REINFORCEMENT LEARNING FOR SEARCH AGENTS WITH STATE-EXTERNALIZING HARNESSES

Anonymous authors
Paper under double-blind review

ABSTRACT

Search agents are often trained as policies over growing transcripts. The model must decide how to search while also remembering what it has seen, which evidence is useful, which constraints remain open, and which claims have been checked. We argue that this puts too much routine state management inside the policy, so reinforcement learning must optimize both semantic search decisions and recoverable bookkeeping that the environment can maintain more reliably. We introduce `Harness-1`, a 20B search agent (retrieval subagent) trained with reinforcement learning inside a stateful search harness. The harness maintains environment-side working memory, including a candidate pool, an importance-tagged curated set, compact evidence links, verification records, compressed and deduplicated observations, and budget-aware context rendering. The policy retains the decisions: what to search, which documents to keep or discard, what to verify, and when to stop. Across eight retrieval benchmarks spanning web, finance, patents, and multi-hop QA, `Harness-1` achieves 0.730 average curated recall, outperforming the next strongest open search subagent by +11.4 points. It also has higher average recall than four of the six much larger frontier-model searchers we test. Its gains over the Context-1 search agent are especially strong on held-out transfer benchmarks, suggesting that reinforcement learning over explicit search state can produce retrieval behaviors that generalize beyond the training domains.

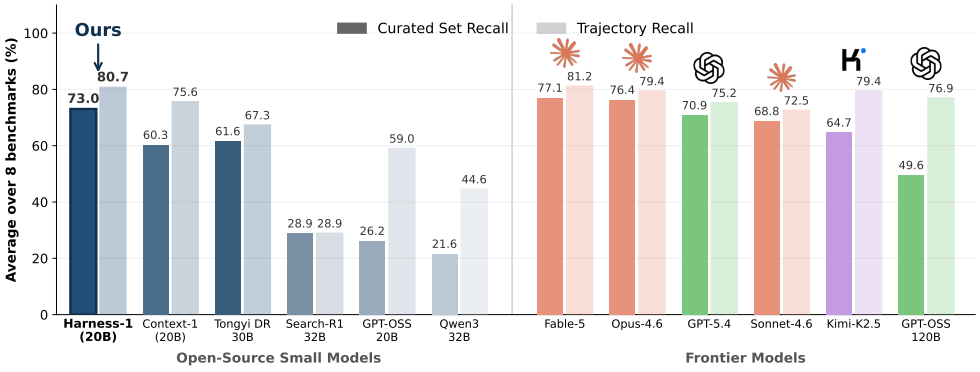

Figure 1: **Average performance across eight search benchmarks.** We report Curated Set Recall and Trajectory Recall (documents encountered anywhere in the episode). `Harness-1`, our 20B open search agent trained with a stateful harness, substantially improves over open search agents and beats four of the six much larger frontier-model searchers.

1 INTRODUCTION

Search agents are usually described as language models that call retrieval tools. Given a question, they issue queries, read the returned evidence, decide what is still missing, and return documents to a downstream answerer. This view follows prior work on reasoning-and-acting agents, iterative retrieval, active retrieval, and tool-use training (Yao et al., 2022; Trivedi et al., 2023; Jiang et al., 2023; Qin et al., 2023; Jiang et al., 2026). It captures what the agent visibly does, but not the search

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.

state it must build. In a successful multi-turn episode, the agent must remember which documents it has seen, which candidates to keep, which constraints remain uncovered, which entities connect separate pieces of evidence, and which claims it has checked against source text.

This distinction becomes especially important for reinforcement learning. Recent work trains LLMs with RL to interact with search engines and retrieval systems, improving query generation, multi-turn search, and downstream retrieval utility (Jiang et al., 2025a; Jin et al., 2025; Jiang et al., 2025b; Wang et al., 2025b). Yet most retrieval-agent training still asks the model to learn both semantic search behavior and routine state management at once. As the transcript grows, the policy must rebuild the useful search state from raw observations. This is inefficient and can make learning poorly conditioned. Hard queries may give rollouts nearly identical empty-set rewards. Larger tool vocabularies may collapse to repeated search calls. Cross-document structure may be present in the transcript but too diffuse to use reliably. When the final curated set is empty or wrong, the reward often does not show whether bad search, forgotten evidence, missing verification, or poor curation caused the failure.

Stateful cognitive offloading. A retrieval policy should make semantic decisions over explicit search state: what to search, which documents to keep, what claim to verify, and when to stop. The harness should maintain the recoverable state around those decisions: candidate pools, curated evidence, cross-document links, verification records, and context-budget summaries. This gives RL a stable interface for improving search behavior. The model does not have to rediscover bookkeeping from an append-only transcript.

This view changes what tool orchestration means. Prior work on agent-computer interfaces and software-engineering agents shows that the interface around a language model can strongly shape agent behavior and performance (Yang et al., 2024; Xia et al., 2025). In retrieval, recent systems treat context management as part of search, for example through context pruning or self-editing search (Bashir et al., 2026). In `Harness-1`, the interface itself is part of the method. The agent acts on a persistent retrieval state that the environment maintains and renders back in each observation. The policy still decides what to search, which documents to promote, what to verify, and when the evidence is sufficient. The harness carries the state maintenance that would otherwise consume the policy's context and optimization budget.

We instantiate this principle in `Harness-1`, a 20B search agent built on `gpt-oss-20b` (OpenAI et al., 2025). At each turn, the model sees recent tool results and a rendered WORKINGMEMORY with the candidate pool, curated set, search history, evidence graph summary, verification state, and context-budget marker. Its actions edit this state instead of only appending to a transcript. `curate` adds, removes, and importance-tags documents. `verify` checks policy-written claims against remembered documents. `review_docs` re-renders previously seen documents without a new corpus call. Search observations are compressed and deduplicated before they reach the prompt. The same state renderer serves teacher data generation, SFT replay, RL rollout, and evaluation.

A richer harness is not automatically suitable for policy training. If every rollout starts with an empty curated set, early rewards can be indistinguishable. If derived state is too verbose, it competes with evidence for context. If reward only values discovery, the policy can ignore curation and verification and converge to repeated search. We therefore identify three requirements for stateful search harnesses. **Warm-started curation** is implemented by auto-seeding a tentative curated set from the first successful search. **Compact derived-state rendering** uses importance tags, evidence-graph summaries, and verification records. **Diversity-preserving incentives** encourage a regular cycle of searching, curating, reviewing, and verifying.

We evaluate `Harness-1` on eight retrieval benchmarks spanning web, finance, patents, and long-context multi-hop search. It reaches 0.730 average curated recall and improves downstream answer accuracy when its curated set is passed to frozen frontier generators. The gains extend beyond the training domains. The same operations over explicit search state improve retrieval on specialized corpora, open-web tasks, and held-out multi-hop QA benchmarks.

Contributions. First, we frame harness design as a central problem in retrieval-agent RL and propose stateful cognitive offloading, a principle for dividing search state between policy and environment. Second, we introduce `Harness-1`, a 20B search agent whose harness keeps the offloaded search state in environment-side working memory. Third, we identify the three trainability require-

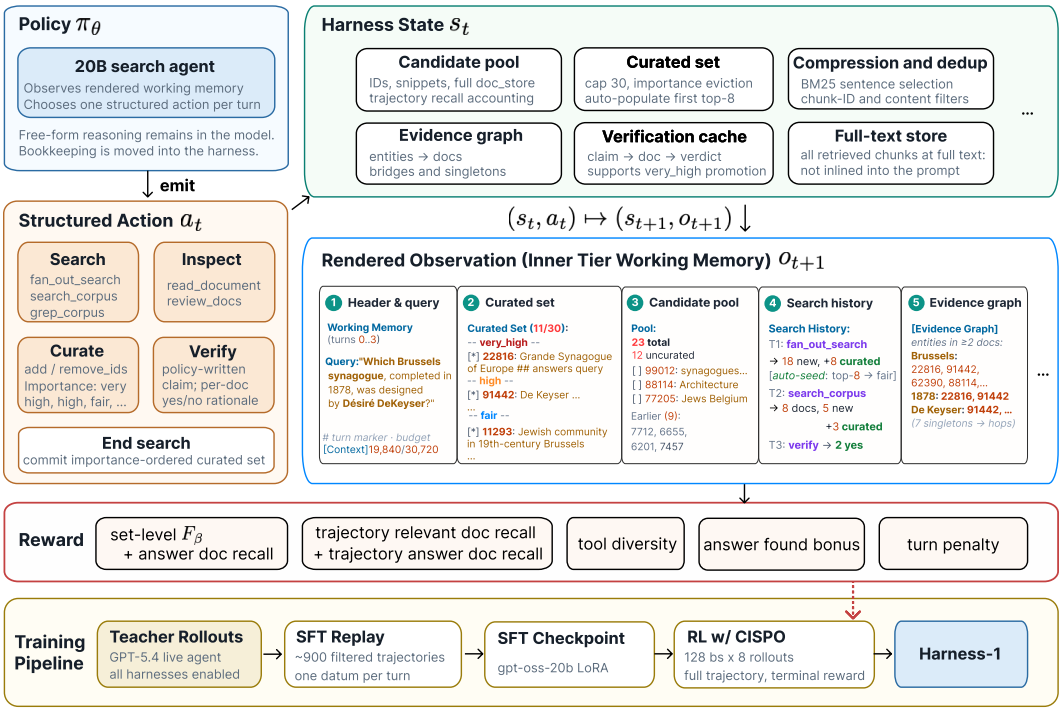

Figure 2: **Overview of** `Harness-1`**.** The policy makes decisions over search, inspection, curation, verification, and termination. The harness maintains the state around those decisions: document pool, importance-tagged curated set, evidence graph, verification records, compression and deduplication, result summaries, and budget markers. The same state interface is used for teacher rollouts, SFT replay, CISPO RL, and evaluation.

Table 1: State notation and offloaded bookkeeping in the `Harness-1` harness. The harness maintains and renders state; the policy retains the semantic choices over that state.

State	Harness-maintained bookkeeping	Policy decision preserved
P_t	Candidate pool after compression and deduplication	Which candidates to inspect, read, or curate
C_t, I_t	Curated output set and importance tags, warm-started by auto-seeding	Which documents to add, remove, promote, or demote
D_t	Full-text memory for retrieved documents	Which seen documents to revisit through `review_docs`
G_t	Evidence graph over entities, dates, and documents	Which bridge, singleton, or relation to pursue next
V_t	Verification records for policy-written claims	Which claim to check and which documents to test
H_t	Search history and result summaries	When to diversify, backtrack, or continue a search thread
B_t	Budget-safe renderer and context marker	When to search, read, summarize, or stop

ments for stateful search harnesses described above. Fourth, we show that stateful harnessing improves retrieval quality across eight benchmarks and modular RAG answer accuracy on BrowseComp+ and SEC.

2 HARNESS-1

`Harness-1` is a search agent that produces a ranked set of documents for a downstream answering model. It runs inside a state-machine harness centered on a per-episode WORKINGMEMORY. At each turn, the harness renders compact search state with recent actions and observations. The model emits one structured Harmony action, and the harness executes it, updates the state, and renders the next observation.

The harness makes no retrieval decisions. It maintains the recoverable state around them. The policy decides what to search, which documents to inspect or keep, what claims to verify, and when to stop. Each action applies a transition $(s_t, a_t) \mapsto (s_{t+1}, o_{t+1})$ over the harness state summarized in Table 1. In this sense the harness is *stateful*. Tool outputs update persistent search state that later actions can inspect, edit, or verify, instead of only adding text to the next prompt.

WORKINGMEMORY has two tiers. The prompt-facing tier renders compact state: P_t, C_t, I_t, G_t, V_t, H_t, and B_t. The outer tier D_t stores full text and metadata for retrieved chunks, which the policy can revisit through `review_docs` or `read_document`. The prompt thus carries actionable search state, not the full retrieval transcript. Appendices H and Q give the full state-transition algorithms and trajectory-level examples.

2.1 POLICY ACTIONS AS EDITS OVER WORKING MEMORY

`Harness-1` exposes five classes of actions: retrieval, memory inspection, curation, verification, and termination. Retrieval actions bring new evidence into the episode. `fan_out_search` runs up to five diverse queries in parallel, `search_corpus` performs targeted hybrid search, `grep_corpus` performs exact pattern matching, and `read_document` returns full text for a known document ID. After compression and deduplication, their outputs update the candidate pool P_t and the full-text store D_t rather than being appended raw to the prompt.

Curation. The main state-editing action is `curate`, which updates the final output set C_t and its importance map I_t. It lists documents to add, documents to remove, and an importance tag for each added document: `very_high`, `high`, `fair`, or `low`. The curated set holds at most $M{=}30$ documents. When the set is full, the harness evicts the lowest-importance documents first (Appendix K). The tags make the policy's confidence and priority explicit.

Auto-seeding. A blank curated set provides little learning signal on hard queries, since many early rollouts terminate with the same empty output. To avoid this, the first successful search automatically seeds C_t with the top $k{=}8$ reranked results from P_t, each assigned $I_t(d) =$ `fair`. The policy must then refine this tentative set by promoting strong documents and removing weak ones. Auto-seeding does not decide relevance for the model. It changes the starting point from building a set to refining one, so early trajectories differ in how they edit candidates. In the BC+ and SEC trajectories (Appendices Q.1 and Q.2), the first search seeds plausible candidates, and later `curate` calls promote documents that survive reading, exact search, or verification.

Verification and memory review. `verify` lets the policy write a claim and select remembered documents to test. The harness then checks support against text in D_t and records yes/no judgments with rationales in V_t. Each check costs about $0.0002 (Appendix M.1.5). `review_docs` re-renders documents already stored in D_t, so the policy can revisit evidence before promoting, demoting, removing, or verifying it, without another corpus search.

2.2 DERIVED-STATE RENDERING: EVIDENCE GRAPH, COMPRESSION, AND DEDUPLICATION

Beyond storing retrieved text, the harness derives compact signals from the search history and renders them into each observation, so the policy sees structured state rather than accumulated raw tool outputs.

Evidence graph. The evidence graph G_t summarizes cross-document structure. Each chunk entering P_t is scanned with a lightweight regex extractor for three entity classes: multi-word capitalized proper nouns, four-digit years or decades, and numeric dates. The harness maintains two maps, `entity_to_docs` and `doc_to_entities`, and renders a compact block with frequent entities (the top entities by document count), bridge documents, and singleton entities. Bridge documents contain two or more frequent entities and are natural candidates for `verify` or high-importance promotion. Singleton entities appear in only one document and often help follow-up search.

```
[Evidence Graph] Entities appearing in
multiple docs (bridges):
  Brussels:  22816_0, 91442_3, 62390_5
  1878:  22816_0, 11293_2
  Grande Synagogue:  22816_0, 91442_3
  (2 singleton entities, potential hops)
```

In the example, document 22816_0 connects all three rendered entities and 91442_3 connects two, so the policy can treat both as candidate bridges without rereading the full transcript. A question such as "what have I seen about entity X?" becomes a direct lookup in working memory instead of a full-context reread. The harness builds the graph with one regex pass per chunk and a top-k scan at render time. The policy decides which bridge to verify, which singleton to pursue, and which relation matters for the query.

Result summaries and budget markers. After each retrieval or memory action, the harness records the tool used, the number of returned and novel documents, updates to P_t and C_t, and the current context budget B_t. Thus the policy observes search progress and remaining budget as explicit state. **Sentence-BM25 compression.** To keep search observations from dominating the context, the harness compresses outputs from `search_corpus`, `fan_out_search`, and `grep_corpus`. It scores sentences with BM25 against the query and renders the top $K{=}4$ in their original order. Explicit `read_document` calls still return full text from D_t. A small embedding model would keep somewhat more answer text (Appendix M.1.2). **Two-level deduplication.** `Harness-1` removes repeated evidence by both chunk ID and content fingerprint. Near-duplicates are omitted from the rendered state, while reward accounting still credits all relevant evidence encountered during the trajectory. The same budgeted context builder is used for teacher generation, SFT replay, RL rollout, and evaluation. It combines the system prompt, rendered working memory, recent actions and observations, and result summaries, while truncating older reasoning and clipping long observations as needed.

2.3 Training: SFT and RL

Training follows the same division of labor as the harness. SFT teaches the model to operate the interface, and RL improves search decisions over the state that the interface maintains.[1]

SFT for harness operation. The role of SFT is deliberately narrow. Because the harness carries much of the bookkeeping, the supervised prior only needs to teach the tool-call format, the search→curate pattern, disciplined importance tagging, memory review, and the `verify`-before-promote norm. A single teacher, GPT-5.4, runs as a live agent inside the full `Harness-1` harness, using the same tools and working-memory rendering later shown to the student. Turn-level guidance encourages curation after search, verification before high-confidence promotion, backtracking after low-yield searches, and early termination once the curated set is sufficient. We keep trajectories that are format-valid, return at least one document, and achieve final curated recall ≥ 0.10. Each retained trajectory is replayed through the same WorkingMemory and context builder used by RL, then expanded into one supervised datum per turn. SFT runs on `gpt-oss-20b` via LoRA with rank 32 for 3 epochs; the step-550 checkpoint initializes RL.

RL over full search episodes. Starting from the SFT checkpoint, we train with on-policy CISPO (Chen et al., 2025a; Khatri et al., 2025) and within-group advantage normalization on SEC training queries. RL uses full-trajectory rollouts, terminal-only reward, a 40-turn cap, and no KL anchor. Groups whose rollouts receive identical rewards are dropped from the gradient. Full rollout scale and optimization hyperparameters are in Appendix D.

Reward. The terminal reward combines search-quality signals, answer-evidence shaping, a tool-diversity bonus, and a turn penalty. Episodes that terminate with an empty curated set short-circuit to $\pi_\emptyset = -0.2$; otherwise,

$$\mathcal{R} = \underbrace{w_F\, F_\beta}_{\text{set quality}} + \underbrace{w_\tau\, \rho_\tau}_{\text{trajectory coverage}} + \underbrace{w_{\mathrm{A}}\, \rho_{\mathrm{A}} + w_{\tau\mathrm{A}}\, \rho_{\tau\mathrm{A}}}_{\text{answer evidence}} + \underbrace{B_{\mathrm{A}}\, \mathbf{1}[\rho_{\mathrm{A}} > 0]}_{\text{answer bonus}} + \underbrace{w_{\mathrm{div}}\, \min(\nu/\nu_0,\, 1)}_{\text{tool diversity}} - \underbrace{w_{\mathrm{miss}}\, (\rho_{\tau\mathrm{A}} - \rho_{\mathrm{A}})_+}_{\text{answer miss}} - \underbrace{\pi_{\mathrm{turn}}(t)}_{\text{turn penalty}}.$$

For any successfully formatted episode, the reward is clipped below by $\mathcal{R} \geq 10^{-3}$. Here ρ_τ is trajectory recall, ρ_{A} is curated final-answer recall, $\rho_{\tau\mathrm{A}}$ is trajectory final-answer recall, ν is the number of distinct tools used, and ν_0 is the diversity target. F_β uses $\beta{=}2$, weighting recall four times more than precision. The reward separates discovery from selection. Trajectory terms credit evidence found anywhere in the pool, curated terms credit evidence selected for the final output, and the answer-miss term penalizes trajectories that found answer evidence but did not promote it. Appendix D lists all hyperparameters and code variables.

[1] We use Tinker (https://thinkingmachines.ai/tinker/) for model training.

Table 2: **Search quality across benchmarks.** We report curated-set Recall, Final-Answer Recall, and Trajectory Recall. Green and blue rows mark the best open-source small model and the best frontier model overall. Results are averaged over three runs. †Fable-5 (Claude Fable 5) is a single run, and queries declined by its safety classifier count as recall 0.

