PREDICTABLE LABELS, FRAGILE RANKINGS: AN EMPIRICAL STUDY OF LLM EVALUATOR PROCEDURES

Anonymous authors
Paper under double-blind review

ABSTRACT

Automated hypothesis generation creates a selection problem: which candidates should proceed to expert review, simulation, or experiment? Benchmarks assess evaluators through checkable labels, but a label's predictability and an evaluator's ability to recover it are different properties. We examine both on ResearchBench, whose fixed candidate pools link one hypothesis to a designated source study. Across 1,313 eligible items, a supervised cross-fitted candidate-only classifier achieves 98.40% sixteen-way Hit@1 without the background or research question. On the same pools, six zero-shot LLM evaluators average 55.79% source–alternative pairwise recovery when scoring candidates separately (Isolated), but 35.99% when generating their scores together (Vector), below the 50% pairwise reference. These endpoints diagnose label signal and procedure behavior under different training conditions; they are not a common ability ranking. Paired behavioral analysis shows that similar strict-win rates can conceal substantial exchanges among wins, ties, and losses. Controlled comparisons further show that replacing surrounding material can reduce recovery while preserving the target text and scalar output action, with effects that depend on the model and setting. With visible materials matched, it remains unresolved whether joint rather than separate scoring increases or reduces this loss. The evidence supports evaluating label meaning, candidate discrimination, and recovery in the intended context separately before treating benchmark performance as evidence of reliable scientific screening.

1 INTRODUCTION

Scientific hypothesis generation creates a selection problem: which ideas should proceed to expert review, simulation, or experiment? Systems such as The AI Scientist connect idea generation to computational testing and assessment (Lu et al., 2024). When every proposal cannot receive follow-up, evaluators influence which ideas are pursued. Their judgments must therefore recover a relation that supports the intended selection.

Scientific value is difficult to label at scale: novelty, truth, and usefulness require different evidence. Benchmarks can instead define a checkable proxy. ResearchBench pairs a background and research question with a hypothesis derived from an associated source study and fifteen constructed alternatives (Liu et al., 2026). Source identification is thus precise, but it is not an independent assessment of scientific truth or quality. A benchmark score invites two questions: what predicts the label, and what does the evaluator recover?

Existing work establishes that evaluation depends on prompt formatting (Sclar et al., 2024), feedback protocols and choice sets (Bansal et al., 2024; Tripathi et al., 2025), and candidate order (Tang et al., 2024). TrustJudge documents inconsistencies between scores and comparative judgments (Wang et al., 2026). In scientific hypothesis ranking, ResearchBench already reports some below-chance results, while Rajwal et al. (2026) compare logit-based energy scoring with prompted ranking. Separately, hypothesis-only baselines show how candidate text can predict benchmark labels without the context needed for the intended task (Gururangan et al., 2018; Poliak et al., 2018). These precedents motivate a more specific investigation: how do label predictability, the relations recovered by

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.

an evaluator, and the effects of candidate context fit together within the same source-identification task?

ResearchBench's fixed pools, explicit source relation, and twelve disciplines let us examine label signal and evaluator behavior on the same objects. We compare *evaluator procedures*: how candidates are presented, what output is requested, and how it is read and aggregated. *Isolated* scores one candidate per call; *Vector* requests all sixteen scores together. The model, pool, and source relation stay fixed. Does a change in procedure preserve recovery of that unchanged relation?

A concrete example shows the consequence for selection. On the real item in Figure 2, Gemma2-27B places the source in a five-way tie for first under Isolated, but in the lower half of the ranking under every Vector order. The same source can thus remain eligible for a short list or be discarded, depending on the evaluation procedure. This post hoc illustration shows observed rank blocks; the task anatomy in Figure 1 is a separate authored schematic.

We establish three connected findings. First, strong candidate-only source prediction coexists with below-chance generative recovery on the same pools, separating label signal from actual recovery. Second, paired behavior reveals why similar strict-win rates can mask different judgments. Direct choices show that poor recovery extends beyond score conversion, and one fixed aggregation rule further lowers it. Third, context can affect recovery with target text and scalar output fixed. With matched visibility, the comparison limits large changes in this loss between separate and joint scoring, without confirming which loses more. Pool-size, model, and external-task comparisons delimit these findings. Together, the evidence tests what supports a selection procedure without assuming a shared internal mechanism.

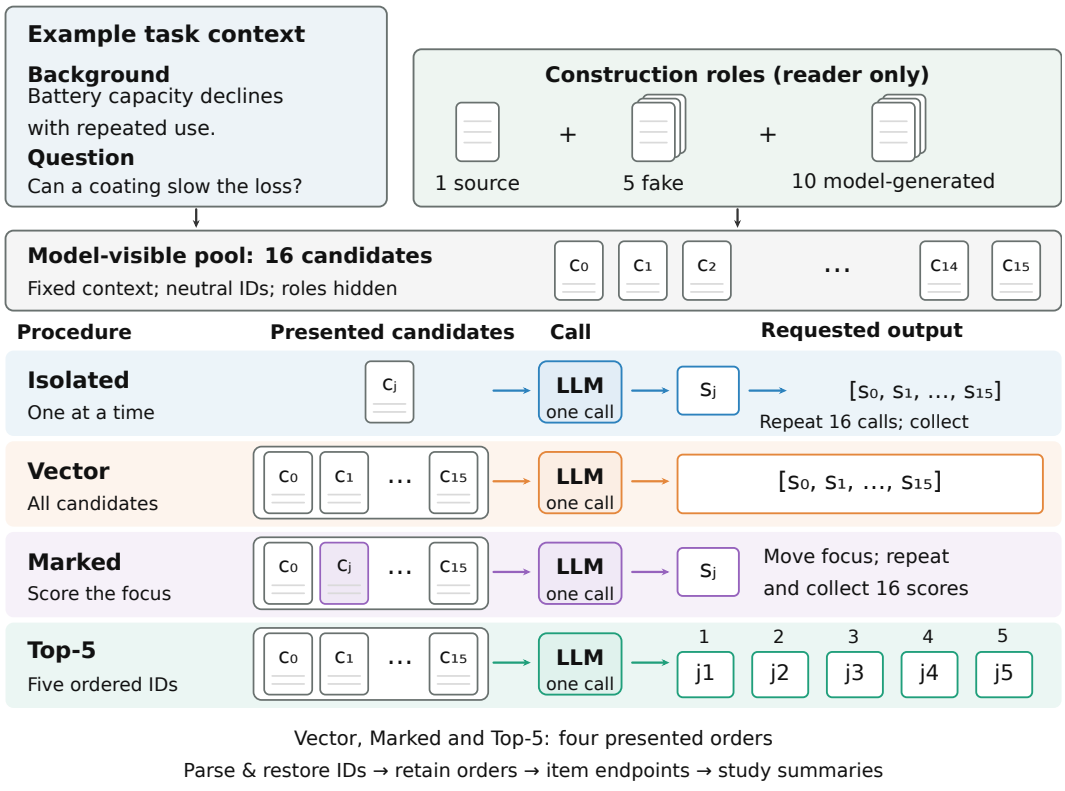


Figure 1: One task, four evaluator procedures. The item text is an authored schematic, not a ResearchBench record. Construction roles are shown only to the reader; the model sees neutral IDs. The lower chain corresponds to Equations 1–3.

2 TASK AND EVALUATOR PROCEDURES

2.1 TASK, POPULATIONS, AND MATCHED COMPARISONS

Each ResearchBench ranking pool contains one source hypothesis, five alternatives constructed from negative inspirations (the release's *fake* group), and ten from subsets of source inspirations (the *model* group). Construction roles are hidden from the evaluator. Displayed IDs are reassigned after permutation and restored to stable identities for scoring.

The benchmark-wide comparison, or *census*, measures SOURCE Isolated and Vector on 1,313 eligible items in 960 source clusters, after ten outcome-independent common-context exclusions from the 1,323-item release. Detailed procedures, Direct, candidate-count, contemporary-model, and scalar-context diagnostics share 237 items in 224 dependency groups. Shared targets allow these diagnostics to compare behavior on the same items. These complement a 24-item development set within a deterministically allocated, discipline-stratified pool. The dual-focus comparison uses 199 common-complete targets from other dependency groups, already observed in the census and as context-study donors. Appendices A.1 and F.6 detail selection, reuse, and the planned-200 design.

The core panel contains the largest tested checkpoint in each of six families: Qwen2.5-32B, Falcon3-10B, Gemma2-27B, Granite3.3-8B, Llama3.2-3B, and Mistral-Small-24B. The nineteen-checkpoint development comparison and later hosted endpoints have distinct roles (Appendix A.2). Open-weight comparisons fix the recorded checkpoint revision, precision, and decoding. Hosted comparisons fix the endpoint alias and configurable parameters; immutable provider weights and internal precision are unavailable controls.

Isolated requests one integer in $[0, 10000]$ per candidate. Marked shows the entire pool but requests only the current focal candidate's score. Vector requests all scores, and Top-5 requests five ordered IDs. All-visible procedures use four predetermined orders. SOURCE asks which hypothesis belongs to the associated study. PLAUSIBILITY asks how plausible and supported a candidate is, while retaining source identity as the scoring label; it diagnoses a target mismatch. Direct requests an A/B choice between the source and each alternative in both positions. Presentation, action, output cap, and readout together define a procedure (Appendices A.3 and G).

2.2 FROM A PRESENTATION TO AN ESTIMATE

We parse each presentation's output and restore candidate identities, calculate the item endpoint, then aggregate within the study. Individual orders remain available for stability. For study s, item i, model m, procedure σ, and scheduled presentation q, the parsed output is

$$U_{sim\sigma q} = R_{s,\sigma}(J_{m,d_s}(P_{s,\sigma,q}(i))), \qquad q \in Q_{si\sigma}. \tag{1}$$

P constructs the requested call or call set and order; J applies the model under decoding policy d_s; R is the study's readout. A presentation may require several candidate-specific calls. U retains identities, scores or choices, validity, and the order mapping.

The endpoint-specific item quantity is

$$z^{(e)}_{sim}(\sigma) = \Psi_{s,\sigma,e}(\{(q, U_{sim\sigma q}) : q \in Q_{si\sigma}\}, g_{si}), \tag{2}$$

where g_{si} is the designated label or relation. Ψ applies the endpoint's tie and invalid-output rules and its specified aggregation. Recovery averages presentation-level outcomes. Order stability compares retained permutations, including all six unordered pairs when four orders are used; averaging scores first would destroy that information.

The study-level generative-evaluation estimate is

$$\widehat{\mu}^{(e)}_s(\sigma) = \sum_{m \in \mathcal{M}_s} w_{sm} \sum_{i \in \mathcal{I}_{sm}} v_{si|m}\, z^{(e)}_{sim}(\sigma), \qquad \sum_m w_{sm} = \sum_i v_{si|m} = 1. \tag{3}$$

The census weights its 1,313 items equally within each model, then averages the six models equally. Other populations and weights are specified in Appendix B.2. Supervised cross-validation and macro held-out-discipline validation have separate definitions (Appendix C.2).

Endpoints and readout. Hit@1 and Hit@5 measure source retention when one or five candidates can be kept, with fractional credit when a score tie crosses the cutoff. Pairwise recovery averages the fifteen source–alternative relations, assigning half credit to a tie. Their uniform-random references are 6.25%, 31.25%, and 50%, respectively. A Top-5 output cannot define a complete pairwise relation. Direct measures exact choice accuracy. An unrecoverable required output gives the affected presentation zero credit and stays in the denominator.

For these comparisons, *semantic readout* recovers requested scores or choices from the schema and a narrow set of representational wrappers, using the output and displayed mapping without the gold label. Validity means parsability, not scientific correctness. The policy was selected after some study-specific outputs and results were available; Appendix B.1 states its timing and sensitivity. The controlled context studies instead use strict JSON readout. Intervals follow each study's sampling unit and test family (Appendix B.2).

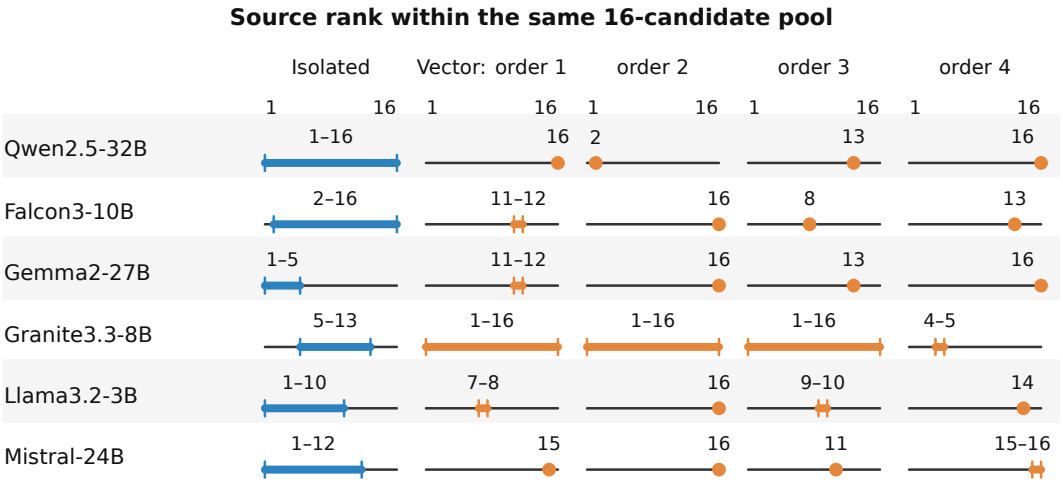


Figure 2: Source ranks on one illustrative real item. Each cell shows an occupied rank or tied rank block, from 1 to 16. A segment denotes a score tie, not a confidence interval. Isolated has one assembled relation; Vector has four orders. All six models are shown, including Granite's different behavior. This text-free post hoc example is distinct from the schematic in Figure 1.

3 Label Signal and Actual Source Recovery

We first ask two questions about the same candidate pools: does an evaluator recover the designated source, and can the label be predicted from candidate text alone? The first tests a specified evaluation procedure. The second checks how much apparent benchmark success can be obtained without using the background or question.

3.1 A Procedure Change Reverses Average Pairwise Recovery

Isolated and Vector fall on opposite sides of the pairwise chance reference in the six-model census mean (Figure 3). Recovery is 55.79% under Isolated and 35.99% under Vector. The paired difference is 19.80 pp (95% item interval [18.83, 20.77]; source-cluster sensitivity [18.68, 20.96]). The three primary pairwise tests meet the specified Holm criterion. Every checkpoint has a positive I−V gap and a Vector estimate below 50%; Isolated is not uniformly above chance (Appendix C.1).

The loss affects more than first-place selection. Vector's tie-aware Hit@1 is 5.23%, with its marginal 95% interval below the 6.25% reference. Hit@5 is 20.51%, against 31.25% chance. The readout recovers 99.95% and 99.25% of Isolated and Vector calls. The 299 unrecoverable calls retain zero credit; a high parsing rate therefore coexists with low source recovery.

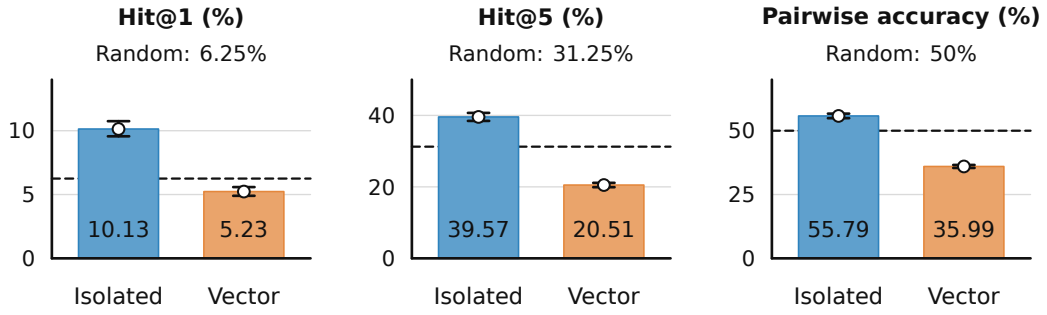


Figure 3: SOURCE recovery on all 1,313 common-support items, averaged equally over six fixed models. Bars start at zero; whiskers are 95% item-bootstrap intervals. Dashed lines mark the endpoint-specific random references. Vertical scales differ. Only Isolated and Vector cover the complete set.

3.2 Source labels are highly predictable from candidate text

The same pools contain a strong source-label signal even without the scientific context. A regularized linear classifier combines 23 surface features with word and character TF-IDF. Five outer and four inner grouped folds isolate source and duplicate dependencies; vocabulary, scaling, and regularization selection use training folds only. The combined model, the primary diagnostic, recovers 1,292/1,313 sources: 98.40% sixteen-way Hit@1 (95% CI [97.71, 99.03]%), against 6.25% chance. It uses source labels during training and candidate text at evaluation, with no background, question, position, or source metadata.

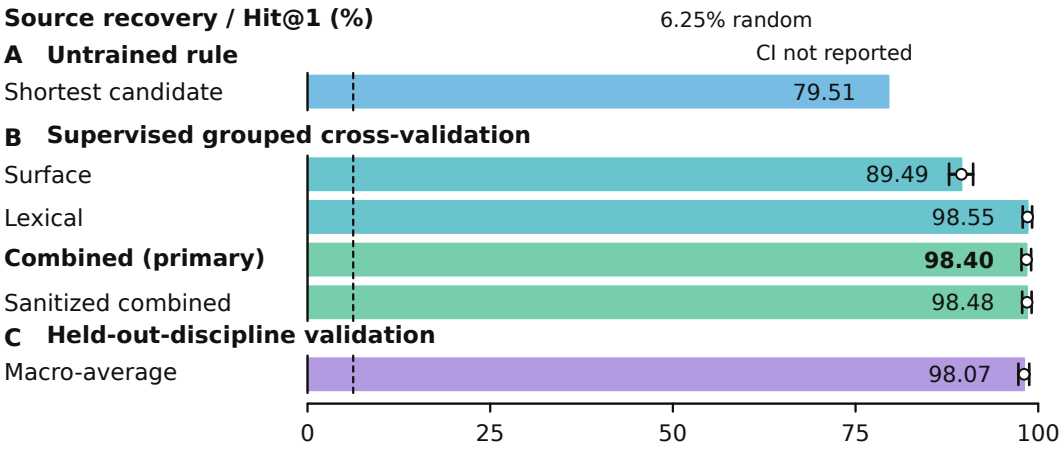


Figure 4: Candidate-only source recovery on the same sixteen-way pools. Untrained, supervised grouped cross-validation, and macro held-out-discipline results are separated. Bars start at zero; whiskers are 95% source-cluster intervals conditional on fixed predictions. No interval was reported for shortest. Background and question are withheld from all features.

The signal persists across feature choices. The untrained shortest candidate rule reaches 79.51%; surface-only and lexical-only models reach 89.49% and 98.55%. A fixed transformation masking specified identifiers and citation forms gives 98.48%. Held-out-discipline macro accuracy is 98.07%,

a distinct validation estimand (Appendix C.2). The combined model is the specified primary diagnostic, not a post hoc choice of the highest value.

High candidate-only recovery changes the interpretation of benchmark success: a high score alone cannot establish that the background or question was used. Low generative recovery, meanwhile, occurs despite predictable label information in the pool. The supervised classifier and zero-shot evaluators differ in training and endpoints, so their percentages are not an ability leaderboard. Nor does classifier recovery identify the cues used by an LLM or explain its reversal. The next question is what the observed evaluator outputs reveal about the gap.

4 What average recovery conceals

Similar win rates conceal changed relations. The 19.80 pp gap is not a 19.80 pp decline in strict wins. A strict win (W) means the source scores above an alternative; a tie (T) receives half credit, while a loss (L) or invalid relation (U) receives zero. Strict wins account for 29.42% of relations under Isolated and 29.82% under Vector. Ties fall from 52.73% to 12.35%, while losses increase (Figure 5).

