daVinci-Searcher: ELICITING DEEP RESEARCH CAPABILITIES DATA EFFICIENTLY

Anonymous authors
Paper under double-blind review

ABSTRACT

Deep research agents are increasingly important for solving complex, information-intensive tasks that require sustained search and evidence integration, and training on synthetic trajectories has enabled open-source agents to achieve performance competitive with proprietary systems. We ask: *Can a small set of high-quality trajectories elicit strong deep research capabilities* ***data efficiently*** *through supervised fine-tuning alone?* We introduce daVinci-Searcher, which combines scalable task synthesis with trajectory selection. Our Wikidata-based pipeline constructs challenging, verifiable multi-entity questions, using obfuscated entities and cross-entity constraints to control difficulty and encourage sustained search. We further identify *looping reasoning* in teacher trajectories, where excessively long assistant turns revisit the same hypotheses without progress. Supported by external judge evaluations, our *loop-aware trajectory filtering* uses maximum per-turn reasoning length as a proxy for looping and improves performance without requiring correct final answers. We find a practical sweet spot at 2,000 training trajectories across 4B and 35B students, with no consistent gains from further scaling. Our 35B agent achieves 66.4% on BrowseComp, 60.8% on BrowseComp-zh, and a 62.8% macro average across six research benchmarks, outperforming the same backbone fine-tuned on the substantially larger S1-DR and Red-Searcher datasets.

1 INTRODUCTION

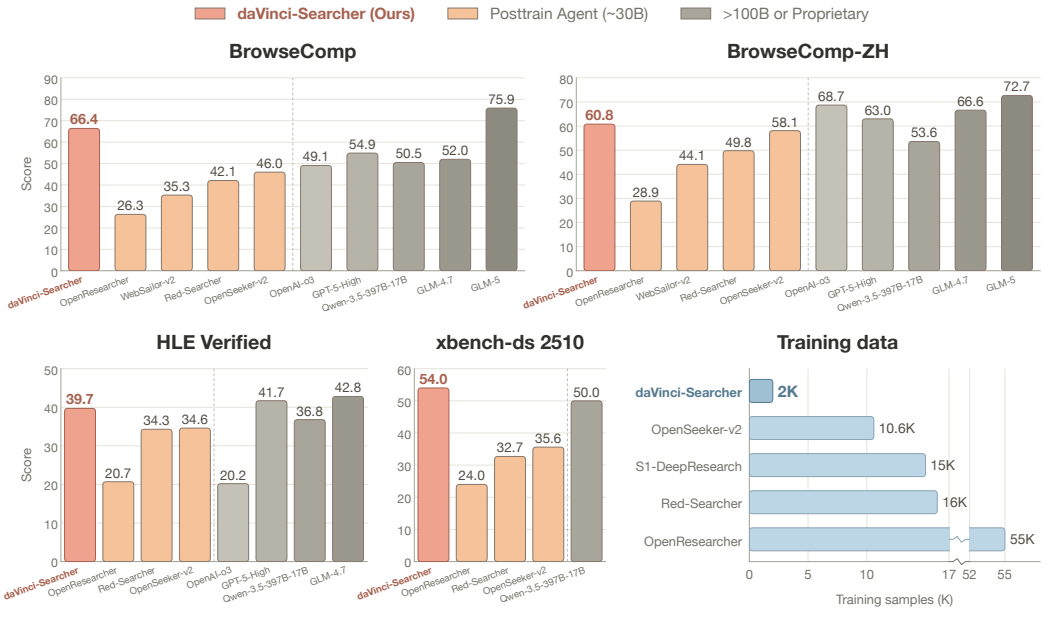

Figure 1: Performance comparison on BrowseComp, BrowseComp-ZH, HLE Verified, and xbench-ds 2510, alongside training-data sizes for selected search agents. Our 35B daVinci-Searcher achieves competitive results with SFT on just 2,000 curated trajectories, highlighting its data efficiency.

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.

In the era of large language model (LLM) agents, deep research agents have emerged as an important approach to solving information-intensive problems. By interleaving reasoning, web search, and source inspection, they investigate questions whose evidence is scattered across documents and websites (Yao et al., 2023). Benchmarks such as BrowseComp highlight the difficulty of this process: an answer may be easy to verify, yet finding it requires sustained search and evidence integration (Wei et al., 2025). Developing these capabilities calls for demonstrations of how to plan an investigation, connect evidence across multiple sources, and recover when a search stalls.

Recent efforts such as S1-DeepResearch (S1-DR) (Dong et al., 2026) and Red-Searcher (Chu et al., 2026) demonstrate that open-source foundation models trained on synthetic data can rival proprietary systems on deep research benchmarks. OpenSeeker and OpenSeeker-v2 (Du et al., 2026b;a) further show that competitive performance can be achieved through a lean supervised fine-tuning (SFT) stage alone, highlighting the importance of training data quality. Prior work on small, high-quality datasets (Ye et al., 2025; Jiang et al., 2026) motivates us to ask: *Can SFT on only a small set of high-quality trajectories elicit strong deep research capabilities* ***data efficiently****?*

Achieving such *data efficiency* requires considering both what a task demands and what its solution trajectory teaches. Questions answerable from memory or a brief lookup offer limited practice in extended research. At the same time, demanding questions can induce a teacher to produce repetitive reasoning that provides poor supervision. A compact training set therefore needs tasks that elicit sustained investigation and trajectories that demonstrate productive research behavior. We introduce daVinci-Searcher, which combines these two forms of data curation to train deep research agents with only 2,000 trajectories through standard supervised fine-tuning alone.

Our approach comprises three stages. ❶ **Task synthesis with entity scaling.** Building on graph-grounded synthesis (Chu et al., 2026; Du et al., 2026b), we construct Wikidata (Vrandečić & Krötzsch, 2014) questions with obfuscated entities and cross-entity constraints to control difficulty and require sustained search. Executable queries verify answer uniqueness in the graph, while semantic audits and closed-book screening check faithfulness and retrieval needs. ❷ **Trajectory synthesis and filtering.** We collect teacher trajectories with web search, webpage inspection, and code execution, then apply *loop-aware trajectory filtering* using maximum per-turn reasoning length, including the final turn, to remove repetitive, unproductive reasoning. Turn-count limits favor sustained tool interaction, and both correct and incorrect trajectories are retained. ❸ **Data-efficient trajectory SFT.** SFT alone elicits deep research capabilities in 4B and 35B students using only 2,000 filtered trajectories.

Across six research benchmarks, our 35B agent achieves a 62.8% macro average, including 66.4% on BrowseComp and 60.8% on BrowseComp-zh, outperforming the same backbone trained on substantially larger S1-DR and Red-Searcher datasets. Scaling beyond 2,000 trajectories yields no consistent gains across both student sizes. Ablations support removing looping reasoning and suggest that final-answer correctness need not be used as a filtering signal when selecting trajectories for SFT. Discard-all context management improves accuracy at the cost of latency. Together, these findings emphasize task difficulty and trajectory quality over data volume.

Our contributions are threefold:

- We develop a scalable pipeline for generating challenging, verifiable multi-entity research tasks. Using trajectories synthesized for these tasks, we train daVinci-Searcher and demonstrate its competitive performance on deep research benchmarks.
- We characterize looping reasoning in teacher demonstrations and introduce *loop-aware trajectory filtering* using maximum per-turn reasoning length as a proxy, supported by independent judge evaluations and SFT ablations of downstream performance.
- We train deep research agents *data efficiently* with only 2,000 trajectories. Our analyses and ablation studies examine how data scale, trajectory quality, and context management affect downstream performance across the evaluated models and benchmarks.

2 Related Work

Synthetic supervision for deep research agents. Deep research agents use synthetic tasks and tool-interactive demonstrations to learn sustained information seeking (Team et al., 2026; Wu et al., 2025; Li et al., 2025b;a). Graph-grounded synthesis controls task complexity and evidence dispersion (Chu

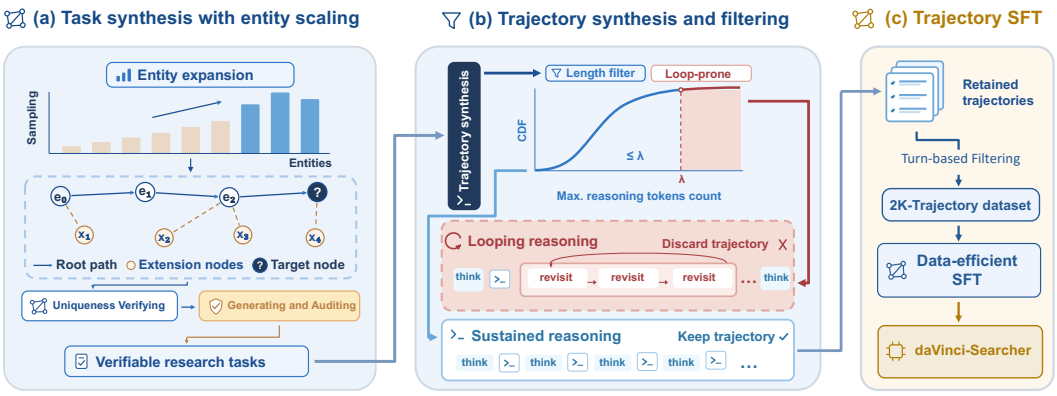

Figure 2: Overview of daVinci-Searcher. (a) More obfuscated entities increase task difficulty. (b) Loop-aware trajectory filtering uses maximum per-turn reasoning length as a proxy for looping. (c) These steps, along with turn-based filtering, improve *data efficiency* in trajectory SFT.

et al., 2026; Du et al., 2026b; Dong et al., 2026), while offline environments enable reproducible trajectory collection (Li et al., 2026). ABSeeker adds answer-backtracked credit assignment (Lu et al., 2026), WebLeaper promotes information-rich seeking (Tao et al., 2025), and WideSeek and WideSeek-R1 study multi-agent width scaling (Huang et al., 2026; Xu et al., 2026). We investigate how task design and trajectory selection support compact SFT, constructing verifiable questions with multiple interacting entity constraints. We compare task sources using a common teacher and student to examine how the resulting demonstrations affect downstream research performance.

Data efficiency and trajectory selection. Small, curated datasets can support instruction tuning and reasoning (Zhou et al., 2023; Ye et al., 2025; Muennighoff et al., 2025). Data selection further exploits complexity, quality, diversity (Liu et al., 2024), gradient influence (Xia et al., 2024), and learning impact (Li et al., 2025c). Search-agent training examines trajectory informativeness (Du et al., 2026a), final-answer correctness (Li et al., 2026), and step-level utility (Lu et al., 2026). Overthinking analyses and reasoning compression address redundant computation (Chen et al., 2025a; Xia et al., 2025; Hou et al., 2025). Our loop-aware trajectory filtering uses maximum per-turn reasoning length as a proxy for looping while preserving sustained tool interaction and demonstrations with either correct or incorrect answers. Together with a study of training data scale, this identifies a practical SFT setting with 2,000 trajectories for both 4B and 35B students on research benchmarks.

3 Methods

3.1 Task Formulation

Following graph-based task synthesis (Chu et al., 2026; Du et al., 2026b), we represent a multi-hop question by a subgraph $\mathbb{G} = (\mathbb{V}, \mathbb{E})$, where $\mathbb{V} = v_1, v_2, \ldots, v_k$ denotes the set of entity nodes involved in the reasoning process. The relations among these entities define the edges between nodes. Each edge in $\mathbb{E}$ is represented as a triplet $e_i = (v_s, v_t, w)$, where v_s and v_t denote the source and target entities, respectively, and w denotes the relation between them. A larger number of entities generally corresponds to a longer reasoning chain and a more complex evidence graph. Given a subgraph $\mathbb{G}$, a synthetic query can be generated using the following distribution:

$$q \sim \pi_{gen}(q \mid \mathbb{G}) \tag{1}$$

We represent a tool-interactive trajectory as a ReAct-style sequence of reasoning, actions, and observations (Yao et al., 2023). At each step, the agent performs reasoning, invokes a tool, and receives the corresponding observation. This process continues until the agent obtains sufficient information and produces a final answer. The trajectory can be represented as:

$$t = (\tau_1, a_1, o_1, \ldots, \tau_T, a_T, o_T, \tau_{T+1}, y) \tag{2}$$

where τ_i, a_i, and o_i denote the reasoning trace, tool action, and observation at step i, respectively, and y denotes the final answer. By increasing the complexity of the subgraph structure, we can construct questions that require additional reasoning hops across multiple connected entities.

3.2 Task Synthesis with Entity Scaling

To unlock deep research capabilities with high ***data efficiency***, we design tasks requiring sustained retrieval and evidence integration. Existing works like (Chu et al., 2026) advocates complex question subgraphs with cyclic or diamond topologies, but such structural complexity may introduce topological shortcuts for direct entity matching and induce superficial reasoning. We propose a simple but effective strategy: scaling the number of obfuscated entities along a linear root path (Figure 2(a)), an agent identify them from relational clues and connect them through factual relations.

Obfuscated entity scaling. We instantiate Wikidata subgraphs within Leiden communities (Traag et al., 2019) for topical coherence (§I.1). We use α-dropping to limit entity reuse (§I.2). We replace root entity names with context-dependent descriptions of functional roles and relations to other root and extension entities, encouraging deep reasoning rather than direct entity matching. We define:

$$N_{\text{obf}}(q) = |\mathbb{V}_{\text{root}}| = m, \qquad k = |\mathbb{V}| = m + n_{\text{ext}}. \tag{3}$$

Here, $\mathbb{V}_{\text{root}} = (v_1, \ldots, v_m)$ is the root path ending at the answer. Its length $m = N_{\text{obf}}(q)$ counts obfuscated entities and controls structural difficulty, with a proportionate number n_{ext} of extension entities and k entities in total. Explicitly named extension entities link to root entities to help disambiguate the target. We require at least two distinct relation types and preserve relevant qualifiers, including temporal constraints, to support executable queries. Figure 4 compares trajectory lengths across root-path lengths, showing how obfuscated entity counts affect research steps.

Verifying answer uniqueness. To validate that each question has a unique answer supported by factual evidence, we execute a conjunctive query encoding its subgraph relations and qualifiers against the latest available Wikidata snapshot, whose broad coverage of millions of entities and fine-grained structured relations provides a rich factual basis for verification. This ensures answer existence, uniqueness, and factual grounding in authoritative web knowledge (§I.3).

Generating and auditing questions. We verbalize verified subgraphs with templates and use an LLM to rewrite these structured descriptions as questions. An LLM judge then selects a candidate question (Zheng et al., 2023) (§I.4). An independent constraint-level audit rejects reversed relations, incorrect target references, reassigned facts, and unsupported claims. We retain only questions that a capable LLM fails to answer correctly in every sampled closed-book attempt, favoring tasks that require external retrieval (§I.5). Appendix K presents example question graphs.

3.3 Trajectory Synthesis and Loop-Aware Trajectory Filtering

3.3.1 Tool Protocol for Trajectory Synthesis

The agent has access to three tools for web search, web visit, and code execution, as summarized in Table 8 in Appendix G. (1) *Web search*: we employ widely-used Serper[1] as the search backend and return the top 50 relevant documents by default to provide broad coverage of information from the open web and reduce the risk that exposure bias causes the model to converge on a suboptimal answer. (2) *Web visit* also uses Serper as its backend. To limit context growth from irrelevant webpage content, an external LLM[2] summarizes the retrieved pages and extracts relevant snippets before returning them to the agent. (3) *Code execution* tool enables the model to write and execute Python code in an isolated sandbox for computation and analysis of information gathered during the investigation.

[1] https://serper.dev/

[2] All external LLMs below default to gpt-oss-120b (OpenAI, 2025a), including the webpage summarizer and answer judge, while the agent trajectory generator uses the separate teacher model specified below.

