SGR-BENCH: BENCHMARKING SEARCH AGENTS ON STATE-GATED RETRIEVAL

Anonymous authors
Paper under double-blind review

ABSTRACT

Recent advances in large language models and tool-using agents have expanded the range of benchmarked web tasks. Yet an important class of specialized retrieval tasks remains undercharacterized. On many specialized data-retrieval websites, answer-bearing evidence becomes accessible only after establishing the correct site-specific retrieval state through filters, views, hierarchies, or scopes. We term this capability *state-gated retrieval* (SGR). We introduce SGR-BENCH, a benchmark for this setting containing 134 task records across 14 public data ecosystems. Each task requires discovering the appropriate website and configuring its site-specific retrieval state to produce a structured answer. SGR-BENCH pairs constraint-guided and goal-oriented formulations of the same underlying problems, enabling controlled comparisons between explicit and implicit guidance for state-gated retrieval. We evaluate nine CLI-based agentic LLM systems and three commercial search-agent products on the 134-task benchmark. GPT-5.6-sol achieves the highest Overall Row-F1 at 62.49%. A manual audit of 156 analyzable failed CLI trajectories shows why: agents often reach a relevant web source, but establish the wrong site-specific retrieval state. Retrieval-scope drift (37.2%) and criterion mismatch (27.6%) dominate, whereas final answer composition accounts for only 10.3%. The dataset and single-case evaluation instructions are available at `https://huggingface.co/datasets/sgr-bench/SGR-Bench`.

1 INTRODUCTION

Recent advances in large language models and tool-using agents, together with corresponding advances in benchmark design, have substantially expanded the range of benchmarked web tasks, spanning knowledge-intensive question answering, interactive search, browser-grounded interaction, and multi-step task execution Nakano et al. (2022); Yao et al. (2023); Schick et al. (2023); Wei et al. (2025); Zhou et al. (2023); Drouin et al. (2024); Yoran et al. (2024). Search agents can now iteratively query the web, access and read documents, and synthesize evidence across multiple sources to address complex information needs. In many professional retrieval settings, however, identifying the appropriate website is only the first step. On specialized data-retrieval websites, answer-bearing evidence is often not exposed on entry pages or under default settings. It becomes accessible only after the correct filters, views, hierarchies, or scopes are set. We refer to this capability as state-gated retrieval (SGR): the ability to identify the appropriate website and configure its site-specific retrieval state so that answer-bearing evidence unavailable under the default state becomes accessible.

State-gated retrieval remains underexplored in existing benchmarks. Search-agent benchmarks such as BrowseComp, WebWalkerQA, and WideSearch mainly evaluate source discovery, search depth and breadth, and cross-page evidence aggregation over the open web Wei et al. (2025); Wu et al. (2025); Wong et al. (2025). DeepSearchQA extends this line toward comprehensive deep research evaluation Gupta et al. (2026). Web-agent benchmarks such as WebArena, Mind2Web, and WorkArena instead emphasize browser-grounded interaction, action sequencing, and end-to-end task completion Zhou et al. (2023); Deng et al. (2023); Drouin et al. (2024). Taken together, these benchmark lines have substantially advanced the evaluation of web-based information seeking. While highly valuable, they still leave open whether an agent can make answer-bearing evidence accessible by establishing the site-specific retrieval state required by a specialized website.

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.

This gap matters especially on specialized data-retrieval websites. Answer-bearing evidence is often governed by the website's site-specific retrieval state rather than exposed through static pages or direct lookup. Agents must therefore map domain constraints to state-setting controls such as filters, views, hierarchies, and result scopes, often under dependencies induced by earlier retrieval steps. In such settings, errors in candidate selection or scope control can easily propagate, causing agents to miss the evidence needed for correct completion.

To close this gap, we introduce SGR-BENCH, a benchmark for state-gated retrieval in specialized web retrieval. The current release contains 134 task records grounded in 14 public data ecosystems, and is designed to isolate state-gated retrieval in a controlled yet realistic setting. Each task presents a natural-language information need without revealing the target website, requiring the agent to discover the appropriate website, configure its site-specific retrieval state, and produce a structured answer grounded in on-site answer-bearing evidence. SGR-BENCH pairs constraint-guided and goal-oriented formulations of the same underlying problems, enabling controlled comparisons between explicit and implicit guidance for state-gated retrieval. We further employ rigorous quality control through a systematic data curation pipeline, including candidate website curation, task-design protocol specification, candidate construction and filtering, and multi-round expert validation for answer identifiability, shortcut resistance, and state-gated-retrieval necessity.

Evaluation on the 134-task benchmark SGR-BENCH reveals a pronounced gap between partial access to answer-bearing evidence and correct structured completion for current search-agent systems. We benchmark nine CLI-based agentic LLM systems, including GPT-5.6-sol, GPT-5.5 and Claude Opus 4.7 Singh et al. (2026); Anthropic (2026b), Gemini 3.1 Pro and Qwen3.6-Plus Google DeepMind (2026); Yang et al. (2025), GLM-5.1 and Seed-2.0 Pro GLM-5-Team et al. (2026); ByteDance Seed (2026), and Kimi K2.5 and DeepSeek V4 Pro Kimi Team et al. (2026); DeepSeek-AI (2026), together with three commercial search-agent products: Google Search AI Mode, Gemini Deep Research, and OpenAI Deep Research Google (2025; 2026); OpenAI (2025). On this evaluation set, GPT-5.6-sol achieves the highest Overall Row-F1 at 62.49%. Audited CLI failures further show that the dominant errors do not arise from failing to locate a relevant source. Instead, agents often reach the appropriate website but fail to preserve the site-specific retrieval state under which answer-bearing evidence remains valid. Retrieval-scope drift (37.2%) and criterion mismatch (27.6%) together account for 64.7% of audited failures, while final answer composition accounts for only 10.3%. Taken together, these results indicate that the main bottleneck is not source discovery, but preserving the site-specific retrieval state needed to turn partial access to answer-bearing evidence into correct structured completion.

Our main contributions are threefold:

- We introduce SGR-BENCH, a benchmark explicitly centered on state-gated retrieval in specialized web retrieval. The benchmark contributes 134 task records across 14 public data ecosystems, with paired constraint-guided and goal-oriented formulations that support diagnosis of explicit and implicit guidance for state-gated retrieval.
- We establish a systematic data curation methodology for building state-gated-retrieval benchmarks, combining candidate website curation, task-design protocol specification, candidate construction and filtering, and multi-round expert validation under six design requirements: domain specificity, long-tail source grounding, answer uniqueness and verifiability, ground-truth stability, shortcut resistance, and logical dependency.
- We provide a broad empirical evaluation of current search agents on this setting, covering nine CLI-based agentic LLM systems and three commercial products. The resulting evidence shows that the main failure mode is operating the right website in the wrong site-specific retrieval state, rather than final formatting alone.

2 RELATED WORK

We review search-agent and web-navigation benchmarks, with a comparison of representative benchmarks in Table 1.

Table 1: Comparison with representative prior benchmarks. *Expert Data-Retrieval Sites*: tasks are grounded in specialized websites. *State-Gated Evidence*: answer-bearing evidence becomes accessible only after the required site-specific retrieval state is established.

Benchmark	Primary Focus	Expert Data Retrieval Sites	State-Gated Evidence	Output Target
BrowseComp	Deep search	✗	✗	Short answer
WebWalkerQA	Deep search	✗	✗	Short answer
WideSearch	Wide search	✗	✗	Organized table / list
DeepSearchQA	Deep+wide search	✗	✗	Exhaustive answer set
WebArena	Task execution	✗	✗	Task completion
Mind2Web	Action grounding	✗	✗	Action sequence
WorkArena	Enterprise task execution	✗	✗	Task completion
SGR-BENCH	**SGR evidence seeking**	✓	✓	**Exhaustive structured outputs**

2.1 SEARCH-AGENT BENCHMARKS

Search-agent benchmarks increasingly evaluate agent performance on open-web information-seeking tasks. Earlier knowledge-intensive and open-domain question answering benchmarks, such as Natural Questions Kwiatkowski et al. (2019), TriviaQA Joshi et al. (2017), and KILT Petroni et al. (2021), evaluate retrieval-based answering with evidence drawn from Wikipedia or the open web. More recent search-augmented evaluations, such as FreshQA Vu et al. (2024) and GAIA Mialon et al. (2024), move closer to dynamic assistants that rely on web access and tool use. Deep-search benchmarks such as BrowseComp Wei et al. (2025) and WebWalkerQA Wu et al. (2025) focus on search tasks in which answer-bearing evidence is difficult to locate and often requires sustained multi-step browsing over complex web structures. Wide-search benchmarks such as WideSearch Wong et al. (2025) instead stress broad source coverage and answer-set completeness, requiring agents to exhaustively collect and deduplicate relevant items from large candidate sets. More recent benchmarks such as DeepSearchQA Gupta et al. (2026) and DeepWideSearch Lan et al. (2025) combine both dimensions, requiring agents to navigate difficult source structures while assembling exhaustive structured outputs. Concurrent deep-research benchmarks further examine report-level research, rubric-based diagnosis, and cross-domain accuracy in long-horizon web research Du et al. (2025); Li et al. (2026); Zhong et al. (2026).

Prior search-agent benchmarks primarily evaluate source discovery, traversal, and cross-page aggregation. SGR-BENCH extends this line of evaluation by explicitly evaluating a complementary capability: whether an agent can surface answer-bearing evidence that remains inaccessible until the appropriate website is brought into the correct site-specific retrieval state. This capability is important because, on many specialized data-retrieval websites, identifying the appropriate website is necessary but not sufficient, and answer-bearing evidence may remain inaccessible under the default site-specific retrieval state until appropriate state-gated retrieval is applied. Accordingly, SGR-BENCH adds a distinct evaluation axis to prior search-agent benchmarks by introducing tasks in which agents must identify the appropriate website among expert data-retrieval sources and then access answer-bearing evidence gated behind site-specific retrieval states.

2.2 WEB NAVIGATION AND INTERACTION BENCHMARKS

Web navigation benchmarks evaluate whether agents can complete tasks effectively in realistic browser environments. Early benchmarks focused on controlled or function-specific web environments. MiniWoB++ Shi et al. (2017) provides synthetic web-interaction tasks, and WebShop Yao et al. (2022) studies goal-directed interaction in an e-commerce setting. Subsequent benchmarks move closer to realistic public-web settings: WebArena Zhou et al. (2023) evaluates multi-site task execution in self-hosted web environments, VisualWebArena Koh et al. (2024) introduces visually grounded tasks on realistic websites, and Mind2Web Deng et al. (2023) uses real interaction traces collected from diverse websites. A further step in this progression emphasizes multi-turn and longer-horizon workflows. WebLINX Reddy et al. (2024) studies multi-turn website navigation from real-world demonstrations, WorkArena Drouin et al. (2024) and WorkArena++ Boisvert et al. (2024) focus on enterprise software and compositional workplace workflows, respectively,

and AssistantBench Yoran et al. (2024) evaluates realistic and time-consuming web tasks. Recent work has also expanded web-agent evaluation toward deterministic replicas, safety-oriented browser tasks, aligned browser-agent behavior, real-website end-to-end agents such as WebVoyager, and standardized browser-agent ecosystems such as BrowserGym Garg et al. (2025); Tur et al. (2025); Kumar et al. (2025); He et al. (2024); de Chezelles et al. (2024).

Web navigation benchmarks primarily assess browser-grounded task execution, including action grounding and completion of multi-step workflows. Recent benchmarks with realistic websites and replicas evaluate important state-changing workflows, but they do not center expert-curated retrieval tasks whose answer-bearing evidence is hidden behind source-specific data controls. SGR-BENCH therefore shifts the evaluation target from task execution to information seeking on specialized data-retrieval websites: the relevant question is whether an agent can identify the appropriate website and surface answer-bearing evidence that remains hidden until the site is brought into the correct site-specific retrieval state. Accordingly, our current system comparison targets production search-agent systems rather than purpose-built browser agents, because the benchmark is designed to measure retrieval-state establishment in realistic search workflows rather than general browser-control competence.

3 SGR-BENCH

3.1 TASK DEFINITION

For each task, let W denote the target website and let s denote a site-specific retrieval state of W, determined by website controls such as filters, views, hierarchy selections, and scopes. We write $V(W, s)$ for the entries, result views, or page content exposed under state s. For a task with answer a, the answer-bearing evidence $E(a)$ is not exposed under the default state s_0, but becomes accessible along a trajectory of states $s_1, \ldots, s_k$ induced by state-setting operations. An SGR task therefore requires an agent to identify W and find an operation sequence such that $E(a) \subseteq \bigcup_{t=1}^{k} V(W, s_t)$. The core difficulty arises when evidence exposed under one state determines which operation is needed next, forcing the agent to maintain and update the site-specific retrieval state across dependent retrieval steps. This focus complements existing benchmarks that emphasize source discovery and cross-page aggregation in search agents, or interface grounding and task execution in web-navigation agents.

