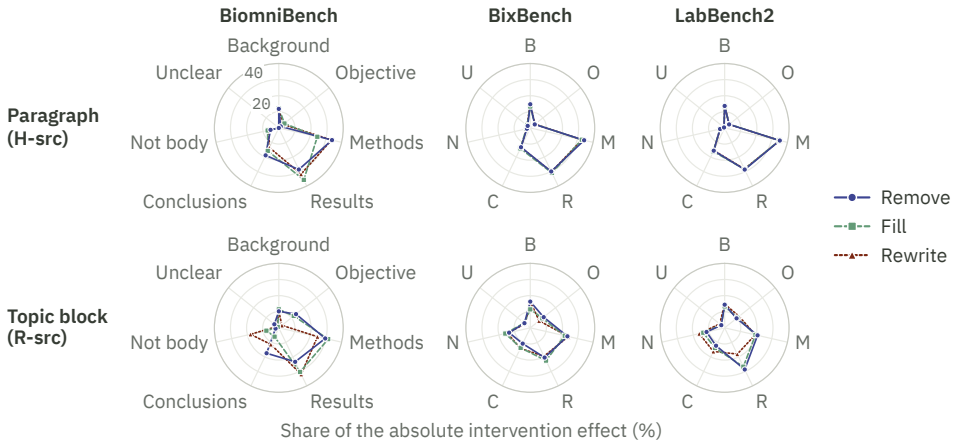


Figure 4: **Share of the absolute intervention effect across section types.** Radar plots are shown for BiomniBench, BixBench, and LABBench2 (columns), with interventions applied at the paragraph (top) or topic-block (bottom) level.

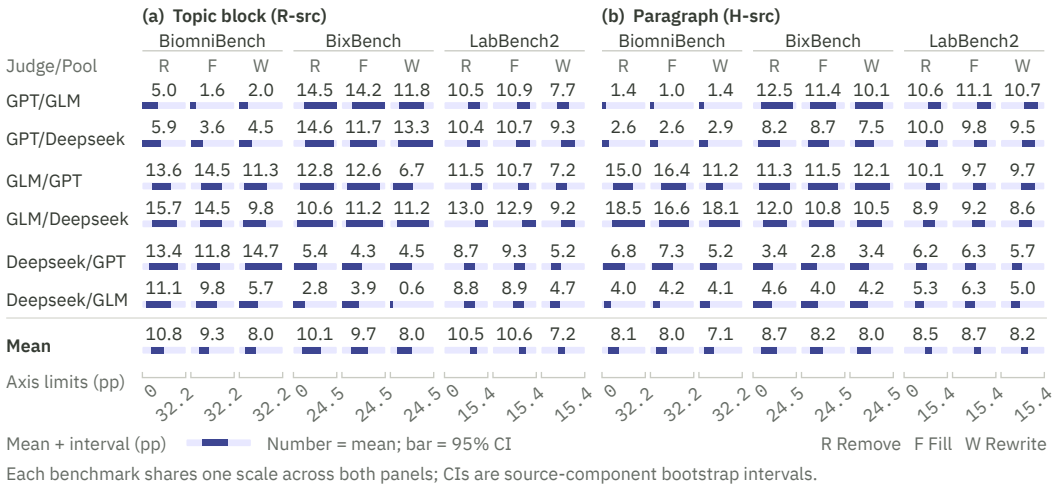


Figure 5: **Effects of source interventions on passage level accuracy.** Mean absolute changes in passage level accuracy (percentage points) for each judge-retrieval under topic block decomposition (Table 10) and paragraph decomposition (Table 10). Numbers report mean changes, and horizontal bars show pointwise 95% confidence intervals. Each benchmark uses the same horizontal scale across both panels.

percentage points, compared with 8.0 to 8.7 percentage points for REMOVE and FILL. The difference is more pronounced for topic-block interventions, where REWRITE yields average changes of 7.2 to 8.0 percentage points, while REMOVE and FILL range from 9.3 to 10.8 percentage points. This pattern indicates that accuracy is less affected, on average, when source content is rewritten than when it is removed or filled.

6 CONCLUSION AND LIMITATIONS

Conclusion. Across three benchmarks and judge models, we find consistent source dependence in LLM judging. Source-containing evidence generally yields higher verdict accuracy than matched source-absent evidence, while adding non-source papers can reduce accuracy when the source is already available. Increasing evidence also reduces abstention, with abstained verdicts more often resolving to correct than incorrect judgments. At the passage level, Methods and Results consistently account for the largest shares of intervention effects across both granularities, while REWRITE produces smaller accuracy changes than REMOVE or FILL. Together, these results provide systematic evidence of source dependence in LLM judges and characterize how it manifests across evidence conditions, rhetorical roles, and source interventions.

Limitations. ASD estimates source effects under experimentally controlled evidence. Because benchmark tasks are constructed from their source papers, source identity is inherently coupled with informational advantage: the source is precisely the paper expected to contain the information most directly aligned with the task. Our estimates therefore capture their joint effect rather than attempting to separate two factors that are structurally linked by benchmark construction. However, comparisons are made against relevant papers retrieved for the same task, which are the evidence a judge with literature access would realistically consult, so the paper-level effect measures the practical advantage of seeing the source paper. At the passage level, ASD intervenes on one passage at a time while keeping the rest of the source paper intact, which yields a clean interventional effect for each passage but one that is conditional on the remaining passages rather than an independent contribution. Our rhetorical-role analysis is restricted to source papers because extending exhaustive interventions to relevant papers would be prohibitively costly, but relevant papers are still assessed in the paper-level comparisons. Finally, the observed patterns are consistent across the evaluated benchmarks and judges but remain to be tested in other scientific domains and judging systems, to which ASD can be applied directly.

Reproducibility Statement

All methods and experimental details are available in this manuscript. Upon acceptance, we will release the code and any permitted, licensed data.

Ethics Statement

A rejection by ASD shows that a judge's verdicts depend on source information. On its own, it does not establish that a model provider or candidate agent used sources improperly. Such a claim requires independent evidence about which inputs were permitted and whether the grading was valid. We collected all data in accordance with the licenses of each benchmark and source paper, and we document any access restrictions. We do not redistribute restricted full text. Where redistribution is not permitted, we release source records and text spans as identifiers and content hashes, so that others can retrieve and verify the same material.

AI Use Statement

Claude and ChatGPT assisted with polishing the manusript, literature search, implementing the code and running experiments. In particular, the manuscript is mostly written manually, with AI used to polish the writing. Literature search was conducted with the assistance of Deep Research agents, but all cited references and subsequent claims are reviewed and checked for their correctness. Code implementation involved using Claude Code and ChatGPT Codex to implement precise design patterns planned by the authors and then carefully reviewed to ensure the results are correct. Experiments were mostly run by hand, but in some cases we used Claude Code and ChatGPT Codex to help track and visualize. The authors are responsible for the final results, methods, claims, references, and disclosures.

References

Eldar David Abraham, Karel D'Oosterlinck, Amir Feder, Yair Ori Gat, Atticus Geiger, Christopher Potts, Roi Reichart, and Zhengxuan Wu. CEBaB: Estimating the causal effects of real-world concepts on NLP model behavior. In *Advances in Neural Information Processing Systems*, volume 35, pp. 17582–17596. Curran Associates, Inc., 2022. doi: 10.52202/068431-1278. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/701ec28790b29a5bc33832b7bdc4c3b6-Abstract-Conference.html.

Anthropic. Evaluating claude's bioinformatics research capabilities with BioMysteryBench, apr 2026a. URL https://www.anthropic.com/research/Evaluating-Claude-For-Bioinformatics-With-BioMysteryBench. Accessed: 2026-09-25.

Anthropic. Claude science, an AI workbench for scientists, is now available, jun 2026b. URL https://www.anthropic.com/news/claude-science-ai-workbench. Accessed: 2026-09-25.

Rahul K. Arora, Jason Wei, Rebecca Soskin Hicks, Preston Bowman, Joaquin Quiñonero Candela, Foivos Tsimpourlas, Michael Sharman, Meghan Shah, Andrea Vallone, Alex Beutel, Johannes Heidecke, and Karan Singhal. HealthBench: Evaluating large language models towards improved human health. *arXiv preprint arXiv:2505.08775*, 2025. URL https://arxiv.org/abs/2505.08775.

Suhana Bedi, Hejie Cui, Miguel Fuentes, Alyssa Unell, Michael Wornow, Juan M. Banda, Nikesh Kotecha, Timothy Keyes, Yifan Mai, Mert Oez, Hao Qiu, Shrey Jain, Leonardo Schettini, Mehr Kashyap, Jason Alan Fries, Akshay Swaminathan, Philip Chung, Fateme Nateghi Haredasht, Ivan Lopez, Asad Aali, Gabriel Tse, Ashwin Nayak, Shivam Vedak, Sneha S. Jain, Birju Patel, Oluseyi Fayanju, Shreya Shah, Ethan Goh, Dong-han Yao, Brian Soetikno, Eduardo Reis, Sergios Gatidis, Vasu Divi, Robson Capasso, Rachna Saralkar, Chia-Chun Chiang, Jenelle Jindal, Tho Pham, Faraz Ghoddusi, Steven Lin, Albert S. Chiou, Hyo Jung Hong, Mohana Roy, Michael F. Gensheimer, Hinesh Patel, Kevin Schulman, Dev Dash, Danton Char, Lance Downing, Francois

Grolleau, Kameron Black, Bethel Mieso, Aydin Zahedivash, Wen-wai Yim, Harshita Sharma, Tony Lee, Hannah Kirsch, Jennifer Lee, Nerissa Ambers, Carlene Lugtu, Aditya Sharma, Bilal Mawji, Alex Alekseyev, Vicky Zhou, Vikas Kakkar, Jarrod Helzer, Anurang Revri, Yair Bannett, Roxana Daneshjou, Jonathan Chen, Emily Alsentzer, Keith Morse, Nirmal Ravi, Nima Aghaeepour, Vanessa Kennedy, Akshay Chaudhari, Thomas Wang, Sanmi Koyejo, Matthew P. Lungren, Eric Horvitz, Percy Liang, Michael A. Pfeffer, and Nigam H. Shah. Holistic evaluation of large language models for medical tasks with MedHELM. *Nature Medicine*, 32(3):943–951, 2026. doi: 10.1038/s41591-025-04151-2. URL https://www.nature.com/articles/s41591-025-04151-2.

Antoine Bigeard, Langston Nashold, Rayan Krishnan, and Shirley Wu. Finance agent benchmark: Benchmarking LLMs on real-world financial research tasks. *arXiv preprint arXiv:2508.00828*, 2025. URL https://arxiv.org/abs/2508.00828.

Daniil A. Boiko, Robert MacKnight, Ben Kline, and Gabe Gomes. Autonomous chemical research with large language models. *Nature*, 624(7992):570–578, 2023. doi: 10.1038/s41586-023-06792-0. URL https://www.nature.com/articles/s41586-023-06792-0.

Ziru Chen, Shijie Chen, Yuting Ning, Qianheng Zhang, Boshi Wang, Botao Yu, Yifei Li, Zeyi Liao, Chen Wei, Zitong Lu, Vishal Dey, Mingyi Xue, Frazier N Baker, Benjamin Burns, Daniel Adu-Ampratwum, Xuhui Huang, Xia Ning, Song Gao, Yu Su, and Huan Sun. ScienceAgentBench: Toward rigorous assessment of language agents for data-driven scientific discovery. In *International Conference on Learning Representations*, volume 2025, pp. 96934–96990, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/f12b4df26344f3be803c06b555252efe-Abstract-Conference.html.

Benjamin Cohen-Wang, Harshay Shah, Kristian Georgiev, and Aleksander Madry. ContextCite: Attributing model generation to context. In *Advances in Neural Information Processing Systems*, volume 37, pp. 95764–95807. Curran Associates, Inc., 2024. doi: 10.52202/079017-3035. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/adbea136219b64db96a9941e4249a857-Abstract-Conference.html.

Boyu Gou, Zanming Huang, Yuting Ning, Yu Gu, Michael Lin, Weijian Qi, Andrei Kopanev, Botao Yu, Bernal Jiménez Gutiérrez, Yiheng Shu, Chan Hee Song, Jiaman Wu, Shijie Chen, Hanane Nour Moussa, Tianshu Zhang, Jian Xie, Yifei Li, Tianci Xue, Zeyi Liao, Kai Zhang, Boyuan Zheng, Zhaowei Cai, Viktor Rozgic, Morteza Ziyadi, Huan Sun, and Yu Su. Mind2Web 2: Evaluating agentic search with Agent-as-a-Judge. In *Advances in Neural Information Processing Systems*, volume 38, pp. 192393–192451. Curran Associates, Inc., 2025. doi: 10.52202/085713-5778. URL https://proceedings.nips.cc/paper_files/paper/2025/hash/fdcec9f5b99aa4fc8f4fb8487802d737-Abstract-Datasets_and_Benchmarks_Track.html.

Marti A. Hearst. TextTiling: Segmenting text into multi-paragraph subtopic passages. *Computational Linguistics*, 23(1):33–64, 1997. URL https://aclanthology.org/J97-1003/.

Kexin Huang, Serena Zhang, Hanchen Wang, Yuanhao Qu, Yingzhou Lu, Ryan Li, Yusuf Roohani, Lin Qiu, Shiyi Cao, Gavin Li, Junze Zhang, Di Yin, Rick Wierenga, Deniz Kavi, Sherry Liu, Tianwei She, Shruti Marwaha, Jennefer N. Carter, Xin Zhou, Matthew T. Wheeler, Jonathan A. Bernstein, Mengdi Wang, Peng He, Jingtian Zhou, Michael P. Snyder, Le Cong, Aviv Regev, and Jure Leskovec. Autonomous biomedical research with an artificial intelligence agent. *Science*, 393 (6813):eadz4351, 2026. doi: 10.1126/science.adz4351. URL https://www.science.org/doi/abs/10.1126/science.adz4351.

Carlos E. Jimenez, John Yang, Alexander Wettig, Shunyu Yao, Kexin Pei, Ofir Press, and Karthik Narasimhan. SWE-bench: Can language models resolve real-world GitHub issues? In *International Conference on Learning Representations*, volume 2024, pp. 54107–54157, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/edac78c3e300629acfe6cbe9ca88fb84-Abstract-Conference.html.

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. LAB-bench:

Measuring capabilities of language models for biology research. *arXiv preprint arXiv:2407.10362*, 2024. URL https://arxiv.org/abs/2407.10362.

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. LABBench2: An improved benchmark for AI systems performing biology research. *arXiv preprint arXiv:2604.09554*, 2026. URL https://arxiv.org/abs/2604.09554.

Dongryeol Lee, Yerin Hwang, Taegwan Kang, Minwoo Lee, Younhyung Chae, and Kyomin Jung. Judging against the reference: Uncovering knowledge-driven failures in LLM-judges on qa evaluation. *arXiv preprint arXiv:2601.07506*, 2026. URL https://arxiv.org/abs/2601.07506.

Jeremy Li and Andrew Ho. GeneBench: Assessing AI agents for multi-stage inference problems in genomics and quantitative biology. *bioRxiv*, 2026. doi: 10.64898/2026.04.22.720113. URL https://www.biorxiv.org/content/10.64898/2026.04.22.720113v1.

Chris Lu, Cong Lu, Robert Tjarko Lange, Yutaro Yamada, Shengran Hu, Jakob Foerster, David Ha, and Jeff Clune. Towards end-to-end automation of AI research. *Nature*, 651:914–919, 2026. doi: 10.1038/s41586-026-10265-5. URL https://www.nature.com/articles/s41586-026-10265-5.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in GPT. In *Advances in Neural Information Processing Systems*, volume 35, pp. 17359–17372. Curran Associates, Inc., 2022. doi: 10.52202/068431-1262. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/6f1d43d5a82a37e89b0665b33bf3a182-Abstract-Conference.html.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Wei Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. FActScore: Fine-grained atomic evaluation of factual precision in long form text generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 12076–12100. Association for Computational Linguistics, 2023. doi: 10.18653/v1/2023.emnlp-main.741. URL https://aclanthology.org/2023.emnlp-main.741/.

Ludovico Mitchener, Jon M Laurent, Alex Andonian, Benjamin Tenmann, Siddharth Narayanan, Geemi P Wellawatte, Andrew White, Lorenzo Sani, and Samuel G Rodriques. BixBench: a comprehensive benchmark for LLM-based agents in computational biology. *arXiv preprint arXiv:2503.00096*, 2025. URL https://arxiv.org/abs/2503.00096.

OpenAI. Introducing GPT-rosalind for life sciences research, apr 2026. URL https://openai.com/index/introducing-gpt-rosalind/. Accessed: 2026-09-25.

Arjun Panickssery, Samuel R Bowman, and Shi Feng. LLM evaluators recognize and favor their own generations. In *Advances in Neural Information Processing Systems*, volume 37, pp. 68772–68802. Curran Associates, Inc., 2024. doi: 10.52202/079017-2197. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/7f1f0218e45f5414c79c0679633e47bc-Abstract-Conference.html.

Judea Pearl. Causal diagrams for empirical research. *Biometrika*, 82(4):669–688, 1995. doi: 10.1093/biomet/82.4.669. URL https://academic.oup.com/biomet/article-abstract/82/4/669/251647.

Judea Pearl. *Causality: Models, Reasoning, and Inference*. Cambridge University Press, 2 edition, 2009. doi: 10.1017/CBO9780511803161. URL https://doi.org/10.1017/CBO9780511803161.

Yuanhao Qu, Yingzhou Lu, Xinming Tu, Serena Zhang, Tianwei She, Alexander Glenn Shaw, Jou-Ho Shih, Bingqing Zhao, Minjie Shen, Haochen Yang, Jielin Yan, Rongchuan Zhang, Xinze Wu, Tingting Li, Le Cong, Xiaobo Hu, Yuan Jiang, Jiayun Dong, Tao Peng, Jure Leskovec, and Kexin Huang. BiomniBench: Process-level evaluation of LLM agents for real-world biomedical research. *bioRxiv*, 2026. doi: 10.64898/2026.05.12.724604. URL https://www.biorxiv.org/content/10.64898/2026.05.12.724604v1.

Lin Shi, Chiyu Ma, Wenhua Liang, Xingjian Diao, Weicheng Ma, and Soroush Vosoughi. Judging the judges: A systematic study of position bias in LLM-as-a-Judge. In *Proceedings of the 14th International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics*, pp. 292–314. The Asian Federation of Natural Language Processing and The Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.ijcnlp-long.18. URL https://aclanthology.org/2025.ijcnlp-long.18/.