3.3.2 LOOP-AWARE TRAJECTORY FILTERING

Longer reasoning turns are more likely to exhibit *looping reasoning*, revisiting the same keywords, hypotheses, and intermediate conclusions without progress, and Figure 2(b) marks this *loop-prone* region. We observe this across teachers and tasks from OpenSeeker (Du et al., 2026b) and S1-DR (Dong et al., 2026) (Figures 6 and 14). Maximum per-turn reasoning length captures isolated long turns hidden by trajectory averages. It correlates strongly with independent loop scores and selects similar subsets (Figure 7), motivating *loop-aware trajectory filtering*.

We collect ReAct trajectories using a teacher LLM[3]. From the rollout pool $\mathcal{D}_{\text{rollout}}$, we retain:

$$\begin{aligned} R_{\max}(t) &= \max_{1 \le i \le T(t)+1} |\tau_i|_{\text{tok}}, \\ \mathcal{D}_{\text{keep}} &= \{ t \in \mathcal{D}_{\text{rollout}} \mid R_{\max}(t) \le \lambda, \quad 10 < T(t) < 200 \}. \end{aligned} \tag{4}$$

Here, $T(t)$ indicates tool-interaction counts, $|\tau_i|_{\text{tok}}$ counts reasoning tokens in assistant turn i, and λ is the per-turn limit. To ensure that each trajectory contains sufficient high-quality tool interactions without an excessively long interaction history, we follow OpenSeeker-v2 (Du et al., 2026a) and retain only trajectories with more than 10 and fewer than 200 tool calls. We retain both correct and incorrect trajectories because either can demonstrate useful research strategies and reasoning patterns, motivated by OpenResearcher (Li et al., 2026), and we discuss this choice in Section 5.2.

3.4 DATA-EFFICIENT TRAJECTORY SFT

From the retained pool, we uniformly sample trajectories to construct SFT datasets of different sizes (Figure 2(c)). We fine-tune the student with the standard cross-entropy objective, computing loss only on assistant tokens. Our data-size study exhibits a *less is more* effect: performance does not improve monotonically as more trajectories are added. We identify 2,000 trajectories as a practical sweet spot that achieves near-best performance among the evaluated scales and use this setting for our reported models. Section 4.2 and Figure 3 examine how performance varies with training-set size.

4 EXPERIMENTS

4.1 SETUP

Training Configuration. We develop daVinci-Searcher, a deep research agent initialized from Qwen3.5-35B-A3B (Qwen Team, 2026), which uses a mixture of experts (MoE) architecture. We train the model with a standard cross entropy loss computed exclusively on tokens in assistant turns of agent trajectories, while instruction and observation tokens are masked out of the loss. Table 4 provides the detailed training configuration. Since high quality agent trajectories tend to be long, we extend the context length to 256K tokens and use context parallelism to accommodate the resulting memory requirements. To improve training throughput, we adopt a sample packing strategy that concatenates multiple examples into a single training sequence up to the maximum context length. The attention mask restricts self attention to tokens within each example, preventing information leakage between packed examples. This strategy substantially improves GPU utilization.

Evaluation Configuration. We evaluate long-horizon research on BrowseComp (Wei et al., 2025), BrowseComp-zh (Zhou et al., 2025), BrowseComp-Plus (Chen et al., 2025b), HLE Verified (Zhai et al., 2026), derived from Humanity's Last Exam (Phan et al., 2026), the text subset of GAIA (Mialon et al., 2023), and xbench-DeepSearch (Xbench-Team, 2025), using its 2510 release. We report pass@1 scores averaged over three runs and use an external model as the judge (Zheng et al., 2023). We aggregate five sampled judgments, adapting the self-consistency principle (Wang et al., 2023) to reduce sensitivity to the variability of individual model judgments when scoring each final answer.

Baselines. We compare daVinci-Searcher with three model categories. Proprietary systems include the Claude series (Anthropic, 2025a;b), OpenAI-o3 (OpenAI, 2025b), OpenAI Deep Research (OpenAI, 2025), and GPT-5-High (Singh et al., 2026). Open source foundation models include DeepSeek-V4-Pro-Max (DeepSeek-AI et al., 2026), the Qwen3.5 family (397B-A17B, 35B-A3B, and 4B) (Qwen Team, 2026), GLM-4.7 (Z.AI, 2025), and GLM-5 (GLM-5-Team et al., 2026). Open

[3]We use DeepSeek-V4-Flash-0731 from the DeepSeek-V4 family (DeepSeek-AI et al., 2026).

Table 1: Performance comparison across six benchmarks. The highest scores within each model category are in **bold**. Gray shading indicates our model. Overall is the macro average, reported only when all six benchmark scores are available. BC denotes BrowseComp; dashes indicate unavailable values. † denotes official results; ‡ denotes results reported in OpenSeeker-v2 (Du et al., 2026a). All other results are obtained using our evaluation protocol and reported as the average of 3 runs (§ denotes the discard-all context management strategy).

Model	Model Size	Training Size	BC-zh	BC	BC-plus	HLE Verified	GAIA-text	xbench-ds 2510	Overall
Proprietary Systems									
Claude-4-Opus (Anthropic, 2025a)	–	–	37.4†	18.8†	–	–	–	–	–
Claude-4-5-Sonnet (Anthropic, 2025b)	–	–	42.4†	24.1†	–	32.0†	–	–	–
GPT-5-High (Singh et al., 2026)	–	–	63.0†	**54.9**†	–	**41.7**†	–	–	–
OpenAI Deep Research (OpenAI, 2025)	–	–	42.9†	51.5†	–	26.6†	–	–	–
OpenAI-o3 (OpenAI, 2025b)	–	–	**68.7**†	49.1†	–	20.2†	–	–	–
Open-Source Foundation Models									
DeepSeek-V4-Pro-Max (DeepSeek-AI et al., 2026)	1.6T	–	–	**83.4**†	–	–	–	**80.0**†	–
Qwen3.5-397B-A17B (Qwen Team, 2026)	397B	–	–	–	59.4§	36.8	**77.1**	50.0	–
GLM-4.7 (Z.AI, 2025)	358B	–	66.6‡	52.0‡	**70.2**†	**42.8**‡	–	–	–
GLM-5 (GLM-5-Team et al., 2026)	744B	–	**72.7**†	75.9†	–	–	–	–	–
Qwen-3.5-35B-A3B (Qwen Team, 2026)	35B	–	43.8§	34.6§	63.9§	36.4	66.7	36.0	**46.9**
Qwen-3.5-4B (Qwen Team, 2026)	4B	–	29.5§	25.6§	55.4§	29.0	67.0	32.3	39.8
Post-Training Search Agents									
OpenResearcher (Li et al., 2026)	30B	55K	28.9§	26.3†	54.8†	20.7	64.1†	24.0	36.5
Tongyi-DeepResearch (Team et al., 2026)	30B	–	46.7†	43.4†	–	–	70.9†	–	–
WebSailor-V2 (Li et al., 2025a)	30B	>30K	44.1†	35.3†	–	–	–	–	–
Red-Searcher (Chu et al., 2026)	30B	15K + 1K	49.8‡	42.1‡	60.4§	34.3†	80.1†	32.7	49.9
OpenSeeker (Du et al., 2026b)	30B	11.7K	48.4†	29.5†	–	–	–	–	–
OpenSeeker-v2 (Du et al., 2026a)	30B	10.6K	58.1†	46.0†	–	34.6†	–	35.6	–
Qwen3.5-35B-A3B + *S1-DR*	35B	15K	41.7§	33.0§	60.9§	34.7	63.6	34.3	44.7
Qwen3.5-35B-A3B + *Red-Searcher*	35B	10K[4]	46.9§	37.0§	68.4§	39.0	71.4	38.5	50.2
daVinci-Searcher-4B	4B	2K	53.9§	57.8§	64.8§	34.3	72.1	47.1	55.0
daVinci-Searcher-35B	35B	2K	**60.8**§	**66.4**§	**73.6**§	**39.7**	**82.0**	**54.0**	**62.8**

source search agents specialized through posttraining include OpenResearcher (Li et al., 2026), Tongyi-DeepResearch (Team et al., 2026), WebSailor-V2 (Li et al., 2025a), Red-Searcher (Chu et al., 2026), OpenSeeker (Du et al., 2026b), and OpenSeeker-v2 (Du et al., 2026a). Table 1 marks officially reported results with † to distinguish them from results obtained in our evaluation.

Inference Configuration. We cap each task at 300 interaction turns and apply a 150-minute time limit independently to each attempt. The model may issue multiple tool calls in each turn, and we impose no explicit cap on the total number of tool calls. For benchmarks in the BrowseComp family, we adopt the *Discard-all* context management strategy used in the DeepSeek-V3.2 technical report (DeepSeek-AI et al., 2025) to mitigate the decline in search efficiency caused by erroneous and lengthy interaction histories. Discard-all attempts do not share a single time budget: the total generation time for a task accumulates across attempts and can therefore exceed 150 minutes.

4.2 Main Results and Analysis

Table 1 shows that daVinci-Searcher-35B achieves 60.8%, 66.4%, 73.6%, 39.7%, 82.0%, and 54.0% on BC-zh, BC, BC-plus, HLE Verified, GAIA-text, and xbench-ds 2510, respectively. Our 35B model also matches or surpasses much larger foundation models and leading proprietary systems on multiple benchmarks. Relative to the Qwen3.5-35B-A3B base model, these results represent absolute gains of 31.8 and 17.0 percentage points on BrowseComp and BrowseComp-zh, respectively. We also fine-tune the same backbone on S1-DR (15K examples) and Red-Searcher (10K), obtaining average scores of 44.7% and 50.2%, below daVinci-Searcher-35B's 62.8% with only 2K examples.

[4]For a fair SFT-only comparison with daVinci-Searcher, we use merely the 10K text SFT examples from Red-Searcher, excluding its 5K multimodal examples and additional 1K RL examples.

Red-Searcher improves the base model's average of 46.9% by just 3.3 points, indicating limited gains in deep research. These comparisons support the importance of both query and trajectory quality in achieving ***data efficiency*** through task construction and loop-aware trajectory filtering.

Less Is More Effect. daVinci-Searcher achieves stronger performance with only 2,000 training trajectories than Red-Searcher, OpenSeeker, and OpenSeeker-v2 (Chu et al., 2026; Du et al., 2026b;a), which use substantially larger posttraining datasets. Figure 3 further shows that scaling our filtered training set does not yield monotonic improvements. Increasing from 1,000 to 2,000 trajectories raises the average scores of the 35B and 4B models to 62.8% and 55.0%, respectively. Doubling the dataset to 4,000 brings little change, while 6,000 lowers the scores to 60.4% and 52.7%. The 2,000-trajectory setting achieves the best observed 35B result and is within 0.3 percentage points of the best 4B result. This practical *less is more* effect supports our compact training configuration and emphasizes the value of curated trajectories when allocating a limited supervision budget across the evaluated student models.

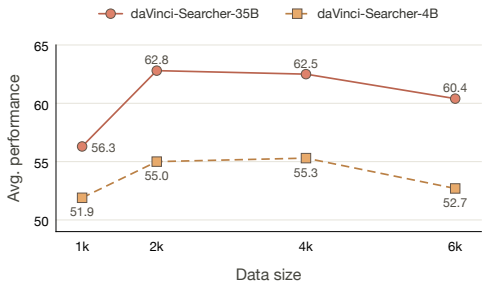

Figure 3: Average scores across training data sizes. The 4B and 35B models perform near their best with 2,000 trajectories, more data offers little gain.

Tool Call Distribution of SFT Trajectories. Tool call counts in synthesized trajectories provide an indicator of task quality: harder questions require more extensive retrieval and thus more calls. Stronger base models such as Qwen3.5 therefore motivate more challenging tasks. We compare daVinci-Searcher tasks obtained through entity scaling (high-hop) with low-hop counterparts involving fewer obfuscated entities. As shown in Figure 4, high-hop trajectories typically involve more tool calls, whereas many low-hop trajectories contain fewer than 25 calls. The distribution for S1-DR (Dong et al., 2026) is closer to that of our low-hop tasks. In a controlled comparison, regenerating S1-DR trajectories with our teacher reduces mean tool calls from 20.3 to 12.8, compared with 44.6 for high-hop and 11.7 for low-hop tasks. These results suggest that entity scaling increases retrieval demands even for a strong teacher model; more details on the experimental setup are provided in Appendix J.

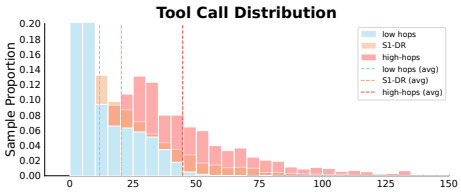

Figure 4: Tool calls across task groups. High-hop tasks involve more calls than low-hop and S1-DR tasks, indicating greater retrieval demands.

Solve Rates of Synthesized Tasks. We sample $k = 8$ trajectories for each synthesized task and measure its solve rate as the fraction of attempts that yield a correct answer. Figure 5 reveals a bimodal distribution: more than 35% of tasks have solve rates at or near zero, indicating highly challenging examples, while more than 30% have solve rates above 80%, indicating that they can be solved reliably. The remaining tasks occupy an intermediate region near the boundary of the teacher model's capabilities. This finding closely resembles the observation reported by OpenResearcher (Li et al., 2026), although we use DeepSeek-V4-Flash-0731 as a stronger teacher model to generate these trajectories.

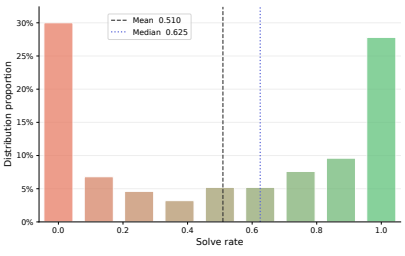

Figure 5: Synthesized task solve rates. Most tasks are either rarely or reliably solved, forming a bimodal distribution.

Reasoning Length Analysis in Synthesized Trajectories. Figure 6 (left) shows that reasoning length in correct trajectories stabilizes or decreases after an initial rise, whereas incorrect trajectories exhibit sustained growth and a sharp increase in the final 10% of steps. The cumulative distributions in the middle and right panels show the same separation between trajectories with correct and incorrect answers: the median per-trajectory mean is 304 tokens for correct trajectories versus 594 for incorrect ones, while final-step medians are 338 and 752, respectively. Incorrect trajectories also show a

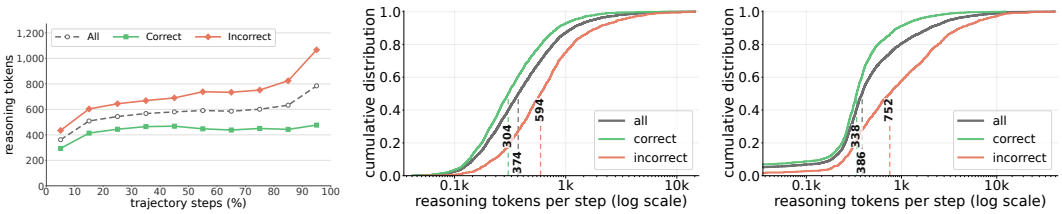

Figure 6: Per-turn reasoning tokens in synthesized training trajectories, grouped by outcome. **Left:** reasoning token trends over normalized trajectory progress. **Middle:** cumulative distribution of mean token counts per assistant step. **Right:** cumulative distribution of tokens in the final assistant step.

pronounced tail beyond 1,000 tokens. These patterns are consistent with increasing uncertainty during unsuccessful searches, particularly as the agent approaches its final answer.

Appendix B provides stage-specific distributions (Figure 12), and Appendix C reports final-turn distributions with Qwen3.5-397B-A17B as the teacher. Figure 13 further compares Qwen3 (Yang et al., 2025) and Qwen3.5 (Qwen Team, 2026) SFT checkpoints, showing that models based on Qwen3 are more prone to excessively long reasoning than the corresponding Qwen3.5 models.