Task. SGR-BENCH evaluates search agents on end-to-end SGR tasks. Given a question q that specifies an information need and an output schema, but does not reveal the target website, an agent $\mathcal{M}$ equipped with web search, webpage browsing, and document-access tools must identify W, configure the required site-specific retrieval state, and produce a structured answer $\hat{a}$ grounded in the answer-bearing evidence exposed along the retrieval trajectory. The model may use any search or browsing strategy, but each reference answer is grounded in the target website, and the intended solution path requires exposing the answer-bearing evidence through the website's site-specific retrieval state.

3.2 DATA CURATION PIPELINE

The authors oversee candidate curation and task design, while human contributors construct candidate tasks with LLM assistance for website exploration and task drafting. Independent human reviewers validate the resulting tasks and reference answers, and the authors make the final inclusion decisions. This use of LLM assistance follows prior work on language models as tool-using assistants Yao et al. (2023); Schick et al. (2023). As shown in Figure 1, the workflow proceeds through candidate website curation, task design protocol specification, task construction, and candidate filtering and validation. These stages progressively narrow from raw candidate sources to validated benchmark tasks.

3.2.1 CANDIDATE WEBSITE CURATION.

Professional data-retrieval websites are scattered across domains and lack a unified index. We therefore use the English Wikipedia external-links dump[1] as an initial pool for discovering candidate

[1] https://dumps.wikimedia.org/enwiki/latest/enwiki-latest-externallinks.sql.gz

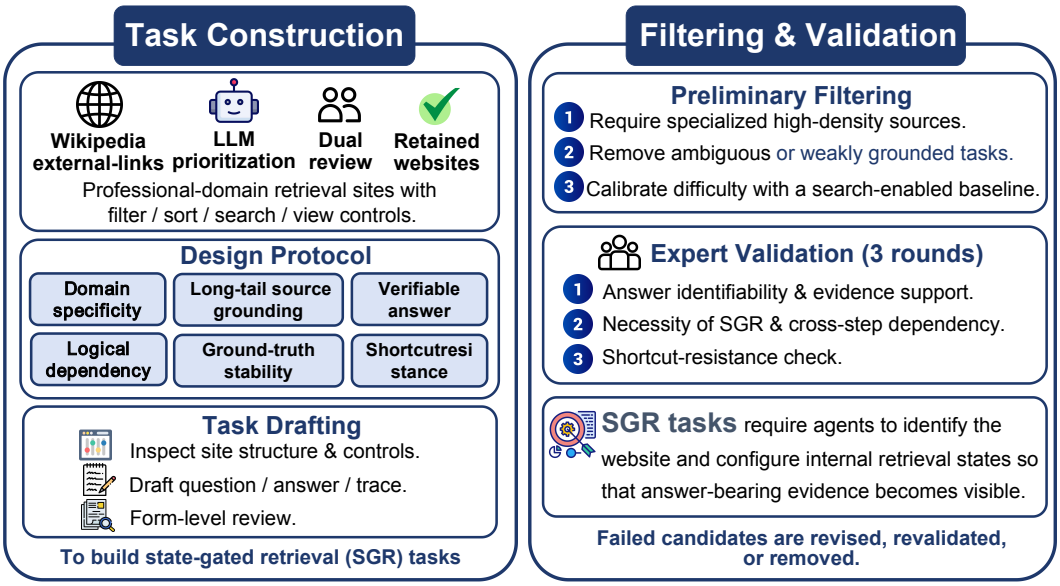

Figure 1: Overview of the SGR-BENCH four-stage data curation pipeline. Candidate websites are drawn from Wikipedia external links, prioritized with an LLM, and retained after dual review. Task candidates are drafted from site structure and retrieval controls under a six-requirement design protocol, then filtered through preliminary screening and three rounds of expert validation for answer identifiability, state-gated-retrieval necessity, and shortcut resistance.

public data-retrieval websites across domains. To scale the initial screening, Qwen-Plus prioritizes URLs likely to correspond to information-dense retrieval websites Yang et al. (2025). Two authors independently review each prioritized candidate website. A site is retained only if it supports professional-domain data retrieval rather than merely presenting static text and exposes retrieval-oriented controls such as filtering, sorting, search, or view switching; sites with discrepant judgments are excluded.

3.2.2 TASK DESIGN PROTOCOL

Before candidate drafting, we define a question-design protocol with six requirements to standardize task construction and support reproducible evaluation. Concretely, each task must satisfy six requirements: *Domain Specificity*, requiring field-specific concepts, terminology, or source knowledge beyond general web search; *Long-Tail Source Grounding*, requiring the answer to be located in specialized, high-density retrieval websites rather than common knowledge or direct web lookup; *Answer Uniqueness and Verifiability*, requiring a unique answer with unambiguous scoring criteria and source support sufficient for independent verification; *Ground-Truth Stability*, requiring the task to be anchored to time-specific or versioned reference sources so that the ground truth remains stable over evaluation; *Shortcut Resistance*, requiring the task to avoid directly searchable answers, explicit identifiers that collapse the search space, and bypass paths that eliminate the intended reasoning; and *Logical Dependency*, requiring at least one intermediate retrieval result to condition a subsequent search, filtering, or branching decision rather than merely combining independent lookup constraints.

3.2.3 TASK CONSTRUCTION.

Translating design requirements into concrete tasks requires both domain knowledge about each website's retrieval interface and careful control over question quality. For each curated website, we inspect the site structure and identify retrieval-oriented controls that expose task-relevant site-specific retrieval states. We use ChatGPT-5.2 Pro to assist in drafting candidate tasks involving these controls, following recent frontier-model use in agentic task drafting and web research settings Singh et al. (2026); Nakano et al. (2022). We then conduct a form-level review, discarding questions with unnatural phrasing and candidates whose required answer formats are ill-specified or difficult to evaluate

reproducibly. Substantive validity is assessed in the subsequent validation stage. Representative task examples are provided in Appendix B.

3.2.4 CANDIDATE FILTERING AND VALIDATION.

Preliminary Filtering. We first exclude candidates whose solutions do not materially depend on specialized, high-information-density domain sources, including questions that can be answered without domain-specific terminology or site-specific interpretation. Within this target scope, we further discard candidates that are ambiguous, underspecified, weakly grounded, otherwise unsuitable for reproducible evaluation, or admit obvious shortcut paths such as direct answer pages or explicit identifiers that collapse the search space. Finally, we calibrate difficulty using a search-enabled baseline model. We exclude tasks that the baseline solves reliably through near-direct search, as such cases provide limited diagnostic value, and tasks for which the baseline makes no meaningful progress, as these often reflect ill-posed task specifications or unstable answer grounding rather than substantive difficulty. We retain tasks on which the baseline exhibits partial but incomplete progress.

Expert Validation. Each candidate that passes preliminary filtering undergoes three rounds of independent expert validation by trained reviewers to assess answer identifiability, state-gated retrieval necessity, and resistance to near-direct shortcut solutions. In the first round, reviewers solve the task from scratch to verify that the reference answer is factually correct, uniquely identifiable, and supported by answer-bearing source evidence sufficient for independent verification. In the second round, reviewers verify from the reference solution path that the state-gated retrieval operations are genuinely required for exposing the answer-bearing evidence and induce at least one explicit, nontrivial dependency across retrieval steps. In the third round, reviewers test shortcut resistance by adversarially probing for residual bypass paths, including pre-existing answer pages, trivial identifiers that collapse the search space, and near-direct external lookup routes that bypass the intended state-gated retrieval operations. This step is designed to reduce the risk that a task can be solved without answer-bearing evidence from the target website or without establishing the required site-specific retrieval state. Candidates that fail any validation round are revised and revalidated or removed; final inclusion decisions are made by the authors. Additional details on contributors, review procedures, revision cycles, and compensation are provided in Appendix B.

3.3 DATASET STATISTICS

Scale and source coverage. SGR-BENCH contains 134 task records across 14 public data ecosystems, covering faceted search, hierarchical browsing, time-window selection, database-specific query fields, and scoped result views. Across six source families, environmental monitoring accounts for 36 records (26.9%); regulatory resources, scholarly archives, life-science resources, and official statistics each account for 22 (16.4%); and vulnerability databases account for 10 (7.5%). We group EPSG/PROJ with environmental monitoring as a geospatial reference resource and Cellosaurus with life-science resources.

Task taxonomy. At the task-taxonomy level, the 134 records form 67 aligned pairs, with 67 constraint-guided and 67 goal-oriented records. The two variants in each pair are derived from the same information need and are grounded in the same target website, reference answer, evidence requirements, and output format. Constraint-guided variants emphasize the retrieval logic needed to reach the answer, whereas goal-oriented variants emphasize the target information need and leave more of that logic implicit. This paired construction reduces confounding factors and supports cleaner comparisons between explicit and implicit guidance for state-gated retrieval. Counts exclude translations as additional tasks. The release retains two CDC WONDER task pairs with identical prompts and reference answers but different rubrics; these are counted as separate records, not independent information needs.

Answer schema. At the output-schema level, all 134 records use structured outputs with prescribed columns and ordering constraints. Reference answers range from 2 to 44 rows, with mean cardinality 7.31 and median 5.0; 80 records (59.7%) require at most seven rows. This unified schema keeps the output space structurally controlled and directly scorable while still spanning a meaningful range of answer-set sizes.

Benchmark preservation and maintenance. We support reproducibility through versioned offline environments[2] and a maintenance protocol for live tasks. The seven data ecosystems with offline environments, including ChemExpo/CPDat, account for 60 task records; these environments preserve fixed source data and task-relevant retrieval operations within their documented coverage. The scope of offline replication is constrained by source data volumes and restrictions on data access and redistribution. We therefore adopt a complementary maintenance protocol for the remaining 74 records, specifying monthly checks of source accessibility, required retrieval states, supporting evidence, and reference answers. Dated logs track changes and task status, while versioned benchmark releases retain historical task definitions and reference answers.

3.4 Evaluation

We deterministically canonicalize answers before scoring and compare field values by exact match after canonicalization and row alignment. Two designated evaluators from the author team jointly specify the rules for each task. For each task, the same canonicalization rules are applied to outputs from all evaluated systems. A rule is adopted only if both agree that it normalizes surface-form differences without changing the meaning of the answer; rules on which they disagree are excluded. The rules cover whitespace, punctuation, capitalization, date and unit formatting, and abbreviation variants. Task-specific aliases are accepted only when the answer-bearing source confirms that they denote the same entity or value. Canonicalization does not fill in missing fields, correct factual errors, or merge distinct source entities. Appendix A describes the rule categories, representative examples, and exclusions.

After canonicalization, predictions and reference answers are parsed into rows of field values and aligned using task-specific keys based on primary identifying fields. Item-F1 measures the correctness of individual fields, whereas Row-F1 requires all fields in a row to be correct. Pairwise order accuracy (P.O.A.) compares the relative order of shared rows (Kendall, 1938), capturing ordering errors that the F1 scores do not. Formal definitions are provided in Appendix A.

4 Experiments

4.1 Experimental Setup

The experiments reported here cover all 134 task records (67 constraint-guided and 67 goal-oriented) across 14 public data ecosystems. We evaluate two categories of search-agent systems: CLI-based agentic search systems and commercial agent systems. For fair comparison, all agents receive identical prompts specifying the task description, output format constraints, current date, and target query.

CLI-based LLM Agentic Search Systems. We evaluate nine frontier LLM-based systems spanning both proprietary and open-weight models, each equipped with search capabilities through a command-line interface (Table 2). The proprietary models include GPT-5.6-sol, GPT-5.5 and Claude Opus 4.7 Singh et al. (2026); Anthropic (2026b), as well as Gemini 3.1 Pro and Qwen3.6-Plus Google DeepMind (2026); Yang et al. (2025). The open-weight models include GLM-5.1 and Seed-2.0 Pro GLM-5-Team et al. (2026); ByteDance Seed (2026), as well as Kimi K2.5 and DeepSeek V4 Pro Kimi Team et al. (2026); DeepSeek-AI (2026). We access GPT-5.5 and GPT-5.6-sol through the official OpenAI API, while all other models are accessed via the OpenRouter platform OpenRouter (2026). For both GPT models, we use Codex CLI as the search interface OpenAI (2026); for all remaining models, we use Claude Code CLI Anthropic (2026a). Each task run has a 6,000-second time limit, with a maximum output length of 32,000 tokens per API request. Reasoning effort is set to medium where configurable, except for DeepSeek V4 Pro, which uses high because only high and max are available. Model-specific temperatures and API costs are reported in Appendix C. Results reflect the complete CLI-based systems rather than isolated model capabilities.

Commercial Agent Systems. We evaluate Google Search AI Mode, Gemini Deep Research, and OpenAI Deep Research Google (2025; 2026); OpenAI (2025) through manual interaction with their

[2]Anonymous environment repository: `https://anonymous.4open.science/r/sgr-webenv-E7D7/README.md`.