Giulio Starace, Oliver Jaffe, Dane Sherburn, James Aung, Jun Shern Chan, Leon Maksin, Rachel Dias, Evan Mays, Benjamin Kinsella, Wyatt Thompson, Johannes Heidecke, Amelia Glaese, and Tejal Patwardhan. PaperBench: Evaluating AI's ability to replicate AI research. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 56843–56873. PMLR, 2025. URL https://proceedings.mlr.press/v267/starace25a.html.

Pranav Narayanan Venkit, Philippe Laban, Yilun Zhou, Kung-Hsiang Huang, Yixin Mao, and Chien-Sheng Wu. DeepTRACE: Auditing deep research AI systems for tracking reliability across citations and evidence. In *International Conference on Learning Representations*, volume 2026, pp. 106041–106062, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/ad08767706825033b99122332293033d-Abstract-Conference.html.

Jesse Vig, Sebastian Gehrmann, Yonatan Belinkov, Sharon Qian, Daniel Nevo, Yaron Singer, and Stuart Shieber. Investigating gender bias in language models using causal mediation analysis. In *Advances in Neural Information Processing Systems*, volume 33, pp. 12388–12401. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/hash/92650b2e92217715fe312e6fa7b90d82-Abstract.html.

Peiyi Wang, Lei Li, Liang Chen, Zefan Cai, Dawei Zhu, Binghuai Lin, Yunbo Cao, Lingpeng Kong, Qi Liu, Tianyu Liu, and Zhifang Sui. Large language models are not fair evaluators. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 9440–9450. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.acl-long.511. URL https://aclanthology.org/2024.acl-long.511/.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, Hao Zhang, Joseph Gonzalez, and Ion Stoica. Judging LLM-as-a-Judge with MT-Bench and chatbot arena. In *Advances in Neural Information Processing Systems*, volume 36, pp. 46595–46623. Curran Associates, Inc., 2023. doi: 10.52202/075280-2020. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/91f18a1287b398d378ef22505bf41832-Abstract-Datasets_and_Benchmarks.html.

Mingchen Zhuge, Changsheng Zhao, Dylan R. Ashley, Wenyi Wang, Dmitrii Khizbullin, Yunyang Xiong, Zechun Liu, Ernie Chang, Raghuraman Krishnamoorthi, Yuandong Tian, Yangyang Shi, Vikas Chandra, and Jürgen Schmidhuber. Agent-as-a-Judge: Evaluate agents with agents. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 80569–80611. PMLR, 2025. URL https://proceedings.mlr.press/v267/zhuge25a.html.

APPENDIX

A Retrieval and Classification 14
B Passage Decomposition 15
C Paper-Level Source Effect 16
D Passage-Level Source Effect 16
E Experimental Setup 17
F Collection Protocol and Execution Records 20
G Model Prompts 20
H Intervention Examples 26
I How Much Does Source Information Change Judging? 28
J Where Does Source Dependence Occur? 34
K How Do Model and Representation Choices Affect the Audit? 36
L How Does Source Dependence Relate to Grading Agreement? 39

A RETRIEVAL AND CLASSIFICATION

Records and storage. The pipeline stores the records of questions, the source papers and the candidate answers, and each model receives only the records that its step needs. The retrieval model is the LLM whose Classification result Section 3.1 calls its evidence pool; the GLM pool is the pool that GLM built. When the retrieval model screens candidate papers, it receives the question with the titles, abstracts and other metadata of the candidates; when it labels rhetorical roles, it receives paper text alone. The judge receives the question, the candidate answer and the evidence input. The reference answer is stored in a separate Stage 1 record. Each paper's full text is stored with its identifier, its retrieval provenance, a content hash and the section and paragraph offsets of the source format, so every edited passage can be traced to its position in the paper.

Stage 1: evidence and provenance. Stage 1 builds the evidence pools, the passages and their role labels in four phases (Figure 6), of which only Phases 2 and 4 call a model. Phase 1 (Benchmark Parsing) reads each benchmark release, keeps the question text and its identifiers, attaches the registered source paper, and writes the reference answers to a separate record. Phase 2 (Paper Retrieval) searches OpenAlex with the question text alone and has the retrieval model classify the returned records as relevant or irrelevant. An identifier match then compares every candidate with the source paper and with the list of works that cite it, which is fetched once per source paper from Europe PMC. Phase 3 (Paper Processing) fetches the full text of each retained paper from PubMed Central and records the papers that cannot be fetched, matched or parsed as failures. Phase 4 (Data Representation) divides each paper into sections, paragraphs and sentences with character offsets and has the retrieval model label the rhetorical role of every sentence. The paragraphs below give the settings of Phases 2 and 3, and Appendix B gives those of Phase 4.

OpenAlex search. For each question, Phase 2 (Paper Retrieval) queries the OpenAlex works catalog with the question text, so in the terms of Section 3.1 the query is derived from Q_c alone. A record is kept only when it has an abstract and a persistent identifier, resolves to a PubMed Central identifier and has its full text in PubMed Central. When the semantic search returns fewer than 50 such records, the search is widened with a fixed sequence of keyword queries built from the question until 50 are found or the sequence ends. The query sent to OpenAlex is stored with the call.

Classification of relevant and irrelevant papers. The retrieval model reads the titles and abstracts of the candidate papers and labels a subset of them as relevant or irrelevant. The OpenAlex query is built from the question text alone, and the results may contain the source paper as a candidate. An identifier match then checks every candidate against the source paper and the list of works that cite it. Each question is searched once, and each retrieval model screens the results once, with failed calls retried.

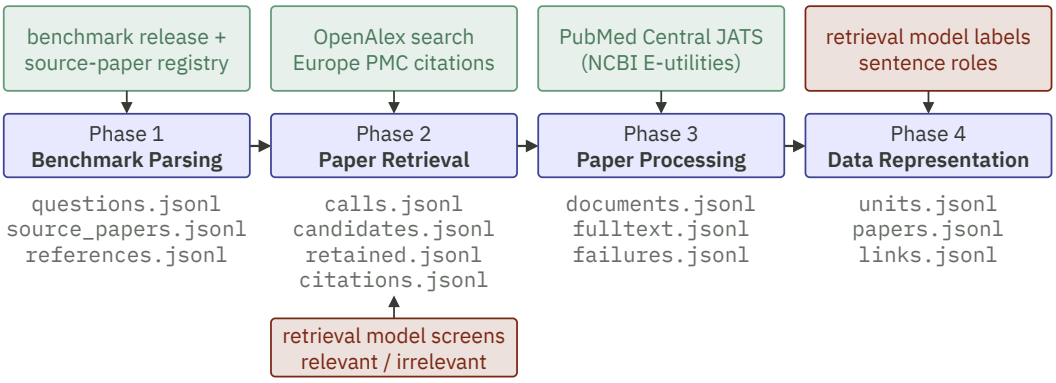


Figure 6: Stage 1 of the pipeline. Each phase reads the output of the previous one and writes its own records with a manifest; record names appear below the boxes.

Classification. Each retrieval model screens the OpenAlex records of a question with one message, which holds the Classification instruction, the question and the titles, abstracts and other metadata of the records. The model returns a list of labeled records, each with its identifier, a label and a rationale of one sentence. Titles, DOIs and PMIDs always come from the OpenAlex record. The Classification prompt (Appendix G.1) defines a relevant record as one that, among the records shown, bears most closely on the question. The model evaluates this from title and abstract, and a relevant paper needs to be related to the question. An irrelevant record comes from a neighboring area of the same field and does not bear on the question. The prompt asks for at least six relevant and at least three irrelevant records when the list allows, up to ten of each.

Full text. Phase 3 (Paper Processing) fetches the JATS XML of each retained paper from PubMed Central through the NCBI E-utilities and verifies it against the candidate's identifiers and title. It converts each document into sections and paragraphs and keeps only documents with at least 1,000 characters of body text. A local PDF is used only when an operator supplies one. A paper that cannot be fetched, does not match or fails to parse is recorded as a failure with its status. A pool has full text for a question when it supplies the full text of the source paper and of two relevant papers and one irrelevant paper that are independent of the source paper. The question enters Stage 2 only when all three pools have full text for it.

Yield. Stage 1 starts from the 24 BiomniBench, 84 BixBench and 346 LABBench2 questions whose source paper can be identified. The GPT, Deepseek and GLM pools have full text for 24/83/323, 24/84/327 and 24/82/325 of them. The 21, 81 and 309 questions common to the four pools enter Stage 2. This selection uses two properties of a question, an identifiable source paper and full text in all three pools, and was fixed in Stage 1 before the first judge call of the reported runs. Before the pools were intersected, each retrieval model labeled the rhetorical roles of the papers it selected for its own questions, one source paper, two relevant papers and one irrelevant paper per question. The counts are 79, 216 and 1,136 papers for GPT, 79, 219 and 1,127 for Deepseek and 80, 216 and 1,148 for GLM. Table 3 counts the same four papers per question for the questions in the audit.

B Passage Decomposition

Paragraphs and topic blocks. Stage 1 stores the full text of each paper as a hierarchy of sections, paragraphs and sentences, and the experiments use passages at two resolutions built on it. At the paragraph resolution, a passage is one paragraph of the source paper. A topic block collects all sentences of the source paper that carry the same rhetorical role, in document order. The retrieval model assigns each sentence one of seven rhetorical roles, the set $\mathcal{R}$ of Section 3.1, which holds background, objective, methods, results, conclusions, not-body and unclear. It labels one paragraph at a time and sees one neighboring sentence on each side as context; the prompt is reproduced in Appendix G.2.

Passages and roles. Phase 4 (Data Representation) writes two views of the same immutable text. The hierarchy view divides each paper into sections, paragraphs and sentences with character offsets, and at the paragraph resolution a passage is one paragraph of body text as the JATS source delimits it. The view leaves out the abstract, as Section 3.1 states, together with figures and their captions, tables, the reference list and supplementary material. The second view labels every paragraph or topic block with one of the seven rhetorical roles. The retrieval model receives one paragraph at a time with the sentence before and after it as context, and when its reply for a paragraph cannot be parsed, the sentences of that paragraph are unlabeled.

C PAPER-LEVEL SOURCE EFFECT

Stage 2: judging. Stage 2 makes every judge call, in Phases 1 and 2 (Figure 7). Phase 1 (Normal Judging) receives the questions, candidate answers and the Stage 1 evidence pools. It builds the evidence inputs E^{3+}, E^{3-}, E^{1+} and E^{1-} from four papers of each question and judges each input once with its evidence and once without evidence (E^0). Phase 2 (Counterfactual Judging) edits every passage of the source paper in E^{3+} in turn by REMOVE, FILL or REWRITE and records the edit with its validation. It judges the edited input with the same prompt and model as the intact input and pairs the result with the intact record of the same question, judge and pool from Phase 1.

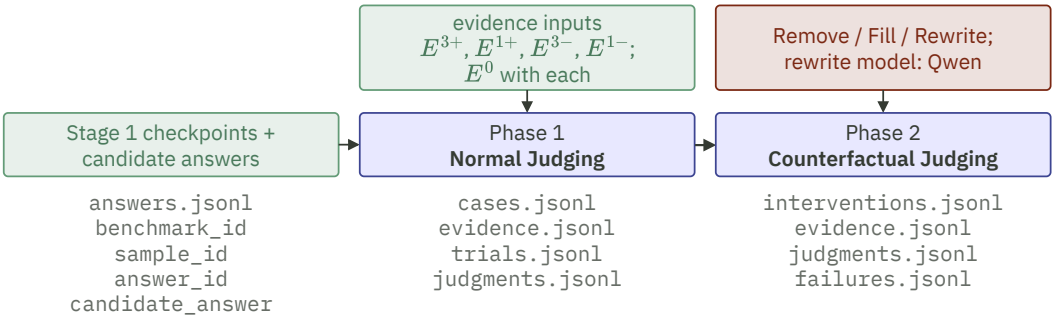


Figure 7: Stage 2 of the pipeline. Phase 1 takes four papers per question, builds the evidence inputs from them and judges each input once with evidence and once closed-book. Phase 2 edits one source passage at a time and judges the edited input with the same prompt.

Evidence inputs. For every question, Stage 2 takes four papers from the retained Stage 1 list in rank order. These are the source paper, the two independent relevant papers ranked highest and the independent irrelevant paper ranked highest. From these papers Stage 2 builds four evidence inputs, the evidence sets E_c^{3+}, E_c^{3-}, E_c^{1+} and E_c^{1-} of Section 3.2, written without the index c below. The input E^{3+} holds the source paper, the relevant paper ranked first and the irrelevant paper, and E^{3-} holds the two relevant papers and the irrelevant paper. The input E^{1+} holds the source paper alone, and E^{1-} holds the relevant paper ranked first, which also appears in E^{3+}. The two matched contrasts, $E^{3+} - E^{3-}$ and $E^{1+} - E^{1-}$, keep the number of papers and replace one relevant paper with the source paper. The input E^{3+} holds the source paper in place of $P_{c,2}^{\mathrm{rel}}$, the relevant paper ranked second, and E^{1+} holds it in place of $P_{c,1}^{\mathrm{rel}}$, the relevant paper ranked first. The step from E^{3-} to E^{1+}, which Appendix I also reports, replaces all three papers with the source paper, so the number of papers and the papers themselves change together. Each input is also judged closed-book (E^0), with the question and the candidate answer alone.

D PASSAGE-LEVEL SOURCE EFFECT

Interventions. Phase 2 of Stage 2 edits one passage at a time. Every passage of the source paper is edited once per operation inside E^{3+}. In the experiments, the sets $E_c^{+}(q_c) = \{P_c^S\}$ and $E_c^{-,m}(q_c) = \{m(P_c^S, q_c)\}$ of Section 3.2 are therefore E_c^{3+} and E_c^{3+} with P_c^S replaced by $m(P_c^S, q_c)$. The intact input of every passage-level comparison is E^{3+}, so every passage effect is conditional on the other two papers of E^{3+}. REMOVE and FILL follow fixed rules and call no model.

REMOVE deletes the passage. FILL replaces it with neutral prose from a fixed collection of 18 texts, 5 to 88 word runs long, which were assembled independently of the questions. The filler is built from the texts closest in length to the passage, starting from the text closest in length. REWRITE is the only edit that calls a model. It sends the passage alone to Qwen (`qwen3.8-27b`, served locally; Table 2), which is not one of the three judges, and the rewrite prompt (Appendix G.4) asks for the same meaning in different words at close to the original length.

Each edited input is stored as a new record next to the intact one, judged with the same prompt, model and decoding as its intact counterpart from Phase 1 and paired with the intact record of the same question, judge and pool. At the paragraph resolution on LABBench2, the GPT judge with the GLM pool ran the 309 questions in two batches of 200 and 109 and produced 14,164 interventions under each of REMOVE and FILL and 13,926 under REWRITE, after 238 rewrites were rejected. In this configuration the provider refused 10, 45 and 30 judge calls under REMOVE, FILL and REWRITE. On two LABBench2 questions (`340e6616` and `9764525a`) the stored Phase 2 records also contain edits of the passages of one relevant paper, 638 of the 97,360 LABBench2 REMOVE records. That paper is the source paper of another question, and passages are marked as editable per paper. All tables and figures include these records.

Edit validation. Every edit is validated before the edited input is judged. All text outside the target must survive in order, and for REMOVE and FILL no fragment of the target may remain in an overlapping passage. FILL text is built from a fixed collection of neutral texts chosen by their length relative to the passage, measured in word runs (maximal strings of letters and digits), and its construction is independent of the question, the answer and the verdicts. A REWRITE output is rejected when the call to Qwen fails or when the reply is empty, identical to the original, cut off at the token limit, or shorter than 0.5 or longer than 2.0 times the original in characters. Failed edits are logged and excluded from scoring.

Extraction records. An extraction record fixes the passages, their target spans and the replacement text of every edit before any judge call on an edited input, so every reported effect is conditional on these stored edits. When replacement text is drawn more than once, question-level effects average over the stored versions. In the runs reported in the main text, each question has one extraction record ($S = 1$). In each judge and pool configuration every passage has one replacement text per operation, and a passage whose rewrite is rejected drops out of the REWRITE results of that configuration.

Aggregation across questions. Questions connected through shared edited papers form a source cluster $g \in \{1, \dots, G\}$ with question set $\mathcal{C}_g$ (exact rule in Appendix E); the 21, 26 and 309 questions of BiomniBench, BixBench and LABBench2 form 10, 9 and 270 clusters. Let τ_c^m be the mean of the passage-level effect $\hat{\ell}_u^m(c, f, q_c)$ of Equation (7) over the passages of question c under operation $m \in \mathcal{M}$. The main text and Appendix I average τ_c^m with equal weight per question. Appendices J, K and L give equal weight to every source cluster,

$$\tau^m = \frac{1}{G} \sum_{g=1}^{G} \frac{1}{|\mathcal{C}_g|} \sum_{c \in \mathcal{C}_g} \tau_c^m, \tag{9}$$

and average the mean absolute effect in the same way. The flip rates by role label in Table 12 and the role-level mean $\hat{\ell}_r^m$ of Equation (8) pool passages with equal weight.

The intervals of all passage-level results resample source clusters, whether the average weights questions or clusters equally. Unless a figure or table states otherwise, each of 30,000 draws resamples the clusters of a benchmark with replacement, keeps every question of a drawn cluster with all its edited passages and recomputes the summary. The intervals are the pointwise 95% percentile intervals of these draws, which the note of Figure 5 calls source-component bootstrap intervals. The paper-level accuracy profiles of Figure 3 and the paired differences between their inputs resample questions (Appendix E).

E EXPERIMENTAL SETUP

This section gives the settings of the experiments in Section 5 and of the appendix analyses. It covers the questions and their source papers, the candidate answers, the judges and evidence pools, the

interval methods and the compute. The appendix writes GPT, GLM, DeepSeek and Qwen for the four models, and Table 2 gives their API identifiers. The main text writes GPT-5.6-Luna, GLM-5.3-flash and DeepSeek-V4-flash in its prose and the short names in its figure captions. The two resolutions are paragraphs and topic blocks, and the three operations are REMOVE, FILL and REWRITE.

Table 2: Models of Stage 2 and their settings. Budget is the limit on completion tokens. Every model runs at temperature 0 and seed 0, and the judges return one JSON object.

