# DRBENCH PRO: A KNOWLEDGE-CENTRIC BENCHMARK FOR PROFESSIONAL DEEP RESEARCH AGENTS

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Deep research agents gather information from multiple sources and synthesize it into reports on complex, open-ended questions. Existing benchmarks assess report quality along several dimensions, but lack a dedicated knowledge-centric evaluation of which research directions a report covers and how deeply it explores them. We introduce **DRBench Pro**, a benchmark that makes structured knowledge coverage the primary evaluation target. Our automated pipeline builds local knowledge trees from Wikipedia and uses sampled subgraphs to generate research queries and evidence-grounded rubrics. Each rubric corresponds to a relation in the tree, preserving the structure of the knowledge being evaluated. Agents receive only the research query. This structure allows us to measure *breadth* across major research directions and *depth* along related knowledge chains, with their geometric mean giving *overall coverage*. The benchmark includes 144 cases across 24 domains and 44,102 rubrics. Evaluating 16 deep research agents reveals different strengths in breadth and depth, with the best-performing agent reaching 57.04% overall coverage. Rubric judgments agree closely with human annotations, and agent rankings remain stable across the tested metric configurations.

## 1 INTRODUCTION

Deep research agents address complex, open-ended information needs through iterative search, evidence selection, and long-form synthesis. They can investigate a topic across multiple sources and produce research reports that connect relevant facts, explain relationships, and organize findings into a coherent account. Unlike short-form question answering, this task requires an agent to decide which research directions deserve investigation, what evidence to gather, and which related knowledge to pursue further. A comprehensive report therefore depends on both covering the major directions relevant to a question and following chains of related knowledge within those directions.

Evaluating deep research is challenging because open-ended questions admit multiple valid reports, while a single reference answer rarely captures all relevant knowledge. Factuality checks can identify unsupported statements but cannot penalize omitted knowledge (Min et al., 2023; Wei et al., 2024; Song et al., 2024b). Existing benchmarks therefore use expert reports, reference facts, and task-specific rubrics to define expected content and assess factuality, comprehensiveness, citation

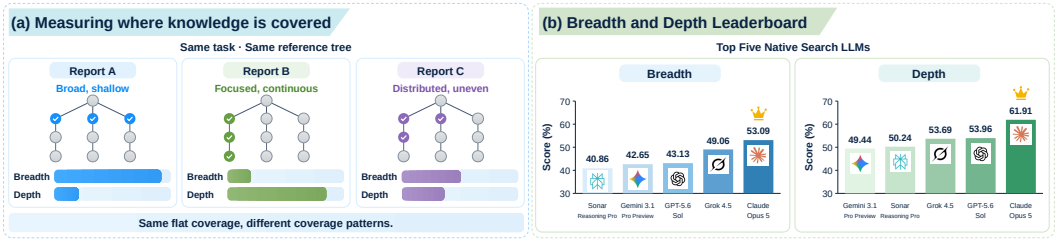



Figure 1: **Structured knowledge coverage in DRBench Pro.** (a) Reports with the same flat coverage can exhibit different coverage patterns over a shared reference tree. (b) Overview of five native search LLMs' performance on DRBench Pro.



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.

Table 1: **Comparison with representative deep research benchmarks.** ✓: fully addressed; ✓̸: partially or indirectly addressed; ✗: absent.

| Comparison Dimension | DRB | DRB-II | ResRub. | WLC | DREval | SuperRes. | SQA→DR | HiEviDR | DR-Arena | Ours |
|---|---|---|---|---|---|---|---|---|---|---|
| Automated Construction | ✗ | ✓̸ | ✗ | ✓̸ | ✓ | ✓̸ | ✓ | ✓̸ | ✓ | ✓ |
| Structured Reference Space | ✗ | ✗ | ✗ | ✗ | ✗ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Structure-Aligned Units | ✗ | ✗ | ✗ | ✗ | ✗ | ✓̸ | ✓ | ✓ | ✓̸ | ✓ |
| Topology-Derived Breadth–Depth | ✗ | ✗ | ✗ | ✗ | ✗ | ✓̸ | ✓̸ | ✓̸ | ✓̸ | ✓ |
| Coverage as Primary Target | ✓̸ | ✓̸ | ✓̸ | ✓̸ | ✓̸ | ✓̸ | ✓̸ | ✓̸ | ✓̸ | ✓ |

quality, and writing (Du et al., 2026; Sharma et al., 2026; Li et al., 2026b). Recent approaches further reduce manual effort by generating evaluation criteria from Wikipedia, research graphs, or structured task representations (Wang et al., 2026c;d; Dong et al., 2026; Wang et al., 2026a), providing explicit criteria for assessing report coverage.

However, existing benchmarks lack a dedicated knowledge-centric evaluation of how broadly and deeply reports cover a structured reference space. Knowledge coverage is typically one component of broader assessments that also consider formatting, organization, and writing style (Du et al., 2026; Wang et al., 2026b). Expert references and manually authored rubrics further limit expansion across domains, while individually specifying or checking detailed criteria makes rubric coverage costly to scale within each task. More fundamentally, even automatically generated criteria lose their relationships under flat aggregation, obscuring whether reports cover more research directions or develop selected directions more deeply. Recent tree- and DAG-based approaches preserve structure for supervision, task completion, evidence traceability, or relative comparison (Zhu et al., 2026; Wang et al., 2026a; Sun et al., 2026; Gao et al., 2026). They do not, however, make structured knowledge coverage itself the primary evaluation target, leaving the breadth and depth of covered knowledge insufficiently characterized. Recent tree- and DAG-based approaches preserve structure, but use it for supervision, task completion, evidence traceability, or relative comparison rather than for scoring the coverage of a final report (Zhu et al., 2026; Wang et al., 2026a; Sun et al., 2026; Gao et al., 2026), leaving the breadth and depth of covered knowledge unmeasured (Table 1)

To fill this gap, we introduce **DRBench Pro**, a knowledge-centric benchmark that treats structured knowledge coverage as the primary evaluation target. An automated pipeline constructs research tasks and structurally aligned evaluation criteria. Starting from a Wikipedia root entity, we build a local entity–relation tree through multi-hop expansion and sample coherent subgraphs that define distinct research scopes. Each sampled subgraph is used both to formulate a research query centered on the root and to generate a hidden rubric set. Every retained relation edge becomes an independent coverage rubric grounded in its corresponding Wikipedia evidence, while preserving its hop distance and parent–child relationships. The evaluated agent receives only the query; the reference tree and rubrics remain hidden. This *shared construction aligns the task scope with its evaluation units* and makes every judgment traceable to an originating relation and structural position. The resulting benchmark contains 144 research cases spanning 24 domains and 48 root entities, with 44,102 evidence-grounded rubrics.

We further propose topology-aware metrics that aggregate edge-level judgments bottom-up over the reference tree. *Breadth* measures how comprehensively a report covers major research directions, allowing descendant coverage to partially compensate for an omitted relation. *Depth* measures how far a report follows chains of related knowledge within selected directions, emphasizing well-developed chains and rewarding continuity between covered relations. Their geometric mean yields *overall coverage*. Evaluating 16 deep research agents reveals substantially interleaved breadth and depth rankings, exposing coverage differences that a single aggregate score can conceal (Figure 1). The strongest evaluated agent, Claude Opus 5, achieves 57.04% overall coverage, followed by Grok 4.5 at 50.98% and o3 Deep Research at 49.45%, leaving substantial room for improvement even among frontier agents. Dedicated deep research systems do not lead: the strongest of them, o3 Deep Research, ranks below two native-search models. Rubric judgments agree with human labels at a Macro-F1 of 0.893, and agent rankings remain stable under perturbations of all metric parameters.

Our main contributions are as follows:

- We introduce DRBench Pro, a knowledge-centric benchmark dedicated to structured knowledge coverage in deep research reports, comprising 144 cases across 24 domains and 44,102 evidence-

grounded rubrics, built by an automated pipeline that jointly generates each research query and its hidden rubrics from the same Wikipedia-derived subgraph while preserving the evidence provenance and structural position of every rubric.

- We propose topology-aware breadth and depth metrics, together with overall coverage, to distinguish coverage across research directions from progression along related knowledge chains, supported by human-consistency and stability analyses.
- We evaluate 16 deep research agents and identify distinct coverage strengths and remaining gaps. We also provide a local research framework with a fixed Wikipedia snapshot, standardized tools, and shared workflow and interaction budgets for controlled evaluation of backbone LLMs.

## 2 Related Work

**Deep research evaluation.** Deep research benchmarks evaluate reports through task-specific criteria covering comprehensiveness, factuality, and citation quality (Du et al., 2026; Sharma et al., 2026; Han et al., 2025; Zhong et al., 2026; Wang et al., 2026b; Avraham et al., 2026). Expected content is drawn from expert reports, Wikipedia, generated criteria, or research graphs, with varying degrees of automation in task and rubric construction (Li et al., 2026b; Wang et al., 2026c;d; Dong et al., 2026). For survey-based reports, reference-literature recall and statement veracity measure source coverage and factual accuracy (Li et al., 2025b). Complementary evaluations examine information gathering through broad exploration, passage-retrieval completeness, and exhaustive collection (Java et al., 2026; Kim et al., 2026; Wong et al., 2026; Gupta et al., 2026).

**Factuality and content coverage.** Factuality and coverage address complementary questions: whether reported claims are supported, and whether relevant information is missing. Claim-level verification and citation audits check generated statements against evidence (Min et al., 2023; Wei et al., 2024; Song et al., 2024b; Gao et al., 2023; Venkit et al., 2025), while joint analysis of retrieval and generation helps locate failures in systems that use retrieved evidence (Ru et al., 2024). Coverage evaluation instead identifies omissions by comparing outputs with reference content units and key facts (Nenkova & Passonneau, 2004; Liu et al., 2023; Song et al., 2024a; Lee et al., 2024), testing whether they answer reference questions (Deutsch et al., 2021; Scialom et al., 2021; Zhang et al., 2024; Tan et al., 2024), or checking for salient aspects and insights across documents (Samarinas et al., 2025; Pradeep et al., 2025; Laban et al., 2024). Automated rubric generation and judgment calibration support the construction and application of these evaluation criteria (Rao & Callison-Burch, 2026; Hashemi et al., 2024).

**Structured construction and evaluation.** Graphs and trees organize multi-hop questions (Welbl et al., 2018; Yang et al., 2018; Ho et al., 2020; Trivedi et al., 2022; Zhu et al., 2024) and synthetic web-search tasks (Li et al., 2025a; Lu et al., 2025; Zhang et al., 2025; Tao et al., 2025; 2026; Li et al., 2026a). For deep research, they connect queries, evidence, and rubrics to support reinforcement learning supervision and propagation of rubric weights (Zhu et al., 2026; Wang et al., 2026a). Structured evidence also enables tracing across research stages and depth and width tests for system comparison (Sun et al., 2026; Gao et al., 2026). Using structure to construct tasks or trace evidence is distinct from retaining it to aggregate report coverage. Our benchmark makes structured knowledge coverage the primary evaluation target: each rubric retains its position in an entity–relation tree during aggregation, yielding separate breadth and depth measures for coverage across research directions and along related knowledge chains. Table 1 summarizes these distinctions.

## 3 Benchmark Construction and Evaluation

Measuring knowledge coverage requires a reference that specifies both what a report could cover and how that knowledge is organized. We introduce **DRBench Pro**, which uses Wikipedia-derived knowledge trees to define research tasks and their evaluation criteria within a shared reference space. The tree structure separates major research directions from chains of related knowledge within them, enabling distinct measures of breadth and depth. Figure 2 illustrates the construction and evaluation pipeline. The prompts used for benchmark construction and evaluation are provided in Appendix J.

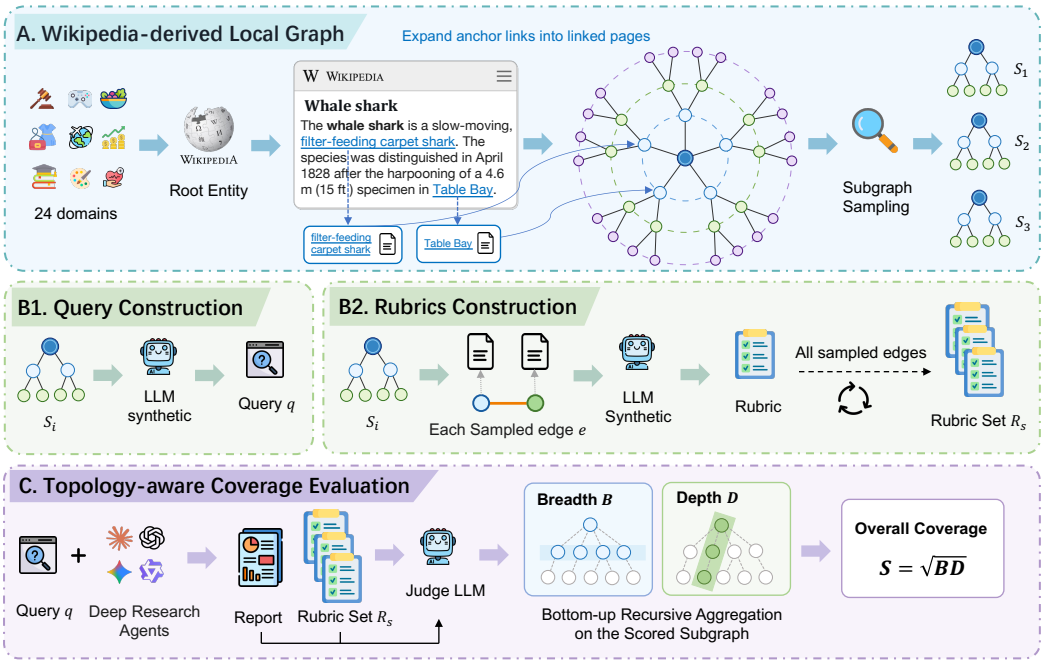



Figure 2: **Overview of DRBench Pro.** **(A)** We construct a Wikipedia-derived local graph around a selected root entity and sample subgraphs by retaining selected children of the root and their complete descendant subtrees together with the root. **(B1)** Each sampled subgraph defines a research query. **(B2)** Each sampled edge is converted into an edge-grounded rubric, forming the hidden rubric set. **(C)** An evaluated deep research agent answers the query with a report; a judge scores the report against the rubric set, and the edge-level scores are mapped back to the local knowledge tree and aggregated bottom-up into breadth and depth.