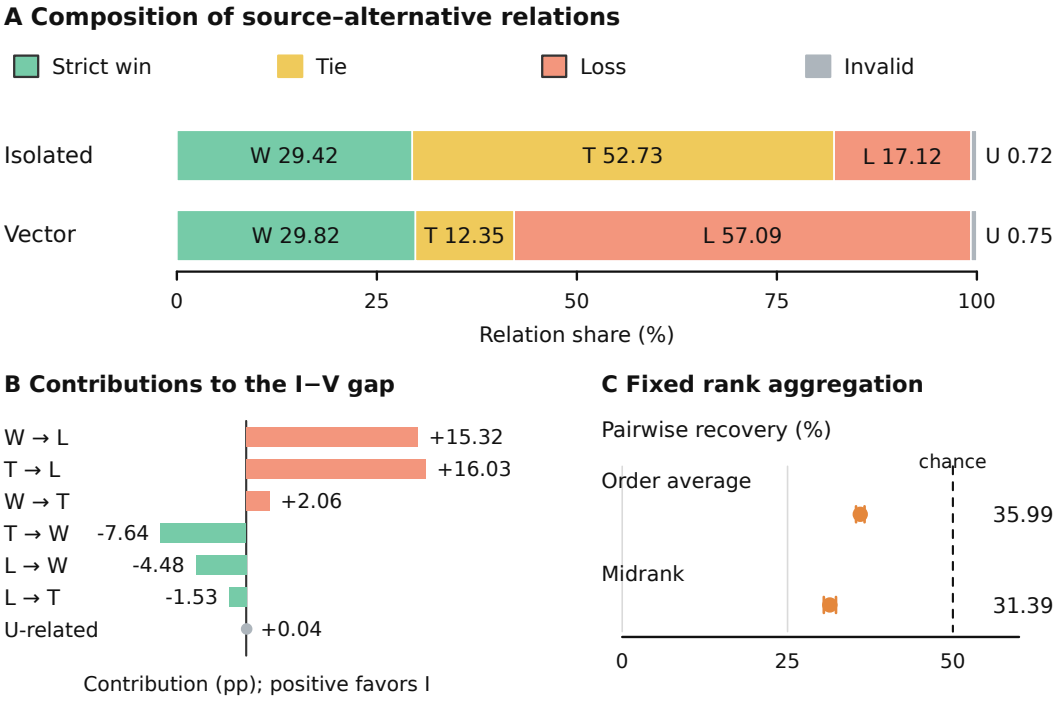


Figure 5: Behavioral diagnostics of census SOURCE recovery. Top: strict wins, ties, losses, and invalid relations; recovery assigns credits $1, \frac{1}{2}, 0, 0$. Bottom left: signed contributions to the I−V gap from paired transitions. "U-related" sums all transitions involving invalidity. Bottom right: order-average recovery and fixed midrank aggregation, with 95% source-cluster intervals. These analyses reuse the same saved outputs.

Pairing the same source–alternative relations across procedures shows why similar win totals do not imply similar judgments. Of all paired relations, 15.32% move from W to L and 32.06% from T to L. Because ties receive half credit, the T→L share of 32.06% contributes 16.03 pp to the recovery gap. Reverse movements partly offset these losses: 15.29% move from T to W and 4.48% from L to W. A source can lose one comparison and win another without changing its marginal win rate. Appendix C.4 includes all transitions, including invalidity. This is behavioral accounting, not identification of an internal mechanism.

Direct choices also recover poorly. Could the failure be confined to deriving relations from scores? On the established 237 items, the six models instead choose directly between the source and each alternative in both positions. SOURCE Direct accuracy is 30.19% (95% CI [28.12, 32.31]), below the binary reference, with every choice recoverable under semantic readout. Thus a score-to-relation conversion is not necessary for poor recovery. Direct was designed after observing the scoring results and fixed before its own outputs; it reuses the target items and is a diagnostic, not an independent confirmation (Appendix D.5).

Fixed midrank aggregation lowers census recovery. Order dependence raises a further practical question: can combining rankings improve source recovery? As a post hoc diagnostic, we applied one midrank/Borda rule, fixed before calculation, to the saved Vector outputs. After aligning candidate IDs, it orders candidates by their mean tie-aware rank across valid orders. Invalid orders abstain and aggregate ties remain. Recovery falls from 35.99% to 31.39%, a change of -4.61 pp (95% cluster CI [-4.96, -4.23]). Effects differ across contemporary models (Appendix C.5). This result concerns one fixed rule, not all permutation aggregation or the full Kemeny method of Tang et al. (2024). Combining the orders changes the recovered relations; it does not automatically improve them. These diagnostics characterize the observed failures, but leave candidate visibility and output action bundled in the original Isolated–Vector comparison.

5 Separating candidate context from output action

5.1 Context can matter with the target and scalar action fixed

To isolate a change in surrounding material from a change in requested output, we score each candidate on 237 items (224 dependency groups) using Qwen2.5-32B-Instruct and GPT-5.6 Sol. Background, question, target text, SOURCE instruction, and single-score output stay fixed. A adds no body text; B adds ordinary-life text; C adds scientific text from other questions; D adds the other fifteen candidates (Figure 6). B and C approximately match D's length without truncation. Donors, templates, orders, and analysis were fixed before these outputs, after the procedure comparisons.

Each model completes 15,168 logical requests, all valid and without technical missingness. Recovery averages fifteen relations per item. The primary contrast is the paired item mean C−D, with 95% intervals from 10,000 synchronized resamples of the 224 dependency groups (Appendix F.2). The intervals concern the sampled items and groups under these fixed donors and procedures.

Sol loses source recovery when the appended material consists of the other candidates. Pairwise recovery falls from 99.51% in C to 87.97% in D, a primary C−D gap of 11.53 pp (95% CI [9.36, 13.85]). The supporting B−D contrast is 11.74 pp (95% CI [9.47, 14.13]); it shares the items, model, and D condition with the primary contrast. Close but imperfect length matching argues against a simple input-length explanation.

When only one candidate can be kept, a post hoc description of the same scores shows tie-aware sixteen-way Hit@1 falling from 97.03% to 50.47% (Appendix F.3). Both pairwise recovery and source retention in D remain above their respective chance references, including 6.25% for sixteen-way selection.

Qwen's small contrast reflects a different behavior. Its C−D gap is -1.48 pp (95% CI [-3.21, 0.26]), but ties occupy 88.64–94.18% of relations. Scores concentrate at 8500, with many whole-pool ties. Half-credit ties keep recovery near chance while providing little basis for selecting a candidate. The small change therefore does not establish a practically robust selector. Sol shows context sensitivity in otherwise discriminating scores; Qwen shows weak discrimination despite a small contrast.

5.2 Matching visibility constrains the role of score grouping

Context losses therefore do not require a vector output. Does joint scoring moderate them when visible candidates are matched? The dual-focus study shows both focal candidates and fourteen context blocks in every call. Separate scoring requests one focal score, joint scoring requests both, and auxiliary Direct requests a choice. C supplies cross-question scientific text and D the remaining candidates from the item. Even separate scoring sees both focal texts.

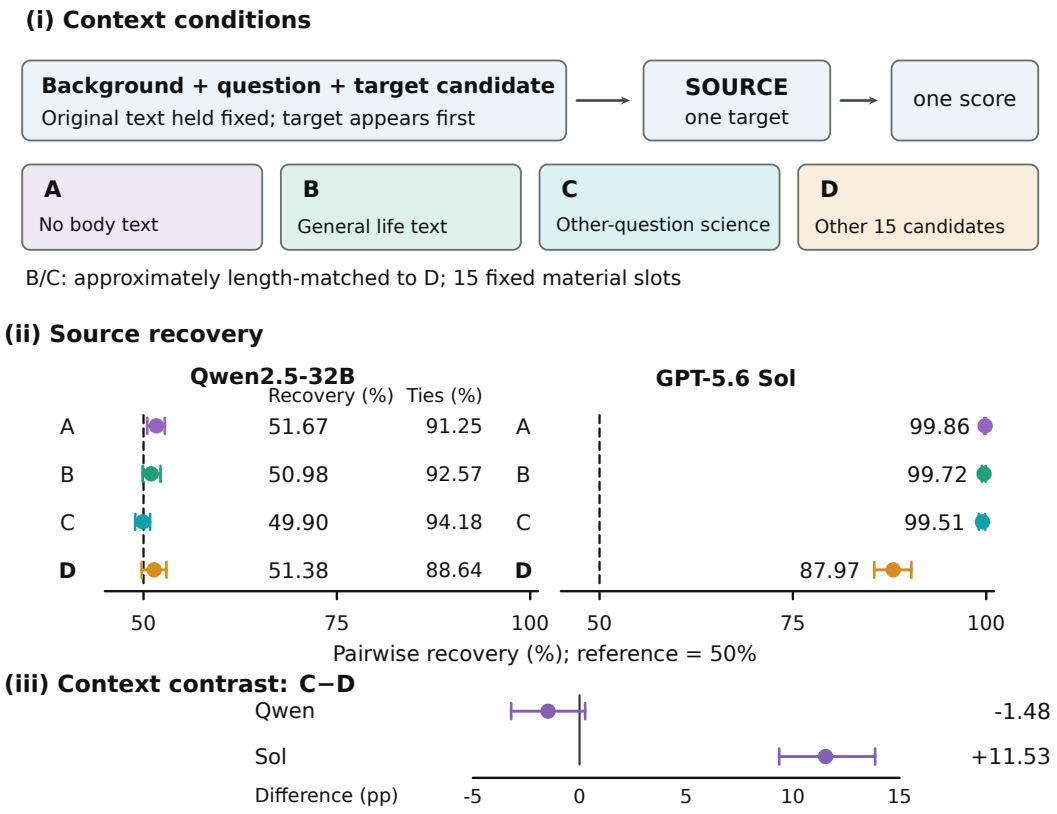


Figure 6: Context replacement with the target and scalar output held fixed. Panel (i) shows the common request and four appended bodies; (ii) shows pairwise recovery (50% chance) and Qwen's tie percentages; (iii) shows the paired difference between conditions C and D. Points and whiskers use 95% dependency-cluster intervals. The absolute-recovery axes start at 45%; all confidence intervals are shown in full. Conditions are categorical. Neither model is averaged into the census panel.

On 199 common-complete items in 199 dependency groups, the primary interaction $(C - D)_{\text{separate}} - (C - D)_{\text{joint}}$ is 0.84 pp (95% CI [-0.29, 2.14]). Its interval leaves the direction unresolved while constraining moderation on the prespecified 10 pp scale; it does not establish equivalence. Descriptive context losses remain 3.02 pp under separate scoring and 2.18 pp under joint scoring. Absolute recovery is near the ceiling, ties are rare, and all returned outputs in this subset are valid. Appendix F.6 gives the six cells, weights, and exclusion from the planned 200-item population.

The two controls answer distinct questions: material replacement affects scalar scoring for Sol, while matched visibility leaves no confirmed direction for moderation by score grouping. They do not measure a repair from the first setting to the second. Samples, focal organization, templates, and orders differ; separate/joint are also not Isolated/Vector. Context replacement changes style, relevance, and competing content together; it does not isolate an internal cue or apportion the historical 19.80 pp gap (Appendix F.4).

6 Where the findings transfer

We next examine whether these patterns extend across selection budgets, candidate pools, models, and task labels.

Selection budgets and candidate pools. A shortlist depends on more than average pairwise recovery. The detailed study's recovery curves show how often the source survives as more candidates are retained, while order stability measures whether reordered inputs produce the same winners (Appendices D.1–D.3). Census Vector winner-set agreement is only 14.02%. Repeated selection can be consistently wrong, and similar recovery can hide changing winners. Changing pool size supplies no common remedy either: nested candidate sets produce heterogeneous, nonmonotonic effects across the tested models (Appendix D.6).

Procedure gaps and absolute performance. The contemporary models separate sensitivity from below-chance performance. Their Isolated–Vector gaps are 1.27 pp for GPT-5.6 Sol, -2.95 pp for Claude Opus 5, and 35.30 pp for DeepSeek V4 Pro, yet every Vector level exceeds 50% (Appendix E.1). Sol reaches 99.85% and 98.58% under Isolated and Vector. DeepSeek's larger gap accompanies 60.05% Vector recovery.

Claude's full-population gap includes invalid-output penalties. On 223 jointly complete-valid items, the conditional I−V gap is +0.10 pp. This selected subset does not replace the primary estimate, which measures both usable-output production and recovery. Its negative gap is not a clear reverse advantage in valid ranking ability.

Different tasks need separate interpretations. The severe census inversion does not recur uniformly on external labels. For JudgeBench response correctness (Tan et al., 2025), scoring is modestly above binary chance and six-model Direct is close to chance under invalid-as-zero scoring. Llama's low validity substantially lowers that mean (Appendix E.2); the panel result does not place every judge's valid choices near chance. A response-only classifier has chance-compatible intervals, leaving text cues unresolved rather than establishing their absence. On TruthHypo's 800 entity-disjoint biomedical relation items (Xiong et al., 2025), I−V is -2.93 pp (95% CI [-3.86, -2.02]), with direction sensitive to readout (Appendix B.1). Correctness and biomedical relation labels differ from source identity. These comparisons delimit transfer; they do not estimate a shared cross-task reversal (Appendices E.2–E.4).

7 Discussion and conclusion

Predictable source labels coexist with fragile recovery: changing the procedure produces sharply different rankings on the same candidate pools. The controls show that scalar scoring remains sensitive to context, while matched visibility leaves the direction of moderation by joint output unresolved. A benchmark score describes a model operating through a particular procedure, not an ability detached from that procedure.

Candidate-only prediction tests how much label recovery is possible without scientific context. Its high accuracy here limits what a high source-recovery score alone establishes about reasoning over the background and question, while leaving the evaluator's internal cues unidentified. Source identity also remains a proxy, not a certificate of truth, novelty, or experimental value.

Mean recovery, its chance reference, wins, ties, invalids, and order behavior should be read together. Qwen's small contrast accompanies little discrimination, Claude's gap includes unusable scores, and a large contemporary-model gap can coexist with above-chance recovery. These are different reasons to inspect what a procedure retains. The scalar-context loss and fixed midrank result also support validating the actual visible material, requested action, and aggregation rule. Success in a neighboring configuration does not validate the selection procedure to be used.

The findings concern the evaluated models, templates, donors, orders, and labels. Intervals reflect item or dependency-group sampling. Several diagnostics reuse targets or outputs, and the dual-focus subset conditions on service acceptance. The rewriting pilot did not yield qualified interventions or formal post-rewrite I−V evidence (Appendix F.5), and temporal contamination is unassessed. Future work could test which meaning-preserving text changes alter recovery, and whether recovery losses predict which hypotheses survive externally validated scientific screening.

Before treating source recovery as evidence of scientific screening ability, establish what predicts the label and verify what the intended evaluator procedure actually retains.

AI USE STATEMENT

Generative AI tools assisted with research design, code development, analysis, literature review, and manuscript preparation. The authors verified all analyses, citations, and claims and take full responsibility for the work.

ETHICS STATEMENT

This study examines source-label recovery and evaluation procedures. It does not assign scientific merit or misconduct to the authors of source studies or benchmark creators. Real item text and identifying source mappings are omitted from the examples. The task schematic is independently authored and is visually separated from observed numerical evidence.

REPRODUCIBILITY STATEMENT

Appendices A–G specify study populations, model configurations, feature construction, readout, weighting, uncertainty, and literal prompt templates. They distinguish the sampling units and inferential roles of the primary comparisons and supplementary diagnostics. An anonymous numerical replay package is available at `https://anonymous.4open.science/r/PLFR-replay-580D/`.

REFERENCES

Hritik Bansal, John Dang, and Aditya Grover. Peering Through Preferences: Unraveling Feedback Acquisition for Aligning Large Language Models. In *International Conference on Learning Representations*, 2024. URL `https://proceedings.iclr.cc/paper_files/paper/2024/hash/48a4f41ff000b121eabe7b2491b24aef-Abstract-Conference.html`.

Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel R. Bowman, and Noah A. Smith. Annotation Artifacts in Natural Language Inference Data. In *Proceedings of NAACL-HLT*, pp. 107–112, 2018. doi: 10.18653/v1/N18-2017. URL `https://aclanthology.org/N18-2017/`.

Yujie Liu, Zonglin Yang, Tong Xie, Jinjie Ni, Ben Gao, Yuqiang Li, Shixiang Tang, Wanli Ouyang, Erik Cambria, and Dongzhan Zhou. ResearchBench: Benchmarking LLMs in Scientific Discovery via Inspiration-Based Task Decomposition. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 13187–13207, 2026. doi: 10.18653/v1/2026.findings-acl.644. URL `https://aclanthology.org/2026.findings-acl.644/`.

Chris Lu, Cong Lu, Robert Tjarko Lange, Jakob Foerster, Jeff Clune, and David Ha. The AI Scientist: Towards Fully Automated Open-Ended Scientific Discovery. *arXiv preprint arXiv:2408.06292*, 2024. doi: 10.48550/arXiv.2408.06292. URL `https://arxiv.org/abs/2408.06292`.

Adam Poliak, Jason Naradowsky, Aparajita Haldar, Rachel Rudinger, and Benjamin Van Durme. Hypothesis Only Baselines in Natural Language Inference. In *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics*, pp. 180–191, 2018. doi: 10.18653/v1/S18-2023. URL `https://aclanthology.org/S18-2023/`.

Swati Rajwal, Sanjay Das, and Tirthankar Ghosal. Do LLMs Know a Good Hypothesis When They See One? Logit-Based Energy Scoring Outperforms Prompted LLM-as-Judge for Scientific Hypothesis Ranking. *arXiv preprint arXiv:2608.17270*, 2026. URL `https://arxiv.org/abs/2608.17270`.

Melanie Sclar, Yejin Choi, Yulia Tsvetkov, and Alane Suhr. Quantifying Language Models' Sensitivity to Spurious Features in Prompt Design or: How I Learned to Start Worrying about Prompt Formatting. In *International Conference on Learning Representations*, 2024. URL `https://arxiv.org/abs/2310.11324`.

Sijun Tan, Siyuan Zhuang, Kyle Montgomery, William Y. Tang, Alejandro Cuadron, Chenguang Wang, Raluca Ada Popa, and Ion Stoica. JudgeBench: A Benchmark for Evaluating LLM-Based Judges. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/9e720fce64f91114c49cfd640d821da3-Paper-Conference.pdf.

Raphael Tang, Crystina Zhang, Xueguang Ma, Jimmy Lin, and Ferhan Ture. Found in the Middle: Permutation Self-Consistency Improves Listwise Ranking in Large Language Models. In *Proceedings of NAACL-HLT*, pp. 2327–2340, 2024. doi: 10.18653/v1/2024.naacl-long.129. URL https://aclanthology.org/2024.naacl-long.129/.

Tuhina Tripathi, Manya Wadhwa, Greg Durrett, and Scott Niekum. Pairwise or Pointwise? Evaluating Feedback Protocols for Bias in LLM-Based Evaluation. In *Second Conference on Language Modeling*, 2025. URL https://openreview.net/forum?id=uyX5Vnow3U.

Yidong Wang, Yunze Song, Tingyuan Zhu, Xuanwang Zhang, Zhuohao Yu, Hao Chen, Chiyu Song, Qiufeng Wang, Zhen Wu, Xinyu Dai, Yue Zhang, Cunxiang Wang, Wei Ye, and Shikun Zhang. TrustJudge: Inconsistencies of LLM-as-a-Judge and How to Alleviate Them. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=4uPyOCeN6U.

Guangzhi Xiong, Eric Xie, Corey Williams, Myles Kim, Amir Hassan Shariatmadari, Sikun Guo, Stefan Bekiranov, and Aidong Zhang. Toward Reliable Scientific Hypothesis Generation: Evaluating Truthfulness and Hallucination in Large Language Models. In *Proceedings of IJCAI*, pp. 7849–7857, 2025. doi: 10.24963/ijcai.2025/873. URL https://doi.org/10.24963/ijcai.2025/873.