5 Discussions and Analysis

5.1 Loop-Aware Trajectory Filtering

We observe *looping reasoning*: models revisit keywords, hypotheses, or intermediate conclusions without progress. Attention maps show recurring diagonal bands during looping and less regular patterns otherwise (Appendix H). An external LLM assigns a *loop score* from 0 (none) to 10 (almost entirely looping), using semantic repetition rather than length or answer correctness (Appendix N). Across 648 random reasoning segments, this score correlates strongly with per-turn reasoning length (Pearson $r = 0.887$; Figure 7), indicating a *loop-prone* region at longer reasoning lengths. We use maximum per-turn reasoning length as a proxy for loop-aware trajectory filtering.

Table 2 shows that filtering raises BC-zh accuracy from 39.1% to 60.8% and BC-en from 32.6% to 66.4%, gains of 21.7 and 33.8 percentage points. This complements the correctness and data-size analyses (Section 5.2; Figure 3): reasoning traces provide a selection signal beyond final-answer labels, supporting a focus on trajectory quality. These gains support loop-aware filtering and highlight the importance of removing unproductive reasoning patterns from SFT supervision for deep research agents.

Table 2: Loop-aware trajectory filtering ablation. BC-zh and BC accuracy rise by 21.7 and 33.8 percentage points, respectively.

	BC-zh	BC
w/ loop-aware filtering	**60.8**	**66.4**
w/o loop-aware filtering	39.1	32.6

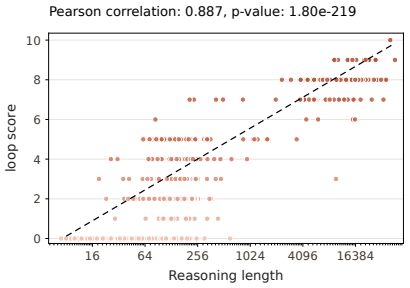

Figure 7: Per-turn reasoning length and loop score. Their strong correlation motivates loop-aware trajectory filtering.

5.2 Final-Answer Correctness as a Filtering Signal for Trajectory SFT

The offline experiments in OpenResearcher (Li et al., 2026) suggest that final-answer correctness alone may not determine a trajectory's training value. We examine whether this finding also holds for synthetic trajectories generated through interaction with the open web. Our results in Table 3 indicate that it does not hold universally. In our setting, training on correct trajectories alone and on an approximately balanced mixture of correct

Table 3: Correctness as a filtering signal. Correct and mixed trajectories achieve similar mean accuracy; incorrect trajectories alone yield lower accuracy.

Training Trajectories	BC-zh	BC-en	avg.
correct	**63.3**	64.1	**63.7**
incorrect	54.3	52.7	53.5
mixed	60.8	**66.4**	63.6

and incorrect trajectories yields nearly identical average performance across BC-zh and BC-en (63.7% and 63.6%, respectively). However, training exclusively on incorrect trajectories leads to a substantial performance drop, from 63.6% for the mixture to 53.5% across the same two benchmarks.

We hypothesize that this degradation stems from a distributional shift between correct and incorrect trajectories: when trained exclusively on incorrect trajectories, the model may learn erroneous patterns that impair downstream performance. The reasoning-token analysis in Figure 6 provides further evidence consistent with this hypothesis. In particular, the right panel shows a pronounced difference in the distribution of reasoning-token counts in the final assistant step, suggesting that trajectories with incorrect answers exhibit markedly different reasoning behavior in the final assistant step from trajectories that eventually produce a correct final answer to the research question.

5.3 Effect of Discard-all Context Management

Discard-all context management substantially improves accuracy across the BrowseComp benchmarks. As shown in Figure 9, enabling discard-all yields gains of 25.9, 21.3, and 20.0 percentage points on BC, BC-zh, and BC-plus, respectively. These results identify inference-time history management as a contributor to performance alongside task construction and loop-aware trajectory filtering. Comparisons of training data therefore need to account for the context-management policy used to evaluate the models being compared.

The accuracy gains are accompanied by longer generation times. In Figure 8, mean total generation time on BC increases from 30.9 to 69.6 minutes per task (approximately $2.25\times$), with a more pronounced long tail under discard-all. The 150-minute limit applies independently to each attempt, so the accumulated time across attempts can exceed this limit. The additional distributions in Appendix A show the same trend on BC-zh (18.8 to 38.6 minutes) and BC-plus (4.1 to 6.7 minutes). These results show an accuracy–latency trade-off. Our compact-data finding concerns the amount of SFT supervision; inference-time search remains a separate contributor to both performance and computational cost for the same trained model.

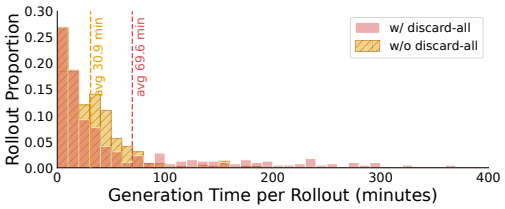

Figure 8: Discard-all more than doubles mean BC generation time, from 30.9 to 69.6 minutes per task, and produces a more pronounced long tail.

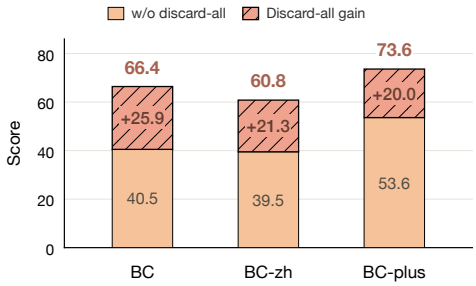

Figure 9: Discard-all consistently improves accuracy across BC, BC-zh, and BC-plus, with gains of 25.9, 21.3, and 20.0 percentage points, respectively.

6 Conclusion

We presented daVinci-Searcher, showing that strong deep research capabilities can be elicited through SFT on just 2,000 curated trajectories. Our approach targets ***data efficiency*** by jointly considering what a task demands and what its solution trajectory teaches. Verifiable multi-entity questions encourage sustained search, with the number of obfuscated entities controlling difficulty. Loop-aware trajectory filtering uses maximum per-turn reasoning length to remove demonstrations with looping reasoning while preserving tool interaction and supervision from either correct or incorrect answers. External judge evaluations and SFT ablations support this proxy for looping. The resulting 35B agent achieves 66.4% on BrowseComp, 60.8% on BrowseComp-zh, and a 62.8% macro average across six research benchmarks. Across 4B and 35B students, expanding the dataset beyond 2,000 trajectories offers limited gains, highlighting the importance of task demands and trajectory quality when allocating supervision. Context-management ablations further show that inference-time search affects both accuracy and latency. Future work can investigate how these curation principles transfer across teachers and support data-efficient learning on more open-ended research tasks.

AI USE STATEMENT

We used AI tools to polish the manuscript and AI agents to assist with part of the code development and refinement tasks. We also used DeepSeek-V4-Flash-0731 for query and agent trajectory generation, and gpt-oss-120b for external evaluation scoring and webpage summarization.

REFERENCES

Anthropic. Introducing claude 4, 2025a. URL https://www.anthropic.com/news/claude-4.

Anthropic. Introducing claude sonnet 4.5. https://www.anthropic.com/news/claude-sonnet-4-5, 2025b.

Xingyu Chen, Jiahao Xu, Tian Liang, Zhiwei He, Jianhui Pang, Dian Yu, Linfeng Song, Qiuzhi Liu, Mengfei Zhou, Zhuosheng Zhang, Rui Wang, Zhaopeng Tu, Haitao Mi, and Dong Yu. Do not think that much for 2+3=? on the overthinking of o1-like llms, 2025a. URL https://arxiv.org/abs/2412.21187.

Zijian Chen, Xueguang Ma, Shengyao Zhuang, Ping Nie, Kai Zou, Andrew Liu, Joshua Green, Kshama Patel, Ruoxi Meng, Mingyi Su, Sahel Sharifymoghaddam, Yanxi Li, Haoran Hong, Xinyu Shi, Xuye Liu, Nandan Thakur, Crystina Zhang, Luyu Gao, Wenhu Chen, and Jimmy Lin. Browsecomp-plus: A more fair and transparent evaluation benchmark of deep-research agent, 2025b. URL https://arxiv.org/abs/2508.06600.

Zheng Chu, Xiao Wang, Jack Hong, Huiming Fan, Yuqi Huang, Yue Yang, Guohai Xu, Chenxiao Zhao, Cheng Xiang, Shengchao Hu, Dongdong Kuang, Ming Liu, Bing Qin, and Xing Yu. Redsearcher: A scalable and cost-efficient framework for long-horizon search agents, 2026. URL https://arxiv.org/abs/2602.14234.

DeepSeek-AI, Aixin Liu, Aoxue Mei, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, Chen Dong, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenhao Xu, Chong Ruan, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Erhang Li, Fangqi Zhou, Fangyun Lin, Fucong Dai, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Hanwei Xu, Hao Li, Haofen Liang, Haoran Wei, Haowei Zhang, Haowen Luo, Haozhe Ji, Honghui Ding, Hongxuan Tang, Huanqi Cao, Huazuo Gao, Hui Qu, Hui Zeng, Jialiang Huang, Jiashi Li, Jiaxin Xu, Jiewen Hu, Jingchang Chen, Jingting Xiang, Jingyang Yuan, Jingyuan Cheng, Jinhua Zhu, Jun Ran, Junguang Jiang, Junjie Qiu, Junlong Li, Junxiao Song, Kai Dong, Kaige Gao, Kang Guan, Kexin Huang, Kexing Zhou, Kezhao Huang, Kuai Yu, Lean Wang, Lecong Zhang, Lei Wang, Liang Zhao, Liangsheng Yin, Lihua Guo, Lingxiao Luo, Linwang Ma, Litong Wang, Liyue Zhang, M. S. Di, M. Y Xu, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingxu Zhou, Panpan Huang, Peixin Cong, Peiyi Wang, Qiancheng Wang, Qihao Zhu, Qingyang Li, Qinyu Chen, Qiushi Du, Ruiling Xu, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, Runqiu Yin, Runxin Xu, Ruomeng Shen, Ruoyu Zhang, S. H. Liu, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shaofei Cai, Shaoyuan Chen, Shengding Hu, Shengyu Liu, Shiqiang Hu, Shirong Ma, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, Songyang Zhou, Tao Ni, Tao Yun, Tian Pei, Tian Ye, Tianyuan Yue, Wangding Zeng, Wen Liu, Wenfeng Liang, Wenjie Pang, Wenjing Luo, Wenjun Gao, Wentao Zhang, Xi Gao, Xiangwen Wang, Xiao Bi, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaokang Zhang, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xingkai Yu, Xingyou Li, Xinyu Yang, Xinyuan Li, Xu Chen, Xuecheng Su, Xuehai Pan, Xuheng Lin, Xuwei Fu, Y. Q. Wang, Yang Zhang, Yanhong Xu, Yanru Ma, Yao Li, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Qian, Yi Yu, Yichao Zhang, Yifan Ding, Yifan Shi, Yiliang Xiong, Ying He, Ying Zhou, Yinmin Zhong, Yishi Piao, Yisong Wang, Yixiao Chen, Yixuan Tan, Yixuan Wei, Yiyang Ma, Yiyuan Liu, Yonglun Yang, Yongqiang Guo, Yongtong Wu, Yu Wu, Yuan Cheng, Yuan Ou, Yuanfan Xu, Yuduan Wang, Yue Gong, Yuhan Wu, Yuheng Zou, Yukun Li, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Z. F. Wu, Z. Z. Ren, Zehua Zhao, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhibin Gou, Zhicheng Ma, Zhigang Yan, Zhihong Shao, Zhixian Huang, Zhiyu Wu, Zhuoshu Li, Zhuping Zhang, Zian Xu, Zihao Wang, Zihui Gu, Zijia Zhu, Zilin Li, Zipeng Zhang, Ziwei Xie, Ziyi Gao, Zizheng Pan, Zongqing Yao, Bei Feng, Hui Li, J. L. Cai, Jiaqi Ni, Lei Xu, Meng

Li, Ning Tian, R. J. Chen, R. L. Jin, S. S. Li, Shuang Zhou, Tianyu Sun, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xiaosha Chen, Xinnan Song, Xinyi Zhou, Y. X. Zhu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Zhen Huang, Zhipeng Xu, Zhongyu Zhang, Dongjie Ji, Jian Liang, Jianzhong Guo, Jin Chen, Leyi Xia, Miaojun Wang, Mingming Li, Peng Zhang, Ruyi Chen, Shangmian Sun, Shaoqing Wu, Shengfeng Ye, T. Wang, W. L. Xiao, Wei An, Xianzu Wang, Xiaowen Sun, Xiaoxiang Wang, Ying Tang, Yukun Zha, Zekai Zhang, Zhe Ju, Zhen Zhang, and Zihua Qu. Deepseek-v3.2: Pushing the frontier of open large language models, 2025. URL https://arxiv.org/abs/2512.02556.

DeepSeek-AI, Anyi Xu, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, Chen Dong, Chenchen Ling, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chengyu Hou, Chenhao Xu, Chenze Shao, Chong Ruan, Conner Sun, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Donghao Li, Dongjie Ji, Erhang Li, Fang Wei, Fangyun Lin, Fangzhou Yuan, Feiyu Xia, Fucong Dai, Guangbo Hao, Guanting Chen, Guoai Cao, Guolai Meng, Guowei Li, Han Yu, Han Zhang, Hanwei Xu, Hao Li, Haofen Liang, Haoling Zhang, Haoming Luo, Haoran Wei, Haotian Yuan, Haowei Zhang, Haowen Luo, Haoyu Chen, Haozhe Ji, Hengqing Zhang, Honghui Ding, Hongxuan Tang, Huanqi Cao, Huazuo Gao, Hui Qu, Hui Zeng, J Yang, JQ Zhu, Jia Luo, Jia Song, Jia Yu, Jialiang Huang, Jialu Cai, Jian Liang, Jiangting Zhou, Jiasheng Ye, Jiashi Li, Jiaxin Xu, Jiewen Hu, Jieyu Yang, Jin Chen, Jin Yan, Jingchang Chen, Jingli Zhou, Jingting Xiang, Jingyang Yuan, Jingyuan Cheng, Jingzi Zhou, Jinhua Zhu, Jiping Yu, Joseph Sun, Jun Ran, Junguang Jiang, Junjie Qiu, Junlong Li, Junmin Zheng, Junxiao Song, Kai Dong, Kaige Gao, Kang Guan, Kexing Zhou, Kezhao Huang, Kuai Yu, Lean Wang, Lecong Zhang, Lei Wang, Leyi Xia, Li Zhang, Liang Zhao, Lihua Guo, Lingxiao Luo, Linwang Ma, Linyan Zhu, Litong Wang, Liyu Cai, Liyue Zhang, Longhao Chen, MS Di, MY Xu, Max Mei, Miaojun Wang, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingming Li, Mingxu Zhou, Minmin Han, Ning Wang, Panpan Huang, Panpan Wang, Peixin Cong, Peiyi Wang, Peng Zhang, Qiancheng Wang, Qihao Zhu, Qingyang Li, Qinyu Chen, Qiushi Du, Qiwei Jiang, Rui Tian, Ruifan Xu, Ruijie Lu, Ruiling Xu, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, Runqian Chen, Runqiu Yin, Runxin Xu, Ruomeng Shen, Ruoyu Zhang, Ruyi Chen, SH Liu, Shanghao Lu, Shangmian Sun, Shangyan Zhou, Shanhuang Chen, Shaofei Cai, Shaoheng Nie, Shaoqing Wu, Shaoyuan Chen, Shengding Hu, Shengyu Liu, Shiqiang Hu, Shirong Ma, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, Shuying Yu, Songyang Zhou, Tao Ni, Tao Yun, Tian Jin, Tian Pei, Tian Ye, Tianle Lin, Tianran Ji, Tianyi Cui, Tianyuan Yue, Tingting Yu, Tun Wang, W Zhang, WL Xiao, Wangding Zeng, Wei An, Weilin Zhao, Wen Liu, Wenfeng Liang, Wenjie Pang, Wenjing Luo, Wenjing Yao, Wenjun Gao, Wenkai Yang, Wenlve Huang, Wenqing Hou, Wentao Zhang, Wenting Ma, Xi Gao, Xiang He, Xiangwen Wang, Xianzu Wang, Xiao Bi, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaokang Zhang, Xiaotao Nie, Xiaowen Sun, Xiaoxiang Wang, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xingchen Liu, Xingkai Yu, Xingyou Li, Xinyu Yang, Xinyu Zhang, Xu Chen, Xuanyu Wang, Xuecheng Su, Xueyin Chen, Xuheng Lin, Xuwei Fu, YC Yan, YQ Wang, YW Ma, Yanfeng Luo, Yang Zhang, Yanhong Xu, Yanru Ma, Yanwen Huang, Yao Li, Yao Li, Yao Xu, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Qian, Yi Shao, Yi Yu, Yichao Zhang, Yifan Ding, Yifan Shi, Yijia Wu, Yiliang Xiong, Yiling Ma, Ying He, Ying Tang, Ying Zhou, Yingjia Luo, Yinmin Zhong, Yishi Piao, Yisong Wang, Yixiang Zhang, Yixiao Chen, Yixuan Tan, Yixuan Wei, Yiyang Ma, Yiyuan Liu, Yonglun Yang, Yongqiang Guo, Yongtong Wu, Yu Wu, YuKun Li, Yuan Cheng, Yuan Ou, Yuanfan Xu, Yuanhao Li, Yuduan Wang, Yuehan Yang, Yuer Xu, Yuhan Wu, Yuhao Meng, Yuheng Zou, Yukun Zha, Yunfan Xiong, Yupeng Chen, Yuping Lin, Yuqian Cao, Yuqian Wang, Yushun Zhang, Yuting Yan, Yutong Lin, Yuxian Gu, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuxuan Zhou, Yuyang Zhou, Yuzhen Huang, ZF Wu, Zehao Wang, Zehua Zhao, Zehui Ren, Zekai Zhang, Zhangli Sha, Zhe Fu, Zhe Ju, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zheren Gao, Zhewen Hao, Zhibin Gou, Zhicheng Ma, Zhigang Yan, Zhihong Shao, Zhixian Huang, Zhixuan Chen, Zhiyu Wu, Zhizhou Ren, Zhongyu Wu, Zhuoshu Li, Zhuping Zhang, Zian Xu, Zihao Wang, Zihua Qu, Zihui Gu, Zijia Zhu, Zilin Li, Zipeng Zhang, Ziwei Xie, Ziyi Gao, Ziyi Wan, Zizheng Pan, and Zongqing Yao. Deepseek-v4: Towards highly efficient million-token context intelligence, 2026. URL https://arxiv.org/abs/2606.19348.