### 3.1 Wikipedia-derived Local Graph Construction

A reference space for knowledge coverage needs to connect research topics with supporting evidence. Wikipedia provides linked entities whose relationships can be examined in their source context. To span diverse domains, we adopt all 24 topic categories from WebOrganizer (Wettig et al., 2025) and select two root entities per category. Each root has a sufficiently detailed, reliably parseable article and enough outgoing links to support expansion.

Starting from each root, we expand Wikipedia links breadth-first. Nodes represent entities, and edges represent relationships supported by link anchors and surrounding text. We organize the retained expansion as a rooted tree, with one parent per non-root node and a unique path from the root. This structure gives each relation an identifiable position for subsequent rubric construction and coverage aggregation. Distance from the root is measured by the number of links along this path, and expansion stops at three hops.

To provide diverse candidate research directions while bounding the size of deeper expansions, we retain 50 children of the root and at most five children of each other node. Invalid or duplicate candidates are filtered before Claude Opus 4.8 selects neighbors for their relevance to the current entity and usefulness for research tasks. The resulting 48 local knowledge trees contain approximately 1.5K nodes each on average. Appendix A reports detailed statistics, and Appendix C.1 analyzes the choice of two roots per domain. Model-call costs for construction are reported in Appendix H.

### 3.2 Subgraph Sampling

A full knowledge tree spans more material than a single research request can reasonably target. We therefore sample smaller subgraphs, each defining the scope of one research case. A branch consists of one child of the root and all its descendants: it represents a research direction, while

paths within it capture chains of related knowledge. Each case retains the root and 10 complete branches, preserving the knowledge structure within the selected directions.

To make each scope coherent, Claude Opus 4.8 jointly selects three groups of branches from each tree, using the candidate entities' page summaries, relations to the root, and descendant information. Branches within a group are selected to provide complementary perspectives on a common research task; the three groups do not share root branches. Retaining complete branches preserves the deeper relations needed to evaluate coverage within each direction. This yields three cases per root and 144 cases across the 24 domains.

### 3.3 Query and Rubric Construction

The research request and its evaluation criteria should refer to the same knowledge scope. We therefore use Claude Opus 4.8 to generate both the query and the rubrics from each sampled subgraph. For query generation, the model receives the root entity and the sampled knowledge scope, and is instructed to frame a natural research request around their broader themes. The prompt discourages listing hidden entities or restating the subgraph as a checklist, so the agent is asked to investigate a topic rather than reproduce an explicit answer outline.

To make the expected coverage traceable to evidence, we create one rubric per retained relation edge. Each rubric asks whether the report covers the relationship between its source and target entities. The generator uses only the supplied evidence: page summaries, the anchor sentence, relevant paragraphs, and available backlink context from the two pages. This ties each scoring requirement to documented content rather than an unsupported model-generated expectation.

Each rubric retains its edge's position in the reference tree, allowing its judgment to be aggregated with those of related rubrics. We also record its hop distance as the distance from the root to the edge's target node for structural diagnostics. These annotations are kept outside the rubric text. Evaluated agents receive only the query; the reference tree and rubrics remain hidden.

### 3.4 Topology-aware Knowledge Coverage Evaluation

Counting satisfied rubrics alone can give similar scores to reports with different coverage patterns. We therefore retain the reference tree when aggregating judgments, distinguishing coverage across research directions from coverage along related knowledge chains. An LLM judge assigns each hidden rubric one of three verdicts: *Satisfied*, *Partially Satisfied*, or *Not Satisfied*. For coverage metric computation, we map *Satisfied* to 1 (covered) and both *Partially Satisfied* and *Not Satisfied* to 0 (uncovered). These binary values are then aggregated over the reference tree.

**Breadth.** Breadth measures how comprehensively a report covers the selected research directions. We give these directions equal weight so that a branch with more rubrics does not dominate the score. Within a branch, deeper relations can demonstrate coverage of a direction even when a shallower relation is missing. We preserve each relation's direct score as a weighted component and use the mean breadth of its child subtrees to account for descendant coverage. Averaging the resulting branch scores gives case-level breadth.

**Depth.** Depth measures coverage along related knowledge chains in the reference tree. A report can cover one chain deeply without covering every branch equally, so we give greater weight to child subtrees with higher depth scores while retaining smaller contributions from the others. Each relation contributes its own score and regulates how much descendant coverage is credited: an uncovered intermediate relation reduces that credit without discarding it. This rewards continuity along knowledge chains. We apply the same rank-based weighting across the selected research directions to obtain case-level depth. The metric describes knowledge covered in the final report, not the agent's reasoning depth or search trajectory.

**Overall coverage.** We combine case-level breadth $B_c$ and depth $D_c$ through their geometric mean,

$$S_c = \sqrt{B_c D_c}, \tag{1}$$

so weak coverage in either dimension lowers the combined score. Each of the three metrics is then averaged separately across the 144 cases.

Both metrics are computed from the leaves toward the root, starting with the direct rubric scores and recursively combining them with descendant coverage. Appendix B provides the complete procedure in Algorithm 1, together with the equations, parameter definitions, and boundary conventions.

# 4 EVALUATION

## 4.1 EVALUATION SETUP

**Evaluation methods.** We evaluate 16 deep research agents and group them into three categories according to how search and research orchestration are provided.

**(1) Deep Research Systems.** These are end-to-end research agents explicitly designed for autonomous, long-horizon investigation. They natively combine planning, iterative evidence gathering, and long-form synthesis. We evaluate o3 Deep Research, Perplexity Sonar Deep Research, and Qwen Deep Research.

**(2) Native Search LLMs.** These models integrate provider-native web search directly into generation without relying on a separate Deep Research product workflow. We evaluate Grok 4.5, Claude Opus 5, GPT-5.6 Sol, Gemini 3.1 Pro Preview, Sonar Reasoning Pro, and Qwen3.8 Max with native search enabled.

**(3) Backbone LLMs under the local research framework.** As a controlled and reproducible evaluation setting, we implement the local research framework backed by the fixed English Wikipedia dump `enwiki-20260701-v1`. The framework exposes SEARCH and OPEN operations over the same snapshot, disables provider-native web search, and keeps the research workflow and budget fixed across backbone LLMs. Its purpose is to provide a common search service into which different backbone LLMs can be integrated, rather than to improve their benchmark scores. This design removes temporal drift in the retrieval corpus, reduces variation from external search infrastructure, and retains auditable evidence records. We evaluate Gemma 4 31B, DeepSeek V4 Pro, DeepSeek V4 Flash, Qwen3.8 Max, Qwen3.7 Plus, GLM 5.2, and MiniMax M3 as backbone LLMs. The execution protocol and formal configuration of the local research framework are detailed in Appendix F.

**Implementation details.** The evaluated reports were collected during the periods listed in Appendix G. We use Gemini 2.5 Pro as the judge for the formal evaluation. It assigns each rubric one of three verdicts: *Satisfied*, *Partially Satisfied*, or *Not Satisfied*. We retain these original judgments and convert them into binary rubric scores of 0 or 1 for the metrics in Section 3.4. Appendix H reports the recorded model-call costs for report generation and judge evaluation.

## 4.2 EVALUATION RESULTS

**Overall performance.** Table 2 shows that Claude Opus 5 achieves the highest overall coverage (57.04%) and ranks first in both breadth (53.09%) and depth (61.91%), yielding strong performance in both dimensions. Grok 4.5 and o3 Deep Research follow, with relatively stronger breadth and depth, respectively. High overall coverage therefore requires both comprehensive coverage across major directions and progress along related knowledge chains. Appendix C.3 lists selected results from prior benchmarks for context.

**Interleaved breadth–depth profiles.** The agent orderings differ substantially between breadth and depth, revealing distinct research strengths and showing that research workflows can produce markedly different coverage profiles. The clearest example is Qwen3.8 Max: under the local research framework, its breadth and depth scores are 15.54 and 6.28 percentage points higher, respectively, than under its Native Search configuration, with the difference concentrated primarily in breadth. The two configurations of the same-named model therefore exhibit different knowledge-exploration patterns: the local research framework configuration covers major research directions more comprehensively, whereas the depth difference is comparatively limited. A similar contrast appears within the Perplexity product line: Perplexity Sonar Deep Research attains higher depth

Table 2: Main results on DRBench Pro (%). **Bold** and underlined values indicate the best and second-best scores across all agents, respectively, based on unrounded values.

| Deep Research Agent | Overall Coverage | Breadth | Depth |
|---|---|---|---|
| **(1) Deep Research Systems** | | | |
| o3 Deep Research | 49.45 | 45.77 | 54.32 |
| Perplexity Sonar Deep Research | 46.06 | 40.00 | 54.08 |
| Qwen Deep Research | 35.16 | 29.33 | 43.37 |
| **(2) Native Search LLMs** | | | |
| Claude Opus 5 | **57.04** | **53.09** | **61.91** |
| Grok 4.5 | 50.98 | 49.06 | 53.69 |
| GPT-5.6 Sol | 47.80 | 43.13 | 53.96 |
| Gemini 3.1 Pro Preview | 45.50 | 42.65 | 49.44 |
| Sonar Reasoning Pro | 44.76 | 40.86 | 50.24 |
| Qwen3.8 Max | 36.53 | 31.08 | 44.09 |
| **(3) Ours: Backbone LLMs under the local research framework** | | | |
| Qwen3.8 Max | 47.94 | 46.62 | 50.37 |
| MiniMax M3 | 45.34 | 44.34 | 47.55 |
| GLM 5.2 | 45.34 | 44.61 | 47.09 |
| DeepSeek V4 Flash | 42.63 | 41.42 | 45.04 |
| DeepSeek V4 Pro | 39.22 | 36.96 | 43.03 |
| Qwen3.7 Plus | 36.10 | 35.37 | 37.94 |
| Gemma 4 31B | 32.00 | 30.43 | 34.82 |

but slightly lower breadth than Sonar Reasoning Pro. Together, these results show that breadth and depth expose agent specializations and research-strategy differences that overall coverage alone cannot reveal.

Figure 3 presents two representative cases. For Lew Grade, similar overall coverage masks a breadth–depth trade-off. For Temperance, agents can deepen their investigations along different knowledge paths. Appendix E presents the case studies in greater detail.

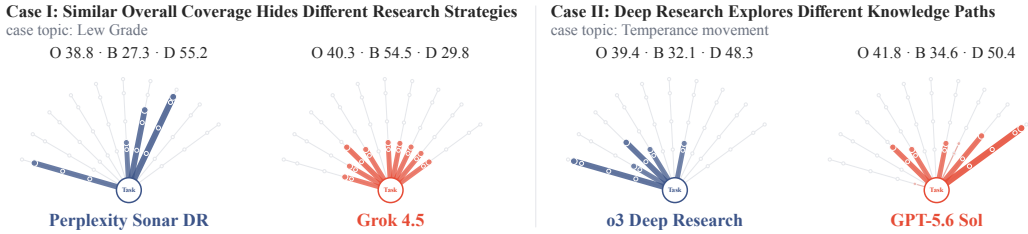



Figure 3: **Case-level coverage profiles.** O/B/D: overall coverage/breadth/depth (%). Gray rays mark the ten shared research directions in each case. Colored ray width and opacity encode branch breadth, while ray length encodes branch depth.

**Coverage under the local research framework.** Under the local research framework, Qwen3.8 Max achieves the highest overall coverage (47.94%). Several backbone LLMs remain competitive under a limited budget. Among the leading backbones, breadth scores are already relatively close, whereas differences in depth further separate their overall coverage. Because these models share the same knowledge source, workflow, tool-use constraints, and budget, the remaining differences reflect backbone-dependent behavior in task decomposition, tool use, evidence selection, and report synthesis. Appendix D.2 analyzes the agents' research trajectories.

### 4.3 Coverage Profiles across Domains

Figure 4 shows overall coverage across 24 domains, with each agent–domain score averaged over six cases. Across agents, coverage is highest in Software Development (53.69%), Science & Technology (52.36%), and Health (50.10%), and lowest in Industrial (28.58%) and Food & Dining (35.45%).

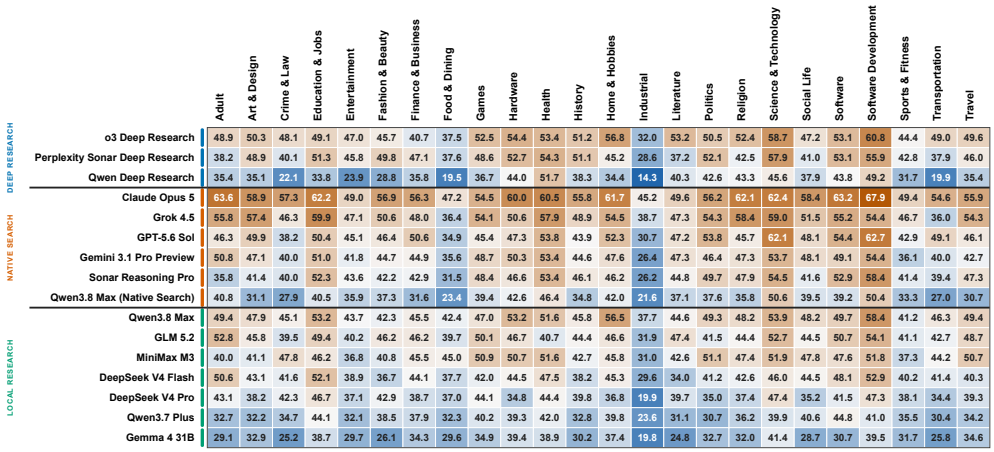



Figure 4: **Overall coverage across domains.** Values are percentages. Orange indicates higher scores, while blue indicates lower scores.