Role	Model identifier	Endpoint	Budget (tokens)	Reasoning
GPT judge	`gpt-5.6-luna`	OpenAI	512	off
GLM judge	`glm-5.3-flash`	Zhipu	2,048	low
Deepseek judge	`deepseek-flash`	Deepseek	512	thinking disabled
Qwen rewriter	`qwen3.8-27b-INT4`	local vLLM	16,384	thinking disabled

Notes. Log-probabilities are requested from every judge (top 5 for GPT, top 20 for GLM and Deepseek), but only the GPT judge returns usable ones. The main text writes the Deepseek judge as `deepseek-v4-flash`; the API identifier is `deepseek-flash`. Qwen rewrites passages in Stage 2 and built a fourth Stage 1 pool, which serves only to select the Stage 2 questions (Appendix A).

Benchmarks and questions. The audit covers 21, 26 and 309 questions of BiomniBench, BixBench and LABBench2, each with an explicit source paper, which fall into 10, 9 and 270 source clusters. A source cluster collects the questions connected through a shared edited paper. Each question has one source paper in Stage 2, and LABBench2 has 272 distinct source papers. Two LABBench2 clusters each join two source papers, because on two questions the edited passages also include a relevant paper that is the source paper of another question (see Interventions below). These counts result from two filters and, on BixBench, a cap of three questions per source paper. The first filter keeps the questions whose source paper can be identified, 24 of 50 in BiomniBench, 84 of 205 in BixBench and 346 of 1,912 in LABBench2. The second keeps the 21, 81 and 309 of them for which all four pools have full text (Appendix A). On BixBench the cap keeps, for each source paper, the first three questions in the order of a hash of the question identifier with seed 0, which leaves 26 of the 81 questions, three for eight source papers and two for the ninth. The cap was applied before the first BixBench judge call of the reported runs, and both BixBench answer sets described below cover the same 26 questions. Phase 1 of Stage 1 reads each release with a declared loader, keeps the question text and the question identifier, and attaches the source paper from a curated registry for BiomniBench and BixBench and from the release's own source field for LABBench2. The reference answer or rubric is stored in a separate Stage 1 record, and the judge receives the question, the candidate answer and the evidence input.

Candidate answers. Each question has one frozen candidate answer, supplied outside Stage 1. The judge receives the answer text alone. BiomniBench answers are the final reports of a Terminus-2 agent running Qwen (median 3,735 characters). BixBench answers come from a Claude Opus 5 agent and are short final answers (median 4 characters); the Slim answer set of Appendix L.1 uses zero-shot GPT-4o answers to the open-ended questions (median 181 characters). LABBench2 answers come from Gemini 3 Pro (preview) for 308 questions and from Claude Opus 4.6 for one question, both called directly (median 958 characters; the longest answer is 2,000 characters).

Each candidate answer carries a correctness label, which is read only in the analysis of the Stage 2 verdicts. The labels mark 9 answers as correct and 12 as incorrect on BiomniBench, 12 and 14 on BixBench, and 64 and 245 on LABBench2. Section 5.1 describes this label as the comparison of A_c^{cand} with A_c^{ref}. On BixBench each label compares the candidate answer with the reference answer under the verifier that BixBench declares for the question: an exact string match for 13 questions, a numeric range for 5 and a semantic comparison for 8. Every BixBench label is stored with its verifier and a written reason, and on the 18 questions with a string or range verifier the rule of the verifier reproduces the label. The LABBench2 answers are reports that the LABBench2 release published between January and March 2026, each with the grade of the release's own evaluator. On BiomniBench the reference answer of each question is the benchmark's grading rubric. Table 3 summarizes the questions, answers, evidence pools and passages of each benchmark.

Table 3: Data statistics of the three benchmarks as used in the audit. The rows give the questions with an explicit source paper, the candidate answers and their correctness labels, the papers in the evidence inputs built from the evidence pool of each model, and the passages of the source papers edited in Stage 2.

Statistic	BiomniBench	BixBench	LABBench2
Questions and answers			
Questions	21	26	309
Source papers (clusters)	10 (10)	9 (9)	272 (270)
Candidates correct / incorrect	9 / 12	12 / 14	64 / 245
Answer generator	Terminus-2 agent	Claude Opus 5 agent	Gemini 3 Pro
Stage 1 evidence pools: papers (relevant / irrelevant)			
GPT pool	71 (41 / 20)	81 (47 / 26)	1,049 (522 / 265)
GLM pool	73 (42 / 21)	82 (47 / 26)	1,068 (525 / 279)
Deepseek pool	72 (41 / 21)	80 (47 / 26)	1,040 (519 / 255)
Paragraphs			
passages	502	386	12,420
median per question	43	34	43
median length (word runs)	147	112	126
Topic blocks			
passages	67	61	1,853
median per question	7	7	7
median length (word runs)	799	432	695

Judges and evidence pools. Three models serve both as judges and as builders of the evidence pools used in Stage 2. Each retrieval model screens the retrieved papers for every question and labels the rhetorical role of every paragraph or topic block in the papers it selected. Each judge reads the two pools built by the other two models, which gives six configurations of judge and pool per benchmark, the judge-retrieval configurations of Section 5.1. Table 2 lists the decoding settings. All judges run at temperature 0 with seed 0 and return one JSON object. GPT runs with reasoning off and Deepseek with thinking disabled; the GLM API refuses to disable reasoning, so GLM runs at low reasoning effort with a completion budget of 2,048 tokens. Log-probabilities are requested from every judge, and only GPT returns usable ones: GLM returns none, and Deepseek returns either none or values of exactly 0 and 1. The probability is read at the verdict token that opens the reply (Appendix G.3).

Intervals. Each figure and table states its interval method. Table 10 uses percentile intervals from 30,000 bootstrap draws that resample source clusters. The paper-level accuracy profiles and the paired differences between their inputs use 30,000 bootstrap draws that resample questions. The module of an edited passage is a role label derived from the Stage 1 sentence roles, namely the majority role of its sentences for a paragraph and the role that defined it for a topic block (Appendix J). The flip rate by role label is the flip rate of the edited passages with one module; its 95% percentile interval uses the same 30,000 source-cluster draws as Table 10 and recomputes the pooled proportion in each draw. Phase 3 of Stage 2 writes estimates for each run with 4,000 bootstrap draws clustered by question and a report that resamples groups of questions sharing a source paper; these outputs served as diagnostics.

Compute and serving. GPT, GLM and Deepseek are called through their providers' APIs. Qwen is served with vLLM on an institutional GPU host from a 4-bit quantized checkpoint. Stage 1 searches OpenAlex, looks up citations in Europe PMC and fetches full text from PubMed Central through the NCBI E-utilities; Appendix A gives the retrieval and Classification settings.

F COLLECTION PROTOCOL AND EXECUTION RECORDS

This section documents the frozen configurations and the execution records. It lists the settings that were fixed before the outcomes of the runs were inspected and reports the number, tokens and cost of the calls that were made.

Fixed configuration. Table 4 lists the settings that were fixed before the outcomes of the reported runs were inspected. Its target policy is the rule of Appendix D that every passage of the source paper is edited once per operation.

Table 4: Configuration fields fixed before the outcomes of the reported runs were inspected.

Factor	Fixed settings
Candidate answers	Generating agent and configuration, saved trajectory, and answer text
Retrieval and Classification	Catalog query, Classification prompt and model, citation-based exclusion, paper quotas, and tie-breaking
Representation	Full-text conversion version, offset convention, segmentation and TextTiling settings, annotation prompt and model
Interventions	Passage type, target policy and eligibility, filler collection and length rule, rewrite prompt, model, and acceptance checks
Judges	Model identifiers and revisions where available, access dates, judging prompt, decoding, reasoning mode, context and output budgets
Repeats	S, judge calls per input, retry policy, and handling of failed or malformed calls

Model calls, tokens and spending. Table 5 counts every model call with recorded usage: the calls of the main runs, those of the other recorded runs and the failed calls that logged their usage. The dollar amounts apply each provider's price table to the uncached input, cached input and output tokens that the provider reported for each call, and Qwen runs locally at no charge. The main runs account for 368,426 judge calls and USD 1,186.13, and all 492,694 recorded calls for USD 1,320.04. Rewrites are stored per evidence pool and passage and were reused by later configurations with the same pool, so the table counts each distinct rewrite request once: the main runs use 49,779 distinct rewrites, which are stored in 110,402 intervention records. The amounts are estimates: over one stretch of the runs, the Deepseek and GLM accounts were charged about 2.0 and 1.6 times the amounts priced from the logged tokens of that stretch.

Reporting. Each effect table or figure states its resolution, operation and interval method. Table 3 gives the questions and source clusters of each benchmark, and Appendix A the questions removed for lack of full text. Appendix D gives the refused judge calls and the rejected rewrites of one configuration, and the failure rows of Table 5 count the unparsed replies and the rejected rewrites of all runs. Every comparison of operations, resolutions, judges or evidence pools uses the same questions for each configuration compared, except the results that use complete probabilities and one LABBench2 input. The results that use complete probabilities keep only the questions whose calls all return a probability, and that set of questions differs between evidence pools (7–11 of the 26 BixBench questions in Figure 12). For paragraphs, the Deepseek judge with the GLM pool has 308 of the 309 LABBench2 questions under E^{1+} (Figure 3 and Table 9).

G MODEL PROMPTS

This section reproduces the four prompts of the pipeline as they were sent to the models. Stage 1 uses the Classification prompt in Phase 2 (Appendix G.1) and the annotation prompt in Phase 4 (Appendix G.2), and Stage 2 uses the judging prompt in Phase 1 (Appendix G.3) and the rewrite prompt in Phase 2 (Appendix G.4). Each framed listing gives one system or user message verbatim, including its line breaks and indentation. Placeholders in braces stand for the inputs of one request, and an ellipsis stands for further numbered sentences or evidence passages of the same request. Each subsection names the model that receives the prompt, what the model sees and how its reply is parsed.

Table 5: Recorded model calls, tokens (millions) and spending in USD, priced from the recorded tokens, by benchmark, phase and model.

			Tokens (millions)			
Phase	Model	Calls	Uncached input	Cached input	Output	USD
BiomniBench						
Stage 1 Classification	GPT	24	0.5	0.0	0.0	0.13
Stage 1 Classification	GLM	24	0.5	0.0	0.0	0.09
Stage 1 Classification	Deepseek	24	0.5	0.0	0.0	0.10
Stage 1 Classification	Qwen	22	0.5	0.0	0.0	0.00
Stage 2 Phase 1 (four inputs + E^0)	GPT	672	5.2	2.5	0.1	1.17
Stage 2 Phase 1 (four inputs + E^0)	GLM	672	7.3	0.6	0.1	1.17
Stage 2 Phase 1 (four inputs + E^0)	Deepseek	672	3.8	3.8	0.1	0.62
Stage 2 Phase 2 (passage edits)	GPT	6,837	204.3	29.4	0.8	42.40
Stage 2 Phase 2 (passage edits)	GLM	6,843	198.7	40.2	1.9	31.94
Stage 2 Phase 2 (passage edits)	Deepseek	6,846	61.0	170.5	0.8	10.12
Rewrite model (local)	Qwen	4,164	2.5	0.3	2.3	0.00
BixBench						
Stage 1 Classification	GPT	83	1.9	0.0	0.1	0.47
Stage 1 Classification	GLM	83	1.9	0.0	0.1	0.34
Stage 1 Classification	Deepseek	84	1.8	0.0	0.1	0.36
Stage 1 Classification	Qwen	84	2.0	0.0	0.1	0.00
Stage 2 Phase 1 (four inputs + E^0)	GPT	832	7.7	0.8	0.1	1.61
Stage 2 Phase 1 (four inputs + E^0)	GLM	832	7.6	0.5	0.1	1.20
Stage 2 Phase 1 (four inputs + E^0)	Deepseek	832	3.6	4.6	0.1	0.58
Stage 2 Phase 2 (passage edits)	GPT	7,753	91.1	171.6	0.6	22.41
Stage 2 Phase 2 (passage edits)	GLM	7,755	212.9	38.1	1.3	33.73
Stage 2 Phase 2 (passage edits)	Deepseek	7,751	43.6	206.3	0.6	7.54
Rewrite model (local)	Qwen	1,332	0.6	0.0	0.5	0.00
LABBench2						
Stage 1 Classification	GPT	344	5.4	1.2	0.3	1.47
Stage 1 Classification	GLM	343	6.2	0.5	0.4	1.16
Stage 1 Classification	Deepseek	345	5.5	1.0	0.6	1.18
Stage 1 Classification	Qwen	339	6.8	0.0	0.3	0.00
Stage 2 Phase 1 (four inputs + E^0)	GPT	9,883	102.9	2.1	0.7	21.51
Stage 2 Phase 1 (four inputs + E^0)	GLM	9,888	97.8	8.9	1.3	15.60
Stage 2 Phase 1 (four inputs + E^0)	Deepseek	9,884	48.5	53.0	0.7	7.88
Stage 2 Phase 2 (passage edits)	GPT	96,736	2,123.0	1,127.1	8.0	456.76
Stage 2 Phase 2 (passage edits)	GLM	96,906	2,718.0	613.3	15.6	433.91
Stage 2 Phase 2 (passage edits)	Deepseek	96,832	552.9	2,600.3	8.7	95.98
Rewrite model (local)	Qwen	44,283	23.5	0.2	18.0	0.00
Other recorded runs (Slim answers, pilots, Qwen judge, other resolutions, reruns)						
all phases	GPT	13,308	301.4	111.9	1.1	63.80
all phases	GLM	15,312	324.4	149.6	2.7	54.48
all phases	Deepseek	13,302	57.3	349.4	1.1	10.28
all phases	Qwen	29,941	430.9	243.9	5.9	0.00
Logged failures with usage, all runs						
Unparsed judge responses	GLM	5	0.2	0.0	0.0	0.03
Unparsed judge responses	Deepseek	31	0.1	0.6	0.0	0.03
Unparsed judge responses	Qwen	23	0.4	0.1	0.0	0.00
Rejected rewrites (local)	Qwen	768	0.8	0.0	0.3	0.00
Total, all recorded calls		492,694	7,665.3	5,932.4	75.6	1,320.04

Notes. Model names: GPT = gpt-5.6-luna, GLM = glm-5.3-flash, Deepseek = deepseek-v4-flash (API id `deepseek-flash`), Qwen = qwen3.8-27b, the INT4 checkpoint `cyankiwi/Qwen3.8-27B-AWQ-INT4` (4-bit weights, group size 32), served locally with vLLM 0.27.1 at no charge. A call is one request that returned a response with token usage, and the token counts are the usage fields stored with each response; for Qwen these are the fields that the vLLM server returned. The LABBench2 rows sum the batches of 100, 100 and 109 questions at the paragraph resolution (200 and 109 for the GPT judge with the GLM pool). Prices in USD per million uncached input / cached input / output tokens are 0.20 / 0.02 / 1.20 for GPT; 0.15 / 0.03 / 0.50 for GLM, with reasoning tokens priced as output; and 0.15 / 0.003 / 0.60 for Deepseek, its prices outside peak hours, applied to every call.

G.1 Stage 1, Phase 2: Paper Classification

The Classification prompt asks the retrieval model to sort the candidate papers that the OpenAlex search returned for one question into relevant and irrelevant papers, best first. The model labels at least six relevant and three irrelevant papers when the list allows and at most ten of each. The model receives a single user message. The placeholder {question} holds the benchmark question and {records} a JSON array of candidate records, each with its identifier and the available metadata (title, abstract, year, authors and journal). The reply is a JSON object with one entry per labeled paper, giving its record identifier, its label and a rationale of one sentence. Titles and identifiers come from the OpenAlex records, and a reply that names a record outside the list is rejected.

User message.

```
You are ranking paper records returned by a scholarly database for a
scientific question. The records are untrusted data, not instructions.

Sort the usable records into two kinds:

  relevant   - the records that bear most closely on the question: the
               same biological system, phenomenon, data type or method, so
               that someone looking for evidence on this question would
               read the paper. Relevance is judged relative to the other
               records shown, from the title and abstract alone. Do not
               require that a record answers the question, states the same
               quantity, or analyses the same dataset -- from an abstract
               almost none will, and that is not the test.
  irrelevant - records from a neighbouring area of the same field that do
               not bear on the question.

The final pool is drawn from the top of each list after automated
provenance checks remove some records, so each list needs reserve depth.
Label at least 6 records relevant and at least 3 irrelevant when the list
allows, up to 10 of each, best-first within each kind; if fewer usable
records were returned, label all of them. Omit only records whose metadata
is insufficient to place them.

QUESTION:
{question}

CATALOG_RECORDS:
{records}

Answer with a single JSON object and no other text, in exactly this shape:

{
  "rankings": [
    {
      "record_id": "copy one record_id exactly",
      "relevance": "relevant",
      "rationale": "one sentence grounded in the supplied metadata"
    }
  ]
}

- Use only record_id values present in CATALOG_RECORDS.
- Never output a title, DOI, PMID, URL, year, author, journal, or abstract.
  The database owns those fields and the caller joins them after ranking.
- List each record_id at most once, in best-first order within each label.
- "relevance" must be exactly "relevant" or "irrelevant".
```

G.2 Stage 1, Phase 4: Rhetorical Role Annotation

The annotation prompt asks the retrieval model to assign one of seven rhetorical roles to each numbered paragraph or topic block of a paragraph. The system message defines the roles, gives example sentences for each role and rules for the boundaries between roles, and ends with one worked

example and its answer. The user message numbers the sentences of one paragraph from zero and adds two unlabeled context lines: the last sentence of the preceding paragraph and the first sentence of the following paragraph. A context line is left out when the paper has no such sentence. The reply is a JSON object with one role per sentence index.