Model	Metric	BC+	Web	Patents	SEC	LongSeal	Seal0	FRAMES	HotpotQA
Open-source small models									
`Harness-1` (20B)	Recall	0.584	0.787	0.890	0.589	0.891	0.551	0.710	0.841
	Final-Answer Recall	0.667	0.845	0.890	0.659	0.891	0.551	0.710	0.841
	Trajectory Recall	0.665	0.881	0.973	0.677	0.944	0.708	0.749	0.860
Context-1 (20B)	Recall	0.500	0.689	0.857	0.467	0.563	0.367	0.638	0.746
	Final-Answer Recall	0.618	0.775	0.857	0.498	0.563	0.367	0.638	0.746
	Trajectory Recall	0.567	0.823	0.975	0.507	0.858	0.697	0.787	0.835
GPT-OSS-20B	Recall	0.109	0.174	0.393	0.133	0.253	0.250	0.391	0.395
	Final-Answer Recall	0.156	0.252	0.393	0.166	0.253	0.250	0.391	0.395
	Trajectory Recall	0.246	0.512	0.742	0.252	0.816	0.632	0.702	0.819
Tongyi DeepResearch 30B	Recall	0.381	0.607	0.780	0.419	0.855	0.577	0.561	0.751
	Final-Answer Recall	0.428	0.679	0.780	0.433	0.855	0.577	0.561	0.751
	Trajectory Recall	0.413	0.722	0.780	0.478	0.855	0.631	0.654	0.852
Search-R1 (32B)	Recall	0.053	0.135	0.405	0.109	0.514	0.312	0.329	0.454
	Final-Answer Recall	0.074	0.240	0.405	0.155	0.514	0.312	0.329	0.454
	Trajectory Recall	0.053	0.135	0.405	0.109	0.514	0.312	0.329	0.454
Qwen3 (32B)	Recall	0.072	0.132	0.257	0.058	0.427	0.279	0.164	0.342
	Final-Answer Recall	0.092	0.211	0.257	0.072	0.427	0.279	0.164	0.342
	Trajectory Recall	0.154	0.410	0.577	0.215	0.704	0.568	0.402	0.535
Frontier models									
Fable-5†	Recall	0.719	0.894	0.790	0.744	0.796	0.613	0.746	0.863
	Final-Answer Recall	0.793	0.962	0.790	0.827	0.796	0.613	0.746	0.863
	Trajectory Recall	0.758	0.916	0.830	0.778	0.851	0.667	0.810	0.882
Opus-4.6	Recall	0.619	0.819	0.915	0.589	0.823	0.636	0.814	0.893
	Final-Answer Recall	0.736	0.928	0.915	0.696	0.823	0.636	0.814	0.893
	Trajectory Recall	0.634	0.832	0.939	0.601	0.876	0.682	0.878	0.910
GPT-5.4	Recall	0.443	0.800	0.909	0.452	0.824	0.555	0.815	0.870
	Final-Answer Recall	0.499	0.858	0.909	0.465	0.824	0.555	0.815	0.870
	Trajectory Recall	0.476	0.821	0.919	0.464	0.885	0.691	0.865	0.898
Sonnet-4.6	Recall	0.505	0.744	0.832	0.411	0.783	0.577	0.787	0.863
	Final-Answer Recall	0.597	0.892	0.832	0.480	0.783	0.577	0.787	0.863
	Trajectory Recall	0.535	0.767	0.845	0.421	0.840	0.658	0.859	0.878
Kimi-K2.5	Recall	0.593	0.707	0.833	0.454	0.661	0.455	0.697	0.776
	Final-Answer Recall	0.670	0.843	0.833	0.478	0.661	0.455	0.697	0.776
	Trajectory Recall	0.739	0.878	0.945	0.549	0.911	0.655	0.840	0.837
GPT-OSS-120B	Recall	0.508	0.478	0.570	0.395	0.464	0.400	0.495	0.657
	Final-Answer Recall	0.641	0.664	0.570	0.504	0.464	0.400	0.495	0.657
	Trajectory Recall	0.682	0.796	0.824	0.488	0.914	0.689	0.863	0.897

3 Experiments

3.1 Experimental Setup

Benchmarks & Data. We evaluate on eight retrieval benchmarks: BrowseComp+ (Chen et al., 2025c), *Web* synthetic, *Patents* from USPTO office actions, *SEC* filings (Bashir et al., 2026), LongSealQA and Seal0QA (Pham et al., 2025), FRAMES (Krishna et al., 2025), and HotpotQA (Yang et al., 2018). They span web, finance, patents, and multi-hop QA, and use both Chroma-backed corpora and a shared Serper+Jina web backend. `Harness-1`'s SFT data covers four benchmark families (BC+, Web, Patents, SEC), and RL uses only SEC data. These four are the source-family benchmarks. The other four (LongSealQA, Seal0QA, FRAMES, HotpotQA) are not used in `Harness-1` SFT or RL and serve as held-out transfer benchmarks. Per-benchmark statistics are in Appendix E. The raw SFT quotas are BC+/300, SEC/250, Patents/150, Web/150, Web-simple/75, and SEC-simple/75. After filtering, 899 trajectories remain. RL uses the SEC training split (3,453 queries).[2]

Search-quality metrics. We evaluate search quality by evidence coverage. *Recall* measures coverage of all annotated relevant documents in the final curated set. *Trajectory Recall* measures the same coverage over every document encountered during the episode, before final selection. *Final-Answer Recall* measures coverage of the gold answer documents only. Formal definitions, including

[2] https://www.trychroma.com/research/context-1

how answer documents differ from supporting documents, are in Appendix N. We focus on recall because the search agent's job is to collect the evidence needed for answering. Precision and F_1 are secondary because curated-set size is part of the agent interface.

Evaluation protocol and baselines. We evaluate on the benchmark splits summarized in Table 8. All methods use the same retrieval primitives, web backend, Qwen3-Reranker-8B reranker, and final budget of at most 30 returned documents, following Context-1 (Bashir et al., 2026). `Harness-1` runs at temperature 1.0 with a 40-turn cap. Context-1 and the proprietary frontier LLMs used as zero-shot retrievers (GPT-5.4, Sonnet-4.6, Opus-4.6, Kimi-K2.5, Fable-5) run in the Context-1 harness at temperature 1.0 (where the API exposes it) with a 64-turn cap. Fable-5 is a single run, and queries that its safety classifier declines are scored as recall 0. Appendix P runs GPT-5.4 and Opus-4.6 in both harnesses on the four source-family benchmarks. Our harness raises the recall of both models. In the same harness, `Harness-1` still beats GPT-5.4 on average (0.713 vs. 0.691), and Opus-4.6 stays ahead (0.769). Open-model baselines include base `gpt-oss-20b`/`gpt-oss-120b`, Qwen3-32B, Context-1, Search-R1 32B (Jin et al., 2025), and Tongyi DeepResearch 30B (Team et al., 2025). For Search-R1 and Tongyi DeepResearch, we use their released agent harnesses with the shared search and reranking APIs. These harnesses do not produce a curated set capped at 30 documents, so we rerank their output and keep the top 30 (Appendix N). This step never removes a relevant document from Search-R1's output, and the margin over Tongyi DeepResearch holds without it. Full evaluation details are in Appendices N and E.

3.2 Results

Table 2 reports Recall, Final-Answer Recall, and Trajectory Recall on all eight benchmarks. `Harness-1` is the strongest open retrieval subagent we evaluate. Its average curated recall is 0.730, +11.4 points above the next open subagent, Tongyi DeepResearch 30B. Under this protocol, it has higher average curated recall than the much larger frontier retrievers GPT-5.4, Sonnet-4.6, Kimi-K2.5, and GPT-OSS-120B, and trails only Fable-5 and Opus-4.6. It is also far cheaper to run. An episode of `Harness-1` costs about $0.03, against $0.15 to $0.74 for GPT-5.4, Sonnet-4.6, Kimi-K2.5, and Opus-4.6. `Harness-1` reaches 97% of Opus-4.6's source-family recall at about 4% of its cost (Figure 7).

Gains are larger on held-out benchmarks. Where the gains occur tells us more than their size. `Harness-1`'s SFT stage uses BC+, Web, Patents, and SEC trajectories, so the stricter transfer test is the four benchmarks excluded from both `Harness-1` SFT and RL. If the gains came mainly from replaying source-family behaviors, the advantage should shrink on these held-out tasks, but it grows.

On the four source-family benchmarks (BC+, Web, Patents, SEC), `Harness-1` gains +8.4, +9.8, +3.3, and +12.2 recall points (mean +8.4) over Context-1, the closest open baseline on these benchmarks. On the four held-out transfer benchmarks (LongSealQA, Seal0QA, FRAMES, HotpotQA), it gains +32.8, +18.4, +7.2, and +9.5 points over Context-1 (mean +17.0). The gain is thus **2.0×** larger on the benchmarks furthest from any training data (Figure 3). This is because the policy learns operations over a domain-general search state. It refines the auto-seeded set, reads bridge entities from the evidence graph, re-inspects uncertain candidates, verifies before promotion, and submits a compact curated set. These operations transfer more readily than domain-specific patterns stored in weights. The FRAMES and HotpotQA trajectories in Appendices Q.3 and Q.4 show them outside the SFT/RL families. Fan-out or targeted search proposes candidate bridges, and reading, exact matching, and curation then pin down the final evidence.

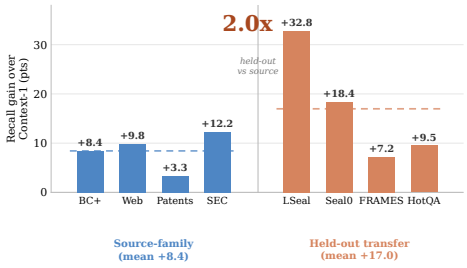

Figure 3: **Transfer pattern.** `Harness-1` gains more on held-out transfer benchmarks (+17.0 pts mean) than on source-family benchmarks (+8.4 pts), a 2.0× gap over Context-1.

Data scale. More training data does not explain the transfer pattern. `Harness-1` uses 899 filtered SFT trajectories and RL on the SEC train split (3,453 queries), 4,352 unique training items in total. Context-1 reports over 8K synthetic SFT tasks, and its RL trainer uses 9,159 unique queries (four

Table 3: **Inference-time component ablation on BrowseComp+** (same trained `Harness-1` checkpoint). Each row disables exactly one harness mechanism. ΔRecall (%) and ΔFA (%) report the *relative* change versus full `Harness-1`, i.e. $100 \times (\text{ablated} - \text{full})/\text{full}$. The rightmost column links to each mechanism's failure mode and an example trajectory.

Configuration	Recall	ΔRecall (%)	FA Recall	ΔFA (%)	Failure mode
Full `Harness-1` (all mechanisms enabled)	0.584	—	0.667	—	—
− Importance tags (binary `curate`, FIFO eviction)	0.560	−4.1	0.614	−7.9	§M.1.1
− Sentence-BM25 compression (raw chunks)	0.585	+0.2	0.620	−7.0	§M.1.2
− Auto-seed on first search	0.582	−0.3	0.624	−6.4	§M.1.3
− Evidence graph hidden in observations	0.569	−2.6	0.631	−5.4	§M.1.4
− `verify` returns "unavailable"	0.566	−3.1	0.641	−3.9	§M.1.5
− `review_docs` returns "unavailable"	0.598	+2.4	0.641	−3.9	§M.1.6
− Content-fingerprint dedup (chunk-ID dedup kept)	0.611	+4.6	0.678	+1.6	§M.1.7
All `Harness-1` harness mechanisms disabled	0.513	−12.2	0.624	−6.4	§M.2

full train splits totaling 8,859, plus two capped 150-query curriculum splits). Search-R1 has no SFT warmup and trains RL on the merged NQ+HotpotQA set (221,328 rows). Figure 4 compares unique task/query counts, not rollout count or token compute. We do not claim that less data always suffices. Our point is that `Harness-1` moves much of the behavioral prior into the stateful interface, so small SFT and focused RL can transfer. Most of the gain comes from RL. The SFT checkpoint that starts RL reaches 0.226 to 0.505 curated recall on the source-family benchmarks, and RL raises this to 0.584 to 0.890. The gain appears even on BC+, Web, and Patents, which RL never trains on (Appendix D).

Inference-time component ablation. To see what each part of the harness contributes, we run the trained `Harness-1` checkpoint on BrowseComp+ test queries and switch off one harness mechanism at a time, without retraining (Table 3). `verify` and `review_docs` are switched off by making them return "unavailable." The last row switches off all mechanisms at once.

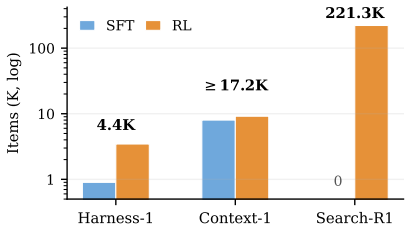

Figure 4: **Training-data scale.** Paired bars show reported unique task/query units used for SFT and RL.

Almost every mechanism matters. Six of the seven mechanisms lower final-answer recall when removed, by about 4% to 8% relative. The failures look alike. On queries that the full system solves and the ablated system misses, the policy searches more and reads and verifies much less. Without the state it was trained with, it falls back to broad, shallow searching and never settles on the right documents. The largest final-answer drops come from importance tags (§M.1.1), sentence compression (§M.1.2), auto-seeding (§M.1.3), and the evidence graph (§M.1.4).

Content deduplication is the exception. Turning it off raises recall slightly. Some BC+ queries have near-duplicate gold documents, and deduplication sometimes keeps only one of them, so the curated set loses a gold ID but no information. Deduplication is there to save context for the downstream answerer, not to raise recall, and we keep it for that reason (§M.1.7).

Removing everything hurts recall most. Switching off all mechanisms at once gives the largest recall drop in the table, from 0.584 to 0.513. The policy still explores, but without the harness it cannot turn what it finds into a focused curated set (§M.2).

Discovery and selection. Trajectory recall shows that `Harness-1` usually discovers the relevant evidence. Across domains, the harness surfaces a large pool of gold documents before final curation. On BC+, `Harness-1` trails Opus-4.6 in final-answer recall (0.667 vs. 0.736) despite similar trajectory recall, so this gap comes mainly from selection, not discovery. The gap to Fable-5 (0.793) also has a discovery part, since Fable-5's trajectory recall is 0.758. The examples in Appendices Q.1 and Q.2 show that the right document often appears early, and that keeping it in the final curated set depends on later reading and promotion.

Training dynamics. Figure 5 compares two RL runs from the same SFT checkpoint with the same hyperparameters. They differ only in whether the tool-diversity reward w_{div} is active. The comparison shows that the harness alone is not enough. Without RL-compatible shaping, the policy collapses to a narrow, search-heavy strategy.

Without w_{div}, the agent quickly learns to issue many `fan_out_search` calls, which improves trajectory recall but bypasses curation and verification. Tool diversity falls from $\sim$6 to $\sim$3.5, and curated recall plateaus at $\sim$0.53. The agent finds relevant documents, but does not reliably promote the right ones into the final output set.

With w_{div} active, tool use stays broader. Diversity stabilizes around $\sim$4.30, and the agent regularly combines search with curation and verification. Early recall rises more slowly because the policy spends more turns editing the candidate set, but final curated recall is higher ($\sim$0.60). This shows why the third RL-compatibility requirement (diversity-preserving incentives) is needed. Exposing useful tools is not enough. The reward must also make them learnable and worth using, or RL takes the easiest reward path back toward a minimal search-only harness.

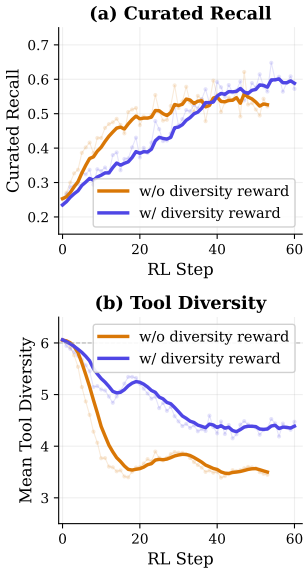

Figure 5: **Training dynamics.** **(a)** Recall during RL training. **(b)** Mean tool diversity. Without the diversity bonus, recall plateaus and tool use collapses; with the bonus, diversity stabilizes and final recall is higher.

Modular RAG answer accuracy. The search gains also translate to downstream answer quality. When each subagent's curated set is passed to frozen frontier generators, better curated sets usually yield higher answer accuracy, and `Harness-1` matches or beats Context-1, the other open subagent in this study, for every generator. Full plots and protocol details are in Appendix O.

4 Conclusion

We introduced `Harness-1`, a 20B search agent trained with reinforcement learning inside a stateful harness. `Harness-1` separates semantic search decisions from recoverable bookkeeping. The policy decides what to search, curate, verify, and submit, and the harness maintains candidate pools, importance-tagged evidence, verification records, evidence links, and budget-aware context rendering. Across eight benchmarks, `Harness-1` achieves the strongest average recall among the open search agents we evaluate and beats four of the six much larger frontier-model searchers. The held-out transfer gains and the component ablations suggest that the harness is a central part of what the policy learns to use. The harness swap in Appendix P points the same way. Our harness raises the recall of both frontier models we test, and the trained `Harness-1` policy does much worse in Context-1's harness than in its own, so the result comes from the policy and the harness together.

These results point to stateful harness design as an important direction for retrieval-agent RL. Future work could improve the evidence graph, for example by replacing regex-based extraction with learned entity linking, relation extraction, and uncertainty-aware evidence organization. We discuss connections to agentic search, tool orchestration, and harness engineering in Appendix A, and limitations, ethics, and broader impact in Appendix B.

AI Use Statement

Generative AI models are part of the system we study and of its evaluation, and we describe each use where it occurs. A frontier model generates the teacher trajectories used for supervised fine-tuning (Appendix J). An LLM-based verifier implements the `verify` tool (Appendix I.2). Frontier models serve as zero-shot retrieval baselines and as frozen answer generators in the modular RAG protocol (Section 3.1 and Appendix N). An LLM judge scores answer accuracy against gold answers (Appendix I.4). These outputs are experimental artifacts that we evaluate against human-annotated benchmark labels. Beyond the studied system, we used AI coding assistants to help implement, de-

bug, and test parts of the harness, training, evaluation, and analysis code. We did not use generative AI tools to generate the research ideas or hypotheses, or to design the method or the experiments. All AI-assisted code was reviewed and tested by the authors. We also used generative AI tools to polish the manuscript for clarity and readability, and all AI-assisted text was reviewed and revised by the authors. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

Ethics Statement

This work trains and evaluates a retrieval subagent on publicly available benchmarks and corpora (encyclopedic web pages, USPTO office actions, SEC filings, and Wikipedia-derived question answering). It involves no human subjects, personal-data collection, or annotation studies. The main risks are misuse and over-reliance. A stronger search agent connected to an unrestricted corpus could retrieve sensitive or private information or amplify misleading evidence, and its curated set should not be treated as a source of truth. Deployment should therefore pair the agent with corpus access controls, logging, and human review in high-stakes domains. Our planned release is limited to the retrieval subagent, harness code, data-generation pipeline, RL recipe, and evaluation recipe, with documentation of intended use and dataset provenance. A fuller discussion of limitations, ethics, and broader impact is given in Appendix B.

Reproducibility Statement

Section 2 specifies the harness state, actions, and rendering. The appendices give the formal algorithms (Appendix H), tool signatures (Appendix C), the exact working-memory layout (Appendix F), per-turn result summaries (Appendix G), and importance-aware curation and content deduplication (Appendices K and L). All prompts are reproduced in Appendix I, the SFT data-generation recipe and filtering thresholds in Appendix J, and reward weights and training hyperparameters in Appendix D. Evaluation is described in Section 3.1, with metric definitions, benchmark statistics, the treatment of released baseline harnesses, and the modular RAG protocol in Appendices N, E, and O. The RL training queries come from a public dataset (Section 3.1). Per-query trajectories are saved for every run so that all reported metrics can be recomputed. We will release the model weights, harness code, data-generation pipeline, RL recipe, and evaluation scripts upon publication.

References

Cognition AI. Introducing swe-grep and swe-grep-mini: Rl for multi-turn, fast context retrieval. Cognition technical report, 2025.

Hammad Bashir, Kelly Hong, Patrick Jiang, and Zhiyi Shi. Chroma context-1: Training a self-editing search agent. Technical report, Chroma, March 2026. URL https://trychroma.com/research/context-1.

Aili Chen, Aonian Li, Bangwei Gong, Binyang Jiang, Bo Fei, Bo Yang, Boji Shan, Changqing Yu, Chao Wang, Cheng Zhu, et al. Minimax-m1: Scaling test-time compute efficiently with lightning attention. *arXiv preprint arXiv:2506.13585*, 2025a.

Kun Chen, Qingchao Kong, Zhao Feifei, and Wenji Mao. Apex-searcher: Augmenting llms' search capabilities through agentic planning and execution, 2026. URL https://arxiv.org/abs/2603.13853.

Mingyang Chen, Linzhuang Sun, Tianpeng Li, Haoze Sun, Yijie Zhou, Chenzheng Zhu, Haofen Wang, Jeff Z. Pan, Wen Zhang, Huajun Chen, Fan Yang, Zenan Zhou, and Weipeng Chen. Research: Learning to reason with search for llms via reinforcement learning, 2025b. URL https://arxiv.org/abs/2503.19470.

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, 2025c. URL https://arxiv.org/abs/2508.06600.

Yong Deng, Guoqing Wang, Zhenzhe Ying, Xiaofeng Wu, Jinzhen Lin, Wenwen Xiong, Yuqin Dai, Shuo Yang, Zhanwei Zhang, Qiwen Wang, Yang Qin, Yuan Wang, Quanxing Zha, Sunhao Dai, and Changhua Meng. Atom-searcher: Enhancing agentic deep research via fine-grained atomic thought reward, 2025. URL https://arxiv.org/abs/2508.12800.

Jiaxuan Gao, Wei Fu, Minyang Xie, Shusheng Xu, Chuyi He, Zhiyu Mei, Banghua Zhu, and Yi Wu. Beyond ten turns: Unlocking long-horizon agentic search with large-scale asynchronous rl, 2025. URL https://arxiv.org/abs/2508.07976.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, Bing Xue, Bingxuan Wang, Bochao Wu, Bei Feng, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chong Ruan, Damai Dai, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Hanwei Xu, Honghui Ding, Huazuo Gao, Hui Qu, Hui Li, Jianzhong Guo, Jiashi Li, Jingchang Chen, Jingyang Yuan, Jinhao Tu, Junjie Qiu, Junlong Li, J. L. Cai, Jiaqi Ni, Jian Liang, Jin Chen, Kai Dong, Kai Hu, Kaichao You, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Liang Zhao, Litong Wang, Liyue Zhang, Lei Xu, Leyi Xia, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingxu Zhou, Meng Li, Miaojun Wang, Mingming Li, Ning Tian, Panpan Huang, Peng Zhang, Qiancheng Wang, Qinyu Chen, Qiushi Du, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, R. J. Chen, R. L. Jin, Ruyi Chen, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shengfeng Ye, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, S. S. Li, Shuang Zhou, Shaoqing Wu, Tao Yun, Tian Pei, Tianyu Sun, T. Wang, Wangding Zeng, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, W. L. Xiao, Wei An, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xiaosha Chen, Xiaowen Sun, Xiaoxiang Wang, Xinnan Song, Xinyi Zhou, Xianzu Wang, Xinxia Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Yang Zhang, Yanhong Xu, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Yu, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yuan Ou, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Y. X. Zhu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Ying Tang, Yukun Zha, Yuting Yan, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhicheng Ma, Zhigang Yan, Zhiyu Wu, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Zizheng Pan, Zhen Huang, Zhipeng Xu, Zhongyu Zhang, and Zhen Zhang. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645(8081):633–638, 2025. ISSN 1476-4687. doi: 10.1038/s41586-025-09422-z. URL http://dx.doi.org/10.1038/s41586-025-09422-z.