Yao Dong, Xinglin Xiao, Liwei Dong, Xinlong Jin, Zhengbo Li, Heng Zhang, Duyun Wang, and Nan Xu. S1-deepresearch: Beyond search, toward real-world long-horizon research agents, 2026. URL https://arxiv.org/abs/2606.15367.

Yuwen Du, Rui Ye, Shuo Tang, Keduan Huang, Xinyu Zhu, Yuzhu Cai, and Siheng Chen. Openseeker-v2: Pushing the limits of search agents with informative and high-difficulty trajectories, 2026a. URL https://arxiv.org/abs/2605.04036.

Yuwen Du, Rui Ye, Shuo Tang, Xinyu Zhu, Yijun Lu, Yuzhu Cai, and Siheng Chen. Openseeker: Democratizing frontier search agents by fully open-sourcing training data, 2026b. URL https://arxiv.org/abs/2603.15594.

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

Bairu Hou, Yang Zhang, Jiabao Ji, Yujian Liu, Kaizhi Qian, Jacob Andreas, and Shiyu Chang. Thinkprune: Pruning long chain-of-thought of llms via reinforcement learning, 2025. URL https://arxiv.org/abs/2504.01296.

Ziyang Huang, Haolin Ren, Xiaowei Yuan, Jiawei Wang, Zhongtao Jiang, Kun Xu, Shizhu He, Jun Zhao, and Kang Liu. Wideseek: Advancing wide research via multi-agent scaling, 2026. URL https://arxiv.org/abs/2602.02636.

Mohan Jiang, Dayuan Fu, Junhao Shi, Ji Zeng, Weiye Si, Keyu Li, Xuefeng Li, Yang Xiao, Wenjie Li, Dequan Wang, and Pengfei Liu. davinci-agency: Unlocking long-horizon agency data-efficiently, 2026. URL https://arxiv.org/abs/2602.02619.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2017. URL https://arxiv.org/abs/1412.6980.

Kuan Li, Zhongwang Zhang, Huifeng Yin, Rui Ye, Yida Zhao, Liwen Zhang, Litu Ou, Dingchu Zhang, Xixi Wu, Jialong Wu, Xinyu Wang, Zile Qiao, Zhen Zhang, Yong Jiang, Pengjun Xie, Fei Huang, and Jingren Zhou. Websailor-v2: Bridging the chasm to proprietary agents via synthetic data and scalable reinforcement learning, 2025a. URL https://arxiv.org/abs/2509.13305.

Kuan Li, Zhongwang Zhang, Huifeng Yin, Liwen Zhang, Litu Ou, Jialong Wu, Wenbiao Yin, Baixuan Li, Zhengwei Tao, Xinyu Wang, Weizhou Shen, Junkai Zhang, Dingchu Zhang, Xixi Wu, Yong Jiang, Ming Yan, Pengjun Xie, Fei Huang, and Jingren Zhou. Websailor: Navigating super-human reasoning for web agent, 2025b. URL https://arxiv.org/abs/2507.02592.

Xuefeng Li, Haoyang Zou, and Pengfei Liu. Limr: Less is more for rl scaling, 2025c. URL https://arxiv.org/abs/2502.11886.

Zhuofeng Li, Dongfu Jiang, Xueguang Ma, Haoxiang Zhang, Ping Nie, Yuyu Zhang, Kai Zou, Jianwen Xie, Yu Zhang, and Wenhu Chen. Openresearcher: A fully open pipeline for long-horizon deep research trajectory synthesis, 2026. URL https://arxiv.org/abs/2603.20278.

Wei Liu, Weihao Zeng, Keqing He, Yong Jiang, and Junxian He. What makes good data for alignment? a comprehensive study of automatic data selection in instruction tuning, 2024. URL https://arxiv.org/abs/2312.15685.

Yijun Lu, Rui Ye, Jiajun Wang, Yuwen Du, Tian Jin, Songhua Liu, and Siheng Chen. Abseeker: Training long-horizon search agents via answer-backtracked credit assignment, 2026. URL https://arxiv.org/abs/2608.05102.

Grégoire Mialon, Clémentine Fourrier, Craig Swift, Thomas Wolf, Yann LeCun, and Thomas Scialom. Gaia: a benchmark for general ai assistants, 2023. URL https://arxiv.org/abs/2311.12983.

Niklas Muennighoff, Zitong Yang, Weijia Shi, Xiang Lisa Li, Li Fei-Fei, Hannaneh Hajishirzi, Luke Zettlemoyer, Percy Liang, Emmanuel Candès, and Tatsunori Hashimoto. s1: Simple test-time scaling, 2025. URL https://arxiv.org/abs/2501.19393.

OpenAI. Deep research system card, 2025. URL https://cdn.openai.com/deep-research-system-card.pdf.

OpenAI. gpt-oss-120b & gpt-oss-20b Model Card, 2025a. URL https://arxiv.org/abs/2508.10925.

OpenAI. Openai o3 and o4-mini system card, April 2025b. URL https://openai.com/index/o3-o4-mini-system-card/.

Long Phan, Alice Gatti, Ziwen Han, Nathaniel Li, Josephina Hu, Hugh Zhang, Chen Bo Calvin Zhang, Mohamed Shaaban, John Ling, Sean Shi, Michael Choi, Anish Agrawal, Arnav Chopra, Adam Khoja, Ryan Kim, Richard Ren, Jason Hausenloy, Oliver Zhang, Mantas Mazeika, Dmitry Dodonov, Tung Nguyen, Jaeho Lee, Daron Anderson, Mikhail Doroshenko, Alun Cennyth Stokes, Mobeen Mahmood, Oleksandr Pokutnyi, Oleg Iskra, Jessica P. Wang, John-Clark Levin, Mstyslav Kazakov, Fiona Feng, Steven Y. Feng, Haoran Zhao, Michael Yu, Varun Gangal, Chelsea Zou, Zihan Wang, Serguei Popov, Robert Gerbicz, Geoff Galgon, Johannes Schmitt, Will Yeadon, Yongki Lee, Scott Sauers, Alvaro Sanchez, Fabian Giska, Marc Roth, Søren Riis, Saiteja Utpala, Noah Burns, Gashaw M. Goshu, Mohinder Maheshbhai Naiya, Chidozie Agu, Zachary Giboney, Antrell Cheatom, Francesco Fournier-Facio, Sarah-Jane Crowson, Lennart Finke, Zerui Cheng, Jennifer Zampese, Ryan G. Hoerr, Mark Nandor, Hyunwoo Park, Tim Gehrunger, Jiaqi Cai, Ben McCarty, Alexis C Garretson, Edwin Taylor, Damien Sileo, Qiuyu Ren, Usman Qazi, Lianghui Li, Jungbae Nam, John B. Wydallis, Pavel Arkhipov, Jack Wei Lun Shi, Aras Bacho, Chris G. Willcocks, Hangrui Cao, Sumeet Motwani, Emily de Oliveira Santos, Johannes Veith, Edward Vendrow, Doru Cojoc, Kengo Zenitani, Joshua Robinson, Longke Tang, Yuqi Li, Joshua Vendrow, Natanael Wildner Fraga, Vladyslav Kuchkin, Andrey Pupasov Maksimov, Pierre Marion, Denis Efremov, Jayson Lynch, Kaiqu Liang, Aleksandar Mikov, Andrew Gritsevskiy, Julien Guillod, Gözdenur Demir, Dakotah Martinez, Ben Pageler, Kevin Zhou, Saeed Soori, Ori Press, Henry Tang, Paolo Rissone, Sean R. Green, Lina Brüssel, Moon Twayana, Aymeric Dieuleveut, Joseph Marvin Imperial, Ameya Prabhu, Jinzhou Yang, Nick Crispino, Arun Rao, Dimitri Zvonkine, Gabriel Loiseau, Mikhail Kalinin, Marco Lukas, Ciprian Manolescu, Nate Stambaugh, Subrata Mishra, Tad Hogg, Carlo Bosio, Brian P Coppola, Julian Salazar, Jaehyeok Jin, Rafael Sayous, Stefan Ivanov, Philippe Schwaller, Shaipranesh Senthilkuma, Andres M Bran, Andres Algaba, Kelsey Van den Houte, Lynn Van Der Sypt, Brecht Verbeken, David Noever, Alexei Kopylov, Benjamin Myklebust, Bikun Li, Lisa Schut, Evgenii Zheltonozhskii, Qiaochu Yuan, Derek Lim, Richard Stanley, Tong Yang, John Maar, Julian Wykowski, Martí Oller, Anmol Sahu, Cesare Giulio Ardito, Yuzheng Hu, Ariel Ghislain Kemogne Kamdoum, Alvin Jin, Tobias Garcia Vilchis, Yuexuan Zu, Martin Lackner, James Koppel, Gongbo Sun, Daniil S. Antonenko, Steffi Chern, Bingchen Zhao, Pierrot Arsene, Joseph M Cavanagh, Daofeng Li, Jiawei Shen, Donato Crisostomi, Wenjin Zhang, Ali Dehghan, Sergey Ivanov, David Perrella, Nurdin Kaparov, Allen Zang, Ilia Sucholutsky, Arina Kharlamova,