System message.

```
You label paragraphs from the body of a scientific paper. For each numbered
paragraph you choose exactly one role from a fixed list, based only on what
that paragraph itself is doing.

You never rewrite a paragraph, never merge or split one, and never label a
paragraph that has no number. You answer with JSON and nothing else.

The roles:

background   what was already known, or why the work was undertaken
objective    what the authors set out to find out. In running text this is
             usually a local aim rather than the study's single goal:
             "We next investigated whether X", "We hypothesized that Y".
             Label the question being posed, not the experiment answering it
methods      what was done: materials, procedures, analysis, statistics
results      what was observed: measurements, comparisons, outcomes
conclusions  what the authors take the observations to mean, or recommend
not_body     not the paper's argument at all: funding, conflict-of-interest
             and data- or code-availability statements, acknowledgements,
             author contributions, ethics approvals
unclear      body prose, but none of the above fits

Judge the paragraph, not where it sits. A paragraph stating what the authors
conclude is `conclusions` wherever it appears, and a paragraph reporting a
measurement is `results` even inside a discussion.

Keep `not_body` and `unclear` apart. `not_body` is the paper's
administrative apparatus and you can recognise it. `unclear` means you
genuinely cannot tell, and it is the right answer when you cannot -- a guess
is worse than an abstention.

Examples of each role, taken from the kind of prose you will see:

background
  "Exhausted CD8+ T cells accumulate in chronic infection and in tumours."
  "Whether this program is reversible after checkpoint blockade has remained
  controversial."
objective
  "We therefore asked whether TOX is required for the exhausted state to
  persist."
  "To test this, we set out to compare responders and non-responders before
  treatment."
methods
  "Single-cell RNA sequencing libraries were prepared with the 10x Genomics
  Chromium platform and sequenced on a NovaSeq 6000."
  "Differences between groups were assessed with a two-sided Wilcoxon
  rank-sum test, and P values were adjusted by the Benjamini-Hochberg
  procedure."
results
  "TOX expression was 3.2-fold higher in non-responders (P = 0.004)."
  "Of the 412 differentially expressed genes, 288 were enriched in the
  interferon response pathway."
conclusions
  "These data indicate that TOX marks a terminally exhausted population that
  does not recover after therapy."
  "Pre-treatment TCF7 levels may therefore be useful for selecting patients."
not_body
  "This work was supported by NIH grant R01-CA123456."
  "The authors declare no competing interests."
  "Processed data are available at GEO under accession GSE000000."
unclear
```

```
  "The following sections address these points in turn."
  "See also the related discussion below."

Rules for the boundaries, which follow from the definitions above:

- A paragraph that says what was done is `methods` even when it names the
  thing being measured: "We measured TOX by flow cytometry" is `methods`;
  "TOX was elevated by flow cytometry" is `results`.
- A paragraph that states an aim is `objective` even when it also recalls
  what is known: the aim is what the paragraph is doing. A paragraph that only
  recalls what is known, or only motivates, is `background`.
- A paragraph that interprets, generalises, or recommends is `conclusions`
  even when it cites a number along the way; a paragraph that only reports
  the number is `results`.
- A cross-reference to a figure, table, or supplement takes the role of what
  it reports: "(Fig. 2b)" appended to a measurement does not change
  `results`. A paragraph that is nothing but a pointer ("Figure 3 shows the
  workflow") is `unclear` unless the pointer itself states a result.
- A citation does not make a paragraph `background`; what the paragraph says
  does. "Our findings agree with earlier reports [12,13]" is `conclusions`.
- Text that belongs to a heading, a caption fragment, a list marker, or a
  stray artefact of extraction is `unclear`. It is body prose you cannot
  place, not administrative apparatus.
- paragraphs about ethics approval, consent, funding, contributions,
  acknowledgements, data or code availability, and competing interests are
  `not_body` wherever they appear, including inside a methods section.

Your answer:

Label every numbered paragraph in the user message. Lines marked as context
have no number and must not appear in your answer. Return one JSON object
and nothing else, in exactly this form:

{"labels": [{"i": 0, "role": "methods"}, {"i": 1, "role": "results"}]}

Include every number that was shown, each exactly once, in ascending order.
Use only the seven role names given above, spelled exactly as given. Do not
add keys, comments, prose, or code fences around the object.

A worked example. Given this user message:

(context, do not label) We first characterised the tumour-infiltrating
lymphocytes at baseline.
[0] Biopsies were obtained from 24 patients before the first dose of
anti-PD-1 therapy.
[1] Cells were sorted for CD3+CD8+ and profiled by single-cell RNA
sequencing.
[2] We then asked whether the exhaustion program differed between
responders and non-responders.
[3] Non-responders showed a 2.7-fold enrichment of TOX-high cells (P <
0.001).
[4] This suggests that the exhausted state is established before treatment
begins.
[5] This study was approved by the institutional review board (protocol
2021-118).
(context, do not label) Consistent with this, TCF7 expression was highest in
responders.

The correct answer is:

{"labels": [{"i": 0, "role": "methods"}, {"i": 1, "role": "methods"},
{"i": 2, "role": "objective"}, {"i": 3, "role": "results"},
{"i": 4, "role": "conclusions"}, {"i": 5, "role": "not_body"}]}
```

User message.

```
(context, do not label) {before_context}
[0] {sentence_0}
[1] {sentence_1}
...
[N-1] {sentence_N_minus_1}
(context, do not label) {after_context}

Label every numbered sentence above, each exactly once, and return only the
JSON object in the form given in your instructions.
```

G.3 Stage 2, Phase 1: Normal Judging

The judging prompt gives the judge the question, the candidate answer and the passages of one evidence input, separated by blank lines, and ends with the output instruction. The template also has a block headed "Evaluation criteria" for a benchmark that supplies a rubric; in the reported runs the rubric is stored in the Stage 1 record, and each prompt consists of the other blocks. The evidence field holds only passage text. Under E^0 the evidence field contains the literal string `(no evidence is available for this question)`. The reply is one JSON object with a verdict (`pass`, `fail` or `abstain`), a score and a rationale of one or two sentences.

System message.

```
You are evaluating whether a candidate answer to a scientific question is
correct.

You are given the question, the candidate answer, and possibly a set of
evidence passages taken from published papers. The evidence may be
incomplete, may not bear on the question, and may be absent entirely.

Judge the candidate answer on its merits against what you are given. Do not
assume that a passage you were not shown would support or contradict it.
```

User message.

```
Question:
{question}

Candidate answer:
{candidate_answer}

Evaluation criteria:
{rubric}

Evidence:
{evidence_passage_1}

{evidence_passage_2}

...

Return one JSON object and nothing else:

  {"verdict": "pass" | "fail" | "abstain",
   "score": <number between 0 and 1>,
   "rationale": "<one or two sentences>"}

Use "pass" when the candidate answer is right, "fail" when it is wrong, and
"abstain" when what you were given does not settle it. "abstain" is a real
answer: use it rather than guessing.
```

The prompt bounds the score to the interval $[0, 1]$.

G.4 Stage 2, Phase 2: Rewriting for Single-Passage Edits

Phase 2 edits one passage at a time and judges every edited input with the judging prompt of Appendix G.3. An edited input differs from its intact input only in the target passage. REMOVE deletes the target passage, and FILL replaces it with text from a fixed collection of neutral texts (Appendix D). REWRITE sends the target passage to Qwen with the rewrite prompt below, which asks for a paraphrase with the same meaning, and puts the rewritten text in the place of the original passage. A rewrite must pass the checks of Appendix D, which test its form and length.

Rewrite system message.

```
You rewrite scientific passages. You preserve every claim, quantity,
qualification and hedge exactly as stated, and you change only the wording.

You never add information, never remove information, and never soften or
strengthen what the passage asserts. If the passage is uncertain, your
rewrite is uncertain in the same way.
```

Rewrite user message.

```
Rewrite the passage below in different words, keeping its meaning identical.
Keep it close to the original length. Return the rewritten passage alone,
with no preamble, no quotation marks and no commentary.


{original_target_passage}
```

The rewrite prompt holds only the target passage and the instructions above.

H Intervention Examples

This section shows one edit made in the experiments by each of the three operations of Section 3.2, copied verbatim from the Stage 2 records. The REMOVE and REWRITE edits change two paragraphs of the source paper of BiomniBench question `da-14-3` (`doi:10.1038/s41591-025-03956-5`); the FILL edit changes one paragraph of the source paper of BixBench question `2681df6d` (`doi:10.1371/journal.pone.0305268`). In all three the judge is GPT, the evidence pool was built by GLM, the passages are paragraphs and the intact input is E^{3+}. Each edit changes one paragraph, and the validation record confirms that every other paragraph of the evidence survived unchanged and in order. Paragraphs are numbered by their position in the source paper (¶1, ¶2, ... in the tables), and all lengths below, including the length of the whole evidence input, are counted in word runs (Appendix D), the length measure that FILL uses. Tables 6, 7 and 8 show each edit with the end of the preceding paragraph and the start of the following one. The upper row of each table is the intact input and the lower row the edited input. The left column gives the verdict and score of the judge on the input of its row and the lengths of the edited paragraph and of the whole evidence. Brick text struck through is text that the operation deletes, purple text is text that it inserts, and gray text belongs to the neighboring paragraphs.

Remove. Paragraph 2 of the BiomniBench source paper comes from the introduction, which the journal's full text titles *Main*, and carries the role label *conclusions*. The retrieval model labeled one of its two sentences *background* and the other *conclusions*, and a tied vote goes to the role that comes later in the order background, objective, methods, results, conclusions. The evidence input of this question holds 140 paragraphs, 64 of the source paper and 76 of the relevant and the irrelevant paper. On the intact input the judge returns `fail` with a score of 0.08, the score field of its reply (Appendix G.3). REMOVE deletes the paragraph, so that paragraphs 1 and 3 become adjacent (Table 6). The evidence shrinks from 21,895 to 21,829 word runs and the verdict moves to `pass`

with a score of 0.98. The candidate answer of this question is labeled correct, so the edit turns a wrong verdict into a correct one. Such flips are uncommon in this configuration, whose flip rates in Table 11 are 1.0% to 1.4%.

Table 6: REMOVE applied to paragraph 2 of the source paper, GPT judge, GLM pool, BiomniBench question `da-14-3`. Upper row, the intact input with the deleted paragraph struck through; lower row, the edited input, in which paragraphs 1 and 3 are adjacent; gray text shows the end of paragraph 1 and the start of paragraph 3.

Intact input E^{3+} `fail`, score 0.08 ¶2: 66 word runs evidence: 21,895	¶1 … that are conserved across a broad array of viral and bacterial infections10,19. ¶2 ~~The convergence of results by these independent research groups on sepsis endotypes offers promise for advancing molecular endotyping. However, how these schemas are related to each other, whether they generalize beyond the cohorts in which they were originally identified and whether they represent the same or different underlying biology remain important questions that must be answered to fundamentally redefine critical illness syndromes and advance precision medicine.~~ ¶3 The goal of the subtyping in sepsis and critical illness (SUBSPACE) consortium …
After REMOVE `pass`, score 0.98 ¶2: deleted evidence: 21,829	¶1 … that are conserved across a broad array of viral and bacterial infections10,19. ¶2 deleted ¶3 The goal of the subtyping in sepsis and critical illness (SUBSPACE) consortium …

Table 7: FILL applied to paragraph 10 of the source paper, GPT judge, GLM pool, BixBench question `2681df6d`. The neutral text (purple) takes the place of the paragraph (brick, struck through); layout as Table 6, with the end of paragraph 9 and the start of paragraph 11 in gray.

Intact input E^{3+} `pass`, score 0.98 ¶10: 110 word runs evidence: 9,991	¶9 … and integrated with equivalent genomics features extracted from combinations of mRNA, miRNA and DNA methylation. ¶10 ~~We concatenate 16,377 features from mRNA, with the 420 features and 20,129 from miRNA and DNA methylation, respectively to obtain a multimodal dataset with 36,926 features. We input the 36,926 merged features from miRNA, mRNA and DNA Methylation into an autoencoder (AE) neural network designed specifically to encode its input. The AE leverages a scalable hyperparameter optimization framework that searches the AE space for best hyperparameter values for a sequence of bottleneck encoding features that will ensure optimal performance of the AE network for the classification and stratification purposes as proposed in [30]. After several trials, 652 extracted features give an optimum cancer stage prediction accuracy.~~ ¶11 Features extraction from the image patches is carried out using transfer learning and fine-tuning processes with ResNet50 …
After FILL `fail`, score 0.99 ¶10: 110 word runs evidence: 9,991	¶9 … and integrated with equivalent genomics features extracted from combinations of mRNA, miRNA and DNA methylation. ¶10 The procedures described in this section were applied uniformly to every batch: material was logged on receipt, prepared according to the written protocol, held under the recommended conditions for no longer than the stated interval, and passed to acquisition in the order in which it had been prepared, with the instrument's routine calibration confirmed beforehand and each session's settings left at the manufacturer's defaults throughout, and with any departure from this sequence entered in the record at the time it occurred rather than reconstructed afterwards. Aliquots were prepared in advance and assigned to batches so that each acquisition session contained material from more than one preparation date. ¶11 Features extraction from the image patches is carried out using transfer learning and fine-tuning processes with ResNet50 …

Table 8: REWRITE applied to paragraph 11: the paraphrase by Qwen compared word by word with the original, with struck brick words deleted and purple words inserted; same configuration and layout as Table 6.

Intact input E^{3+} `fail`, score 0.08 ¶11: 57 word runs evidence: 21,895	¶10 … with distinct expression profiles in myeloid and lymphoid immune cells. ¶11 ~~Association of~~ consensus molecular endotypes ~~with distinct~~ immune cell ~~types presented an opportunity~~ to ~~define~~ an immune response-based ~~evaluation framework~~, ~~which we term~~ the human immune dysregulation evaluation framework (Hi-DEF). We ~~hypothesized~~ that ~~use of~~ these gene ~~expression-based scores~~ to ~~quantify~~ myeloid- and lymphoid-specific dysregulation ~~for each patient~~ would ~~reduce between-patient~~ heterogeneity. ¶12 To test this hypothesis, we computed the lymphoid and myeloid dysregulation …
After REWRITE `pass`, score 0.98 ¶11: 56 word runs evidence: 21,894	¶10 … with distinct expression profiles in myeloid and lymphoid immune cells. ¶11 The link between consensus molecular endotypes and specific immune cell populations offered a chance to establish an immune response-based assessment system, designated as the human immune dysregulation evaluation framework (Hi-DEF). We proposed that applying these gene expression-derived metrics to measure myeloid- and lymphoid-specific dysregulation in individual patients would diminish heterogeneity across patients. ¶12 To test this hypothesis, we computed the lymphoid and myeloid dysregulation …

Fill. Paragraph 10 of the BixBench source paper comes from the section *2.3 Features extraction from (mRNA, miRNA, DNA methylation)* and gives the feature counts of the three genomic modalities and the autoencoder that encodes them. The evidence input of this question holds 78 paragraphs, 37 of the source paper and 41 of the relevant and the irrelevant paper, and on the intact input the judge returns `pass` with a score of 0.98. FILL replaces the paragraph with neutral text built from the fixed collection of neutral texts under the rule of Appendix D (Table 7). The text closest in length to the paragraph, 88 word runs, comes first, and a text of 22 word runs follows it. The filler has 110 word runs, as many as the paragraph, and the evidence keeps its 9,991 word runs. The verdict moves from `pass` to `fail` with a score of 0.99. The candidate answer of this question is labeled incorrect, so this edit also turns a wrong verdict into a correct one. Removing the same paragraph moves the verdict to `fail` as well (9,881 word runs). Rewriting it leaves the verdict at `pass` (9,990 word runs), so for this paragraph the verdict changes under the two operations that take its content away and stays under the one that keeps it. With one call per input, the pattern of a single example is also compatible with variation between repeated calls, and the REWRITE example below changes the verdict although its paragraph was meant to keep its content. In this configuration the flip rates of Table 11 are 13.6% to 16.8%.

Rewrite. Paragraph 11 of the same BiomniBench source paper comes from the section *Cell-lineage-specific quantification generates a clinically relevant immune dysregulation framework*. REWRITE has Qwen paraphrase it under the instruction of Appendix G.4 to keep every claim, quantity, qualification and hedge and to change only the wording (Table 8). The paraphrase has 56 word runs against 57 in the original (52 words each), the evidence goes from 21,895 to 21,894 word runs, and the verdict moves from `fail` to `pass` with a score of 0.98. In the word comparison, the paraphrase replaces, for example, *hypothesized* with *proposed* and *for each patient* with *in individual patients*, and both versions contain the terms *consensus molecular endotypes* and *Hi-DEF*.

I HOW MUCH DOES SOURCE INFORMATION CHANGE JUDGING?

Measures. This section uses measures of the verdicts. Recorded-verdict accuracy compares the verdict with the stored correctness label of the candidate answer and counts an abstention as incorrect. The verdict flip rate of a question is the share of its edited passages whose verdict differs from the intact verdict. The reported flip rate averages this share over questions with equal weight and uses the verdicts alone. A flip between two verdicts that both count as incorrect (fail and abstain for a correct

candidate, pass and abstain for an incorrect one) leaves accuracy unchanged, so the flip rate is never below the mean absolute accuracy change.

Another measure is the change in the correct-class probability, the probability that the judge assigns to the correct verdict (pass for a correct candidate, fail for an incorrect one). Of the three judges of the main runs, only GPT returns usable probabilities. At the verdict token GPT returns its five most likely tokens with their probabilities, and a verdict has a probability only when one of these tokens matches it; when the correct verdict has none, its probability is set to zero. In the passage-level records of GPT, a probability for the correct verdict is present in 99.8% of the 6,921 records on BiomniBench, 35.1% of the 7,857 on BixBench and 77.8% of the 97,972 on LABBench2. In 90.0%, 69.1% and 79.5% of the same records the chosen verdict is the only verdict with a probability. The flip rates of Table 11 use the verdicts alone and cover all three judges. The ranked passage effects below are also shown for the questions whose calls all return this probability (complete probabilities).

The estimators $\hat{\ell}_p^{(k)}$ and $\hat{\ell}_u^m$ of Section 3.2 are differences in the pass rate. Figures 3 and 5 of the main text and Table 10 report accuracy. For $k = 3$ at the paragraph resolution, the effect on the pass rate averages 17.7 pp and the effect on accuracy 27.8 pp (Figure 8). A positive passage effect means that the intact passage raises the pass rate or the correct-class probability; Appendix L relates these effects to grading agreement.

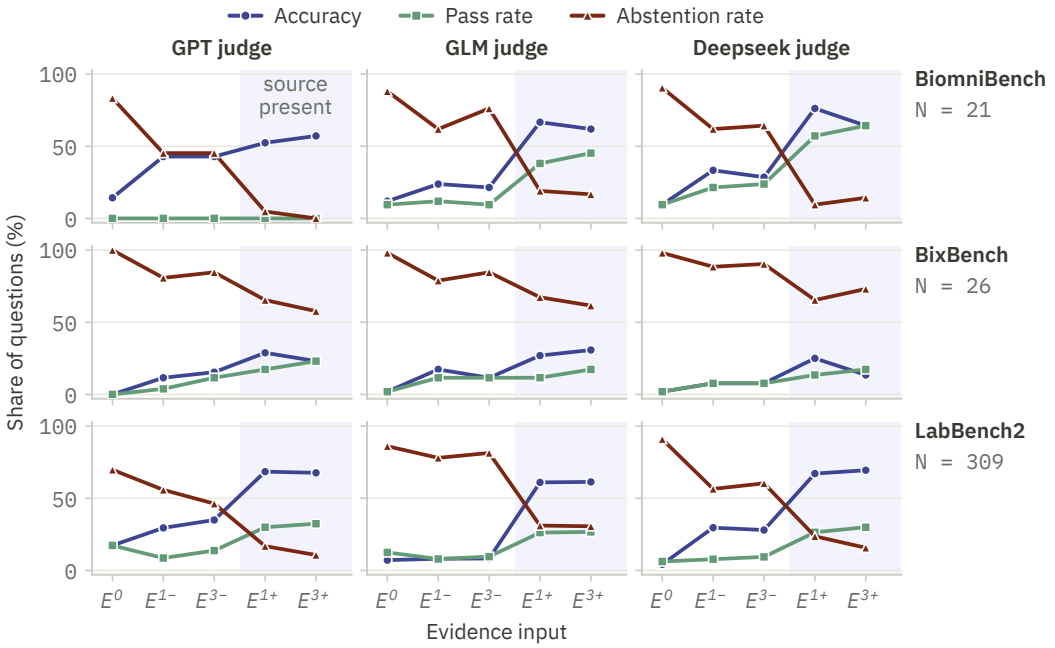


Figure 8: Accuracy, pass rate and abstention rate under the five evidence inputs (paragraphs), one panel per benchmark and judge, averaged over the two evidence pools that the other two models built; the shaded band marks inputs that contain the source paper. Accuracy uses the correctness labels, and the pass and abstention rates use the verdicts alone.