Pengrui Han, Rafal Kocielnik, Peiyang Song, Ramit Debnath, Dean Mobbs, Anima Anandkumar, and R. Michael Alvarez. The personality illusion: Revealing dissociation between self-reports & behavior in llms, 2025. URL https://arxiv.org/abs/2509.03730.

Pengrui Han, Xueqiang Xu, Keyang Xuan, Peiyang Song, Siru Ouyang, Runchu Tian, Yuqing Jiang, Cheng Qian, Pengcheng Jiang, Jiashuo Sun, Junxia Cui, Ming Zhong, Ge Liu, Jiawei Han, and Jiaxuan You. Steer2adapt: Dynamically composing steering vectors elicits efficient adaptation of llms, 2026. URL https://arxiv.org/abs/2602.07276.

Kelly Hong, Anton Troynikov, and Jeff Huber. Context rot: How increasing input tokens impacts llm performance. Technical report, Chroma, July 2025. URL https://trychroma.com/research/context-rot.

Jian Hu, Jason Klein Liu, Haotian Xu, and Wei Shen. Reinforce++: Stabilizing critic-free policy optimization with global advantage normalization, 2025. URL https://arxiv.org/abs/2501.03262.

Pengcheng Jiang, Jiacheng Lin, Lang Cao, Runchu Tian, SeongKu Kang, Zifeng Wang, Jimeng Sun, and Jiawei Han. Deepretrieval: Hacking real search engines and retrievers with large language models via reinforcement learning. *arXiv preprint arXiv:2503.00223*, 2025a.

Pengcheng Jiang, Xueqiang Xu, Jiacheng Lin, Jinfeng Xiao, Zifeng Wang, Jimeng Sun, and Jiawei Han. s3: You don't need that much data to train a search agent via RL. In Christos

Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 21599–21617, Suzhou, China, November 2025b. Association for Computational Linguistics. ISBN 979-8-89176-332-6. doi: 10.18653/v1/2025.emnlp-main.1095. URL https://aclanthology.org/2025.emnlp-main.1095/.

Pengcheng Jiang, Jiacheng Lin, Zhiyi Shi, Zifeng Wang, Luxi He, Yichen Wu, Ming Zhong, Peiyang Song, Qizheng Zhang, Heng Wang, Xueqiang Xu, Hanwen Xu, Pengrui Han, Dylan Zhang, Jiashuo Sun, Chaoqi Yang, Kun Qian, Tian Wang, Changran Hu, Manling Li, Quanzheng Li, Hao Peng, Sheng Wang, Jingbo Shang, Chao Zhang, Jiaxuan You, Liyuan Liu, Pan Lu, Yu Zhang, Heng Ji, Yejin Choi, Dawn Song, Jimeng Sun, and Jiawei Han. Adaptation of agentic ai: A survey of post-training, memory, and skills, 2026. URL https://arxiv.org/abs/2512.16301.

Zhengbao Jiang, Frank F Xu, Luyu Gao, Zhiqing Sun, Qian Liu, Jane Dwivedi-Yu, Yiming Yang, Jamie Callan, and Graham Neubig. Active retrieval augmented generation. In *Proceedings of the 2023 conference on empirical methods in natural language processing*, pp. 7969–7992, 2023.

Bowen Jin, Hansi Zeng, Zhenrui Yue, Jinsung Yoon, Sercan Arik, Dong Wang, Hamed Zamani, and Jiawei Han. Search-r1: Training llms to reason and leverage search engines with reinforcement learning. *arXiv preprint arXiv:2503.09516*, 2025.

Devvrit Khatri, Lovish Madaan, Rishabh Tiwari, Rachit Bansal, Sai Surya Duvvuri, Manzil Zaheer, Inderjit S. Dhillon, David Brandfonbrener, and Rishabh Agarwal. The art of scaling reinforcement learning compute for llms, 2025. URL https://arxiv.org/abs/2510.13786.

Satyapriya Krishna, Kalpesh Krishna, Anhad Mohananey, Steven Schwarcz, Adam Stambler, Shyam Upadhyay, and Manaal Faruqui. Fact, fetch, and reason: A unified evaluation of retrieval-augmented generation. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 4745–4759, 2025.

Tamera Lanham, Anna Chen, Ansh Radhakrishnan, Benoit Steiner, Carson Denison, Danny Hernandez, Dustin Li, Esin Durmus, Evan Hubinger, Jackson Kernion, Kamilė Lukošiūtė, Karina Nguyen, Newton Cheng, Nicholas Joseph, Nicholas Schiefer, Oliver Rausch, Robin Larson, Sam McCandlish, Sandipan Kundu, Saurav Kadavath, Shannon Yang, Thomas Henighan, Timothy Maxwell, Timothy Telleen-Lawton, Tristan Hume, Zac Hatfield-Dodds, Jared Kaplan, Jan Brauner, Samuel R. Bowman, and Ethan Perez. Measuring faithfulness in chain-of-thought reasoning, 2023. URL https://arxiv.org/abs/2307.13702.

Yoonho Lee, Roshen Nair, Qizheng Zhang, Kangwook Lee, Omar Khattab, and Chelsea Finn. Meta-harness: End-to-end optimization of model harnesses. *arXiv preprint arXiv:2603.28052*, 2026.

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 superhuman reasoning for web agent, 2025a. URL https://arxiv.org/abs/2507.02592.

Xiaoxi Li, Guanting Dong, Jiajie Jin, Yuyao Zhang, Yujia Zhou, Yutao Zhu, Peitian Zhang, and Zhicheng Dou. Search-o1: Agentic search-enhanced large reasoning models. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 5420–5438, Suzhou, China, November 2025b. Association for Computational Linguistics. ISBN 979-8-89176-332-6. doi: 10.18653/v1/2025.emnlp-main.276. URL https://aclanthology.org/2025.emnlp-main.276/.

Junteng Liu, Yunji Li, Chi Zhang, Jingyang Li, Aili Chen, Ke Ji, Weiyu Cheng, Zijia Wu, Chengyu Du, Qidi Xu, Jiayuan Song, Zhengmao Zhu, Wenhu Chen, Pengyu Zhao, and Junxian He. Webexplorer: Explore and evolve for training long-horizon web agents, 2025. URL https://arxiv.org/abs/2509.06501.

Ryan Lopopolo. Harness engineering: leveraging codex in an agent-first world, 2026.

Xinghua Lou, Miguel Lázaro-Gredilla, Antoine Dedieu, Carter Wendelken, Wolfgang Lehrach, and Kevin P. Murphy. Autoharness: improving llm agents by automatically synthesizing a code harness, 2026. URL https://arxiv.org/abs/2603.03329.

Lance Martin, Gabe Cemaj, and Michael Cohen. Scaling managed agents: Decoupling the brain from the hands. https://www.anthropic.com/engineering/managed-agents, April 2026. Anthropic Engineering Blog.

OpenAI, :, Sandhini Agarwal, Lama Ahmad, Jason Ai, Sam Altman, Andy Applebaum, Edwin Arbus, Rahul K. Arora, Yu Bai, Bowen Baker, Haiming Bao, Boaz Barak, Ally Bennett, Tyler Bertao, Nivedita Brett, Eugene Brevdo, Greg Brockman, Sebastien Bubeck, Che Chang, Kai Chen, Mark Chen, Enoch Cheung, Aidan Clark, Dan Cook, Marat Dukhan, Casey Dvorak, Kevin Fives, Vlad Fomenko, Timur Garipov, Kristian Georgiev, Mia Glaese, Tarun Gogineni, Adam Goucher, Lukas Gross, Katia Gil Guzman, John Hallman, Jackie Hehir, Johannes Heidecke, Alec Helyar, Haitang Hu, Romain Huet, Jacob Huh, Saachi Jain, Zach Johnson, Chris Koch, Irina Kofman, Dominik Kundel, Jason Kwon, Volodymyr Kyrylov, Elaine Ya Le, Guillaume Leclerc, James Park Lennon, Scott Lessans, Mario Lezcano-Casado, Yuanzhi Li, Zhuohan Li, Ji Lin, Jordan Liss, Lily, Liu, Jiancheng Liu, Kevin Lu, Chris Lu, Zoran Martinovic, Lindsay McCallum, Josh McGrath, Scott McKinney, Aidan McLaughlin, Song Mei, Steve Mostovoy, Tong Mu, Gideon Myles, Alexander Neitz, Alex Nichol, Jakub Pachocki, Alex Paino, Dana Palmie, Ashley Pantuliano, Giambattista Parascandolo, Jongsoo Park, Leher Pathak, Carolina Paz, Ludovic Peran, Dmitry Pimenov, Michelle Pokrass, Elizabeth Proehl, Huida Qiu, Gaby Raila, Filippo Raso, Hongyu Ren, Kimmy Richardson, David Robinson, Bob Rotsted, Hadi Salman, Suvansh Sanjeev, Max Schwarzer, D. Sculley, Harshit Sikchi, Kendal Simon, Karan Singhal, Yang Song, Dane Stuckey, Zhiqing Sun, Philippe Tillet, Sam Toizer, Foivos Tsimpourlas, Nikhil Vyas, Eric Wallace, Xin Wang, Miles Wang, Olivia Watkins, Kevin Weil, Amy Wendling, Kevin Whinnery, Cedric Whitney, Hannah Wong, Lin Yang, Yu Yang, Michihiro Yasunaga, Kristen Ying, Wojciech Zaremba, Wenting Zhan, Cyril Zhang, Brian Zhang, Eddie Zhang, and Shengjia Zhao. gpt-oss-120b & gpt-oss-20b model card, 2025. URL https://arxiv.org/abs/2508.10925.

Charles Packer, Sarah Wooders, Kevin Lin, Vivian Fang, Shishir G. Patil, Ion Stoica, and Joseph E. Gonzalez. Memgpt: Towards llms as operating systems, 2024. URL https://arxiv.org/abs/2310.08560.

Thinh Pham, Nguyen Nguyen, Pratibha Zunjare, Weiyuan Chen, Yu-Min Tseng, and Tu Vu. Sealqa: Raising the bar for reasoning in search-augmented language models. *arXiv preprint arXiv:2506.01062*, 2025.

Ofir Press, Muru Zhang, Sewon Min, Ludwig Schmidt, Noah Smith, and Mike Lewis. Measuring and narrowing the compositionality gap in language models. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 5687–5711, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-emnlp.378. URL https://aclanthology.org/2023.findings-emnlp.378/.

Yujia Qin, Shihao Liang, Yining Ye, Kunlun Zhu, Lan Yan, Yaxi Lu, Yankai Lin, Xin Cong, Xiangru Tang, Bill Qian, Sihan Zhao, Lauren Hong, Runchu Tian, Ruobing Xie, Jie Zhou, Mark Gerstein, Dahai Li, Zhiyuan Liu, and Maosong Sun. Toolllm: Facilitating large language models to master 16000+ real-world apis, 2023. URL https://arxiv.org/abs/2307.16789.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms, 2017. URL https://arxiv.org/abs/1707.06347.

Biswa Sengupta and Jinhua Wang. Harbor: Automated harness optimization. *arXiv preprint arXiv:2604.20938*, 2026.

Huatong Song, Jinhao Jiang, Yingqian Min, Jie Chen, Zhipeng Chen, Wayne Xin Zhao, Lei Fang, and Ji-Rong Wen. R1-searcher: Incentivizing the search capability in llms via reinforcement learning. *arXiv preprint arXiv:2503.05592*, 2025.

Hongjin Su, Shizhe Diao, Ximing Lu, Mingjie Liu, Jiacheng Xu, Xin Dong, Yonggan Fu, Peter Belcak, Hanrong Ye, Hongxu Yin, Yi Dong, Evelina Bakhturina, Tao Yu, Yejin Choi, Jan Kautz, and

Pavlo Molchanov. Toolorchestra: Elevating intelligence via efficient model and tool orchestration, 2025. URL https://arxiv.org/abs/2511.21689.

Hao Sun, Zile Qiao, Bo Wang, Guoxin Chen, Yingyan Hou, Yong Jiang, Pengjun Xie, Fei Huang, and Yan Zhang. Decouplesearch: Decouple planning and search via hierarchical reward modeling. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 4465–4478, 2025.

Tongyi DeepResearch Team, Baixuan Li, Bo Zhang, Dingchu Zhang, Fei Huang, Guangyu Li, Guoxin Chen, Huifeng Yin, Jialong Wu, Jingren Zhou, et al. Tongyi deepresearch technical report. *arXiv preprint arXiv:2510.24701*, 2025.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. Interleaving retrieval with chain-of-thought reasoning for knowledge-intensive multi-step questions. In *Proceedings of the 61st annual meeting of the association for computational linguistics (volume 1: long papers)*, pp. 10014–10037, 2023.

Miles Turpin, Julian Michael, Ethan Perez, and Samuel R. Bowman. Language models don't always say what they think: unfaithful explanations in chain-of-thought prompting. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA, 2023. Curran Associates Inc.

United States Patent and Trademark Office. USPTO Open Data Portal. https://data.uspto.gov/, 2026. Accessed 2026-05-06.

U.S. Securities and Exchange Commission. EDGAR Application Programming Interfaces. https://www.sec.gov/search-filings/edgar-application-programming-interfaces, 2024. Accessed 2026-05-06.

Hanlin Wang, Chak Tou Leong, Jiashuo Wang, Jian Wang, and Wenjie Li. Spa-rl: Reinforcing llm agents via stepwise progress attribution. *arXiv preprint arXiv:2505.20732*, 2025a.

Ziliang Wang, Xuhui Zheng, Kang An, Cijun Ouyang, Jialu Cai, Yuhang Wang, and Yichao Wu. Stepsearch: Igniting llms search ability via step-wise proximal policy optimization. *arXiv preprint arXiv:2505.15107*, 2025b.

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.

Chunqiu Steven Xia, Yinlin Deng, Soren Dunn, and Lingming Zhang. Agentless: Demystifying llm-based software engineering agents. *arXiv preprint arXiv:2407.01489*, 2024.

Chunqiu Steven Xia, Yinlin Deng, Soren Dunn, and Lingming Zhang. Demystifying llm-based software engineering agents. *Proceedings of the ACM on Software Engineering*, 2(FSE):801–824, 2025.

John Yang, Carlos E Jimenez, Alexander Wettig, Kilian Lieret, Shunyu Yao, Karthik Narasimhan, and Ofir Press. Swe-agent: Agent-computer interfaces enable automated software engineering. *Advances in Neural Information Processing Systems*, 37:50528–50652, 2024.

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

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. *arXiv preprint arXiv:2210.03629*, 2022.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, Wang Zhang, Hang Zhu, Jinhua Zhu, Jiaze Chen, Jiangjie Chen, Chengyi Wang, Hongli Yu, Yuxuan Song, Xiangpeng Wei, Hao Zhou, Jingjing Liu, Wei-Ying Ma, Ya-Qin Zhang, Lin Yan, Mu Qiao, Yonghui Wu, and Mingxuan Wang. Dapo: An open-source llm reinforcement learning system at scale, 2025. URL https://arxiv.org/abs/2503.14476.

Weizhi Zhang, Yangning Li, Yuanchen Bei, Junyu Luo, Guancheng Wan, Liangwei Yang, Chenxuan Xie, Yuyao Yang, Wei-Chieh Huang, Chunyu Miao, Henry Peng Zou, Xiao Luo, Yusheng Zhao, Yankai Chen, Chunkit Chan, Peilin Zhou, Xinyang Zhang, Chenwei Zhang, Jingbo Shang, Ming Zhang, Yangqiu Song, Irwin King, and Philip S. Yu. From web search towards agentic deep research: Incentivizing search with reasoning agents, 2025a. URL https://arxiv.org/abs/2506.18959.

Yuxuan Zhang, Haoyang Yu, Lanxiang Hu, Haojian Jin, and Hao Zhang. General modular harness for llm agents in multi-turn gaming environments. *arXiv preprint arXiv:2507.11633*, 2025b.

Yuxiang Zheng, Dayuan Fu, Xiangkun Hu, Xiaojie Cai, Lyumanshan Ye, Pengrui Lu, and Pengfei Liu. DeepResearcher: Scaling deep research via reinforcement learning in real-world environments. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 414–431, Suzhou, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-332-6. doi: 10.18653/v1/2025.emnlp-main.22. URL https://aclanthology.org/2025.emnlp-main.22/.

Zijian Zhou, Ao Qu, Zhaoxuan Wu, Sunghwan Kim, Alok Prakash, Daniela Rus, Jinhua Zhao, Bryan Kian Hsiang Low, and Paul Pu Liang. Mem1: Learning to synergize memory and reasoning for efficient long-horizon agents, 2025. URL https://arxiv.org/abs/2506.15841.

Contents of Appendix

A Related Work **17**

B Limitations, Ethics, and Broader Impact **18**

C Tool Signatures **19**

D Reward & Training Hyperparameters **19**

E Benchmark Statistics **21**

F Two-Tier Memory and Working-Memory Rendering **22**

G Per-Turn Programmatic Nudges (Result Summaries) **23**

H Harness Algorithms **23**

I System Prompts and Templates **25**

I.1 Agent System Prompt (`Harness-1`) 25

I.2 `verify` Sub-Prompt 26

I.3 Answer-Generator Prompt (modular RAG) 26

I.4 LLM-as-Judge Prompt 27

J SFT Data Generation Recipe **27**

K Importance-Aware Curation: Algorithmic Detail **28**

L Content Deduplication: Algorithmic Detail **28**

M Component Ablation: Error Analysis **29**

M.1 Per-mechanism failure modes 29

M.2 All harness mechanisms disabled 31

N Evaluation Recipe **31**

O Modular RAG: Curated Sets Yield Higher Answer Accuracy **34**

P Harness as Confound: Same LLM, Different Harness **34**

Q Qualitative Case Studies **36**

Q.1 Case Study A: Stateful Search over a Drug-Approval Query 36

Q.2 Case Study B: Exact-Date Recovery in SEC Filings 43

Q.3 Case Study C: Fan-Out Search Resolves an Ambiguous Transfer Query 47

Q.4 Case Study D: Held-Out Multi-Hop QA via Search-State Editing 54

Q.5 Case Study E: Backtracking after Rejecting the Right Entity 58

A Related Work

Harness engineering. Harness engineering designs the environment layer between a language model and its task (Lopopolo, 2026; Martin et al., 2026). Recent work shows that the same model can vary by tens of points when placed in different harnesses (Yang et al., 2024; Xia et al., 2024), so the interface is a major factor in system-level performance. Tool orchestration (Qin et al., 2023; Jin et al., 2025; Liu et al., 2025; Su et al., 2025) usually fixes this interface and trains the policy to call the exposed tools more effectively. Harness engineering asks the lower-level design question of which interface the policy should operate in.

Most existing harnesses are designed by hand (Yang et al., 2024; Zhang et al., 2025b), often as empirical responses to model behaviors that may change as models improve (Martin et al., 2026). Recent systems automate harness construction by searching over code space (Lou et al., 2026; Lee et al., 2026) or configuration space (Sengupta & Wang, 2026). In `Harness-1`, we study the complementary question of how to shape a retrieval harness so that a trainable policy can use it well. We offload mechanical search bookkeeping into environment-side state and then train the policy to make semantic decisions over that state.

Tool orchestration vs. stateful harnessing. Let an agent interface be $\mathcal{I} = (\mathcal{A}, \mathcal{O}, T, r)$: actions, observation renderer, environment transition, and reward. Tool orchestration usually fixes an interface $\mathcal{I}_0$ and learns a policy inside it:

$$\max_{\theta} \ \mathbb{E}[R(\pi_\theta; \mathcal{I}_0)] \, .$$

Stateful harnessing changes $\mathcal{I}$ by adding the editable retrieval state from Table 1:

$$s_t = (P_t, C_t, I_t, D_t, G_t, V_t, H_t, B_t).$$

The transition is

$$(s_t, a_t) \mapsto (s_{t+1}, o_{t+1}),$$

instead of only $a_t \mapsto o_{t+1}$. The policy still learns tool use, but it learns over a state representation maintained by the harness:

$$\max_{\theta} \ \mathbb{E}[R(\pi_\theta; \mathcal{I}_{\texttt{Harness-1}})] \, .$$

Agentic search. Agentic search refers to large language model agents that gather evidence over multiple rounds of writing queries, reading the returned documents, and deciding when to stop (Zhang et al., 2025a). Early work in this direction interleaved retrieval with reasoning in tight loops (Li et al., 2025b; Gao et al., 2025; Liu et al., 2025; AI, 2025). ReAct (Yao et al., 2022), Self-Ask (Press et al., 2023), and IRCoT (Trivedi et al., 2023) established the pattern of alternating reasoning and tool calls. However, prompting-based reasoning alone is often inconsistent, and surface behavior does not reliably translate into the intended decision patterns (Turpin et al., 2023; Lanham et al., 2023; Han et al., 2025; 2026).