Claude Opus 5 leads in 23 domains, and o3 Deep Research leads in Literature. Appendix D.1 provides the corresponding breadth and depth profiles.

### 4.4 Metric Sensitivity

We test ranking sensitivity by varying aggregation parameter and recomputing each deep research agent's mean breadth and depth over all 144 cases. Table 3 compares the resulting rankings of 16 agents with the default configuration. Exact reversals count agent pairs whose ordering changes; practical reversals additionally require the absolute score difference between the two agents to exceed 0.5 percentage points under both the default configuration and the tested parameter configuration.

Table 3: **Sensitivity of agent rankings to metric hyperparameters.** Spearman correlations compare each ranking with the default; exact and practical reversals are counted over 120 agent pairs. All paired values are breadth/depth ($B/D$), with breadth first.

| Parameter configuration | | | | | Ranking stability | | |
|---|---|---|---|---|---|---|---|
| Variant | $\alpha$ | $\eta$ | $\rho$ | $\beta$ | Spearman $r_s$ | Exact | Practical |
| Default | 0.25 | 0.75 | 0.50 | 0.60 | 1.000/1.000 | 0/0 | 0/0 |
| $\alpha$-Low | **0.10** | 0.75 | 0.50 | 0.60 | 1.000/0.979 | 0/5 | 0/2 |
| $\alpha$-High | **0.40** | 0.75 | 0.50 | 0.60 | 1.000/0.976 | 0/6 | 0/1 |
| $\eta$-Low | 0.25 | **0.50** | 0.50 | 0.60 | 1.000/0.997 | 0/1 | 0/0 |
| $\eta$-High | 0.25 | **0.90** | 0.50 | 0.60 | 1.000/1.000 | 0/0 | 0/0 |
| $\rho$-Low | 0.25 | 0.75 | **0.40** | 0.60 | 1.000/0.994 | 0/2 | 0/0 |
| $\rho$-High | 0.25 | 0.75 | **0.60** | 0.60 | 1.000/0.985 | 0/4 | 0/0 |
| $\beta$-Low | 0.25 | 0.75 | 0.50 | **0.50** | 1.000/1.000 | 0/0 | 0/0 |
| $\beta$-High | 0.25 | 0.75 | 0.50 | **0.75** | 0.997/1.000 | 1/0 | 0/0 |

Rankings remain stable across the tested perturbations: Spearman correlations are at least 0.997 for breadth and 0.976 for depth. Depth is most sensitive to $\alpha$: raising it from 0.25 to 0.40 causes one practical reversal, whereas lowering it to 0.10 causes two. This reflects $\alpha$'s balance between a relation's direct coverage and its descendants' coverage. A large $\alpha$ suppresses descendant evidence during bottom-up propagation; a small one gives little independent credit to a covered non-leaf relation when its descendants are uncovered. Neither extreme matches our intended depth definition. All other perturbations cause no practical reversals, supporting stable agent comparisons within the tested ranges.

### 4.5 Research Task Quality and Leakage

Because each research query is generated from a sampled subgraph, we manually review all 144 queries using three criteria: research value, public answerability, and alignment with their assigned domains (Table 4). All are rated Good on the first two criteria; 141 are Good and three Medium on domain match.

Table 4: **Research-task quality.** The three criteria are Research Value, Public Answerability, and Query–Domain Match, as defined in Appendix C.2.

| Criterion | Poor | Medium | Good |
|---|---|---|---|
| Research value | 0 | 0 | 144 |
| Public answerability | 0 | 0 | 144 |
| Query–domain match | 0 | 3 | 141 |

Table 5: **Lexical leakage by hop distance.**

| Hop distance | Inspected entities | Generic-term matches | Confirmed leaks, $n$ (%) |
|---|---|---|---|
| $h(v) = 1$ | 240 | 3 | 0 (0%) |
| $h(v) = 2$ | 1,200 | 9 | 5 (0.417%) |
| $h(v) = 3$ | 5,945 | 22 | 3 (0.050%) |
| Total | 7,385 | 34 | 8 (0.108%) |

The three Medium cases arise from natural cross-domain overlap, as subgraph sampling produces topic-specific research scopes that are not necessarily confined to a single domain; the queries nevertheless remain coherent, publicly answerable, and sufficiently aligned with their assigned domains for benchmark evaluation.

We also check whether queries reveal the hidden entity information used for evaluation. For each of 24 sampled cases (one per domain), we compare its query with every non-root entity title in its subgraph, yielding 7,385 query–entity comparisons, and manually review literal matches. The 34 generic-term overlaps arise because some subgraph titles, or components of those titles, are ordinary words or phrases that naturally occur in research queries and therefore do not unambiguously identify the corresponding entities. These overlaps are excluded from the leakage count. 8 entity-title mentions are confirmed as leaks, with none at the first hop (Table 5). Therefore, the low leakage rate is unlikely to materially affect subsequent report generation or benchmark evaluation.

### 4.6 Human Consistency with LLM Judges

We compare three LLM judges with human-reviewed labels on 3,695 rubric judgments from 12 reports, each representing a different domain. Gemini 2.5 Pro, Qwen 3.8 Max, and DeepSeek V4 Pro assign *Satisfied*, *Partially Satisfied*, or *Not Satisfied* verdicts. For binary agreement, we map *Satisfied* to 1 (covered) and the other two verdicts to 0 (uncovered) for both human and judge labels. We compute Macro-F1 per report and average it across the 12 reports. Appendix I details the human annotation protocol.

**Binary versus ternary scoring.** In our benchmark, binary scoring yields higher mean Macro-F1 than ternary scoring for all three judges (Table 6). This finding aligns with the trend reported by ResearchRubrics (Sharma et al., 2026). We therefore adopt binary rubric scores for metric computation in our benchmark.

Table 6: **Human–LLM agreement.**

| Judge | Macro-F1 | |
|---|---|---|
| | Binary | Ternary |
| Gemini 2.5 Pro | **0.893** | **0.806** |
| Qwen 3.8 Max | 0.840 | 0.718 |
| DeepSeek V4 Pro | 0.865 | 0.630 |

**Judge-model comparison.** Under binary scoring, Gemini 2.5 Pro has the highest agreement with the human-reviewed labels among the three judges. We therefore retain Gemini 2.5 Pro as the judge and convert its ternary outputs to binary scores for the final evaluation.

## 5 Conclusion

We introduce **DRBench Pro**, a knowledge-centric benchmark for professional deep research agents. From each sampled multi-hop Wikipedia-derived entity subgraph, it jointly generates a research query and traceable edge-grounded rubrics, giving the case, knowledge scope, and evaluation criteria a shared structural foundation. Topology-aware aggregation captures breadth across major research directions and depth along related knowledge chains, with their geometric mean yielding overall coverage. Evaluation across 16 deep research agents reveals interleaved breadth–depth profiles; with the knowledge source, workflow, tool interface, and interaction budget fixed, the local research framework further distinguishes the coverage behavior of different backbone LLMs. Overall, **DRBench Pro** uses graph topology in both benchmark construction and metric formulation, providing a scalable, structure-aware foundation for evaluating knowledge coverage in deep research reports.

### AI USE STATEMENT

We used generative AI tools to assist with manuscript drafting and language editing, code implementation, and the development and refinement of the research methodology and experimental design.

LLMs also serve as components of the benchmark construction and evaluation pipeline: they assist with Wikipedia-derived graph construction and subgraph selection, generate research queries and edge-grounded rubrics, and evaluate report coverage against these rubrics. The evaluated deep research agents generate the reports analyzed in this study. These uses are described in Sections 3 and 4, with prompts provided in Appendix J.

The authors manually reviewed the AI-assisted writing, code, and methodological and experimental design decisions. We take full responsibility for the final manuscript, implementation, and reported findings, including content and artifacts produced with the assistance of generative AI.

### ETHICS STATEMENT

The benchmark is constructed from publicly available Wikipedia content and does not require private personal data. Source content and model-generated reports may contain errors or biases. Coverage scores should therefore not be interpreted as measures of factual accuracy or used alone for high-stakes decisions.

### REPRODUCIBILITY STATEMENT

We will provide the evaluation data and evaluation code used for the reported coverage analysis. The code converts rubric judgments into binary scores and aggregates them into case-level breadth, depth, and overall coverage, followed by system-level averages. Together, these materials will allow readers to recompute the reported scores and verify the comparisons among evaluated agents.

## REFERENCES

Elad Ben Avraham, Changhao Li, Ron Dorfman, Roy Ganz, Oren Nuriel, Amir Dudai, Aviad Aberdam, Noah Flynn, Elman Mansimov, Aditya Kalyanpur, et al. Dream: Deep research evaluation with agentic metrics. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 9879–9904, 2026.

Daniel Deutsch, Tania Bedrax-Weiss, and Dan Roth. Towards question-answering as an automatic metric for evaluating the content quality of a summary. *Transactions of the Association for Computational Linguistics*, 9:774–789, 2021.

Yubo Dong, Nianhao You, Yuxuan Hou, Zixun Sun, Yue Zhang, Liang Zhang, Siyuan Zhao, and Hehe Fan. Super research: Answering highly complex questions with large language models through super deep and super wide research. *arXiv preprint arXiv:2603.00582*, 2026.

Mingxuan Du, Benfeng Xu, Chiwei Zhu, Licheng Zhang, Xiaorui Wang, and Zhendong Mao. Deep-research bench: A comprehensive benchmark for deep research agents. In *International Conference on Learning Representations*, volume 2026, pp. 42414–42448, 2026.

Tianyu Gao, Howard Yen, Jiatong Yu, and Danqi Chen. Enabling large language models to generate text with citations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 6465–6488, 2023.

Yiwen Gao, Ruochen Zhao, Yang Deng, and Wenxuan Zhang. Dr-arena: an automated evaluation framework for deep research agents. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 27130–27152, 2026.

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

Janghoon Han, Heegyu Kim, Changho Lee, Dahm Lee, Min Hyung Park, Hosung Song, Stanley Jungkyu Choi, Moontae Lee, and Honglak Lee. Deer: A benchmark for evaluating deep research agents on expert report generation. *arXiv preprint arXiv:2512.17776*, 2025.

Helia Hashemi, Jason Eisner, Corby Rosset, Benjamin Van Durme, and Chris Kedzie. Llm-rubric: A multidimensional, calibrated approach to automated evaluation of natural language texts. In *Proceedings of the 62nd annual meeting of the association for computational linguistics (volume 1: Long papers)*, pp. 13806–13834, 2024.

Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara, and Akiko Aizawa. Constructing a multi-hop qa dataset for comprehensive evaluation of reasoning steps. In *Proceedings of the 28th International Conference on Computational Linguistics*, pp. 6609–6625, 2020.

Abhinav Java, Ashmit Khandelwal, Sukruta Midigeshi, Aaron Halfaker, Amit Jayant Deshpande, Navin Goyal, Ankur Gupta, Nagarajan Natarajan, and Amit Sharma. Characterizing deep research: A benchmark and formal definition. In *International Conference on Learning Representations*, volume 2026, pp. 10315–10349, 2026.

Remy Kim, Minseung Lee, Shuo Li, and Osbert Bastani. Seekergym: A benchmark for reliable information seeking. *arXiv preprint arXiv:2604.17143*, 2026.

Philippe Laban, Alexander Richard Fabbri, Caiming Xiong, and Chien-Sheng Wu. Summary of a haystack: A challenge to long-context llms and rag systems. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 9885–9903, 2024.

Yuho Lee, Taewon Yun, Jason Cai, Hang Su, and Hwanjun Song. Unisumeval: Towards unified, fine-grained, multi-dimensional summarization evaluation for llms. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 3941–3960, 2024.

Kuan Li, Zhongwang Zhang, Huifeng Yin, Liwen Zhang, Litu Ou, Jialong Wu, Wenbiao Yin, Baixuan Li, Zhengwei Tao, Xinyu Wang, et al. Websailor: Navigating super-human reasoning for web agent. *arXiv preprint arXiv:2507.02592*, 2025a.

Kuan Li, Zhongwang Zhang, Huifeng Yin, Rui Ye, Yida Zhao, Liwen Zhang, Litu Ou, Ding-Chu Zhang, Xixi Wu, Xinmiao Yu, et al. Websailor-v2: Bridging the chasm to proprietary agents via synthetic data and scalable reinforcement learning. In *International Conference on Learning Representations*, volume 2026, pp. 93919–93948, 2026a.

Minghao Li, Ying Zeng, Zhihao Cheng, Cong Ma, and Kai Jia. Reportbench: Evaluating deep research agents via academic survey tasks. *arXiv preprint arXiv:2508.15804*, 2025b.

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. *arXiv preprint arXiv:2601.08536*, 2026b.

Yixin Liu, Alex Fabbri, Pengfei Liu, Yilun Zhao, Linyong Nan, Ruilin Han, Simeng Han, Shafiq Joty, Chien-Sheng Wu, Caiming Xiong, et al. Revisiting the gold standard: Grounding summarization evaluation with robust human evaluation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 4140–4170, 2023.

Rui Lu, Zhenyu Hou, Zihan Wang, Hanchen Zhang, Xiao Liu, Yujiang Li, Shi Feng, Jie Tang, and Yuxiao Dong. Deepdive: Advancing deep search agents with knowledge graphs and multi-turn rl. *arXiv preprint arXiv:2509.10446*, 2025.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. Factscore: Fine-grained atomic evaluation of factual precision in long form text generation. In *Proceedings of the 2023 conference on empirical methods in natural language processing*, pp. 12076–12100, 2023.

Ani Nenkova and Rebecca J Passonneau. Evaluating content selection in summarization: The pyramid method. In *Proceedings of the human language technology conference of the north american chapter of the association for computational linguistics: Hlt-naacl 2004*, pp. 145–152, 2004.

Ronak Pradeep, Nandan Thakur, Shivani Upadhyay, Daniel Campos, Nick Craswell, Ian Soboroff, Hoa Trang Dang, and Jimmy Lin. The great nugget recall: Automating fact extraction and rag evaluation with large language models. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 180–190, 2025.