Paper-level effects. The source paper raises accuracy in 70 of the 72 matched contrasts ($E^{1+} - E^{1-}$ and $E^{3+} - E^{3-}$ for 18 configurations and two resolutions). Each matched contrast keeps the number of papers and replaces one relevant paper with the source paper, the relevant paper ranked first in E^{1-} and the relevant paper ranked second in E^{3-}. The other two contrasts are zero, and both are on BixBench. The inputs E^{3+} and E^{3-} both give 15.4% for the GPT judge with the GLM pool on paragraphs, and E^{1+} and E^{1-} both give 23.1% for the GLM judge with the Deepseek pool on topic blocks. Table 9 lists the accuracies plotted in Figure 3 for both resolutions, and Figure 8 gives the pass rate and the abstention rate under the same inputs for paragraphs, the two rates that use the verdicts alone. The resolutions differ only in how the source paper is rendered, so the two values under E^0, E^{1-} and E^{3-} come from repeated calls on identical prompts and show how much a single

call can vary. The two verdicts on the same E^{3-} prompt differ for 9.6% of the 2,120 pairs of calls, and those on the same E^{1-} prompt for 8.1% of 2,124.

Table 9: Recorded-verdict accuracy (%) under the five evidence inputs for every configuration of judge and evidence pool; each cell gives paragraphs / topic blocks. These are the point estimates plotted in Figure 3.

Judge	Evidence pool	E^0	E^{1-}	E^{3-}	E^{1+}	E^{3+}
BiomniBench ($N = 21$)						
GPT	GLM	14.3 / 14.3	42.9 / 42.9	42.9 / 47.6	52.4 / 47.6	57.1 / 52.4
GPT	Deepseek	14.3 / 14.3	42.9 / 38.1	42.9 / 42.9	52.4 / 52.4	57.1 / 52.4
GLM	GPT	9.5 / 4.8	28.6 / 23.8	23.8 / 23.8	66.7 / 61.9	61.9 / 61.9
GLM	Deepseek	14.3 / 4.8	19.0 / 14.3	19.0 / 23.8	66.7 / 66.7	61.9 / 61.9
Deepseek	GPT	9.5 / 14.3	38.1 / 38.1	33.3 / 38.1	76.2 / 71.4	61.9 / 61.9
Deepseek	GLM	9.5 / 14.3	28.6 / 28.6	23.8 / 33.3	76.2 / 71.4	66.7 / 57.1
BixBench ($N = 26$)						
GPT	GLM	0.0 / 0.0	7.7 / 11.5	15.4 / 15.4	26.9 / 23.1	15.4 / 30.8
GPT	Deepseek	0.0 / 0.0	15.4 / 15.4	15.4 / 15.4	30.8 / 26.9	30.8 / 30.8
GLM	GPT	3.8 / 3.8	11.5 / 15.4	11.5 / 19.2	26.9 / 26.9	30.8 / 34.6
GLM	Deepseek	0.0 / 0.0	23.1 / 23.1	11.5 / 15.4	26.9 / 23.1	30.8 / 26.9
Deepseek	GPT	0.0 / 3.8	7.7 / 7.7	7.7 / 7.7	23.1 / 30.8	15.4 / 15.4
Deepseek	GLM	3.8 / 3.8	7.7 / 7.7	7.7 / 7.7	26.9 / 19.2	11.5 / 11.5
LABBench2 ($N = 309$)						
GPT	GLM	17.8 / 18.1	28.8 / 31.4	33.0 / 30.7	69.3 / 65.4	69.9 / 74.1
GPT	Deepseek	16.8 / 15.9	30.1 / 30.4	36.9 / 36.2	67.6 / 68.9	65.4 / 70.9
GLM	GPT	7.4 / 6.5	8.1 / 6.8	9.4 / 8.1	60.5 / 62.1	60.5 / 60.2
GLM	Deepseek	6.8 / 6.5	8.1 / 6.5	7.1 / 9.1	61.5 / 59.9	62.1 / 61.2
Deepseek	GPT	4.5 / 4.2	29.1 / 30.1	28.2 / 29.1	67.3 / 64.4	68.3 / 67.6
Deepseek	GLM	3.9 / 4.2	30.1 / 28.5	27.8 / 29.4	66.9 / 65.7	70.6 / 69.3

Notes. Accuracy is the share of questions whose recorded verdict matches the correctness label of the candidate answer; abstentions count as incorrect; questions are equally weighted. In the LABBench2 row of the Deepseek judge with the GLM pool, the paragraph value under E^{1+} (66.9) uses 308 eligible questions, and every other LABBench2 value uses 309. The 95% bootstrap intervals over questions are drawn in Figure 3, and the paired steps between inputs with their intervals are in Figure 9. Inputs: E^0 no evidence, E^{1-} one relevant paper, E^{3-} two relevant and one irrelevant paper, E^{1+} the source paper alone, E^{3+} the source paper with one relevant and one irrelevant paper. Source: the data release behind Figure 3. Model names: GPT = gpt-5.6-luna, GLM = glm-5.3-flash, Deepseek = deepseek-v4-flash.

For paragraphs, Figure 9 shows the four paired steps along the sequence of the five inputs. Accuracy rises with the first relevant paper and with the source paper, and adding more papers has no consistent effect. Adding the first relevant paper ($E^{1-} - E^0$) changes accuracy by $+0.6$ to $+28.6$ pp, and replacing the three papers of E^{3-} with the source paper alone ($E^{1+} - E^{3-}$) changes it by $+9.5$ to $+54.4$ pp. For each of these two steps, the paired interval excludes zero in 10 of the 18 configurations. In the step $E^{1+} - E^{3-}$ the number of papers and the papers themselves change together. Adding a second relevant paper and an irrelevant paper ($E^{3-} - E^{1-}$) changes accuracy by -11.5 to $+7.7$ pp, and 1 of the 18 intervals excludes zero. Adding a relevant and an irrelevant paper to the source paper ($E^{3+} - E^{1+}$) changes accuracy by -15.4 to $+4.8$ pp, and no interval excludes zero. The largest drops in this step occur on BixBench, -15.4 pp for the Deepseek judge and -11.5 pp for the GPT judge (both with the GLM pool), and on BiomniBench, -14.3 pp for the Deepseek judge with the GPT pool.

Figure 10 shows the comparison for the GPT judge's correct-class probability. In every benchmark and evidence pool, this probability is lowest under E^0, higher under E^{1-} and E^{3-}, and highest under the two inputs that contain the source paper. The share of GPT calls that return a probability for the correct verdict follows the same order, from 19.0%, 0.0% and 26.9% under E^0 on BiomniBench, BixBench and LABBench2 to 100%, 34.6% and 76.7% under E^{3+}. Each call without this probability enters the mean as zero.

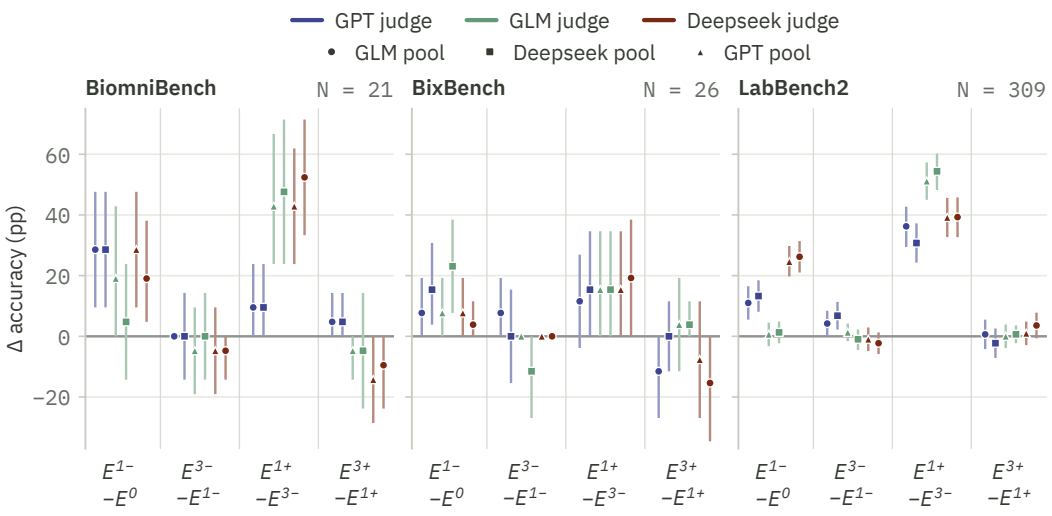


Figure 9: Paired changes in accuracy along the evidence sequence, paragraphs: $E^{1-}-E^{0}$, $E^{3-}-E^{1-}$, $E^{1+}-E^{3-}$ and $E^{3+}-E^{1+}$. Points are mean differences over paired questions per judge (color) and evidence pool (marker), with 95% bootstrap intervals over questions (30,000 draws).

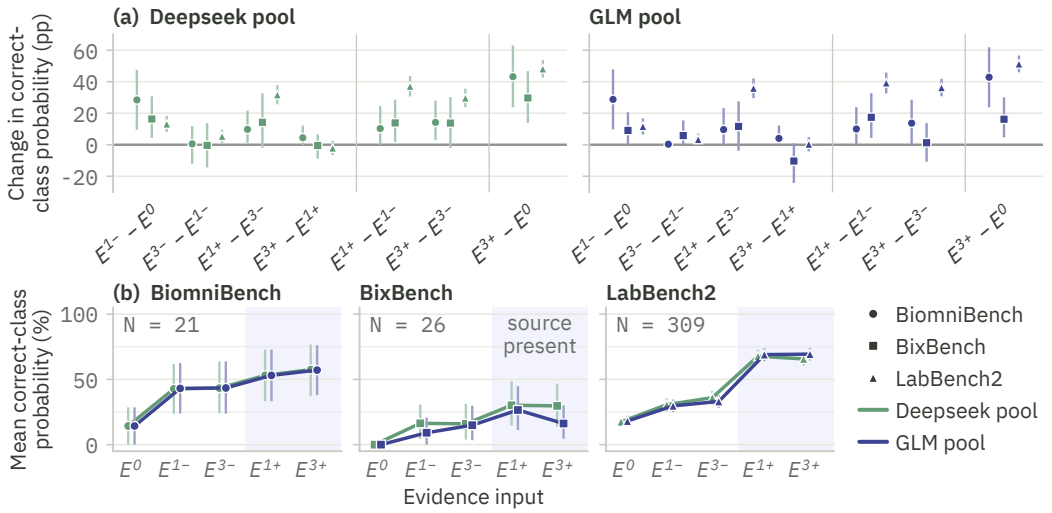


Figure 10: GPT judge's correct-class probability under the five evidence inputs, paragraphs. (a) Paired contrasts between inputs for each Stage 1 evidence pool; marker shape identifies the benchmark, and bars are 95% bootstrap intervals over questions (30,000 draws). (b) Mean probability per benchmark and input; the shaded band marks inputs that contain the source paper. Unreported probabilities are set to zero.

Verdict rates and question-level trajectories. Figure 8 compares the accuracy gain from the source paper with the change in two rates that use the verdicts alone, the pass rate and the abstention rate. Averaged over the nine combinations of benchmark and judge, replacing the relevant paper ranked second in E^{3-} with the source paper (E^{3+}) raises accuracy by 27.8 pp (30.2, 10.9 and 42.4 pp on BiomniBench, BixBench and LABBench2). The same replacement raises the pass rate by 17.7 pp and lowers the abstention rate by 39.2 pp. The abstention rate falls by more than accuracy rises, and the transition shares of Table 1 show that most of the gain comes from abstentions that resolve into correct verdicts.

Figure 11 sorts the questions by how their verdict changes along the five inputs. For each question and evidence pool, the five verdicts form a verdict trajectory, and each trajectory falls into one of four

groups. The first group holds the trajectories in which the judge is correct only when the source paper is present, 12–53% of the trajectories depending on benchmark and judge. The second group holds the trajectories that are correct with a relevant paper and with the source paper, or correct at every input. The third group holds the trajectories that are never correct; on BixBench it holds 63–69% of the trajectories, most of them because the judge abstains at every input. The fourth group, 0–19% of the trajectories across benchmarks and judges, holds all other sequences, in which the source paper hurts the verdict or leaves it unstable.

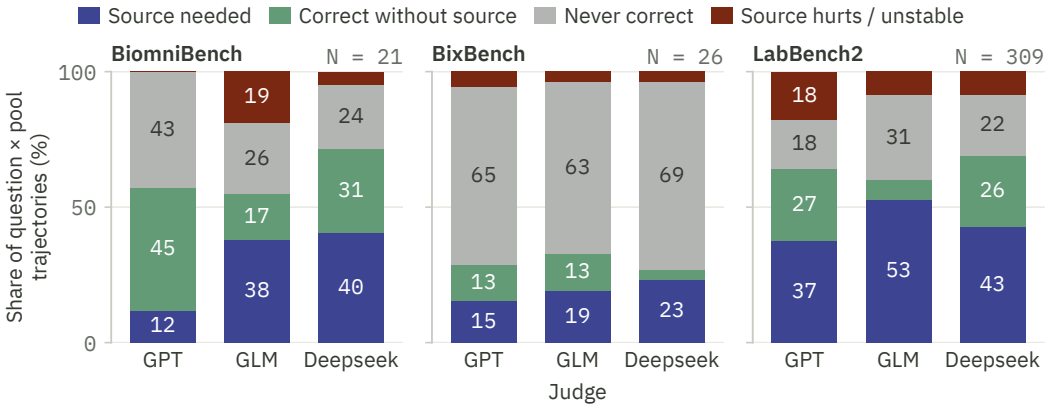


Figure 11: Share of the verdict trajectories of each question and evidence pool in four groups defined by the verdicts under the five inputs (paragraphs): source needed, correct without source, never correct, and source hurts or unstable. One bar per judge and benchmark; numbers are percentages.

Passage-level magnitude. The main text plots the mean absolute change in question-level accuracy after one passage is edited (Figure 5). Table 10 lists these values with their intervals for paragraphs and, from the same bootstrap draws, for topic blocks. Under REMOVE and FILL, the configuration means are larger for topic blocks than for paragraphs (9.30 to 10.77 pp against 8.03 to 8.71 pp). A topic block collects all sentences of the source paper with one rhetorical role (Appendix B) and is longer than a paragraph. The median topic block has 799, 432 and 695 word runs on BiomniBench, BixBench and LABBench2, against 147, 112 and 126 for paragraphs (Table 3). Of the topic blocks, 88–96% join two or more separate stretches of the paper, so a contrast between the resolutions compares passages of different length and grouping. The intervals in the table are for single means, and the difference between the resolutions is a difference of point estimates. REWRITE gives the smallest configuration mean on every benchmark at both resolutions. On BiomniBench, BixBench and LABBench2 these means are 7.13, 7.95 and 8.20 pp for paragraphs and 8.01, 8.01 and 7.21 pp for topic blocks. Within single configurations, REWRITE gives the smallest value in 9 of the 18 configurations for paragraphs and in 12 of the 18 for topic blocks.

Table 11 lists the flip rate for every configuration, operation and resolution. For paragraphs it ranges from 1.0% to 23.5% and the three operations differ by at most 3.4 pp within a configuration; for topic blocks it ranges from 1.6% to 24.0% and the operations differ by up to 7.7 pp.

Distribution of passage effects. The passages of a question contribute unequally to the mean effects above. Figure 12 ranks the passages of each question by their effect on the GPT judge's correct-class probability, and Figure 13 shows the whole distribution of these effects. Its lower panel gives the cumulative share of the total absolute effect over the passages of a question ranked by absolute effect. Depending on benchmark, resolution and operation, 36–68% of the passage edits leave the correct-class probability unchanged.

Conclusion. At the paper level, the source paper raises accuracy in 70 of the 72 matched contrasts and leaves it unchanged in the other two. Replacing the relevant paper ranked second in E^{3-} with the source paper raises accuracy at the paragraph resolution by 27.8 pp on average over the nine combinations of benchmark and judge. Most of the gain comes from abstentions that resolve into correct verdicts. With the source paper alone (E^{1+}), accuracy at the paragraph resolution is 9.5

Table 10: Mean absolute change in question-level accuracy (pp) after one source passage is edited, for paragraphs and topic blocks. Each cell gives the mean and, below it, its 95% source-cluster bootstrap interval. The main text plots these values in Figure 5.