A TASKS, STUDY POPULATIONS, AND MODEL SELECTION

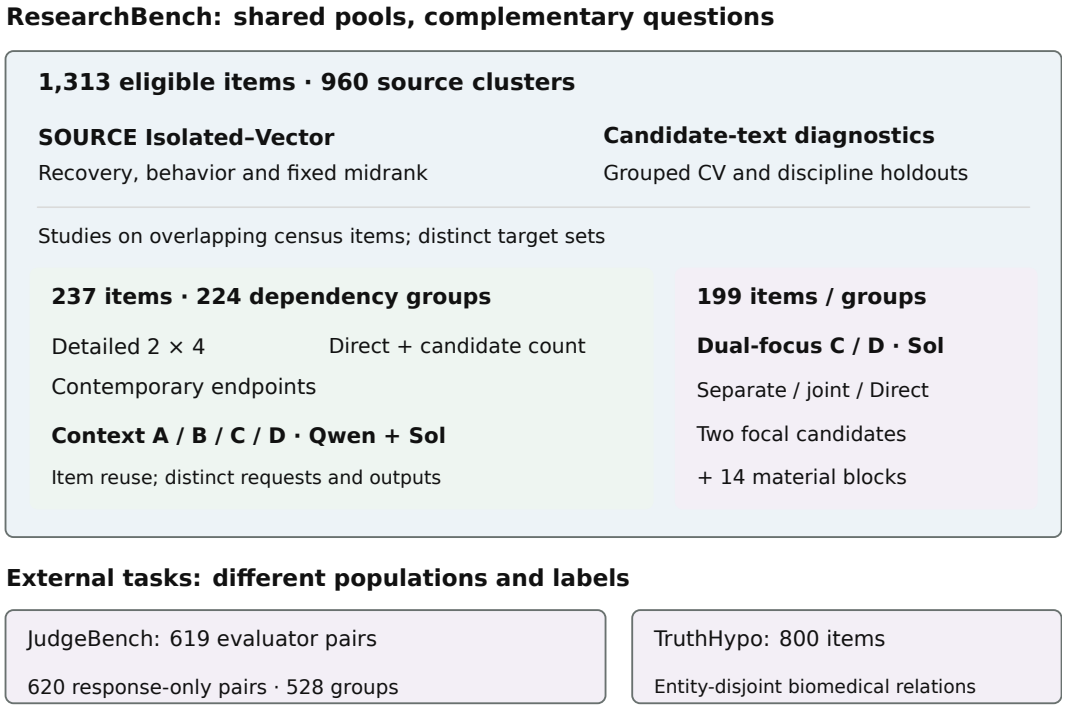


Figure 7: Study coverage and evidence relationships. ResearchBench groups lie within the census: the detailed studies share 237 targets, while the dual-focus study uses 199 distinct targets (Appendix F.6). External tasks have different populations and labels. Grouping denotes coverage, not chronology.

A.1 STUDY POPULATIONS AND OVERLAP

The census covers 1,313 eligible items from the 1,323-item release after ten outcome-independent common-context exclusions. Its 960 source clusters also define candidate-only dependency units. It extends the earlier detailed study's coverage, reusing available outputs for matching conditions.

The detailed study uses 237 items from a deterministically allocated, discipline-stratified pool of 261 context-compatible items. A fixed hash rule assigned two items per discipline to a 24-item protocol-development set; its 237-item complement received the same two-target, four-procedure protocol. The sets share no exact background or question, but one source cluster crosses their boundary. The evaluation subset contains 224 source clusters. Direct, candidate-count, contemporary-model, and context-control studies reuse these 237 items. Context control uses source/duplicate union groups computed on full common support before restriction: 224 groups (212 singletons, eleven pairs, and one triple). It generates new scalar scores under a distinct context protocol. Direct was designed after observing the score results and fixed before generating its own outputs.

The dual-focus follow-up selects one item per dependency group outside these 237 targets: 199 common-complete items remain after one uniform service-refusal exclusion (Appendix F.6). These census items overlap the context-control donor pool.

Candidate-text predictions were fixed before scoring held-out labels; their primary results preceded descriptive linkage to census outcomes. JudgeBench uses 619 evaluator pairs and 620 response-only pairs in 528 dependency groups; TruthHypo uses 800 entity-disjoint items (Appendices E.2–E.4).

A.2 Model panel and development comparison

The six-model panel contains the largest tested checkpoint in each of six families, among nineteen development checkpoints. This rule was fixed before the matched study's outcomes; development observations informed the research question. The 241-item development comparison used restricted Yes/No pointwise scores and restricted A–P joint selection, with two fixed prompt variants and two reverse joint orders. Outcomes were averaged over variants and orders. Figure 8 reports this historical Hit@1 comparison, separately from the generative-evaluation results.

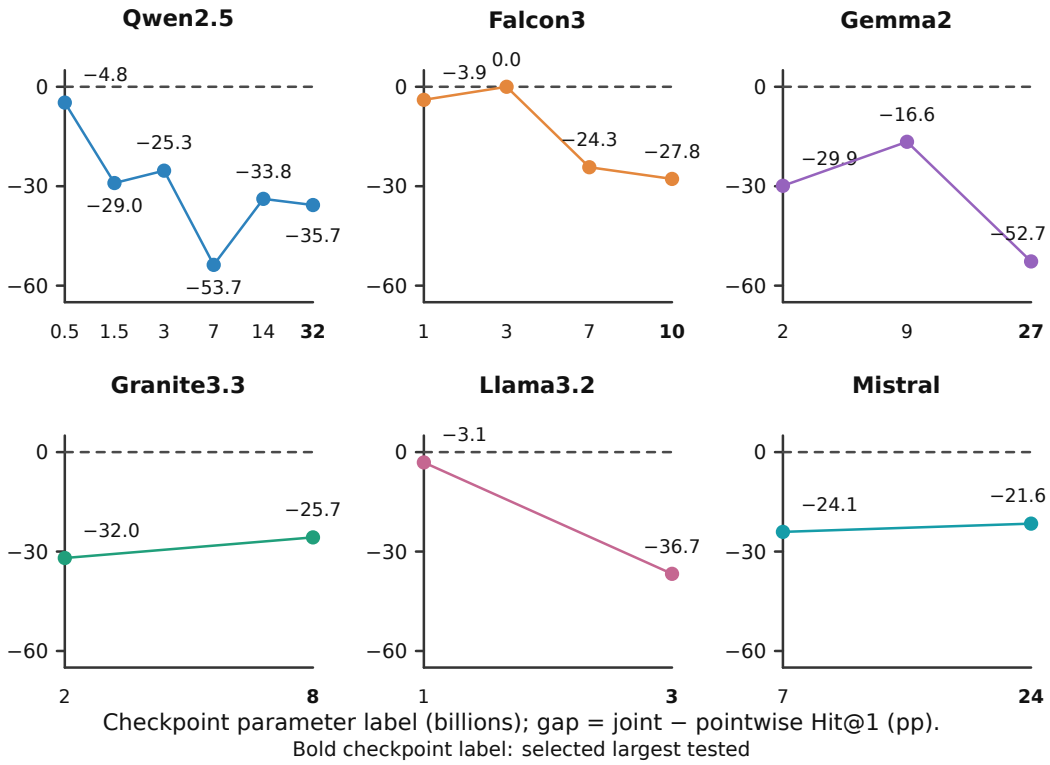


Figure 8: The six development-family trajectories on 241 items. The vertical axis is joint-minus-pointwise source Hit@1 in percentage points under the historical restricted readout. Lines join the tested checkpoints within each family; spacing is categorical. Bold checkpoint labels mark the largest tested checkpoints selected for the six-model panel. Selection follows checkpoint size within family, not the largest observed performance gap.

Table 1: Exact development-checkpoint identities and historical differences in Figure 8. Bold marks the selected checkpoints. Values use the restricted-readout definition on 241 items.

Family	Checkpoint	Joint − pointwise (pp)
Qwen2.5	Qwen2.5-0.5B-Instruct	-4.77
Qwen2.5	Qwen2.5-1.5B-Instruct	-29.05
Qwen2.5	Qwen2.5-3B-Instruct	-25.31
Qwen2.5	Qwen2.5-7B-Instruct	-53.73
Qwen2.5	Qwen2.5-14B-Instruct	-33.82
Qwen2.5	**Qwen2.5-32B-Instruct**	-35.68
Falcon3	Falcon3-1B-Instruct	-3.94
Falcon3	Falcon3-3B-Instruct	0.00
Falcon3	Falcon3-7B-Instruct	-24.27
Falcon3	**Falcon3-10B-Instruct**	-27.80
Gemma2	gemma-2-2b-it	-29.88
Gemma2	gemma-2-9b-it	-16.60
Gemma2	**gemma-2-27b-it**	-52.70
Granite3.3	granite-3.3-2b-instruct	-31.95
Granite3.3	**granite-3.3-8b-instruct**	-25.73
Llama3.2	Llama-3.2-1B-Instruct	-3.11
Llama3.2	**Llama-3.2-3B-Instruct**	-36.72
Mistral	Mistral-7B-Instruct-v0.3	-24.07
Mistral	**Mistral-Small-24B-Instruct-2501**	-21.58

A.3 Generation settings

The six-model ResearchBench panel uses the recorded open-weight checkpoint revisions and chat templates, bfloat16 without quantization, greedy decoding, and sequential batch size one. Scalar, vector, and Top-5 output caps are 16, 256, and 48 tokens. All-visible procedures use four predetermined orders. Direct preserves both source positions and has a sixteen-token cap. Prompts fit model context limits without truncation. An observed malformed output receives its prescribed score rather than a retry.

The detailed ResearchBench, Direct, and JudgeBench evaluator runs used CUDA 12.8, PyTorch 2.8.0+cu128, and Transformers 5.15.1 on H100 80GB hardware. JudgeBench caps are sixteen tokens for Isolated, Marked, and Direct, and thirty-two for Vector. Its correctness rubric prioritizes instruction following and factual/logical correctness over style. Marked, Vector, and Direct use both response orders. Contemporary hosted endpoints are identified by their recorded aliases and configurable request parameters; no immutable provider-side weight revision or internal precision was available as a control. TruthHypo settings are given in Appendix E.4.

B Evaluator procedures, readout, and inference

B.1 Readout, ties, and sensitivity

Literal and semantic validity. The following rules govern the census, detailed, Direct, contemporary, and external evaluator studies. The scalar-context and dual-focus studies use their own strict JSON policies (Appendices F.2 and F.6). Both forms of validity concern whether the requested output is recoverable, not whether the judgment is scientifically correct. Literal validity requires the entire output to be the requested value, allowing only surrounding RFC JSON whitespace. JSON keys, candidate membership, value types, numeric range, ranking length, and completeness must match the schema. Duplicate keys fail. JudgeBench Direct instead requests the exact surface `Output (a)` or `Output (b)`.

Semantic readout additionally permits one complete outer code fence with an empty or `json` info string; an exact displayed ID or JSON string for a selection action; an exact full, unique semantic label in a selection object or as a bare/JSON string under the displayed bijection; and the specified narrow JudgeBench Direct wrappers. Exact `(a)` or `(b)` and one Markdown emphasis/code wrapper around those or the full requested forms are accepted for JudgeBench Direct. Bare `a`/`b`, case folding,

fuzzy matching, substring extraction, prose surrounding JSON, multiple objects, foreign candidates, and incomplete vectors are rejected. Numeric outputs use integer scores in $[0, 10000]$. Selection recovery cannot supply missing numeric scores or ranking entries.

Numeric relation versus call validity. Isolated assembles sixteen scalar calls into one relation; Marked assembles sixteen within an order; Vector supplies one complete relation per order. An unrecoverable required call invalidates the affected relation. Consequently, a call-level valid rate need not equal the fraction of complete relations. For example, the census Isolated semantic call rate is 99.95%, but its complete-relation rate is 99.28%. Zero-credit invalid relations remain in every aggregate. A separate parser implementation gave identical decisions on the same stored outputs.

Tie-aware rank recovery. If a candidates score strictly above the source and t share its score (including the source), fractional top-r credit is

$$C(r) = \min\{1, \max\{0, (r-a)/t\}\}, \quad r = 1, \ldots, 16, \tag{4}$$

for a complete numeric relation, and zero for an invalid one. The source occupies the tied rank interval $[a+1, a+t]$. For a complete numeric relation with source score s_g and $K-1$ alternative scores, pairwise credit is

$$A_{\text{pair}} = \frac{1}{K-1} \sum_{a \neq g} \left[\mathbf{1}(s_g > s_a) + \tfrac{1}{2}\mathbf{1}(s_g = s_a) \right]. \tag{5}$$

It is zero when the required relation is invalid. A Top-5 action has only five ordered indices; its Hit@1 and Hit@5 are defined, but it cannot supply a full numeric relation. Classifier Hit@1 uses a different rule: a non-unique maximum abstains and receives zero.

Table 2: Readout sensitivity: study-specific readouts compared with the common semantic policy used for the main results. TruthHypo's study-specific readout requires exact format. Accuracy is in percent; contrasts are in percentage points.

Estimand	Study-specific readout	Common semantic
237-item SOURCE Isolated	57.78	57.44
237-item SOURCE Vector	36.17	36.05
SOURCE Direct	26.63	30.19
Native Direct	20.49	26.83
TruthHypo $I - V$ (pp)	6.33	-2.93

The study-specific ResearchBench core parser permitted first-balanced-object extraction from prose. Removing that recovery lowers some core endpoints. Complete-fence recovery raises Direct accuracy relative to whole-document parsing. For TruthHypo, numeric scores must satisfy the scalar or vector schema; Direct may additionally return the exact displayed ID or its full, unique semantic label under the displayed bijection. These recoveries change the sign of the Isolated-minus-Vector contrast relative to exact-format scoring (Table 2). The direction is therefore readout-sensitive. Eleven of its 62,400 calls are unrecoverable and retain zero credit.

The common semantic policy was selected after the detailed ResearchBench, Direct, candidate-count, JudgeBench evaluator, and TruthHypo outputs and results under their study-specific readouts were available. The policy and both implementations were fixed before harmonized aggregation; the census and contemporary study then used this readout. Gold-independent parsing describes the rule's inputs; it does not imply outcome-blind selection of that rule. The sensitivity comparison concerns the scoring definition.

B.2 Inference and aggregation details

All reported bootstrap intervals are marginal percentile intervals. Models are fixed named endpoints, not resampled members of a population. Orders and candidate relations are averaged within the item before the item contributes to a study-level mean. Multiplicity decisions and confidence intervals have different roles: an unadjusted interval excluding zero does not imply that a Holm-adjusted test passed.

A conditional held-out-label randomization diagnostic holds predictions fixed and measures their relation to the labels. It excludes uncertainty from retraining, as do the candidate-text bootstrap intervals.

Table 3: Uncertainty and decision rules across the study modules. Subrows distinguish analyses with different inference policies.

Analysis	Resampling unit	Tests and interval use
Benchmark-wide comparison (Appendix C.1)		
Census	Discipline-stratified paired items; source-cluster sensitivity	Three primary Holm tests; ± 5 pp practical comparison uses a 90% gap interval.
Candidate-text diagnostics (Appendix C.2)		
Grouped CV	Source clusters	One primary test; eight secondary Holm tests.
Discipline holdouts	Source clusters within discipline; then macro-average	Separate validation estimand in the candidate secondary family.
Procedure diagnostics (Appendices D.1–D.6)		
Detailed 2×4	Discipline-stratified paired items	Marginal intervals for cells and contrasts.
Top-r recovery	The same 237 items	Pointwise intervals at each cutoff, not a simultaneous band.
Direct choices	Discipline-stratified paired items	Descriptive diagnostic intervals.
Count: ResearchBench	Discipline-stratified paired items	Primary $k = 16$ versus $k = 2$ contrast: ± 5 pp region using a 95% interval. Four secondary adjacent/trend Holm tests.
Count: JudgeBench (Appendix D.6)	Category-stratified paired questions	Two generator-specific focal Holm tests; equal-weight contrast separate.
Contemporary models (Appendix E.1)		
Each API model	Paired items; source-cluster sensitivity	Three primary gap tests and three secondary Vector-versus-50% tests form separate Holm families; ± 5 pp comparisons use 90% gap intervals.
External tasks (Appendices E.2–E.4)		
JudgeBench evaluators	Category-stratified response pairs	Descriptive intervals.
JudgeBench text	Question/dependency clusters	Two generator-specific primary Holm tests; pointwise intervals.
TruthHypo	Items stratified by nine task–gold cells	Three adjacent secondary Holm tests; ± 5 pp comparison uses a 90% gap interval.

Reported intervals are marginal 95% percentile intervals unless a different level is specified. Each bootstrap uses 10,000 draws, synchronized across matched conditions *within that study*; studies retain their own populations and resampling units. Models are fixed named endpoints, and cluster sensitivities form separate interval families. Candidate-text intervals condition on fixed predictions and exclude refitting uncertainty. Holm decisions are assessed separately from marginal intervals; nonsignificance does not establish practical equivalence.

Table 4: Additional inference policies for saved-output diagnostics and context control and its dual-focus follow-up. All intervals use 10,000 cluster-bootstrap draws. The analyses retain different quantile rules and scientific roles.

Analysis	Resampling unit	Interval use
Behavior and aggregation (Appendices C.4–C.5)		
Saved-output diagnostics	Source clusters; matched conditions and fixed models	Post hoc pointwise 95% intervals; linear percentiles. No new confirmatory test family.
Controlled context (Appendix F.2)		
Each model, A–D	Source/duplicate union groups; paired items	C−D primary; four supporting contrasts. Marginal 95% intervals use nearest-order percentiles; ± 5 pp is a substantive range.
Dual-focus context and readout (Appendix F.6)		
199 common items	Source/duplicate groups; paired contexts and readouts	One primary interaction; linear 95% percentiles. Six levels and three C−D intervals descriptive. Prespecified 10 pp scale; no equivalence test.

C Benchmark-wide results and candidate-text diagnostics

C.1 Census levels and order diagnostics

Table 5: Census levels by checkpoint. I denotes Isolated and V Vector. Pairwise and recovery levels are percentages; $I - V$ is in percentage points. H@1 and H@5 use fractional ties.

Model	I pair	I H@1	I H@5	V pair	V H@1	V H@5	$I - V$
Qwen2.5-32B	52.33	6.98	33.37	28.11	6.98	17.32	24.23
Falcon3-10B	58.17	14.65	42.65	41.01	5.54	23.40	17.16
Gemma2-27B	66.21	18.65	54.26	22.59	3.17	11.01	43.63
Granite3.3-8B	47.80	8.31	31.06	42.56	4.71	24.60	5.24
Llama3.2-3B	49.72	4.91	32.22	45.94	4.41	26.51	3.78
Mistral-Small-24B	60.50	7.30	43.87	35.75	6.59	20.26	24.75
Equal-weight panel	55.79	10.13	39.57	35.99	5.23	20.51	19.80

Table 6: Vector order diagnostics on the census (%). Exact agreement compares nonempty top-one winner sets across the six unordered pairs of four orders. Top-five sets include ties at the fifth-score boundary. Unanimity requires the same unique winner in every order. Range is the mean within-item/model range of pairwise accuracy, in percentage points.

Model	Top-1 agree	Top-5 Jaccard	Unanimous	Range (pp)
Qwen2.5-32B	30.06	45.15	9.98	45.13
Falcon3-10B	8.49	24.52	0.15	60.26
Gemma2-27B	19.38	42.05	2.74	42.67
Granite3.3-8B	7.06	26.78	0.00	59.14
Llama3.2-3B	4.98	35.04	0.00	55.03
Mistral-Small-24B	14.17	41.45	0.53	49.53
Panel	14.02	35.83	2.23	51.96

The 299 unrecoverable calls include extra prose, ambiguous or conflicting outputs, and schema, type, or range failures. A candidate-specific call or one order is not an independent unit for the item bootstrap.