Delip Rao and Chris Callison-Burch. Autorubric: A unifying framework for rubric-based llm evaluation on non-verifiable tasks. In *Third Conference on Language Modeling*, 2026.

Dongyu Ru, Lin Qiu, Xiangkun Hu, Tianhang Zhang, Peng Shi, Shuaichen Chang, Cheng Jiayang, Cunxiang Wang, Shichao Sun, Huanyu Li, et al. Ragchecker: A fine-grained framework for diagnosing retrieval-augmented generation. *Advances in Neural Information Processing Systems*, 37:21999–22027, 2024.

Chris Samarinas, Alexander Krubner, Alireza Salemi, Youngwoo Kim, and Hamed Zamani. Beyond factual accuracy: Evaluating coverage of diverse factual information in long-form text generation. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 13468–13482, 2025.

Thomas Scialom, Paul-Alexis Dray, Sylvain Lamprier, Benjamin Piwowarski, Jacopo Staiano, Alex Wang, and Patrick Gallinari. Questeval: Summarization asks for fact-based evaluation. In *Proceedings of the 2021 conference on empirical methods in natural language processing*, pp. 6594–6604, 2021.

Manasi Sharma, Chen Bo Calvin Zhang, Chaithanya Bandi, Clinton Wang, Ankit Aich, Huy Nghiem, Tahseen Rabbani, Ye Htet, Brian Jang, Sumana Basu, et al. Researchrubrics: A benchmark of prompts and rubrics for evaluating deep research agents. In *International Conference on Learning Representations*, volume 2026, pp. 90447–90472, 2026.

Hwanjun Song, Hang Su, Igor Shalyminov, Jason Cai, and Saab Mansour. Finesure: Fine-grained summarization evaluation using llms. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 906–922, 2024a.

Yixiao Song, Yekyung Kim, and Mohit Iyyer. Veriscore: Evaluating the factuality of verifiable claims in long-form text generation. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 9447–9474, 2024b.

Yubo Sun, Chunyi Peng, Yukun Yan, Zhenghao Liu, Sen Mei, Bangrui Xu, Xuanhe Zhou, Chi Chen, and Maosong Sun. Hievidr-bench: A benchmark for hierarchical evidence aggregation in deep research. *arXiv preprint arXiv:2607.25151*, 2026.

Haochen Tan, Zhijiang Guo, Zhan Shi, Lu Xu, Zhili Liu, Yunlong Feng, Xiaoguang Li, Yasheng Wang, Lifeng Shang, Qun Liu, et al. Proxyqa: An alternative framework for evaluating long-form text generation with large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 6806–6827, 2024.

Zhengwei Tao, Haiyang Shen, Baixuan Li, Wenbiao Yin, Jialong Wu, Kuan Li, Zhongwang Zhang, Huifeng Yin, Rui Ye, Liwen Zhang, et al. Webleaper: Empowering efficiency and efficacy in webagent via enabling info-rich seeking. *arXiv preprint arXiv:2510.24697*, 2025.

Zhengwei Tao, Jialong Wu, Wenbiao Yin, Pu Wu, Junkai Zhang, Baixuan Li, Haiyang Shen, Kuan Li, Liwen Zhang, Xinyu Wang, et al. Webshaper: Agentically data synthesizing via information-seeking formalization. In *International Conference on Learning Representations*, volume 2026, pp. 101872–101889, 2026.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. ♫ musique: Multi-hop questions via single-hop question composition. *Transactions of the Association for Computational Linguistics*, 10:539–554, 2022.

Pranav Narayanan Venkit, Philippe Laban, Yilun Zhou, Kung-Hsiang Huang, Yixin Mao, and Chien-Sheng Wu. Deeptrace: Auditing deep research ai systems for tracking reliability across citations and evidence. *arXiv preprint arXiv:2509.04499*, 2025.

Can Wang, Haoran Chen, Haowen Gao, Hao Ding, Zhaoyang Liu, and Zhiying Tu. From simple qa to deep research: A verifiable benchmark constructed through iterative task evolution. *arXiv preprint arXiv:2608.02163*, 2026a.

Jiayu Wang, Yifei Ming, Riya Dulepet, Qinglin Chen, Austin Xu, Zixuan Ke, Frederic Sala, Aws Albarghouthi, Caiming Xiong, and Shafiq Joty. Liveresearchbench: A live benchmark for user-centric deep research in the wild. In *International Conference on Learning Representations*, volume 2026, pp. 95096–95143, 2026b.

Shaohan Wang, Benfeng Xu, Licheng Zhang, Mingxuan Du, Chiwei Zhu, Xiaorui Wang, Zhendong Mao, and Yongdong Zhang. Wiki live challenge: Challenging deep research agents with expert-level wikipedia articles. *arXiv preprint arXiv:2602.01590*, 2026c.

Yibo Wang, Lei Wang, Yue Deng, Keming Wu, Yao Xiao, Huanjin Yao, Liwei Kang, Hai Ye, Yongcheng Jing, and Lidong Bing. Deepresearcheval: An automated framework for deep research task construction and agentic evaluation. *arXiv preprint arXiv:2601.09688*, 2026d.

Jerry Wei, Chengrun Yang, Xinying Song, Yifeng Lu, Nathan Hu, Jie Huang, Dustin Tran, Daiyi Peng, Ruibo Liu, Da Huang, et al. Long-form factuality in large language models. *Advances in Neural Information Processing Systems*, 37:80756–80827, 2024.

Johannes Welbl, Pontus Stenetorp, and Sebastian Riedel. Constructing datasets for multi-hop reading comprehension across documents. *Transactions of the Association for Computational Linguistics*, 6:287–302, 2018.

Alexander Wettig, Kyle Lo, Sewon Min, Hannaneh Hajishirzi, Danqi Chen, and Luca Soldaini. Organize the web: Constructing domains enhances pre-training data curation. *arXiv preprint arXiv:2502.10341*, 2025.

Ryan Wong, Jiawei Wang, Li Chen, Yan Gao, Xuan Zhou, Zuo Wang, Kai Xiang, Ge Zhang, Wenhao Huang, Yang Wang, et al. Widesearch: Benchmarking agentic broad info-seeking. In *International Conference on Learning Representations*, volume 2026, pp. 10012–10086, 2026.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D Manning. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, pp. 2369–2380, 2018.

Gongrui Zhang, Jialiang Zhu, Ruiqi Yang, Kai Qiu, Miaosen Zhang, Zhirong Wu, Qi Dai, Bei Liu, Chong Luo, Zhengyuan Yang, et al. Infoagent: Advancing autonomous information-seeking agents. *arXiv preprint arXiv:2509.25189*, 2025.

Shiyue Zhang, David Wan, Arie Cattan, Ayal Klein, Ido Dagan, and Mohit Bansal. Qapyramid: Fine-grained evaluation of content selection for text summarization. *arXiv preprint arXiv:2412.07096*, 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. *arXiv preprint arXiv:2602.11685*, 2026.

Andrew Zhu, Alyssa Hwang, Liam Dugan, and Chris Callison-Burch. Fanoutqa: A multi-hop, multi-document question answering benchmark for large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 18–37, 2024.

Minghang Zhu, Chuyang Wei, Junhao Xu, Yilin Cheng, Zhumin Chen, and Jiyan He. Deeprubric: Evidence-tree rubric supervision for efficient reinforcement learning of deep research agents. *arXiv preprint arXiv:2606.17029*, 2026.

# A BENCHMARK STATISTICS

For the structural statistics below, $h(v)$ denotes the number of edges from the root to node $v$. For a rubric $r_e$ associated with an edge $e = (u, v)$, its hop distance is $\delta(r_e) = h(v)$.

Our DRBench Pro benchmark is domain-balanced and contains 24 topical domains, 48 root entities, and 144 research cases. Each domain contributes two root entities, and each root yields three case subgraphs, giving six cases per domain. As summarized in Table 7, each of the three case subgraphs sampled from a root entity selects 10 of that root's 50 children and retains their complete branches. The three sets of selected children are mutually disjoint, maintaining distinct case scopes.

Table 7: **DRBench Pro statistics.** Computed over 48 local graphs and 144 sampled subgraphs.

**Panel A. Benchmark composition**

| Statistic | Value |
|---|---|
| Domains | 24 |
| Root entities | 48 |
| Cases | 144 |
| Cases per domain | 6 |
| Cases per root | 3 |
| Total rubrics | 44,102 |

**Panel B. Graph and case structure**

| Graph scope | Statistic | Mean / value | Observed range |
|---|---|---|---|
| Complete local graph | Edges | 1,528.6 | 1,365–1,550 |
| Sampled case subgraph | Edges/rubrics | 306.3 | 236–310 |
| Sampled case subgraph | Root children | 10 | fixed |
| Sampled case subgraph | Maximum hop distance | 3 | fixed |

**Panel C. Rubric distribution by hop distance**

| Rubric hop distance $\delta(r_e)$ | Rubrics | Raw share | Mean/case |
|---|---|---|---|
| $\delta = 1$: root $\rightarrow$ first hop | 1,440 | 3.27% | 10.0 |
| $\delta = 2$: first $\rightarrow$ second hop | 7,184 | 16.29% | 49.9 |
| $\delta = 3$: second $\rightarrow$ third hop | 35,478 | 80.45% | 246.4 |

Rubric counts are uneven across hop distances, with most rubrics occurring at greater hop distances. Our topology-aware scheme uses normalized aggregation of child-subtree values at each recursive step. Rubric contributions therefore depend on coverage, positions in the tree, and the aggregation rules; larger rubric counts at greater hop distances alone do not imply greater aggregate weight.

# B METRIC DETAILS

This section provides the complete coverage computation for Section 3.4. Algorithm 1 summarizes the procedure; the following subsections define its notation, recursive equations, and parameters.

## B.1 NOTATION AND RECURSIVE EVALUATION

For a case $c$, let $T = (V, E)$ be its reference tree, with root $s$ and child sets $\mathrm{Ch}(v)$. The children of the root, $\mathcal{A} = \mathrm{Ch}(s)$, represent the selected research directions. Each non-root node $v$ has a binary incoming-edge rubric score $x(v) \in \{0, 1\}$: 1 (covered) for a *Satisfied* judge verdict and 0 (uncovered) for a *Partially Satisfied* or *Not Satisfied* verdict. The root has no incoming-edge rubric.

We associate two recursive values with each non-root node: $B(v)$ for breadth and $D(v)$ for depth. Both recursions begin at every actual leaf, including those at shallower levels:

$$B(v) = D(v) = x(v), \qquad \mathrm{Ch}(v) = \emptyset. \tag{2}$$

Table 8: **Root entities in DRBench Pro benchmark.** The table lists the two selected root entities from each of the 24 domains, for a total of 48 root entities.

| Domain | Root entity 1 | Root entity 2 |
|---|---|---|
| Adult | Fanny Hill | American burlesque |
| Art & Design | Black Mountain College | Constructivism (art) |
| Crime & Law | Guildford Four and Maguire Seven | Lindbergh kidnapping |
| Education & Jobs | Highlander Research and Education Center | Educational Testing Service |
| Entertainment | The Second City | Lew Grade |
| Fashion & Beauty | Beauty pageant | Mod (subculture) |
| Finance & Business | John Law (economist) | Grameen Bank |
| Food & Dining | Horn & Hardart | Culinary Institute of America |
| Games | Roguelike | Games Workshop |
| Hardware | Xerox Alto | Cray-1 |
| Health | Ignaz Semmelweis | John Snow |
| History | Taiping Rebellion | Bandung Conference |
| Home & Hobbies | Royal Horticultural Society | Amateur radio |
| Industrial | Birmingham Small Arms Company | Saltaire |
| Literature | Hugo Award | New Journalism |
| Politics | Non-Aligned Movement | Treaty of Waitangi |
| Religion | Albigensian Crusade | Guru Granth Sahib |
| Science & Technology | International Geophysical Year | Cavendish Laboratory |
| Social Life | Mardi Gras in New Orleans | Temperance movement |
| Software | Apache Hadoop | OpenVMS |
| Software Development | Agile software development | GNU Compiler Collection |
| Sports & Fitness | Kabaddi | Bandy |
| Transportation | Cunard Line | Romney, Hythe and Dymchurch Railway |
| Travel | Periplus of the Erythraean Sea | Via Francigena |

**Algorithm 1** Bottom-up recursive evaluation of structured knowledge coverage

**Input:** Rooted tree $T$ with root $s$ and child sets $\mathrm{Ch}(v)$; binary incoming-edge scores $x(v) \in \{0,1\}$ for $v \neq s$
**Input:** Parameters $\beta = 0.60$, $\alpha = 0.25$, $\eta = 0.75$, $\rho = 0.50$
**Output:** Case-level breadth $B_c$, depth $D_c$, and overall coverage $S_c$

**Main procedure**

$\mathcal{A} \leftarrow \mathrm{Ch}(s)$ *Selected research directions*
**for** each $b \in \mathcal{A}$ **do** $(B(b), D(b)) \leftarrow \mathrm{Evaluate}(b)$
$B_c \leftarrow \frac{1}{|\mathcal{A}|} \sum_{b \in \mathcal{A}} B(b)$; $\quad D_c \leftarrow \mathrm{RankAgg}\big((D(b))_{b \in \mathcal{A}}, \rho\big)$
$S_c \leftarrow \sqrt{B_c D_c}$; $\quad$ **return** $(B_c, D_c, S_c)$

**Recursive evaluation**

**function** $\mathrm{Evaluate}(v)$, $v \neq s$:
  $C \leftarrow \mathrm{Ch}(v)$
  **if** $C = \emptyset$, **return** $(x(v), x(v))$ *Leaf initialization*
  **for** each $w \in C$ **do** $(B(w), D(w)) \leftarrow \mathrm{Evaluate}(w)$
  $B(v) \leftarrow \beta x(v) + (1-\beta)\, \frac{1}{|C|} \sum_{w \in C} B(w)$ *Local-descendant coverage fusion*
  $d \leftarrow \mathrm{RankAgg}\big((D(w))_{w \in C}, \rho\big)$
  $D(v) \leftarrow \alpha x(v) + (1-\alpha) d\big[\eta + (1-\eta) x(v)\big]$ *Chain continuity reward*
  **return** $(B(v), D(v))$
**end function**

*Helper function*

**function** $\mathrm{RankAgg}(Y, \rho)$: *Rank-decayed aggregation*
  **if** $Y$ is empty, **return** $0$
  Sort $Y$ as $y_{(1)} \geq \cdots \geq y_{(n)}$, where $n = |Y|$
  **return** $\big(\sum_{i=1}^{n} \rho^{i-1} y_{(i)}\big) \,/\, \big(\sum_{i=1}^{n} \rho^{i-1}\big)$
**end function**

Algorithm 1 invokes $\mathrm{Evaluate}$ on each child of the root. Each call recursively evaluates the child subtrees before computing and returning the current node's breadth and depth values, so scores are aggregated from children to parents following a postorder traversal.