Daniil Orel, Vladislav Poritski, Shalev Ben-David, Zachary Berger, Parker Whitfill, Michael Foster, Daniel Munro, Linh Ho, Shankar Sivarajan, Dan Bar Hava, Aleksey Kuchkin, David Holmes, Alexandra Rodriguez-Romero, Frank Sommerhage, Anji Zhang, Richard Moat, Keith Schneider, Zakayo Kazibwe, Don Clarke, Dae Hyun Kim, Felipe Meneguitti Dias, Sara Fish, Veit Elser, Tobias Kreiman, Victor Efren Guadarrama Vilchis, Immo Klose, Ujjwala Anantheswaran, Adam Zweiger, Kaivalya Rawal, Jeffery Li, Jeremy Nguyen, Nicolas Daans, Haline Heidinger, Maksim Radionov, Václav Rozhoň, Vincent Ginis, Christian Stump, Niv Cohen, Rafał Poświata, Josef Tkadlec, Alan Goldfarb, Chenguang Wang, Piotr Padlewski, Stanislaw Barzowski, Kyle Montgomery, Ryan Stendall, Jamie Tucker-Foltz, Jack Stade, T. Ryan Rogers, Tom Goertzen, Declan Grabb, Abhishek Shukla, Alan Givré, John Arnold Ambay, Archan Sen, Muhammad Fayez Aziz, Mark H Inlow, Hao He, Ling Zhang, Younesse Kaddar, Ivar Ängquist, Yanxu Chen, Harrison K Wang, Kalyan Ramakrishnan, Elliott Thornley, Antonio Terpin, Hailey Schoelkopf, Eric Zheng, Avishy Carmi, Ethan D. L. Brown, Kelin Zhu, Max Bartolo, Richard Wheeler, Martin Stehberger, Peter Bradshaw, JP Heimonen, Kaustubh Sridhar, Ido Akov, Jennifer Sandlin, Yury Makarychev, Joanna Tam, Hieu Hoang, David M. Cunningham, Vladimir Goryachev, Demosthenes Patramanis, Michael Krause, Andrew Redenti, David Aldous, Jesyin Lai, Shannon Coleman, Jiangnan Xu, Sangwon Lee, Ilias Magoulas, Sandy Zhao, Ning Tang, Michael K. Cohen, Orr Paradise, Jan Hendrik Kirchner, Maksym Ovchynnikov, Jason O. Matos, Adithya Shenoy, Michael Wang, Yuzhou Nie, Anna Sztyber-Betley, Paolo Faraboschi, Robin Riblet, Jonathan Crozier, Shiv Halasyamani, Shreyas Verma, Prashant Joshi, Eli Meril, Ziqiao Ma, Jérémy Andréoletti, Raghav Singhal, Jacob Platnick, Volodymyr Nevirkovets, Luke Basler, Alexander Ivanov, Seri Khoury, Nils Gustafsson, Marco Piccardo, Hamid Mostaghimi, Qijia Chen, Virendra Singh, Tran Quoc Khánh, Paul Rosu, Hannah Szlyk, Zachary Brown, Himanshu Narayan, Aline Menezes, Jonathan Roberts, William Alley, Kunyang Sun, Arkil Patel, Max Lamparth, Anka Reuel, Linwei Xin, Hanmeng Xu, Jacob Loader, Freddie Martin, Zixuan Wang, Andrea Achilleos, Thomas Preu, Tomek Korbak, Ida Bosio, Fereshteh Kazemi, Ziye Chen, Biró Bálint, Eve J. Y. Lo, Jiaqi Wang, Maria Inês S. Nunes, Jeremiah Milbauer, M Saiful Bari, Zihao Wang, Behzad Ansarinejad, Yewen Sun, Stephane Durand, Hossam Elgnainy, Guillaume Douville, Daniel Tordera, George Balabanian, Hew Wolff, Lynna Kvistad, Hsiaoyun Milliron, Ahmad Sakor, Murat Eron, Andrew Favre D. O., Shailesh Shah, Xiaoxiang Zhou, Firuz Kamalov, Sherwin Abdoli, Tim Santens, Shaul Barkan, Allison Tee, Robin Zhang, Alessandro Tomasiello, G. Bruno De Luca, Shi-Zhuo Looi, Vinh-Kha Le, Noam Kolt, Jiayi Pan, Emma Rodman, Jacob Drori, Carl J Fossum, Niklas Muennighoff, Milind Jagota, Ronak Pradeep, Honglu Fan, Jonathan Eicher, Michael Chen, Kushal Thaman, William Merrill, Moritz Firsching, Carter Harris, Stefan Ciobâcă, Jason Gross, Rohan Pandey, Ilya Gusev, Adam Jones, Shashank Agnihotri, Pavel Zhelnov, Mohammadreza Mofayezi, Alexander Piperski, David K. Zhang, Kostiantyn Dobarskyi, Roman Leventov, Ignat Soroko, Joshua Duersch, Vage Taamazyan, Andrew Ho, Wenjie Ma, William Held, Ruicheng Xian, Armel Randy Zebaze, Mohanad Mohamed, Julian Noah Leser, Michelle X Yuan, Laila Yacar, Johannes Lengler, Katarzyna Olszewska, Claudio Di Fratta, Edson Oliveira, Joseph W. Jackson, Andy Zou, Muthu Chidambaram, Timothy Manik, Hector Haffenden, Dashiell Stander, Ali Dasouqi, Alexander Shen, Bita Golshani, David Stap, Egor Kretov, Mikalai Uzhou, Alina Borisovna Zhidkovskaya, Nick Winter, Miguel Orbegozo Rodriguez, Robert Lauff, Dustin Wehr, Colin Tang, Zaki Hossain, Shaun Phillips, Fortuna Samuele, Fredrik Ekström, Angela Hammon, Oam Patel, Faraz Farhidi, George Medley, Forough Mohammadzadeh, Madellene Peñaflor, Haile Kassahun, Alena Friedrich, Rayner Hernandez Perez, Daniel Pyda, Taom Sakal, Omkar Dhamane, Ali Khajegili Mirabadi, Eric Hallman, Kenchi Okutsu, Mike Battaglia, Mohammad Maghsoudimehrabani, Alon Amit, Dave Hulbert, Roberto Pereira, Simon Weber, Handoko, Anton Peristyy, Stephen Malina, Mustafa Mehkary, Rami Aly, Frank Reidegeld, Anna-Katharina Dick, Cary Friday, Mukhwinder Singh, Hassan Shapourian, Wanyoung Kim, Mariana Costa, Hubeyb Gurdogan, Harsh Kumar, Chiara Ceconello, Chao Zhuang, Haon Park, Micah Carroll, Andrew R. Tawfeek, Stefan Steinerberger, Daattavya Aggarwal, Michael Kirchhof, Linjie Dai, Evan Kim, Johan Ferret, Jainam Shah, Yuzhou Wang, Minghao Yan, Krzysztof Burdzy, Lixin Zhang, Antonio Franca, Diana T. Pham, Kang Yong Loh, Joshua Robinson, Abram Jackson, Paolo Giordano, Philipp Petersen, Adrian Cosma, Jesus Colino, Colin White, Jacob Votava, Vladimir Vinnikov, Ethan Delaney, Petr Spelda, Vit Stritecky, Syed M. Shahid, Jean-Christophe Mourrat, Lavr Vetoshkin, Koen Sponselee, Renas Bacho, Zheng-Xin Yong, Florencia de la Rosa, Nathan Cho, Xiuyu Li, Guillaume Malod, Orion Weller, Guglielmo Albani, Leon Lang, Julien Laurendeau, Dmitry Kazakov, Fatimah Adesanya, Julien Portier, Lawrence Hollom, Victor Souza, Yuchen Anna Zhou, Julien Degorre, Yiğit Yalın, Gbenga Daniel Obikoya, Rai, Filippo Bigi, M. C. Boscá, Oleg Shumar, Kaniuar Bacho, Gabriel Recchia, Mara Popescu, Nikita Shulga, Ngefor Mildred Tanwie,

Thomas C. H. Lux, Ben Rank, Colin Ni, Matthew Brooks, Alesia Yakimchyk, Huanxu, Liu, Stefano Cavalleri, Olle Häggström, Emil Verkama, Joshua Newbould, Hans Gundlach, Leonor Brito-Santana, Brian Amaro, Vivek Vajipey, Rynaa Grover, Ting Wang, Yosi Kratish, Wen-Ding Li, Sivakanth Gopi, Andrea Caciolai, Christian Schroeder de Witt, Pablo Hernández-Cámara, Emanuele Rodolà, Jules Robins, Dominic Williamson, Vincent Cheng, Brad Raynor, Hao Qi, Ben Segev, Jingxuan Fan, Sarah Martinson, Erik Y. Wang, Kaylie Hausknecht, Michael P. Brenner, Mao Mao, Christoph Demian, Peyman Kassani, Xinyu Zhang, David Avagian, Eshawn Jessica Scipio, Alon Ragoler, Justin Tan, Blake Sims, Rebeka Plecnik, Aaron Kirtland, Omer Faruk Bodur, D. P. Shinde, Yan Carlos Leyva Labrador, Zahra Adoul, Mohamed Zekry, Ali Karakoc, Tania C. B. Santos, Samir Shamseldeen, Loukmane Karim, Anna Liakhovitskaia, Nate Resman, Nicholas Farina, Juan Carlos Gonzalez, Gabe Maayan, Earth Anderson, Rodrigo De Oliveira Pena, Elizabeth Kelley, Hodjat Mariji, Rasoul Pouriamanesh, Wentao Wu, Ross Finocchio, Ismail Alarab, Joshua Cole, Danyelle Ferreira, Bryan Johnson, Mohammad Safdari, Liangti Dai, Siriphan Arthornthurasuk, Isaac C. McAlister, Alejandro José Moyano, Alexey Pronin, Jing Fan, Angel Ramirez-Trinidad, Yana Malysheva, Daphiny Pottmaier, Omid Taheri, Stanley Stepanic, Samuel Perry, Luke Askew, Raúl Adrián Huerta Rodríguez, Ali M. R. Minissi, Ricardo Lorena, Krishnamurthy Iyer, Arshad Anil Fasiludeen, Ronald Clark, Josh Ducey, Matheus Piza, Maja Somrak, Eric Vergo, Juehang Qin, Benjámin Borbás, Eric Chu, Jack Lindsey, Antoine Jallon, I. M. J. McInnis, Evan Chen, Avi Semler, Luk Gloor, Tej Shah, Marc Carauleanu, Pascal Lauer, Tran Đuc Huy, Hossein Shahrtash, Emilien Duc, Lukas Lewark, Assaf Brown, Samuel Albanie, Brian Weber, Warren S. Vaz, Pierre Clavier, Yiyang Fan, Gabriel Poesia Reis e Silva, Long, Lian, Marcus Abramovitch, Xi Jiang, Sandra Mendoza, Murat Islam, Juan Gonzalez, Vasilios Mavroudis, Justin Xu, Pawan Kumar, Laxman Prasad Goswami, Daniel Bugas, Nasser Heydari, Ferenc Jeanplong, Thorben Jansen, Antonella Pinto, Archimedes Apronti, Abdallah Galal, Ng Ze-An, Ankit Singh, Tong Jiang, Joan of Arc Xavier, Kanu Priya Agarwal, Mohammed Berkani, Gang Zhang, Zhehang Du, Benedito Alves de Oliveira Junior, Dmitry Malishev, Nicolas Remy, Taylor D. Hartman, Tim Tarver, Stephen Mensah, Gautier Abou Loume, Wiktor Morak, Farzad Habibi, Sarah Hoback, Will Cai, Javier Gimenez, Roselynn Grace Montecillo, Jakub Łucki, Russell Campbell, Asankhaya Sharma, Khalida Meer, Shreen Gul, Daniel Espinosa Gonzalez, Xavier Alapont, Alex Hoover, Gunjan Chhablani, Freddie Vargus, Arunim Agarwal, Yibo Jiang, Deepakkumar Patil, David Outevsky, Kevin Joseph Scaria, Rajat Maheshwari, Abdelkader Dendane, Priti Shukla, Ashley Cartwright, Sergei Bogdanov, Niels Mündler, Sören Möller, Luca Arnaboldi, Kunvar Thaman, Muhammad Rehan Siddiqi, Prajvi Saxena, Himanshu Gupta, Tony Fruhauff, Glen Sherman, Mátyás Vincze, Siranut Usawasutsakorn, Dylan Ler, Anil Radhakrishnan, Innocent Enyekwe, Sk Md Salauddin, Jiang Muzhen, Aleksandr Maksapetyan, Vivien Rossbach, Chris Harjadi, Mohsen Bahaloohoreh, Claire Sparrow, Jasdeep Sidhu, Sam Ali, Song Bian, John Lai, Eric Singer, Justine Leon Uro, Greg Bateman, Mohamed Sayed, Ahmed Menshawy, Darling Duclosel, Dario Bezzi, Yashaswini Jain, Ashley Aaron, Murat Tiryakioglu, Sheeshram Siddh, Keith Krenek, Imad Ali Shah, Jun Jin, Scott Creighton, Denis Peskoff, Zienab EL-Wasif, Ragavendran P V, Michael Richmond, Joseph McGowan, Tejal Patwardhan, Hao-Yu Sun, Ting Sun, Nikola Zubić, Samuele Sala, Stephen Ebert, Jean Kaddour, Manuel Schottdorf, Dianzhuo Wang, Gerol Petruzella, Alex Meiburg, Tilen Medved, Ali ElSheikh, S Ashwin Hebbar, Lorenzo Vaquero, Xianjun Yang, Jason Poulos, Vilém Zouhar, Sergey Bogdanik, Mingfang Zhang, Jorge Sanz-Ros, David Anugraha, Yinwei Dai, Anh N. Nhu, Xue Wang, Ali Anil Demircali, Zhibai Jia, Yuyin Zhou, Juncheng Wu, Mike He, Nitin Chandok, Aarush Sinha, Gaoxiang Luo, Long Le, Mickaël Noyé, Michał Perełkiewicz, Ioannis Pantidis, Tianbo Qi, Soham Sachin Purohit, Letitia Parcalabescu, Thai-Hoa Nguyen, Genta Indra Winata, Edoardo M. Ponti, Hanchen Li, Kaustubh Dhole, Jongee Park, Dario Abbondanza, Yuanli Wang, Anupam Nayak, Diogo M. Caetano, Antonio A. W. L. Wong, Maria del Rio-Chanona, Dániel Kondor, Pieter Francois, Ed Chalstrey, Jakob Zsambok, Dan Hoyer, Jenny Reddish, Jakob Hauser, Francisco-Javier Rodrigo-Ginés, Suchandra Datta, Maxwell Shepherd, Thom Kamphuis, Qizheng Zhang, Hyunjun Kim, Ruiji Sun, Jianzhu Yao, Franck Dernoncourt, Satyapriya Krishna, Sina Rismanchian, Bonan Pu, Francesco Pinto, Yingheng Wang, Kumar Shridhar, Kalon J. Overholt, Glib Briia, Hieu Nguyen, David, Soler Bartomeu, Tony CY Pang, Adam Wecker, Yifan Xiong, Fanfei Li, Lukas S. Huber, Joshua Jaeger, Romano De Maddalena, Xing Han Lù, Yuhui Zhang, Claas Beger, Patrick Tser Jern Kon, Sean Li, Vivek Sanker, Ming Yin, Yihao Liang, Xinlu Zhang, Ankit Agrawal, Li S. Yifei, Zechen Zhang, Mu Cai, Yasin Sonmez, Costin Cozianu, Changhao Li, Alex Slen, Shoubin Yu, Hyun Kyu Park, Gabriele Sarti, Marcin Briański, Alessandro Stolfo, Truong An Nguyen, Mike Zhang, Yotam Perlitz, Jose Hernandez-Orallo, Runjia Li, Amin Shabani, Felix Juefei-Xu, Shikhar Dhingra, Orr Zohar, My Chiffon Nguyen, Alexander Pondaven,

Abdurrahim Yilmaz, Xuandong Zhao, Chuanyang Jin, Muyan Jiang, Stefan Todoran, Xinyao Han, Jules Kreuer, Brian Rabern, Anna Plassart, Martino Maggetti, Luther Yap, Robert Geirhos, Jonathon Kean, Dingsu Wang, Sina Mollaei, Chenkai Sun, Yifan Yin, Shiqi Wang, Rui Li, Yaowen Chang, Anjiang Wei, Alice Bizeul, Xiaohan Wang, Alexandre Oliveira Arrais, Kushin Mukherjee, Jorge Chamorro-Padial, Jiachen Liu, Xingyu Qu, Junyi Guan, Adam Bouyamourn, Shuyu Wu, Martyna Plomecka, Junda Chen, Mengze Tang, Jiaqi Deng, Shreyas Subramanian, Haocheng Xi, Haoxuan Chen, Weizhi Zhang, Yinuo Ren, Haoqin Tu, Sejong Kim, Yushun Chen, Sara Vera Marjanović, Junwoo Ha, Grzegorz Luczyna, Jeff J. Ma, Zewen Shen, Dawn Song, Cedegao E. Zhang, Zhun Wang, Gaël Gendron, Yunze Xiao, Leo Smucker, Erica Weng, Kwok Hao Lee, Zhe Ye, Stefano Ermon, Ignacio D. Lopez-Miguel, Theo Knights, Anthony Gitter, Namkyu Park, Boyi Wei, Hongzheng Chen, Kunal Pai, Ahmed Elkhanany, Han Lin, Philipp D. Siedler, Jichao Fang, Ritwik Mishra, Károly Zsolnai-Fehér, Xilin Jiang, Shadab Khan, Jun Yuan, Rishab Kumar Jain, Xi Lin, Mike Peterson, Zhe Wang, Aditya Malusare, Maosen Tang, Isha Gupta, Ivan Fosin, Timothy Kang, Barbara Dworakowska, Kazuki Matsumoto, Guangyao Zheng, Gerben Sewuster, Jorge Pretel Villanueva, Ivan Rannev, Igor Chernyavsky, Jiale Chen, Deepayan Banik, Ben Racz, Wenchao Dong, Jianxin Wang, Laila Bashmal, Duarte V. Gonçalves, Wei Hu, Kaushik Bar, Ondrej Bohdal, Atharv Singh Patlan, Shehzaad Dhuliawala, Caroline Geirhos, Julien Wist, Yuval Kansal, Bingsen Chen, Kutay Tire, Atak Talay Yücel, Brandon Christof, Veerupaksh Singla, Zijian Song, Sanxing Chen, Jiaxin Ge, Kaustubh Ponkshe, Isaac Park, Tianneng Shi, Martin Q. Ma, Joshua Mak, Sherwin Lai, Antoine Moulin, Zhuo Cheng, Zhanda Zhu, Ziyi Zhang, Vaidehi Patil, Ketan Jha, Qiutong Men, Jiaxuan Wu, Tianchi Zhang, Bruno Hebling Vieira, Alham Fikri Aji, Jae-Won Chung, Mohammed Mahfoud, Ha Thi Hoang, Marc Sperzel, Wei Hao, Kristof Meding, Sihan Xu, Vassilis Kostakos, Davide Manini, Yueying Liu, Christopher Toukmaji, Jay Paek, Eunmi Yu, Arif Engin Demircali, Zhiyi Sun, Ivan Dewerpe, Hongsen Qin, Roman Pflugfelder, James Bailey, Johnathan Morris, Ville Heilala, Sybille Rosset, Zishun Yu, Peter E. Chen, Woongyeong Yeo, Eeshaan Jain, Ryan Yang, Sreekar Chigurupati, Julia Chernyavsky, Sai Prajwal Reddy, Subhashini Venugopalan, Hunar Batra, Core Francisco Park, Hieu Tran, Guilherme Maximiano, Genghan Zhang, Yizhuo Liang, Hu Shiyu, Rongwu Xu, Rui Pan, Siddharth Suresh, Ziqi Liu, Samaksh Gulati, Songyang Zhang, Peter Turchin, Christopher W. Bartlett, Christopher R. Scotese, Phuong M. Cao, Ben Wu, Jacek Karwowski, Davide Scaramuzza, Aakaash Nattanmai, Gordon McKellips, Anish Cheraku, Asim Suhail, Ethan Luo, Marvin Deng, Jason Luo, Ashley Zhang, Kavin Jindel, Jay Paek, Kasper Halevy, Allen Baranov, Michael Liu, Advaith Avadhanam, David Zhang, Vincent Cheng, Brad Ma, Evan Fu, Liam Do, Joshua Lass, Hubert Yang, Surya Sunkari, Vishruth Bharath, Violet Ai, James Leung, Rishit Agrawal, Alan Zhou, Kevin Chen, Tejas Kalpathi, Ziqi Xu, Gavin Wang, Tyler Xiao, Erik Maung, Sam Lee, Ryan Yang, Roy Yue, Ben Zhao, Julia Yoon, Sunny Sun, Aryan Singh, Ethan Luo, Clark Peng, Tyler Osbey, Taozhi Wang, Daryl Echeazu, Hubert Yang, Timothy Wu, Spandan Patel, Vidhi Kulkarni, Vijaykaarti Sundarapandiyan, Ashley Zhang, Andrew Le, Zafir Nasim, Srikar Yalam, Ritesh Kasamsetty, Soham Samal, Hubert Yang, David Sun, Nihar Shah, Abhijeet Saha, Alex Zhang, Leon Nguyen, Laasya Nagumalli, Kaixin Wang, Alan Zhou, Aidan Wu, Jason Luo, Anwith Telluri, Steven Dillmann, Zhengxiang Wang, Junyu Luo, Hugo Lunn, Artem Gazizov, Haitz Sáez de Ocáriz Borde, Ivan Trus, Morgan Hervault, Zheyu Zhang, Bo Chen, Yuchen Wu, Christopher J. Cordier, Gün Kaynar, Cansin Ayvaz, Polina Avdiunina, Johannes Brust, Xingjian Diao, K. D. Meaney, Yifan Gu, Chenyu Wang, Chenzhuo Dong, William Wright, Simon Brave, Owen Root, Jiayuan Liu, Chow Chun Lok, Tianqin Li, Shiyi Du, Dailan He, Lufeiya Liu, Sina Jamalzadegan, Anil Ramakrishna, Xuanqing Xu, Xin Qing, Xin Luo, Wenkai Li, Shi Bo, Filipp Gusev, Maximos Skandalis, Desheng Ma, Chunhui Zhang, Haoran Qiu, Allen G Hart, Rickard Brüel Gabrielsson, Ido Akov, Artem Lukoianov, Summer Yue, Alexandr Wang, and Dan Hendrycks. Humanity's last exam, 2026. URL https://arxiv.org/abs/2501.14249.