Reinforcement learning post-training (Schulman et al., 2017; Hu et al., 2025; Yu et al., 2025; Guo et al., 2025) has become an effective way to shape LLM reasoning, and a growing line of work applies it to retrieval and agentic search (Song et al., 2025; Chen et al., 2025b; Jiang et al., 2025a; Zheng et al., 2025). DeepRetrieval (Jiang et al., 2025a) shows that LLMs can be trained with RL to interact directly with real search engines and retrievers, improving retrieval through learned query and search behavior. Search-R1 (Jin et al., 2025) trains a single LLM to interleave reasoning and search calls under a downstream question-answering reward. s3 (Jiang et al., 2025b) decouples the searcher from the generator, training only the search policy for modular optimization with frozen generators. These methods work well at short horizons. However, as tasks grow longer and more complex, for example in report generation (Wu et al., 2025; Li et al., 2025a), the policy has to do too much at once. It must plan the next search (Chen et al., 2026; Sun et al., 2025), remember which documents it has seen (Zhou et al., 2025; Packer et al., 2024), track which constraints remain uncovered (Chen et al., 2025c), and manage an ever-growing context (Hong et al., 2025). Under this load, the reward signal becomes sparse relative to the work the policy is doing, which degrades RL training (Deng et al., 2025; Wang et al., 2025a). Recent work begins to address parts of this

overload. Context-1 (Bashir et al., 2026) trains a 20B retrieval agent that learns to selectively prune its own context during search. It thus starts to treat context management as part of the policy itself.

In all of these systems, the agent's harness is still a thin tool wrapper. Any state beyond the prune action must be reconstructed by the policy from a raw, append-only observation stream. Rather than training a stronger policy on a thin interface, `Harness-1` makes the interface itself part of the method. The harness is stateful, and its environment-side working memory takes over the mechanical bookkeeping of multi-turn search. We call this division of labor cognitive offloading, and we treat harness design as a central axis of agentic search.

B Limitations, Ethics, and Broader Impact

Limitations. `Harness-1` is designed for evidence-seeking retrieval tasks where the goal is to identify documents that support an answer. Most benchmarks in our evaluation are needle-in-a-haystack or multi-hop search tasks with annotated relevant evidence. This setting is important, but it does not cover all retrieval use cases. Breadth-oriented research tasks, open-ended report generation, abstention under missing evidence, and adversarial web environments remain outside the main scope of this work.

Our harness also introduces several engineered components whose quality may vary across domains. The evidence graph uses a lightweight regex-based extractor for entities, dates, and years, rather than a full entity-linking system. The `verify` tool uses an LLM-based verifier as a proxy entailment model, which can make mistakes on ambiguous, highly technical, or underspecified claims. The sentence-level BM25 compressor may remove useful context when relevance depends on discourse structure rather than local sentence overlap. A small embedding model keeps more answer text than BM25 at a small extra cost (Appendix M.1.2).

Training uses gold documents only to compute the RL reward and to filter the SFT teacher trajectories. At inference time `Harness-1` needs only the query and the corpus. When gold labels are not available, another measure of search quality can replace the reward. Examples include downstream answer accuracy, as in s3 (Jiang et al., 2025b), and an LLM judge that scores the curated set against the query. The same score can also replace recall in the SFT filter. Because the reward is a scoring rule over the final curated set and the trajectory, changing it leaves the harness, the action space, and the RL recipe unchanged.

Evaluation is limited by benchmark size and annotation coverage. Some datasets have non-negligible confidence intervals, and near-duplicate or incomplete qrels can affect recall-oriented metrics. We therefore report multiple retrieval metrics, trajectory-level diagnostics, ablations, and modular RAG answer accuracy. These measurements should still be read as evidence of retrieval behavior under the specified benchmark and harness conditions, not as a complete measure of real-world search reliability.

Ethics. This work studies how to train language-model search agents to retrieve supporting evidence more effectively. The primary ethical motivation is to make retrieval-augmented systems more transparent and auditable. Instead of relying only on a model's parametric memory, the system returns an explicit curated set of documents that downstream models or human users can inspect, verify, and use.

The main ethical risks come from misuse and over-reliance. A stronger search agent could retrieve sensitive, private, copyrighted, or otherwise inappropriate information if connected to an unrestricted corpus. It could also amplify misleading evidence if the retrieval backend contains low-quality, biased, or adversarial documents. Our method should therefore be deployed only with appropriate corpus access controls, logging, rate limits, privacy filters, and human oversight for sensitive domains.

The system should not be treated as a source of truth by itself. `Harness-1` retrieves and curates evidence; it does not guarantee that the evidence is complete, unbiased, or correctly interpreted. In high-stakes domains such as medicine, law, finance, or public policy, any downstream answer should be reviewed by qualified experts and validated against authoritative sources.

Broader impact. Stateful retrieval agents can reduce the cost of evidence gathering and make long-horizon search more accessible. This may benefit scientific literature review, legal and financial document analysis, fact-checking, education, and other workflows where users need to locate and organize evidence across many documents. Because `Harness-1` separates retrieval from generation, its outputs can be audited more directly than closed-book answers.

At the same time, lower-cost search agents can increase the scale of undesirable information seeking, such as privacy-invasive search, automated cherry-picking of evidence, and targeted collection of sensitive information. These risks are not unique to our method, but stronger and cheaper retrieval agents can make them easier to execute. Responsible deployment should therefore restrict corpus access, monitor usage, and pair retrieval with verification and human review.

Our planned release is limited to the retrieval subagent, harness code, data-generation pipeline, RL recipe, and evaluation recipe, with documentation of intended use and dataset provenance. The released system is not a general autonomous web agent and is not intended to replace expert judgment in high-stakes decision-making.

C Tool Signatures

This section lists the tools exposed to the `Harness-1` policy. The tools are grouped around the main operations in the harness: retrieving new candidates, revisiting stored evidence, editing the curated set, verifying claims, and ending the episode. The policy chooses among these tools, while the harness executes the calls and updates the working-memory state described in Table 1.

Table 4 gives the functional signature of each tool used in all teacher, SFT, RL, and evaluation runs.

Table 4: `Harness-1` tool signatures.

Tool	Effect
`fan_out_search(queries)`	Up to five parallel hybrid searches, RRF + rerank
`search_corpus(query)`	Single hybrid search (BM25 + dense), RRF, rerank
`grep_corpus(pattern)`	Regex match over the corpus
`read_document(doc_id)`	Full-text read, reranked and truncated to budget
`review_docs(doc_ids)`	Re-render seen docs from WorkingMemory (no corpus call)
`curate` (add, remove, importance)	Edit curated set with four-level importance tag
`verify` (doc_ids, claim)	Per-document LLM entailment check against the claim
`end_search(reasoning)`	Terminate episode, return curated set
For backward compatibility (registered, no-op):	
`prune_chunks(...)`	Returns "managed via working memory; no pruning needed."

D Reward & Training Hyperparameters

Table 5 lists the reward weights and penalty coefficients used in the deployed RL run, alongside their configuration names.

We also report the training configuration for the released `Harness-1` checkpoint. We separate hyperparameters for SFT, RL, the search environment, retrieval infrastructure, and evaluation. The aim is to make the training recipe reproducible and to show which settings belong to model optimization and which belong to the harness and retrieval backend.

SFT is used only to teach the model how to operate the stateful interface. RL is then applied over full search episodes using the same harness state renderer and tool set. Table 6 lists the exact settings used in the deployed run.

One configuration for all benchmarks. We use one harness configuration for all eight benchmarks and do not tune any setting per benchmark. The values in Table 6, such as the MinHash threshold, the compression and auto-seed top-K, and the 30-document cap, were set once and kept fixed. We did not run a sensitivity sweep over these values. Table 3 shows how much each mechanism contributes when it is removed.

Table 5: Reward hyperparameters for the deployed `Harness-1` RL run. **Note on the empty-set penalty:** when the curated set is empty at episode termination, the reward short-circuits to $\pi_\emptyset = -0.2$, bypassing the rest of the formula. For all other terminations, the formula in §2.3 applies and is clipped below by $\mathcal{R} \geq 10^{-3}$.

Symbol	Code variable	Value
w_F (set quality)	`OUTCOME_WEIGHT`	0.7
w_τ (trajectory recall)	`TRAJECTORY_RECALL_WEIGHT`	0.3
β (F_β exponent)	`RECALL_BETA`	2.0
w_{A} (FA recall)	`FINAL_ANSWER_RECALL_WEIGHT`	0.8
$w_{\tau\mathrm{A}}$ (trajectory FA recall)	`TRAJECTORY_FA_RECALL_WEIGHT`	0.4
B_{A} (binary FA bonus)	`FINAL_ANSWER_BONUS`	1.0
w_{miss} (FA miss penalty)	`FA_MISS_PENALTY_WEIGHT`	0.35
$\pi_\emptyset$ (empty-set short-circuit)	`NO_CURATE_PENALTY`	-0.2
Turn penalty start	`TURN_PENALTY_MIN_TURNS`	20
Turn penalty max	`TURN_PENALTY_MAX`	0.02
Max turns hard cap	`MAX_TURNS`	40
Format-reward floor	`MIN_FORMAT_REWARD`	10^{-3}
KL coefficient (anchor to SFT)	`KL_PENALTY_COEF`	0.0
w_{div} (tool-diversity bonus weight)	`TOOL_DIVERSITY_BONUS`	**0.15**
ν_0 (tool-diversity target)	`TOOL_DIVERSITY_TARGET`	**6**

Does the answer-miss penalty discourage discovery? The term $w_{\mathrm{miss}}(\rho_{\tau\mathrm{A}} - \rho_{\mathrm{A}})_+$ is paid only when the trajectory found an answer document that the curated set does not contain. One could worry that the policy learns to avoid finding answer documents so that it never pays this penalty. The weights rule this out. Finding an answer document raises the trajectory answer term by $w_{\tau\mathrm{A}} = 0.4$ per unit of recall. The penalty for not promoting it is at most $w_{\mathrm{miss}} = 0.35$ per unit. Finding it therefore never lowers the reward, even if the document is never promoted. Promoting it adds the curated answer term ($w_{\mathrm{A}} = 0.8$), the set-quality term, and the answer bonus ($B_{\mathrm{A}} = 1.0$). The training logs agree. During RL, the share of answer documents found in the trajectory rises from 0.44 to 0.61, while the share found but not promoted falls from 0.17 to 0.07. The policy closed the gap by promoting more, not by finding less.

SFT only versus SFT and RL. Table 7 and Figure 6 compare three models on the same query sets: the base `gpt-oss-20b` (repeated from Table 2), the SFT checkpoint at step 550 that starts RL, and the final `Harness-1` after RL. SFT teaches the model to use the harness and gives a moderate gain over the base model. RL adds most of the search and curation quality. RL trains only on SEC, and its gain also holds on BC+, Web, and Patents, which it never sees.

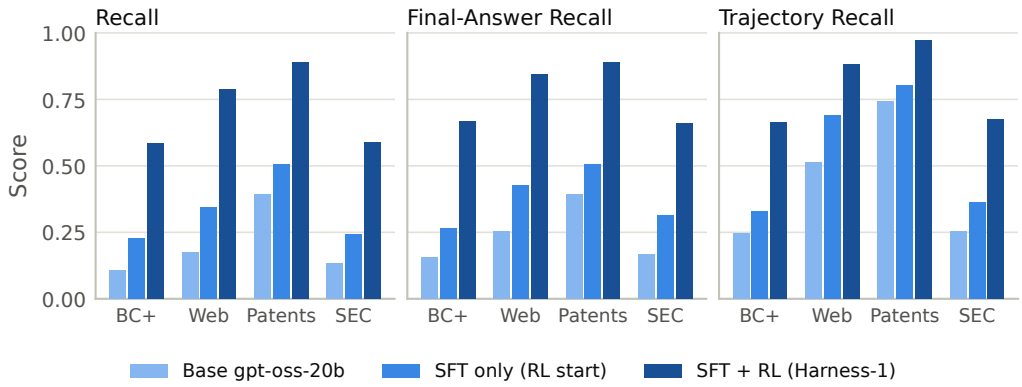

Figure 6: Base model, SFT only, and SFT with RL on the four source-family benchmarks. RL trains only on SEC.

Table 6: SFT and RL hyperparameters for the deployed `Harness-1` checkpoint.

Stage	Setting	Value
SFT	Base model	`openai/gpt-oss-20b` (MoE)
SFT	Adapter	LoRA, rank 32
SFT	Optimiser	AdamW (Tinker default)
SFT	Learning rate	5×10^{-6}
SFT	Batch size	128
SFT	Max sequence length	32,768
SFT	Epochs	3
SFT	Min recall filter	0.10 (final output recall)
SFT	Per-trajectory expansion	one datum per turn
SFT	Selected checkpoint	step 550 (sampler weights)
RL	Algorithm	on-policy CISPO (clip $[0, 5]$)
RL	Adapter	LoRA, rank 32
RL	Learning rate	1×10^{-5}
RL	Queries per step (BATCH)	128
RL	Rollouts per query (GROUP)	8
RL	Total rollouts per step	1,024
RL	Substeps per step	4
RL	Total steps	80
RL	Total rollouts	$\sim$82K
RL	Rollout temperature	1.0
RL	Generation budget	2,048 tokens
RL	Model context limit	32,768 tokens
RL	KL anchor	disabled
RL	Drop constant-reward groups	yes
RL	Dataset	`sec` (single-domain)
Env	MAX_TURNS	40
Env	RECENT_K (recent window)	5
Env	SEARCH_DISPLAY_LIMIT	10
Env	MAX_OBS_CHARS	15,000
Env	SEARCH_TOKEN_BUDGET	4,096
Env	SENTENCE_COMPRESS_K	4
Env	MINHASH_DEDUP_THRESHOLD	0.85
Env	MINHASH_NUM_PERM	64
Env	EVIDENCE_GRAPH_MAX_ENT	8
Env	AUTO_POPULATE_TOP_K	8
Env	MAX_CURATED_DOCS	30
Env	DOC_SNIPPET_CHARS	120
Retr	Retrieval	hybrid BM25 + dense, RRF
Retr	Reranker	Qwen3-Reranker-8B (Baseten-hosted)
Retr	Reranker max tokens	4,096
Eval	Sampling temperature	1.0
Eval	Max tokens per turn	2,048 (`Harness-1`), 4,096 (Context-1)
Eval	Max trajectory length	40 turns (`Harness-1`), 64 turns (Context-1)

E BENCHMARK STATISTICS

This section summarizes the evaluation benchmarks used in the main experiments. The suite includes static Chroma-backed corpora and live web retrieval. It covers encyclopedic web search, finance filings, patents, and multi-hop question answering. The main text also splits these benchmarks into source-family benchmarks and held-out transfer benchmarks.

Table 8 reports the backend, number of test queries, domain, and task style for each benchmark.

Table 7: Base model, SFT only, and SFT with RL on the source-family benchmarks.

Model	Metric	BC+	Web	Patents	SEC
Base `gpt-oss-20b`	Recall	0.109	0.174	0.393	0.133
	Final-Answer Recall	0.156	0.252	0.393	0.166
	Trajectory Recall	0.246	0.512	0.742	0.252
SFT only (step 550)	Recall	0.226	0.344	0.505	0.242
	Final-Answer Recall	0.266	0.426	0.505	0.315
	Trajectory Recall	0.328	0.690	0.802	0.364
SFT + RL (`Harness-1`)	Recall	0.584	0.787	0.890	0.589
	Final-Answer Recall	0.667	0.845	0.890	0.659
	Trajectory Recall	0.665	0.881	0.973	0.677

Table 8: Benchmark statistics. Sources: BrowseComp+ (Chen et al., 2025c); Web, Patents, and SEC (Bashir et al., 2026; United States Patent and Trademark Office, 2026; U.S. Securities and Exchange Commission, 2024); LongSealQA and Seal0QA (Pham et al., 2025); FRAMES (Krishna et al., 2025); HotpotQA (Yang et al., 2018).

Benchmark	**Backend**	**Test queries**	**Domain**	**Style**
BrowseComp+	Chroma (BC+ corpus)	166	Web encyclopaedic	Multi-constraint
Web	Chroma (web 1.17)	554	Web encyclopaedic	Multi-hop
Patents	Chroma (USPTO 1.18)	718	Legal/USPTO	Single-doc heavy
SEC	Chroma (SEC 1.4)	502	Finance filings	Multi-filing
LongSealQA	Chroma (longsealqa)	254	Open web	Multi-hop, long-context
Seal0QA	Web (Serper+Jina)	111	Open web	Factoid / multi-hop mix
FRAMES	Web (Serper+Jina)	824	Open web	Multi-hop
HotpotQA-subset	Web (Serper+Jina)	493	Open web	2-hop

F Two-Tier Memory and Working-Memory Rendering

`Harness-1` uses a two-tier memory: an inner WorkingMemory summary that fits in the prompt, and an outer `doc_store` with the full text of every retrieved chunk, read through `review_docs`. This section gives the exact rendering protocol so the reported behavior can be reproduced.

Working-memory text layout. Each rendered WorkingMemory block contains, in order:

1. Header: `== Working Memory (summarising turns 0--`n`) ==` followed by the original query.
2. Curated set: entries grouped by importance level (`very_high` first, then `high`, `fair`, and `low` last). Each entry shows the doc ID and a 120-character snippet.
3. Document pool: the most recent 50 uncurated docs are shown with snippets. Older uncurated docs are summarized as a comma-separated ID list, truncated at 30 IDs with a `(+N more)` marker if longer.
4. Search history: the last 12 entries. Older entries are summarized as `...` (`(`k `earlier searches)`.
5. Evidence-graph block (only if ≥ 1 multi-doc bridge entity exists): up to 8 frequent entities listed with their doc IDs, plus a singleton count.
6. Dedup notice (only if any near-duplicates were suppressed): `[Dedup]` k `near-duplicate chunk(s) auto-suppressed`.

Per-turn observation layout. Each observation shown to the policy contains, in order, (a) the WorkingMemory summary block, (b) up to `RECENT_K=5` prior (action, result) pairs, and (c) the new tool result for the most recent action. In (b), reasoning is truncated to 300 characters for all but the most recent turn. Search-style results in (c) go through sentence-BM25 compression (K=4 kept sentences) and content-fingerprint deduplication. Search results are truncated to `SEARCH_DISPLAY_LIMIT=10` entries per call, and each result snippet is capped at `DOC_SNIPPET_CHARS=120`.

Context-budget truncation (5-pass progressive degradation). The prompt token budget is 30,720 tokens, leaving 2,048 for generation within the 32,768 context limit. When the rendered context exceeds it, `render_context_within_budget` tries five passes in order and returns the first that fits: (1) normal render; (2) truncate the document-pool section of the WORKINGMEMORY summary while keeping the curated set, search history, and evidence graph; (3) truncate analysis text in older turns to 100 chars and the WORKINGMEMORY summary itself to 2,000 chars; (4) drop the oldest recent (action, result) pair and retry, one pair at a time; (5) minimal context (system + tool descriptions + query only). Pass 5 always succeeds because the system prompt is constant-size. This mechanism guarantees that no rollout fails due to context overflow, and that the curated set and recent working memory are the last things to be cut.

G PER-TURN PROGRAMMATIC NUDGES (RESULT SUMMARIES)

Between each turn's tool result and the next observation, `Harness-1` adds a short *result summary* as a user message. The summary is generated programmatically (no LLM call) and combines factual status with conditional prescriptive nudges. The SFT teacher (GPT-5.4) sees the same summaries as the trained policy does during RL and at inference, so this part of the prompt is the same across all three stages.

Always-on factual lines. `[STATUS]`: tool name, number of docs returned, novel docs, total pool size; or curate/review status. `Curated:` current curated set size and a preview of the first five doc IDs.

Conditional prescriptive lines (gated by `HARNESS_PRESCRIPTIVE=1`, which was on for all reported runs). `[ACTION REQUIRED]`: triggered after a search turn with ≥ 1 turn since last curate; instructs the agent to curate before searching again. `[WARN]`: triggered on multiple consecutive non-curate turns or a blank curated set while the pool already has candidates. `[TIP]`: triggered when truncation is detected (suggesting `read_document`), when only one tool type has been used by turn ≥ 3, or when `grep_corpus` or `read_document` have not been used by turn $\geq 4/6$. `[NEXT]`: short suggestion of the immediate next action.

These nudges are part of the harness, *not* part of the trained policy's intrinsic capability. They are among the harness components ablated in Table 3, where the "all `Harness-1` mechanisms disabled" row removes them.

H HARNESS ALGORITHMS

This appendix gives the formal harness algorithms used by `Harness-1`. The driver in Algorithm 1 is kept small on purpose. Explicit subroutines handle all mechanism-specific behavior. The same subroutines are used during teacher trajectory generation, SFT replay, RL rollout, and inference, so the policy is trained on the same state interface it uses at test time.

State variables. For a query q, the harness maintains a document pool P, a curated output set C with cap $M{=}30$, and an importance map $I : C \to \{\texttt{very_high}, \texttt{high}, \texttt{fair}, \texttt{low}\}$. It also keeps a full-text store D, an evidence graph G, a verification cache V, a search history H, a near-duplicate index U, and a Boolean flag AUTOSEEDED. The policy makes the semantic choices (queries, claims, promotions, removals, and termination). The harness handles the deterministic state updates.

Algorithm 1 `Harness-1` stage driver.

Require: query q, tools $\mathcal{T}$, stage $s \in \{\text{teacher}, \text{sft}, \text{rl}, \text{infer}\}$, policy π or replay trajectory τ
Ensure: trajectory record for training stages, or curated set C for inference

Initialize $P, C, I, D, G, V, H, U \leftarrow \emptyset$ and AUTOSEEDED$\leftarrow$ false
Build a per-episode rerank instruction from q and the dataset family
for $t = 1, \ldots, T_{\max}$ **do**
  $x_t \leftarrow$ RENDERCONTEXT$(q, P, C, I, D, G, V, H, \text{recent turns})$ Algorithm 5
  if $s = \text{sft}$ **then**
    Read stored action a_t and stored observation o_t from τ
  else
    Sample a structured tool action $a_t \sim \pi(\cdot \mid x_t)$
    Execute a_t with tools $\mathcal{T}$ to obtain raw observation o_t
  end if
  if a_t is a retrieval or memory-inspection action **then**
    $(o_t, P, D, G, U, C, I, \text{AUTOSEEDED}) \leftarrow$ PROCESSOBSERVATION(a_t, o_t) Algorithm 2
  else if $a_t =$ `curate` **then**
    $(C, I) \leftarrow$ CURATE(a_t, C, I, P) Algorithm 3
  else if $a_t =$ `verify` **then**
    $V \leftarrow V \cup$ VERIFYCLAIM(a_t, D) Algorithm 4
  else if $a_t =$ `review_docs` **then**
    $o_t \leftarrow$ REVIEWDOCS(a_t, D) Algorithm 4
  else if $a_t =$ `end_search` **then**
    break
  end if
  Update search history H, result summary, token marker, and a working-memory snapshot
end for
Apply the stage-specific finalizer in Algorithm 6

Algorithm 2 Observation processing and working-memory update.