Table 2: Main results on the 134-task benchmark SGR-BENCH. We report Row-F1, Item-F1, and pairwise order accuracy (P.O.A.); Overall denotes the arithmetic mean of CG and GO Row-F1.

Model / System	Constraint-Guided Tasks			Goal-Oriented Tasks			Overall
	Ordered Table			Ordered Table			Row-F1
	Row-F1	Item-F1	P.O.A.	Row-F1	Item-F1	P.O.A.	
CLI-based LLM Agentic Search Systems							
GPT-5.6-sol	**59.69**	**73.72**	**91.75**	**65.29**	**68.73**	90.10	**62.49**
GPT-5.5	45.48	68.22	90.91	41.26	64.15	89.90	43.37
Claude Opus 4.7	41.52	64.35	80.81	35.50	58.41	81.82	38.51
Gemini 3.1 Pro	23.30	56.43	89.61	30.92	61.70	**90.91**	27.11
Qwen3.6-Plus	11.11	44.25	79.29	12.12	29.50	42.42	11.62
GLM-5.1	30.54	63.17	86.87	36.09	68.10	87.63	33.32
Seed-2.0 Pro	22.73	31.48	45.45	18.18	28.27	27.27	20.46
Kimi K2.5	30.19	47.54	81.82	37.28	45.17	72.73	33.74
DeepSeek V4 Pro	25.50	56.67	80.00	39.78	65.29	90.00	32.64
Commercial Agent Systems							
Gemini Deep Research	11.11	29.72	45.45	9.42	30.15	45.45	10.27
Google Search AI Mode	1.40	16.73	51.52	3.31	13.01	36.36	2.36
OpenAI Deep Research	39.67	57.27	71.72	43.33	51.14	62.63	41.50

web interfaces, providing only the task prompt and collecting the final answer. Their internal retrieval pipelines are controlled by the respective services.

Metrics. We report Item-F1, Row-F1, and pairwise order accuracy (P.O.A.) after answer canonicalization and row alignment (Section 3.4; Appendix A). Item-F1 credits individual fields, whereas Row-F1 requires complete records. Overall averages CG and GO Row-F1. Final-answer evaluation covers all systems; evidence-grounding audits cover the nine CLI-based systems.

4.2 MAIN RESULTS

Table 2 summarizes performance across systems and task formulations.

Finding 1: Answer-bearing evidence is found, but the site-specific retrieval state is not preserved. Across all 12 evaluated systems, mean Item-F1 is 49.72%, showing that agents often reach pages or result entries containing some of the required field values. Row-F1 is much lower, with a mean of 29.78%, yielding a 19.94-point gap. GPT-5.6-sol achieves the highest Overall Row-F1 at 62.49%. The same separation appears in both constraint-guided tasks (50.80% Item-F1 vs. 28.52% Row-F1) and goal-oriented tasks (48.64% vs. 31.04%). This pattern is consistent with retrieval-state loss: agents can copy locally correct values from a page, but the trajectory audit below shows that failures usually enter when they do not preserve the site-specific retrieval state that made those values valid, including the active filters, selected hierarchy node, result scope, and row identity. Figure 2(a) visualizes this gap.

To assess whether correct answers were supported by retrieved evidence, we used GLM-5.2 with a fixed prompt to audit all 1,206 trajectories from the nine CLI-based systems across the 134 tasks. Among these, all 192 trajectories with fully correct final answers were judged to have retrieved sufficient official evidence before producing the answer. A manual spot-check of 50 of these correct trajectories agreed with the automated judgments in every case.

Finding 2: The hard step is configuring the website, not finding the website. We manually audited 176 trace-bearing trajectories from a subset of eight CLI-based agents and analyzed 156 failed runs. The dominant failure modes are *Retrieval-Scope Drift* (37.2%) and *Criterion Mismatch* (27.6%), together accounting for 64.7% of audited failures. For example, on a Reptile Database task, Qwen3.6-Plus failed to establish the reference candidate set and omitted four of eight required records, including one supported by evidence it had already retrieved (Appendix F). This identifies

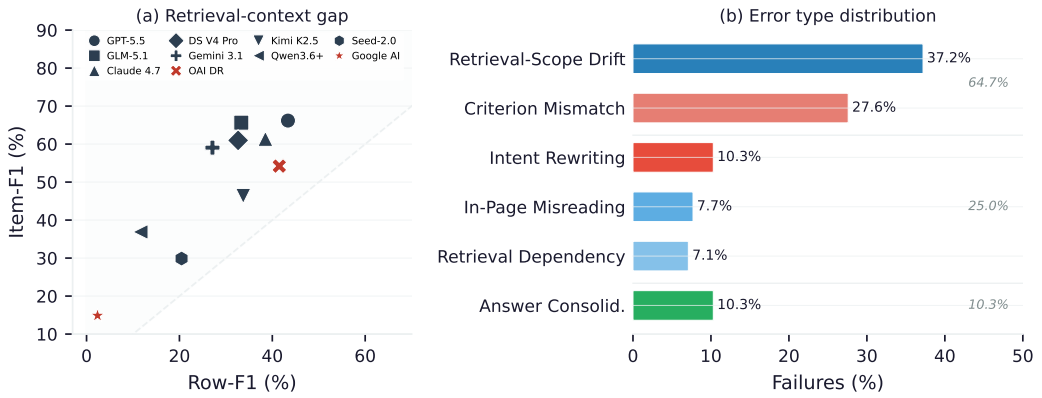

Figure 2: (a) Item-F1 vs. Row-F1 for the ten systems shown. All points sit above the diagonal: agents recover field values more often than complete rows. (b) Error type distribution across 156 analyzable failed CLI trajectories. Most failures enter while configuring or maintaining the website's site-specific retrieval state (64.7%), rather than during final answer composition (10.3%).

within-site state control, rather than final answer assembly, as the central bottleneck (Figure 2(b); Appendix G).

Finding 3: Hard sites require keeping several web controls aligned. Figure 3 presents a source-family breakdown of Item-F1 on the 134-task benchmark for the systems and families shown. Life-science resources generally yield the highest scores, followed broadly by scholarly archives and environmental monitoring. Regulatory resources and official statistics tend to be more challenging.

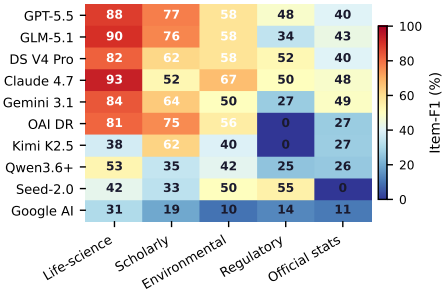

Figure 3: Item-F1 (%) for the ten systems and five source families shown.

The trajectory audit reveals a recurring failure pattern: agents retrieve plausible evidence without ensuring that it satisfies all task constraints. Even accurately extracted values can therefore come from records outside the required scope. Within the audited subset, scope drift is the most common error in scholarly and environmental tasks, while criterion mismatch is frequent in regulatory and official-statistics tasks (Appendix I). These findings underscore the need to maintain task constraints throughout retrieval.

CLI-based systems also outperform commercial systems on mean Item-F1 (55.29% vs. 33.00%), indicating that stronger interaction control still matters in this setting. Constraint-guided tasks also yield slightly higher Item-F1 ($\Delta = 2.16$), while goal-oriented tasks achieve marginally higher Row-F1 ($\Delta = 2.52$), suggesting that explicit guidance helps locate fields but does not solve scope maintenance.

5 Conclusion

We introduce SGR-Bench, a benchmark of 134 task records across 14 public data ecosystems for evaluating state-gated retrieval. Versioned offline environments and a maintenance protocol for live tasks support benchmark reproducibility. Within the audited failed trajectories, retrieval-scope drift and criterion mismatch are the dominant failure modes, showing how relevant evidence can yield incorrect answers when the retrieval state does not satisfy task constraints. These findings motivate agent training and evaluation that explicitly account for retrieval-state preservation across dependent steps.

AI USE STATEMENT

We used generative AI tools to assist with benchmark construction, trajectory analysis, and manuscript revision. Qwen-Plus prioritized candidate websites, while ChatGPT-5.2 Pro supported exploration of website content and task context during question drafting (Section 3.2). Human reviewers independently validated the tasks and reference answers, and the authors made the final inclusion decisions. GLM-5.2 was used for the automated evidence-grounding audit described in Section 4.2, with a subset of its judgments checked manually. We also used generative AI to refine the manuscript's language and presentation. The authors take responsibility for the final research decisions and all submitted content.

REPRODUCIBILITY STATEMENT

The benchmark tasks and reference answers are available via the dataset link in the abstract. Section 3 and Appendices A–B detail benchmark construction, quality control, answer canonicalization, and evaluation metrics. Appendix C describes the experimental settings and the workflow for executing and scoring individual tasks. Section 3.3 outlines the benchmark preservation and maintenance strategy.

ETHICS STATEMENT

We recruited compensated doctoral-student contributors for benchmark task construction and independent validation. Participants were informed of the nature of the work and the compensation arrangements before participation. We did not collect personal or sensitive information from contributors, nor did we conduct experiments on or analyze individual-level characteristics of the contributors. Compensation and workload are reported in Appendix B.

REFERENCES

Anthropic. Claude Code Overview. https://code.claude.com/docs/en/overview, 2026a. Accessed: 2026-05-06.

Anthropic. System Card: Claude Opus 4.7. https://www.anthropic.com/claude-opus-4-7-system-card, apr 2026b. Accessed: 2026-05-06.

Léo Boisvert, Megh Thakkar, Maxime Gasse, Massimo Caccia, Thibault Le Sellier de Chezelles, Quentin Cappart, Nicolas Chapados, Alexandre Lacoste, and Alexandre Drouin. Workarena++: Towards compositional planning and reasoning-based common knowledge work tasks. *Advances in Neural Information Processing Systems*, 37:5996–6051, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/0b82662b6c32e887bb252a74d8cb2d5e-Abstract-Datasets_and_Benchmarks_Track.html.

ByteDance Seed. Seed2.0 Model Card: Towards Intelligence Frontier for Real-World Complexity. https://lf3-static.bytednsdoc.com/obj/eden-cn/lapzild-tss/ljhwZthlaukjlkulzlp/seed2/0214/Seed2.0%20Model%20Card.pdf, feb 2026. Accessed: 2026-05-06.

Thibault Le Sellier de Chezelles, Maxime Gasse, Alexandre Drouin, Massimo Caccia, Léo Boisvert, Megh Thakkar, Tom Marty, Rim Assouel, Sahar Omidi Shayegan, Lawrence Keunho Jang, Xing Han Lù, Ori Yoran, Dehan Kong, Frank F. Xu, Siva Reddy, Quentin Cappart, Graham Neubig, Ruslan Salakhutdinov, Nicolas Chapados, and Alexandre Lacoste. The browsergym ecosystem for web agent research, 2024. URL https://arxiv.org/abs/2412.05467.

DeepSeek-AI. DeepSeek V4 Technical Documentation. https://fe-static.deepseek.com/chat/transparency/deepseek-V4-model-card-EN.pdf, apr 2026. Accessed: 2026-05-06.

Xiang Deng, Yu Gu, Boyuan Zheng, Shijie Chen, Sam Stevens, Boshi Wang, Huan Sun, and Yu Su. Mind2web: Towards a generalist agent for the web. *Advances in Neural Information Processing Systems*, 36:28091–28114, 2023.

Alexandre Drouin, Maxime Gasse, Massimo Caccia, Issam H. Laradji, Manuel Del Verme, Tom Marty, Léo Boisvert, Megh Thakkar, Quentin Cappart, David Vazquez, Nicolas Chapados, and Alexandre Lacoste. Workarena: How capable are web agents at solving common knowledge work tasks? 2024.

Mingxuan Du, Benfeng Xu, Chiwei Zhu, Xiaorui Wang, and Zhendong Mao. Deepresearch bench: A comprehensive benchmark for deep research agents, 2025. URL https://arxiv.org/abs/2506.11763.

Divyansh Garg, Shaun VanWeelden, Diego Caples, Andis Draguns, Nikil Ravi, Pranav Putta, Naman Garg, Tomas Abraham, Michael Lara, Federico Lopez, James Liu, Atharva Gundawar, Prannay Hebbar, Youngchul Joo, Jindong Gu, Charles London, Christian Schroeder de Witt, and Sumeet Motwani. REAL: Benchmarking autonomous agents on deterministic simulations of real websites. 2025. URL https://openreview.net/forum?id=Un1sWxmZuI.