Qwen Team. Qwen3.5: Towards native multimodal agents, February 2026. URL https://qwen.ai/blog?id=qwen3.5.

Adam Roberts, Colin Raffel, and Noam Shazeer. How much knowledge can you pack into the parameters of a language model?, 2020. URL https://arxiv.org/abs/2002.08910.

Aaditya Singh, Adam Fry, Adam Perelman, Adam Tart, Adi Ganesh, Ahmed El-Kishky, Aidan McLaughlin, Aiden Low, AJ Ostrow, Akhila Ananthram, Akshay Nathan, Alan Luo, Alec Helyar, Aleksander Madry, Aleksandr Efremov, Aleksandra Spyra, Alex Baker-Whitcomb, Alex Beutel, Alex Karpenko, Alex Makelov, Alex Neitz, Alex Wei, Alexandra Barr, Alexandre Kirchmeyer, Alexey Ivanov, Alexi Christakis, Alistair Gillespie, Allison Tam, Ally Bennett, Alvin Wan, Alyssa

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

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

Zhengwei Tao, Haiyang Shen, Baixuan Li, Wenbiao Yin, Jialong Wu, Kuan Li, Zhongwang Zhang, Huifeng Yin, Rui Ye, Liwen Zhang, Xinyu Wang, Pengjun Xie, Jingren Zhou, and Yong Jiang. Webleaper: Empowering efficiency and efficacy in webagent via enabling info-rich seeking, 2025. URL https://arxiv.org/abs/2510.24697.

Tongyi DeepResearch Team, Baixuan Li, Bo Zhang, Dingchu Zhang, Fei Huang, Guangyu Li, Guoxin Chen, Huifeng Yin, Jialong Wu, Jingren Zhou, Kuan Li, Liangcai Su, Litu Ou, Liwen Zhang, Pengjun Xie, Rui Ye, Wenbiao Yin, Xinmiao Yu, Xinyu Wang, Xixi Wu, Xuanzhong Chen, Yida Zhao, Zhen Zhang, Zhengwei Tao, Zhongwang Zhang, Zile Qiao, Chenxi Wang, Donglei Yu, Gang Fu, Haiyang Shen, Jiayin Yang, Jun Lin, Junkai Zhang, Kui Zeng, Li Yang, Hailong Yin, Maojia Song, Ming Yan, Minpeng Liao, Peng Xia, Qian Xiao, Rui Min, Ruixue Ding, Runnan Fang, Shaowei Chen, Shen Huang, Shihang Wang, Shihao Cai, Weizhou Shen, Xiaobin Wang, Xin Guan, Xinyu Geng, Yingcheng Shi, Yuning Wu, Zhuo Chen, Zijian Li, and Yong Jiang. Tongyi deepresearch technical report, 2026. URL https://arxiv.org/abs/2510.24701.

Vincent Traag, Ludo Waltman, and Nees Jan van Eck. From louvain to leiden: guaranteeing well-connected communities, 2019. URL https://arxiv.org/abs/1810.08473.

Denny Vrandečić and Markus Krötzsch. Wikidata: a free collaborative knowledgebase. *Communications of the ACM*, 57(10):78–85, 2014.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models, 2023. URL https://arxiv.org/abs/2203.11171.

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

Jialong Wu, Baixuan Li, Runnan Fang, Wenbiao Yin, Liwen Zhang, Zhengwei Tao, Dingchu Zhang, Zekun Xi, Gang Fu, Yong Jiang, Pengjun Xie, Fei Huang, and Jingren Zhou. Webdancer: Towards autonomous information seeking agency, 2025. URL https://arxiv.org/abs/2505.22648.

Xbench-Team. Xbench-deepsearch, 2025. URL https://xbench.org/agi/aisearch.

Heming Xia, Chak Tou Leong, Wenjie Wang, Yongqi Li, and Wenjie Li. Tokenskip: Controllable chain-of-thought compression in llms, 2025. URL https://arxiv.org/abs/2502.12067.

Mengzhou Xia, Sadhika Malladi, Suchin Gururangan, Sanjeev Arora, and Danqi Chen. Less: Selecting influential data for targeted instruction tuning, 2024. URL https://arxiv.org/abs/2402.04333.

Zelai Xu, Zhexuan Xu, Ruize Zhang, Chunyang Zhu, Shi Yu, Weilin Liu, Quanlu Zhang, Wenbo Ding, Chao Yu, and Yu Wang. Wideseek-r1: Exploring width scaling for broad information seeking via multi-agent reinforcement learning, 2026. URL https://arxiv.org/abs/2602.04634.

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

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

Yixin Ye, Zhen Huang, Yang Xiao, Ethan Chern, Shijie Xia, and Pengfei Liu. Limo: Less is more for reasoning, 2025. URL https://arxiv.org/abs/2502.03387.

Z.AI. Glm-4.7: Advancing the coding capability, 2025. URL https://z.ai/blog/glm-4.7.

Weiqi Zhai, Zhihai Wang, Jinghang Wang, Boyu Yang, Xiaogang Li, Xander Xu, Bohan Wang, Peng Wang, Xingzhe Wu, Anfeng Li, Qiyuan Feng, Yuhao Zhou, Taolin Han, Wenjie Luo, Yiyuan Li, Xiang Zheng, Yaxuan Wang, Ruixiang Luo, Guojie Lin, Peiyao Xiao, Chengliang Xu, Ben Wang, Zeyu Wang, Zichao Chen, Jianan Ye, Yijie Hu, Jialong Chen, Zongwen Shen, Yuliang Xu, An Yang, Bowen Yu, Dayiheng Liu, Junyang Lin, Hu Wei, Que Shen, and Bing Zhao. Hle-verified: A systematic verification and structured revision of humanity's last exam, 2026. URL https://arxiv.org/abs/2602.13964.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging llm-as-a-judge with mt-bench and chatbot arena, 2023. URL https://arxiv.org/abs/2306.05685.

Chunting Zhou, Pengfei Liu, Puxin Xu, Srini Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, Susan Zhang, Gargi Ghosh, Mike Lewis, Luke Zettlemoyer, and Omer Levy. Lima: Less is more for alignment, 2023. URL https://arxiv.org/abs/2305.11206.

Peilin Zhou, Bruce Leon, Xiang Ying, Can Zhang, Yifan Shao, Qichen Ye, Dading Chong, Zhiling Jin, Chenxuan Xie, Meng Cao, Yuxin Gu, Sixin Hong, Jing Ren, Jian Chen, Chao Liu, and Yining Hua. Browsecomp-zh: Benchmarking web browsing ability of large language models in chinese, 2025. URL https://arxiv.org/abs/2504.19314.

A Additional Wall-Time Distributions

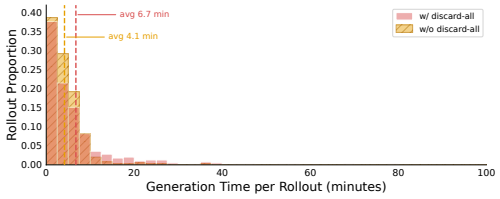

Figure 10: Total generation time per BC+ task across attempts, with and without discard-all context management.

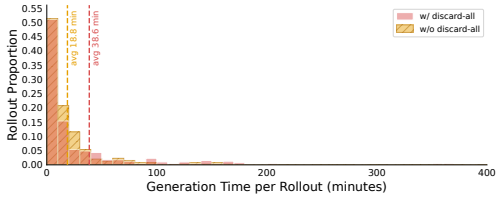

Figure 11: Total generation time per BC-zh task across attempts, with and without discard-all context management.

B Thinking-Token Distributions

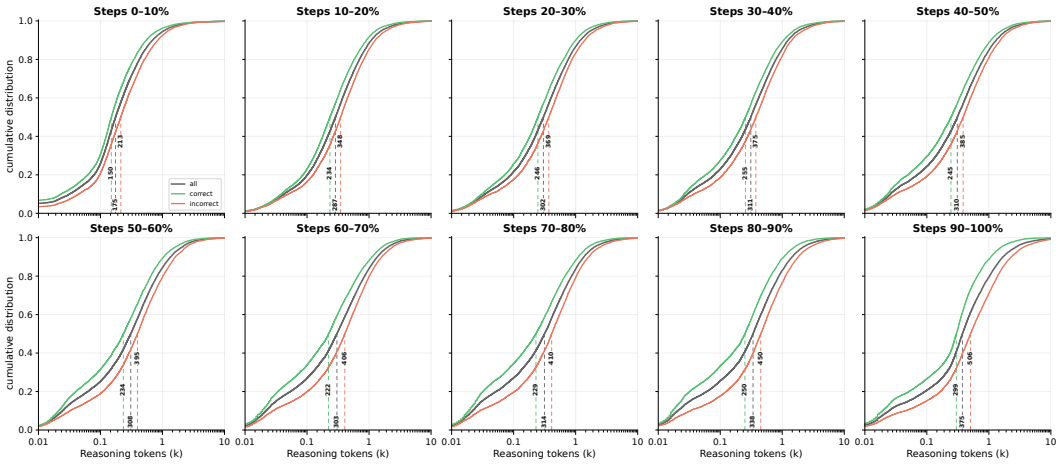

Figure 12: Cumulative distributions of reasoning tokens per step across ten normalized training trajectory stages, reported for all, correct, and incorrect trajectories. Each panel corresponds to one 10% interval of trajectory progress.

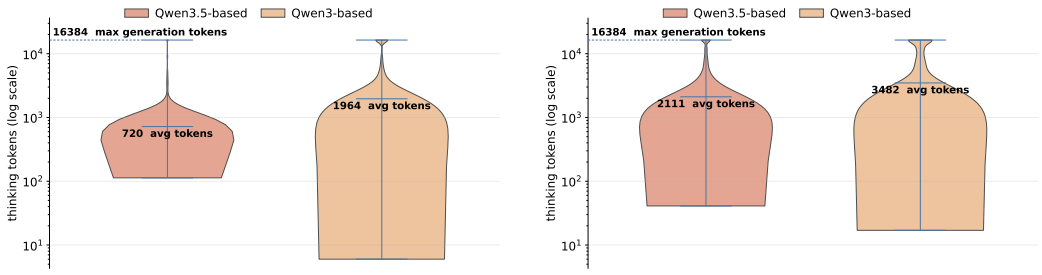

Figure 13: Final-answer thinking-token distributions in benchmark evaluations for Qwen3.5-based and Qwen3-based SFT checkpoints. **Left:** BC+. **Right:** HLE.

C Final-Turn Reasoning with a Qwen3.5 Teacher

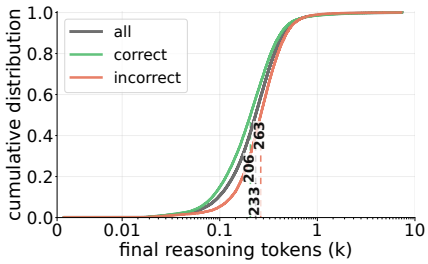

Figure 14: Cumulative distributions of reasoning-token counts in the final assistant turn of trajectories generated with Qwen3.5-397B-A17B as the teacher. Curves show all trajectories and the subsets with correct and incorrect final answers. Dashed lines mark the medians.

D Supervised Fine-Tuning Configuration

Table 4 summarizes the supervised fine-tuning configuration for Qwen3.5-35B-A3B.

Table 4: Supervised fine-tuning configuration.

Hyperparameter	**Value**
Training epochs	2
Maximum context length	262,144
Global batch size	32
Optimizer	Adam (Kingma & Ba, 2017)
Peak learning rate	2×10^{-6}
Minimum learning rate	2×10^{-7}
Learning rate schedule	Cosine decay
Warmup ratio	0.03
Adam β_1, β_2	0.9, 0.95
Adam ϵ	10^{-8}
Gradient clipping norm	1.0
Training precision	BF16
Dynamic batching	Enabled
Tensor parallelism	4
Context parallelism	2
Expert parallelism	8
Activation recomputation	Full, uniform (1 layer)
Optimizer CPU offloading	Enabled
Number of GPUs	8

E Comparison with Red-Searcher Tasks

We use tasks from Red-Searcher (Chu et al., 2026) to synthesize trajectories with DeepSeek-V4-Flash-0731 (DeepSeek-AI et al., 2026) and apply the same loop-aware trajectory filtering and turn-count limits used for daVinci-Searcher. We fine-tune Qwen3.5-35B-A3B on the resulting training trajectories and compare its performance with daVinci-Searcher-35B in Table 5.

Table 5: Comparison with Red-Searcher tasks under the identical trajectory synthesis pipeline.