Require: action a, raw observation o, ranked IDs R, query text q, state $(P, D, G, U, C, I, \text{AUTOSEEDED})$
Ensure: processed observation and updated memory state

if $a \in \{$`search_corpus`, `fan_out_search`, `grep_corpus`$\}$ **then**
  Split each returned chunk into sentences and keep the top K=4 BM25-scored sentences, preserving original order
end if
Parse document IDs and document text blocks from o
for each returned chunk (r, text) **do**
  Normalize r to a document ID d when the dataset uses document-level IDs
  if $d \in P$ **then**
    skip adding d to the pool, but retain the observation for trajectory accounting
  else if text is a near-duplicate under U **then**
    suppress d from rendered memory and increment the dedup counter
  else
    Add d to P and store full text plus snippet in $D[d]$
    Extract proper nouns, years, and numeric dates from text and update G
  end if
end for
if $a \in \{$`search_corpus`, `fan_out_search`$\}$ and AUTOSEEDED is false and $R \neq \emptyset$ **then**
  Add the first k=8 ranked IDs from R to C with $I(d) =$ `fair`
  Set AUTOSEEDED$\leftarrow$ true and append an `[AUTO-POPULATED]` notice
end if
if token-budget markers are enabled **then**
  Append `[Context: X/Y]` using the current approximate prompt budget
end if
return processed observation and updated state

Algorithm 3 Importance-aware curation and capacity management.

Require: add IDs A, remove IDs R, proposed importance map J, curated set C, importance map I, pool P
Ensure: updated (C, I) and a capacity-status observation

Normalize all IDs in A, R, and J to the dataset's output ID convention
Remove every $d \in R$ from C and delete $I(d)$
for each $d \in A$ **do**
  Let $j \leftarrow J(d)$ if valid, otherwise $j \leftarrow$ `fair`
  if $d \in C$ **then**
    Re-tag $I(d) \leftarrow j$ and continue
  end if
  if $|C| < M$ **then**
    Insert d into C with $I(d) \leftarrow j$
  else
    Let $w = \arg\max_{c \in C} \mathrm{rank}(I(c))$, where `low` has worst rank
    if $\mathrm{rank}(j) < \mathrm{rank}(I(w))$ **then**
      Evict w; insert d with $I(d) \leftarrow j$
    else
      Reject d and report a `[CAPACITY]` marker
    end if
  end if
end for
return (C, I) rendered by priority `very_high` $\succ$ `high` $\succ$ `fair` $\succ$ `low`

Algorithm 4 Verification and memory review.

Require: action a, full-text store D, verification cache V
Ensure: observation o and updated verification cache V

if $a =$ `review_docs`$(\mathcal{D})$ **then**
  Return the full text of up to 5 requested IDs from D, without a corpus call
else if $a =$ `verify`$(\mathcal{D}, c)$ **then**
  for each requested document $d \in \mathcal{D}$ **do**
    Ask the verifier whether $D[d]$ directly supports every constraint in claim c
    Store $(d, c,$ `yes/no`, short rationale$)$ in V
  end for
  Return the per-document yes/no judgments and rationales
end if

I System Prompts and Templates

I.1 Agent System Prompt (Harness-1)

Agent system prompt

Role. You are a retrieval subagent. Find the most relevant documents from the corpus for the question; do not answer the question yourself.

Input. `QUERY: [query]`

Tools. `fan_out_search` runs up to five diverse queries; `search_corpus` runs a single hybrid query; `grep_corpus` performs exact regex matching; `read_document` opens full text; `review_docs` re-renders remembered documents; `curate` edits the curated set (maximum 30 documents); `verify` checks a claim against remembered documents; `end_search` submits the curated set.

Loop. Decompose the query into constraints. Search, curate immediately, search again from a different angle, inspect or review uncertain documents, verify multi-constraint evidence, and end when the curated set is sufficient. The episode has at most 40 turns.

State rules. Use importance tags {`very_high`, `high`, `fair`, `low`}; `very_high` requires prior verification. The first successful search seeds tentative `fair` candidates; promote strong documents and remove weak ones. Use evidence-graph bridge entities as follow-up signals. Each observation ends with `[Context: X/Y]`; wrap up above 75% and call `end_search` above 90%.

Algorithm 5 Budget-safe context rendering.

Require: system prompt, rendered working memory W, recent actions and observations, result summaries, token budget B
Ensure: tokenized prompt x with $|x| \leq B$
1: Build W from query, curated set grouped by importance, recent uncurated pool, search history, evidence graph, verification cache, dedup notice, and context marker
2: Render system message, tool descriptions, query, optional W, recent turns, and result summaries
3: **if** the tokenized prompt fits within B **then**
4: **return** prompt
5: **end if**
6: Truncate the document-pool section of W, preserving curated set, search history, and evidence graph; retry
7: If needed, truncate older reasoning to 100 characters and cap W at 2,000 characters; retry
8: If needed, drop the oldest recent action-observation pair one at a time; retry after each drop
9: If all retries fail, return the minimal prompt: system message, tool descriptions, and query

Algorithm 6 Stage-specific finalizers.

Require: stage s, trajectory τ, pool P, curated set C, policy π
1: **if** $s = \text{teacher}$ **then**
2: Save all turns, observations, working-memory snapshots, P, C, document store D, and final retrieval metrics
3: **else if** $s = \text{sft}$ **then**
4: Filter trajectories below the recall threshold
5: Replay each kept trajectory through Algorithms 2–5
6: For every turn t, train on the supervised pair $\textsc{RenderContext}_t \mapsto a_t$
7: **else if** $s = \text{rl}$ **then**
8: Score C and P with curated recall, precision, final-answer recall, trajectory recall, and trajectory final-answer recall
9: Compute terminal reward: set quality, trajectory coverage, answer bonus, dense answer shaping, answer-miss penalty, tool-diversity shaping, and turn penalty
10: Update π with on-policy CISPO; save checkpoint and rollout summaries
11: **else if** $s = \text{infer}$ **then**
12: Return the priority-sorted curated set C to the downstream answer generator
13: **end if**

I.2 VERIFY SUB-PROMPT

`verify` is implemented as a small per-document LLM call.

`verify` sub-prompt

System. You are a strict document verifier. Given one `CLAIM` and one document's full text, answer only `yes` or `no`, followed by a rationale of at most 20 words. Answer `yes` only if the document directly supports every part of the claim. Answer `no` if any constraint is missing or contradicted. Be conservative.

User template.

```
CLAIM: [claim]
DOCUMENT: [document text]
Answer: yes/no. rationale
```

I.3 ANSWER-GENERATOR PROMPT (MODULAR RAG)

Answer-generator prompt

Task. You are given a question and retrieved documents. Use only the documents to reason step by step and answer the question.

Question. `[query]`

Retrieved documents.

`[documents]`

Output rule. End the response with the final answer on the last line, prefixed exactly with `Final Answer:`.

For **Closed-Book**, the prompt omits the documents and the instruction is changed to "Using only your own knowledge, reason step by step and provide your final answer." For **Naive RAG**, a single BM25+dense hybrid query retrieves the top-10 chunks and those replace the curated set.

I.4 LLM-AS-JUDGE PROMPT

We use GPT-5.4 at temperature 0 as the judge and follow the official BrowseComp+ template. We report the strict CORRECT rate as "answer accuracy."

LLM-as-judge prompt

Task. Judge whether `[response]` answers `[question]` correctly according to the precise `[correct_answer]`. Do not solve the problem independently or comment on irrelevant background.

Inputs.

```
[question]: [query]
[response]: [generated_answer]
[correct_answer]: [gold_answer]
```

Return fields. `extracted_final_answer` is the exact final answer, or `None` if absent. `reasoning` explains only meaningful differences. `correct` is `yes` when the extracted answer matches the gold answer, allowing small numerical tolerances, and `no` otherwise. `confidence` is 0–100%.

J SFT DATA GENERATION RECIPE

SFT trajectories are produced by GPT-5.4 (`tool_choice=required`, `max_completion_tokens=4096`) running natively inside the `Harness-1` harness. The teacher sees the same system prompt (Appendix I.1), the same observation layout, and the same tool schemas as the trained policy will see at RL and inference time.

Teacher agent loop. The generation script runs a single-phase loop. At each turn, it sends GPT-5.4 all 7+1 tool schemas and the teacher's full conversation history: the system prompt, the working-memory text, the most recent $K{=}5$ turn transcripts, and result summaries for older turns. The teacher selects a tool via `tool_choice=required` and returns structured JSON arguments including a mandatory `reasoning` field. The harness executes the tool against the live Chroma corpus and returns the observation.

Guidance protocol. The generation script adds turn-level guidance to the prompt. It enforces the search→curate pattern. It nudges the teacher to call `verify` on multi-constraint queries (after ≥ 6 turns and ≥ 3 curated docs) and prompts it to backtrack when the last 3 search-history entries yielded ≤ 1 new doc. It also gives tool-diversity hints if `grep_corpus`/`read_document` are unused after 4 turns, and urgency and curate-aggressiveness messages near the turn cap. This guidance ensures that teacher trajectories use the full tool vocabulary and follow the search→curate pattern.

Per-dataset quotas and filtering. The raw quotas per dataset are BC+/300, SEC/250, Patents/150, Web/150, Web-simple/75, and SEC-simple/75, for ∼1K raw trajectories in total. A 0.10 recall gate discards trajectories with recall < 0.10 and leaves 899 trajectories. Expanding each trajectory into turn-conditional datums gives ∼26K training examples.

Auto-populate on first search. During trajectory generation, the first successful search triggers `auto_populate_from_first_search`, which adds the top-k reranked results ($k{=}8$) to the curated set at `fair` importance. A per-trajectory `wm.auto_populated` flag makes this step idempotent. An `[AUTO-POPULATED]` marker is added to the observation and tells the teacher to promote good docs and remove irrelevant ones.

Observation processing. Search observations are processed exactly as in RL, with BM25 sentence compression (top-4 sentences per chunk), content-hash dedup, and the `[Context: X/Y]` token marker. The dedup uses MinHash-LSH (64 permutations, 5-gram shingles, threshold 0.85). When `datasketch` is unavailable, it falls back to a SHA hash of a normalized 4,000-char prefix.

K IMPORTANCE-AWARE CURATION: ALGORITHMIC DETAIL

This section details the eviction policy central to the subtractive-curation mechanism in `Harness-1` (§2.1).

Importance levels and ordering. Every document carries an importance tag $\in \{\texttt{very_high}, \texttt{high}, \texttt{fair}, \texttt{low}\}$ with rank $\texttt{very_high}{=}0 < \texttt{high}{=}1 < \texttt{fair}{=}2 < \texttt{low}{=}3$ (lower rank = higher importance). Auto-populated documents enter at `fair`.

ID normalization. Both `add_ids` and `importance` keys are normalized by stripping a trailing `_N` chunk suffix *only if* the resulting ID is not already in the pool. This lets the policy pass either chunk IDs (`42982_3`) or doc IDs (`42982`) interchangeably, while preserving doc IDs that legitimately contain underscores.

Eviction protocol. When the curated set is at capacity ($M{=}30$) and the agent issues a `curate` call with new `add_ids`:

1. For each incoming doc, compute its `incoming_rank` from the provided importance tag (default `fair`).
2. Scan the current curated set for the document with the *worst* (highest) importance rank.
3. If `worst_rank` $>$ `incoming_rank`, evict the worst-ranked doc and insert the new doc.
4. If no evictable doc exists, the add is rejected; the status string includes a `[CAPACITY]` marker listing up to 5 rejected IDs.

Re-tagging in place. A `curate` call with an `add_id` already in the curated set updates the tag in-place without eviction. This allows the agent to promote a document after a successful `verify` call.

Rendering. The curated set is rendered grouped by importance level (`very_high` first, `low` last) with the tag visible next to each document ID:

```
Curated Set (15/30):
  very_high:
  [*] doc_42982:  Tesla 10-K 2023 mentioning SEC fine...
  high:
  [*] doc_11293:  Tesla annual report 2023...
  fair:
  [*] doc_55017:  General Motors 10-K 2023...
```

L CONTENT DEDUPLICATION: ALGORITHMIC DETAIL

`Harness-1`'s content-fingerprint dedup is implemented in `ultra_core.ContentDedupTracker`.

Primary path (MinHash-LSH). When `datasketch` is available, each chunk is fingerprinted by tokenizing on the regex `[a-z0-9]+`, building 5-gram shingles, and inserting them into a MinHash with 64 permutations. The LSH index uses Jaccard threshold 0.85. For each new chunk, the tracker queries the LSH index. Any match is treated as a near-duplicate. It is dropped from the pool but *retained* in the `trajectory_recall` accounting, so the agent is still credited for having encountered the gold information.

Fallback path (SHA prefix). If `datasketch` is unavailable, the tracker computes a SHA-1 of the first 4,000 characters of the normalized token stream and compares it against a hash set. This path catches only exact duplicates, not near-duplicates, but it adds no extra dependency.

Diagnostic. The WORKINGMEMORY rendering shows the number of suppressed duplicates as a `[Dedup] k near-duplicate chunk(s) auto-suppressed` line. This lets the policy reason about whether the corpus has heavy duplication.

M COMPONENT ABLATION: ERROR ANALYSIS

This appendix looks at the results in Table 3 at the trajectory level. For each ablation we identify queries that full `Harness-1` solves (FA Recall ≥ 0.5) but the ablated condition fails on (FA Recall $=0$), then compare action mixes and trajectories on those *paired* queries. All numbers are over the same BrowseComp+ test queries; no retraining is involved.

Across ablations, removing a single state mechanism does more than drop information. It *changes the policy's behavior*. On failing paired queries the share of `search_corpus` actions rises by 3–7 points, while `read_document` and `verify` drop sharply, by up to $7\times$ or to zero. The trained policy falls back to many shallow searches and often fails to reach, or to keep, the relevant documents. The per-mechanism analyses below each illustrate the pattern with one representative failing query (truncated for length).

M.1 PER-MECHANISM FAILURE MODES

M.1.1 IMPORTANCE TAGS

ΔRecall -4.1%, ΔFA -7.9% relative (15 hard fails). Binary `curate` with FIFO eviction loses the `very_high/high/fair/low` confidence signal over the curated set. On failing queries, `read_document` drops $7\times$ ($4.7\%\rightarrow 0.7\%$), `verify` drops to 0%, and `search_corpus` climbs to 94.1% of all actions.

Example. Q893 asks for the birth name of a 1920s-born actress married to a director-translator (gold answer: "Medea Japaridze"). Full `Harness-1` reads `doc_3161`, identifies Medea Japaridze on T4, then anchors on her again through her spouse Revaz Tabukashvili (T28, T37, T39) and reaches curated recall 1.0 in 40 turns. The ablation issues 19 `search_corpus` calls without a single `read_document` or `verify`, never anchors on "Medea Japaridze", and ends with curated recall 0.12 on *the same* 40-turn budget. Without graded importance tags, the agent cannot decide which candidate to follow up on.

M.1.2 SENTENCE-BM25 COMPRESSION

ΔRecall $+0.2\%$, ΔFA -7.0% relative (13 hard fails). Without per-return sentence compression, the top-k chunks come back raw, so returns are longer and noisier. The aggregate Recall delta is ~ 0, but FA Recall drops sharply because the agent loses the bridge sentences it normally uses to seed follow-up reads. `read_document` on failing queries drops from 4.1% to 2.6% and `verify` from 2.0% to 0.4%.

Example. Q1012 asks for the title of a 2018–2023 EP whose songs all end in a question mark and were drawn from the artist's journal entries. Full `Harness-1` recovers it in 26 turns: T14 `grep`s for `"gave my all"` (a phrase from the artist interview), reads `doc_22401` on T22, and `verify` confirms "Maki's Tanong" on T24. The ablation reads `doc_77931` on T2 (a near-miss), then issues 18 more searches without reading another document. Without compressed bridge sentences in the search returns, it cannot identify the next phrase to query.

BM25 or an embedding model? We also compare the BM25 sentence scorer with a small MiniLM sentence-embedding model. On 939 answer-bearing gold chunks from BC+, we check whether the gold answer string survives when each chunk is cut to its top-K sentences (Table 9). The embedding model keeps more answer text at every K. It costs about 4 ms per chunk on a GPU in batches, against about 1 ms for BM25 on a CPU. All results in this paper use BM25. The embedding scorer can replace it without other changes to the harness.

Table 9: Share of answer-bearing BC+ chunks whose gold answer string survives compression to the top-K sentences.

K	BM25	MiniLM
2	36.7%	44.3%
4 (ours)	54.3%	65.7%
8	75.1%	82.9%

M.1.3 AUTO-SEED

ΔRecall -0.3%, ΔFA -6.4% relative (12 hard fails). With C_0 empty, the first `curate` has no comparison context, and the importance-tag space is undefined until C fills up. Aggregate Recall barely moves because the policy still curates the right number of documents. FA Recall drops sharply because the policy retains the wrong documents during the cold-start window.

Example. Q394 asks for an investment-bank founder who was a Sagittarius rugby player born in the 1920s (gold answer: Ron Jarden). Full `Harness-1` identifies Ron Jarden on T2, when the auto-seeded curated set already holds related New Zealand rugby chunks for comparison. It reads `doc_1124` on T4, confirms with `verify` on T6, and ends in 8 turns with FA recall 1.0. The ablation makes 11 `search_corpus` calls before reading `doc_49990` on T18. With no warm C_0 to judge importance against, every candidate looks equally plausible. The trained policy's `verify` call on T20 fails because the supporting chunks were never curated.

M.1.4 EVIDENCE GRAPH

ΔRecall -2.6%, ΔFA -5.4% relative (10 hard fails). With the graph block hidden, the policy can no longer see which entities bridge documents in C. On failing queries it searches more (91.8% of actions vs. 88.9%) and `read_document` drops $3\times$, leaving the curated set without the bridging entity chunks needed to close the chain.

Example. Q754 asks about a youngest-eligible governor elected in a newly formed territory with a self-deprecating slogan and a flamboyant predecessor (gold: Fernando de la Rúa). Full `Harness-1` uses the evidence graph to follow the chain from nickname to territory to predecessor across several documents and reaches the correct answer by T34 (FA 1.0). The ablation has no information on which entities co-occur, so on T39 it tries `"Fernando de la Rúa"` as a free-text guess without the structural anchor. It never connects the name to the "baby bottle" / "pacifier" nickname or the territory-status clue, and ends at FA 0.

M.1.5 VERIFY TOOL

ΔRecall -3.1%, ΔFA -3.9% relative (12 hard fails). When `verify` returns `unavailable`, `read_document` drops from 5.1% to 0.8% on failing queries and search rises to 94.6%.

Example. Q751 asks for the full name of a University of Arizona management professor with a 2010 PhD whose undergraduate university was founded in the 1800s by a specific person (gold: William J. Becker, UCD). Full `Harness-1` reads `doc_15523` on T28, then on T32–T36 traces "William Becker" through the "University College Dublin" bridge. It never calls `verify` explicitly, although the tool stays *implicitly* usable. Without `verify`, the policy has no closed-loop check and rushes to commit documents to C. On T28–T38 it issues ten unverified searches and never re-reads its own candidate, so the curated set ends up with no Becker chunk at all.

Cost of `verify` and choice of verifier. Each `verify` check sends the claim and the text of one document to a small LLM (`gpt-4o-mini`). On real calls this is about 1.1K prompt tokens and 30 completion tokens per checked document, or about $0.0002 per check at list prices, a small part of a cent per episode. Removing `verify` lowers FA recall by 3.9% relative (Table 3), so the check is worth its cost. We also replayed 360 real `verify` calls and scored the small LLM and an NLI model (DeBERTa-large, with its best threshold) against GPT-5.4 as the reference judge (Table 10). The small LLM finds 87% of the supported claims and the NLI model finds 43%. A `verify` claim usually combines several conditions, such as a name, a date range, and a role, and the sentences that support them sit in different parts of the document. The NLI model reads one short passage at a

time, so it rarely sees one passage that supports the whole claim. The small LLM reads the whole document and answers yes only if every part is supported.

Table 10: Verifier choice on 360 replayed `verify` calls, with GPT-5.4 as the reference judge.

Verifier	Supported claims found	Cost per check
Small LLM (`gpt-4o-mini`)	87%	$0.0002
NLI (DeBERTa-large, best threshold)	43%	$0.00011

M.1.6 `REVIEW_DOCS` TOOL

ΔRecall $+2.4\%$, ΔFA -3.9% relative (9 hard fails). The aggregate Recall delta is mildly positive because the policy replaces the bulk sweep with extra turn-level reads. FA Recall still drops, because the policy cannot cheaply scan the curated set when it must combine partial information from many documents. `read_document` drops $4\times$ ($4.0\%\rightarrow1.1\%$) on failing queries and search rises to 93.6%.

Example. Q912 asks for the doctorate university of a researcher named on a 1995–2005 whale-research paper with 12 students and 4 project leaders (gold: University of Hawai'i at Manoa). Full `Harness-1` reads `doc_33637` on T4, calls `verify` on T6, then uses `review_docs` on T7–T13 to sweep the curated set for "Adam Frankel" co-occurring with "UH Manoa". It ends in 15 turns. The ablation reads the same `doc_33637` on T2. With no bulk re-read, it loses track of the candidate after a few new searches and spends T15–T39 cycling through name-extraction guesses.