GLM-5-Team, Aohan Zeng, Xin Lv, Zhenyu Hou, Zhengxiao Du, Qinkai Zheng, Bin Chen, Da Yin, Chendi Ge, Chenghua Huang, Chengxing Xie, Chenzheng Zhu, Congfeng Yin, Cunxiang Wang, Gengzheng Pan, Hao Zeng, Haoke Zhang, Haoran Wang, Huilong Chen, Jiajie Zhang, Jian Jiao, Jiaqi Guo, Jingsen Wang, Jingzhao Du, Jinzhu Wu, Kedong Wang, Lei Li, Lin Fan, Lucen Zhong, Mingdao Liu, Mingming Zhao, Pengfan Du, Qian Dong, Rui Lu, Shuang-Li, Shulin Cao, Song Liu, Ting Jiang, Xiaodong Chen, Xiaohan Zhang, Xuancheng Huang, Xuezhen Dong, Yabo Xu, Yao Wei, Yifan An, Yilin Niu, Yitong Zhu, Yuanhao Wen, Yukuo Cen, Yushi Bai, Zhongpei Qiao, Zihan Wang, Zikang Wang, Zilin Zhu, Ziqiang Liu, Zixuan Li, Bojie Wang, Bosi Wen, Can Huang, Changpeng Cai, Chao Yu, Chen Li, Chengwei Hu, Chenhui Zhang, Dan Zhang, Daoyan Lin, Dayong Yang, Di Wang, Ding Ai, Erle Zhu, Fangzhou Yi, Feiyu Chen, Guohong Wen, Hailong Sun, Haisha Zhao, Haiyi Hu, Hanchen Zhang, Hanrui Liu, Hanyu Zhang, Hao Peng, Hao Tai, Haobo Zhang, He Liu, Hongwei Wang, Hongxi Yan, Hongyu Ge, Huan Liu, Huanpeng Chu, Jia'ni Zhao, Jiachen Wang, Jiajing Zhao, Jiamin Ren, Jiapeng Wang, Jiaxin Zhang, Jiayi Gui, Jiayue Zhao, Jijie Li, Jing An, Jing Li, Jingwei Yuan, Jinhua Du, Jinxin Liu, Junkai Zhi, Junwen Duan, Kaiyue Zhou, Kangjian Wei, Ke Wang, Keyun Luo, Laiqiang Zhang, Leigang Sha, Liang Xu, Lindong Wu, Lintao Ding, Lu Chen, Minghao Li, Nianyi Lin, Pan Ta, Qiang Zou, Rongjun Song, Ruiqi Yang, Shangqing Tu, Shangtong Yang, Shaoxiang Wu, Shengyan Zhang, Shijie Li, Shuang Li, Shuyi Fan, Wei Qin, Wei Tian, Weining Zhang, Wenbo Yu, Wenjie Liang, Xiang Kuang, Xiangmeng Cheng, Xiangyang Li, Xiaoquan Yan, Xiaowei Hu, Xiaoying Ling, Xing Fan, Xingye Xia, Xinyuan Zhang, Xinze Zhang, Xirui Pan, Xu Zou, Xunkai Zhang, Yadi Liu, Yandong Wu, Yanfu Li, Yidong Wang, Yifan Zhu, Yijun Tan, Yilin Zhou, Yiming Pan, Ying Zhang, Yinpei Su, Yipeng Geng, Yong Yan, Yonglin Tan, Yuean Bi, Yuhan Shen, Yuhao Yang, Yujiang Li, Yunan Liu, Yunqing Wang, Yuntao Li, Yurong Wu, Yutao Zhang, Yuxi Duan, Yuxuan Zhang, Zezhen Liu, Zhengtao Jiang, Zhenhe Yan, Zheyu Zhang, Zhixiang Wei, Zhuo Chen, Zhuoer Feng, Zijun Yao, Ziwei Chai, Ziyuan Wang, Zuzhou Zhang, Bin Xu, Minlie Huang, Hongning Wang, Juanzi Li, Yuxiao Dong, and Jie Tang. Glm-5: from vibe coding to agentic engineering, 2026. URL https://arxiv.org/abs/2602.15763.

Google. Expanding AI Overviews and Introducing AI Mode. https://blog.google/products-and-platforms/products/search/ai-mode-search/, mar 2025. Accessed: 2026-05-06.

Google. Gemini Deep Research. https://gemini.google/overview/deep-research/?hl=en-US, 2026. Accessed: 2026-05-06.

Google DeepMind. Model Evaluation – Approach, Methodology & Results: Gemini 3.1 Pro. https://storage.googleapis.com/deepmind-media/gemini/gemini_3-1_pro_model_evaluation.pdf, feb 2026. Accessed: 2026-05-06.

Nikita Gupta, Riju Chatterjee, Lukas Haas, Connie Tao, Andrew Wang, Chang Liu, Hidekazu Oiwa, E. Gribovskaya, Jan Ackermann, John Blitzer, S. Goldshtein, and Dipanjan Das. Deepsearchqa: Bridging the comprehensiveness gap for deep research agents. *arXiv preprint arXiv:2601.20975*, 2026.

Hongliang He, Wenlin Yao, Kaixin Ma, Wenhao Yu, Yong Dai, Hongming Zhang, Zhenzhong Lan, and Dong Yu. Webvoyager: Building an end-to-end web agent with large multimodal models. In

Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 6864–6890, Bangkok, Thailand, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.371. URL https://aclanthology.org/2024.acl-long.371/.

Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1601–1611, 2017.

Maurice G Kendall. A new measure of rank correlation. *Biometrika*, 30(1-2):81–93, 1938.

Kimi Team, Tongtong Bai, Yifan Bai, Yiping Bao, S. H. Cai, Yuan Cao, Y. Charles, H. S. Che, Cheng Chen, Guanduo Chen, Huarong Chen, Jia Chen, Jiahao Chen, Jianlong Chen, Jun Chen, Kefan Chen, Liang Chen, Ruijue Chen, Xinhao Chen, Yanru Chen, Yanxu Chen, Yicun Chen, Yimin Chen, Yingjiang Chen, Yuankun Chen, Yujie Chen, Yutian Chen, Zhirong Chen, Ziwei Chen, Dazhi Cheng, Minghan Chu, Jialei Cui, Jiaqi Deng, Muxi Diao, Hao Ding, Mengfan Dong, Mengnan Dong, Yuxin Dong, Yuhao Dong, Angang Du, Chenzhuang Du, Dikang Du, Lingxiao Du, Yulun Du, Yu Fan, Shengjun Fang, Qiulin Feng, Yichen Feng, Garimugai Fu, Kelin Fu, Hongcheng Gao, Tong Gao, Yuyao Ge, Shangyi Geng, Chengyang Gong, Xiaochen Gong, Zhuoma Gongque, Qizheng Gu, Xinran Gu, Yicheng Gu, Longyu Guan, Yuanying Guo, Xiaoru Hao, Weiran He, Wenyang He, Yunjia He, Chao Hong, Hao Hu, Jiaxi Hu, Yangyang Hu, Zhenxing Hu, Ke Huang, Ruiyuan Huang, Weixiao Huang, Zhiqi Huang, Tao Jiang, Zhejun Jiang, Xinyi Jin, Yu Jing, Guokun Lai, Aidi Li, C. Li, Cheng Li, Fang Li, Guanghe Li, Guanyu Li, Haitao Li, Haoyang Li, Jia Li, Jingwei Li, Junxiong Li, Lincan Li, Mo Li, Weihong Li, Wentao Li, Xinhang Li, Xinhao Li, Yang Li, Yanhao Li, Yiwei Li, Yuxiao Li, Zhaowei Li, Zheming Li, Weilong Liao, Jiawei Lin, Xiaohan Lin, Zhishan Lin, Zichao Lin, Cheng Liu, Chenyu Liu, Hongzhang Liu, Liang Liu, Shaowei Liu, Shudong Liu, Shuran Liu, Tianwei Liu, Tianyu Liu, Weizhou Liu, Xiangyan Liu, Yangyang Liu, Yanming Liu, Yibo Liu, Yuanxin Liu, Yue Liu, Zhengying Liu, Zhongnuo Liu, Enzhe Lu, Haoyu Lu, Zhiyuan Lu, Junyu Luo, Tongxu Luo, Yashuo Luo, Long Ma, Yingwei Ma, Shaoguang Mao, Yuan Mei, Xin Men, Fanqing Meng, Zhiyong Meng, Yibo Miao, Minqing Ni, Kun Ouyang, Siyuan Pan, Bo Pang, Yuchao Qian, Ruoyu Qin, Zeyu Qin, Jiezhong Qiu, Bowen Qu, Zeyu Shang, Youbo Shao, Tianxiao Shen, Zhennan Shen, Juanfeng Shi, Lidong Shi, Shengyuan Shi, Feifan Song, Pengwei Song, Tianhui Song, Xiaoxi Song, Hongjin Su, Jianlin Su, Zhaochen Su, Lin Sui, Jinsong Sun, Junyao Sun, Tongyu Sun, Flood Sung, Yunpeng Tai, Chuning Tang, Heyi Tang, Xiaojuan Tang, Zhengyang Tang, Jiawen Tao, Shiyuan Teng, Chaoran Tian, Pengfei Tian, Ao Wang, Bowen Wang, Chensi Wang, Chuang Wang, Congcong Wang, Dingkun Wang, Dinglu Wang, Dongliang Wang, Feng Wang, Hailong Wang, Haiming Wang, Hengzhi Wang, Huaqing Wang, Hui Wang, Jiahao Wang, Jinhong Wang, Jiuzheng Wang, Kaixin Wang, Linian Wang, Qibin Wang, Shengjie Wang, Shuyi Wang, Si Wang, Wei Wang, Xiaochen Wang, Xinyuan Wang, Yao Wang, Yejie Wang, Yipu Wang, Yiqin Wang, Yucheng Wang, Yuzhi Wang, Zhaoji Wang, Zhaowei Wang, Zhengtao Wang, Zhexu Wang, Zihan Wang, Zizhe Wang, Chu Wei, Ming Wei, Chuan Wen, Zichen Wen, Chengjie Wu, Haoning Wu, Junyan Wu, Rucong Wu, Wenhao Wu, Yuefeng Wu, Yuhao Wu, Yuxin Wu, Zijian Wu, Chenjun Xiao, Jin Xie, Xiaotong Xie, Yuchong Xie, Yifei Xin, Bowei Xing, Boyu Xu, Jianfan Xu, Jing Xu, Jinjing Xu, L. H. Xu, Lin Xu, Suting Xu, Weixin Xu, Xinbo Xu, Xinran Xu, Yangchuan Xu, Yichang Xu, Yuemeng Xu, Zelai Xu, Ziyao Xu, Junjie Yan, Yuzi Yan, Guangyao Yang, Hao Yang, Junwei Yang, Kai Yang, Ningyuan Yang, Ruihan Yang, Xiaofei Yang, Xinlong Yang, Ying Yang, Yi Yang, Yi Yang, Zhen Yang, Zhilin Yang, Zonghan Yang, Haotian Yao, Dan Ye, Wenjie Ye, Zhuorui Ye, Bohong Yin, Chengzhen Yu, Longhui Yu, Tao Yu, Tianxiang Yu, Enming Yuan, Mengjie Yuan, Xiaokun Yuan, Yang Yue, Weihao Zeng, Dunyuan Zha, Haobing Zhan, Dehao Zhang, Hao Zhang, Jin Zhang, Puqi Zhang, Qiao Zhang, Rui Zhang, Xiaobin Zhang, Y. Zhang, Yadong Zhang, Yangkun Zhang, Yichi Zhang, Yizhi Zhang, Yongting Zhang, Yu Zhang, Yushun Zhang, Yutao Zhang, Yutong Zhang, Zheng Zhang, Chenguang Zhao, Feifan Zhao, Jinxiang Zhao, Shuai Zhao, Xiangyu Zhao, Yikai Zhao, Zijia Zhao, Huabin Zheng, Ruihan Zheng, Shaojie Zheng, Tengyang Zheng, Junfeng Zhong, Longguang Zhong, Weiming Zhong, M. Zhou, Runjie Zhou, Xinyu Zhou, Zaida Zhou, Jinguo Zhu, Liya Zhu, Xinhao Zhu, Yuxuan Zhu, Zhen Zhu, Jingze Zhuang, Weiyu Zhuang, Ying Zou, and Xinxing Zu. Kimi k2.5: Visual agentic intelligence, 2026. URL https://arxiv.org/abs/2602.02276.

Jing Yu Koh, Robert Lo, Lawrence Jang, Vikram Duvvur, Ming Lim, Po-Yu Huang, Graham Neubig, Shuyan Zhou, Russ Salakhutdinov, and Daniel Fried. Visualwebarena: Evaluating

multimodal agents on realistic visual web tasks. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 881–905, 2024. doi: 10.18653/v1/2024.acl-long.50. URL https://aclanthology.org/2024.acl-long.50/.

P. Kumar, E. Lau, S. Vijayakumar, T. Trinh, E. T. Chang, V. Robinson, S. Zhou, and M. Fredrikson. Aligned LLMs are not aligned browser agents. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=NsFZZU9gvk.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466, 2019. doi: 10.1162/tacl_a_00276. URL https://aclanthology.org/Q19-1026/.

Tian Lan, Bin Zhu, Qianghuai Jia, Junyang Ren, Haijun Li, Longyue Wang, Zhao Xu, Weihua Luo, and Kaifu Zhang. Deepwidesearch: Benchmarking depth and width in agentic information seeking. *arXiv preprint arXiv:2510.20168*, 2025.