### B.2 BREADTH: LOCAL-DESCENDANT COVERAGE FUSION

Breadth measures how comprehensively a report covers the selected research directions. A covered relation provides local evidence but does not alone establish coverage of its descendant subtopics. We therefore compute breadth as a weighted average of the current relation's coverage and the mean breadth of its child subtrees, allowing coverage of descendant subtopics to contribute recursively to the current direction's breadth.

For a non-leaf, non-root node $v$, let $m_B(v)$ be the mean of $B(w)$ over $w \in \mathrm{Ch}(v)$. We define

$$B(v) = \beta x(v) + (1-\beta) m_B(v). \tag{3}$$

Here, $0 < \beta < 1$ weights the current relation's coverage, while $1-\beta$ weights the mean breadth of its child subtrees. Covering a non-leaf relation alone therefore does not yield full breadth credit for its subtree. Conversely, covered descendants can still contribute when the current relation is uncovered, partially compensating for the current relation's lack of coverage. Their contribution propagates upward through the recursion to the branch-level breadth score.

Case-level breadth is

$$B_c = \frac{1}{|\mathcal{A}|} \sum_{b \in \mathcal{A}} B(b). \tag{4}$$

This gives equal weight to the selected directions while retaining the descendant coverage within each branch.

### B.3 DEPTH: KNOWLEDGE-CHAIN AGGREGATION AND CONTINUITY

Depth measures how far a report follows chains of related knowledge within selected research directions. Deeper coverage along one knowledge chain can demonstrate substantial depth; coverage along additional chains provides further evidence but should have diminishing influence. We capture this objective through rank-decayed aggregation. For child-subtree values sorted as $d_{(1)} \geq \cdots \geq d_{(n)}$, define

$$\mathrm{RankAgg}_\rho(d_1, \ldots, d_n) = \frac{\sum_{i=1}^{n} \rho^{i-1} d_{(i)}}{\sum_{i=1}^{n} \rho^{i-1}}. \tag{5}$$

For $0 < \rho < 1$, higher-valued subtrees receive greater weight, while the others retain a contribution.

Depth should also reward continuous coverage along a knowledge chain, where successive parent–child relations are covered. Descendant evidence deserves greater credit when the report also covers the relation linking the current node to its parent, preserving coverage continuity at that link. When this relation is uncovered, it creates a gap in the chain's coverage; evidence farther along the chain should receive reduced credit without being discarded entirely. The current relation must also contribute independently. For a non-leaf node, let $d(v)$ be the rank-decayed aggregation of its children's depth values. We combine these requirements as

$$D(v) = \alpha x(v) + (1-\alpha) d(v)[\eta + (1-\eta) x(v)]. \tag{6}$$

The term $\alpha x(v)$ credits the current relation directly. The factor $\eta + (1-\eta)x(v)$ rewards continuity: with binary scores, it equals one when $x(v) = 1$ and $\eta$ when $x(v) = 0$. This favors continuously covered knowledge chains while preserving limited credit for descendant evidence beyond a coverage gap.

After recursive evaluation, case-level depth applies the same operator across the selected directions:

$$D_c = \mathrm{RankAgg}_\rho\big((D(b))_{b \in \mathcal{A}}\big). \tag{7}$$

Breadth therefore weights directions equally, whereas depth emphasizes those with deeper coverage along related knowledge chains.

### B.4 METRIC PARAMETERS AND FINAL AGGREGATION

Table 9 lists the fixed defaults used across cases and agents; Section 4.4 examines ranking sensitivity to these choices.

Table 9: **Coverage metric parameters.**

| Parameter | Default | Role |
|---|---|---|
| $\beta$ | 0.60 | Weight assigned to the current relation's coverage in the breadth recursion at non-leaf nodes; $1-\beta$ weights the mean breadth of its child subtrees. |
| $\alpha$ | 0.25 | Weight of the direct current-edge contribution in depth at non-leaf nodes. |
| $\eta$ | 0.75 | Retention factor applied to descendant depth when the current edge is uncovered. |
| $\rho$ | 0.50 | Ratio between successive rank weights in depth aggregation. |

We treat breadth and depth as equally important and compute case-level overall coverage as $S_c = \sqrt{B_c D_c}$, making weak coverage in either dimension consequential. For each deep research agent, we report the mean case-level breadth, depth, and overall coverage across all 144 benchmark cases.

# C Benchmark Validation

## C.1 Benchmark Size Selection

In fact, we initially constructed 720 research cases, covering 10 root entities in each of 24 domains and three cases per root. We then evaluated o3 Deep Research, Perplexity Sonar Deep Research, and Qwen Deep Research on all 720 cases. To determine whether a smaller benchmark could maintain stable results, we compare different numbers of sampled roots per domain.

Table 10: **Sample-size stability of the overall score.** MAE and mean SD are in percentage points.

| $k$ | Cases | MAE ↓ | SD ↓ | Within $\pm 2$ pp ↑ |
|---|---|---|---|---|
| 1 | 72 | 1.481 | 1.610 | 75.0% |
| **2** | **144** | **0.768** | **0.926** | **96.7%** |
| 3 | 216 | 0.570 | 0.740 | 98.3% |
| 4 | 288 | 0.458 | 0.533 | 100.0% |
| 5 | 360 | 0.350 | 0.448 | 100.0% |

We use $k$ to denote the number of roots sampled from each domain. One sample is constructed by selecting $k$ roots from each of the 24 domains and retaining all three cases under every selected root. For each value of $k$, we independently construct 20 such samples. We then compare the three systems' overall scores on these samples with their results on all 720 cases.

The overall score for each case is the geometric mean of its breadth and depth scores, and a system's overall score is averaged across the selected cases. For each $k$, the three systems and 20 samples produce 60 scores. MAE measures their average absolute difference from the corresponding 720-case scores. Mean SD averages the standard deviations of the 20 scores for each system. Within $\pm 2$ pp reports the proportion of the 60 scores that differ from their corresponding 720-case scores by no more than two percentage points.

At $k = 2$, MAE decreases from 1.481 to 0.768 percentage points, 96.7% of the sampled scores remain within two percentage points of the complete-pool results, and all 20 samples preserve the system ordering. Larger samples provide only modest further improvements. We therefore select two roots per domain and use the resulting 144-case benchmark.

## C.2 Research Query Quality Criteria

**Research Value.** This criterion asks whether the query warrants a research report. A strong query calls for substantive investigation, retrieval, and synthesis across relevant evidence; a request answerable through a short factual lookup has limited research value.

**Public Answerability.** This criterion asks whether reliable, publicly available sources can, in principle, support an answer to the query. The answer need not appear in a single source, but the requested findings must be verifiable without private information or unsupported speculation.

**Query–Domain Match.** This criterion asks whether the query's central topic and requested analysis fit its assigned domain. The domain is shown alongside the query during review. Cross-domain connections are compatible with a match when the assigned domain remains central; a merely incidental connection is insufficient.

### C.3 Additional Results from Prior Benchmarks

Table 11 compiles selected results from prior benchmarks, focusing on model families represented in Table 2 and evaluation dimensions with definitions similar to ours. These results serve as an external reference for our evaluation, with each row labeled by its operating setting in the source study. Setting 1 (Deep Research product) denotes an end-to-end research product or system that integrates research planning, iterative retrieval, and report generation. Setting 2 (native search / search tools) denotes a model using native search or search tools during generation. Setting 3 (external framework + backbone) denotes a backbone model embedded in an external retrieval or research framework that organizes tool calls and the research workflow. These labels describe the source experiments; the same model can use different settings across studies. In HiEviDR (Sun et al., 2026), $3_{\mathrm{R}}$ and $3_{\mathrm{D}}$ denote the RAG and Deep Research workflows, respectively. For ResearchRubrics (Sharma et al., 2026), we report binary compliance on tasks with high conceptual breadth or deep logical nesting; DR-Arena's Depth/Width values describe the composition of failure types, rather than standalone ability scores (Gao et al., 2026).

Cross-benchmark comparisons require accounting for differences in experimental settings, model versions, and metric definitions. Studies may use different retrieval resources, tool interfaces, and execution budgets; evaluations conducted at different times often cover different versions within the same model family; and even similarly named metrics may differ in their reference content, scoring criteria, and aggregation rules. We therefore compare relative performance trends among related models and interpret agreement and variation in light of these conditions.

For model pairs with relatively clear correspondences, our results agree with several prior studies. In Table 2, o3 Deep Research achieves higher overall coverage than Perplexity Sonar Deep Research. The corresponding OpenAI–Perplexity product-family ordering also appears in the selected dimensions of DeepResearch Bench (Du et al., 2026), DeepResearch Bench II (Li et al., 2026b), ResearchRubrics, Wiki Live Challenge's Broad metric (Wang et al., 2026c), DeepResearchEval (Wang et al., 2026d), and LiveResearchBench (Wang et al., 2026b). The advantage of o3 Deep Research over Qwen Deep Research is also consistent with the corresponding product-family results in DeepResearch Bench II, Wiki Live Challenge, and DeepResearchEval. Among native-search models, LiveResearchBench orders GPT-5 above Gemini 2.5 Pro and Sonar Reasoning Pro, matching the overall-coverage order of the corresponding model families in Table 2. Although versions and protocols differ, these recurring relative advantages provide useful external context for our results.

Despite these recurring trends, some metrics and model pairs exhibit different relative orderings, including Wiki Live Challenge's Cov. Wiki, Super Research's Coverage (Dong et al., 2026), and the GLM-5.2–DeepSeek-V4-Pro comparison in From Simple QA to Deep Research (Wang et al., 2026a). Taken together, these comparisons show that our evaluation recovers several relative performance trends reported in prior benchmarks, providing complementary evidence for its validity.

## D Detailed Results and Analysis

### D.1 Breadth and depth across Domains

Figures 5 and 6 complement the overall coverage results in Section 4.3 and Figure 4 by showing breadth and depth separately across the 24 domains.

Table 11: **Selected results from prior benchmarks.** Slash-separated scores follow the listed metric order; score scales are specific to each benchmark and metric.

**Source setting:** 1 = Deep Research product; 2 = native search / search tools; 3 = external framework + backbone. For HiEviDR, $3_R$ = RAG and $3_D$ = Deep Research.

| Setting / Model or system | Score(s) |
|---|---|
| **(a) DeepResearch Bench** — Table 1 | |
| *RACE Comp. ↑ / RACE Depth ↑* | |
| 1 OpenAI DR | 46.87 / 45.25 |
| 1 Perplexity DR | 40.69 / 39.39 |
| 1 Grok Deeper Search | 37.97 / 35.37 |
| 1 Gemini-2.5-Pro DR | 48.53 / 48.50 |
| 2 Perplexity-Sonar-Reasoning-Pro(high) | 37.38 / 36.11 |
| 2 Gemini-2.5-Pro-Grounding | 34.06 / 29.79 |
| **(b) DeepResearch Bench II** — Table 3 | |
| *InfoRecall ↑* | |
| 1 OpenAI-GPT-o3 DR | 39.98 |
| 1 Gemini-3-Pro DR | 39.09 |
| 1 Gemini-2.5-Pro DR | 34.91 |
| 1 Qwen3-Max DR | 34.18 |
| 1 Grok Deep Search | 33.52 |
| 1 Perplexity Research | 33.05 |
| **(c) ResearchRubrics** — Figure 12 | |
| *High conceptual breadth ↑ / Deep logical nesting ↑* | |
| 1 Gemini DR | 56.5% / 59.6% |
| 1 ChatGPT DR | 53.1% / 57.5% |
| 1 Perplexity DR | 50.8% / 47.5% |
| **(d) Wiki Live Challenge** — Table 1 | |
| *Broad ↑ / Cov. Wiki ↑* | |
| 1 OpenAI o3 DR | 45.75 / 25.12 |
| 1 Perplexity DR | 37.63 / 29.21 |
| 1 Qwen-3-max DR | 40.00 / 22.22 |
| 1 Grok Deep Search | 35.75 / 20.73 |
| 1 Gemini-2.5-pro DR | 59.88 / 30.76 |
| 1 Gemini-3-pro DR | 67.12 / 28.83 |
| 3 Langchain (GPT-5) | 59.88 / 20.96 |
| **(e) DeepResearchEval** — Table 3 | |
| *Covera. ↑* | |
| 1 OpenAI DR | 8.6 |
| 1 Perplexity DR | 8.2 |
| 1 Grok4 DR | 8.5 |
| 1 Qwen-3-235B-A22B DR | 8.0 |
| 1 Gemini-2.5-Pro DR | 9.2 |