M.1.7 CONTENT-FINGERPRINT DEDUP

ΔRecall $+4.6\%$, ΔFA $+1.6\%$ relative (0 hard fails). This is the one ablation that nominally improves both metrics. The cause is mechanical. BC+ qrels occasionally contain near-duplicate gold documents (e.g., a press release and an article that quotes it). `Harness-1`'s MinHash-LSH dedup (θ_{Jaccard}=0.85) sometimes merges two such gold variants into one, which costs one gold ID per affected query. Disabling dedup recovers those IDs but inflates the curated context with redundant chunks. Dedup is by design a token-budget mechanism, not a recall mechanism. In a paired answer-accuracy comparison, the deduped sets win the majority of queries despite the slightly lower curated recall. We report this row in the table instead of hiding it, because the trade-off is intended.

M.2 ALL HARNESS MECHANISMS DISABLED

ΔRecall -12.2%, ΔFA -6.4% relative. With every `Harness-1` mechanism off at once, the policy has the same set of actions but no compact state to reason over. Recall drops to 0.513 and FA recall to 0.624. This relative Recall drop is larger than that of any single ablation. The agent keeps searching over a wide space of raw documents but cannot rank what it has seen. This is clear evidence that the trained policy relies on the harness state at inference time. On the same queries, the trained checkpoint searches about as hard but cannot tell candidates apart once the harness is removed.

N EVALUATION RECIPE

Metric Definitions For each query, let A_q denote the annotated *answer-bearing* documents: documents that directly contain or support the final answer. Let R_q denote the full annotated relevant set used for search-quality recall. On benchmarks with extra evidence annotations, R_q may include additional supporting documents beyond A_q; in BrowseComp+ and Web, for example, the answer-bearing documents are a subset of the broader relevant set. On benchmarks without separate answer-document annotations, the evaluator sets $A_q = R_q$.

Let C_q be the final curated set returned by the search subagent, after normalizing retrieved chunks to document identifiers. Let P_q be the trajectory pool: all documents encountered anywhere in the episode, again normalized to document identifiers. We report three recall-oriented metrics:

$$\text{Recall}(q) = \frac{|C_q \cap R_q|}{|R_q|}, \qquad \text{FinalAnswerRecall}(q) = \frac{|C_q \cap A_q|}{|A_q|},$$

$$\text{TrajectoryRecall}(q) = \frac{|P_q \cap R_q|}{|R_q|}.$$

Dataset-level scores are macro-averages over queries. For fact-level datasets, the same definitions apply with facts in place of documents: a fact is counted as covered if any annotated chunk for that fact is retrieved.

Final-Answer Recall can be either higher or lower than Recall. When $A_q \subset R_q$, the denominators differ: Recall averages over all relevant annotations, while Final-Answer Recall averages only over answer-bearing annotations. So a system that finds answer documents but misses many auxiliary supporting documents can have Final-Answer Recall $>$ Recall. A system that finds supporting documents but misses answer documents can have Final-Answer Recall $<$ Recall. Trajectory Recall measures discovery before final curation and works as a rough upper bound on Recall. It can differ from final Recall because the agent may find evidence and later leave it out of C_q.

Reranking for released open-agent harnesses. Search-R1 and Tongyi DeepResearch are evaluated with their released agent harnesses, since the harness is part of each method. These harnesses do not natively return an importance-tagged curated set capped at 30 documents. We therefore standardize their final outputs with a shared reranker. For each query, we collect all unique documents encountered during the episode and normalize chunks to document identifiers. If the trajectory pool contains at most 30 documents, we use the full pool as the curated set C_q without reranking. Otherwise, we rerank the trajectory pool with the shared reranker and keep the top-30 documents as C_q. Recall and Final-Answer Recall are computed from C_q, while Trajectory Recall is computed from the full trajectory pool P_q before reranking.

For Search-R1, this final-set construction usually does not change the retrieval outcome much. Although Search-R1 is evaluated under the same protocol, its released harness typically performs only four to five text-search rounds in our runs. This is consistent with its training setup, which optimizes short search-and-answer trajectories rather than long-horizon evidence collection and curation. As a result, Search-R1 rarely accumulates a trajectory pool much larger than the final document budget. In Table 2, Search-R1's Recall and Trajectory Recall are nearly identical across benchmarks. This suggests that the trajectory pool holds little extra evidence for post-hoc curation to recover.

Search-R1 with a larger pool. Search-R1's short horizon comes from its own policy, which was trained with RL on NQ and HotpotQA to stop after a few rounds. Its prompt allows unlimited searches, and its harness has no setting that changes this. In Table 2 its curated recall equals its trajectory recall on all eight benchmarks, so the top-30 step never removes a relevant document. To give it a larger pool without retraining, we sample it five times per query at temperature 1.0, take the union of the five pools, and apply the same top-30 reranking (Table 11). The pool grows and curated recall follows, but it stays far below `Harness-1` (BC+ 0.082 vs. 0.584) at five times the inference compute. The $+11.4$-point margin over the next open subagent also does not depend on the cap. Of the two open agents that we rerank to the top 30, only Tongyi DeepResearch has a pool larger than 30 documents. Even when we compare `Harness-1`'s capped curated recall (0.730) with Tongyi's uncapped trajectory recall (0.673), `Harness-1` leads by 5.7 points.

Table 11: Search-R1 sampled five times per query (temperature 1.0), pooled, and reranked to the top 30 documents.

Setting	Metric	BC+	Web	Patents	SEC
Search-R1 (Table 2)	Recall	0.053	0.135	0.405	0.109
	Final-Answer Recall	0.074	0.240	0.405	0.155
	Trajectory Recall	0.053	0.135	0.405	0.109
Search-R1 $\times 5$, then rerank	Recall	0.082	0.151	0.442	0.141
	Final-Answer Recall	0.102	0.279	0.442	0.198
	Trajectory Recall	0.118	0.244	0.530	0.168
`Harness-1`	Recall	0.584	0.787	0.890	0.589

Curated-set precision and the 30-document cap. We treat recall as the main metric because precision across systems mostly reflects how many documents each system returns. Frontier models in the Context-1 harness return small sets, about 4 to 10 documents on average, and a small set is precise by construction. `Harness-1` is trained with a recall-weighted objective and usually fills the 30-document budget, which is a cap and not a target. We did not test whether a smaller set would

give better answers. In Figure 8, the small sets of Opus-4.6 give higher answer accuracy than the larger sets of `Harness-1`, in line with Opus-4.6's higher final-answer recall on both benchmarks (0.736 vs. 0.667 on BC+ and 0.696 vs. 0.659 on SEC), so set size and recall are mixed together there. Users who need a smaller context can add one reranking step on top of the curated set.

Inference cost per episode. Table 12 and Figure 7 estimate the cost of one search episode from the input and output tokens in the saved trajectories, at each model's list price in July 2026. `Harness-1` is a fine-tuned `gpt-oss-20b`, so we price it at the open-weight rate of `gpt-oss-20b`. The naive search-add loop appends every search result to the transcript, with no curation, compression, or deduplication, and re-sends the whole transcript at every turn. It generates few tokens, but its input grows each turn, and input tokens make up most of its cost. This makes it by far the most expensive setup. Context-1's pruning cuts this cost by about three to ten times. `Harness-1` is cheap for two reasons: it is a 20B open-weight model, and its working memory keeps every turn within a fixed context budget. It costs less than every frontier searcher except GPT-OSS-120B, which has much lower recall (0.488). It reaches 97% of Opus-4.6's source-family curated recall (0.713 vs. 0.736) at about 4% of its cost ($0.031 vs. $0.74). It also has higher recall than Sonnet-4.6 and Kimi-K2.5 at a quarter or less of their cost. Fable-5 is not part of this comparison.

Table 12: Generated tokens and estimated cost per episode, from the saved trajectories at list prices (July 2026). The cost also counts input tokens, which the table does not list.

Searcher	Harness	Generated tokens per episode	Est. cost per episode
`Harness-1` (20B)	`Harness-1`	8.2K	$0.031
GPT-5.4	Context-1	7.5K	$0.50
Sonnet-4.6	Context-1	7.0K	$0.31
Kimi-K2.5	Context-1	6.3K	$0.154
Opus-4.6	Context-1	4.8K	$0.74
GPT-OSS-120B	Context-1	10.7K	$0.006
GPT-5.4	Naive search-add	∼0.4K	$2.6
Sonnet-4.6	Naive search-add	∼0.4K	$3.1
Kimi-K2.5	Naive search-add	∼0.4K	$0.4
Opus-4.6	Naive search-add	∼0.4K	$5.18
GPT-OSS-120B	Naive search-add	∼0.4K	$0.03

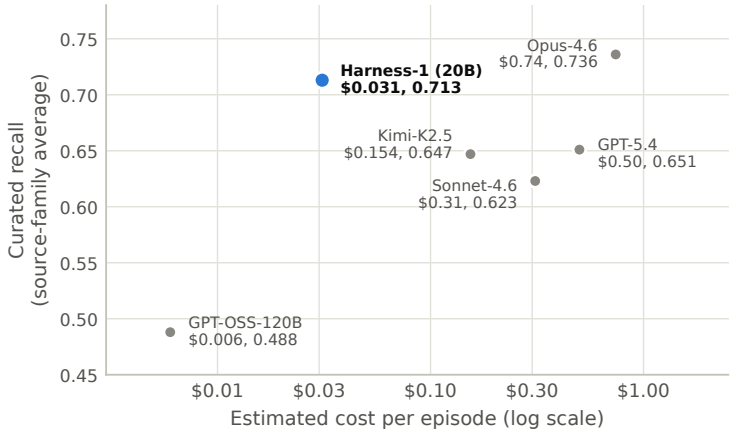

Figure 7: Estimated cost per episode against average curated recall on the four source-family benchmarks. Frontier models run in the Context-1 harness.

Answer-accuracy pipeline (modular RAG protocol). To separate retrieval from generation, we pass each subagent's curated set to a frozen frontier generator. The generator sees *only* the query and curated documents, with no trajectory, intermediate reasoning, or tool-use history. For each (subagent × benchmark) condition, we read the saved trajectory JSON and extract the curated document identifiers and their chunk text from the observation logs. We format them with the prompt

in Appendix I.3 and send them to four generators: GPT-5.4, Sonnet-4.6, Opus-4.6, and Kimi-K2.5. Fable-5 is not used in this protocol, either as searcher or as generator. We use the same benchmark instances as the main evaluation. Closed-Book and Naive-RAG (top-10 BM25+dense) serve as reference baselines. The judge (Appendix I.4) is GPT-5.4 at T=0 with the BrowseComp+ template, and we report the strict CORRECT rate. Every (query, generator, generated-answer, verdict) tuple is saved.

O MODULAR RAG: CURATED SETS YIELD HIGHER ANSWER ACCURACY

Search quality measures what the subagent returns, but what matters in the end is the answer that the downstream system gives. Under the modular RAG protocol described above, the answer-generator LLM sees only the curated set, never the trajectory. So any accuracy difference between subagents comes from the curated set alone.

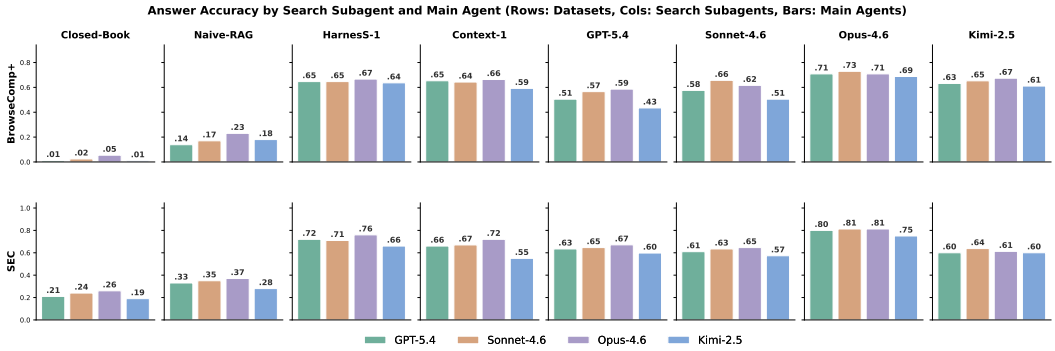

Figure 8: **Answer accuracy** (fraction of questions answered correctly) under the modular RAG protocol. Each row of subplots is a benchmark, and each subplot is a search method (or baseline). Each bar color is a different frontier answer generator. *Closed-Book*: generator answers from parametric knowledge alone. *Naive RAG*: single BM25+dense top-10 chunk retrieval, no agentic search. In all other groups, an agentic search subagent curates the document set, and the generator receives *only* the query and curated documents, not the trajectory. Judging follows the official BrowseComp+ template (GPT-5.4, T=0).

Figure 8 shows three patterns. First, Closed-Book and Naive RAG are weak on the hardest benchmark shown (BC+). This confirms that these tasks require deep agentic search and that parametric knowledge and single-hop retrieval are not enough. Second, the ranking of subagents by answer accuracy broadly follows their search-quality ranking in Table 2. Better curated sets give higher answer accuracy with all four generators. Third, the variation across generators within a subplot is modest for strong searchers, since every frontier generator can reach the right answer when given the right documents. It is larger for weaker searchers, where the generator must make up for missing evidence. Because the generator never sees the trajectory, the ranking of subagents is a direct test of curated-set quality. `Harness-1`'s curated sets give answer accuracy at least as high as Context-1's for every generator.

P HARNESS AS CONFOUND: SAME LLM, DIFFERENT HARNESS

One concern is that all frontier baselines are evaluated in the Context-1 harness and not in `Harness-1`'s harness. This is deliberate. We compare *trained agents* (Context-1 RL and `Harness-1` RL), each in its own harness, while the frontier LLMs serve as untrained baselines that share Context-1's harness as a common reference environment.

To measure what the harness adds on its own, we run GPT-5.4 and Opus-4.6 in both harnesses on the same query sets as Table 2, for the four source-family benchmarks. We also run the trained `Harness-1` policy in Context-1's harness on all eight benchmarks. Figure 9 shows curated recall, and Tables 13 and 14 give all numbers.

Our harness helps both frontier models. It raises curated recall, final-answer recall, and trajectory recall for GPT-5.4 and Opus-4.6 on every benchmark. Averaged over the four benchmarks, GPT-5.4

gains 4.0 points of curated recall (0.651 to 0.691) and Opus-4.6 gains 3.3 points (0.736 to 0.769). The frontier rows of Table 2, which use Context-1's harness, are therefore below what these models reach in our harness.

The trained policy still beats GPT-5.4 in the same harness. With both in our harness, `Harness-1` reaches 0.713 average curated recall on these benchmarks and GPT-5.4 reaches 0.691. Opus-4.6 stays ahead in both harnesses. The lead over GPT-5.4 comes from BC+ and SEC; GPT-5.4 is ahead on Web and Patents. So the comparison in Table 2 is not only a result of which harness each model uses.

A trained policy does best in its own harness. In Context-1's harness, the `Harness-1` policy reaches 0.510 average curated recall over the eight benchmarks, below both its own result in our harness (0.730) and Context-1 in its own harness (0.603). The trained `Harness-1` policy works best in the harness it was trained in. Neither the harness alone nor the policy alone explains the result: what we call `Harness-1` is the policy and the harness trained together.

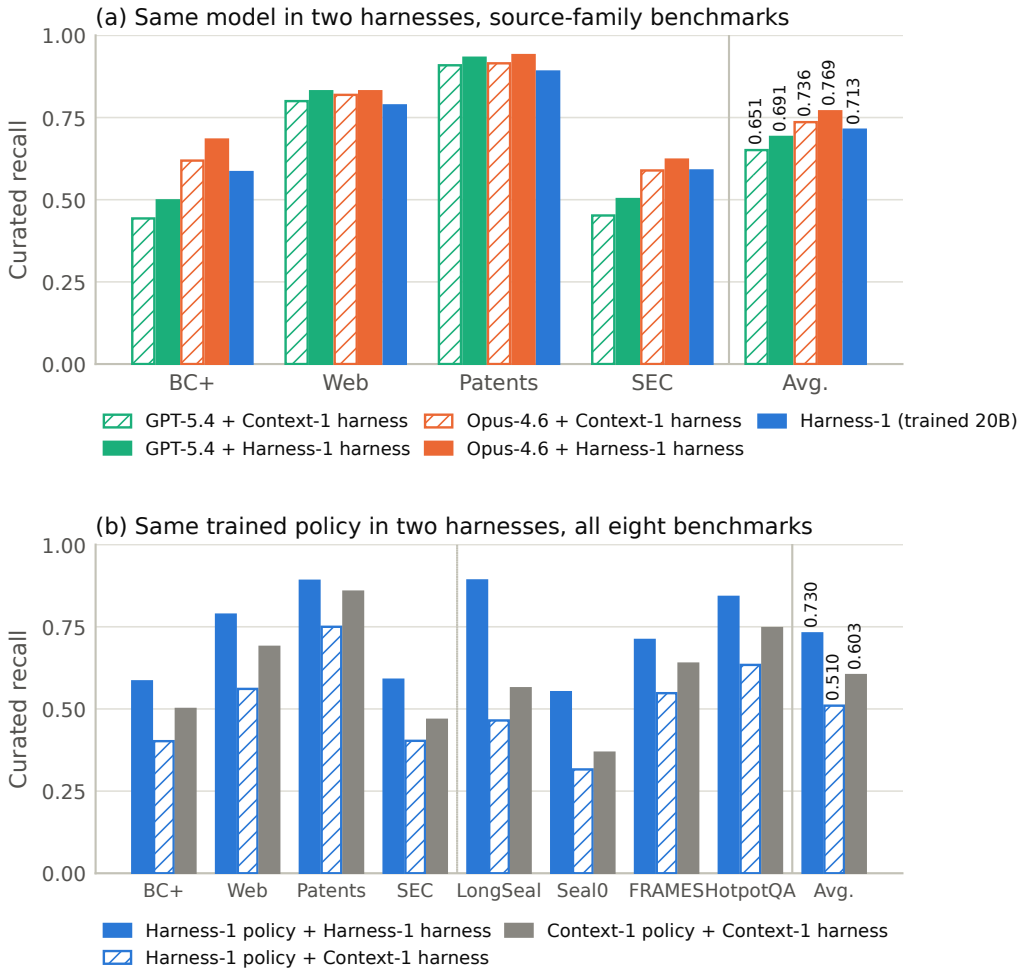

Figure 9: **Swapping harnesses.** (a) GPT-5.4 and Opus-4.6 in Context-1's harness (hatched) and in our harness (solid), next to the trained `Harness-1`. (b) The trained `Harness-1` policy in its own harness and in Context-1's harness, next to Context-1 in its own harness. The number above each bar in the last group is the average.

Table 13: Same model, two harnesses, on the source-family benchmarks. Each cell shows [Context-1 harness, our harness]. The Context-1 values repeat Table 2.

Model	Metric	BC+	Web	Patents	SEC	Avg.
`Harness-1` (20B)	Recall	0.584	0.787	0.890	0.589	0.713
	Final-Answer Recall	0.667	0.845	0.890	0.659	0.765
	Trajectory Recall	0.665	0.881	0.973	0.677	0.799
Opus-4.6	Recall	[0.619, 0.683]	[0.819, 0.830]	[0.915, 0.940]	[0.589, 0.622]	[0.736, 0.769]
	Final-Answer Recall	[0.736, 0.771]	[0.928, 0.965]	[0.915, 0.940]	[0.696, 0.711]	[0.819, 0.847]
	Trajectory Recall	[0.634, 0.696]	[0.832, 0.870]	[0.939, 0.954]	[0.601, 0.656]	[0.752, 0.794]
GPT-5.4	Recall	[0.443, 0.498]	[0.800, 0.830]	[0.909, 0.932]	[0.452, 0.502]	[0.651, 0.691]
	Final-Answer Recall	[0.499, 0.572]	[0.858, 0.875]	[0.909, 0.932]	[0.465, 0.533]	[0.683, 0.728]
	Trajectory Recall	[0.476, 0.551]	[0.821, 0.862]	[0.919, 0.939]	[0.464, 0.525]	[0.670, 0.719]

Table 14: Same trained policy, two harnesses (curated recall).

Policy and harness	BC+	Web	Patents	SEC	LongSeal	Seal0	FRAMES	HotpotQA	Avg.
`Harness-1` policy, our harness (Table 2)	0.584	0.787	0.890	0.589	0.891	0.551	0.710	0.841	0.730
`Harness-1` policy, Context-1 harness	0.402	0.561	0.750	0.403	0.465	0.316	0.548	0.634	0.510
Context-1 policy, Context-1 harness (Table 2)	0.500	0.689	0.857	0.467	0.563	0.367	0.638	0.746	0.603

Q Qualitative Case Studies

The abridged trajectories below show where the harness changes the search process. Auto-seeding creates an editable state, curation promotes evidence, exact search finds hard-to-find facts, backtracking removes false leads, and full-document reading settles the final answer. We omit routine prompt boilerplate, redundant curation calls, false-positive documents, and repeated metadata. The full logs are kept separately. Case Study E also shows that a wrong step can be undone. The policy first rejects the correct entity, later reads evidence against that choice, and returns to it before it ends the search. The page for the correct game had stayed in the curated set at low importance, so nothing was lost. Curation can always be revised, and the search history stays in working memory, so fixing a step does not erase the record that it was tried.