Ruizhe Li, Mingxuan Du, Benfeng Xu, Chiwei Zhu, Xiaorui Wang, and Zhendong Mao. Deepresearch bench ii: Diagnosing deep research agents via rubrics from expert report, 2026. URL https://arxiv.org/abs/2601.08536.

Grégoire Mialon, Clémentine Fourrier, Craig Swift, Thomas Wolf, Yann LeCun, and Thomas Scialom. Gaia: a benchmark for general ai assistants. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=fibxvahvs3.

Reiichiro Nakano, Jacob Hilton, Suchir Balaji, Jeff Wu, Long Ouyang, Christina Kim, Christopher Hesse, Shantanu Jain, Vineet Kosaraju, William Saunders, Xu Jiang, Karl Cobbe, Tyna Eloundou, Gretchen Krueger, Kevin Button, Matthew Knight, Benjamin Chess, and John Schulman. Webgpt: Browser-assisted question-answering with human feedback, 2022. URL https://arxiv.org/abs/2112.09332.

OpenAI. Deep Research System Card. https://cdn.openai.com/deep-research-system-card.pdf, feb 2025. Accessed: 2026-05-06.

OpenAI. CLI – Codex. https://developers.openai.com/codex/cli, 2026. Accessed: 2026-05-06.

OpenRouter. OpenRouter Models. https://openrouter.ai/docs/guides/overview/models, 2026. Accessed: 2026-05-06.

Fabio Petroni, Aleksandra Piktus, Angela Fan, Patrick Lewis, Majid Yazdani, Nicola De Cao, James Thorne, Yacine Jernite, Vassilis Plachouras, Tim Rocktäschel, and Sebastian Riedel. Kilt: a benchmark for knowledge intensive language tasks. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 2523–2544, 2021.

Siva Reddy, Xing Lu, and Zdeněk Kasner. Weblinx: Real-world website navigation with multi-turn dialogue. In *Institute of Formal and Applied Linguistics (ÚFAL)*, 2024.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language models can teach themselves to use tools, 2023. URL https://arxiv.org/abs/2302.04761.

Tianlin Shi, Andrej Karpathy, Linxi Fan, Jonathan Hernandez, and Percy Liang. World of bits: An open-domain platform for web-based agents. In *International Conference on Machine Learning*, pp. 3135–3144. PMLR, 2017.

Aaditya Singh, Adam Fry, Adam Perelman, Adam Tart, Adi Ganesh, Ahmed El-Kishky, Aidan McLaughlin, Aiden Low, AJ Ostrow, Akhila Ananthram, Akshay Nathan, Alan Luo, Alec Helyar, Aleksander Madry, Aleksandr Efremov, Aleksandra Spyra, Alex Baker-Whitcomb, Alex Beutel, Alex Karpenko, Alex Makelov, Alex Neitz, Alex Wei, Alexandra Barr, Alexandre Kirchmeyer,

Alexey Ivanov, Alexi Christakis, Alistair Gillespie, Allison Tam, Ally Bennett, Alvin Wan, Alyssa Huang, Amy McDonald Sandjideh, Amy Yang, Ananya Kumar, Andre Saraiva, Andrea Vallone, Andrei Gheorghe, Andres Garcia Garcia, Andrew Braunstein, Andrew Liu, Andrew Schmidt, Andrey Mereskin, Andrey Mishchenko, Andy Applebaum, Andy Rogerson, Ann Rajan, Annie Wei, Anoop Kotha, Anubha Srivastava, Anushree Agrawal, Arun Vijayvergiya, Ashley Tyra, Ashvin Nair, Avi Nayak, Ben Eggers, Bessie Ji, Beth Hoover, Bill Chen, Blair Chen, Boaz Barak, Borys Minaiev, Botao Hao, Bowen Baker, Brad Lightcap, Brandon McKinzie, Brandon Wang, Brendan Quinn, Brian Fioca, Brian Hsu, Brian Yang, Brian Yu, Brian Zhang, Brittany Brenner, Callie Riggins Zetino, Cameron Raymond, Camillo Lugaresi, Carolina Paz, Cary Hudson, Cedric Whitney, Chak Li, Charles Chen, Charlotte Cole, Chelsea Voss, Chen Ding, Chen Shen, Chengdu Huang, Chris Colby, Chris Hallacy, Chris Koch, Chris Lu, Christina Kaplan, Christina Kim, CJ Minott-Henriques, Cliff Frey, Cody Yu, Coley Czarnecki, Colin Reid, Colin Wei, Cory Decareaux, Cristina Scheau, Cyril Zhang, Cyrus Forbes, Da Tang, Dakota Goldberg, Dan Roberts, Dana Palmie, Daniel Kappler, Daniel Levine, Daniel Wright, Dave Leo, David Lin, David Robinson, Declan Grabb, Derek Chen, Derek Lim, Derek Salama, Dibya Bhattacharjee, Dimitris Tsipras, Dinghua Li, Dingli Yu, DJ Strouse, Drew Williams, Dylan Hunn, Ed Bayes, Edwin Arbus, Ekin Akyurek, Elaine Ya Le, Elana Widmann, Eli Yani, Elizabeth Proehl, Enis Sert, Enoch Cheung, Eri Schwartz, Eric Han, Eric Jiang, Eric Mitchell, Eric Sigler, Eric Wallace, Erik Ritter, Erin Kavanaugh, Evan Mays, Evgenii Nikishin, Fangyuan Li, Felipe Petroski Such, Filipe de Avila Belbute Peres, Filippo Raso, Florent Bekerman, Foivos Tsimpourlas, Fotis Chantzis, Francis Song, Francis Zhang, Gaby Raila, Garrett McGrath, Gary Briggs, Gary Yang, Giambattista Parascandolo, Gildas Chabot, Grace Kim, Grace Zhao, Gregory Valiant, Guillaume Leclerc, Hadi Salman, Hanson Wang, Hao Sheng, Haoming Jiang, Haoyu Wang, Haozhun Jin, Harshit Sikchi, Heather Schmidt, Henry Aspegren, Honglin Chen, Huida Qiu, Hunter Lightman, Ian Covert, Ian Kivlichan, Ian Silber, Ian Sohl, Ibrahim Hammoud, Ignasi Clavera, Ikai Lan, Ilge Akkaya, Ilya Kostrikov, Irina Kofman, Isak Etinger, Ishaan Singal, Jackie Hehir, Jacob Huh, Jacqueline Pan, Jake Wilczynski, Jakub Pachocki, James Lee, James Quinn, Jamie Kiros, Janvi Kalra, Jasmyn Samaroo, Jason Wang, Jason Wolfe, Jay Chen, Jay Wang, Jean Harb, Jeffrey Han, Jeffrey Wang, Jennifer Zhao, Jeremy Chen, Jerene Yang, Jerry Tworek, Jesse Chand, Jessica Landon, Jessica Liang, Ji Lin, Jiancheng Liu, Jianfeng Wang, Jie Tang, Jihan Yin, Joanne Jang, Joel Morris, Joey Flynn, Johannes Ferstad, Johannes Heidecke, John Fishbein, John Hallman, Jonah Grant, Jonathan Chien, Jonathan Gordon, Jongsoo Park, Jordan Liss, Jos Kraaijeveld, Joseph Guay, Joseph Mo, Josh Lawson, Josh McGrath, Joshua Vendrow, Joy Jiao, Julian Lee, Julie Steele, Julie Wang, Junhua Mao, Kai Chen, Kai Hayashi, Kai Xiao, Kamyar Salahi, Kan Wu, Karan Sekhri, Karan Sharma, Karan Singhal, Karen Li, Kenny Nguyen, Keren Gu-Lemberg, Kevin King, Kevin Liu, Kevin Stone, Kevin Yu, Kristen Ying, Kristian Georgiev, Kristie Lim, Kushal Tirumala, Kyle Miller, Lama Ahmad, Larry Lv, Laura Clare, Laurance Fauconnet, Lauren Itow, Lauren Yang, Laurentia Romaniuk, Leah Anise, Lee Byron, Leher Pathak, Leon Maksin, Leyan Lo, Leyton Ho, Li Jing, Liang Wu, Liang Xiong, Lien Mamitsuka, Lin Yang, Lindsay McCallum, Lindsey Held, Liz Bourgeois, Logan Engstrom, Lorenz Kuhn, Louis Feuvrier, Lu Zhang, Lucas Switzer, Lukas Kondraciuk, Lukasz Kaiser, Manas Joglekar, Mandeep Singh, Mandip Shah, Manuka Stratta, Marcus Williams, Mark Chen, Mark Sun, Marselus Cayton, Martin Li, Marvin Zhang, Marwan Aljubeh, Matt Nichols, Matthew Haines, Max Schwarzer, Mayank Gupta, Meghan Shah, Melody Y. Guan, Melody Huang, Meng Dong, Mengqing Wang, Mia Glaese, Micah Carroll, Michael Lampe, Michael Malek, Michael Sharman, Michael Zhang, Michele Wang, Michelle Pokrass, Mihai Florian, Mikhail Pavlov, Miles Wang, Ming Chen, Mingxuan Wang, Minnia Feng, Mo Bavarian, Molly Lin, Moose Abdool, Mostafa Rohaninejad, Nacho Soto, Natalie Staudacher, Natan LaFontaine, Nathan Marwell, Nelson Liu, Nick Preston, Nick Turley, Nicklas Ansman, Nicole Blades, Nikil Pancha, Nikita Mikhaylin, Niko Felix, Nikunj Handa, Nishant Rai, Nitish Keskar, Noam Brown, Ofir Nachum, Oleg Boiko, Oleg Murk, Olivia Watkins, Oona Gleeson, Pamela Mishkin, Patryk Lesiewicz, Paul Baltescu, Pavel Belov, Peter Zhokhov, Philip Pronin, Phillip Guo, Phoebe Thacker, Qi Liu, Qiming Yuan, Qinghua Liu, Rachel Dias, Rachel Puckett, Rahul Arora, Ravi Teja Mullapudi, Raz Gaon, Reah Miyara, Rennie Song, Rishabh Aggarwal, RJ Marsan, Robel Yemiru, Robert Xiong, Rohan Kshirsagar, Rohan Nuttall, Roman Tsiupa, Ronen Eldan, Rose Wang, Roshan James, Roy Ziv, Rui Shu, Ruslan Nigmatullin, Saachi Jain, Saam Talaie, Sam Altman, Sam Arnesen, Sam Toizer, Sam Toyer, Samuel Miserendino, Sandhini Agarwal, Sarah Yoo, Savannah Heon, Scott Ethersmith, Sean Grove, Sean Taylor, Sebastien Bubeck, Sever Banesiu, Shaokyi Amdo, Shengjia Zhao, Sherwin Wu, Shibani Santurkar, Shiyu Zhao, Shraman Ray Chaudhuri, Shreyas Krishnaswamy, Shuaiqi, Xia, Shuyang Cheng, Shyamal Anadkat, Simón Posada Fishman, Simon Tobin, Siyuan Fu, Somay Jain, Song

Mei, Sonya Egoian, Spencer Kim, Spug Golden, SQ Mah, Steph Lin, Stephen Imm, Steve Sharpe, Steve Yadlowsky, Sulman Choudhry, Sungwon Eum, Suvansh Sanjeev, Tabarak Khan, Tal Stramer, Tao Wang, Tao Xin, Tarun Gogineni, Taya Christianson, Ted Sanders, Tejal Patwardhan, Thomas Degry, Thomas Shadwell, Tianfu Fu, Tianshi Gao, Timur Garipov, Tina Sriskandarajah, Toki Sherbakov, Tomek Korbak, Tomer Kaftan, Tomo Hiratsuka, Tongzhou Wang, Tony Song, Tony Zhao, Troy Peterson, Val Kharitonov, Victoria Chernova, Vineet Kosaraju, Vishal Kuo, Vitchyr Pong, Vivek Verma, Vlad Petrov, Wanning Jiang, Weixing Zhang, Wenda Zhou, Wenlei Xie, Wenting Zhan, Wes McCabe, Will DePue, Will Ellsworth, Wulfie Bain, Wyatt Thompson, Xiangning Chen, Xiangyu Qi, Xin Xiang, Xinwei Shi, Yann Dubois, Yaodong Yu, Yara Khakbaz, Yifan Wu, Yilei Qian, Yin Tat Lee, Yinbo Chen, Yizhen Zhang, Yizhong Xiong, Yonglong Tian, Young Cha, Yu Bai, Yu Yang, Yuan Yuan, Yuanzhi Li, Yufeng Zhang, Yuguang Yang, Yujia Jin, Yun Jiang, Yunyun Wang, Yushi Wang, Yutian Liu, Zach Stubenvoll, Zehao Dou, Zheng Wu, and Zhigang Wang. Openai gpt-5 system card, 2026. URL https://arxiv.org/abs/2601.03267.