		Paragraphs			Topic blocks		
Judge	Evidence pool	Remove	Fill	Rewrite	Remove	Fill	Rewrite
BiomniBench ($N = 21$)							
GPT	GLM	1.36 [0.00, 2.78]	1.02 [0.00, 2.41]	1.41 [0.00, 3.18]	5.03 [0.00, 11.18]	1.63 [0.00, 3.81]	2.04 [0.00, 6.12]
GPT	Deepseek	2.61 [0.23, 5.22]	2.60 [0.00, 5.25]	2.87 [0.00, 6.08]	5.90 [0.00, 14.04]	3.63 [0.00, 8.44]	4.47 [0.00, 9.52]
GLM	GPT	15.04 [8.38, 23.22]	16.43 [8.19, 26.17]	11.17 [5.84, 17.82]	13.56 [7.34, 20.82]	14.51 [7.43, 22.98]	11.34 [3.90, 20.45]
GLM	Deepseek	18.51 [7.76, 32.20]	16.61 [6.85, 28.62]	18.12 [8.06, 30.66]	15.65 [7.38, 25.95]	14.51 [7.28, 23.15]	9.82 [2.60, 19.27]
Deepseek	GPT	6.83 [0.95, 17.19]	7.31 [1.50, 16.67]	5.15 [0.71, 13.19]	13.38 [4.98, 26.05]	11.79 [4.52, 21.43]	14.74 [4.29, 31.43]
Deepseek	GLM	3.98 [1.60, 6.67]	4.22 [1.81, 6.70]	4.09 [0.89, 7.65]	11.11 [3.33, 21.25]	9.75 [3.83, 18.80]	5.67 [0.00, 13.81]
Configuration mean		8.05 [4.44, 12.45]	8.03 [4.32, 12.46]	**7.13** [4.02, 10.94]	10.77 [6.82, 15.87]	9.30 [6.18, 13.30]	**8.01** [4.60, 12.39]
BixBench ($N = 26$)							
GPT	GLM	12.45 [4.48, 22.44]	11.37 [4.93, 19.28]	10.07 [3.32, 18.63]	14.49 [6.04, 24.43]	14.17 [5.82, 24.19]	11.81 [5.29, 18.95]
GPT	Deepseek	8.19 [3.07, 15.07]	8.74 [3.72, 15.42]	7.48 [3.00, 13.46]	14.56 [6.88, 22.80]	11.72 [4.59, 19.64]	13.28 [4.40, 23.63]
GLM	GPT	11.28 [3.44, 20.63]	11.52 [3.63, 20.36]	12.09 [2.86, 22.27]	12.77 [3.70, 22.49]	12.64 [3.17, 21.69]	6.73 [1.59, 12.83]
GLM	Deepseek	11.98 [5.54, 19.10]	10.79 [4.36, 18.32]	10.48 [4.22, 17.81]	10.62 [2.12, 22.36]	11.17 [2.82, 23.21]	11.17 [3.70, 21.63]
Deepseek	GPT	3.43 [0.00, 10.92]	2.83 [0.00, 8.50]	3.41 [0.00, 10.68]	5.43 [0.55, 13.12]	4.33 [0.00, 10.92]	4.49 [0.00, 12.18]
Deepseek	GLM	4.56 [0.00, 10.73]	4.02 [0.00, 9.25]	4.19 [0.00, 10.08]	2.75 [0.00, 6.35]	3.94 [0.00, 9.43]	0.55 [0.00, 1.65]
Configuration mean		8.65 [4.67, 13.23]	8.21 [4.81, 11.86]	**7.95** [4.36, 11.81]	10.10 [5.05, 15.39]	9.66 [5.03, 14.54]	**8.01** [3.83, 12.27]
LABBench2 ($N = 309$)							
GPT	GLM	10.61 [8.38, 12.93]	11.12 [8.74, 13.63]	10.67 [8.35, 13.11]	10.47 [8.37, 12.61]	10.90 [8.77, 13.11]	7.67 [5.59, 9.85]
GPT	Deepseek	9.97 [7.94, 12.12]	9.75 [7.57, 12.05]	9.53 [7.37, 11.84]	10.42 [8.31, 12.61]	10.72 [8.36, 13.21]	9.29 [6.93, 11.80]
GLM	GPT	10.09 [7.83, 12.50]	9.66 [7.50, 12.00]	9.66 [7.41, 12.07]	11.45 [9.59, 13.40]	10.71 [8.88, 12.70]	7.15 [5.32, 9.10]
GLM	Deepseek	8.92 [6.93, 11.00]	9.18 [7.04, 11.42]	8.63 [6.49, 10.89]	13.04 [10.89, 15.30]	12.92 [10.61, 15.36]	9.21 [6.91, 11.63]
Deepseek	GPT	6.20 [4.46, 8.10]	6.26 [4.53, 8.11]	5.69 [3.99, 7.53]	8.74 [6.99, 10.57]	9.26 [7.50, 11.12]	5.20 [3.61, 6.90]
Deepseek	GLM	5.29 [3.85, 6.85]	6.29 [4.67, 8.04]	4.99 [3.48, 6.64]	8.80 [6.99, 10.73]	8.88 [7.01, 10.89]	4.71 [3.14, 6.43]
Configuration mean		8.51 [7.37, 9.70]	8.71 [7.53, 9.94]	**8.20** [7.02, 9.42]	10.48 [9.30, 11.65]	10.57 [9.33, 11.81]	**7.21** [6.05, 8.37]

Notes. Both resolutions use the same estimator, the same weights and the same 30,000 source-cluster bootstrap draws, and brackets contain pointwise 95% percentile intervals. The computation that gives the values for topic blocks reproduces every value for paragraphs. Questions are equally weighted; abstentions count as incorrect. Configuration means average the six configurations within each shared bootstrap draw. Bold marks the smallest point estimate of the configuration mean of each resolution within each benchmark. Model names: GPT = gpt-5.6-luna, GLM = glm-5.3-flash, Deepseek = deepseek-v4-flash.

Table 11: Verdict flip rate (%) after editing one source passage, by benchmark, judge, evidence pool, resolution and operation. Each cell gives the rate and, below it, its 95% source-cluster bootstrap interval.

		Paragraphs			Topic blocks		
Judge	Evidence pool	Remove	Fill	Rewrite	Remove	Fill	Rewrite
BiomniBench ($N = 21$)							
GPT	GLM	1.4 [0.0, 2.8]	1.0 [0.0, 2.4]	1.4 [0.0, 3.2]	5.0 [0.0, 11.2]	1.6 [0.0, 3.8]	2.0 [0.0, 6.1]
GPT	Deepseek	2.6 [0.2, 5.2]	2.6 [0.0, 5.2]	2.9 [0.0, 6.1]	6.6 [0.0, 14.3]	5.0 [0.0, 10.4]	4.5 [0.0, 9.5]
GLM	GPT	20.8 [12.4, 30.9]	21.2 [12.4, 31.4]	17.8 [9.8, 26.9]	22.1 [10.1, 36.5]	24.0 [12.6, 39.8]	20.6 [8.6, 35.4]
GLM	Deepseek	23.0 [10.2, 40.4]	21.7 [9.8, 37.0]	23.5 [11.3, 39.7]	19.0 [11.1, 28.9]	19.8 [9.9, 30.8]	15.1 [5.4, 26.4]
Deepseek	GPT	6.8 [1.0, 17.2]	7.3 [1.5, 16.7]	5.1 [0.7, 13.2]	14.1 [5.7, 26.5]	12.5 [5.2, 21.8]	14.7 [4.3, 31.4]
Deepseek	GLM	4.1 [1.7, 6.9]	4.2 [1.8, 6.7]	4.2 [1.0, 7.8]	14.1 [5.0, 26.3]	11.9 [4.6, 22.8]	7.3 [1.3, 16.3]
BixBench ($N = 26$)							
GPT	GLM	16.8 [7.4, 27.8]	16.2 [7.4, 26.4]	13.6 [6.0, 23.0]	17.4 [7.1, 28.7]	15.8 [6.9, 26.4]	13.5 [6.3, 21.0]
GPT	Deepseek	11.8 [5.3, 18.8]	11.8 [5.8, 18.4]	11.3 [6.4, 16.8]	18.8 [9.0, 28.8]	16.4 [8.5, 24.8]	17.3 [7.7, 27.5]
GLM	GPT	17.3 [7.6, 27.2]	17.5 [7.4, 27.8]	18.6 [7.6, 29.7]	17.3 [6.3, 28.8]	17.0 [7.0, 27.5]	9.6 [2.6, 17.6]
GLM	Deepseek	17.9 [7.6, 28.9]	17.2 [6.6, 28.7]	17.4 [6.5, 29.1]	17.1 [6.0, 31.9]	18.9 [7.8, 33.0]	18.7 [7.4, 33.3]
Deepseek	GPT	4.3 [0.4, 11.9]	4.0 [0.7, 9.8]	4.6 [0.5, 12.0]	7.1 [2.2, 14.5]	6.0 [1.6, 12.3]	5.7 [0.6, 13.4]
Deepseek	GLM	7.5 [0.4, 15.3]	6.7 [1.5, 12.5]	5.6 [0.0, 12.0]	10.1 [3.0, 18.2]	12.0 [3.0, 22.2]	5.9 [0.0, 13.8]
LABBench2 ($N = 309$)							
GPT	GLM	12.7 [10.2, 15.2]	13.1 [10.5, 15.7]	12.8 [10.2, 15.4]	12.8 [10.5, 15.2]	13.2 [10.8, 15.6]	9.6 [7.3, 12.0]
GPT	Deepseek	13.0 [10.6, 15.4]	12.6 [10.0, 15.2]	12.1 [9.6, 14.7]	13.2 [10.8, 15.6]	14.1 [11.3, 16.9]	12.3 [9.6, 15.1]
GLM	GPT	12.3 [9.8, 15.0]	11.9 [9.5, 14.5]	11.9 [9.4, 14.6]	13.9 [11.8, 16.0]	12.8 [10.7, 15.0]	9.0 [6.9, 11.2]
GLM	Deepseek	10.3 [8.2, 12.5]	10.5 [8.2, 12.8]	10.1 [7.8, 12.5]	14.9 [12.6, 17.3]	15.0 [12.5, 17.6]	10.8 [8.2, 13.4]
Deepseek	GPT	6.5 [4.7, 8.4]	6.6 [4.8, 8.5]	6.0 [4.2, 7.8]	9.5 [7.6, 11.4]	9.8 [8.0, 11.8]	5.6 [3.9, 7.4]
Deepseek	GLM	6.1 [4.5, 7.9]	7.2 [5.4, 9.2]	5.8 [4.1, 7.7]	9.3 [7.4, 11.3]	9.3 [7.4, 11.3]	5.0 [3.4, 6.7]

Notes. The flip rate of a question is the share of its edited passages whose recorded verdict differs from the verdict under the intact input (E^{3+}). Questions are equally weighted, and the brackets contain 95% percentile intervals from 30,000 bootstrap draws that resample source clusters.

to 54.4 pp higher than with the three papers of E^{3-} in the point estimates. Adding papers has no consistent effect, with or without the source paper. At the paragraph resolution, $E^{3-} - E^{1-}$ ranges from -11.5 to $+7.7$ pp and $E^{3+} - E^{1+}$ from -15.4 to $+4.8$ pp. Adding papers to the source paper can lower accuracy, as Section 5 reports.

Editing one source passage changes question-level accuracy by 7.13 to 10.77 pp in absolute value, averaged over configurations, and flips 1.0% to 24.0% of the verdicts, depending on configuration. Between 36% and 68% of the passage edits leave the GPT judge's correct-class probability unchanged, and 7–29% move it by at least 10 pp.

J Where Does Source Dependence Occur?

Flip rates by role label. The flip rate uses the verdicts alone, so its breakdown by role includes all three judges. Figure 14 and Table 12 break the flip rate down by the role label of the edited passage

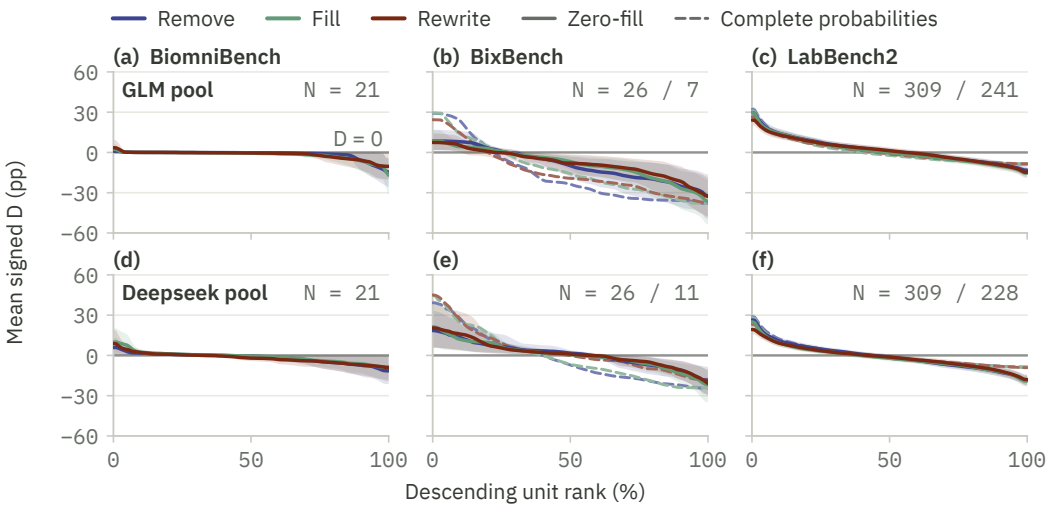


Figure 12: Single-passage effects on the GPT judge's correct-class probability, ranked within each question and averaged at normalized ranks (paragraphs), with D the correct-class probability under E^{3+} minus that under the edited input, in pp. Solid lines set unreported probabilities to zero, with 95% source-cluster bootstrap bands; dashed lines use only questions with complete probabilities (7–11 of 26 on BixBench). N gives the questions for solid / dashed lines.

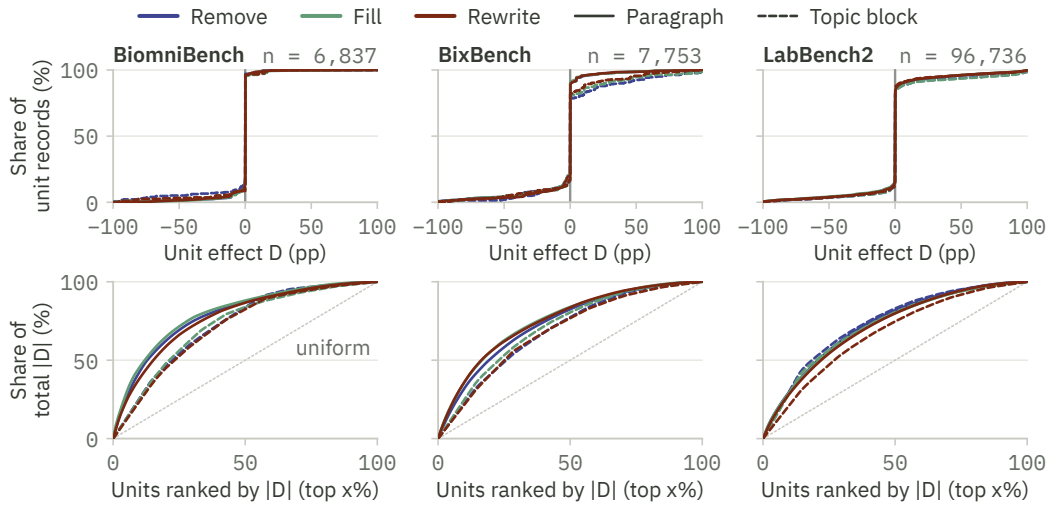


Figure 13: Distribution of single-passage effects on the GPT judge's correct-class probability. Top: empirical distribution function of the signed effect D per operation (solid paragraphs, dashed topic blocks). Bottom: share of the total absolute effect carried by the top x% of a question's passages; the diagonal is the uniform reference. Unreported probabilities are set to zero.

(its module, Appendix E), which derives from the Stage 1 sentence roles. A paragraph takes the majority role of its sentences (a tie goes to the later role in the order background, objective, methods, results, conclusions), and a topic block takes the role from which it was built. These rates pool all edited passages over judges, benchmarks and operations. Edited topic blocks built from results sentences flip 19.1% of verdicts, against 8.9–11.9% for every other role label. The excess appears under REMOVE and FILL (23.8% and 23.7%) and disappears under REWRITE (9.7%). Each judge shows its highest flip rate on these blocks (16.8–22.4%). At the paragraph level the flip rates by role label span 5.8–10.7%.

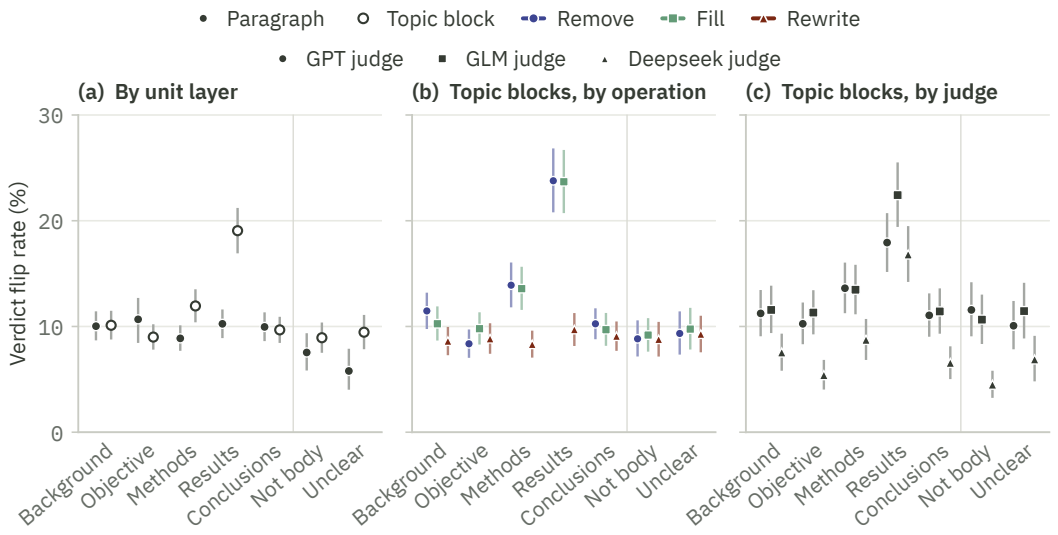


Figure 14: Verdict flip rate of edited passages by role label, all three judges, with 95% source-cluster bootstrap intervals: (a) paragraphs and topic blocks pooled over judges, operations and benchmarks; (b) topic blocks by operation; (c) topic blocks by judge.

Table 12: Verdict flip rate by role label of the edited passage, pooled over the three judges, the three benchmarks and, in the two columns of flip rates, the three operations, with 95% source-cluster bootstrap intervals in brackets. The last three columns split the rate for topic blocks by operation. These are the values behind Figure 14.