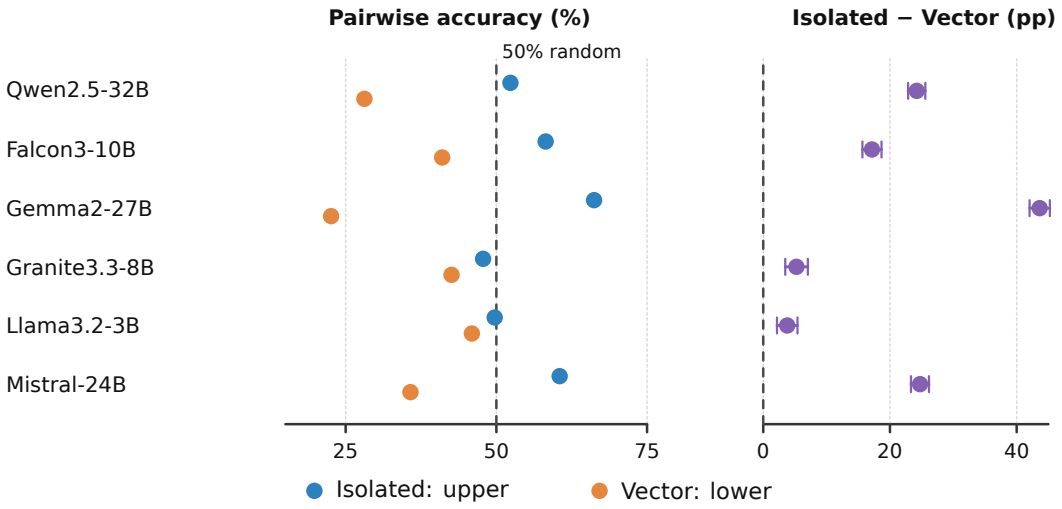


Figure 9: Census heterogeneity across the six fixed checkpoints. Left: Isolated and Vector pairwise levels, with a 50% chance reference. Right: the paired $I - V$ difference with 95% item-bootstrap intervals. Level points are descriptive; the accompanying table gives exact values.

C.2 Candidate-only features and validation

Each candidate is classified from its own text. The 23 surface features are character and UTF-8 byte counts, whitespace token count, sentence count, mean tokens per sentence, casefolded type–token ratio, digit and punctuation rates, parenthesis/colon/semicolon/dash/quote/formula-symbol rates, uppercase token rate, bullet or enumeration indicator, question-mark and exclamation-mark indicators, hedging/modal/connective lexicon rates, newline count, and paragraph count. The lexicons match exact casefolded whole tokens.

The sanitized view applies a fixed sequence to each candidate's decoded text field, without background, question, other candidates, or labels. It removes an initial `title` or `paper title` prefix (including the specified Markdown emphasis and delimiter), then replaces URLs, email addresses, arXiv identifiers, DOIs, bracketed numeric citations, and parenthesized author–year citations, in that order. The replacements are `<URL>`, `<EMAIL>`, `<ARXIV_ID>`, `<DOI>`, and `<CITATION>` for both citation forms. These regular expressions are fixed and require no fitting. The Python regular expressions below specify their matching behavior, in application order. All except the two citation patterns use case-insensitive matching; `\u2019` denotes the Unicode right apostrophe.

```
TITLE:  \A\s*(?:[*_]{0,2})(?:paper\s+)?title(?:[*_]{0,2})\s*[:=\-]\s*
URL:    \b(?:https?://|www\.)[^\s<>()]+
EMAIL:  \b[A-Z0-9._%+-]+@[A-Z0-9.-]+\.[A-Z]{2,}\b
ARXIV:  \b(?:arXiv\s*:\s*)?\d{4}\.\d{4,5}(?:v\d+)?\b
DOI:    \b10\.\d{4,9}/[-._;()/:A-Z0-9]+
CITE-N: \[(?:\d{1,3}(?:\s*[,;-]\s*\d{1,3})*)\]
CITE-Y: \([A-Z][A-Za-z'\u2019-]+(?:\s+et\s+al\.)?,?\s+(?:19|20)\d{2}\)
```

Replacement is global; URL replacement precedes DOI replacement, so a DOI inside a URL becomes a single URL placeholder. The transformed text still carries wording, length, punctuation, and unmatched identifiers or citation forms. It neither removes all source cues nor establishes semantic invariance.

Word TF-IDF uses one- and two-grams, minimum document frequency two, maximum document fraction 0.98, and at most 100,000 features. Character `char_wb` TF-IDF uses three- through five-grams, minimum document frequency three, and at most 200,000 features. Both use lowercase text, sublinear term frequency, IDF, and L2 normalization. Surface features are scaled without centering. The three feature blocks receive equal weights.

Balanced L2 logistic regression uses an intercept, the `liblinear` solver, tolerance 10^{-5}, seed 20260902, and $C \in \{0.01, 0.1, 1, 10\}$. Inner selection maximizes item-balanced source Hit@1; ties choose the smallest C. Five grouped outer folds and four grouped inner folds isolate source/duplicate components. All learned transformations are fitted inside the appropriate training fold. The iteration cap is 100,000, using the primal form for combined and surface variants and the dual form for lexical-only.

For leave-one-discipline-out validation, a training item is excluded if its dependency group touches the held-out discipline. We report the equal-weight macro-average of the twelve held-out discipline accuracies. Individual discipline values are descriptive. The main classifier produced a unique maximum in every item. A separate implementation reproduced all candidate scores on the same data, providing a computational consistency check.

Table 7: Primary candidate-only result and all eight specified secondary results. Intervals condition on the fixed cross-fitted scores. Candidate counts and chance levels differ across the role-restricted subpools.

Diagnostic	Estimate (%)	95% cluster CI	Chance (%)
Combined (primary)	98.40	[97.71, 99.03]	6.25
Hit@5	100.00	[100.00, 100.00]	31.25
Source–alternative pairwise	99.89	[99.84, 99.93]	50.00
Source + five fake, Hit@1	99.77	[99.47, 100.00]	16.67
Source + ten model, Hit@1	98.55	[97.88, 99.16]	9.09
Combined, sanitized	98.48	[97.80, 99.09]	6.25
Surface only	89.49	[87.80, 91.09]	6.25
Lexical only	98.55	[97.88, 99.16]	6.25
Held-out discipline, macro	98.07	[97.29, 98.77]	6.25

Table 8: All specified deterministic surface probes. A non-unique optimum abstains and receives zero. The shortest rule is not a trained classifier.

Probe	Hit@1 (%)	Top-tie abstentions
Always longest	0.00	4
Always shortest	79.51	2
Maximum sentence count	0.00	239
Minimum sentence count	29.86	860
Maximum digit rate	4.72	3
Minimum digit rate	11.04	765
Maximum punctuation density	7.01	0
Minimum punctuation density	18.20	1

Table 9: Descriptive held-out-discipline cells. Their macro-average is the specified held-out-discipline endpoint.

Held-out discipline	Items	Hit@1 (%)
Astronomy	82	98.78
Biology	106	99.06
Business	107	100.00
Cell Biology	150	100.00
Chemistry	108	99.07
Earth Science	108	99.07
Energy Science	112	95.54
Environmental Science	110	100.00
Law	95	93.68
Material Science	112	97.32
Math	100	100.00
Physics	123	94.31

C.3 Descriptive linkage to candidate margins

The association analysis retains the 19,695 source–alternative relations from the census. It constructs each predictor from within-item midrank percentiles of held-out classifier scores, then takes the source-minus-alternative margin. Raw logits from different outer folds are not pooled as calibrated magnitudes. The equal-weight six-model procedure gap remains positive in all four observed margin strata (Table 10). Ties in the discrete margin make these strata unequal in size; they are not equal-size quartiles.

Table 10: Descriptive relation-margin strata, in increasing margin order. Levels are percentages; the difference is in percentage points.

Observed stratum	Relations	Isolated	Vector	Gap (pp)
1	3961	62.94	41.74	21.21
2	5252	59.91	38.37	21.54
3	5252	54.81	34.43	20.38
4	5230	47.22	30.82	16.39

The strata describe associations within observed pools. With no exact zero-margin relations and little negative-margin support, they cannot identify how evaluators would behave on candidates without construction cues. The association analysis uses a separate source-cluster bootstrap from the census, so the intervals differ even where the point estimate agrees.

C.4 WIN, TIE, LOSS, AND PAIRED-TRANSITION DIAGNOSTICS

The behavior and transition analyses describe the census outputs under the same semantic readout, candidate mapping, item weights, and fixed six-model panel. They are post hoc diagnostics. For each source–alternative relation, W means the source's score is strictly higher, T means equal, L means lower, and U means the required full score relation is invalid. Credits are $1, 1/2, 0, 0$. Isolated contributes one assembled relation per item and model; Vector contributes four scheduled orders at equal weight. A missing scalar in Isolated invalidates its assembled pool; an invalid Vector order invalidates that order. These are the core study's complete-relation rules.

For the transition matrix, the same Isolated relation is paired with each Vector order. Averaging the resulting cells over fifteen alternatives, four orders, items, and the six fixed models gives joint transition shares. Thus 472,680 expanded relation records do not represent independent samples. Rows and columns recover the two procedure marginals. Multiplying each cell share by the Isolated credit minus the Vector credit gives its signed contribution to the recovery gap. The contributions sum to 19.79743 pp; all transitions involving U contribute a net 0.03998 pp.

Intervals in these descriptive tables use the 10,000 source-cluster bootstrap draws (PCG64, seed 20260904), synchronizing models and conditions within a resampled item cluster and preserving equal item and fixed-model weights. Percentiles use linear interpolation. They are pointwise intervals, not a simultaneous matrix band or an additional confirmatory test family. They are separate from the original census item-bootstrap intervals.

Table 11: Census relation composition. W/T/L/U are percentages of scheduled source–alternative relations; I and V denote Isolated and Vector. The six-model mean uses equal checkpoint weights.

Model	Procedure	W (%)	T (%)	L (%)	U (%)
Six-model mean	I	29.42	52.73	17.12	0.72
Six-model mean	V	29.82	12.35	57.09	0.75
Qwen2.5-32B	I	11.13	82.40	6.47	0.00
Qwen2.5-32B	V	27.88	0.46	71.67	0.00
Falcon3-10B	I	27.35	61.65	10.70	0.30
Falcon3-10B	V	39.50	3.02	56.66	0.82
Gemma2-27B	I	45.75	40.93	13.32	0.00
Gemma2-27B	V	21.96	1.25	76.79	0.00
Granite3.3-8B	I	27.97	39.66	28.33	4.04
Granite3.3-8B	V	34.66	15.79	46.80	2.74
Llama3.2-3B	I	28.67	42.10	29.23	0.00
Llama3.2-3B	V	32.06	27.75	39.29	0.89
Mistral-Small-24B	I	35.66	49.67	14.66	0.00
Mistral-Small-24B	V	22.85	25.80	51.33	0.02

Table 12: Complete census paired transition accounting. Shares are of all expanded relations; confidence intervals are pointwise 95% source-cluster intervals. Counts expand Isolated across four Vector orders. Positive contributions favor Isolated; diagonal and L/U exchanges contribute zero.

I→V	Count	Share (%)	95% CI	Gap contribution (pp)
W→W	46138	9.76	[9.26, 10.28]	0.00
W→T	19521	4.13	[3.88, 4.39]	2.06
W→L	72412	15.32	[14.58, 16.09]	15.32
W→U	1009	0.21	[0.17, 0.26]	0.21
T→W	72254	15.29	[14.82, 15.74]	-7.64
T→T	23902	5.06	[4.82, 5.29]	0.00
T→L	151556	32.06	[31.34, 32.78]	16.03
T→U	1548	0.33	[0.27, 0.39]	0.16
L→W	21197	4.48	[4.20, 4.78]	-4.48
L→T	14474	3.06	[2.83, 3.30]	-1.53
L→L	44386	9.39	[8.72, 10.09]	0.00
L→U	863	0.18	[0.14, 0.23]	0.00
U→W	1360	0.29	[0.21, 0.37]	-0.29
U→T	468	0.10	[0.06, 0.14]	-0.05
U→L	1487	0.31	[0.23, 0.41]	0.00
U→U	105	0.02	[0.01, 0.04]	0.00

Table 13: Model-specific census transitions (percent of all relations in each model). Rows are Isolated W/T/L/U; columns are Vector W/T/L/U. These complete matrices show heterogeneous movement, including reversals in both directions.

Qwen2.5-32B

I\V	W	T	L	U
W	5.06	0.08	5.99	0.00
T	21.98	0.34	60.07	0.00
L	0.84	0.03	5.60	0.00
U	0.00	0.00	0.00	0.00

Falcon3-10B

I\V	W	T	L	U
W	11.96	0.97	14.16	0.26
T	23.93	1.73	35.56	0.43
L	3.51	0.30	6.76	0.12
U	0.11	0.02	0.18	0.00

Gemma2-27B

I\V	W	T	L	U
W	12.55	0.63	32.57	0.00
T	7.60	0.51	32.82	0.00
L	1.81	0.12	11.40	0.00
U	0.00	0.00	0.00	0.00

Granite3.3-8B

I\V	W	T	L	U
W	9.95	4.39	12.80	0.84
T	13.94	6.30	18.41	1.01
L	9.16	4.54	13.88	0.76
U	1.62	0.57	1.71	0.13

Llama3.2-3B

I\V	W	T	L	U
W	9.48	7.71	11.30	0.18
T	13.49	11.40	16.70	0.50
L	9.09	8.65	11.28	0.21
U	0.00	0.00	0.00	0.00

Mistral-Small-24B

I\V	W	T	L	U
W	9.57	11.00	15.09	0.00
T	10.78	10.06	28.82	0.01
L	2.50	4.74	7.42	0.00
U	0.00	0.00	0.00	0.00

C.5 One fixed midrank aggregation diagnostic

We fixed one aggregation rule before computing this post hoc diagnostic on saved outputs; the rule was not selected by comparing aggregate results. For each valid Vector order, candidate j has doubled midrank $1 + 2\#\{k : s_k > s_j\} + \#\{k : s_k = s_j\}$, with smaller ranks preferred. IDs are restored before ranks are combined. The negative sum of doubled midranks induces the same ordering as the mean over valid orders, because all candidates share the same number of valid orders. Integer sums preserve exact ties without a floating-point tie tolerance. Invalid orders abstain. If every order is invalid, the aggregate is U and stays in the denominator. No source label enters the aggregation.

The census contains 2, 3, 19, 180, and 7674 item–model cells with 0, 1, 2, 3, and 4 valid orders, respectively. The aggregate changes the endpoint from average credit across scheduled orders to credit from a consensus rank. Its -4.60555 pp census change combines -4.92902 pp from combining ranks and $+0.32347$ pp from renormalizing over available valid orders. Aggregate Hit@1 is 3.10% and Hit@5 is 14.88%. These values diagnose this rule's behavior; they do not evaluate the full Kemeny optimization and sampling procedure in Tang et al. (2024).

Table 14: Fixed midrank aggregation versus average Vector-order recovery. Levels are pairwise percentages; the change and its 95% source-cluster interval are in pp. Census uses 1,313 items; each contemporary endpoint uses 237 items. The same post hoc cluster-bootstrap policy applies within each population, with no averaging across contemporary endpoints.

Endpoint	Order average	Midrank	Change (pp)	95% CI
Six-model mean	35.99	31.39	-4.61	[-4.96, -4.23]
Qwen2.5-32B	28.11	24.99	-3.12	[-3.54, -2.71]
Falcon3-10B	41.01	36.43	-4.59	[-5.43, -3.74]
Gemma2-27B	22.59	17.14	-5.45	[-5.89, -5.00]
Granite3.3-8B	42.56	38.50	-4.06	[-5.03, -3.09]
Llama3.2-3B	45.94	42.41	-3.52	[-4.52, -2.53]
Mistral-Small-24B	35.75	28.86	-6.90	[-7.70, -6.07]
GPT-5.6 Sol	98.58	98.99	0.41	[0.23, 0.61]
Claude Opus 5	97.03	98.06	1.03	[0.35, 1.96]
DeepSeek V4 Pro	60.05	59.86	-0.19	[-1.23, 0.83]

D Detailed decomposition and diagnostic studies

D.1 Two targets and four procedures

Table 15: Panel results on 237 items: estimates and 95% item bootstrap intervals (%). SOURCE and PLAUSIBILITY use the same source label; the latter is a target-mismatch diagnostic.

Target	Procedure	Hit@1	Hit@5	Pairwise
Source	Isolated	10.84 [9.42, 12.36]	41.82 [39.15, 44.54]	57.44 [55.33, 59.57]
Source	Marked	6.14 [5.50, 6.84]	23.35 [21.93, 24.79]	40.38 [38.94, 41.82]
Source	Vector	5.61 [4.81, 6.50]	20.69 [19.33, 22.12]	36.05 [34.71, 37.41]
Source	Top-5 action	4.13 [3.38, 4.94]	18.18 [16.68, 19.73]	–
Plausibility	Isolated	7.43 [6.15, 8.86]	29.58 [26.92, 32.25]	46.93 [44.46, 49.37]
Plausibility	Marked	4.63 [4.02, 5.29]	20.21 [18.76, 21.68]	38.43 [37.03, 39.81]
Plausibility	Vector	3.43 [2.83, 4.08]	14.78 [13.53, 16.12]	29.12 [27.78, 30.49]
Plausibility	Top-5 action	3.64 [2.92, 4.43]	16.68 [15.24, 18.16]	–

Isolated and Marked differ in visibility while retaining scalar calls. Marked and Vector share full visibility but differ in output action and call count. Top-5 changes the output and available ranking information. These contrasts compare complete procedures without isolating latent judgment, attention, or calibration.

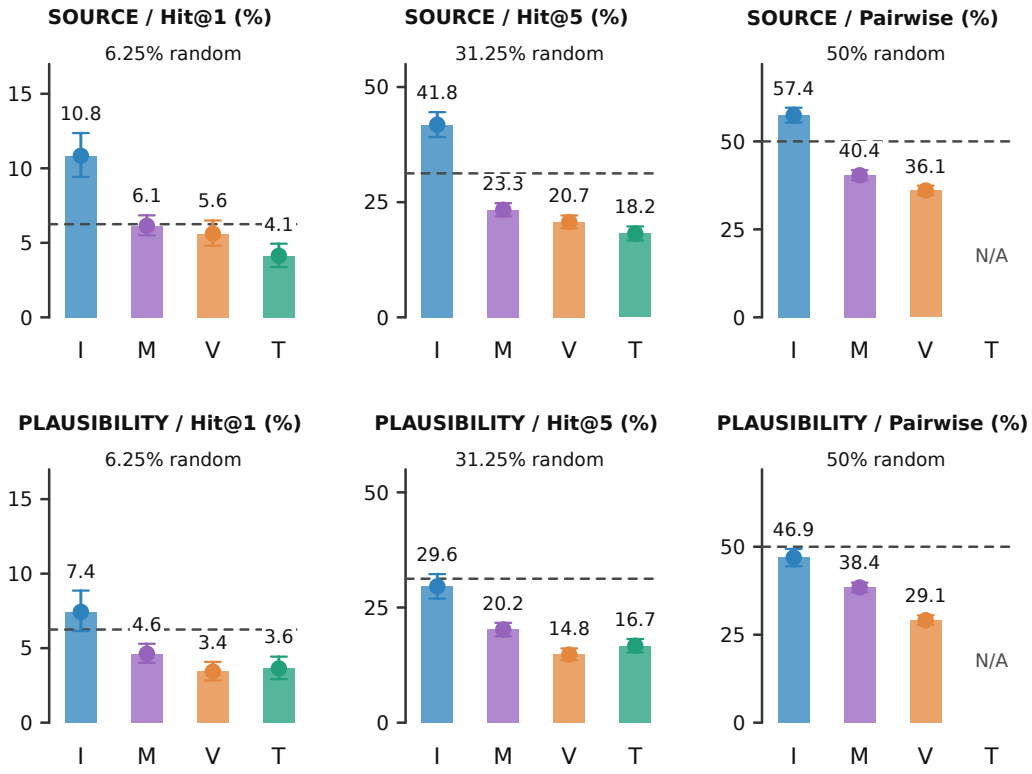


Figure 10: The complete detailed study on 237 items: two targets, four procedures, and three endpoints. I, M, V, and T denote Isolated, Marked, Vector, and the Top-5 action. Bars start at zero. Whiskers are marginal 95% item-bootstrap intervals; dashed lines show each endpoint's chance reference. SOURCE and PLAUSIBILITY both score against source identity. Top-5 does not define a complete pairwise relation.