| Setting / Model or system | Score(s) |
|---|---|
| **(f) Super Research** — Table 1 | |
| *Coverage ↑* | |
| 1 o3 DR | 23.44 |
| 1 Sonar DR | 28.26 |
| 1 Gemini DR | 33.15 |
| 2 Grok-4-1-fast | 30.09 |
| 3 Deepseek-r1 | 24.21 |
| 3 Minimax-m2.1 | 20.91 |
| **(g) From Simple QA to Deep Research** — Table 4 | |
| *Agent $q_h$ ↑ / $q_o$ ↑ / $q_a$ ↑* | |
| 3 GPT-5.6 Terra | 0.86 / 0.67 / 0.70 |
| 3 GLM-5.2 | 0.80 / 0.68 / 0.69 |
| 3 DeepSeek-V4-Pro | 0.83 / 0.68 / 0.73 |
| 3 Qwen3.7-Max | 0.83 / 0.67 / 0.71 |
| 3 MiniMax-M2.5 | 0.67 / 0.42 / 0.55 |
| **(h) HiEviDR** — Table 3 | |
| *Wiki Report ↑ / Arxiv Report ↑* | |
| $3_R$ Gemma4-31B-it | 18.28 / 19.26 |
| $3_R$ GPT-5.6-sol | 18.88 / 19.69 |
| $3_R$ Grok-4.5 | 18.66 / 19.43 |
| $3_D$ Gemma4-31B-it | 19.04 / 19.58 |
| $3_D$ GPT-5.6-sol | 19.33 / 19.73 |
| $3_D$ Grok-4.5 | 19.00 / 18.90 |
| **(i) DR-Arena** — Table 3 / Figure 6 | |
| *DR-Arena Elo ↑ / Depth-type share / Width-type share* | |
| 2 GPT-5.1-Search | 1084 / 27% / 26% |
| 2 Gemini-2.5-Pro-Grounding | 1054 / 33% / 24% |
| 2 o3-Search | 1041 / 19% / 30% |
| 2 Grok-4-Search | 958 / 30% / 25% |
| 2 Claude-Opus-4.1-Search | 921 / 31% / 25% |
| 2 Perplexity-Sonar-Pro-High | 942 / 37% / 26% |
| **(j) LiveResearchBench** — Table 1 | |
| *Coverage & Comprehensiveness ↑* | |
| 1 OpenAI o3 DR | 85.0 |
| 1 Perplexity Sonar DR | 65.5 |
| 1 Grok-4 DR | 86.3 |
| 1 Gemini DR | 75.8 |
| 2 GPT-5 | 83.4 |
| 2 Gemini 2.5 Pro | 73.1 |
| 2 Claude 4.1 Opus | 50.8 |
| 2 Perplexity Sonar Reasoning Pro | 46.7 |
| 3 Deerflow+ (w. GPT-5) | 61.6 |
| 3 Open DR (w. GPT-5) | 65.3 |

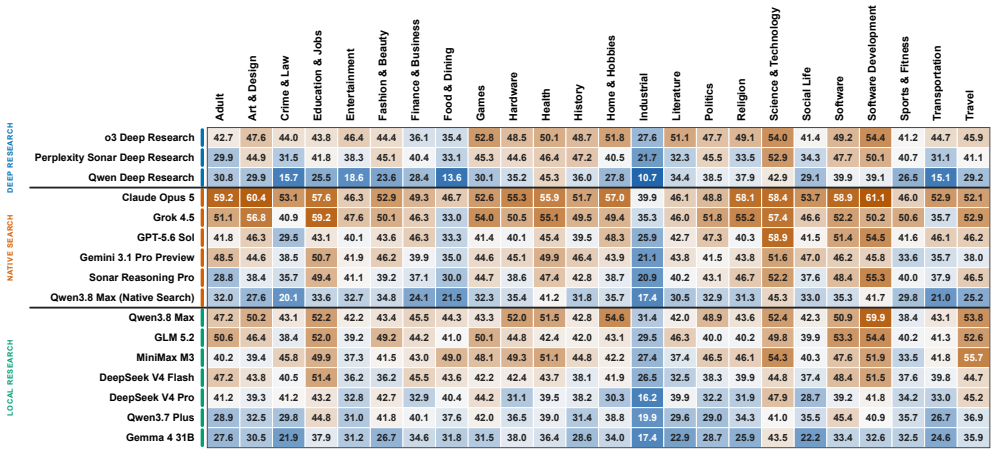



Figure 5: **Breadth across domains.**

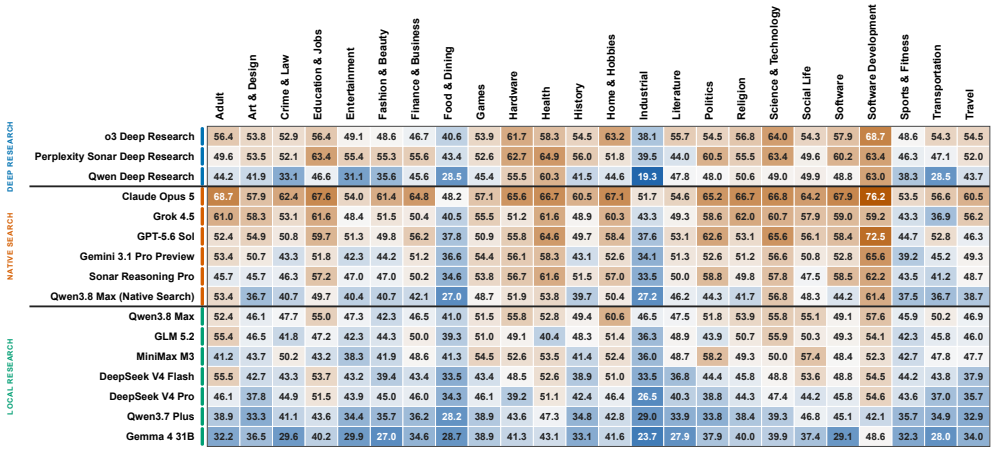



Figure 6: **Depth across domains.**

### D.2 Research Process and Trajectory Analysis

The local research framework provides unified and complete research trajectories. All seven backbone LLMs share the same Wikipedia snapshot, Search/Open interface, research workflow, and action budget. We analyze the complete $144 \times 7 = 1{,}008$ trajectories and jointly report overall coverage, breadth, and depth together with six process observables: final-report length, issued Search calls, unique retrieved Wikipedia pages, issued Open calls, Gap Unit incidence, and source-ID-valid citation markers. Table 12 provides the exact means, and Figure 7 visualizes the process profiles.

Table 12: **Coverage and process profiles under the local research framework.** Gap Unit is the percentage of cases that trigger an additional Research Unit.

| | Coverage (%) | | | Process observables | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **Backbone** | **overall ↑** | **breadth ↑** | **depth ↑** | **Report words** | **Search** | **Retrieved pages** | **Open** | **Gap Unit (%)** | **Valid citations** |
| Qwen3.8 Max | **47.94** | **46.62** | **50.37** | 2,301.6 | 10.19 | 74.31 | 30.01 | 66.67 | 18.50 |
| GLM 5.2 | 45.34 | 44.61 | 47.09 | 1,599.2 | 11.58 | 119.47 | 27.85 | 95.83 | 19.85 |
| MiniMax M3 | 45.34 | 44.34 | 47.55 | 1,995.4 | 11.22 | 67.23 | 27.72 | 97.92 | 17.76 |
| DeepSeek V4 Flash | 42.63 | 41.42 | 45.04 | 1,477.3 | 10.22 | 70.38 | 17.38 | 53.47 | 11.54 |
| DeepSeek V4 Pro | 39.22 | 36.96 | 43.03 | 1,380.1 | 11.08 | 79.29 | 13.95 | 78.47 | 10.11 |
| Qwen3.7 Plus | 36.10 | 35.37 | 37.94 | 859.0 | 9.91 | 63.44 | 24.15 | 52.08 | 12.20 |
| Gemma 4 31B | 32.00 | 30.43 | 34.82 | 629.9 | 10.82 | 55.37 | 9.31 | 88.89 | 7.84 |

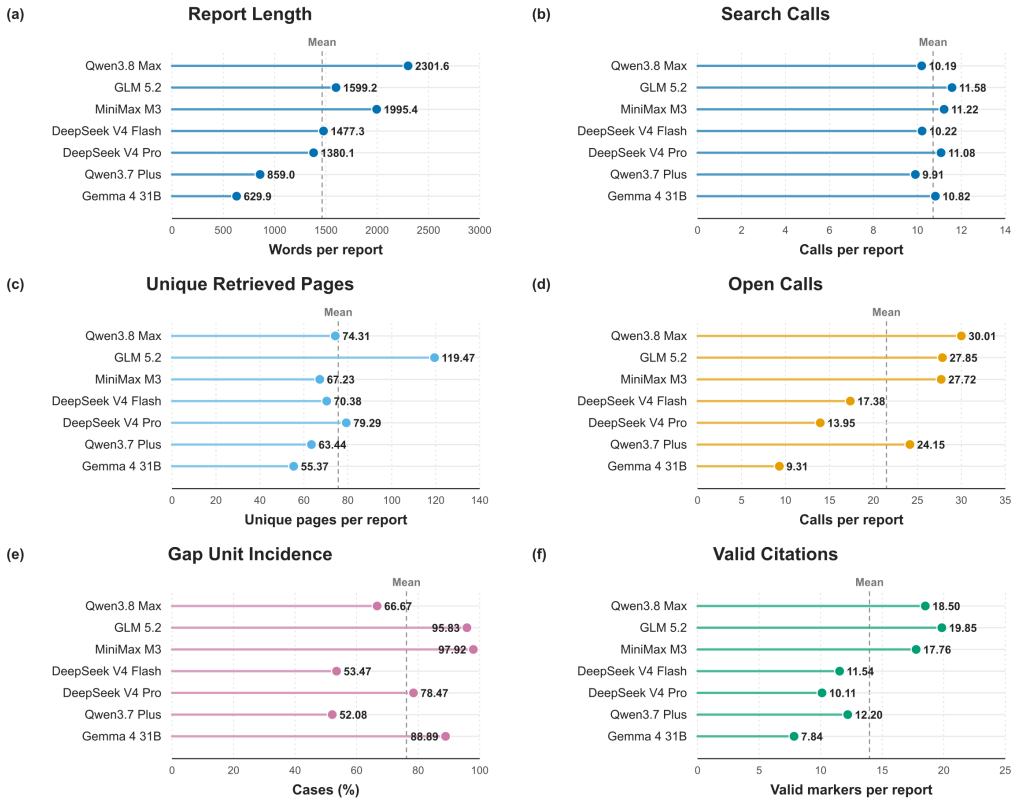



Figure 7: Research-process profiles under the local research framework: (a) final-report length, (b) issued Search calls, (c) unique retrieved Wikipedia pages, (d) issued Open calls, (e) Gap Unit incidence, and (f) source-ID-valid citation markers. Dashed lines mark the mean of the seven displayed backbone-LLM means.

Even under this unified setting, the backbones exhibit distinct research processes. Their Search-call counts are similar, but retrieved pages, Open calls, Research Unit usage, valid citations, and final-report length vary substantially. For example, Gap Unit usage is only weakly associated with coverage performance, suggesting that strong performance depends more on effective planning than on simply expanding the number of Research Units.

# E Case Study

We use three diagnostic cases to connect aggregate scores to concrete report behavior. The first requires similar overall coverage with a crossed breadth–depth profile; the second examines reports that follow largely different chains of related knowledge within the same research space; and the third is restricted to the unified local research framework and contrasts higher research activity with lower coverage performance.

## E.1 Similar overall coverage Can Hide Different Research Strategies

**Query.**

> *I'm trying to understand the career of the British entertainment mogul Lew Grade, particularly his transition from television into big-budget international film production during the 1970s and early 1980s. I'd like a well-researched overview that explains how his instincts and financing strategies produced both remarkable successes and costly failures, and how these outcomes shaped his reputation and financial standing over time. Rather than just listing his projects, I'm interested in the bigger picture: why he pursued ambitious, prestige, and family-oriented entertainment for global audiences, how his partnerships with major*

> *directors and creative figures worked in practice, and what balance of artistic risk and commercial calculation drove his decisions. I'd also like to understand how a combination of overreach and shifting business pressures eventually undermined his control of his own company and led to a corporate takeover. Please connect his personal management style to the broader changes in how film and television were financed and distributed during his era, and assess how his legacy has been judged.*

Perplexity Sonar Deep Research and Grok 4.5 obtain similar overall coverage on this case (38.83 vs. 40.31), despite exhibiting sharply inverted coverage profiles. Perplexity Sonar Deep Research covers major research directions less comprehensively but follows chains of related knowledge farther within selected directions (breadth 27.33, depth 55.19). By contrast, Grok 4.5 exhibits the opposite pattern (breadth 54.48, depth 29.82). Figure 8 shows how these contrasting profiles can yield similar overall coverage scores.

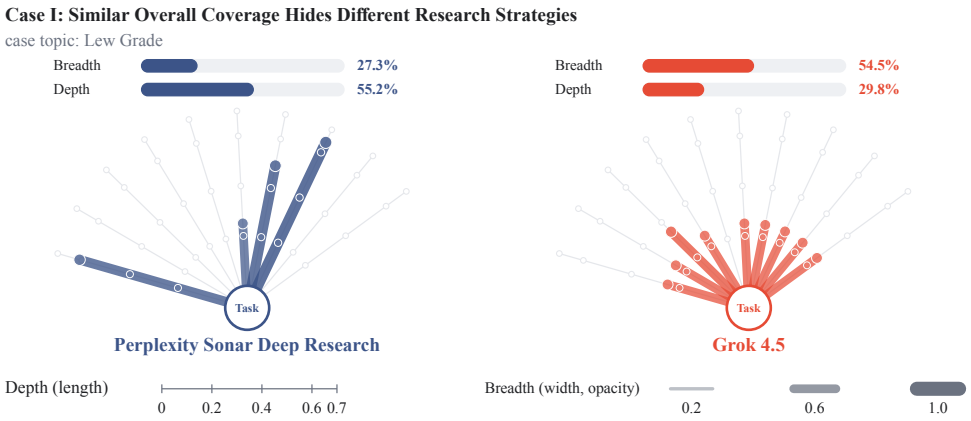



Figure 8: **Similar overall coverage can conceal different research strategies.** Bars show aggregate breadth and depth (%). Gray rays mark ten shared research directions. Colored ray width and opacity encode branch breadth, while ray length encodes branch depth.