Ada Defne Tur, Nicholas Meade, Xing Han Lù, Alejandra Zambrano, Arkil Patel, Esin DURMUS, Spandana Gella, Karolina Stanczak, and Siva Reddy. Safearena: Evaluating the safety of autonomous web agents. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=7TrOBcxSvy.

Tu Vu, Mohit Iyyer, Xuezhi Wang, Noah Constant, Jerry W. Wei, Jason Wei, Chris Tar, Yun-Hsuan Sung, Denny Zhou, Quoc V. Le, and Thang Luong. Freshllms: Refreshing large language models with search engine augmentation. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 13697–13720, 2024.

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

Ryan Wong, Jiawei Wang, Junjie Zhao, Li Chen, Yan Gao, Long Zhang, Xuan Zhou, Zuo Wang, Kai Xiang, Ge Zhang, Wenhao Huang, Yang Wang, and Ke Wang. Widesearch: Benchmarking agentic broad info-seeking. *arXiv preprint arXiv:2508.07999*, 2025.

Jialong Wu, Wenbiao Yin, Yong Jiang, Zhenglin Wang, Zekun Xi, Runnan Fang, Linhai Zhang, Yulan He, Deyu Zhou, Pengjun Xie, and Fei Huang. Webwalker: Benchmarking llms in web traversal. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 10290–10305, 2025. doi: 10.18653/v1/2025.acl-long.508. URL https://aclanthology.org/2025.acl-long.508/.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388.

Shunyu Yao, Howard Chen, John Yang, and Karthik Narasimhan. Webshop: Towards scalable real-world web interaction with grounded language agents. *Advances in Neural Information Processing Systems*, 35:20744–20757, 2022.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models, 2023. URL https://arxiv.org/abs/2210.03629.

Ori Yoran, Samuel Joseph Amouyal, Chaitanya Malaviya, Ben Bogin, Ofir Press, and Jonathan Berant. Assistantbench: Can web agents solve realistic and time-consuming tasks? In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 8938–8968, 2024.

Joey Zhong, Hao Zhang, Clare Southern, Jeremy Yang, Thomas Wang, Kate Jung, Shu Zhang, Denis Yarats, Johnny Ho, and Jerry Ma. Draco: a cross-domain benchmark for deep research accuracy, completeness, and objectivity, 2026. URL https://arxiv.org/abs/2602.11685.

Shuyan Zhou, Frank F. Xu, Hao Zhu, Xuhui Zhou, Robert Lo, Abishek Sridhar, Xianyi Cheng, Tianyue Ou, Yonatan Bisk, Daniel Fried, Uri Alon, and Graham Neubig. Webarena: A realistic web environment for building autonomous agents. In *The Twelfth International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=oKn9c6ytLx.

A Answer Canonicalization and Metric Definitions

Canonicalization rules. Following the consensus procedure in Section 3.4, we canonicalize alternative answer representations according to the task schema. Equivalence is determined by the information required by each field. Formatting changes preserve the underlying value, and task-specific aliases are accepted only when the answer-bearing source establishes their equivalence for that field. Table 3 summarizes the rule categories, illustrative conversions, and conditions for acceptance.

Table 3: Answer canonicalization rules. Each conversion is valid only under the stated field definition and acceptance conditions.

Rule category	Illustrative conversion	Conditions for acceptance
Whitespace, punctuation, and capitalization	Leading or trailing spaces are removed; `Holotype` → `HOLOTYPE`	The change affects presentation only. Meaningful characters in names, identifiers, and values are preserved.
Date formatting	July 11, 2023 → 2023-07-11	Both expressions unambiguously denote the same calendar date.
Numeric and unit formatting	\$80.4 million → \$80,400,000	Both expressions denote exactly the same amount in the same currency. The conversion introduces no rounding or approximation.
Abbreviation variants	Phase 1 → P1	The field definition or source establishes the abbreviation, preserving the category and level of specificity.
Task-specific aliases	A source-listed alternative name → the canonical entity name	The source confirms that both names identify the same entity, and the field does not require a particular name variant.
Explicit status expressions	Holotype: BMNH 1946.1.13.61 → HOLOTYPE	The field requests the BMNH type status, and that status is explicitly stated in the prediction.

Metric computation. All metrics are computed after the answer canonicalization described above and task-specific row alignment. Let the canonical reference answer be represented as a structured row sequence $\mathcal{R}_{\text{gold}} = (r_1, \ldots, r_m)$ and the canonical prediction as $\mathcal{R}_{\text{pred}} = (\hat{r}_1, \ldots, \hat{r}_n)$, where each row is a tuple of field values under the task schema. Rows are aligned by task-specific row keys defined over the primary identifying fields, and all field-level comparisons are performed after this alignment, following the structured-output evaluation style used in recent broad and deep web-search benchmarks Wong et al. (2025); Gupta et al. (2026); Li et al. (2026).

Item-level F1. Let C_{item} denote the number of field slots whose canonicalized values exactly match the corresponding reference values after row alignment, and let N_{gold} and N_{pred} denote the total numbers of field slots in the reference and prediction, respectively. We define

$$\text{Item-F1} = \frac{2C_{\text{item}}}{N_{\text{gold}} + N_{\text{pred}}}.$$

This metric gives partial credit when some but not all field values are correct.

Row-level F1. Let C_{row} denote the number of aligned rows for which every canonicalized field value exactly matches its corresponding reference value. We define

$$\text{Row-F1} = \frac{2C_{\text{row}}}{|\mathcal{R}_{\text{gold}}| + |\mathcal{R}_{\text{pred}}|}.$$

A row contributes to C_{row} only if all of its fields exactly match the reference after canonicalization and alignment.

Pairwise Order Accuracy (P.O.A.). Let $\mathcal{S}$ denote the set of row keys shared by the prediction and the reference. Let $\pi(u)$ and $\hat{\pi}(u)$ denote the positions of a shared row key $u \in \mathcal{S}$ in the reference and prediction, respectively, and let

$$\mathcal{P}(\mathcal{S}) = \{(u, v) : u, v \in \mathcal{S},\ \pi(u) < \pi(v)\}$$

be the set of comparable ordered row pairs induced by the reference order. We define

$$\text{P.O.A.} = \begin{cases} \frac{1}{|\mathcal{P}(\mathcal{S})|} \sum_{(u,v)\in\mathcal{P}(\mathcal{S})} \mathbf{1}[\hat{\pi}(u) < \hat{\pi}(v)], & |\mathcal{S}| \geq 2, \\ 0, & |\mathcal{S}| < 2. \end{cases}$$

P.O.A. therefore measures the fraction of shared-row pairs whose relative order is preserved by the prediction, following the pairwise rank-agreement perspective underlying Kendall's τ Kendall (1938). When fewer than two row keys are shared, relative order is not evaluable; in this case, we assign $\text{P.O.A.} = 0$ as a conservative convention, treating the prediction as providing no correct ordering information.

B Task Construction, Quality Control, and Examples

Contributors and responsibilities. The authors were responsible for website curation, protocol specification, preliminary task filtering, and final task inclusion decisions. We recruited 24 paid doctoral students from universities ranked among the top 20 in the QS World University Rankings and assigned tasks according to their academic backgrounds and research experience. Each task was constructed by one participant and independently reviewed by three other participants, one in each validation round.

Design protocol and review criteria. Two authors developed the task-design protocol described in Section 3.2, comprising six requirements and two exemplar tasks. The operational rubric included two checks for assessing retrieval necessity. Under the counterfactual requirement, constructors specified the systematic error that would result from omitting a key retrieval step. Under the interaction requirement, reviewers checked that each state-setting action materially changed the candidate space. These checks translated the protocol's requirements for state-gated retrieval and dependencies between retrieval steps into concrete review criteria.

Validation and revision. Submitted tasks were first screened for difficulty and compliance with the design protocol. Candidates that passed this screening underwent the three validation rounds described in Section 3.2: independent solution and reference-answer verification, assessment of the necessity of state-gated retrieval, and adversarial checks for shortcut solutions. Candidates that failed a round were revised and revalidated or removed. Each accepted task underwent an average of 3.8 submission–feedback–revision cycles. Because the three validation rounds assessed complementary criteria rather than collecting repeated labels for the same judgment, we do not report conventional inter-annotator agreement for this stage.

Workload and compensation. The first five pilot tasks indicated that constructing a task package required more than ten hours of work. Based on this observed workload, constructors received approximately USD 222 for each accepted task package and USD 44 for each package that was not retained. Reviewers were paid approximately USD 15 per hour, and each retained task required approximately six hours of paid review across the three validation rounds.

Task examples. We illustrate the resulting paired task design with an example from the Europe PMC ecosystem. Both variants share the same reference answer, evidence requirements, and output schema; they differ only in how much retrieval procedure is made explicit in the prompt.

Constraint-guided variant (europemc_003). The instruction decomposes the retrieval into explicit steps: (1) find the 2023 and 2024 open-access *Antibodies to Watch* articles on Europe PMC; (2) extract all candidates predicted to file first marketing applications in 2022–2023 from the 2023 article; (3) cross-reference each candidate against the 2024 article and retain only those that have received first approval or entered first regulatory review; (4) output each qualifying molecule with its 2023 category, 2023 prediction label, 2024 status (A for approved, R for under review), 2024 indication, and 2024 region, sorted alphabetically by INN.

Goal-oriented variant (europemc_003-g). The instruction states the same information need but omits the step-by-step decomposition: "I want to see which candidate molecules predicted in the 2023 *Antibodies to Watch* to file in 2022–2023 have actually been approved or entered first regulatory review in the 2024 article of the same series." The agent must independently determine how to locate both articles, extract the 2023 forecast cohort, and reconcile it against the 2024 outcomes.

Reference answer. The reference answer contains 8 rows, each representing one qualifying antibody candidate. The answer requires cross-referencing two separate article tables: Table 1 (first approvals) and Table 2 (first regulatory reviews) from the 2024 article, matched against the 2023 forecast baseline. This inter-table dependency is a representative instance of the logical-dependency design requirement.

C Detailed Experimental Setup

Controlled CLI evaluation scope. The configuration described here applies to the nine CLI-based systems used in the main evaluation: Kimi K2.5, GLM-5.1, Qwen3.6-Plus, DeepSeek V4 Pro, Seed-2.0 Pro, Claude Opus 4.7, Gemini 3.1 Pro, GPT-5.5, and GPT-5.6-sol. All nine are evaluated under a common search–fetch–PDF retrieval setup. Observed performance differences therefore reflect system behavior under a matched retrieval-tool regime rather than differences in the available external tools.

Prompt template and controlled protocol. All agents receive the same task prompt, consisting of the current date, the benchmark task instruction, and the required structured output schema. Prompts do not provide a start URL, and they instruct agents to solve the task through the designated search, fetch, and PDF tools while prohibiting alternative retrieval paths.

Exposed tools and runtime restrictions. Across all nine CLI systems, the runtime exposes only three external retrieval tool classes: Serper search, fetch-based webpage reading, and PDF reading. For the Claude Code systems, each run uses a project-level MCP configuration under a strict MCP configuration, so that only the designated search, fetch, and PDF tools are available during execution. A companion settings file additionally disables the Claude-native `WebSearch` and `WebFetch` tools, ensuring that all evaluated systems operate under the same external retrieval-tool budget. GPT-5.5 and GPT-5.6-sol use the official OpenAI API through Codex CLI, with the same prompt constraints, exposed retrieval tools, task isolation, and runtime budget as the other CLI systems.

Execution budget and task isolation. Each task is executed in an independent session, with no context shared across tasks. The maximum runtime per task is 6000 seconds; runs are marked as stalled after 3000 seconds without trace progress; and answer extraction is polled every 5 seconds during execution.

API settings and cost. All nine CLI-based systems use a maximum output length of 32,000 tokens per API request. Reasoning effort is set to medium for GPT-5.5 and GPT-5.6-sol, and for other models where configurable; DeepSeek V4 Pro uses high because its available settings are high and max. We retain the model or API default temperatures listed in Table 4. Qwen3.6-Plus uses a temperature of 0.6 in Thinking mode, whereas temperature has no effect for DeepSeek V4 Pro in Thinking mode. For all nine CLI-based systems, mean model API costs are reported per task for a single run over the 134-task evaluation set. Evaluating all nine systems on that evaluation set costs approximately USD 3,450.50.

Table 4: API access, default temperatures, and mean model API cost per task for all nine CLI-based systems. Costs refer to one run per system on the 134-task evaluation set.

Model	API platform	Default temperature	Cost/task (USD)
Kimi K2.5	OpenRouter	1.0	1.02
GLM-5.1	OpenRouter	1.0	3.96
Qwen3.6-Plus	OpenRouter	0.6	1.46
DeepSeek V4 Pro	OpenRouter	Not effective	3.41
Seed-2.0 Pro	OpenRouter	1.0	0.24
Claude Opus 4.7	OpenRouter	1.0	6.17
Gemini 3.1 Pro	OpenRouter	1.0	1.73
GPT-5.5	OpenAI	1.0	3.81
GPT-5.6-sol	OpenAI	1.0	3.95

Commercial system evaluation. Google Search AI Mode, Gemini Deep Research, and OpenAI Deep Research Google (2025; 2026); OpenAI (2025) are evaluated through manual interaction with their respective web interfaces. The task prompt is provided as-is, and the final output is collected without modification. No intermediate trajectory data is available for these systems.