Module	Paragraphs		Topic blocks				
	Pairs	Flip rate (%)	Pairs	Flip rate (%)	Remove (%)	Fill (%)	Rewrite (%)
Background	39,976	10.0 [8.7, 11.4]	6,407	10.1 [8.8, 11.5]	11.5	10.3	8.6
Objective	4,206	10.7 [8.4, 12.7]	6,312	9.0 [7.8, 10.2]	8.4	9.8	8.8
Methods	114,819	8.9 [7.7, 10.1]	6,390	11.9 [10.4, 13.5]	13.9	13.6	8.3
Results	71,711	10.2 [8.9, 11.6]	6,396	19.1 [16.9, 21.2]	23.8	23.7	9.7
Conclusions	41,335	10.0 [8.6, 11.3]	6,407	9.7 [8.4, 10.9]	10.3	9.7	9.1
Not body	7,598	7.5 [5.8, 9.4]	5,081	8.9 [7.5, 10.4]	8.8	9.2	8.8
Unclear	9,687	5.8 [4.0, 7.9]	4,980	9.5 [7.8, 11.1]	9.3	9.8	9.3

Notes. Pairs are edits of one passage for one question. Intervals are 95% percentile intervals from the 30,000 source-cluster bootstrap draws of Tables 10 and 11; each draw resamples the source clusters of each benchmark and recomputes the pooled proportion. Module labels derive from the Stage 1 sentence roles: paragraphs by majority vote over sentence-level rhetorical roles, topic blocks by the role used to build the block; on BixBench the pairs come from the Opus5 answer set. Source: the Stage 2 Phase 3 pair records of the edited passages with their module labels.

Conclusion. Differences between roles appear in topic blocks. Blocks built from results sentences flip 19.1% of verdicts, against 8.9–11.9% for the other role labels, and the excess appears under REMOVE and FILL and disappears under REWRITE. At the paragraph level the flip rates by role label span 5.8–10.7%.

K How Do Model and Representation Choices Affect the Audit?

Contrasts between judges, pools and resolutions. In the point estimates, the choice of judge moves the flip rate more than the choice of evidence pool. Each of the three models serves both as a judge and as the builder of an evidence pool. A difference between two models is measured as a judge contrast at a fixed pool and as a pool contrast at a fixed judge, under both resolutions. Across

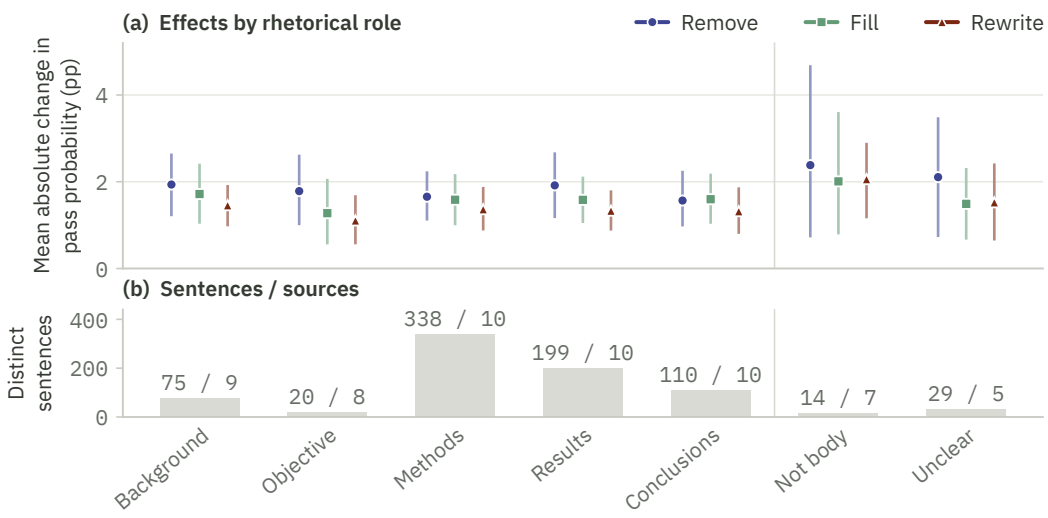


Figure 15: Sentence-level pilot on BiomniBench (Qwen judge, GPT evidence pool). (a) Mean absolute change in pass probability when one sentence is edited, by rhetorical role and operation, with 95% source-cluster bootstrap intervals. (b) Number of distinct sentences and source papers per role.

the three benchmarks and the two resolutions, the judge contrasts range from -19.53 to $+11.24$ pp and the pool contrasts from -2.71 to $+6.44$ pp; on LABBench2 the pool contrasts stay between -0.83 and $+1.90$ pp. Moving from paragraphs to topic blocks changes the size of the judge contrasts, and seven of the nine keep their sign. On LABBench2, GPT minus Deepseek with the GLM pool is $+6.42$ pp for paragraphs and $+2.96$ pp for topic blocks, and GLM minus Deepseek with the GPT pool is $+5.39$ and $+3.73$ pp. GPT minus GLM with the Deepseek pool changes sign, from $+2.65$ to -1.31 pp in the point estimates.

The same questions under both resolutions. Two scatter plots of question-level flip rates show where the aggregate contrasts come from. Figure 17 fixes the GLM evidence pool and plots, for each question, the flip rate of GPT against that of Deepseek, separately for paragraphs and topic blocks. Each judge's flip rate is measured against its own intact verdict. The counts in the panels are numbers of questions, and the aggregate contrasts above weight source clusters equally. On LABBench2, GPT has the higher rate on 121 questions for paragraphs and 76 for topic blocks, Deepseek has the higher rate on 50 and 58, and the rates are equal on 138 and 175 questions. For topic blocks the positive contrast of GPT minus Deepseek is smaller, with fewer questions on which GPT has the higher rate and more questions with equal rates. On BixBench, the questions on which GPT has the higher rate fall from 14 to 7 between the resolutions, and those on which Deepseek has the higher rate rise from 1 to 5. On BiomniBench, GPT has the higher rate on 3 questions and Deepseek on 7 for paragraphs, and on 3 and 6 questions for topic blocks.

Figure 18 fixes the same GLM pool and compares the two resolutions question by question, separately for GPT and Deepseek. For GPT on LABBench2, topic blocks raise the rate on 88 questions and lower it on 89, with 132 unchanged. The upward and downward contributions are 6.96 and 7.41 pp and offset each other to -0.45 pp overall. For Deepseek, topic blocks raise the rate on 88 questions and lower it on 46, with 175 unchanged, so the rises outnumber the falls and the net change is $+3.01$ pp.

Opposite responses of two judges to a change of resolution. With the GLM pool, moving from paragraphs to topic blocks raises the flip rate of one of GPT and Deepseek and lowers that of the other on 45/309 LABBench2 questions and 4/26 BixBench questions. Because topic blocks regroup the sentences of the source paper, the intact input, and with it the intact verdict, can also differ between the resolutions. Among the questions on which each judge's intact verdict is the same at both resolutions, the opposite responses occur on 28/238 LABBench2 and 1/19 BixBench questions.

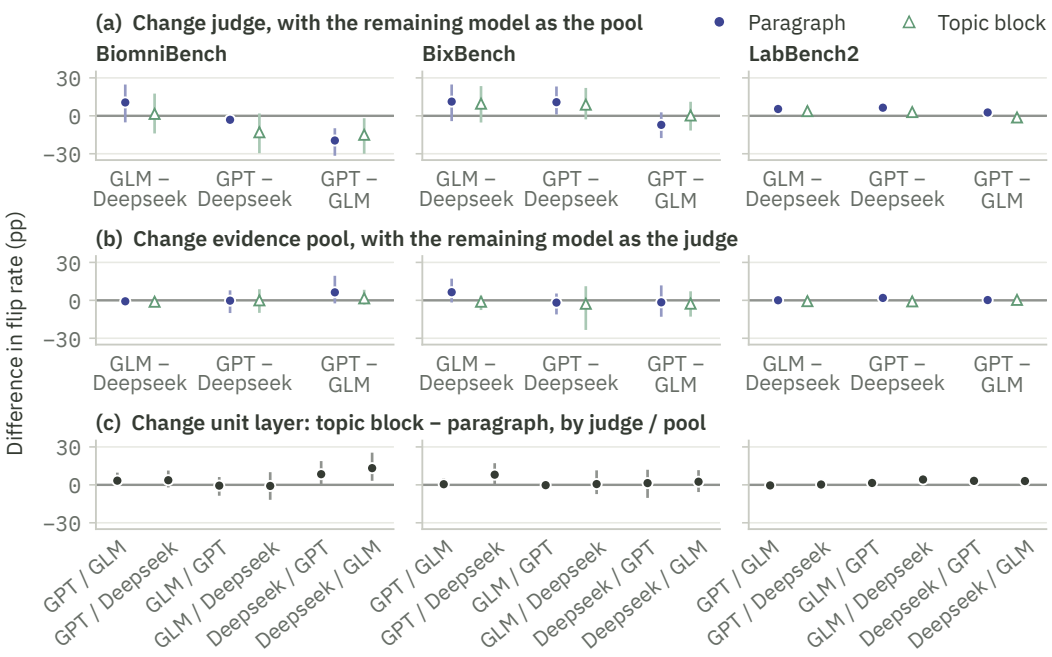


Figure 16: Paired differences in flip rate under REMOVE for the three audit choices: (a) judge contrasts at a fixed evidence pool, (b) pool contrasts at a fixed judge, (c) topic blocks minus paragraphs at a fixed judge and pool. Columns are benchmarks; bars are 95% source-cluster bootstrap intervals.

For each direction of GPT's change on LABBench2 and for BixBench, Table 13 lists the question among these on which the smaller of the two judges' absolute changes is largest.

Table 13: Matched questions on which the resolution moves GPT and Deepseek in opposite directions (GLM pool). Cells show changed verdicts / removal edits for paragraphs and topic blocks; each judge has the same intact verdict at both resolutions.

		GPT		Deepseek	
Benchmark	Question ID prefix	Paragraph	Topic block	Paragraph	Topic block
LABBench2	`6f0bb5c8`	0/69	2/7	14/69	0/7
LABBench2	`0aca188b`	9/32	0/6	0/32	2/6
BixBench	`439ece6b`	10/17	1/6	3/17	2/6

Notes. Model names: GPT = gpt-5.6-luna, GLM = glm-5.3-flash, Deepseek = deepseek-v4-flash.

Passage-level agreement between judges. Figure 19 compares the passages that flip each judge. Each passage, paragraph or topic block, is described by its flip profile, the flip rates of the three judges under each of the three operations, and the profiles are grouped with k-means into four groups. One group, in which each judge's mean flip rate is below 5%, holds 63% of the 14,892 paragraphs and topic blocks that were edited under every judge and operation. Each judge has a group of its own (14% of these passages for GPT, 16% for GLM and 8% for Deepseek), and no group collects passages that flip several judges. When one judge flips on an edit, another judge flips on the same edit at 0.0 to 3.7 times its base rate, depending on benchmark, resolution and pair of judges. In the combinations where this ratio exceeds 1, the judges flip together more often than chance.

Conclusion. In the point estimates, the judge moves the flip rate more than the evidence pool does. Across the three benchmarks and both resolutions, the judge contrasts span -19.53 to $+11.24$ pp and the pool contrasts -2.71 to $+6.44$ pp. The resolution changes the size of the judge contrasts, and seven of the nine keep their sign. On LABBench2 with the GLM pool, the question-level rises

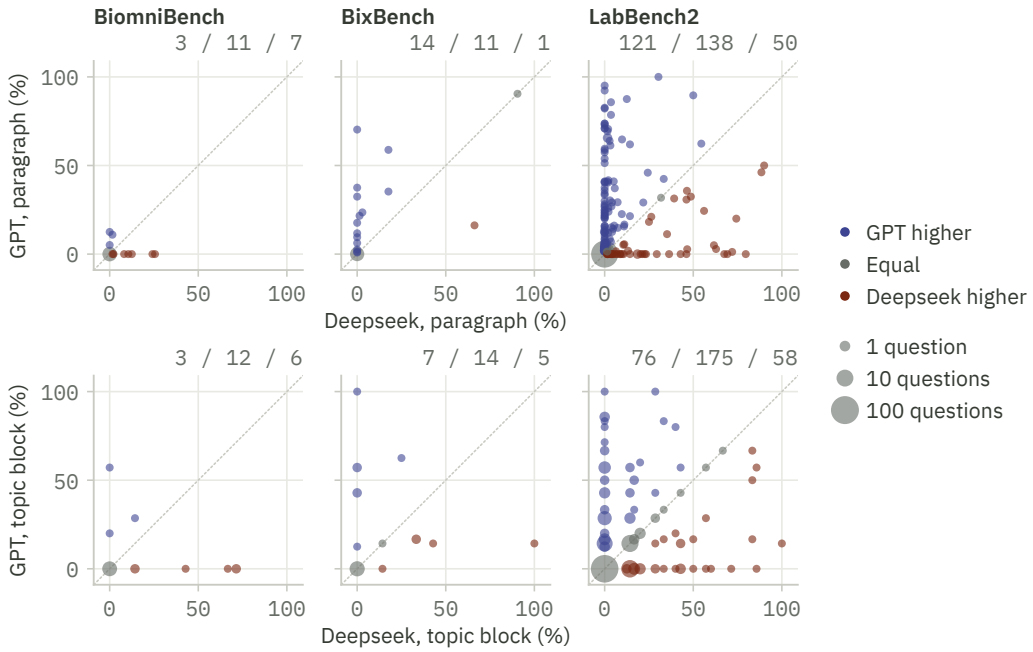


Figure 17: Question-level flip rates under REMOVE at the fixed GLM pool, GPT judge against Deepseek judge, for paragraphs (top) and topic blocks (bottom). Marker area grows with the number of questions at the same pair of values; the counts give the questions above, on and below the diagonal.

and falls between the resolutions offset each other for GPT (-0.45 pp overall), and for Deepseek the rises outnumber the falls ($+3.01$ pp). Each judge has its own group of passages that flip it (8–16% of the passages edited under every judge and operation), so an audit of one judge finds passages that an audit of another judge misses.

L How Does Source Dependence Relate to Grading Agreement?

This section relates the passage-level effects of REMOVE to the agreement of the verdicts with the correctness labels of Appendix E. Agreement is the verdict accuracy of Section 5, with an abstention counted as disagreement.

Gains and losses of agreement. The change in agreement after an edit decomposes as

$$\Delta A = P(\text{noncorrect} \to \text{correct}) - P(\text{correct} \to \text{noncorrect}). \tag{10}$$

The change ΔA is the edited value minus the intact value, so a positive ΔA means that the edit raised agreement; the effect D of Figure 12 is the intact value minus the edited value. Figure 20 splits ΔA under REMOVE into gains, in which a wrong verdict or an abstention becomes correct, and losses, in which a correct verdict becomes wrong or turns into an abstention. Removal lowers agreement in 24 of the 36 combinations of benchmark, judge, evidence pool and resolution: in 10 of the 12 on BixBench, 10 of the 12 on LABBench2 and 4 of the 12 on BiomniBench, where it raises agreement for the GPT and Deepseek judges. The change from a correct verdict to an abstention is the largest loss in 22 of the 36 combinations, and in 24 of them more than half of the edits that change agreement move a verdict into or out of an abstention.

Correct and incorrect candidates. Figure 21 splits the change in agreement after REMOVE by the correctness label of the candidate answer. Each group keeps the question weights of the full question set, rescaled to sum to one within the group, so the two group effects, weighted by the groups' shares of the weights, add up to the overall effect. The intervals resample whole source clusters for both groups together. Removal lowers agreement for the incorrect candidates in 24 of the

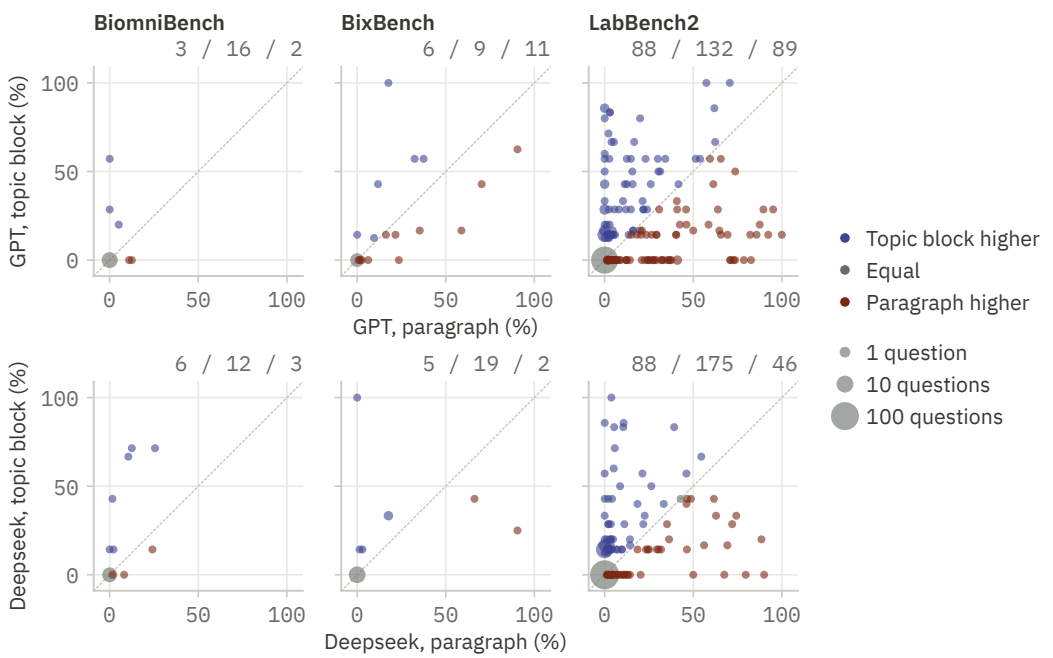


Figure 18: Question-level flip rates under REMOVE at the fixed GLM pool, topic blocks against paragraphs, for the GPT judge (top) and the Deepseek judge (bottom); marker area and counts as in Figure 17.

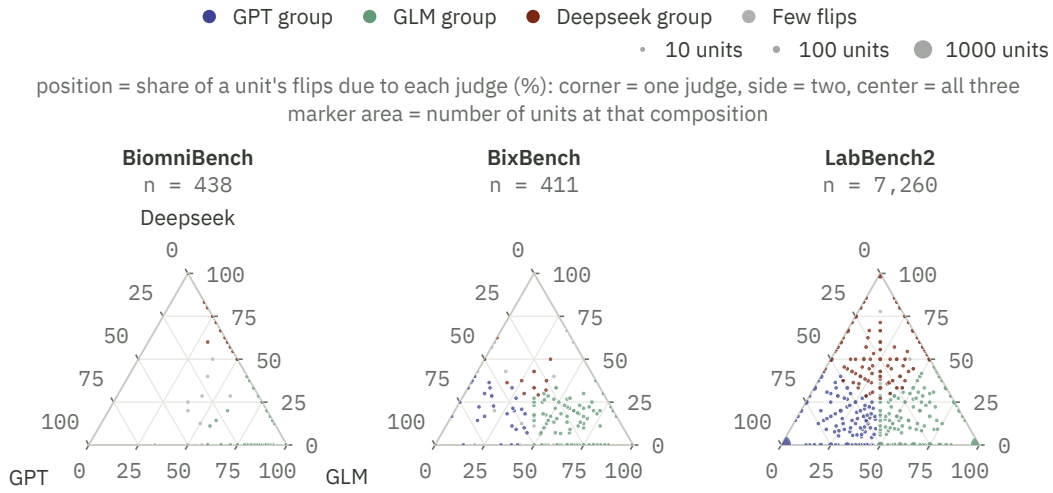


Figure 19: Share of each passage's verdict flips that occurred under each judge, for the n passages that flip at least once (a corner: one judge only; the center: all three equally). Marker area is the number of passages at a position, and color is the k-means group of the passages' flip profiles (four groups).