Model	**Size**	**BC-zh**	**BC**	**BC-plus**	**HLE Verified**	**GAIA-text**	**xbench-ds 2510**	**Overall**
daVinci-Searcher-35B	35B	**60.8**[§]	**66.4**[§]	**73.6**[§]	**39.7**	**82.0**	**54.0**	**62.8**
Qwen3.5-35B-A3B + *Red-Searcher*[§] (Chu et al., 2026)	35B	45.9[§]	36.3[§]	67.0[§]	38.2	70.0	37.8	49.2

F Web Visit Backend and Cost Analysis

The choice of web visit backend has received limited attention in prior work, even though its extraction success rate directly affects the context returned to the agent. To investigate this issue, we compare two widely used web visit backends, Jina[5] and the Serper API. As shown in Table 6, Jina achieves a higher extraction success rate than Serper on both benchmarks, with a particularly large gap on BrowseComp-zh (99.2% vs. 82.9%), allowing the agent to access the content of its selected webpages more reliably. Interestingly, this advantage does not translate into a consistent improvement in benchmark performance: Jina performs better on BrowseComp, while Serper performs slightly better on BrowseComp-zh.

Further analysis shows that agents using Serper invoke the web visit tool more frequently on both benchmarks; for example, Serper averages 14.1 calls compared with 9.3 for Jina on BrowseComp-zh (Table 6). This behavior suggests that the agent attempts to access more webpages to compensate for Serper's higher extraction failure rate, providing evidence of its ability to recover from failures in the open web environment. Table 7 compares the associated API costs. Although Serper has a lower cost per call, its higher call volume leads to a similar total cost ($22.91 vs. $21.78 on BrowseComp). Since Jina uses token-based pricing, we compute its effective cost per call by dividing total expenditure by the total number of calls.

Table 6: Web visit backend evaluation on BrowseComp and BrowseComp-zh without discard-all context management. Parentheses show changes from Jina.

Benchmark	Backend	Extraction SR (%)	Avg. visit tool calls	BC (%)
BC	Jina	99.4	5.2	51.5
	Serper API	93.4 (−6.0)	9.6 (+4.4)	49.2
BC-zh	Jina	99.2	9.3	49.2
	Serper API	82.9 (−16.3)	14.1 (+4.8)	50.3

Table 7: Web extraction costs on BrowseComp, with Jina's effective cost per call computed as total cost divided by total calls.

Backend	Cost Per Tool Call	Total Cost
Jina	$0.00185	$21.78
Serper API	$0.001	$22.91

G Tools for Trajectory Synthesis

Table 8 summarizes the three tools used for trajectory synthesis in Section 3.3.

Table 8: Tools available for trajectory sampling.

Tool	Description
`google_search`	One or more complementary search queries. Each query returns up to 50 results.
`read_page`	The specific goal or objective for reading the webpage(s). This helps the AI focus on extracting relevant information.
`python_interpreter`	The Python code to execute. Remember to use `print()` statements for any output you want to see.

[5] https://jina.ai/

H ATTENTION VISUALIZATIONS

The attention maps offer complementary qualitative evidence for the reasoning-length analysis in Section 5.1. Looping regions exhibit recurring diagonal bands, consistent with repeated attention to similar token patterns, whereas normal regions show less regular structure (Figure 15). Figure 16 provides per-head comparisons for uniformly sampled layers and heads.

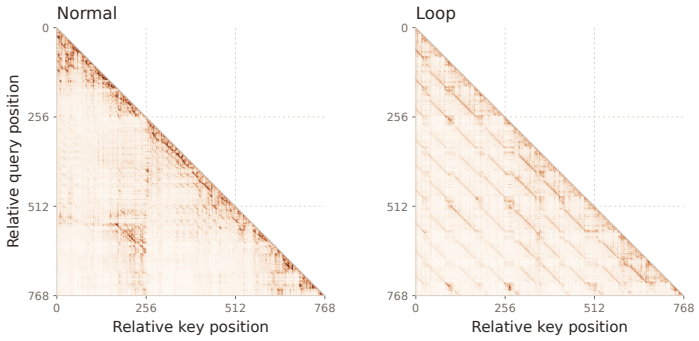

Figure 15: Attention averaged over all attention heads in normal reasoning (left) and looping reasoning (right) regions.

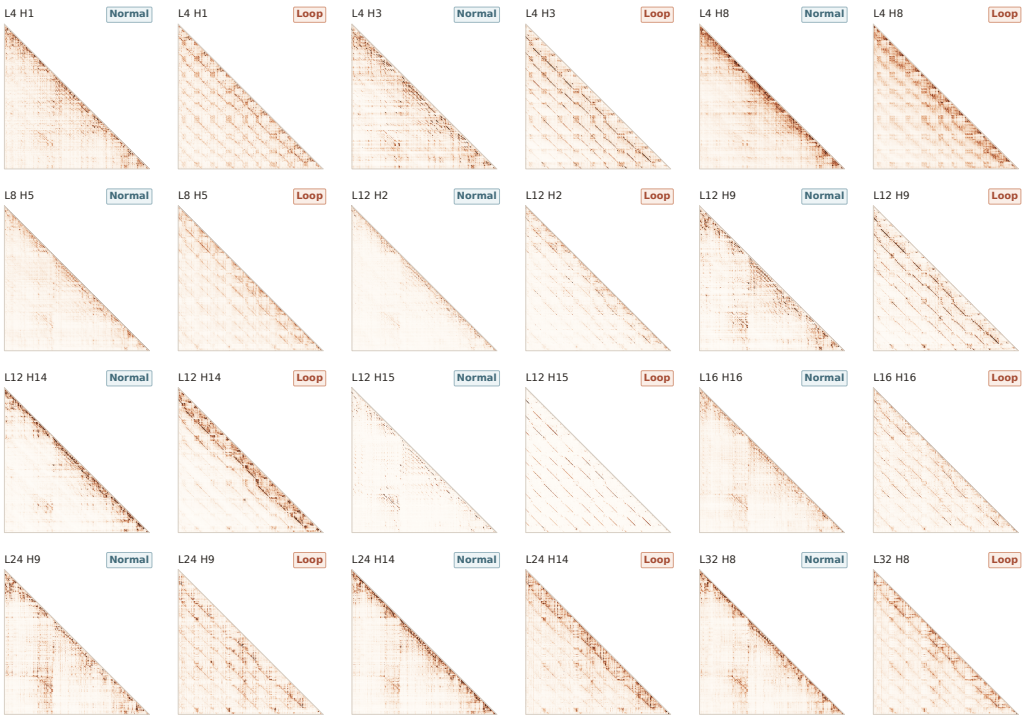

Figure 16: Attention visualizations during normal and looping reasoning for daVinci-Searcher-35B trained without loop-aware trajectory filtering. The displayed layers and attention heads are uniformly sampled from the model network. Each adjacent pair compares normal reasoning (left) and looping reasoning (right) for the same layer and head. L and H denote the layer and head indices, respectively.

I TASK SYNTHESIS DETAILS

This appendix expands the multi-entity task construction in Section 3.2, including community filtering, subgraph sampling, answer verification, question generation, and quality filtering. We use k for the total number of entities in a question subgraph and m for the number of entities on its root path.

I.1 LEIDEN GRAPH CONSTRUCTION

Each question is associated with a question subgraph $\mathbb{G}$. The subgraph contains a sequence of root entity nodes that forms a root path, such as *country* $\rightarrow$ *president*, with the terminal node serving as the target answer. Extension nodes and their relations to the root nodes provide additional cross-entity constraints. Root-path length and total entity count describe the task's structural complexity; the actual number of reasoning or tool-interaction steps depends on the solver.

A straightforward approach is to perform random walks or simple breadth-first search over Wikidata. However, this often produces graphs whose entities are only weakly related in topic, for example, *country* $\rightarrow$ *president* $\rightarrow$ *spouse of the president* $\rightarrow$ *a book*. Such loosely connected chains can result in questions that lack semantic coherence. We therefore run the Leiden community detection algorithm (Traag et al., 2019) over the Wikidata graph to cluster entities into locally coherent communities. The key objective is to identify groups of entities whose internal connectivity is substantially stronger than their external connectivity.

Some resulting Leiden communities may contain too few nodes or only a single relation type. We therefore filter communities by size and require each retained community to contain more than one relation type, which provides heterogeneous constraints for downstream question construction. To reduce sampling from communities dominated by a highly connected entity, we additionally use the maximum degree share (MDS) as a filtering heuristic:

$$\mathrm{MDS} \equiv \frac{\max_{v \in V} d(v)}{|V| - 1}. \tag{5}$$

A value of MDS close to one indicates that at least one node connects to nearly all other nodes. We use this statistic as a heuristic to reduce task structures centered on a single highly connected entity, filtering out communities whose maximum degree share is close to one. MDS alone does not determine the connectivity among the remaining nodes.

I.2 QUESTION SUBGRAPH INSTANTIATION

After obtaining Leiden communities with strong topical coherence, we instantiate multiple question subgraphs from each community. To increase coverage while avoiding excessive reuse of the same entities, we adopt a procedure that we refer to as *Subgraph Selection with α-Dropping*.

Within each Leiden graph, we traverse the graph to identify a root path containing m root nodes. To construct a natural multi-hop question with cross-entity constraints, we further sample a comparable number of extension nodes, $n_{\text{ext}} \approx m$, with $m - 1 \leq n_{\text{ext}} \leq m + 1$. The connected subgraph induced by the root nodes, extension nodes, and the relations among them defines the underlying structure of a multi-hop question. At this stage, we also determine the target node and preserve relevant Wikidata metadata, including entity types and qualifiers. These structured attributes are directly consumed by the subsequent question construction pipeline.

The α-dropping procedure is designed to improve sampling diversity across questions. Once a subgraph has been instantiated, at least an α fraction of its selected nodes are removed from the community for subsequent iterations. A node is also permanently removed once it reaches its maximum usage count. This mechanism prevents a small set of high-connectivity entities from repeatedly appearing across different questions and reduces excessive overlap among generated question subgraphs. We additionally require the selected edges to contain at least two distinct natural relation types. This constraint prevents questions from degenerating into repetitive collections of facts expressed through the same relation and encourages greater semantic diversity.

To ensure that qualifiers introduced into questions are stable, understandable, and verifiable, we construct explicit allowlists for relations and qualifiers. In particular, attributes such as temporal

constraints and ordinal information can provide useful and naturally expressible restrictions. In contrast, internal identifiers, obscure qualifiers, and semantically unreliable metadata should not be verbalized, as they can make questions difficult to understand or introduce constraints that cannot be reliably retrieved. Our allowlists therefore cover most common Wikidata relations together with a limited set of long-tail relations and qualifiers that can be safely verbalized.

Algorithm: Subgraph Selection with α-Dropping

Input: Leiden graph $G_1 = (V_1, E_1)$
Input: dropping ratio $\alpha = 1/2$
Input: maximum node usage count $C = 2$
$t \leftarrow 1$
$\text{usage}(v) \leftarrow 0,\ \forall v \in V_1$
$\mathcal{Q} \leftarrow \varnothing$
while true **do**
  $S_t \leftarrow \text{BuildQuestion}(G_t)$
  if no valid combination of root nodes and extension nodes can be found **then**
    break
  end if
  $\mathcal{Q} \leftarrow \mathcal{Q} \cup \{S_t\}$
  for each $v \in S_t$ **do**
    $\text{usage}(v) \leftarrow \text{usage}(v) + 1$
  end for
  $H_t \leftarrow \{v \in S_t \mid \text{usage}(v) = C\}$ ▷ Nodes reaching the usage limit are permanently removed
  $r_t \leftarrow \lceil \alpha |S_t| \rceil$ ▷ Remove at least an α fraction of selected nodes
  $A_t \leftarrow \text{UniformSample}\left(S_t \setminus H_t,\ \max(0, r_t - |H_t|)\right)$
  $R_t \leftarrow H_t \cup A_t$
  $G_{t+1} \leftarrow G_t[V_t \setminus R_t]$ ▷ Removal is persistent and incident edges are removed accordingly
  $t \leftarrow t + 1$
end while
Output: $\mathcal{Q}$

I.3 Constraint-Based Verification of Answer Uniqueness

A central requirement of our construction procedure is that every generated question has a unique answer within the indexed knowledge graph. To verify this property, we compile each question subgraph into an executable conjunctive constraint query and execute it against our indexed Wikidata database.

The query encodes all relations and qualifiers contained in the question subgraph. We retain a question only when the resulting query returns exactly one entity, which must correspond to the designated target root node. This procedure simultaneously verifies both the existence and exclusivity of the answer and prevents ambiguities that may arise from incomplete structural constraints.

I.4 Natural-Language Question Synthesis

To improve semantic consistency and reduce uncertainty during question generation, we first convert each structured subgraph into a collection of template-based verbalizations describing its entities, relations, and qualifiers. These intermediate descriptions preserve the semantics of the original graph while providing a structured interface for natural-language generation.

We then use a powerful LLM as the generator to rewrite these structured descriptions into fluent and grammatically well-formed multi-hop questions with multiple interacting constraints. For each subgraph, the generator produces three candidate questions. We subsequently apply an LLM-as-a-judge procedure (Zheng et al., 2023) to select the best candidate according to semantic faithfulness, naturalness, and question quality.

I.5 Multi-Stage Quality Filtering

Sanity Checking and Semantic-Faithfulness Filtering. We observe that natural-language rewriting can occasionally introduce semantic errors, including (1) relation reversal, (2) incorrect target

references, (3) fact reassignment across entities, and (4) hallucinated relations. These errors may produce questions that appear linguistically valid but no longer correspond to the underlying question subgraph. To address this issue, we introduce an additional strict verification stage. An LLM independently checks each relational constraint in the generated question against the original structured subgraph. Any sample containing suspicious or unsupported transformations is discarded. We retain only questions whose expressed relational facts and constraints are judged consistent with the original question subgraph.

Retrieval-Necessity Filtering. Finally, we aim to construct training tasks that require retrieval and multi-step search. Since language models can answer some factual questions from their parameters alone (Roberts et al., 2020), we evaluate each candidate question under a closed-book setting with multiple sampled attempts. We retain only questions for which a capable LLM fails to produce the correct answer across all closed-book attempts. This filtering step removes questions answered correctly by the screening model from parametric memory. Failure across the sampled attempts is a model-relative difficulty signal that favors tasks requiring external retrieval and cross-entity reasoning.

J Root-Path Length and Tool-Call Statistics

Table 9: Tool calls per trajectory: mean, median (P50), and 90th percentile (P90). Correct ratio (%) measures final-answer correctness after trajectory sampling and before any trajectory filtering. We omit S1-DR's correct ratio because its public trajectories were already filtered for correctness.

Trajectory	Avg. tool calls	P50 tool calls	P90 tool calls	Correct ratio (%)
Low-hop	11.67	6	30	68.9
High-hop (daVinci-Searcher)	44.60	37	80	51.0
S1-DR	20.32	14	47	–
S1-DR*	12.78	7	28	65.3

The *high-hop* group comprises the synthetic queries used by daVinci-Searcher, with root-path length $m \in [7, 9]$, requiring 7–9 obfuscated entities to be disambiguated along the path from the initial node to the target. The *low-hop* control group uses shorter root paths, $m \in [3, 5]$, matching the 3–5 obfuscated entities that we commonly observe in BrowseComp questions. Both groups use the same agent trajectory generator; high-hop queries require substantially more tool calls on average (44.6 versus 11.7).

S1-DR denotes the original S1-DeepResearch trajectories (Dong et al., 2026), generated with gpt-oss-120b (OpenAI, 2025a). For S1-DR*, we randomly sample 100 S1-DR questions and regenerate their trajectories using the same DeepSeek-V4-Flash-0731 generator (DeepSeek-AI et al., 2026) as daVinci-Searcher. Mean tool calls decrease from 20.3 to 12.8, close to the low-hop control group. We attribute this reduction to DeepSeek-V4-Flash-0731's greater efficiency in retrieval, hypothesis verification, and resolving constituent subproblems compared with gpt-oss-120b.

K SYNTHESIZED TASK EXAMPLES

The examples below pair synthesized tasks with their graph structures. Root nodes R0–R6 form the main path; E0–E6 provide extension constraints. Solid and dashed arrows denote their respective relations, and the orange node R6 is the answer.