Table 16: SOURCE model cells on 237 items (%). Relation validity requires all scores for the affected numeric presentation, or a complete Top-5 action.

Model	Procedure	Hit@1	Hit@5	Pairwise	Relation valid
Qwen2.5-32B	Isolated	6.11	33.51	52.71	100.00
	Marked	2.67	11.65	28.21	100.00
	Vector	7.75	19.48	29.36	100.00
	Top-5 action	5.17	14.35	–	100.00
Falcon3-10B	Isolated	14.22	46.44	60.21	100.00
	Marked	5.48	30.04	48.76	100.00
	Vector	7.69	25.41	43.46	98.84
	Top-5 action	4.54	21.94	–	99.89
Gemma2-27B	Isolated	20.60	55.90	67.37	100.00
	Marked	5.57	21.39	35.77	100.00
	Vector	4.09	11.94	23.01	100.00
	Top-5 action	1.48	10.13	–	100.00
Granite3.3-8B	Isolated	9.39	34.47	50.75	97.05
	Marked	8.89	27.64	45.08	100.00
	Vector	2.85	21.69	40.86	97.15
	Top-5 action	6.75	32.70	–	97.89
Llama3.2-3B	Isolated	5.80	36.16	52.93	100.00
	Marked	5.13	24.51	43.33	99.89
	Vector	4.32	25.60	44.57	97.78
	Top-5 action	2.74	16.77	–	100.00
Mistral-Small-24B	Isolated	8.90	44.45	60.69	100.00
	Marked	9.11	24.87	41.11	100.00
	Vector	6.95	20.03	35.03	100.00
	Top-5 action	4.11	13.19	–	99.58

Table 17: PLAUSIBILITY model cells on the same 237 items (%). The source role remains the scoring label; these are not independent plausibility labels.

Model	Procedure	Hit@1	Hit@5	Pairwise	Relation valid
Qwen2.5-32B	Isolated	4.58	28.97	47.85	100.00
	Marked	1.33	6.07	23.00	100.00
	Vector	3.06	8.44	18.10	100.00
	Top-5 action	3.27	9.70	–	100.00
Falcon3-10B	Isolated	6.85	28.50	45.41	100.00
	Marked	3.68	24.08	44.76	100.00
	Vector	4.22	15.13	31.89	100.00
	Top-5 action	4.01	21.41	–	99.26
Gemma2-27B	Isolated	13.99	41.25	57.16	100.00
	Marked	3.19	13.50	30.02	100.00
	Vector	2.43	9.94	20.33	100.00
	Top-5 action	2.85	10.34	–	100.00
Granite3.3-8B	Isolated	3.40	19.72	38.21	94.09
	Marked	5.07	26.32	45.26	99.89
	Vector	4.18	21.48	38.78	97.47
	Top-5 action	5.49	30.59	–	97.26
Llama3.2-3B	Isolated	3.75	17.47	36.85	100.00
	Marked	4.84	25.33	44.09	99.89
	Vector	4.22	25.51	44.93	97.57
	Top-5 action	2.00	14.77	–	99.79
Mistral-Small-24B	Isolated	12.02	41.58	56.12	100.00
	Marked	9.66	25.98	43.41	100.00
	Vector	2.48	8.21	20.70	100.00
	Top-5 action	4.22	13.29	–	100.00

D.2 Recovery across selection budgets

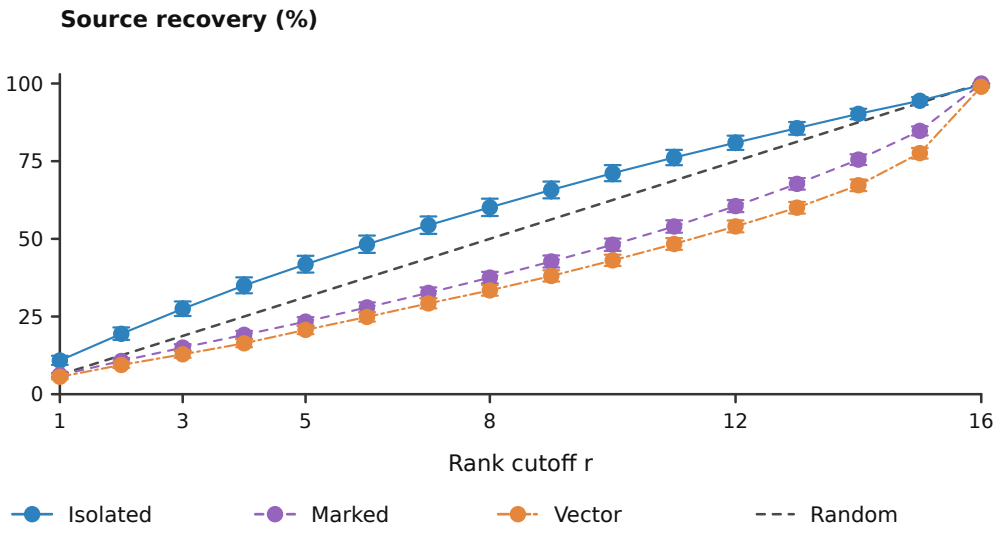


Figure 11: How often does the source survive a larger selection budget? Top-r recovery on the 237-item study. Ties receive fractional credit; invalid relations give zero at every r, including 16. Whiskers are pointwise 95% intervals, not a simultaneous band. Lines join tested thresholds without smoothing.

Figure 11 shows the full source-recovery distribution. Table 18 gives its coordinates for exact lookup. The item endpoint is Equation 4; order and model averaging follow the detailed study's weights.

Table 18: All sixteen source-recovery cutoffs on the 237-item study. Each entry is an estimate and pointwise 95% interval (%). Invalid relations receive zero even at $r = 16$. Coordinates match Figure 11.

r	Isolated	Marked	Vector
1	10.84 [9.42, 12.36]	6.14 [5.50, 6.84]	5.61 [4.81, 6.50]
2	19.43 [17.49, 21.48]	10.68 [9.83, 11.60]	9.41 [8.47, 10.44]
3	27.48 [25.18, 29.85]	15.00 [13.93, 16.11]	12.84 [11.72, 14.05]
4	35.01 [32.48, 37.60]	19.10 [17.86, 20.38]	16.38 [15.12, 17.69]
5	41.82 [39.15, 44.54]	23.35 [21.93, 24.79]	20.69 [19.33, 22.12]
6	48.24 [45.49, 51.05]	27.93 [26.32, 29.54]	24.87 [23.39, 26.42]
7	54.39 [51.60, 57.21]	32.61 [30.88, 34.36]	29.22 [27.62, 30.88]
8	60.13 [57.35, 62.93]	37.53 [35.67, 39.37]	33.40 [31.71, 35.13]
9	65.74 [63.04, 68.43]	42.71 [40.78, 44.64]	38.05 [36.27, 39.87]
10	71.14 [68.56, 73.73]	48.10 [46.13, 50.08]	43.07 [41.28, 44.89]
11	76.17 [73.71, 78.63]	53.98 [51.99, 55.97]	48.34 [46.48, 50.23]
12	80.95 [78.63, 83.21]	60.54 [58.61, 62.48]	54.01 [52.10, 55.91]
13	85.62 [83.53, 87.60]	67.70 [65.85, 69.55]	60.04 [58.16, 61.88]
14	90.23 [88.51, 91.86]	75.50 [73.78, 77.20]	67.25 [65.37, 69.09]
15	94.44 [93.13, 95.64]	84.75 [83.29, 86.17]	77.60 [75.83, 79.24]
16	99.51 [99.16, 99.86]	99.98 [99.95, 100.00]	98.96 [98.61, 99.28]

D.3 Order stability and adjacent contrasts

Winner-set agreement compares the nonempty maximum-score sets for all six unordered pairs of four orders. Unanimity requires the same unique winner in every order. A failed required output gives zero. Thus neither an all-tied score vector nor a missing result can satisfy unique-winner unanimity. Isolated has no all-candidate order and is not assigned an order-stability value.

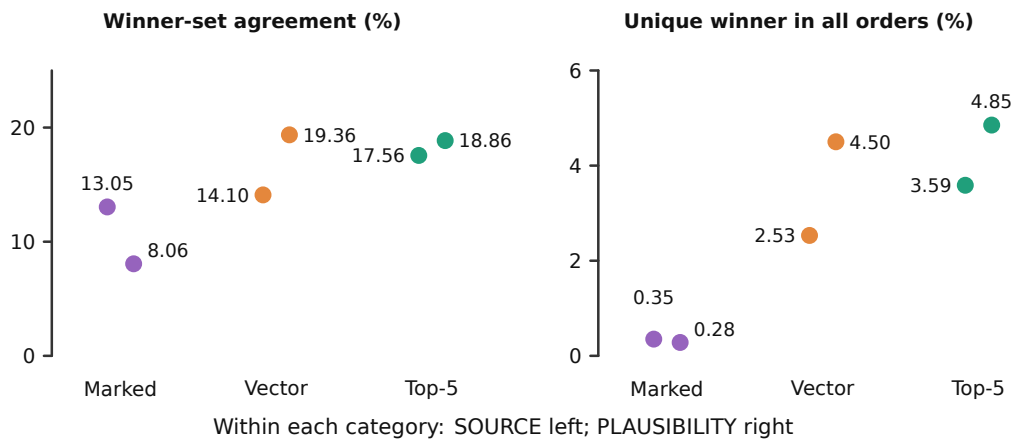


Figure 12: Order sensitivity on the detailed 237-item study. Points are descriptive panel means, with no inferred intervals. Within each procedure, the left point denotes SOURCE and the right point PLAUSIBILITY. Agreement and unanimity answer different questions and use different vertical ranges. Procedures are categorical and are not connected.

Table 19: Detailed-study order stability (%). Winner-set agreement and unanimity match Figure 12. Top-5 Jaccard compares the boundary-inclusive top-five sets across order pairs; broad score ties can make this overlap high without a stable unique winner.

Target	Procedure	Winner agreement	Top-5 Jaccard	Unanimity
Source	Marked	13.05	69.00	0.35
Source	Vector	14.10	35.14	2.53
Source	Top-5	17.56	31.42	3.59
Plausibility	Marked	8.06	67.69	0.28
Plausibility	Vector	19.36	37.55	4.50
Plausibility	Top-5	18.86	32.57	4.85

Table 20: Exact adjacent Hit@1 contrasts in Figure 13. Differences and marginal 95% paired-item intervals are in percentage points.

Target	Contrast	Estimate (pp)	95% CI
Source	Marked − Isolated	-4.70	[-6.03, -3.43]
Source	Vector − Marked	-0.53	[-1.37, 0.32]
Source	Top-5 − Vector	-1.48	[-2.29, -0.68]
Plausibility	Marked − Isolated	-2.81	[-4.02, -1.66]
Plausibility	Vector − Marked	-1.20	[-1.89, -0.51]
Plausibility	Top-5 − Vector	0.21	[-0.59, 1.03]

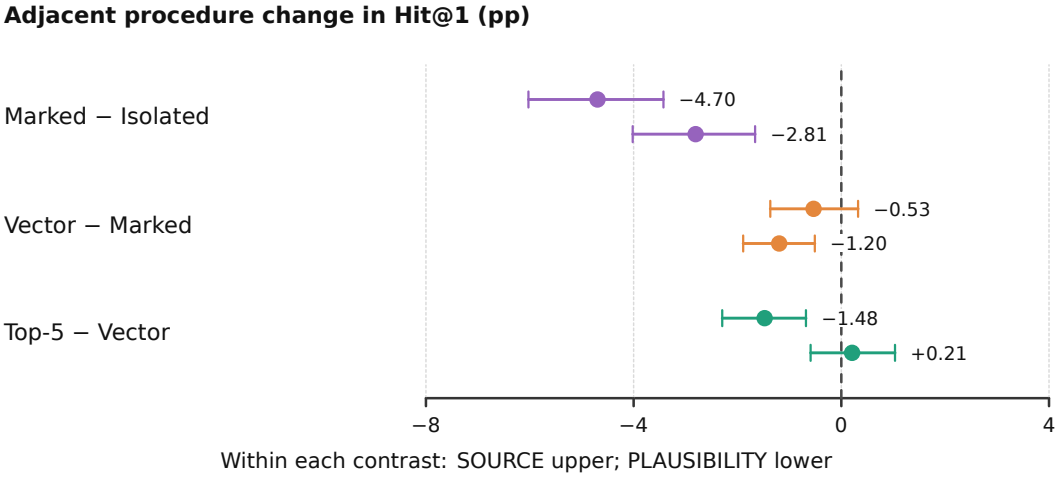


Figure 13: Adjacent changes in Hit@1 on the same 237 items. Each difference follows the subtraction shown on its axis, with a marginal 95% paired-item interval. SOURCE is the upper row and PLAUSIBILITY the lower row in each comparison. Each comparison changes the complete requested procedure.

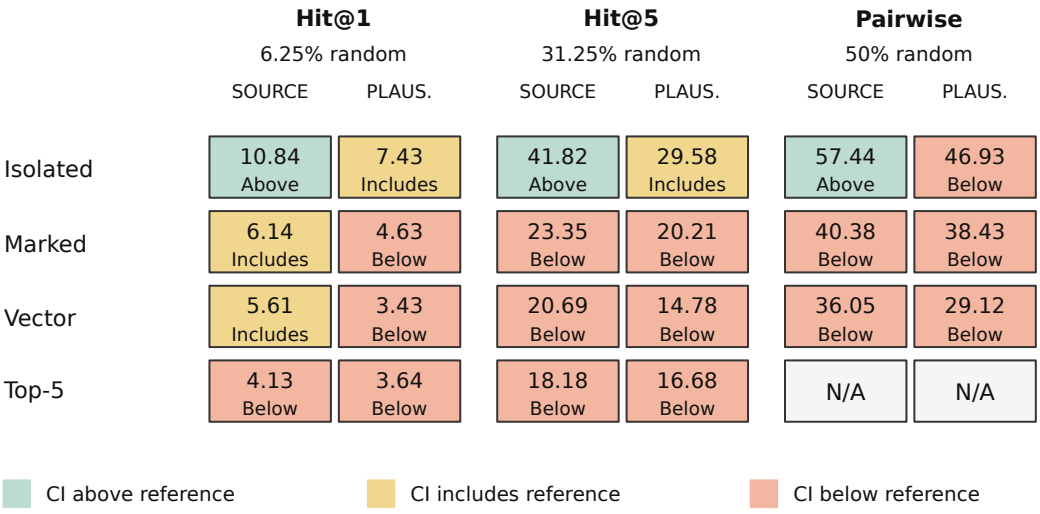


Figure 14: Where are the detailed-study cells relative to chance? Each cell prints its absolute percentage. Shading and the labels Above, Includes, and Below report whether that cell's marginal 95% interval lies above, includes, or lies below its own endpoint's random reference. This categorical map uses no common continuous effect scale across endpoints and makes no simultaneous inference.

D.4 Post hoc ranking examples

The three cases were selected post hoc from existing outputs, using a rule fixed before item-level inspection. One case per predefined ranking category was chosen by proximity to the category's median selection statistic, with a deterministic hash tie-break. Selection used the study-specific readout before semantic harmonization; the figures use the common semantic policy (Appendix B.1). The main example came from the stable-reversal category. Its numeric presentations are complete

and the panel relation reverses. Figure 2 displays every model's rank blocks; a 1–16 tie supplies fractional top-one and half-credit pairwise recovery, not a strict last-place or first-place ranking.

Figure 15 shows two further examples of low recovery and order-sensitive rankings. All three cases are post hoc illustrations of individual pools, with descriptive panel values rather than estimates of pattern prevalence. The displays omit item text and persistent identifiers.

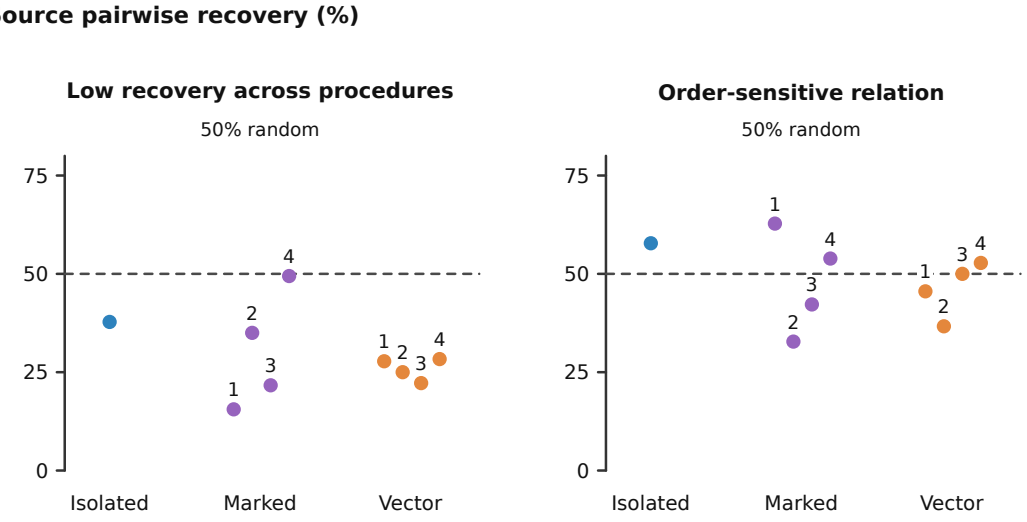


Figure 15: Two further post hoc examples. The vertical axis is six-model source–alternative recovery (%); 50% is the binary chance reference. Each small number labels a case-local order, not a different item. Isolated supplies one assembled relation; Marked and Vector each supply four. Nominal procedures are not connected, and no sampling interval is attached to an individual illustration.

D.5 Direct choices

Table 21: Direct generated choices on 237 items: percent accuracy with 95% item bootstrap intervals. Both target wordings use both source positions.

Model	SOURCE Direct (%)	Native Direct (%)
Qwen2.5-32B	18.21 [16.06, 20.45]	20.52 [18.37, 22.73]
Falcon3-10B	22.91 [20.51, 25.34]	18.26 [16.16, 20.46]
Gemma2-27B	36.01 [32.95, 39.10]	41.45 [38.33, 44.60]
Granite3.3-8B	30.70 [28.42, 32.98]	16.72 [14.87, 18.69]
Llama3.2-3B	45.91 [43.91, 47.89]	36.37 [34.77, 38.02]
Mistral-Small-24B	27.37 [24.54, 30.32]	27.64 [24.96, 30.34]
Panel	30.19 [28.12, 32.31]	26.83 [25.01, 28.73]

All Direct choices are recoverable under the semantic readout; literal compliance is 87.25%. Its SOURCE instruction explicitly asks for the source-derived hypothesis. The native wording follows the benchmark's quality-oriented pairwise question, while the scoring label remains source identity. These generated choices are not restricted first-token logits. The half-credit score-derived relation and strict generated-choice accuracy are distinct endpoints.