This case illustrates a general limitation of scalar report evaluation: similar aggregate performance does not imply similar research behavior. One report covers major research directions more comprehensively, while the other follows chains of related knowledge farther within selected directions.

### E.2 Deep Research Can Explore Different Knowledge Paths

**Query.**

> *I'm trying to understand the temperance movement not just as a campaign against alcohol, but as a broad historical phenomenon that grew out of specific social conditions and then spread and changed in surprising ways. I'd like a research-style report that traces how early alarm over heavy drinking in one society, combined with new medical and moral arguments, laid the groundwork for organized reform. I'm especially curious about how different religious traditions shaped competing versions of the movement, giving it distinct doctrinal flavors and leaders, and how this influenced its appeal across communities. Beyond its origins in Britain and America, I'd like to understand how the movement traveled to colonial and settler societies, where it became entangled with other causes like women's political rights and, strikingly, with anti-colonial and independence struggles and religious revivals. Please explain how these motivations interacted, why the same idea took on such different meanings in different places, and what tensions or contradictions emerged as it evolved, without simply listing every organization or figure involved.*

Reports on the same question may emphasize different directions and knowledge chains even when their aggregate scores are similar. On this case, o3 Deep Research and GPT-5.6 Sol achieve breadth scores of 32.11 and 34.64, depth scores of 48.26 and 50.43, and overall coverage scores of 39.37 and 41.79, respectively. Despite these similar aggregate results, their branch-level coverage profiles differ substantially. Figure 9 summarizes both reports across the same ten research directions, making their differing emphases visible.

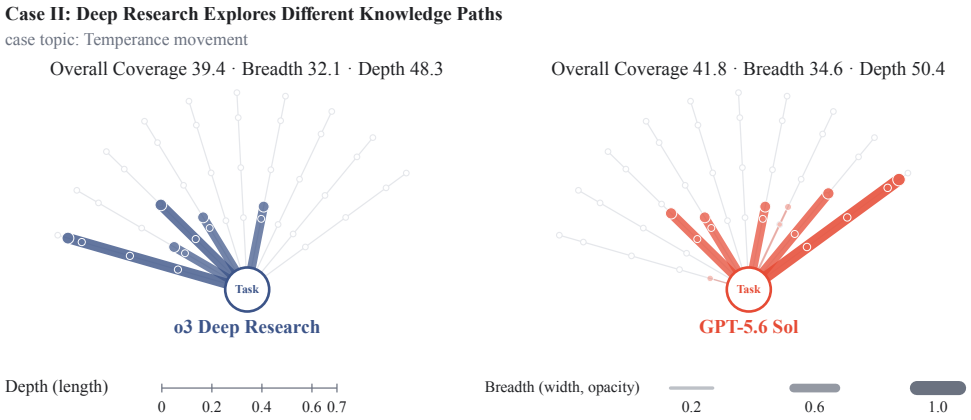



Figure 9: **Deep research can explore different knowledge paths.** Values above the models report overall coverage, breadth, and depth (%). Matched gray rays mark the same research directions. Colored ray width and opacity encode branch breadth, while ray length encodes branch depth.

This case illustrates how reports on the same question can emphasize different parts of the underlying knowledge space. Branch-level analysis makes these differences visible by showing each report's progression along knowledge chains within individual directions.

### E.3 More Research Activity Does Not Guarantee Higher Coverage

**Query.**

> *I'd like a well-researched cultural history of Horn & Hardart, the automat chain, that goes beyond describing how the vending-style restaurants worked and instead examines why this company became such a durable fixture in American popular imagination. I'm especially curious about how the company built its public presence through sponsorship of broadcast entertainment in the early days of radio, including how those programs helped launch performers who later became famous, and how that kind of promotional and media involvement shaped its brand. Second, I'd like to understand why writers, songwriters, filmmakers, and stage productions kept referencing or recreating its dining rooms across decades and what the automat came to symbolize about class, city life, and nostalgia in songs, comedies, and films. Finally, please connect its rise and eventual decline to broader shifts in urban dining, entertainment, and changing tastes, so the report reads as an interpretive account of its lasting legacy rather than just a timeline of locations and menu items.*

Table 13 summarizes the research activity, report length, and coverage scores of GLM 5.2 and DeepSeek V4 Flash on this case. Under the same frozen corpus, tool interface, workflow, and action budgets, GLM 5.2 makes more Search and Open calls, retrieves more pages, and uses more tokens, with part of its effort devoted to revisiting shared sources and expanding contextual discussion. Its report is also longer, yet DeepSeek V4 Flash achieves higher breadth, depth, and overall coverage. This case shows that greater research activity and longer reports do not necessarily imply higher knowledge coverage, highlighting the importance of effective planning and evidence selection.

Table 13: **Research activity and coverage on Horn & Hardart S03.** Pages denotes unique Wikipedia pages returned during research.

| Agent | Search | Open | Pages | Tokens (k) | Words | breadth (%) | depth (%) | overall coverage (%) |
|---|---|---|---|---|---|---|---|---|
| GLM 5.2 | 12 | 34 | 112 | 302.9 | 2,214 | 26.40 | 32.28 | 29.19 |
| DeepSeek V4 Flash | 9 | 28 | 43 | 150.4 | 1,372 | 45.42 | 45.65 | 45.54 |

# F LOCAL RESEARCH FRAMEWORK

The local research framework provides a common retrieval and orchestration setting for evaluating backbone LLMs. It standardizes the evidence corpus, tool interface, research stages, and interaction budget while leaving planning decisions, tool-use choices, and report generation to each backbone. Figure 10 summarizes the resulting workflow.

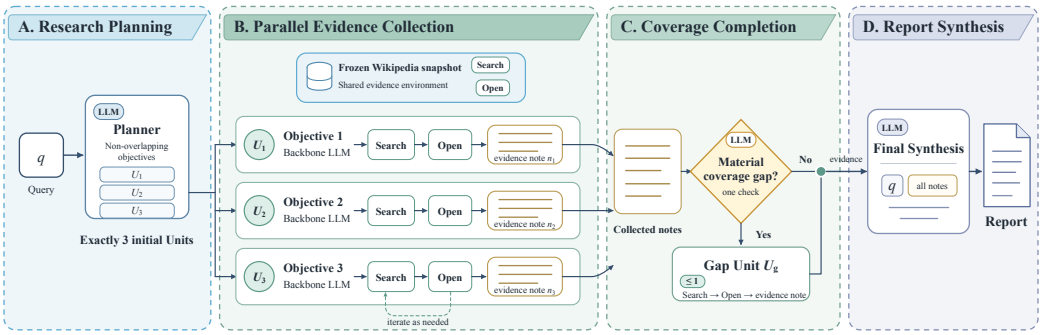



Figure 10: **Local research framework workflow.** The backbone LLM decomposes the research query into three Research Units, which execute concurrently against the shared frozen Wikipedia snapshot through SEARCH and OPEN. Their evidence notes undergo one coverage-gap check; when a material gap is found, at most one Gap Unit collects supplemental evidence before the original query and all collected notes are synthesized into the final report.

## F.1 RETRIEVAL ENVIRONMENT AND TOOL INTERFACE

All evaluations under the local research framework use the fixed English Wikipedia snapshot `enwiki-20260701-v1`. The report generator receives the benchmark query. Provider-native web search is disabled, so every backbone searches the same time-invariant evidence space through the framework's local tools.

The framework exposes two operations over this snapshot. SEARCH and OPEN serve distinct roles: SEARCH identifies potentially relevant Wikipedia pages, whereas OPEN reads their contents. Given a source identifier and a cursor position, OPEN returns a bounded text span. If additional text remains, it also returns a `next_cursor` that can be used to retrieve the subsequent span. All backbone LLMs use the same tool schemas, although they independently decide what to search for and which sources to read.

## F.2 RESEARCH WORKFLOW AND FORMAL CONFIGURATION

Given a benchmark query, a planner first decomposes the task into exactly three non-overlapping Research Units, each defined by a short title and a bounded objective. The three initial Units execute concurrently. Within each Unit, the backbone uses SEARCH and OPEN to investigate its assigned objective under fixed round and tool budgets, then returns concise evidence notes rather than a complete report. This decomposition preserves a common research structure while allowing each backbone to decide which queries to issue and which pages to read.

After the initial Units finish, the framework performs one coverage-gap check over the query and collected evidence notes. If the check identifies a material missing direction that can be investigated within the frozen Wikipedia snapshot, it creates one supplemental Gap Unit. The workflow therefore executes three Units by default and at most four Units in total, providing a bounded second stage of evidence collection without changing the protocol across backbone LLMs.

Finally, the synthesis stage receives the original query and the evidence notes produced by the Research Units and combines them into a coherent Markdown report. The resulting report is subsequently scored with the same rubric set and judge protocol used for all evaluated agents.

Table 14 summarizes the configuration of our local research framework used for the seven backbone LLMs in the main experiments.

Table 14: **Key local research framework configuration.** The same settings are used for all seven backbone LLMs evaluated under the framework.

| Parameter | Value |
|---|---|
| Wikipedia snapshot | `enwiki-20260701-v1` |
| Retrieval tools | SEARCH and OPEN |
| Provider-native web search | Disabled |
| Initial Research Units | 3 |
| Concurrent Research Units | 3 |
| Gap Units | At most 1 |
| Total Research Units | At most 4 |
| Model rounds per Unit | 7 |
| Tool calls per Unit | At most 15 |
| SEARCH calls per Unit | At most 3 |
| OPEN calls per Unit | At most 12 |
| OPEN calls per model turn | At most 4 |
| Retrieved content per OPEN call | At most 6,000 characters |

## G EVALUATION TIMEFRAMES FOR EVALUATED DEEP RESEARCH AGENTS

The outputs evaluated in this paper were collected during the timeframes listed in Table 15. These dates indicate when the reports used in our experiments were generated.

Table 15: Data collection timeframes for the evaluated deep research agents. All dates refer to 2026.

| Agent Category | Evaluated Model | Data Collection Date Range |
|---|---|---|
| Deep Research Systems | o3 Deep Research | July 21–22 |
| | Perplexity Sonar Deep Research | July 21–24 |
| | Qwen Deep Research | July 21–24 |
| Native Search LLMs | Claude Opus 5 | August 12–14 |
| | Grok 4.5 | August 12–14 |
| | GPT-5.6 Sol | August 12–14 |
| | Gemini 3.1 Pro Preview | August 12–14 |
| | Sonar Reasoning Pro | August 12–14 |
| | Qwen3.8 Max (Native Search) | August 12–14 |
| local research framework | Qwen3.8 Max | August 12–14 |
| | GLM 5.2 | August 12–14 |
| | MiniMax M3 | August 12–14 |
| | DeepSeek V4 Flash | August 12–14 |
| | DeepSeek V4 Pro | August 12–14 |
| | Qwen3.7 Plus | August 12–14 |
| | Gemma 4 31B | August 12–14 |

## H COST ACCOUNTING

We report recorded model-call costs in USD for benchmark construction, report generation, and judge evaluation.

### H.1 BENCHMARK CONSTRUCTION

We initially constructed local graphs for 240 candidate root entities and sampled three cases from each, yielding 720 candidate cases. Based on the benchmark-size analysis in Appendix C.1, we retained 48 roots and their 144 cases for the formal benchmark. All four construction stages use Claude Opus 4.8 for their model calls. Table 16 reports the construction costs for this retained set.

Table 16: Model-call cost of constructing the retained 48-root, 144-case benchmark.

| Stage | Mean cost/root (USD) | Cost/48 roots (USD) |
|---|---|---|
| Local graph construction | 26.9463 | 1,293.42 |
| Subgraph sampling | 0.1124 | 5.39 |
| Research query generation | 0.3505 | 16.82 |
| Rubric generation | 6.1453 | 294.97 |
| **Total** | **33.5544** | **1,610.61** |

### H.2 Report Generation

For each of the three Deep Research Systems, we generated reports for all 720 candidate cases. Table 17 reports the mean cost per accepted report and normalizes every agent to the final 144-case evaluation scope.

Table 17: Model-call cost of report generation for the 16 evaluated deep research agents.

| Track | Model | Mean cost/case (USD) | Cost/144 cases (USD) |
|---|---|---|---|
| Deep Research Systems | o3 Deep Research | 2.1302 | 306.75 |
| | Perplexity Sonar Deep Research | 0.5571 | 80.22 |
| | Qwen Deep Research | 0.2069 | 29.80 |
| Native Search LLMs | Claude Opus 5 | 1.8110 | 260.79 |
| | GPT-5.6 Sol | 1.2967 | 186.72 |
| | Gemini 3.1 Pro Preview | 0.1703 | 24.52 |
| | Grok 4.5 | 0.2274 | 32.75 |
| | Qwen3.8 Max | 0.0873 | 12.58 |
| | Sonar Reasoning Pro | 0.0510 | 7.34 |
| local research framework | DeepSeek V4 Flash | 0.0125 | 1.79 |
| | DeepSeek V4 Pro | 0.1336 | 19.24 |
| | Gemma 4 31B | 0.0699 | 10.06 |
| | GLM 5.2 | 0.0795 | 11.45 |
| | MiniMax M3 | 0.0620 | 8.92 |
| | Qwen3.7 Plus | 0.0925 | 13.32 |
| | Qwen3.8 Max | 0.4377 | 63.03 |
| **All 16 deep research agents** | | **7.4255** | **1,069.27** |

### H.3 Judge Evaluation

The formal evaluation contains $144 \times 16 = 2{,}304$ report–case evaluations. We use Gemini 2.5 Pro as the judge model for all evaluations. Table 18 reports the corresponding model-call costs.

Table 18: Model-call cost of judging all reports in the formal evaluation.

| Statistic | Cost (USD) |
|---|---|
| Mean per report–case evaluation | 0.2731 |
| Total for 2,304 evaluations | 629.11 |

## I Human Annotation Protocol

**Annotators and sample.** Ten PhD-level annotators participated in the human evaluation. The annotation sample comprised 3,695 report–rubric pairs from 12 reports, with each report representing a different benchmark domain. Each pair required a judgment about whether the report covered the specific knowledge expressed by one rubric. These human-reviewed labels serve as the reference for the judge-consistency analysis in Section 4.6.