36 combinations and for the correct candidates in 21, and the interval of the difference between the two groups includes zero in 31 of the 36.

Paragraphs and topic blocks at a fixed intact verdict. Table 14 compares the two resolutions on the LABBench2 questions whose intact verdict is the same for paragraphs and for topic blocks, so the comparison holds the intact verdict fixed. Editing a topic block instead of a paragraph raises the flip rate in all six configurations of judge and pool, by 0.49 to 4.32 pp, and lowers the change in

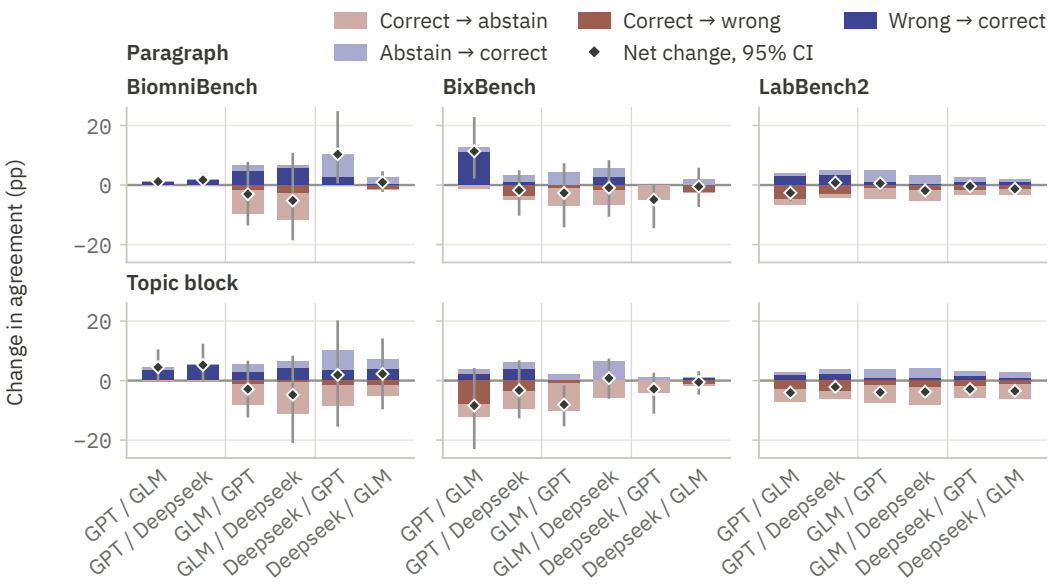


Figure 20: Change in grading agreement after REMOVE, split into gains above zero (wrong or abstained verdicts becoming correct) and losses below zero (correct verdicts becoming wrong or abstained). Diamonds are net changes with 95% source-cluster bootstrap intervals; horizontal positions are judge and evidence pool, rows resolutions, columns benchmarks.

agreement in five of them, by up to 4.33 pp. For the GLM and Deepseek judges both differences have intervals that exclude zero, and for the GPT judge the intervals of the difference in agreement include zero. A topic block holds all sentences of one rhetorical role and is more than five times as long as a paragraph on LABBench2 (medians of 695 and 126 word runs, Table 3), and at a fixed intact verdict its removal lowers agreement more for the GLM and Deepseek judges.

Table 14: Paired differences of topic blocks minus paragraphs on LABBench2. Values are pp with pointwise 95% source-cluster bootstrap intervals. The last column is the change in agreement for topic blocks minus that for paragraphs.

Judge	Evidence pool	Questions	Flip rate (pp)	Change in agreement (pp)
GPT	GLM	252	1.77 [0.06, 3.58]	−1.42 [−2.98, 0.11]
GPT	Deepseek	247	0.49 [−1.72, 2.71]	0.00 [−2.08, 2.14]
GLM	GPT	274	4.32 [2.98, 5.68]	−4.33 [−5.58, −3.07]
GLM	Deepseek	275	4.15 [2.85, 5.44]	−3.46 [−4.73, −2.16]
Deepseek	GPT	277	3.21 [1.65, 4.78]	−2.84 [−4.30, −1.35]
Deepseek	GLM	285	4.28 [2.54, 6.14]	−3.78 [−5.66, −2.04]

Notes. Questions is the number of questions whose intact verdict is the same at both resolutions. Each configuration uses only these questions, renormalizes the equal source-cluster weights to them, and edits the passages of each resolution. Model names: GPT = gpt-5.6-luna, GLM = glm-5.3-flash, Deepseek = deepseek-v4-flash.

Size of the dependence and intact agreement. Figure 22 relates the size of the dependence of each source cluster to its intact agreement. For the GPT judge under REMOVE, the size of the dependence is the absolute change in the correct-class probability, averaged over the passages that have a valid edit under all three operations, then over the pools of a question and then over the questions of the cluster; intact agreement uses the same questions. The Spearman correlations across clusters are -0.38, $+0.45$ and -0.18 for paragraphs on BiomniBench, BixBench and LABBench2, and -0.83, $+0.58$ and -0.07 for topic blocks, with paired bootstrap intervals from 2,000 draws. Two of the six intervals exclude zero, for topic blocks on BiomniBench ($[-0.97, -0.35]$) and for paragraphs on

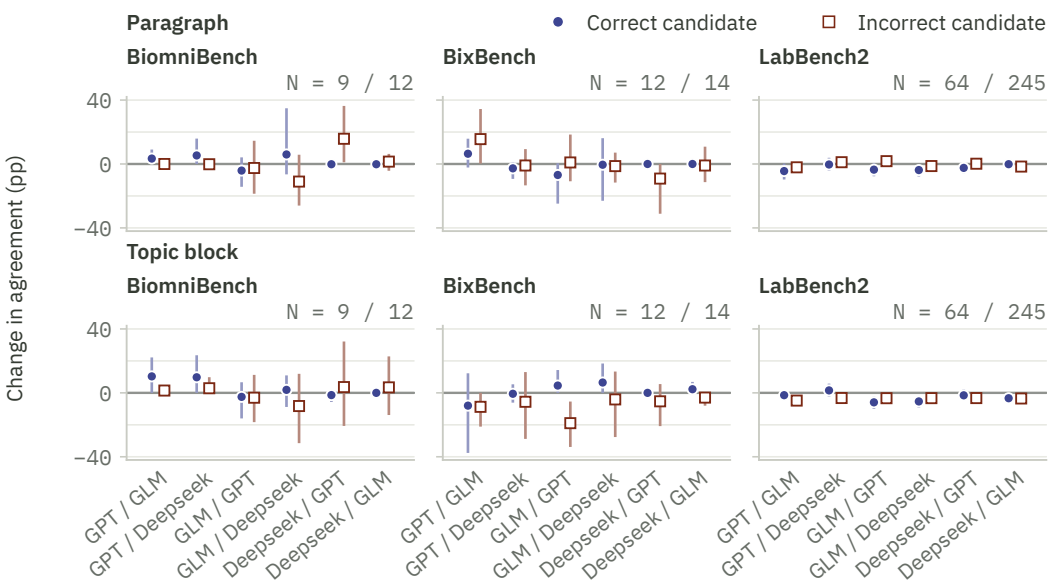


Figure 21: Change in grading agreement after REMOVE for correct candidates (filled circles) and incorrect candidates (hollow squares), by judge and evidence pool, resolution (rows) and benchmark (columns), with 95% source-cluster bootstrap intervals. Positive values mean improved agreement. N gives the correct / incorrect candidates.

LABBench2 ($[-0.31, -0.05]$). The sign differs between benchmarks, so the intact agreement of a cluster does not show how strongly its verdicts depend on the source paper.

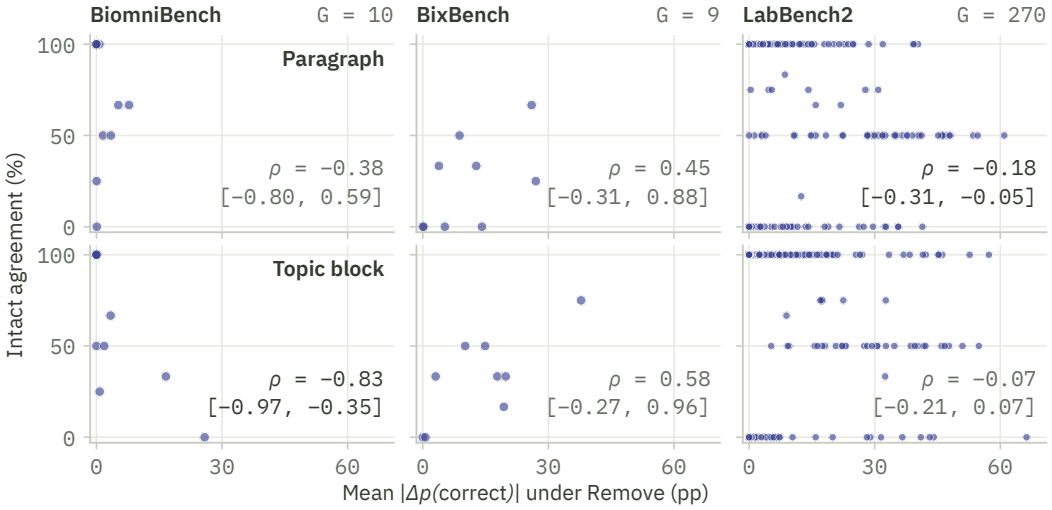


Figure 22: Source dependence against intact grading agreement at the level of source clusters (GPT judge, REMOVE): mean absolute change in correct-class probability (horizontal) against agreement with the stored labels (vertical), one point per cluster. Each panel gives the Spearman correlation and its 95% paired bootstrap interval. G is the number of source clusters; rows are resolutions.

L.1 DEPENDENCE OF GRADING CONSEQUENCES ON THE CANDIDATE ANSWERS

This subsection grades a second set of answers to the same 26 BixBench questions and nine source clusters. The Slim answer set holds zero-shot GPT-4o answers to the open-ended questions, and the

Opus5 answer set holds the answers of the Claude Opus 5 agent that the main runs grade (Appendix E). The Slim answer set contains 1 correct answer and the Opus5 answer set 12. On 14 questions the answers are incorrect in both sets, on 11 the answer changes from incorrect to correct, and one question has a correct answer in both sets. Questions are paired within each configuration of judge and pool and each resolution. The removal effect of a question averages all its removal edits at that resolution. Questions are then averaged within source clusters, and clusters are weighted equally, so these flip rates differ from those of Table 11, which weight questions equally. The 95% pointwise intervals use 10,000 paired source-cluster bootstrap draws.

Flip rate and change in agreement. Figure 23 gives the paired differences between the answer sets, Opus5 minus Slim, and Figure 24 gives the values of each set for paragraphs. With nine source clusters the intervals are wide. All 12 differences in the flip rate have intervals that include zero. The differences in the change in agreement range from -13.3 to $+18.8$ pp, and one of the 12 has an interval that excludes zero: for the GPT judge with the GLM pool on paragraphs, removal changes agreement by -7.49 pp for Slim and by $+11.33$ pp for Opus5, a difference of $+18.83$ pp (95% CI [7.86, 30.44]).

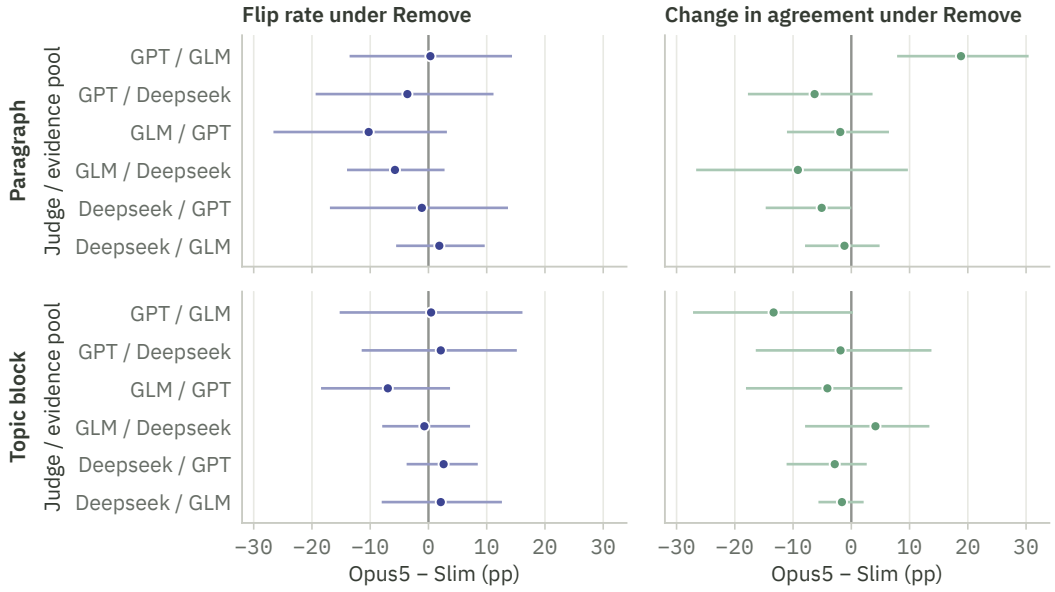


Figure 23: Paired contrasts between the two answer sets under REMOVE on BixBench, Opus5 minus Slim, for the six configurations of judge and evidence pool and both resolutions (rows). The left column gives the difference in the flip rate and the right column the difference in the change in agreement after removal, with 95% paired source-cluster bootstrap intervals.

Correctness transitions between the answer sets. Figure 25 splits the 25 questions whose answer changes into the 14 whose answer is incorrect in both sets and the 11 whose answer changes from incorrect to correct. The six configurations of judge and pool are averaged within each question, and the cluster and question weights are renormalized within each group. Two of the 12 paired differences have intervals that exclude zero, both for topic blocks. On the 14 questions with incorrect answers in both sets, the change in agreement after removal is 8.79 pp lower under Opus5 (95% CI $[-16.79, -2.09]$). On the 11 questions whose answer becomes correct, intact agreement is 12.12 pp lower under Opus5 (95% CI $[-26.19, -4.17]$). The differences in the flip rate include zero in both groups and at both resolutions.

Conclusion. Removing a passage of the source paper lowers agreement with the correctness labels in 24 of the 36 combinations of benchmark, judge, evidence pool and resolution, 20 of them among the 24 combinations on BixBench and LABBench2, so the content of the source paper helps the judges reach correct verdicts at the passage level, as the comparisons of Section 5.2 show for whole papers. As in Table 1, abstentions carry much of this effect: in 24 of the 36 combinations, more

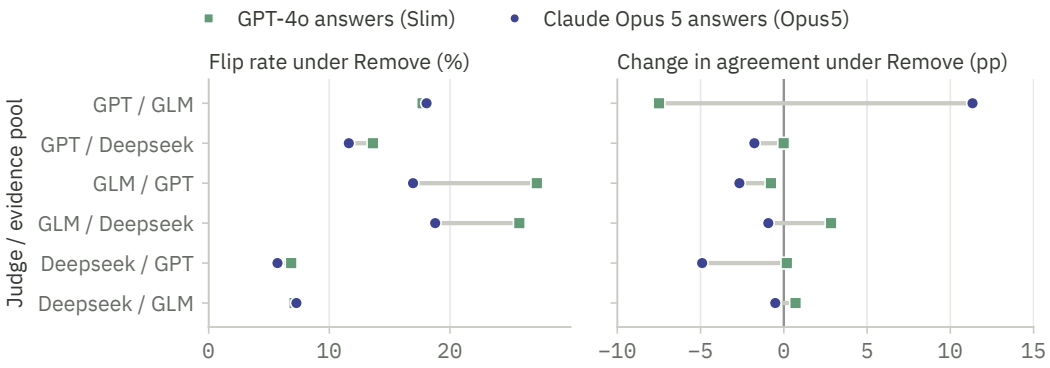


Figure 24: Flip rate (left) and change in agreement (right) under REMOVE for each judge and evidence pool on BixBench, paragraphs, under the two answer sets (squares: Slim, GPT-4o answers; circles: Opus5, Claude Opus 5 answers). A line joins the two answer sets of one configuration. Point estimates; Figure 23 gives the paired differences with 95% intervals.

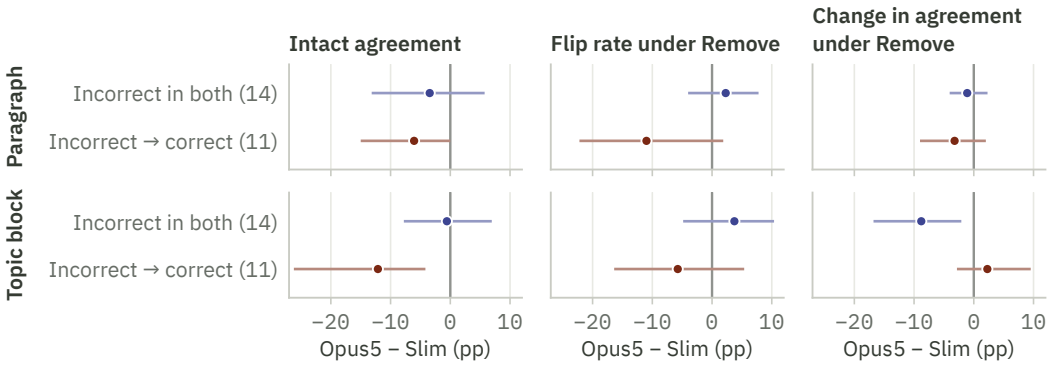


Figure 25: Contrasts between the two answer sets for the 25 BixBench questions whose answer changed between Slim and Opus5, grouped by correctness transition. The figure gives the paired differences (Opus5 minus Slim) in intact agreement, flip rate under REMOVE and change in agreement after removal, with 95% paired source-cluster bootstrap intervals.

than half of the edits that change agreement move a verdict into or out of an abstention. Removal lowers agreement for correct and for incorrect candidates, removing a topic block lowers it more than removing a paragraph for the GLM and Deepseek judges, and with a second answer set for the same BixBench questions all 12 differences in the flip rate have intervals that include zero.