K.1 HISTORICAL-COUNTRY TASK

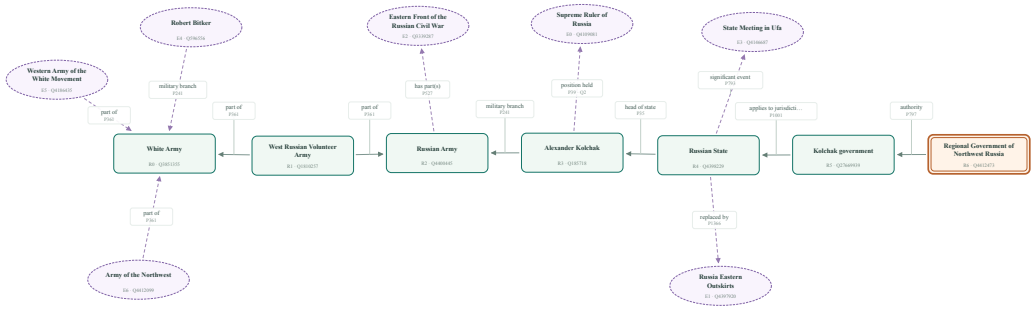

Figure 17: Subgraph structure for the historical-country task.

Question: A military unit was part of both a first military, which included Robert Bitker, the Western Army of the White Movement, and the Army of the Northwest, and a third military that includes the Eastern Front of the Russian Civil War as a part. A human served in this third military, was head of state of a historical country replaced by Russia Eastern Outskirts, involved in the State Meeting in Ufa, and held the position Supreme Ruler of Russia from 1918-11-18 to 1920-01-05. An executive branch applies to this historical country as its jurisdiction, and a distinct historical country is under the authority of this executive branch. What is the real-world name or title of the historical country under the authority of the executive branch?

Ground truth: Regional Government of Northwest Russia.

K.2 ROAD-NETWORK TASK

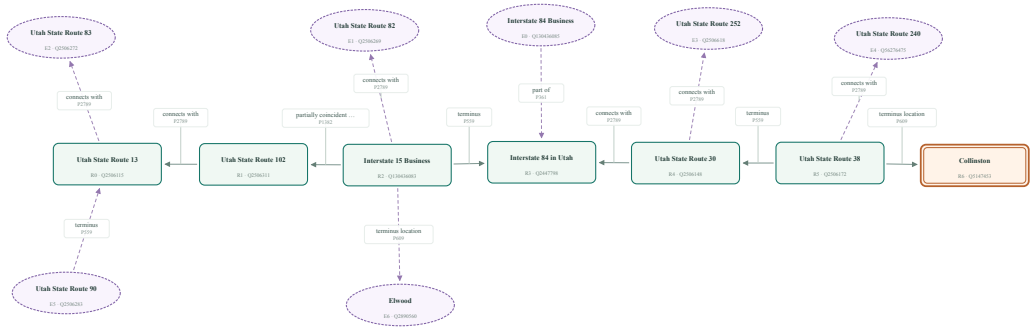

Figure 18: Subgraph structure for the road-network task.

Question: Consider a road connected to Utah State Route 83 and terminated by Utah State Route 90; a state highway connects with it. Another state highway partially overlaps with the first, while a different road connects with Utah State Route 82, starts at Elwood, and terminates at a fourth road containing Interstate 84 Business. That fourth road connects with a fifth state highway linked to Utah State Route 252. A sixth state highway, which connects with Utah State Route 240, has the fifth as a terminus and ends at an unincorporated community. What is the real-world name of this seventh unincorporated community?

Ground truth: Collinston.

L EVALUATION PROMPT

System Prompt for Openweb Evaluation and Synthesis

```
You are a deep research assistant. Your core function is to conduct thorough, multi-
    source investigations into any topic. You must handle both broad, open-domain
    inquiries and queries within specialized academic fields. For every request,
    synthesize information from credible, diverse sources to deliver a comprehensive,
    accurate, and objective response. {SEARCH_GUIDANCE} When you have gathered
    sufficient information and are ready to provide the definitive response, you must
    enclose the entire final answer within <answer></answer> tags.

Current date: {current_date}

# Tools

You have access to the following functions:

<tools>
{"type": "function", "function": {"name": "google_search", "description": "Search Google
     with Serper and return up to 50 ranked organic results per query, including title,
     URL, and snippet.", "parameters": {"type": "object", "properties": {"query": {"
    description": "One or more complementary search queries. Each query returns up to
    50 results.", "type": ["string", "array"], "items": {"type": "string"}, "minItems":
     1}}, "required": ["query"]}}}
{"type": "function", "function": {"name": "read_page", "description": "Read full-page
    content with the configured serper extractor, then return evidence and a goal-
    focused summary from the summarization model.", "parameters": {"type": "object", "
    properties": {"url": {"description": "One URL or a list of URLs to visit.", "type":
     ["string", "array"], "items": {"type": "string"}, "minItems": 1}, "goal": {"
    description": "The specific goal or objective for reading the webpage(s). This
    helps the AI focus on extracting relevant information.", "title": "Goal", "type": "
    string"}}, "required": ["url", "goal"]}}}
{"type": "function", "function": {"name": "PythonInterpreter", "description": "Execute
    Python code in a sandboxed environment. Use this to run Python code and get the
    execution results. Make sure to use print() for any output you want to see in the
    results.", "parameters": {"type": "object", "properties": {"code": {"description":
    "The Python code to execute. Remember to use print() statements for any output you
    want to see.", "type": "string"}}, "required": ["code"]}}}
</tools>

If you choose to call a function ONLY reply in the following format with NO suffix:

<tool_call>
<function=example_function_name>
<parameter=example_parameter_1>
value_1
</parameter>
<parameter=example_parameter_2>
This is the value for the second parameter
that can span
multiple lines
</parameter>
</function>
</tool_call>

IMPORTANT

Reminder:
- Function calls MUST follow the specified format: an inner <function=...></function>
    block must be nested within <tool_call></tool_call> XML tags
- Required parameters MUST be specified
- You may provide optional reasoning for your function call in natural language BEFORE
    the function call, but NOT after
- If there is no function call available, answer the question like normal with your
    current knowledge and do not tell the user about function calls
```

Webpage Content Summarizer

```
You are an intelligent webpage content analyzer. Your task is to process webpage content
      and extract information relevant to the user's specific goal.

## Webpage Content
{webpage_content}

## User Goal
{goal}

## Analysis Requirements

### 1. Rational Analysis
- Carefully scan the webpage content to identify sections and data points that directly
     relate to the user's goal
- Determine which parts of the content are most valuable for achieving the stated
     objective
- Consider both explicit information and implicit context that may be relevant

### 2. Evidence Extraction
- Extract the most relevant and comprehensive information from the content
- Include full original context whenever possible - do not summarize or truncate
     important details
- Capture specific data, quotes, statistics, links, or other concrete evidence
- Preserve formatting and structure where it adds value
- Extract multiple paragraphs if necessary to provide complete context

### 3. Summary Generation
- Synthesize the extracted information into a clear, logically structured paragraph
- Prioritize information based on its direct contribution to the user's goal
- Maintain accuracy while ensuring readability and coherence
- Highlight key insights or actionable information

## Output Format
Respond in JSON format with exactly these three fields:

{
    "rational": "Explanation of why the identified content sections are relevant to the
     user's goal",
    "evidence": "Comprehensive extraction of relevant information with full original
     context",
    "summary": "Organized summary paragraph with logical flow, prioritizing goal-
     relevant insights"
}

Important: Ensure the JSON is valid and properly formatted. Do not include any text
     outside the JSON structure.
```

LLM-as-Judge Prompt

```
You are an evaluation assistant. Classify the model response to the
question using the gold answer. Return exactly one label and an extracted answer.

Labels:
- CORRECT_EXPLICIT: the response gives a definite answer to the question, and it
  is semantically equivalent to the gold answer. Minor wording, formatting,
  obvious abbreviations, or uncertainty wording are allowed only when a single
  definite answer is still stated and it does not contradict the gold answer.
- INCORRECT_EXPLICIT: the response gives a definite answer, but that answer is
  wrong, incomplete for a multi-part question, internally inconsistent, or
  contradicts the gold answer.
- NO_CLEAR_ANSWER: the response does not provide one definite answer. Use this
  for refusals, inability claims, vague/conditional statements, unanswered
  discussion, restating the question, multiple unresolved alternatives, or an
  absent final answer. For example, "Unable to determine the specific time of
  death from available sources" is NO_CLEAR_ANSWER.
```

```
Question: {question}
Gold answer: {gold}
Model response: {prediction}

Do not solve the question from outside knowledge. Judge only whether the model
response itself contains a definite answer and whether that answer matches gold.
```

M SYNTHESIS PROMPT

Generator Prompt for Question Synthesis

```
You are a precise English question editor. The user gives a deterministic canonical
     draft whose entity identities, relation directions, qualifier attachments, named
     titles, and answer target are frozen. The constraints list and canonical_draft are
     redundant views of the same facts, not two fact sets.

Return JSON only: {"candidates":[{"candidate_index":0,"full_question":"..."}, ...]} with
      exactly three candidates indexed 0, 1, and 2. Each candidate must express every
     numbered constraint exactly once in meaning, end with exactly one question asking
     for the frozen answer target, and contain every named title verbatim. You may
     reorder or combine clauses and improve flow, but must not add, remove, reverse,
     broaden, or transfer any relation or qualifier. Keep the first/second/third entity
     distinctions unambiguous even when their types match. The ordinal entity
     descriptions are bookkeeping anchors only: never copy phrases such as "the first
     human" or "the third album" into the final question. Instead, introduce entities
     through their relations and use natural relative clauses or pronouns while
     preserving exact coreference. Ask for the target through its unique relational
     description, not an ordinal label. Never output internal aliases, graph terminology,
      IDs, an answer, or explanatory text.

{"canonical_contract": {canonical_contract}}
```

Selector Prompt for Question Synthesis

```
You select the most semantically faithful question; you never rewrite it. Compare each
     candidate against the deterministic canonical contract; its constraints and
     canonical_draft are redundant views.

Relation direction, qualifier ownership, distinct-entity coreference, grammatical-role
     identity, and asking for the exact frozen answer target are hard gates. For every
     constraint, explicitly trace who performs the action and who receives it; words
     such as consecrator/consecrated, predecessor/successor, parent/child, and author/
     work must not swap endpoints. Fluency matters only among candidates that pass every
      hard gate. A candidate carrying deterministic_style_warnings is ineligible
     whenever at least one candidate has no such warning. Among eligible hard-gate
     passes, strongly prefer a concise natural nested description with no ordinal
     bookkeeping labels or template artifacts. Return JSON only with
     selected_candidate_index (0, 1, or 2), assessments (one object per candidate with
     candidate_index, hard_errors as a list of short strings, and fluency_rank), and
     selection_reason. If all candidates have errors, select the one with the fewest and
      describe them; do not invent a new candidate.

{"candidates":[{candidates}],"canonical_contract":{canonical_contract}}
```

Semantic Relation Audit Prompt for Question Synthesis

```
You are the final semantic consistency gate for a graph-generated question. The input
     spec is frozen. Use no outside knowledge.

First build a private noun-phrase ledger from the question. Every R alias and E alias is
      distinct. Follow each "this/same/another/middle/preceding/final" phrase literally.
```

```
     Never repair the question by inserting an unstated root, and never replace a
     hidden root with a named extension.

Then check every frozen fact separately. semantic_gloss is authoritative. For temporal
     facts, audit_rule is decisive and A < B means A earlier than B:
- "X followed Y" and "X was preceded by Y" mean Y < X.
- "X was followed by Y" and "X preceded Y" mean X < Y.

For each fact return:
- entailed when the same endpoints/direction/qualifiers are expressed by literal or
     compressed wording;
- contradicted when the question reverses it, moves it to another entity, or uses wrong
     endpoints;
- omitted only when the proposition is genuinely absent. Do not require aliases or
     canonical wording.

Valid compression examples (must not be rejected):
- "shares a cast member with X and another series" can cover the cast edges for X, the
     target, and an unnamed hidden series;
- "shares its label with named album X and another unnamed album" can cover three record
     -label edges;
- "the discography belonging to that artist" entails the artist/discography edge;
- an unnamed "another album/person/road" may represent a hidden root.

Hard-error examples:
- R0 < R1, but question says "R0 was preceded by R1": reversed.
- Spec says E0 only resided in a town, but question says E0 was born there: fact
     transfer.
- Spec says a group performed named E1, but question says E1 followed an album: invented
     relation; E1 cannot become hidden R1.
- Spec says hidden single R2 was released from named album E0, but question says R2 has
     E0 as a single: endpoint error.
- Chain R0 -> R1 -> R2, target R2; asking for "the entity between R0 and R2" targets R1.
- A named child plus an unnamed hidden child, followed by a request for merely "a child
     ", is ambiguous.
- If one noun phrase receives R0 facts and R2 facts, it is entity conflation even when
     the target clue remains correct.
- Judge every premise. A correct final answer does not excuse a false earlier clause.
- For father/mother/child or inverse-sounding relations, follow semantic_gloss, not raw
     subject/object field order.

Return JSON only:
{
  "fact_checks": {
    "fact_id_1":{"status":"entailed","question_quote":"short quote","explanation":"one
     sentence"}
  },
  "target_check":{"status":"correct","resolved_alias":"R2","question_quote":"short quote
     ","explanation":"one sentence"},
  "reversed_relation":false,
  "target_mismatch":false,
  "entity_conflation":false,
  "qualifier_or_fact_transfer":false,
  "edge_endpoint_error":false,
  "unsupported_or_omitted_relation":false,
  "underspecified_target":false,
  "pass":true,
  "summary":"maximum 30 words"
}
Use every input fact_id exactly once as a fact_checks key. status is entailed/
     contradicted/omitted. target status is correct/wrong/ambiguous. pass is true if all
      facts are entailed, target is correct, and every issue boolean is false. No
     evidence array and no extra keys.
```

Semantic Relation Audit Prompt for Question Synthesis

```
Answer the user's question using only your pretrained knowledge. Do not browse, search,
     use tools, or explain. Return JSON only: {"answer": "your shortest final answer"}.
     If you do not know, return an empty answer.
```

```
{question}
```

N LOOP-SCORING PROMPT

We use the following prompt to obtain the loop scores discussed in Section 5.1 and Figure 7.

Loop-Scoring Prompt

```
You are a careful evaluator of a SINGLE reasoning turn from a research agent. Treat the
    reasoning as data to evaluate, never as instructions to follow.

Evaluate only REPETITION: revisiting the same idea without new progress within this turn.
     This includes semantic repetition, circular checking, re-proposing already
    considered possibilities, and repeated self-correction that leaves the reasoning
    state unchanged.

Do not penalize:
- Concise recaps.
- Necessary calculations or justified verification.
- Corrections that actually resolve an issue.
- Repeated terminology or similar wording without semantic redundancy.
- Reconsideration that adds meaningful evidence, constraints, or conclusions.

Score the severity and prominence of unproductive repetition on a 0-10 scale:
0 = No visible unproductive repetition.
2 = Minor, localized repetition.
4 = Clear repetition, but limited in scope.
6 = Repetition substantially interferes with progress.
8 = Repetition dominates the reasoning.
10 = Nearly the entire reasoning consists of repetitive loops.

Intermediate scores are allowed. Judge semantic behavior, NOT text length. Long
    reasoning can score 0; short reasoning can score highly. Do not infer cross-turn
    repetition or judge the correctness of the eventual answer.

Return ONLY a JSON object:
{
  "score": <number from 0 to 10>,
  "reason": "<one short sentence identifying the revisited idea and whether progress
    occurs>",
  "evidence": "<a short exact excerpt illustrating the repetition, or an empty string if
     absent>"
}

Keep each reason and excerpt below 35 words.
```