Q.1 Case Study A: Stateful Search over a Drug-Approval Query

BrowseComp+ query

Query. An agent is not indicated for patients who weigh 30 kg or are 10 years old. It can be taken via oral administration, and one of the side effects reported is nausea. A study from 2022 mentions that its half-life is less than 10 hours and that the liver primarily eliminates it. The study is authored by two people, published in December, and features 31 references. The application for tablets containing the agent was received on March 23, and it was approved eight months later by a letter. What is the full name (as mentioned in the letter) of the Regulatory Project Manager to be called in case of any questions?

Gold answer. Alicia Moruf

Outcome. 18 turns, tool diversity 6, curated recall 1.0, final-answer recall 1.0.

Why this case is interesting. The query asks for one name hidden in regulatory documents. The agent must connect label constraints, a pharmacology study, an NDA package, and an approval letter naming the project manager. Here the harness does useful bookkeeping beyond storing a transcript. Auto-seeding creates the first regulatory evidence set, and curation separates maribavir and FDA-package evidence from distractors. After a brittle exact phrase match fails, the agent recovers with a search by NDA number, and full-document reading exposes the answer sentence.

How the trajectory is abridged. The full log keeps every token. Here we omit the system prompt, repeated curation confirmations, low-value false-positive documents, and two no-op verify/review calls caused by chunk-id mismatch. The retained turns preserve the model reasoning, exact tool calls, decisive returns, and state snapshots needed to see the harness at work.

Trajectory overview.

Turn	Tool	Policy move	Harness state / why it matters
0	search	Search all medical clues at once	auto-seed creates the first editable curated set (6 docs, because only 6 were returned)
1	curate	Promote Livtencity and pharmacology candidates	curation separates strong maribavir evidence from low-value distractors
2	search	Search the 2022 maribavir study	connects half-life, hepatic metabolism, two authors, and references
3	curate	Promote study and label chunks	working memory now contains drug identity plus PK evidence
4	search	Search FDA approval-letter wording	same document family reappears with NDA context
5	curate	Consolidate FDA package and study chunks	curated state keeps both regulatory and pharmacology evidence
6	search	Search by regulatory-manager wording	narrows to approval package chunks
7	curate	Add later FDA-letter chunks	candidate pool now includes the likely answer document
8	grep	Exact grep for phrase	a failed exact match prevents overtrusting a brittle pattern
10	search	Search by NDA number	needle appears: Alicia Moruf in the approval letter
11	curate	Promote NDA-letter chunk	final evidence enters curated memory
12	read	Read full approval-letter chunk	outer-tier memory exposes the decisive text
13	curate	Reaffirm final evidence	curated set keeps approval letter high

PHASE 1 - AUTO-SEEDED WORKING MEMORY FROM A BROAD MEDICAL SEARCH

Reasoning excerpt

Key constraints: agent not indicated for patients >=30kg or 10y; oral administration; side effect nausea; 2022 study mentions half–life <10h, primarily liver elimination; study by two authors, December, 31 references; tablet application received 23 Mar; approved 8 months later via letter identifying a Regulatory Project Manager. Likely a US FDA approval letter for an oral drug for weight >=30kg/age >=10y contraindication. The unique medicine clue is the age/weight contraindication plus nausea side effect. Search angle: distinct clinical label and side effect plus half–life/liver elimination trait.

Exact tool call


```
[
  {
    "tool": "search_corpus",
    "params": {
      "query": "oral contraindicated for under 30 kg age 10 nausea half–life less than 10 hours liver eliminated 2022 study December 31 references FDA approval letter Regulatory Project Manager March 23 seventeen Feb 2023 app tablets received March 23 approved eight months letter"
    }
  }
]
```


Search return: first hit creates the regulatory anchor

DOCUMENT ID: 59631_0 (743 tokens) CENTER FOR DRUG EVALUATION AND RESEARCH Approval Package for: APPLICATION NUMBER: 215596Orig1s000 Trade Name: LIVTENCITY oral tablets Generic or Proper Name: maribavir Sponsor: Takeda

Pharmaceuticals USA, Inc. Approval Date: November 23, 2021 Indication: For the treatment of adults and pediatric patients 12 years of age and older and weighing at least 35 kg with post−transplant cytomegalovirus (CMV) infection/disease that is refractory to treatment (with or without genotypic resistance) with ganciclovir, valganciclovir, cidofovir or foscarnet. CENTER FOR DRUG EVALUATION AND RESEARCH 215596Orig1s000 CONTENTS Reviews / Information Included in this NDA Review. Approval Letter X Other Action Letters Labeling X REMS Officer/Employee List X Integrated Review(s) X Product Quality Review(s) X Other Reviews X Risk Assessment and Risk Mitigation Review(s) X Proprietary Name Review(s) X Administrative/Correspondence Document(s) X CENTER FOR DRUG EVALUATION AND RESEARCH APPLICATION NUMBER: 215596Orig1s000 APPROVAL LETTER NDA 215596 NDA APPROVAL Takeda Pharmaceuticals USA, Inc. Attention: Michael A. Cronin, PharmD Director, Global Regulatory Affairs 650 ...

Curation update

Curated set updated (6/30): 25814[high], 59631[high], 15546[fair], 50858[fair], 66387[low], 97801[low]

State after turn 1

pool_size: 6
curated_size: 6/30
importance: very_high=0, high=2, fair=2, low=2
top_curated:
- 25814 [high]
- 59631 [high]
- 15546 [fair]
- 50858 [fair]
- 66387 [low]
- 97801 [low]

PHASE 2 - THE POLICY PIVOTS TO PHARMACOKINETIC EVIDENCE

Reasoning excerpt

The first search hit a known FDA submission doc for Livtencity (maribavir). That fits oral administration, nausea side effect, and contraindication at 30kg/10y, but the drug class/youth contraindication constraint is strong. We need the 2022 study page and the approval letter identifying the Regulatory Project Manager. The search angle using the study constraints is still relevant; I'll first search for the 2022 study on maribavir to bridge to the approval letter.

Exact tool call

```
[
  {
    "tool": "search_corpus",
    "params": {
      "query": "maribavir 2022 study liver eliminated half−life less than 10 hours oral migration
 31 references December two authors"
    }
  }
]
```

Search return: study and label evidence

DOCUMENT ID: 83797_4 (723 tokens) As a postmarketing commitment, the FDA is requiring the manufacturer to conduct phenotypic analysis of maribavir against human CMV. Substitutions designated as a high priority for analysis are pUL97 M460I/T, A594E/P/T/V, L595F/W, and C603R/W/Y. Substitutions designated as medium priority are pUL97 A440V, V466M, A591V, E596G, K599E, and C607F/Y. Substitutions designated as low priority are pUL97 P132L, L405P, C518Y, I610T, and A613V; pUL27 P10L, N289D, H29Y, D298G, N300G, P307L, V310A, D351N, and I367V; and pUL54 S290R and K475S.18 Maribavir does not prolong the QT interval to a clinically important extent.1,19 Pharmacokinetics Following oral administration, median T max is 1 to 3 hours.1,19,20 Plasma exposure (C max and AUC) increased approximately dose proportionally following a single dose of 50 to 1600 mg and multiple doses of up to 2400 mg/day.1,20 Compared with administration in the fasted state, administration with a moderate−fat meal resulted in a small reduction in AUC 0−infinity (geometric mean ratio, 0.864; 90% CI, 0.804−0.929) and C max (0.722; 90% CI, 0.656−0.793) . Steady state is reached within 2 days with twice−daily dosing.1 Mean steady−state volume of distribution is 27.3 L. Maribavir is 98% plasma protein bound. The blood−to−plasma ratio is 1.37.1,20 Maribavir does not appear to cross the blood−brain barrier.9 Maribavir, the parent drug, is responsible for the ...

DOCUMENT ID: 17183_8 (746 tokens) due to the parent drug. Following oral administration, plasma maribavir exposure (Cmax and AUC) increased approximately dose−proportionally following a single dose of 50 to 1600 mg (0.125 to four times the recommended dose) and multiple doses up to 2400 mg per day (three times the recommended daily dose). Maribavir PK is time−independent. With twice−daily dosing, steady state is reached within 2 days, with mean accumulation ratios of Cmax and AUC ranging from 1.37 to 1.47. The pharmacokinetic properties of maribavir following administration of LIVTENCITY are displayed in Table 4. The multiple−dose pharmacokinetic parameters are provided in Table 5. 9 Reference ID: 4893709 This label may not be the latest approved by FDA. For current labeling information, please visit Table 4: Pharmacokinetic Properties of Maribavir Absorptiona Tmax (h), median 1.0 to 3.0 Distribution Mean apparent steady−state volume of distribution (Vss, L) 27.3 % bound to human plasma proteins 98.0 across the concentration range of 0.05−200 ug/mL Blood−to plasma ratio 1.37 Elimination Major route of elimination Hepatic metabolism Half−life (t1/2) in transplant patients (h), mean 4.32 Oral clearance (CL/F) in transplant patients (L/h), mean 2.85 Metabolism Metabolic pathwaysb CYP3A4 (major) and CYP1A2 (minor) Excretion % of dose excreted as total 14C (unchanged drug) in urinec 61 (<2) % of dose excreted as total 14 C ...

DOCUMENT ID: 73877_2 (477 tokens) B: May be acceptable. Either animal studies show no risk but human studies not available or animal studies showed minor risks and human studies done and showed no risk. C: Use with caution if benefits outweigh risks. Animal studies show risk and human studies not available or neither animal nor human studies done. D: Use in LIFE−THREATENING emergencies when no safer drug available. Positive evidence of human fetal risk. X: Do not use in pregnancy. Risks involved outweigh potential benefits. Safer alternatives exist. NA: Information not available.Pharmacology Mechanism of Action Antiviral activity mediated by competitive inhibition of the protein kinase activity of human CMV enzyme pUL97, which inhibits protein phosphorylation Maribavir inhibited human CMV replication in virus yield reduction, DNA hybridization, and plaque reduction assays in human lung fibroblast cell line, human embryonic kidney, and human foreskin fibroblast cells Absorption Peak plasma time: 1−3 hr AUC: 128 ug⋆hr/mL Peak plasma concentration: 17.2 ug ⋆hr/mL Distribution Vd (steady−state): 27.3 L Plasma bound: 98% Blood−to−plasma ratio: 1.37 Metabolism Metabolized by CYP3A4 (major) and CYP1A2 minor Elimination Clearance: 2.85 L/hr Half−life: 4.32 hr Excretion: Urine (61% [<2% unchanged]); feces (14% [5.7%]) Administration Oral Administration Take with or without food Storage Store at 20−25oC (68−77oF); excursions ...

Curation update

Curated set updated (9/30): 17183[high], 25814[high], 59631[high], 83797[high], 15546[fair], 50858[fair], 73877[fair], 66387[low], 97801[low]

State after turn 3

pool_size: 9
curated_size: 9/30
importance: very_high=0, high=4, fair=3, low=2
top_curated:
- 17183 [high]
- 25814 [high]
- 59631 [high]
- 83797 [high]
- 15546 [fair]
- 50858 [fair]
- 73877 [fair]
- 66387 [low]
- 97801 [low]

PHASE 3 - EXACT SEARCH MISSES, SO STATE PREVENTS A BLIND LOOP

Exact grep call

```
[
  {
    "tool": "grep_corpus",
    "params": {
      "pattern": "Maribavir.*Regulatory Project Manager"
    }
  }
]
```

Grep return

No results found
[Context: 11553/30720]

Recovery search by NDA number

```
[
  {
    "tool": "search_corpus",
    "params": {
      "query": "NDA 215596 letter to maribavir regulatory project manager NDA 215596"
    }
  }
]
```

Decisive return around answer

[... clipped before ...]
EPORTING REQUIREMENTS We remind you that you must comply with reporting requirements for an approved NDA (21 CFR 314.80 and 314.81). POST APPROVAL FEEDBACK MEETING New molecular entities qualify for a post approval feedback meeting. Such meetings are used to discuss the quality of the application and to evaluate the communication process during drug development and marketing application review. The purpose is to 4 For the most recent version of a guidance, check the FDA guidance web page at 5 6 Reference ID: 4893709 NDA 215596 Page 6 U.S. Food and Drug Administration Silver

Spring, MD 20993 learn from successful aspects of the review process and to identify areas that could benefit from improvement. If you would like to have such a meeting with us, call the Regulatory Project Manager for this application. If you have any questions, call Alicia Moruf, PharmD, MPH, RAC–US, Regulatory Project Manager, at 301–796–3953. Sincerely, {See appended electronic signature page} John Farley, MD, MPH Director Office of Infectious Diseases Office of New Drugs Center for Drug Evaluation and Research ENCLOSURE(S): – Content of Labeling o Prescribing Information o Patient Package Insert – Carton and Container Labeling Reference ID: 4893709 ––––––––––––––––––––––––This is a representation of an electronic record that was signed electronically. Following this are manifestations of any and all electronic signatures for this electronic record.

DOCUMENT ID: 73877_0 (719 tokens)
–––
title: maribavir (Rx)
date: 2025–01–01
–––
Dosing & Uses

Dosage Forms & Strengths

tablet

– 200mg

Cytomegalovirus Infection

Indicated for t
[... clipped after ...]

Curation update

Curated set updated (10/30): 17183[high], 25814[high], 15546[fair], 50858[fair], 59631[fair], 73877[fair], 66387[low], 72434[low], 83797[low], 97801[low]

State after turn 11

pool_size: 11
curated_size: 10/30
importance: very_high=0, high=2, fair=4, low=4
top_curated:
  – 17183 [high]
  – 25814 [high]
  – 15546 [fair]
  – 50858 [fair]
  – 59631 [fair]
  – 73877 [fair]
  – 66387 [low]
  – 72434 [low]
  – 83797 [low]
  – 97801 [low]

PHASE 4 - FULL-DOCUMENT READING CONFIRMS THE FINAL EVIDENCE

Exact read call


```
[
  {
    "tool": "read_document",
    "params": {
      "doc_id": "59631_3"
    }
  }
]
```


Full-document excerpt around answer

[... clipped before ...]
ilable at FDA.gov.5 Information and Instructions for completing the form can be found at FDA. gov.6 REPORTING REQUIREMENTS We remind you that you must comply with reporting requirements for an approved NDA (21 CFR 314.80 and 314.81). POST APPROVAL FEEDBACK MEETING New molecular entities qualify for a post approval feedback meeting. Such meetings are used to discuss the quality of the application and to evaluate the communication process during drug development and marketing application review. The purpose is to 4 For the most recent version of a guidance, check the FDA guidance web page at 5 6 Reference ID: 4893709 NDA 215596 Page 6 U.S. Food and Drug Administration Silver Spring, MD 20993 learn from successful aspects of the review process and to identify areas that could benefit from improvement. If you would like to have such a meeting with us, call the Regulatory Project Manager for this application. If you have any questions, call Alicia Moruf, PharmD, MPH, RAC–US, Regulatory Project Manager, at 301–796–3953. Sincerely, {See appended electronic signature page} John Farley, MD, MPH Director Office of Infectious Diseases Office of New Drugs Center for Drug Evaluation and Research ENCLOSURE(S): – Content of Labeling o Prescribing Information o Patient Package Insert – Carton and Container Labeling Reference ID: 4893709 ––––––––––––––––––––––––This is a representation of an electronic record that was signed electronically. Following this are manifestations of any and all electronic signatures for this electronic record .––––––––––––––––––––––/s/ ––––––––––––––––––––––JOHN J FARLEY 11/23/2021 03:34:13 PM Signature Page 1 of 1 Reference ID: 4893709 (
[Context: 10329/30720]

Final curation update

Curated set updated (10/30): 17183[high], 25814[high], 59631[high], 15546[fair], 50858[fair], 73877[fair], 66387[low], 72434[low], 83797[low], 97801[low]

State after turn 13

pool_size: 11
curated_size: 10/30
importance: very_high=0, high=3, fair=3, low=4
top_curated:
- 17183 [high]
- 25814 [high]
- 59631 [high]
- 15546 [fair]
- 50858 [fair]
- 73877 [fair]
- 66387 [low]
- 72434 [low]
- 83797 [low]

– 97801 [low]

Q.2 Case Study B: Exact-Date Recovery in SEC Filings

SEC query

Query. There is a company where a senior financial executive with over two decades of consumer packaged goods experience joined as a senior vice president in the early spring of the mid-2020s. Around the early summer of the same year, a different long-serving financial executive concluded their tenure after announcing their decision to step down several months earlier. That same date marked when the spring hire assumed the top financial role, succeeding the retiring executive. On what date did this leadership transition in the financial executive role become effective?

Gold answer. July 3, 2025

Outcome. 9 turns, tool diversity 5, curated recall 1.0, final-answer recall 1.0.

Why this case matters. SEC queries often fail when the agent treats filings as isolated snippets. This trajectory shows the source-family behavior behind the SEC gains. Auto-seeding starts inside the right filing family, and curation keeps the relevant 8-Ks. `read_document` then exposes the full text of the executive transition, and `grep_corpus` verifies the exact effective date in later filings.

Trajectory overview.

Turn	Tool	Policy move	Harness state / why it matters
0	`search`	Search executive-transition clues in SEC filings	first hit already binds company, outgoing CFO, incoming CFO, and effective date
1	`curate`	Promote the relevant 8-Ks	curated set separates the correct filing family from other finance hits
3	`read`	Read the May 8-K	full text resolves succession details rather than relying on snippets
4	`curate`	Keep the read filing high	state retains the source document for final answer recall
6	`grep`	Grep the exact effective date	regex recovers later proxy mentions of the same transition
7	`curate`	Promote corroborating proxy chunks	final state keeps multiple filings about the CFO transition

Phase 1 - Auto-seeding finds the right company and date

Exact search call


```
[
  {
    "tool": "search_corpus",
    "params": {
      "query": "spring 2024 senior vice president food consumer packaged goods 20 years CPG effective CFO transition summer 2024 retire step down senior financial executive announced decision months earlier"
    }
  }
]
```

Decisive first return

DOCUMENT ID: 0001104659−25−006698_0 (387 tokens)
Company: Simply Good Foods Co
Filing Date: 2025−01−28
Form: 8−K
Item: Item 5.02
Chunk 0
−−−
Item 5.02 Departure of Directors or Certain Officers; Election of
Directors; Appointment of Certain Officers; Compensatory Arrangements of Certain Officers.

(b) On January 27, 2025, Shaun P. Mara, Chief Financial Officer
of The Simply Good Foods Company (the ” Company”), notified the Company of his decision to retire effective July 3, 2025.
Mr. Mara will remain as Chief Financial Officer through the date of his retirement, at which time he will resign from all positions
with the Company and its subsidiaries.

In connection with Mr. Mara’s retirement, on January 28,
2025, the Company announced the hiring of Mr. Christopher J. Bealer as Senior Vice President of Finance of the Company, as of
April 1, 2025. Mr. Bealer is expected to succeed Mr. Mara as Chief Financial Officer upon Mr. Mara’s retirement
on July 3, 2025.

Mr. Bealer has almost 23 years of experience in consumer packaged
goods and consumer durables in North America and global markets. In his most recent executive leadership position as EVP Corporate Controller
at Reckitt Benckiser Group PLC (” Reckitt”), Mr. Bealer leads a global team of over 600 employees accountable for global
controls, financial shared services, group reporting and accounting, as well as leading the functional operating model transformation.
Prior to his current and prior roles at Reckitt, Mr. Bealer served as Regional CFO for Whirlpool Corporation for both the North American
and Europe, Middle East and Africa (EME
[... clipped after ...]

Curation update

Curated set updated (7/30): 0001104659−25−006698_0[high], 0001104659−25−053287_0[high], 0001104659−25−121934_20[high], 0000909832−25−000101_19[fair], 0001364954−25−000039_57[fair], 0001673985−25−000053_24[fair], 0001530950−25−000260_18[low]

State after turn 1

pool_size: 7
curated_size: 7/30
importance: very_high=0, high=3, fair=3, low=1
top_curated:
- 0001104659−25−006698_0 [high]
- 0001104659−25−053287_0 [high]
- 0001104659−25−121934_20 [high]
- 0000909832−25−000101_19 [fair]
- 0001364954−25−000039_57 [fair]
- 0001673985−25−000053_24 [fair]
- 0001530950−25−000260_18 [low]

PHASE 2 - FULL-DOCUMENT READING KEEPS THE ANSWER GROUNDED

Exact read call

```
[
  {
    "tool": "read_document",
    "params": {
      "doc_id": "0001104659–25–053287_0"
    }
  }
]
```

Full-document excerpt around transition

```
# Document (1500 tokens)
Company: Simply Good Foods Co
Filing Date: 2025–05–28
Form: 8–K
Item: Item 5.02
Chunk 0
–––
Item 5.02

Departure of Directors or Certain Officers; Election of Directors; Appointment of Certain
Officers; Compensatory Arrangements of Certain Officers.

Appointment of New CFO

In connection with the previously announced Chief Financial Officer
transition, on May 23, 2025, The Simply Good Foods Company (the " Company") appointed
Christopher J. Bealer as the Company's
Chief Financial Officer effective July 3, 2025. Mr. Bealer is currently serving as the Company's
 Senior Vice President,
Finance having joined the Company on April 1, 2025. As Chief Financial Officer, Mr. Bealer
will continue to report to Geoff
E. Tanner, the Company's President and Chief Executive Officer.

Mr. Bealer, age 48, has almost 23 years of experience in consumer–packaged
goods and consumer durables in North America and Global markets. Mr. Bealer was most
recently EVP Corporate Controller at Reckitt
Benckiser Group PLC (" Reckitt") from March 2024 to March 31, 2025, and CFO Health at
Reckitt from April 2021
to February 2024. In his EVP Corporate Controller role with Reckitt, Mr. Bealer led a global
team of over 600 employees accountable
for global controls, financial shared services, group reporting and accounting, as well as
leading the functional operating model transformation.
Prior to his roles at Reckitt, M
[... clipped after ...]
```

State after turn 4

```
pool_size: 7
curated_size: 7/30
importance: very_high=0, high=3, fair=3, low=1
top_curated:
  – 0001104659–25–006698_0 [high]
  – 0001104659–25–053287_0 [high]
  – 0001104659–25–121934_20 [high]
```

– 0000909832–25–000101_19 [fair]
– 0001364954–25–000039_57 [fair]
– 0001673985–25–000053_24 [fair]
– 0001530950–25–000260_18 [low]

PHASE 3 - EXACT GREP ADDS CORROBORATING FILINGS

Exact grep call

```
[
  {
    "tool": "grep_corpus",
    "params": {
      "pattern": "effective July 3, 2025"
    }
  }
]
```

Grep return

[... clipped before ...]
| CHRISTOPHER J.
BEALER | | | | TIMOTHY R.
KRAFT | | | | AMY C.
HELD | | | | RYAN A.
THOMAS | | | | SHAUN P.
MARA | |
| | | President and Chief Executive Officer | | | | Chief Financial Officer | | | | Chief Legal and Corporate Development Officer | | | | SVP and Chief Human Resources Officer | | | | SVP and General
Manager Atkins and OWYN | | | | Former Chief Financial Officer | |

Mr. Bealer became an NEO for the first time in fiscal year 2025. Mr. Bealer was hired on March 31, 2025. Mr. Mara retired as our Chief Financial Officer and resigned effective July 3, 2025. Mr. Mara's executive compensation is outlined further in the compensation tables beginning on page 54 but is otherwise not discussed with other NEOs except where noted. Executive Summary Our compensation program for our NEOs and other executive officers is designed to meet the following primary objectives:

| | | Management Development and Continuity.
Attract, retain and motivate individuals of superior ability and managerial talent to develop, grow and manage our business by offering competitive compensation opportunities with both short–term and significant long–term components | | | | Pay for Performance.
[... clipped after ...]