Public release and reproduction instructions. The public release separates benchmark data from evaluation infrastructure. The current Hugging Face release contains 134 English task records and their reference answers, together with Chinese counterparts and loading instructions. The experiments in this paper cover all 134 task records. Companion reproduction instructions describe the MCP configuration, required API keys, task loading procedure, single-case execution protocol, and single-case scoring procedure. We do not release the full internal batch-running infrastructure or the complete trajectory corpus, since those artifacts include implementation-specific orchestration details and mixed-language traces that are not part of the benchmark definition. Instead, the release focuses on the dataset itself and on a minimal, inspectable path for reproducing individual benchmark cases.

D LIMITATIONS

The current release of SGR-BENCH reflects deliberate trade-offs among coverage, diagnostic precision, and reproducibility. These choices make the benchmark suitable for controlled evaluation of state-gated retrieval.

Coverage favors stable public interfaces over rapidly changing web content. To ensure stable ground truth, reproducible evidence verification, and reliable scoring, we prioritize public sources whose relevant tables, records, and filtering logic remain comparatively stable over time. This design choice improves annotation quality and benchmark longevity, but it underrepresents retrieval settings driven by highly dynamic content, such as breaking news, live operational dashboards, rapidly refreshed public records, or interfaces whose ranking and availability change substantially within short time windows. As a result, SGR-BENCH should be interpreted as a benchmark for *structured, stateful retrieval under relatively stable public interfaces*, rather than as a comprehensive proxy for all real-time web-search environments.

Trajectory evaluation is constrained by trace availability. The commercial systems evaluated here do not expose the intermediate execution traces needed for comparable trajectory-level evaluation. We therefore apply final-answer metrics to all systems, audit evidence grounding across the nine CLI-based systems, and analyze failure patterns within an eight-system subset. These analyses concern the audited trajectories; their findings do not extend to the unobserved execution processes of commercial systems.

Benchmark scale is intentionally controlled, and is not designed for large-scale post-training by itself. Because benchmark construction requires domain-specific candidate discovery, task-specific decomposition, reference-answer verification, and expert review of shortcut resistance, the current release remains modest in scale relative to corpora intended for large-scale model optimization. At the same time, it means that the benchmark is not intended to serve as a standalone resource for

high-volume post-training pipelines, including reinforcement-learning-based optimization, which typically require substantially larger and more diverse task collections. In that sense, SGR-BENCH is better viewed as an evaluation and diagnosis resource than as a self-sufficient training corpus.

LLM-assisted drafting may still introduce construction-time biases. LLMs are used only in a restricted supporting role during benchmark construction, such as candidate prioritization and draft generation, while human experts retain responsibility for website verification, answer validation, shortcut-resistance checks, and final inclusion decisions. Even under this workflow, the benchmark may still inherit subtle drafting biases from the assisting models, including biases in website selection, question phrasing, candidate structuring, or initial decomposition style. Human review substantially reduces these risks and serves as the primary quality-control mechanism, but it cannot guarantee complete removal of all construction-time biases. This limitation should be kept in mind when interpreting both source-family coverage and the stylistic regularities of benchmark tasks.

E FULL PER-MODEL RESULTS

Table 5 reports per-model averages across all tasks for the evaluated CLI-based and commercial systems.

Table 5: Per-model performance, sorted by mean Item-F1. Item-F1 and P.O.A. average the displayed CG and GO values in Table 2; Row-F1 uses its Overall column.

Model	Item-F1	Row-F1	P.O.A.
GPT-5.6-sol	71.23	62.49	90.93
GPT-5.5	66.19	43.37	90.41
GLM-5.1	65.64	33.32	87.25
Claude Opus 4.7	61.38	38.51	81.32
DeepSeek V4 Pro	60.98	32.64	85.00
Gemini 3.1 Pro	59.07	27.11	90.26
OpenAI Deep Research	54.21	41.50	67.18
Kimi K2.5	46.36	33.74	77.28
Qwen3.6-Plus	36.88	11.62	60.86
Gemini Deep Research	29.94	10.27	45.45
Seed-2.0 Pro	29.88	20.46	36.36
Google Search AI Mode	14.87	2.36	43.94

The results reveal a clear tier structure. The top tier (GPT-5.6-sol, GPT-5.5, GLM-5.1, Claude Opus 4.7, DeepSeek V4 Pro) achieves Item-F1 above 60%, while GPT-5.6-sol achieves the highest Overall Row-F1 at 62.49% (Table 2). The middle tier (Gemini 3.1 Pro, OpenAI Deep Research) clusters around 54–59% Item-F1. The bottom tier (Kimi K2.5, Qwen3.6-Plus, Gemini Deep Research, Seed-2.0 Pro, Google Search AI Mode) falls below 48%, with Google Search AI Mode achieving only 14.87% Item-F1.

Notably, OpenAI Deep Research achieves the third-highest Row-F1 (41.50%) despite ranking seventh in Item-F1 (54.21%), suggesting that commercial deep-research systems may be better at assembling complete rows from the evidence they do retrieve. Conversely, Gemini 3.1 Pro achieves high P.O.A. (90.26%) but relatively low Row-F1 (27.11%), indicating correct ordering of retrieved rows but frequent row-level omissions.

F EXTENDED CASE STUDIES

We first present six brief case studies: one success case, one failure case from the main text, and four additional failure cases. We then provide a seven-step audit of Qwen3.6-Plus on `reptile_001-g`, tracing the initial retrieval-state failure through subsequent searches to the final answer (Table 6).

Success case: GPT-5.5 on reptile_001. This task requires cross-referencing Boulenger's 1890 Indian snake descriptions against the Reptile Database to determine current nomenclature and type

specimen status for each species. The reference answer contains 8 species with original names, current names, type status (UNIQUE or AMBIG), and British Museum specimen numbers.

GPT-5.5 achieved perfect scores (Item-F1 = Row-F1 = P.O.A. = 100%). The trajectory shows the agent correctly: (1) searched for "Boulenger 1890 India snakes" on the Reptile Database; (2) configured the advanced search with Author = Boulenger, Year = 1890, Distribution = India; (3) visited each species page to extract current nomenclature and type information; (4) classified each as UNIQUE or AMBIG based on type-specimen information. This trajectory demonstrates successful state-gated retrieval: the agent identified the correct website, configured the site-specific retrieval state through conjunctive filter constraints, and maintained consistent scope across all 8 species pages.

Failure case: GPT-5.5 on waterquality_003-g. This goal-oriented task requires retrieving water quality monitoring data from the USGS Water Quality Portal for specific HUC codes and parameters. GPT-5.5 scored zero on all metrics, with no common row identifiers between its output and the reference.

Trajectory analysis reveals a retrieval-scope drift failure: the agent identified the correct website (Water Quality Portal) but configured incorrect filter parameters, retrieving data for wrong monitoring stations. This error propagated through the entire output, making all subsequent extraction and formatting irrelevant. The case illustrates how a single upstream state-configuration error can invalidate an otherwise well-structured retrieval trajectory.

Retrieval-scope drift: Seed-2.0 on reptile_001. This task required querying the Reptile Database using a structured advanced search (Author=Boulenger, Year=1890, Distribution=India) to enumerate all 12 candidate species and extract taxonomic status fields from each official species page. The agent bypassed the required official root query entirely and instead relied on general-purpose search engine snippets to identify candidates, reducing the 12-candidate systematic pipeline to a single-object guessing workflow. This scope collapse propagated through every downstream stage: only 2 of 12 species pages were visited, the sole output row contained incorrect synonym and type-specimen fields, and 7 of 8 gold rows were entirely absent (Item-F1 = 11.1%, Row-F1 = 0%).

Criterion mismatch: Gemini 3.1 on consumerfinance_012. This task required querying the CFPB Complaint Database with the filter `search_term = medical OR doctor` to obtain a 585-complaint cohort, then performing hierarchical aggregation to identify peak complaint months, top states, and top companies. The earliest error occurred when the agent translated the textual search criterion into the API query `medical bills OR doctors` instead of the required `medical OR doctor`, yielding a spurious 2,311-complaint cohort. Although the agent successfully completed the full analytical pipeline, every downstream computation operated on the incorrect base population. All output slots diverged from the reference (Item-F1 = 0%, Row-F1 = 0%).

Intent rewriting: Claude 4.7 on arxiv_003. This task required an exhaustive search of arXiv for papers with v1 submissions in 2020 Q4 under cs.CV whose titles or abstracts mention Vision Transformer variants, followed by per-paper verification of submission history and publication trail. The primary model delegated the entire task to a sub-agent with a prompt that replaced the required exhaustive search with a pre-seeded list of "well-known early Vision Transformer papers," omitting two gold-standard papers (2101.01097 and 2011.08019) from the candidate space entirely. The sub-agent returned a 3-row summary that the primary model accepted without secondary verification, permanently fixing the incomplete candidate set as the final answer (Item-F1 = 68.8%, Row-F1 = 50.0%).

Retrieval dependency: Kimi K2.5 on europemc_002. This task required cross-referencing three annual Alzheimer's drug pipeline reviews (2021–2023) from Europe PMC, tracking 24 Phase 1 clinical trials by NCT identifiers to detect phase transitions and drug name changes. The agent successfully located all three correct PMC articles and extracted the relevant tables, including the critical evidence that NCT03634007 (originally AAVrh.10hAPOE2, renamed LX1001) advanced to Phase 2 in 2023. However, during local evidence integration, the analysis script failed to map the 2022 entry back to the 2021 baseline due to the drug name change, erroneously removing it from the candidate set. The final answer contained 5 of 6 gold rows but missed the single most diagnostically important trial, the only one that changed phase across years (Row-F1 = 90.9%).

Detailed trajectory audit: Qwen3.6-Plus on reptile_001-g. We examine Qwen3.6-Plus on `reptile_001-g`, a goal-oriented task concerning Indian snakes documented by Boulenger in 1890. The task asks for species that have changed genus, retain BMNH type-specimen information, and have an India-compatible type locality. For each qualifying species, the answer must give the current accepted name, the most recent historical binomial outside the current genus, whether that historical name uniquely identifies the species, and the BMNH type status.

The reference workflow first combines *Higher taxa* = snake, *Author* = Boulenger, *Year* = 1890, and *Distribution* = India, yielding 12 candidates. Checking their species pages leaves eight qualifying records. Exact-name searches then determine UNIQUE or AMBIG while preserving the original species identity. These controls implement the task goal; the goal-oriented prompt does not prescribe the individual form operations.

Table 6 summarizes the recorded trajectory in seven audit stages, rather than seven individual tool calls. The observations refer to fetched page content and query parameters. The reference states are drawn from the task's verification checkpoints. Stage 4 summarizes species-page coverage across the run; stages 5–6 isolate subsequent name checks and evidence omissions.

For this case, mapping the verbose BMNH descriptions to HOLOTYPE, LECTOTYPE, or SYNTYPES yields Item-F1 = 58.33%, Row-F1 = 33.33%, and P.O.A. = 100%. This normalization preserves the incorrect historical name and uniqueness judgment. The perfect ordering score concerns only shared rows and therefore does not reflect the four omissions. The case illustrates the distinction behind Findings 1–2: accessible pages and locally correct evidence can coexist with incomplete retrieval scope and an incorrect final answer. The trace establishes that *I. jerdoni* was retrieved but omitted; it does not establish the model's internal reason for that omission.

G Trajectory Audit Methods and Error Categories

To characterize recurrent failure patterns, we manually audited a subset of eight CLI-based systems on the same set of 22 randomly sampled tasks, yielding 176 trajectories: 156 failed runs and 20 correct runs. Commercial systems were excluded because their intermediate execution traces were unavailable for this analysis. The error analysis below concerns the 156 failed trajectories in this audited subset.

Each failed run receives a single primary error label corresponding to the earliest non-recoverable root cause. When an upstream error leads to downstream answer-writing errors, we label the upstream error rather than its consequences. We use the following six categories.

Retrieval-Scope Drift. The agent fails to establish or preserve the correct retrieval workspace, such as the required object set, candidate space, jurisdiction, time range, or result scope. Under this label, later extraction steps may be locally plausible, but they are grounded in the wrong slice of the source.

Criterion Mismatch. The agent reaches relevant resources but applies an incorrect decision rule, field definition, aggregation level, denominator, phase space, or time window. The resulting answers often contain locally correct values, but they are bound to the wrong evaluative criterion.

Intent Rewriting. The agent implicitly replaces the original task with an easier but non-equivalent surrogate objective, typically by dropping required constraints, weakening exhaustive search requirements, or converting structured retrieval into approximate summarization.