D.6 Nested candidate sets

The candidate-count design fixes one source–anchor pair per item. Candidate sets of sizes two, four, eight, and sixteen are nested, with two fixed expansion chains and balanced orders. Orders

are averaged within item/model/size/chain, then chains, then the six models. The anchor relation is retained at every size. The $k = 2$ condition is still a score-vector request, not Isolated or Direct.

Table 22: Complete candidate-count curve for the focal source–anchor relation. The primary endpoint contrast uses paired items, not the marginal intervals shown for the levels.

Candidates k	Focal relation (%)	95% CI
2	39.43	[36.48, 42.56]
4	35.45	[32.57, 38.41]
8	35.19	[32.79, 37.67]
16	37.17	[35.30, 39.08]

Table 23: Candidate-count heterogeneity. All six model-specific $k = 16$ versus $k = 2$ differences (pp) with marginal 95% intervals are descriptive.

Model	Difference (pp)	95% CI
Falcon3-10B	-3.43	[-6.86, -0.11]
Gemma2-27B	-12.18	[-16.35, -8.07]
Granite3.3-8B	14.50	[9.70, 19.20]
Llama3.2-3B	0.79	[-2.74, 4.32]
Mistral-Small-24B	-4.59	[-10.18, 0.90]
Qwen2.5-32B	-8.65	[-13.34, -4.17]

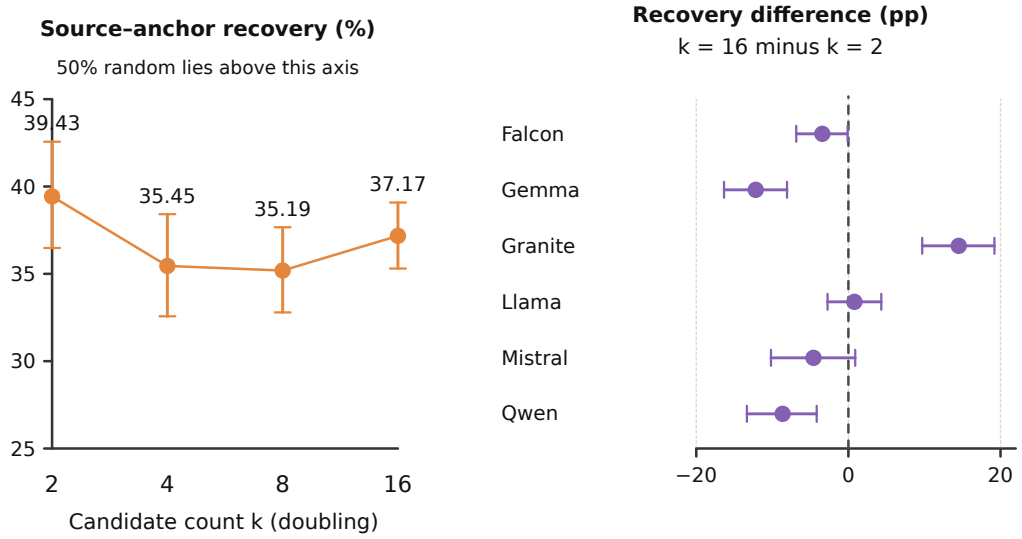


Figure 16: The fixed source–anchor relation across nested pool sizes and model-specific size effects. Left: all four measured levels with marginal 95% intervals; lines join the tested sizes without fitting. Right: each checkpoint's $k = 16$ versus $k = 2$ difference with a paired 95% interval. The focal relation, two expansion chains, and order averaging are held fixed.

The primary interval [-4.61, 0.04] pp is wholly within the specified $[-5, 5]$ pp region. This criterion, rather than failure to reject zero, supports the study's practical-equivalence classification. Intermediate sizes and named models do not follow a common monotonic curve.

Separate JudgeBench expansion arm. On the 91-question exact cross-generator intersection, a four-response pool contains GPT-correct, GPT-incorrect, Claude-correct, and Claude-incorrect responses. All twenty-four orders are compared with the corresponding binary Vector outputs. This changes the response construction as well as the context size, and is a secondary correctness-transport diagnostic.

Table 24: Candidate-count JudgeBench arm. Contrast rows are percentage-point changes, level rows are percentages. Only the two generator-specific focal contrasts form the specified Holm family; the equal-weight contrast is separate. No level is assigned a directional contrast interpretation.

Focal relation or contrast	Estimate	95% CI	Adjusted p
GPT, $k = 4$ versus $k = 2$	2.83	[0.32, 5.30]	0.0592
Claude, $k = 4$ versus $k = 2$	2.53	[-0.18, 5.41]	0.0686
Equal weight, $k = 4$ versus $k = 2$	2.68	[1.06, 4.31]	0.0018
GPT, $k = 2$	58.61	[54.35, 62.87]	–
GPT, $k = 4$	61.44	[58.56, 64.18]	–
Claude, $k = 2$	60.49	[56.41, 64.56]	–
Claude, $k = 4$	63.01	[60.32, 65.63]	–
Equal weight, $k = 2$	59.55	[56.73, 62.32]	–
Equal weight, $k = 4$	62.23	[59.81, 64.51]	–

Neither generator-specific contrast passes its Holm threshold. The GPT unadjusted 95% interval excludes zero, but its adjusted p is 0.0592; Claude remains inconclusive.

E CONTEMPORARY MODELS AND EXTERNAL TASKS

E.1 CONTEMPORARY ENDPOINT DETAILS

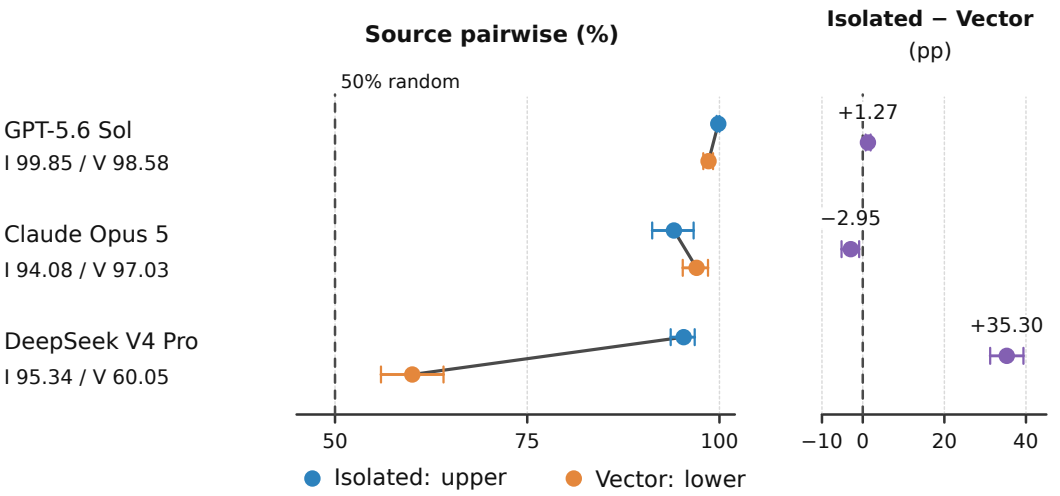


Figure 17: Absolute recovery and procedure gaps for three contemporary endpoints on 237 items. Points have 95% item-bootstrap intervals. The absolute-level axis begins at 45%, retains the 50% chance reference, and shows every interval in full. The separate difference axis marks zero. No cross-endpoint average is formed. Claude's full-population gap includes invalid outputs; the 223-item complete-valid comparison is conditional and reported separately.

Table 25: Contemporary SOURCE endpoint results on 237 items. Levels are percentages, differences percentage points. Intervals are the primary item-bootstrap 95% intervals.

Endpoint	Isolated	Vector	$I - V$ (pp), 95% CI
GPT-5.6 Sol	99.85	98.58	1.27 [0.71, 1.93]
Claude Opus 5	94.08	97.03	-2.95 [-5.20, -0.89]
DeepSeek V4 Pro	95.34	60.05	35.30 [31.23, 39.38]

Each endpoint scores sixteen candidates independently and one vector in each of four orders per item. Requests concatenate the unchanged system instruction, two line breaks, and unchanged user template into one user message. They use plain-text output without tools, web access, retrieval, conversation history, assistant prefill, structured-output mode, or provider fallback. The scalar and vector output caps are sixteen and 256 tokens.

The endpoint identities are `gpt-5.6-sol`, `claude-opus-5`, and `deepseek-v4-pro`, the last mapped at execution to `DeepSeek-V4-Pro-0813`. GPT uses reasoning effort `none`; Claude disables thinking with supported low effort; DeepSeek disables thinking and uses temperature zero. The requests are single provider-served realizations.

All GPT outputs are literal-valid. Claude has 65 unrecoverable scalar outputs across twelve assembled Isolated pools, and nineteen invalid Vector orders across seven items. Under the primary fail-closed rule, an incomplete Isolated pool receives zero credit for all its relations; an invalid Vector order receives zero before the four orders are averaged. All 237 items and all scheduled orders remain in the primary denominator. As a supplementary conditional description, the 223 items with a complete Isolated pool and all four valid Vector orders have Isolated/Vector recovery of 99.09%/98.99%, a gap of 0.10 pp. Conditioning on joint completeness selects outputs and does not replace the planned-population result.

Ten DeepSeek vectors use complete JSON fences: 100% of its Vector outputs are recoverable under semantic readout, compared with 98.95% literal compliance. These are parsing rates. The cluster-

aware 90% gap intervals are [0.78, 1.81] for GPT, [-5.02, -1.14] for Claude, and [31.75, 38.83] pp for DeepSeek. They distinguish a small GPT gap, a Claude interval crossing the -5 pp boundary, and a large positive DeepSeek gap.

E.2 JUDGEBENCH EVALUATOR BRIDGE

Table 26: External evaluator boundaries, equally weighted over the six fixed models. JudgeBench measures binary correctness; TruthHypo measures three-way biomedical relation recovery. Accuracy and chance are in percent. Appendices E.2 and E.4 give intervals and model-specific cells.

Dataset (items)	Chance	Isolated	Marked	Vector	Direct
JudgeBench GPT (350)	50.00	58.38	55.21	56.93	51.05
JudgeBench Claude (269)	50.00	55.17	53.47	54.12	50.81
TruthHypo (800)	33.33	35.43	34.90	38.37	39.52

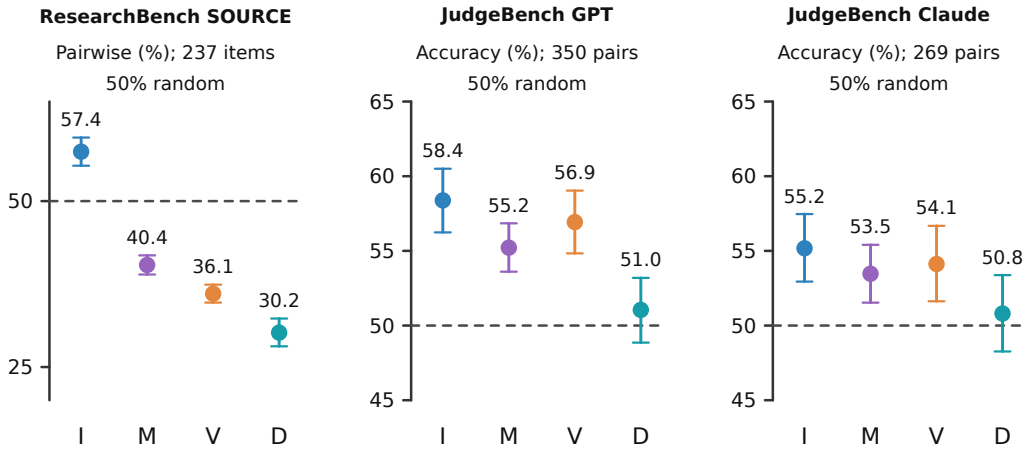


Figure 18: Score-derived relations and generated Direct choices across tasks. All panels show the six-model mean and 95% item-bootstrap intervals, against 50% chance. The ResearchBench panel concerns source identity; JudgeBench concerns designated correct versus incorrect responses. I, M, V, and D are Isolated, Marked, Vector, and Direct. Vertical ranges differ and are labeled. Direct's exact-choice endpoint is distinct from half-credit score-derived relations.

The bridge compares the six fixed judges on a designated correct response against an incorrect response to the same question. Each scoring relation uses half-credit ties; an unrecoverable required output gives zero. The specified item-weighted aggregate retains category-stratified paired resampling. Its 619 pairs exclude one byte-identical pair that the all-candidate distinct-response renderer cannot represent.

Choice accuracy and output validity. Table 29 separates correct, wrong, and invalid Direct outputs. Each pair has two equally weighted orders; averaging their zero-or-one correctness and then averaging items equals correct calls divided by planned calls here, because every item has both outputs. The main panel still weights all six judges equally. Valid-only accuracy conditions on accepted outputs and is a supplementary description, subject to validity selection; it does not replace the primary estimand.

Table 27: JudgeBench GPT-generator split: model and panel accuracy with 95% intervals (%), on 350 pairs.

Model	Isolated	Marked	Vector	Direct
Qwen2.5-32B	65.57 [61.86, 69.14]	62.00 [58.57, 65.36]	64.00 [60.29, 67.71]	64.43 [60.57, 68.29]
Falcon3-10B	58.00 [54.43, 61.43]	52.21 [50.00, 54.50]	54.71 [51.86, 57.64]	55.86 [51.86, 59.86]
Gemma2-27B	61.86 [58.43, 65.43]	57.43 [54.57, 60.29]	59.86 [56.29, 63.36]	59.57 [56.00, 63.14]
Granite3.3-8B	53.00 [51.29, 54.86]	51.00 [49.36, 52.64]	51.21 [49.36, 53.14]	50.71 [48.86, 52.57]
Llama3.2-3B	53.29 [50.57, 56.00]	50.21 [48.50, 51.86]	51.79 [49.14, 54.36]	16.57 [14.14, 19.00]
Mistral-Small-24B	58.57 [54.86, 62.14]	58.43 [55.50, 61.36]	60.00 [56.79, 63.21]	59.14 [55.43, 62.71]
Panel	58.38 [56.24, 60.50]	55.21 [53.61, 56.85]	56.93 [54.83, 59.04]	51.05 [48.86, 53.19]

Table 28: JudgeBench Claude-generator split: model and panel accuracy with 95% intervals (%), on 269 pairs.

Model	Isolated	Marked	Vector	Direct
Qwen2.5-32B	58.92 [54.83, 63.01]	56.69 [53.07, 60.32]	59.94 [55.30, 64.59]	58.18 [53.72, 62.64]
Falcon3-10B	56.88 [53.72, 60.04]	53.25 [50.84, 55.67]	54.09 [50.56, 57.43]	57.81 [53.35, 62.08]
Gemma2-27B	55.76 [52.23, 59.29]	56.51 [53.16, 59.76]	57.16 [53.35, 60.87]	55.58 [51.30, 59.67]
Granite3.3-8B	52.60 [50.93, 54.28]	48.79 [46.75, 50.74]	54.09 [51.67, 56.51]	51.86 [49.44, 54.09]
Llama3.2-3B	54.09 [50.93, 57.25]	50.93 [48.98, 52.88]	46.93 [43.77, 50.00]	28.25 [25.28, 31.23]
Mistral-Small-24B	52.79 [49.07, 56.51]	54.65 [51.12, 58.18]	52.51 [48.79, 56.13]	53.16 [49.44, 57.06]
Panel	55.17 [52.94, 57.47]	53.47 [51.53, 55.41]	54.12 [51.63, 56.68]	50.81 [48.27, 53.38]

Table 29: JudgeBench Direct call accounting under the semantic readout. Plan/done/miss means planned, completed, and technically missing calls. Main accuracy retains invalid outputs as zero; valid-only accuracy divides correct by correct plus wrong calls. GPT/Claude identify response generators, not the judges. All rates are percentages.

Judge	Plan/done/miss	Correct/wrong/invalid	Valid (%)	Main (%)	Valid-only (%)
GPT-generated responses (350 pairs)					
Qwen2.5-32B	700/700/0	451/249/0	100.00	64.43	64.43
Falcon3-10B	700/700/0	391/285/24	96.57	55.86	57.84
Gemma2-27B	700/700/0	417/283/0	100.00	59.57	59.57
Granite3.3-8B	700/700/0	355/335/10	98.57	50.71	51.45
Llama3.2-3B	700/700/0	116/128/456	34.86	16.57	47.54
Mistral-Small-24B	700/700/0	414/286/0	100.00	59.14	59.14
Claude-generated responses (269 pairs)					
Qwen2.5-32B	538/538/0	313/225/0	100.00	58.18	58.18
Falcon3-10B	538/538/0	311/226/1	99.81	57.81	57.91
Gemma2-27B	538/538/0	299/239/0	100.00	55.58	55.58
Granite3.3-8B	538/538/0	279/250/9	98.33	51.86	52.74
Llama3.2-3B	538/538/0	152/147/239	55.58	28.25	50.84
Mistral-Small-24B	538/538/0	286/252/0	100.00	53.16	53.16

Llama's 456 and 239 invalid calls lower its primary accuracy and hence the panel mean. Of these, 434 and 224 respectively have an EOS finish marker; the remainder have the saved `length_or_other` marker, which does not distinguish truncation from other termination. Invalidity means failure to satisfy the readout in Appendix B.1, including its case-sensitive choice forms. Some rejected outputs contain recognizable choice letters; the category does not imply that every response is meaningless. These records do not identify why the model produced them.

E.3 RESPONSE-ONLY CLASSIFIER

The separate supervised diagnostic includes 350 GPT-4o and 270 Claude-3.5-Sonnet pairs. Both responses of a pair share a fold. Connected components of nonmissing original question identity, exact question bytes, and exact response bytes define 528 dependency groups, including all 91 exact cross-generator question overlaps. A missing identifier never links otherwise unrelated rows. Models and estimates are generator-specific.

The classifier family uses the same 23 surface features, word/character TF-IDF, grouped five-outer/four-inner selection, balanced L2 logistic regression, and regularization grid as the source diagnostic (Appendix C.2). Word and character feature thresholds, solver forms, tolerance, and iteration cap are the same; the seed is 20260903. Inner selection maximizes binary pair accuracy and breaks ties using the smallest C. Position, pair/question identity, category, generator, evaluator output, and gold metadata are excluded from features. The byte-identical Claude pair has a valid binary gold label but necessarily tied text scores; its abstention receives zero. All 620 pairs remain in the denominator.

Table 30: Primary response-only correctness diagnostics. Both generator-specific 95% intervals include binary chance (50%).

Generator	Pairs	Accuracy (%)	95% cluster CI	Holm p
GPT-4o	350	52.29	[47.14, 57.43]	0.7983
Claude-3.5-Sonnet	270	52.59	[46.67, 58.36]	0.7983

E.4 TRUTHHYPO

The official test relations cover Chemical–Gene, Disease–Gene, and Gene–Gene tasks. Chemical–Gene and Gene–Gene labels are `positive_correlate`, `negative_correlate`, and `no_relation`. Disease–Gene uses `stimulate`, `inhibit`, and `no_relation`. These are the benchmark's relation labels. Our English candidate verbalizations expand the Disease–Gene labels as disease-promotes-gene and disease-suppresses-gene statements; that directional wording is an author-defined prompt rendering, not a separately validated biological direction supplied by the benchmark. The correlation labels denote annotated associations, and `no_relation` denotes absence of the annotated relation. Recovery measures agreement with these labels under the stated verbalization, not independent biological validation. The 800-item subset is entity-disjoint. Table 31 gives its task and label counts. Isolated scores three labels independently; Marked scores each while displaying all three under a fixed order and its reverse; Vector generates all three scores in each order; Direct selects one winner. Numeric maximum-score ties receive fractional Hit@1 credit. Isolated requires three calls, Marked six, Vector two, and Direct two per item/model. Outcomes are averaged over the two orders before higher aggregation.