**Annotation criteria.** Each report–rubric pair received one of three labels: *Satisfied*, *Partially Satisfied*, or *Not Satisfied*. *Satisfied* indicates that the report explicitly states or clearly paraphrases the

rubric's required knowledge, including its essential relation and qualifications. *Partially Satisfied* indicates that the report covers a substantive part of the requirement but omits an essential detail or qualification. *Not Satisfied* indicates that the required knowledge is absent; merely mentioning a related entity or discussing the general topic does not constitute coverage. Judgments concern content present in the report, without filling gaps from external knowledge.

**Review and comparison.** The labels underwent a second human review, and the resulting human-reviewed labels serve as the reference for comparison with three LLM judges. We compare the judges' original three-category verdicts with this reference. For the binary comparison, *Satisfied* is mapped to 1, while *Partially Satisfied* and *Not Satisfied* are mapped to 0 for both human and judge labels. Macro-F1 is computed separately for each report and then averaged equally across the 12 reports. This analysis reports agreement between each judge and the final human-reviewed reference; it does not report inter-annotator agreement.

# J Prompt Templates

We release the prompts used for benchmark construction and for evaluation to enable reproducibility and independent analysis.

## J.1 Benchmark Construction Prompts

### J.1.1 Root Entity Neighbor Selection

This prompt selects the children of the root from candidate entities extracted across its complete Wikipedia page.

**System Prompt**

```
You are selecting a set of directly related first-hop entities from the candidate links
      found in the full page of a core Wikipedia topic. These entities will become the
     main directions for later deep-research tasks, so you should balance direct
     relationship to the core topic, aspect coverage, and later composability.

Return ONLY JSON that follows the required schema. The JSON object MUST contain exactly
      two top-level keys: `selected_candidate_ids` and `selected_neighbors`.

Rules:
1. Candidate scope and count: choose entities only from the provided `
     candidate_neighbors`. The selection scope is the full-article candidate pool,
     including both lead-section and body-section candidates. If there are at least `
     target_neighbor_count` available unique candidates, return `target_neighbor_count`
      unique candidates. Return fewer only when the candidate list itself contains
     fewer available unique candidates than the target.
2. Selection objective: choose candidates that best support future root-centered deep-
     research tasks. Value here means research usefulness, aspect coverage, and
     composability for broad knowledge-coverage evaluation.
3. Coverage, combination, and duplicate filtering: prefer candidates that cover
     different research aspects of the root, and avoid concentrating on one narrow
     direction. Exclude highly similar titles, aliases, repeated directions, and
     obvious-common-knowledge or low-information candidates such as incidental mentions
     , dates, generic list pages, or weak parenthetical references. Form a
     differentiated branch set suitable for systematic DeepResearch.
4. Output structure: put the canonical selection in `selected_candidate_ids`. For each
     id in `selected_candidate_ids`, include exactly one object with the same `
     candidate_id` in `selected_neighbors`; each selected neighbor object MUST contain
     only `candidate_id` and `relation_explanation`.
5. Relationship explanation: `relation_explanation` MUST be one concise sentence
     explaining the relationship between the root and the selected entity, and it MUST
     be based only on that candidate's `anchor_text` and `context_text`.
```

**User Prompt**

```
Select first-hop root expansion entities for the following Wikipedia root.

root_title:
{root_title}

root_summary:
```

```
{root_summary}

target_neighbor_count:
{target_neighbor_count}

candidate_neighbors:
{candidate_neighbors}

Return ONLY JSON.
```

### J.1.2 Non-root Entity Neighbor Selection

This prompt selects bounded outgoing neighbors for expanding a non-root entity.

**System Prompt**

```
Given a source Wikipedia page and the candidate links from its full article, select the
     outgoing entities that are best suited as next-layer expansion topics. These
    entities should be central and directly related to the source topic, and they
    should provide valuable evidence entry points for the later research scope.

Return ONLY JSON that follows the required schema.

Rules:
1. Select from the full source-article candidate pool, including both lead-section and
    body-section candidates. You may choose only from the provided `
    candidate_neighbors`.
2. Choose the candidates that are most semantically central, most directly related to
    the source entity, and most useful for expanding understanding of it.
3. If the candidate count is at least `target_neighbor_count`, return exactly `
    target_neighbor_count` unique candidates.
4. Return only `selected_candidate_ids`.
```

**User Prompt**

```
Select the best outgoing neighbors for the following source node.

source_title:
{source_title}

source_depth:
{source_depth}

source_summary:
{source_summary}

target_neighbor_count:
{target_neighbor_count}

candidate_neighbors:
{candidate_neighbors}

Return ONLY JSON.
```

### J.1.3 Subgraph Sampling

This prompt groups the children of the root to select branches that form coherent and complementary subgraphs for downstream task construction.

**System Prompt**

```
You are selecting and combining candidate directions that are directly connected to a
    core Wikipedia topic, forming several subtopic sets suitable for deep research.
    Each set should feel like a natural research scope: its internal directions should
     be related and complementary, able to support later research-question framing,
    evidence organization, and evaluation of whether a report covers key knowledge
    relationships.

Return ONLY JSON that follows the required schema.

Rules:
```

```
1. When the candidate pool allows it, construct exactly `target_subgraphs` groups; when
     the candidate pool allows it, each group should contain exactly `
     depth_1_branches_per_subgraph` unique candidates. Choose only from the provided
     depth-1 candidate list.
2. Each group should form a coherent research scope: the selected first-hop branches
     should have semantic, historical, institutional, causal, comparative, or
     conceptual connections, and should be able to support a natural deep-research
     report. Do not simply choose the first candidates. Prefer branches whose
     descendant tree is more likely to provide effective supporting evidence.
3. Branches within a group should be complementary. Avoid homogeneous branches, near
     duplicates, or branches that can be represented by the same angle. Unless the seed
     topic naturally connects them, do not combine unrelated incidental mentions.
4. Different groups should overlap as little as possible. If overlap is unavoidable,
     prefer reusing candidates that have the most direct relationship to the seed/root,
     can connect multiple research angles, and whose descendant tree is more likely to
     provide effective supporting evidence.
5. Use `selected_candidate_ids` as the canonical output.
```

**User Prompt**

```
Select depth-1 branch groups for subgraph construction.

seed_entity:
{seed_entity}

root_summary:
{root_summary}

target_subgraphs:
{target_subgraphs}

depth_1_branches_per_subgraph:
{depth_1_branches_per_subgraph}

candidate_count:
{candidate_count}

depth_1_candidates:
{depth_1_candidates}

Return ONLY JSON.
```

#### J.1.4 Research Query Generation

This prompt converts the hidden sampled scope into a research query centered on the root entity while preventing entity-list leakage.

**System Prompt**

```
You convert a set of hidden entity-relation clues into a research question that an
     outside user might ask.

Your task is to write ONE research-style query centered on the seed/root entity. This
     query will be given to a downstream DeepResearch model to generate a report, and
     coverage will be evaluated using rubrics corresponding to the hidden clues. `
     subgraph_nodes` is hidden construction context: use it only to understand the
     overall topical composition and depth profile. The final query must still present
     the root/seed entity as the explicit central topic.

Return ONLY JSON that follows the required schema. The JSON object MUST contain exactly
     one top-level key: 'query'.

Rules:
1. Central topic: the final `query` MUST clearly center on `seed_entity`; the seed/root
     entity name may appear. The query should use the full hidden context to form a
     research task, rather than only asking about the root article itself.
2. Anti-leakage: the final `query` MUST NOT directly mention any entity name in `
     forbidden_entity_names`, any exact title from `depth_1_branch_inputs`, or any
     exact title from `subgraph_nodes` except the seed/root entity. Also avoid semantic
     leakage: do not rewrite hidden candidate entities, node titles, summaries, or
     relation explanations into downstream-facing concrete topics, mechanisms,
     institutions, places, events, works, methods, people, effects, or checklists.
     Deeper hidden entities may not all appear in the forbidden list, but the query
     should still not read like an entity checklist.
```

```
3. Writing style: imitate a thoughtful but non-expert user asking for research help.
     The query should be specific and purposeful, but should not make the user appear
     to know hidden entities, internal technical terms, hidden graph structure, or
     specific field methods. Use a natural-language paragraph.
4. Source of specificity: the query should usually be one coherent paragraph, around
     100-200 words in English or a similar level of detail in the target language. It
     must not be too short or too generic, and it must not become specific by
     enumerating hidden candidate directions. Specificity should come from root-
     centered research framing, such as causal relationships, historical development,
     changes in evidence and interpretation, interactions among different forces, or
     how internal practices and external pressures shaped development.
```

**User Prompt**

```
Generate a query from the hidden construction context below.

query_gen_mode:
{query_gen_mode}

graph_metadata:
{graph_metadata}

seed_entity:
{seed_entity}

subgraph_overview:
{subgraph_overview}

root_node:
{root_node}

subgraph_nodes:
{subgraph_nodes}

depth_1_branch_inputs:
{depth_1_branch_inputs}

forbidden_entity_names:
{forbidden_entity_names}

Important:
- The hidden node set is rooted at `seed_entity`.
- The root node is depth 0.
- Each `subgraph_nodes` item contains a node id, Wikipedia title, and root-hop depth.
- A smaller depth means fewer knowledge hops from the root and usually a closer
    structural relationship to it.
- The final query should be a research task based on the full hidden context while
    staying centered on the root entity.
- Node titles are hidden construction evidence and must not become a topic checklist in
     the final query.

Return ONLY JSON.
```

### J.1.5 Rubric Generation

This prompt converts each edge in a case subgraph into an evaluation rubric based on its associated evidence.

**System Prompt**

```
You will receive a batch of relationship records, each made from a source entity and a
     target entity. Each record includes evidence from Wikipedia page summaries and
     relationship context. Your task is to write one verifiable knowledge-coverage
     requirement for each record, so that a later research report can be checked for
     whether it clearly covers the specific knowledge relationship expressed by that
     record.

For each relationship record, use only the materials provided by that record: source
     page summary, target page summary, and relation evidence sentence / paragraph. Do
     not use internal structural information such as node depth, path position,
     centrality, or subgraph topology.

Return ONLY JSON that follows the required schema.
```

```
Rules:
1. Generate exactly one rubric object for each provided `rubric_id`.
2. Each rubric must be faithful to that relationship record's own source/target/
     evidence materials, and must be written as a knowledge coverage requirement for
     report content, not as a writing-quality judgment.
3. The rubric must be based on the provided page summaries and relationship evidence.
     Do not invent claims beyond the summaries and relation evidence. If the evidence
     is weak, keep the rubric narrow and concrete. If the evidence sentence supports a
     clear relationship, prefer explicit relation coverage. A report may satisfy the
     same knowledge content through a clear paraphrase.
4. Use the exact key names from the output schema. Each rubric may return only `
     rubric_id` and `rubric_text`.
```

**User Prompt**

```
For each relationship record in the following list, generate one rubric object based on
      the provided Wikipedia page evidence.

edge_rubric_inputs:
{edge_rubric_inputs}

Return ONLY JSON.
```

## J.2 Evaluation Prompt

### J.2.1 Judge Prompt

**System Prompt**

```
You need to evaluate whether a report covers a set of predefined scoring points.

The input will include a user research question, several rubric objects, and the report
      to be evaluated. Each rubric object represents one specific knowledge point to
     check, and `rubric_text` is the only scoring criterion. Your task is to judge,
     item by item, whether the report satisfies the corresponding `rubric_text`, and to
      provide a brief reason for each judgment.

Return only JSON that conforms to the required schema.

Rules:
1. Score each rubric object using only its `rubric_text` and a three-tier scoring
     scheme: `1` means satisfied, `0.5` means partially satisfied, and `0` means not
     satisfied. Base your judgment only on what is explicitly present in the report. Do
      not use external knowledge, common-sense completion, or subjective inference.
2. Assign `1` when the report explicitly states or clearly paraphrases the knowledge
     required by the rubric, covering the key facts, relationships, qualifiers, or
     numerical details required by the rubric. Reasonable paraphrases are acceptable,
     but the rubric must be substantively satisfied.
3. Assign `0.5` when the report mentions the core object, relationship, or fact
     required by the rubric and covers at least one substantive key element, but misses
      other key elements, qualifiers, dates, numerical values, causal links, or
     necessary context. General discussion of a related topic, mere mention of related
     terms, or content that does not cover any specific knowledge required by the
     rubric is not enough for `0.5`.
4. Assign `0` when the report does not mention the specific knowledge required by the
     rubric, only discusses a related topic generally, or makes a factual statement
     that contradicts the rubric. If the report's core factual claim contradicts the
     rubric, assign `0` rather than `0.5`, even if related content is mentioned.
5. For every input rubric object, return exactly one object in `rubric_results`, and
     preserve the original `rubric_id` exactly. Do not omit, merge, rewrite, or add
     rubrics.
6. Each result must contain only `rubric_id`, `score`, and `reason`. The `reason`
     should briefly explain why the rubric is satisfied, partially satisfied, or not
     satisfied, and should indicate which key elements the report covered, which key
     elements it missed, or which fact contradicts the rubric.
7. Keep each `reason` within 50 words.
```

**User Prompt**

```
Please formally score a report against a provided rubric set.

query:
```

```
{query}

The report to evaluate is provided in the next message. The rubric_objects to score are
     provided after the report. When asked to respond, return only JSON. Each rubric
    result must contain only `rubric_id`, `score`, and `reason`.

# Report To Evaluate
{report_text}

rubric_objects:
{rubric_objects_json}

Now evaluate the report against exactly the rubric object(s) listed above.
Use only the provided rubric ids.
score must be one of:
- 1: The report explicitly states or clearly paraphrases the knowledge required by the
    rubric and covers the key elements
- 0.5: The report covers some substantive key elements of the rubric, but misses other
    necessary elements or qualifiers
- 0: The report does not mention the specific knowledge, is only generally related, or
    factually contradicts the rubric

Each reason must be concise: within 50 words.

Return ONLY JSON.
```