In-Page Evidence Misreading. The agent reaches the correct page or table but misreads a local value, label, priority rule, or field interpretation on that page. This category is reserved for errors that can be localized to incorrect page-internal evidence reading rather than to broader scope or criterion selection.

Retrieval Dependency Errors. The agent obtains key intermediate evidence but fails to close the dependency chain needed for correct completion, such as cross-page alignment, backtracking, evidence propagation, or final candidate arbitration. The central issue is not missing access, but incomplete dependency resolution across retrieval steps.

Table 6: Seven-stage audit of Qwen3.6-Plus on `reptile_001-g`. The comparison separates observed behavior from the reference retrieval state and its implications for the answer.

Step	Observed action and evidence	Reference action or state	Audit finding	Consequence
1	**Inspected the search controls.** The advanced-search page exposed Higher taxa, Author, Year, Distribution, and Common name or synonym, with Exact match options.	Identify the controls needed to delimit candidates and verify historical names.	The relevant source and controls were accessible.	Source discovery succeeded.
2	**Attempted an incomplete initial query.** The query specified snakes, Boulenger, and 1890, but omitted India. The response contained site navigation, with no species list.	Combine all four filters in advanced search to establish the 12-candidate set.	**First persistent deviation:** the initial candidate set was never established.	Later searches lacked a complete candidate list to check against.
3	**Continued without recovering the candidate set.** The model searched the web for site-specific pages mentioning snakes, Boulenger, and 1890, and retried the database query without the distribution filter.	Recover the complete candidate set and check coverage against it throughout retrieval.	Candidate discovery remained open-ended.	Out-of-scope species, including *Lycodon travancoricus* and *Sibynophis collaris*, entered the investigation.
4	**Read only part of the candidate set.** Official pages were fetched for seven of the 12 candidates: *Spalerosophis arenarius*, *Hebius khasiensis*, *H. parallelus*, *Fowlea sanctijohannis*, *Dendrelaphis bifrenalis*, *Indotyphlops jerdoni*, and *Gerrhopilus beddomii*. Some evidence was read correctly, including the historical name and BMNH lectotype for *H. khasiensis*.	Inspect each candidate's Synonym, Type locality, and Types fields; retain only records meeting all inclusion criteria.	Correct local evidence did not ensure complete coverage.	Official pages for three qualifying species were never fetched: *Argyrophis oatesii*, *Dendrelaphis grandoculis*, and *Myriopholis blanfordii*.
5	**Used the wrong name-check state.** For *H. parallelus*, the model queried *Tropidonotus parallela* without Exact match. The query returned the species page; the final answer labeled the name UNIQUE.	Search for the task-selected name *Amphiesma parallelum* with Exact match. The reference search returns three species, requiring AMBIG for the original candidate.	Both the query term and matching mode differed from the reference state.	*H. parallelus* was labeled UNIQUE instead of AMBIG.
6	**Omitted evidence already retrieved.** A later non-exact search for *Typhlops jerdoni* again returned the *I. jerdoni* page, including the Boulenger 1890 India locality and BMNH syntypes. This species was absent from the answer.	Associate the Boulenger 1890 locality and BMNH type evidence with *I. jerdoni*, and retain its qualifying row.	The trace shows an omission despite successful evidence retrieval.	A fourth qualifying species was missing from the answer.
7	**Returned four of eight required rows.** In addition to the four omissions above, the answer used the earlier synonym *Typhlops beddomei* instead of *T. beddomii*, assigned the wrong uniqueness label to *H. parallelus*, and gave full specimen descriptions instead of type-status labels.	Return all eight qualifying records, sorted by accepted name, with the selected historical name, UNIQUE/AMBIG judgment, and normalized BMNH status.	The final answer combined coverage gaps, name-selection errors, and output-format deviations.	After status normalization, only two predicted rows were fully correct.

Final Answer Composition. The upstream retrieval and interpretation are largely correct, but the final answer is corrupted during aggregation, slot mapping, sorting, normalization, deduplication, or transcription. We assign this label only when earlier stages are substantially correct and the failure is concentrated in the final answer assembly step.

H Error Profile by Model

Table 7 and Figure 4 decompose the six error types across the eight CLI-based models in the audited subset. The 176 annotated trajectories include 156 failures and 20 correct cases.

Table 7: Error type counts per model across 176 annotated trajectories (22 task slots per model). Percentages are computed over failures only (excluding Correct).

Model	Scope Drift	Crit. Mis.	Intent Rew.	Page Mis.	Dep. Err.	Final Ans.	Fail	Corr.
Seed-2.0	15	1	2	1	1	0	20	2
Gemini 3.1	11	6	0	0	0	2	19	3
Kimi K2.5	9	4	1	3	5	0	22	0
Qwen3.6+	7	4	2	2	0	5	20	2
GPT-5.5	5	6	2	2	1	3	19	3
GLM-5.1	5	8	1	1	1	3	19	3
DS V4 Pro	4	8	3	1	2	1	19	3
Claude 4.7	2	6	5	2	1	2	18	4
Total	**58**	**43**	**16**	**12**	**11**	**16**	**156**	**20**

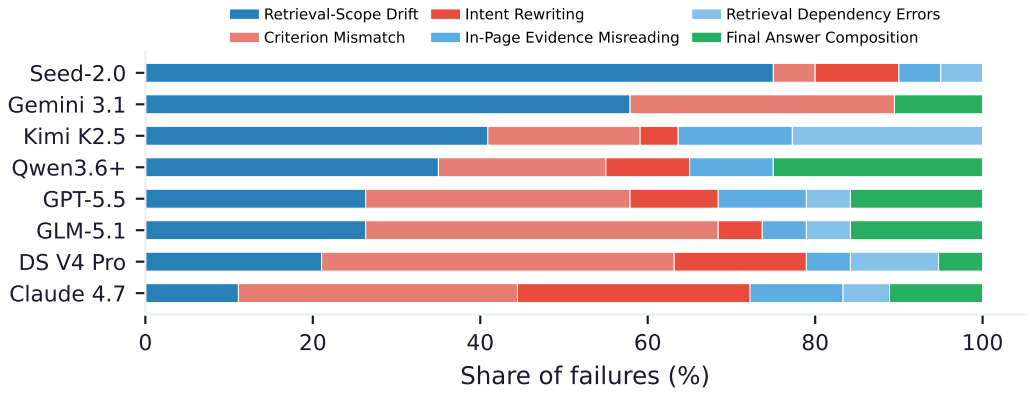

Figure 4: Share of each error type within each model's failures. Models are sorted by retrieval-scope drift share (descending). Seed-2.0 and Gemini 3.1 are dominated by scope drift; DS V4 Pro and GLM-5.1 concentrate on criterion mismatch; Claude 4.7 uniquely concentrates on intent rewriting.

Three qualitative clusters emerge. *Drift-dominant* models (Seed-2.0 at 75%, Gemini 3.1 at 58%) fail primarily in establishing the correct initial retrieval workspace. *Mismatch-dominant* models (DS V4 Pro and GLM-5.1, each 42%) reach relevant resources but apply incorrect field definitions. *Mixed-profile* models (GPT-5.5, Claude 4.7) distribute failures more evenly, with Claude 4.7 uniquely exhibiting high intent rewriting (28%), reflecting its tendency to delegate subtasks to subagents that rephrase the original query. Kimi K2.5 is the only model with zero correct trajectories across all 22 tasks, and has the highest retrieval-dependency failure rate (23%), suggesting difficulty maintaining evidence chains across dependent retrieval steps.

I Error Profile by Source Family

Table 8 and Figure 5 break down error types by source family for the audited trajectory subset. The 22 audited task slots cover five source-family groupings; vulnerability-database tasks are part of the full benchmark but are not included in this trajectory audit.

Table 8: Error type counts per source-family grouping across 156 failed trajectories in the audited subset.

Source Family	Scope Drift	Crit. Mis.	Intent Rew.	Page Mis.	Dep. Err.	Final Ans.	Total
Scholarly	35	7	8	6	5	3	64
Official stats	9	12	0	0	0	11	32
Life-science	3	11	0	6	3	1	24
Environmental	11	0	8	0	1	1	21
Regulatory	0	13	0	0	2	0	15

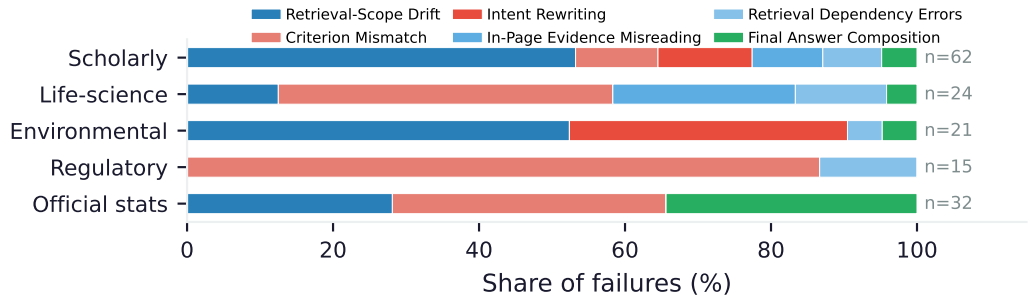

Figure 5: Error type distribution per source family. Scholarly and environmental tasks are drift-dominated; regulatory and life-science tasks are mismatch-dominated; official statistics split between criterion mismatch and final answer composition.

Scholarly archive tasks concentrate failures in retrieval-scope drift (55%), reflecting the challenge of establishing correct candidate spaces via advanced search filters. Regulatory tasks (CFPB) show 87% criterion mismatch: agents reach the correct API or data source but apply incorrect query parameters. Official-statistics tasks (Census) split between criterion mismatch (38%) and final answer composition (34%), the latter reflecting that Census tasks require assembling multiple API fields into correctly formatted output rows, a step that agents complete semantically but fail to format correctly. Environmental tasks (Water Quality Portal) uniquely combine scope drift (52%) with intent rewriting (38%), as agents often simplify multi-HUC comparison tasks into single-station lookups.

J Error Type and Task Performance

Figure 6 shows how each error type manifests in Item-F1 and Row-F1 scores. Table 9 provides the corresponding statistics.

The most revealing pattern is criterion mismatch: it achieves moderate Item-F1 (52.5%) but the lowest Row-F1 among all error types (7.3%), yielding a 45.2-point gap. This means agents with criterion mismatch errors still extract many locally correct field values, but because the underlying field definitions or decision rules are wrong, almost no row is fully correct. By contrast, retrieval-scope drift depresses both metrics roughly equally (38.7% Item-F1, 19.2% Row-F1), consistent with global workspace errors that corrupt all downstream evidence. Correct trajectories confirm the metrics: 95.0% on both Item-F1 and Row-F1.

Table 9: Mean performance by error type. Upstream errors (scope drift, intent rewriting) produce lower Item-F1, while criterion mismatch preserves moderate Item-F1 but collapses Row-F1.

Error Type	n	**Item-F1**	**Row-F1**	**Gap**
Retrieval-Scope Drift	58	38.7	19.2	19.5
Criterion Mismatch	43	52.5	7.3	45.2
Intent Rewriting	16	31.4	22.3	9.1
In-Page Evidence Misreading	12	62.8	43.1	19.7
Retrieval Dependency Errors	11	58.5	49.6	8.9
Final Answer Composition	16	58.5	26.4	32.1
Correct	20	95.0	95.0	0.0

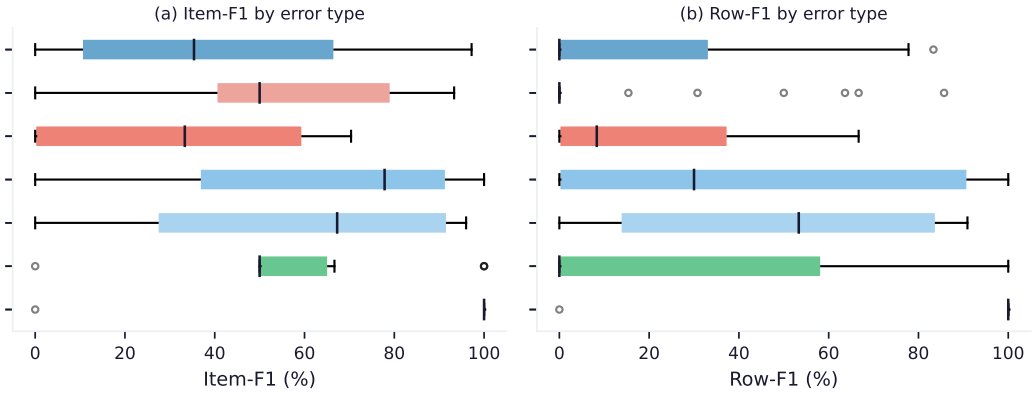

Figure 6: Item-F1 and Row-F1 distributions grouped by error type. Criterion mismatch shows the widest Item-F1 vs. Row-F1 gap: agents recover many field values but almost no complete rows. Upstream errors (scope drift, intent rewriting) depress both metrics.