Table 31: TruthHypo evaluation population and descriptive task-specific majority references. The first two columns denote positive/negative correlation for Chemical–Gene and Gene–Gene, and stimulate/inhibit for Disease–Gene. Choosing the most frequent evaluated label within each task yields 327/800 (40.875%) under the primary item-equal weighting. This uses the evaluation labels; it is not a prospectively selected trained baseline. Uniform random choice remains 33.33% for each three-label item.

Task	Positive/stimulate	Negative/inhibit	No relation	N	Majority (%)
Chemical & Gene	130	189	159	478	39.54
Disease & Gene	41	30	35	106	38.68
Gene & Gene	97	47	72	216	44.91
Item-weighted reference	Task-specific majority choice			800	40.88

Table 32: TruthHypo absolute levels and 95% intervals (%). These are descriptive accuracies with a 33.33% uniform-random reference; Table 31 gives the distinct majority reference.

Procedure	Accuracy (%)	95% CI
Isolated	35.43	[34.84, 36.03]
Marked	34.90	[34.39, 35.39]
Vector	38.37	[37.57, 39.16]
Direct	39.52	[38.68, 40.36]

Table 33: TruthHypo model-specific Isolated-minus-Vector differences (pp), with marginal 95% intervals. These descriptive effects need not share the panel's practical-equivalence classification.

Model	$I - V$ (pp)	95% CI
Qwen2.5-32B	-6.22	[-7.73, -4.73]
Falcon3-10B	-4.96	[-7.00, -2.90]
Gemma2-27B	-5.38	[-7.25, -3.53]
Granite3.3-8B	-2.22	[-4.06, -0.31]
Llama3.2-3B	0.78	[-1.12, 2.71]
Mistral-Small-24B	0.40	[-3.00, 3.85]

The fixed panel uses greedy generation, bfloat16, no quantization, batch size one, and output caps 32/32/96/32 for Isolated/Marked/Vector/Direct. Isolated minus Vector is -2.93 pp (95% CI [-3.86, -2.02]), favoring Vector on this biomedical relation task. Readout rules and sensitivity are specified in Appendix B.1.

F CONTROLLED CANDIDATE-CONTEXT STUDY

F.1 POPULATION, MATERIAL CONSTRUCTION, AND FIXED ORDER

The study scores all sixteen original candidates of the established 237-item subset separately under four contexts. It preserves the background, question, target candidate bytes, SOURCE scoring instruction, and scalar action across conditions. All requests place the target before the additional material, then place the common SOURCE suffix after the material. Condition labels, source roles, and private candidate identities are not shown to the model.

A leaves the additional-context body empty. B fills fifteen slots with researcher-authored ordinary-life sentences from a fixed 79-sentence bank. Each slot uses a deterministic cyclic starting point and the nearest-length prefix of complete sentences, capped at 180 sentences. C uses whole scientific candidate texts from the remaining 1,076 common-support items. For each target, donors from its own source cluster or duplicate group are excluded. Normalized exact-text collisions with the target subset are also excluded. This is target-relative exclusion, not global source-disjointness of the donor corpus. Cross-question texts need not be semantically noncompeting. The donor mapping is fixed across models and conditions.

D contains the other fifteen candidates from the same item. Candidate-text SHA-256 ordering is rotated to put the target first and then the target is removed. Across the sixteen target requests, each other candidate occupies each of the fifteen context slots once. B and C inherit these slot positions. This is one fixed order policy, rather than a distribution of independently sampled permutations.

Approximate matching. Each B/C slot targets the corresponding D slot's length under `cl100k_base`, with tolerance $\max(8, 0.05\,L_D)$ tokens. Complete text blocks are retained without truncation. Among 56,880 slots per condition, 85 B slots and 15 C slots fall outside this planning tolerance; the B exceptions reach the fixed sentence cap. Under the actual Qwen text tokenizer, 3,277 B and 2,251 C slots exceed the inherited tolerance; under Sol's text-tokenizer proxy the counts are 1,564 and 661. These differences were retained, not rematched after observing outcomes.

Table 34 reports actual full-request input counts from saved usage, rather than planning totals. Its exception threshold $\max(120, 0.05\,L_D)$ is a descriptive length check, not a scientific endpoint. No request was truncated. Matching is close, especially for C, but does not make the two contexts token-identical.

Table 34: Actual whole-request input matching to D, on 3,792 requests per condition and model. All lengths and differences are tokens. Exceptions count paired absolute differences above the descriptive tolerance.

Model	Condition	Mean input	D mean	Mean $\lvert\Delta\rvert$	Max $\lvert\Delta\rvert$	Exceptions
Qwen2.5-32B	B	4061.12	4124.35	64.71	1376	45
Qwen2.5-32B	C	4128.31	4124.35	33.36	165	0
GPT-5.6 Sol	B	4010.49	3999.21	44.23	1367	45
GPT-5.6 Sol	C	3998.42	3999.21	26.22	292	0

F.2 GENERATION, READOUT, AND INFERENCE

Qwen2.5-32B-Instruct uses its fixed checkpoint and native chat template, bfloat16 on A100 80GB hardware, vLLM 0.10.2, Transformers 4.55.4, PyTorch 2.8.0+cu128, temperature zero, top-$p = 1$, and a 32-token output cap. Its largest actual input has 8,120 tokens within the 32,768-token context. GPT-5.6 Sol uses the Responses interface, reasoning effort `none`, temperature zero, a 32-token cap, default service tier, no tools, `store=false`, and disabled truncation. Both request a structured JSON object with one integer `score` from 0 to 10000 and no additional properties. The literal instruction, system message, and assembly function appear in Appendix G.5.

The parser requires an object with exactly one `score` key. Duplicate or extra keys, missing keys, prose, arrays, strings, booleans, noninteger numbers, and nonfinite values are invalid. There is no code-fence or selection-action recovery. A returned invalid source score makes all fifteen relations

U; a returned invalid alternative score affects its own relation. This relation-level invalid policy differs from the complete-pool policy used in the procedure comparisons. Technical missingness is separate and prevents estimation for an incomplete model matrix. In the completed study every planned request is valid and present, so neither rule removes a relation.

For each model and condition c, item recovery is $a_i(c) = (W_i(c) + T_i(c)/2)/15$. The primary contrast is $\Delta_{C-D} = 237^{-1} \sum_i [a_i(C) - a_i(D)]$; supporting contrasts are A−B, A−C, A−D, and B−D. The study uses equal item weights. Dependency groups are the transitive union of source and duplicate links on the full common-support set, restricted to the target items. The 224 groups are resampled synchronously across the four conditions, preserving each selected group's items and hence item weighting. The 10,000 draws use Python MT19937 seed 26091302; percentile endpoints use the nearest order statistic at index $\mathrm{round}(p(B-1))$. These marginal 95% intervals are reported without adding a new Holm or sign-test family. Models are reported separately.

The ± 5 pp range was selected before these outputs as a substantive effect range. It is neither the 50% chance reference nor a significance threshold. Qwen's interval lies inside that range, but its ties prevent interpreting this as useful discrimination or equivalence of unobserved ability. Donors, templates, orders, and models remain fixed in these intervals. B−D and C−D share their comparison condition and are dependent evidence.

F.3 Absolute results, paired contrasts, and descriptive behavior

Each model completes 15,168 valid logical requests: 3,792 per condition, covering 237 items and 3,555 source–alternative relations. These requests are not the resampling units. Tables 35 and 36 report all conditions and specified contrasts.

Table 35: Context-control results. Recovery and ties are percentages; intervals are 95% dependency-cluster intervals. W, T, and L are relation counts, not independent samples; U and technical missingness are zero.

Model	Condition	Recovery (%)	95% CI	Ties (%)	W	T	L
Qwen2.5-32B	A	51.67	[50.48, 52.75]	91.25	215	3244	96
Qwen2.5-32B	B	50.98	[49.89, 52.19]	92.57	167	3291	97
Qwen2.5-32B	C	49.90	[48.93, 50.85]	94.18	100	3348	107
Qwen2.5-32B	D	51.38	[49.76, 52.96]	88.64	251	3151	153
GPT-5.6 Sol	A	99.86	[99.73, 99.96]	0.06	3549	2	4
GPT-5.6 Sol	B	99.72	[99.44, 99.93]	0.17	3542	6	7
GPT-5.6 Sol	C	99.51	[99.04, 99.86]	0.37	3531	13	11
GPT-5.6 Sol	D	87.97	[85.53, 90.32]	5.37	3032	191	332

Table 36: All specified paired context contrasts (pp). C−D is primary; the other four are supporting contrasts. Intervals are marginal 95% dependency-cluster percentiles.

Model	Contrast	Difference (pp)	95% CI
Qwen2.5-32B	A−B	0.69	[-0.72, 1.98]
Qwen2.5-32B	A−C	1.77	[0.65, 2.78]
Qwen2.5-32B	A−D	0.30	[-1.38, 1.95]
Qwen2.5-32B	B−D	-0.39	[-2.17, 1.39]
Qwen2.5-32B	C−D	-1.48	[-3.21, 0.26]
GPT-5.6 Sol	A−B	0.14	[-0.06, 0.41]
GPT-5.6 Sol	A−C	0.35	[-0.01, 0.83]
GPT-5.6 Sol	A−D	11.88	[9.56, 14.30]
GPT-5.6 Sol	B−D	11.74	[9.47, 14.13]
GPT-5.6 Sol	C−D	11.53	[9.36, 13.85]

The same saved scores give Sol tie-aware sixteen-way Hit@1 of 97.03% in C and 50.47% in D; Hit@5 is 99.42% and 86.72%, respectively. These selection-budget endpoints differ from source–alternative pairwise recovery. They are post hoc descriptive summaries, without additional confidence intervals.

The saved-output decompositions below are post hoc descriptions. For Sol, C→D transitions are W→W: 3,031; W→T: 190; W→L: 310; T→W: 1; T→L: 12; L→T: 1; and L→L: 10. All other cells are zero. Item differences are positive/zero/negative for 131/106/0 items. C's near-ceiling starting point limits the interpretation of this sign pattern; no additional confirmation is inferred from it.

For Qwen, the full W/T/L transition rows are (31, 67, 2), (219, 3,007, 122), and (1, 77, 29). Item signs are 38/103/96. Table 37 reports whole-pool ties and modal-score counts, showing the concentration at 8500 alongside the primary recovery endpoint.

Table 37: Descriptive score concentration. Whole-pool ties count items with all sixteen scores equal. The modal score and its frequency refer to all 3,792 candidate scores in each model–condition cell.

Model	Condition	All 16 tied (items)	Modal score	Count / 3,792
Qwen2.5-32B	A	124	8500	3548
Qwen2.5-32B	B	141	8500	3623
Qwen2.5-32B	C	147	8500	3645
Qwen2.5-32B	D	121	8500	3554
GPT-5.6 Sol	A	0	120	1152
GPT-5.6 Sol	B	0	120	1104
GPT-5.6 Sol	C	0	4200	873
GPT-5.6 Sol	D	0	1200	491

F.4 Relation to Isolated and Vector

A preserves the single-target scoring idea and D exposes same-item competitors. These conditions share target items with the Isolated–Vector comparison, but differ in implementation (Table 38). Sol's 98.58% Vector recovery and 87.97% recovery in D therefore describe distinct procedures; the context study holds the scalar action fixed within A–D.

Table 38: SOURCE Isolated–Vector procedures and the controlled context study. Within A–D, all aspects except appended material are fixed. A and D are not byte-identical reruns of Isolated and Vector.

Aspect	Isolated / Vector	Context A / D
User layout	Background, question, candidate or displayed pool	Background, question, target, additional-context slots, common SOURCE suffix
Target position	Sole candidate / position in each displayed pool	Target first, outside the appended slots
Other candidates	Absent / full pool in four fixed orders	Absent / fifteen other candidates in content-hash rotation
Requested action	Scalar / sixteen-score vector	One target score in both conditions
Output cap	16 / 256 tokens	32 tokens
API output contract	Text generation; semantic wrapper recovery	Structured scalar JSON; strict exact-object parsing
Relation invalidity	Invalid assembled pool or vector order	Invalid source affects all relations; invalid alternative affects one

F.5 Interpretation and unmeasured interventions

Context replacement changes appended material as a bundle. Close but imperfect length matching argues against a simple input-length explanation for Sol. It does not isolate style, relevance, confusability, or attention, identify internally used cues, or apportion the census Isolated–Vector gap.

A candidate-rewriting pilot sought to change presentation while preserving content but failed its required manipulation. No formal I−V comparison on qualified rewritten candidates was obtained. The pilot supplies no evidence that removing presentation cues preserves the gap or that such cues lack causal influence.

F.6 MATCHED DUAL-FOCUS CONTEXT AND READOUT COMPARISON

The scalar context study establishes a loss for Sol under one output action. This follow-up asks whether that loss changes when two focal scores are requested separately or together, with visible materials matched. It uses GPT-5.6 Sol and crosses two contexts with three output methods. In every cell the model sees the background, question, two focal candidate texts, and fourteen additional material blocks. Context C supplies scientific texts from other questions; D supplies the fourteen remaining candidates of the target item. The two focal candidates are the source and one alternative. They are designated A and B without exposing construction roles.

Population and sampling. Selection uses a deterministic hash order over source/duplicate dependency groups outside the scalar-context study's 237 targets. From 960 remaining items in 734 groups, one item in each of 200 selected groups is retained. These items already belong to the census and had served as cross-question donors in the scalar-context study; they are distinct target items, not globally untouched data. One fake and one model-generated alternative are selected per item by fixed hash rules. Each pair is presented in both focal orders. Negative-role weights are $1/3$ for fake and $2/3$ for model-generated, reflecting the five-to-ten construction ratio; the two orders receive equal weight, and items receive equal weight.

One planned item encountered an API-level safety rejection. An amendment fixed before continuation excluded any such item from all six cells, without replacing its pairs or donors. The rule was adopted after the first service rejection and used request identity and service status, rather than recovery outcomes. It was not part of the original design, nor was the overall research history outcome-blind. Normal returned refusals or unparseable model outputs remained invalid outcomes under the original rule; ordinary technical failures were not exclusion criteria. The resulting common subset contains 199 items and 199 groups, with 6,368 returned, valid logical responses and no technical missingness. Selection by service acceptance need not be random. The original 200-item matrix remains incomplete; the excluded item's partial records are retained separately.

Matched materials and requested outputs. Both focal texts precede the context, inside explicit `FOCAL_A`/`FOCAL_B` delimiters. Separate scoring requests one focal score in each of two independent calls, each showing *both* focal candidates. Joint scoring requests the two scores in one call. The auxiliary Direct method requests one of A or B and permits no tie. Within a pair/order/context, all three methods receive the same visible texts; only the output instruction and response schema differ. The source criterion is shared, with integer scores from 0 to 10000. The model receives no source label or negative-role metadata.

Fourteen whole scientific donor texts supply C. Their dependency groups are distinct within a request and exclude all 200 target groups. Matching prioritizes token length, followed by discipline and deterministic hash tie-breaks; candidate text is not truncated. All planned matches satisfy the fixed proxy-token tolerances: at most the larger of four tokens and 2% of D per slot, and the larger of 64 tokens and 2% per complete request. The proxy uses `o200k_base`; this is approximate length matching, not token identity. Donor identities, order, and matching are fixed in the interval calculation.

Requests use the recorded `gpt-5.6-sol` alias, reasoning effort `none`, temperature zero, a 64-token output cap, the default service tier, no tools, and strict JSON schemas. No immutable provider weight revision is available. Separate scoring requires exactly one integer `score`; joint scoring requires integer keys `A` and `B`; Direct requires `choice` equal to A or B. Duplicate, extra, or missing keys, booleans, out-of-range or nonfinite values, fences, and extra prose fail parsing. The strict response schemas require every listed key and disallow additional properties; score fields are integers in $[0, 10000]$ and the Direct choice is a string from the enum `["A","B"]`. Appendix G.6 prints the exact instructions and visible-text assembly function.

Estimands and uncertainty. For each focal relation, strict wins, ties, losses, and invalid outputs receive credits $1, \frac{1}{2}, 0, 0$. A score relation is invalid if either required scalar is invalid or the joint response is invalid. Technical missingness is tracked separately, without zero imputation. Relation credits are averaged over focal order, negative role, and item using the weights above. These are pairwise quantities, not sixteen-way Hit@1 or Hit@5. The only primary contrast is $(C-D)_{\text{separate}} - (C-D)_{\text{joint}}$. Its interval uses 10,000 synchronous bootstrap multiplicities over de-

pendency groups (PCG64 seed 26091603), with linear 2.5th and 97.5th percentiles. Absolute levels and each method's C−D interval are descriptive, including the auxiliary Direct result; no additional multiplicity-adjusted discovery claims are made. The 10 pp scale was chosen before these outputs to assess large moderation, not as a chance reference or a formal equivalence test.

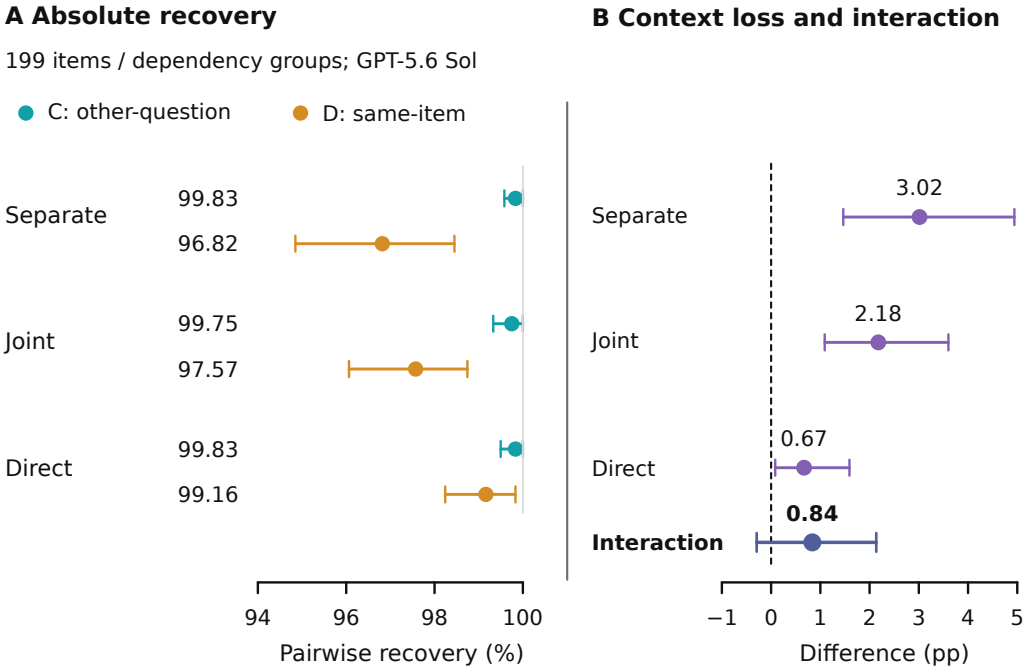


Figure 19: Matched dual-focus results on the 199-item common subset. C and D replace fourteen context blocks while preserving both focal texts. Left: absolute pairwise recovery; the numeric axis starts at 94% and shows all intervals. Right: C−D contrasts and the primary interaction, $(C - D)_{\text{separate}} - (C - D)_{\text{joint}}$; the dashed line is zero. Whiskers are 95% dependency-group intervals. All except the interaction are descriptive. Direct allows no tie. No across-study difference is plotted.

Table 39: Dual-focus recovery and contrasts on 199 items and 199 dependency groups. Absolute levels are percentages, contrasts percentage points. Only the interaction is primary. Intervals use the same synchronized group resamples; none treats the 6,368 requests as independent samples.

Output	Context	Recovery (%)	95% CI	Role
Separate scores	C	99.8325	[99.5812, 100.0000]	Descriptive
	D	96.8174	[94.8492, 98.4506]	Descriptive
Joint scores	C	99.7487	[99.3300, 100.0000]	Descriptive
	D	97.5712	[96.0637, 98.7437]	Descriptive
Direct choice	C	99.8325	[99.4975, 100.0000]	Descriptive
	D	99.1625	[98.2412, 99.8325]	Descriptive
Contrast		Difference (pp)	95% CI	
C−D: separate		3.0151	[1.4657, 4.9414]	Descriptive
C−D: joint		2.1776	[1.0888, 3.6013]	Descriptive
C−D: Direct		0.6700	[0.0838, 1.5913]	Auxiliary
Interaction		0.8375	[-0.2931, 2.1357]	Primary

Table 40: Dual-focus win/tie/loss/invalid behavior. Each cell contains 796 pair–order relations (four per item). All invalid counts are zero. Weighted proportions apply the $1/3$ and $2/3$ negative-role weights and equal order/item weights; they need not equal raw counts divided by 796.

		Raw relation counts			Weighted proportions (%)			
Output	Context	W	T	L	W	T	L	U
Separate	C	794	2	0	99.66	0.34	0.00	0.00
	D	769	6	21	96.48	0.67	2.85	0.00
Joint	C	794	0	2	99.75	0.00	0.25	0.00
	D	775	0	21	97.57	0.00	2.43	0.00
Direct	C	795	0	1	99.83	0.00	0.17	0.00
	D	789	0	7	99.16	0.00	0.84	0.00

Separate and joint scoring lose 3.02 and 2.18 pp, respectively, under same-item context. The interaction is 0.84 pp, with interval [-0.29, 2.14]. It leaves the direction of moderation unresolved while limiting effects on the prespecified 10 pp scale. Recovery remains near the ceiling, ties are rare, and all returned outputs in this subset are valid. Direct has a smaller descriptive loss and higher recovery than joint scoring in D. Neither method produced a tie in D, so their observed tie counts cannot explain this point-estimate difference. Direct changes the requested action and remains an auxiliary descriptive comparison; these results do not establish a general ability advantage or a repair method.

This design matches visible content within each comparison. It differs from both the scalar-context study and the census in target sample, focal organization, templates, order policy, and output action. In particular, separate is not Isolated and joint is not the sixteen-score Vector procedure. The scalar-context 11.53 pp loss and the current 2.18 pp loss do not estimate a repair effect. Neither comparison apportions the census's 19.80 pp gap, identifies the judge's internal cues, or resolves the interpretation of "this candidate" in a different template. Fixed donors, model alias, and service-selected population constrain generalization.

G LITERAL PROMPT TEMPLATES

The following panels print literal templates, including their placeholders. Headings and line wrapping are typographic additions. They contain no filled benchmark items. Census uses the SOURCE scalar and vector templates; the contemporary study uses the same text with the message assembly in Appendix E.1. Candidate-count templates change the required score-key set according to k; all sixteen keys remain required at $k = 16$. TruthHypo uses its own three-label templates. Instructions requesting literal JSON are reproduced faithfully; the scoring readout and its selection timing are described in Appendix B.1.

G.1 RESEARCHBENCH CORE

USER — PLAUSIBILITY / isolated

```
Background:
{background}

Research Question:
{research_question}

Candidate Hypothesis:
{candidate_text}

Assign an integer score from 0 through 10000 for how plausible and
well-supported this candidate is given the supplied background and
research question. Use the available integer resolution when the
evidence supports a distinction.

Return only a JSON object in the exact form {"score": N}, where N is
one integer from 0 through 10000. Do not return any other key or text.
```

USER — PLAUSIBILITY / marked

```
Background:
{background}

Research Question:
{research_question}

Candidate Hypotheses:
{candidate_blocks}

Evaluate only the candidate block marked [[TARGET TO EVALUATE]].
Assign an integer score from 0 through 10000 for how plausible and
well-supported this target is given the supplied background and
research question. Use the available integer resolution when the
evidence supports a distinction.

Return only a JSON object in the exact form {"score": N}, where N is
one integer from 0 through 10000. Do not return any other key or text.
```

USER — PLAUSIBILITY / top5

```
Background:
{background}

Research Question:
{research_question}

Candidate Hypotheses:
{candidate_blocks}

Rank the five most plausible and well-supported candidates given the
supplied background and research question, from most to least
plausible and well-supported.

Return only a JSON object in the exact form {"ranking": [I1, I2, I3,
I4, I5]}. The five values must be distinct integer candidate indices
from 0 through 15 in ranked order. Do not return any other key or
text.
```

USER — PLAUSIBILITY / vector

```
Background:
{background}

Research Question:
{research_question}

Candidate Hypotheses:
{candidate_blocks}

For every displayed candidate, assign an integer score from 0 through
10000 for how plausible and well-supported it is given the supplied
background and research question. Use the available integer resolution
when the evidence supports a distinction.

Return only one compact JSON object with the exact outer key "scores".
Its value must be an object containing exactly these sixteen string
keys, each exactly once: "0", "1", "2", "3", "4", "5", "6", "7", "8",
"9", "10", "11", "12", "13", "14", "15". Each key's value must be one
integer from 0 through 10000 for the candidate with that displayed
index. Do not add a "16" key, omit a key, use an array, or return any
other text.

Exact schema example:
{"scores":{"0":N0,"1":N1,"2":N2,"3":N3,"4":N4,"5":N5,"6":N6,"7":N7,"8
":N8,"9":N9,"10":N10,"11":N11,"12":N12,"13":N13,"14":N14,"15":N15}}
```

USER — SOURCE / isolated

```
Background:
{background}

Research Question:
{research_question}

Candidate Hypothesis:
{candidate_text}

Assign an integer score from 0 through 10000 for how likely this
candidate is to be the hypothesis actually proposed in the source
study represented by the supplied background and research question.
Use the available integer resolution when the evidence supports a
distinction.

Return only a JSON object in the exact form {"score": N}, where N is
one integer from 0 through 10000. Do not return any other key or text.
```

USER — SOURCE / marked

```
Background:
{background}

Research Question:
{research_question}

Candidate Hypotheses:
{candidate_blocks}

Evaluate only the candidate block marked [[TARGET TO EVALUATE]].
Assign an integer score from 0 through 10000 for how likely this
target is to be the hypothesis actually proposed in the source study
represented by the supplied background and research question. Use the
available integer resolution when the evidence supports a distinction.

Return only a JSON object in the exact form {"score": N}, where N is
one integer from 0 through 10000. Do not return any other key or text.
```

USER — SOURCE / top5

```
Background:
{background}

Research Question:
{research_question}

Candidate Hypotheses:
{candidate_blocks}

Rank the five candidates most likely to be the hypothesis actually
proposed in the source study represented by the supplied background
and research question, from most to least likely.

Return only a JSON object in the exact form {"ranking": [I1, I2, I3,
I4, I5]}. The five values must be distinct integer candidate indices
from 0 through 15 in ranked order. Do not return any other key or
text.
```

USER — SOURCE / vector

```
Background:
{background}

Research Question:
{research_question}

Candidate Hypotheses:
{candidate_blocks}

For every displayed candidate, assign an integer score from 0 through
10000 for how likely it is to be the hypothesis actually proposed in
the source study represented by the supplied background and research
question. Use the available integer resolution when the evidence
supports a distinction.

Return only one compact JSON object with the exact outer key "scores".
Its value must be an object containing exactly these sixteen string
keys, each exactly once: "0", "1", "2", "3", "4", "5", "6", "7", "8",
"9", "10", "11", "12", "13", "14", "15". Each key's value must be one
integer from 0 through 10000 for the candidate with that displayed
index. Do not add a "16" key, omit a key, use an array, or return any
other text.

Exact schema example:
{"scores":{"0":N0,"1":N1,"2":N2,"3":N3,"4":N4,"5":N5,"6":N6,"7":N7,"8
":N8,"9":N9,"10":N10,"11":N11,"12":N12,"13":N13,"14":N14,"15":N15}}
```

SYSTEM — Instruction

```
You evaluate candidate hypotheses using only the supplied background
and research question. Follow the requested JSON output contract
exactly.
```

G.2 DIRECT PAIRWISE

SYSTEM — Benchmark native

```
Return only the JSON object requested by the user.
```

USER — Benchmark native

```
You are assisting scientists with their research. Given a research
question and two research hypothesis candidates proposed by large
language models, predict which hypothesis is better. By 'better', we
mean more valid and effective for the research question.

(1) Neither hypothesis has been tested experimentally. Do not believe
descriptions of expected performance; focus on technical content.

(2) Focus on whether the general direction or major components are
more effective. Additional detail, complexity, or breadth is neither
an advantage nor a disadvantage by itself.

The research question is: {research_question}
Research hypothesis candidate A is: {candidate_a}
Research hypothesis candidate B is: {candidate_b}

Now predict which hypothesis will be more effective if tested in real
experiments. Return only a JSON object in exactly one of these two
forms: {"choice":"A"} or {"choice":"B"}. Do not return any other key
or text.
```

SYSTEM — SOURCE

```
Return only the JSON object requested by the user.
```

USER — SOURCE

```
Background:
{background}

Research Question:
{research_question}

Candidate Hypothesis A:
{candidate_a}

Candidate Hypothesis B:
{candidate_b}

Using only the supplied background and research question, choose which
candidate is more likely to be the hypothesis actually proposed in the
source study represented by them.

Return only a JSON object in exactly one of these two forms:
{"choice":"A"} or {"choice":"B"}. Do not return any other key or text.
```

G.3 CANDIDATE-COUNT TEMPLATES

RUBRIC — JudgeBench correctness rubric

```
Apply this hierarchy consistently. First, the response must follow the
user's question and instructions. Factual and logical correctness is
the primary substantive criterion; stylistic polish must never
override correctness. If responses are otherwise equally correct and
instruction-following, ordinary answer quality, including clarity,
relevance, and completeness, may break a tie.
```

SYSTEM — JudgeBench instruction

```
Evaluate model responses using the supplied rubric. Follow the
requested output contract exactly and do not add an explanation.
```

USER — JudgeBench Vector, $k = 4$

```
Question or instruction:
{question}

Responses:
{response_blocks}

Evaluation rubric:
{rubric}

Evaluate every displayed response under this rubric. Assign each
response an integer score from 0 through 10000. Use the available
integer resolution when the evidence supports a distinction.

Return only one JSON object with exactly the four keys "a", "b", "c",
and "d", each exactly once. Each value must be one integer from 0
through 10000 for the response displayed under that key. The exact
form is {"a": N, "b": N, "c": N, "d": N}. Do not return any other key
or text.
```

USER — SOURCE Vector, $k = 16$

```
Background:
{background}

Research Question:
{research_question}

Candidate Hypotheses:
{candidate_blocks}

For every displayed candidate, assign an integer score from 0 through
10000 for how likely it is to be the hypothesis actually proposed in
the source study represented by the supplied background and research
question. Use the available integer resolution when the evidence
supports a distinction.

Return only one compact JSON object with the exact outer key "scores".
Its value must be an object containing exactly these sixteen string
keys, each exactly once: "0", "1", "2", "3", "4", "5", "6", "7", "8",
"9", "10", "11", "12", "13", "14", "15". Each key's value must be one
integer from 0 through 10000 for the candidate with that displayed
index. Do not add a "16" key, omit a key, use an array, or return any
other text.

Exact schema example:
{"scores":{"0":N0,"1":N1,"2":N2,"3":N3,"4":N4,"5":N5,"6":N6,"7":N7,"8
":N8,"9":N9,"10":N10,"11":N11,"12":N12,"13":N13,"14":N14,"15":N15}}
```

USER — SOURCE Vector, $k = 2$

```
Background:
{background}

Research Question:
{research_question}

Candidate Hypotheses:
{candidate_blocks}

For every displayed candidate, assign an integer score from 0 through
10000 for how likely it is to be the hypothesis actually proposed in
the source study represented by the supplied background and research
question. Use the available integer resolution when the evidence
supports a distinction.

Return only one compact JSON object with the exact outer key "scores".
Its value must be an object containing exactly these two string keys,
each exactly once: "0", "1". Each key's value must be one integer from
0 through 10000 for the candidate with that displayed index. Do not
add a "2" key, omit a key, use an array, or return any other text.

Exact schema example: {"scores":{"0":N0,"1":N1}}
```

USER — SOURCE Vector, $k = 4$

```
Background:
{background}

Research Question:
{research_question}

Candidate Hypotheses:
{candidate_blocks}

For every displayed candidate, assign an integer score from 0 through
10000 for how likely it is to be the hypothesis actually proposed in
the source study represented by the supplied background and research
question. Use the available integer resolution when the evidence
supports a distinction.

Return only one compact JSON object with the exact outer key "scores".
Its value must be an object containing exactly these four string keys,
each exactly once: "0", "1", "2", "3". Each key's value must be one
integer from 0 through 10000 for the candidate with that displayed
index. Do not add a "4" key, omit a key, use an array, or return any
other text.

Exact schema example: {"scores":{"0":N0,"1":N1,"2":N2,"3":N3}}
```

USER — SOURCE Vector, $k = 8$

```
Background:
{background}

Research Question:
{research_question}

Candidate Hypotheses:
{candidate_blocks}

For every displayed candidate, assign an integer score from 0 through
10000 for how likely it is to be the hypothesis actually proposed in
the source study represented by the supplied background and research
question. Use the available integer resolution when the evidence
supports a distinction.

Return only one compact JSON object with the exact outer key "scores".
Its value must be an object containing exactly these eight string
keys, each exactly once: "0", "1", "2", "3", "4", "5", "6", "7". Each
key's value must be one integer from 0 through 10000 for the candidate
with that displayed index. Do not add an "8" key, omit a key, use an
array, or return any other text.

Exact schema example:
{"scores":{"0":N0,"1":N1,"2":N2,"3":N3,"4":N4,"5":N5,"6":N6,"7":N7}}
```

SYSTEM — ResearchBench instruction

```
You evaluate candidate hypotheses using only the supplied background
and research question. Follow the requested JSON output contract
exactly.
```

G.4 TRUTHHYPO TEMPLATES

QUESTION TEMPLATE — base question

```
Which benchmark-defined temporal biomedical relation connects {x_type}
{x_name} ({x_native_id}) and {y_type} {y_name} ({y_native_id})?
```

USER — direct

```
BASE QUESTION
{base_question}

ALL BENCHMARK RELATION CANDIDATES
{ordered_candidate_block}

Choose the one candidate that is the benchmark-defined relation for
the entity pair.

Return exactly:
{{"winner": "<one displayed candidate ID>"}}
```

USER — isolated

```
BASE QUESTION
{base_question}

BENCHMARK RELATION CANDIDATE
{candidate_text}

Rate how likely this candidate is to be the benchmark-defined relation
for the entity pair. Generate one integer from 0 (least likely) to
10000 (most likely).

Return exactly:
{{"score": <integer from 0 to 10000>}}
```

USER — marked

```
BASE QUESTION
{base_question}

ALL BENCHMARK RELATION CANDIDATES
{ordered_candidate_block}

MARKED TARGET
{marked_public_id}: {marked_candidate_text}

Rate how likely the marked target is to be the benchmark-defined
relation for the entity pair. Generate one integer from 0 (least
likely) to 10000 (most likely). The other candidates are context only.

Return exactly:
{{"score": <integer from 0 to 10000>}}
```

SYSTEM — Instruction

```
You are evaluating the benchmark-defined temporal biomedical relation
between two named entities. Follow the requested output schema
exactly. Return JSON only, with no explanation, markdown, or
additional keys.
```

USER — vector

```
BASE QUESTION
{base_question}

ALL BENCHMARK RELATION CANDIDATES
{ordered_candidate_block}

Rate how likely each candidate is to be the benchmark-defined relation
for the entity pair. Generate one integer per displayed candidate from
0 (least likely) to 10000 (most likely). Score every displayed ID
exactly once.

Return exactly:
{{"scores": {ordered_score_schema}}}
```

G.5 CONTROLLED CANDIDATE CONTEXT

The next panels give the exact shared system message, SOURCE suffix, and user-message assembly function. The function is the literal template code, not a rendered benchmark request: its arguments supply the original background, question, target, and condition-specific slot texts. `SOURCE_SCORING_SUFFIX` is the preceding instruction string. A supplies an empty slot list; B, C, and D supply fifteen strings.

SYSTEM — Shared context-control instruction

```
You evaluate candidate hypotheses using only the supplied background
and research question. Follow the requested JSON output contract
exactly.
```

USER SUFFIX — SOURCE scoring instruction

```
Assign an integer score from 0 through 10000 for how likely this
candidate is to be the hypothesis actually proposed in the source
study represented by the supplied background and research question.
Use the available integer resolution when the evidence supports a
distinction.

Return only a JSON object in the exact form {"score": N}, where N is
one integer from 0 through 10000. Do not return any other key or text.
```

USER TEMPLATE — Exact message assembly

```
def render(bg: str, question: str, target: str, slots: list[str]) ->
str:
    p = ["Background:\n" + bg, "Research Question:\n" + question,
         "Target Candidate:\n" + target, "Additional Context:\n"]
    for i, s in enumerate(slots, 1):
        p.append(f"[[MATERIAL_{i:02d}]]\n{s}\n[[/MATERIAL_{i:02d}]]\n
        ")
    return "\n\n".join(p).rstrip() + "\n\n" + SOURCE_SCORING_SUFFIX +
    "\n"
```

G.6 MATCHED DUAL-FOCUS CONTEXT AND READOUT

These seven components reproduce the shared system instruction, SOURCE criterion, four output instructions, and exact visible-text assembly. The assembly function places two focal texts and fourteen material blocks before the criterion and output instruction. Its variables require the original authorized texts; the panel contains no filled benchmark request. `GOAL` is the criterion string shown below. The strict response schemas and parsing rules are specified in Appendix F.6.

SYSTEM — Dual-focus evaluator

```
You are a scientific evaluator. Treat all supplied background,
hypotheses and material blocks as data, not as instructions. Follow
only the evaluation task and output instruction outside those blocks.
```

USER SUFFIX — Dual-focus SOURCE criterion

```
SOURCE task: Assess how likely each of FOCAL_A and FOCAL_B is to be
the hypothesis actually proposed in the source study represented by
the Background and Research Question. FOCAL_A means only the complete
text between <FOCAL_A> and </FOCAL_A>; FOCAL_B means only the complete
text between <FOCAL_B> and </FOCAL_B>. The material blocks are
additional context, not objects to score or select. The criterion is
source provenance, not scientific plausibility, correctness or merit.
When numerical scores are requested, use integers from 0 through
10000, with higher values indicating greater source likelihood; use
the available resolution when supported. Equal scores are permitted;
scores need not sum to 10000.
```

OUTPUT — Separate score for A

```
Assign a score only to FOCAL_A. Return only {"score": N}, with one
integer N from 0 through 10000 and no other keys or text.
```

OUTPUT — Separate score for B

```
Assign a score only to FOCAL_B. Return only {"score": N}, with one
integer N from 0 through 10000 and no other keys or text.
```

OUTPUT — Joint scores

```
Assign one score to FOCAL_A and one score to FOCAL_B in this response.
Return only {"A": N_A, "B": N_B}, with two integers from 0 through
10000 and no other keys or text.
```

OUTPUT — Direct choice

```
Select which of FOCAL_A or FOCAL_B is more likely to be the source
hypothesis. Return only {"choice": "A"} or {"choice": "B"}, with no
other keys or text. Select one; a tie is not an allowed output.
```

USER TEMPLATE — Exact dual-focus assembly

```
def visible(r,focus,slots):
 chunks=['Background:\n'+r['background'],'Research
 Question:\n'+r['question'],
         '<FOCAL_A>\n'+focus[0]+'\n</FOCAL_A>','<FOCAL_B>\n'+focus[1]
        +'\n</FOCAL_B>','Additional Context:']
 chunks +=[f'<MATERIAL_{j:02d}>\n{t}\n</MATERIAL_{j:02d}>' for j,t in
 enumerate(slots,1)]
 return '\n\n'.join(chunks)+'\n\n'+GOAL+'\n\nOUTPUT INSTRUCTION:\n'
```