Curation update

Curated set updated (9/30): 0001104659–25–006698_0[high], 0001104659–25–053287_0[high], 0001104659–25–121934_20[high], 0001104659–25–121934_21[high], 0001104659–25–121934_42[high], 0000909832–25–000101_19[fair], 0001364954–25–000039_57[fair], 0001673985–25–000053_24[fair], 0001530950–25–000260_18[low]

State after turn 7

pool_size: 10
curated_size: 9/30
importance: very_high=0, high=5, fair=3, low=1
top_curated:
- 0001104659–25–006698_0 [high]
- 0001104659–25–053287_0 [high]
- 0001104659–25–121934_20 [high]
- 0001104659–25–121934_21 [high]
- 0001104659–25–121934_42 [high]
- 0000909832–25–000101_19 [fair]
- 0001364954–25–000039_57 [fair]
- 0001673985–25–000053_24 [fair]
- 0001530950–25–000260_18 [low]

Q.3 Case Study C: Fan-Out Search Resolves an Ambiguous Transfer Query

FRAMES query

Query. Which university did the actor who has appeared in the most movies in the American pie film series (including spinoffs, as of 2020) deliver a commencement address at in 2012?
Gold answer. Dalhousie University, in Halifax, Nova Scotia
Outcome. 16 turns, tool diversity 6, curated recall 1.0, final-answer recall 1.0.

Why this case is interesting. The first hop is ambiguous because several *American Pie* actors are plausible. The commencement-address clue helps only after the candidate actors are narrowed down. The trajectory shows why fan-out and tagged memory matter. The agent keeps competing actor hypotheses open and switches from Seann William Scott to Eugene Levy. It then uses exact search and full-document reading to expose the Dalhousie address evidence.

How the trajectory is abridged. The full log includes many generic web snippets from fan-out search. Here we omit generic pages, repeated low-impact curation text, and the no-op verifier call. We keep the reasoning and exact calls for the decisive turns: fan-out, candidate switch, exact search, full read, and final promotion.

Trajectory overview.

Turn	Tool	Policy move	Harness state / why it matters
0	`fan-out`	Fan out over actor identity and address clues	auto-seed stores 8 docs; broader pool contains the key American Pie character page
1	`curate`	Promote American Pie cast evidence	curation shifts memory from generic hits to candidate actors
2	`search`	Search Eugene Levy commencement	tests the candidate implied by Noah Levenstein
3	`curate`	Add Eugene Levy pages	memory now tracks the alternative to Seann William Scott
4	`search`	Search most-appearances clue	keeps the actor-identity subproblem explicit
5	`curate`	Promote cast pages	curated set expands but stays tagged
6	`search`	Search Seann William Scott address	tests a competing hypothesis
7	`curate`	Demote weak address hits	low tags protect the final set from noise
8	`grep`	Exact grep for both candidates	fails to settle the address but broadens evidence
9	`curate`	Fill remaining memory slots	curated set reaches cap; tags govern later eviction
12	`search`	Search Eugene Levy + 2012 address	directly targets the missing bridge

Turn	Tool	Policy move	Harness state / why it matters
13	`curate`	Add university/address candidates	state prepares for final read
14	`read`	Read Eugene Levy page	full text reveals Dalhousie University
15	`curate`	Promote Eugene Levy and Dalhousie	final answer evidence receives very_high/high tags

PHASE 1 - FAN-OUT KEEPS COMPETING HYPOTHESES ALIVE

Reasoning excerpt

We need: Which university did the actor who has appeared in the most movies in the American Pie film series (including spinoffs, as of 2020) deliver a commencement address at in 2012.

We can break it down.

Candidates: Need actor with most appearances in American Pie series.

List of movies: American Pie (1999), American Pie 2 (2001), American Pie 3 (2003), American Pie 4 (2008), American Wedding (2003?), Wait 2003? Actually American Wedding (2003) is the third? Let's check full series: American Pie (1999); American Pie 2 (2001); American Wedding (2003); American Reunion (2012); American Pie Presents: Band Party (2007); American Pie Presents: The Wedding (2009); American Pie Presents: The Book of Love (2010); American Pie Presents: The Naked Mile (2006); American Pie Presents: The First Kiss (2009). Also spinoffs: American Wedding (2003), American Reunion 2012. The actor who appears in the most movies: I recall Skyler or others? Let's recall main casts: Jason Biggs (Jim), Alyson Hannigan (Michelle), Natasha Lyonne (Stifler?), actually Stifler is Seann William Scott. He appears in many? Let's count: Stifler from first through fourth, plus spinoffs? Seann William Scott appears in 1) American Pie 2? Wait Stifler is in first: yes first. In American Pie 2? yes Stifler. In American Wedding? maybe he appears as Stifler. In American Pie Presents: The Book of Love? Not sure. Also spinoffs: He appears in American Pie Presents: The Book of Love? Let's recall: Seann William Scott as Stifler appears in original series plus the spin–offs? There's also his film within the series. Let's search.

So likely the actor is Seann William Scott, or maybe the actor is Tom Gills? No. The actor with most appearances likely is Seann William Scott or Jason Biggs? Jason Biggs appears first through first three? He appears also in reunion and maybe New movies. Let's verify counts.

We need actor with most appearances among all US movies up to 2020. Let's search for ' Seann William Scott American Pie appearances' or 'American Pie actor with most appearances'.

We also need commencement address in 2012 at a university. Which university? So Seann William Scott delivered commencement address in 2012 at some university. Need to find that.

So we need to find doc about Seann William Scott commencement address 2012. Alternatively, if actor is Jason Biggs, find if he delivered commencement address 2012. But Jason Biggs may have spoken at University of Notre Dame? Let's think: Jason Biggs Speech 2013? Wait.

Let's recall: Seann William Scott on his biography: He gave a commencement address at University of Kansas? Let's quickly search.

Given time, best to do fan_out_search queries.

First query: "Seann William Scott commencement address 2012 university".

Second query: "American Pie series actor most appearances".

Third: "American Pie spinoffs actor who appears in most films".

Fourth: ”Jason Biggs commencement address 2012”.

Fifth: ”American Pie actor commencement 2012”.

But my first size: ”Seann William Scott commencement address 2012 university” should find doc. Let’s do that.

Exact fan-out call

```
[
  {
    ”tool”: ”fan_out_search”,
    ”params”: {
      ”queries”: [
        ”Seann William Scott commencement address 2012 university”,
        ”most American Pie films actor”,
        ”American Pie actor who appeared most movies”,
        ”Seann William Scott 2012 commencement”,
        ”Jason Biggs commencement 2012”
      ]
    }
  }
]
```

Fan-out return: actor identity clue

[... clipped before ...]
cott – Wikipedia
Scott is best known for his breakout role as Steve Stifler in the American Pie film series (1999–2012). He also starred in a lead role as Doug Glatt in Goon (...

DOCUMENT ID: https://en.wikipedia.org/wiki/Thomas_Ian_Nicholas (44 tokens)
Thomas Ian Nicholas – Wikipedia
In film, Nicholas is best known for his lead roles as Kevin Meyers in the American Pie theatrical film series (1999–2012) and Henry Rowengartner in the sports ...

DOCUMENT ID: https://en.wikipedia.org/wiki/Chris_Owen_(actor) (46 tokens)
Chris Owen (actor) – Wikipedia
Chris Owen (born September 25, 1980) is an American actor. He is best known for his supporting role as Chuck ”Sherminator” Sherman in the American Pie ...

DOCUMENT ID: https://en.wikipedia.org/wiki/List_of_American_Pie_characters (50 tokens)
List of American Pie characters – Wikipedia
Noah Levenstein, played by Eugene Levy, has appeared in eight of the nine released films. Biggs, Hannigan, Scott, Nicholas, Thomas, Coolidge, Cho, Isfield, ...

DOCUMENT ID: https://en.wikipedia.org/wiki/Tad_Hilgenbrink (52 tokens)
Tad Hilgenbrink – Wikipedia
Tad Hilgenbrink (born October 9, 1981) is an American actor, known for his role as Matt Stifler in American Pie Presents: Band Camp, Chance Marquis in The ...

DOCUMENT ID: https://en.wikipedia.org/wiki/Chris_Klein_(actor) (49 tokens)
Chris Klein (actor) – Wikipedia
He reprised his role in two sequels: American Pie 2 (2001) and American Reunion (2012). Chris Klein. Klein in March 2012. Born. Frederick Christopher Klein.

DOCUMENT ID: https://en.wikipedia.org/wiki/Eddie_Kaye_Thomas (41 tokens)
Eddie Kaye Thomas – Wikipedia
Thomas rose to prominence for his supporting role as Pau

[... clipped after ...]

Curation update

Curated set updated (15/30): https://en.wikipedia.org/wiki/American_Pie_(film_series)[high], https://en.wikipedia.org/wiki/List_of_American_Pie_characters[high], https://en.wikipedia.org/wiki/Seann_William_Scott[high], https://en.wikipedia.org/wiki/Ben_Carson[fair], https://en.wikipedia.org/wiki/Bradley_Cooper[fair], https://en.wikipedia.org/wiki/Chris_Owen_(actor)[fair], https://en.wikipedia.org/wiki/Ellis_Paul[fair], https://en.wikipedia.org/wiki/Jason_Biggs[fair], https://en.wikipedia.org/wiki/Magic_Johnson[fair], https://en.wikipedia.org/wiki/Steven_Curtis_Chapman[fair], https://en.wikipedia.org/wiki/Susan_Rice[fair], https://en.wikipedia.org/wiki/Thomas_Ian_Nicholas[fair], https://en.wikipedia.org/wiki/Tim_Reid[fair], https://en.wikipedia.org/wiki/Tucker_Carlson[fair], https://en.wikipedia.org/wiki/Chris_Klein_(actor)[low]

State after turn 1

pool_size: 40
curated_size: 15/30
importance: very_high=0, high=3, fair=11, low=1
top_curated:
- https://en.wikipedia.org/wiki/American_Pie_(film_series) [high]
- https://en.wikipedia.org/wiki/List_of_American_Pie_characters [high]
- https://en.wikipedia.org/wiki/Seann_William_Scott [high]
- https://en.wikipedia.org/wiki/Ben_Carson [fair]
- https://en.wikipedia.org/wiki/Bradley_Cooper [fair]
- https://en.wikipedia.org/wiki/Chris_Owen_(actor) [fair]
- https://en.wikipedia.org/wiki/Ellis_Paul [fair]
- https://en.wikipedia.org/wiki/Jason_Biggs [fair]
- https://en.wikipedia.org/wiki/Magic_Johnson [fair]
- https://en.wikipedia.org/wiki/Steven_Curtis_Chapman [fair]
- https://en.wikipedia.org/wiki/Susan_Rice [fair]
- https://en.wikipedia.org/wiki/Thomas_Ian_Nicholas [fair]

... 3 more

Phase 2 - The policy pivots from Seann William Scott to Eugene Levy

Search Eugene Levy address


```
[
  {
    "tool": "search_corpus",
    "params": {
      "query": "Eugene Levy 2012 commencement address university"
    }
  }
]
```


Search most-appearances clue


```
[
  {
    "tool": "search_corpus",
    "params": {
      "query": "most American Pie films actor appeared in most movies american pie series"
```

```
    }
  }
]
```

Curation after candidate expansion

Curated set updated (21/30): https://en.wikipedia.org/wiki/American_Pie_(film)[high], https://en.wikipedia.org/wiki/American_Pie_(film_series)[high], https://en.wikipedia.org/wiki/Eugene_Levy[high], https://en.wikipedia.org/wiki/List_of_American_Pie_characters[high], https://en.wikipedia.org/wiki/Seann_William_Scott[high], https://en.wikipedia.org/wiki/Ben_Carson[fair], https://en.wikipedia.org/wiki/Bradley_Cooper[fair], https://en.wikipedia.org/wiki/Chris_Owen_(actor)[fair], https://en.wikipedia.org/wiki/Ellis_Paul[fair], https://en.wikipedia.org/wiki/Jason_Biggs[fair], https://en.wikipedia.org/wiki/John_Cho[fair], https://en.wikipedia.org/wiki/Magic_Johnson[fair], https://en.wikipedia.org/wiki/Shannon_Elizabeth[fair], https://en.wikipedia.org/wiki/Steven_Curtis_Chapman[fair], https://en.wikipedia.org/wiki/Susan_Rice[fair], https://en.wikipedia.org/wiki/Thomas_Ian_Nicholas[fair], https://en.wikipedia.org/wiki/Tim_Reid[fair], https://en.wikipedia.org/wiki/Tucker_Carlson[fair], https://en.wikipedia.org/wiki/Chris_Klein_(actor)[low], https://en.wikipedia.org/wiki/Dan_Petronijevic[low], https://en.wikipedia.org/wiki/Eugene_Levy_(politician)[low]

State after turn 5

pool_size: 53
curated_size: 21/30
importance: very_high=0, high=5, fair=13, low=3
top_curated:
- https://en.wikipedia.org/wiki/American_Pie_(film) [high]
- https://en.wikipedia.org/wiki/American_Pie_(film_series) [high]
- https://en.wikipedia.org/wiki/Eugene_Levy [high]
- https://en.wikipedia.org/wiki/List_of_American_Pie_characters [high]
- https://en.wikipedia.org/wiki/Seann_William_Scott [high]
- https://en.wikipedia.org/wiki/Ben_Carson [fair]
- https://en.wikipedia.org/wiki/Bradley_Cooper [fair]
- https://en.wikipedia.org/wiki/Chris_Owen_(actor) [fair]
- https://en.wikipedia.org/wiki/Ellis_Paul [fair]
- https://en.wikipedia.org/wiki/Jason_Biggs [fair]
- https://en.wikipedia.org/wiki/John_Cho [fair]
- https://en.wikipedia.org/wiki/Magic_Johnson [fair]

... 9 more

Phase 3 - Exact search tests the candidate bridge

Exact grep call

```
[
  {
    "tool": "grep_corpus",
    "params": {
      "pattern": "Seann William Scott commencement|Eugene Levy commencement|Seann William Scott address|Eugene Levy address|2012 commencement"
    }
  }
]
```

Grep return excerpt

DOCUMENT ID: https://en.wikipedia.org/wiki/Donald_Sutherland (44 tokens) Donald Sutherland – Wikipedia Donald McNichol Sutherland (17 July 1935 – 20 June 2024) was a Canadian actor. With a career spanning six decades, he received numerous accolades, ...

DOCUMENT ID: https://en.wikipedia.org/wiki/John_Lithgow (36 tokens) John Lithgow – Wikipedia John Arthur Lithgow is an American actor. He studied at Harvard University and the London Academy of Music and Dramatic Art before becoming known for his ...

Direct bridge search


```
[
  {
    "tool": "search_corpus",
    "params": {
      "query": "Eugene Levy 2012 commencement address university biographical speaker 2012"
    }
  }
]
```


Search return before full read

DOCUMENT ID: https://en.wikipedia.org/wiki/Princeton_University (38 tokens) Princeton University – Wikipedia Princeton University is a private Ivy League research university in Princeton, New Jersey, United States. Founded in 1746 in Elizabeth as the College of New ... # DOCUMENT ID: https://en.wikipedia.org/wiki/List_of_University_of_Michigan_alumni (41 tokens) List of University of Michigan alumni – Wikipedia The following is a list of University of Michigan alumni. There are more than 640,000 living alumni of the University of Michigan in 180 countries across ... # DOCUMENT ID: https://en.wikipedia.org/wiki/List_of_Cornell_University_alumni (31 tokens) List of Cornell University alumni – Wikipedia This list of Cornell University alumni includes notable graduates, non–graduate former students, and current students of Cornell University. # DOCUMENT ID: https://en.wikipedia.org/wiki/University_of_St._Thomas_(Minnesota) (37 tokens) University of St. Thomas (Minnesota) – Wikipedia The University of St. Thomas is a private Catholic research university with campuses in Saint Paul and Minneapolis, Minnesota, United States. # DOCUMENT ID: https://en.wikipedia.org/wiki/List_of_Queens_College_people (43 tokens) List of Queens College people – Wikipedia This is a list of notable alumni and faculty of Queens College, City University of New York. Contents. 1 Business; 2 Education; 3 ...

State after turn 13

pool_size: 74
curated_size: 30/30
importance: very_high=0, high=5, fair=14, low=11
top_curated:
- https://en.wikipedia.org/wiki/American_Pie_(film) [high]
- https://en.wikipedia.org/wiki/American_Pie_(film_series) [high]
- https://en.wikipedia.org/wiki/Eugene_Levy [high]
- https://en.wikipedia.org/wiki/List_of_American_Pie_characters [high]
- https://en.wikipedia.org/wiki/Seann_William_Scott [high]
- https://en.wikipedia.org/wiki/Ben_Carson [fair]
- https://en.wikipedia.org/wiki/Bradley_Cooper [fair]
- https://en.wikipedia.org/wiki/Chris_Owen_(actor) [fair]
- https://en.wikipedia.org/wiki/Ellis_Paul [fair]
- https://en.wikipedia.org/wiki/Jason_Biggs [fair]
- https://en.wikipedia.org/wiki/John_Cho [fair]

– https://en.wikipedia.org/wiki/Magic_Johnson [fair]
... 18 more

PHASE 4 - FULL-DOCUMENT READING EXPOSES THE ANSWER SENTENCE

Exact read call


```
[
  {
    "tool": "read_document",
    "params": {
      "doc_id": "https://en.wikipedia.org/wiki/Eugene_Levy"
    }
  }
]
```


Full-document excerpt around answer

[... clipped before ...]
me honourees, alongside fellow SCTV alumni John Candy, Martin Short, and Catherine O' Hara. Levy is one of only a handful of people who have won at least five Canadian Comedy Awards, including two for Best Writing (Best in Show in 2001 and A Mighty Wind in 2004) and three for Best Male Performer (Best in Show, American Pie 2 in 2002, and A Mighty Wind). In 2008, the governor general of Canada presented Levy with the Governor General's Performing Arts Awards (GGPAA), a lifetime achievement award considered "for their outstanding body of work and enduring contribution to the performing arts in Canada.". In 2010, Levy was awarded the ACTRA Award by the union representing Canada's actors. In 2011, Levy was made a Member of the Order of Canada "for his contributions as a comic actor and writer, and for his dedication to charitable causes" and promoted to the rank of Companion in 2022.
On May 22, 2012, Levy delivered a commencement address at Dalhousie University, in Halifax, Nova Scotia, and was awarded the degree Doctor of Laws (honoris causa). On June 11, 2012, Levy was presented with the Queen Elizabeth II Diamond Jubilee Medal by the lieutenant governor of Ontario.
On March 13, 2016, Levy took home the award for "Best Performance by an Actor in a Continuing Leading Comedic Role" at the 4th Annual Canadian Screen Awards, for his performance as Johnny Rose in the CBC/Pop TV sitcom Schitt's Creek.

References
External links

Eugene Levy at IMDb
Eugene Levy at the Internet Broadway Database
Eugene Levy discography at Discogs
Article at thecanadianencyclopedia.ca
Eugene Levy on X
[Context: 9839/30720]

Final curation update

Curated set updated (30/30): https://en.wikipedia.org/wiki/Eugene_Levy[very_high], https://en.wikipedia.org/wiki/American_Pie_(film)[high], https://en.wikipedia.org/wiki/American_Pie_(film_series)[high], https://en.wikipedia.org/wiki/Dalhousie_University[high], https://en.wikipedia.org/wiki/List_of_American_Pie_characters[high], https://en.wikipedia.org/wiki/Seann_William_Scott[high], https://en.wikipedia.org/wiki/Ben_Carson[fair], https://en.wikipedia.org/wiki/Bradley_Cooper[fair], https://en.wikipedia.org/wiki/Chris_Owen_(actor)[fair], https://en.wikipedia.org/wiki/Ellis_Paul[fair], https://en.wikipedia.org/wiki/Jason_Biggs[fair], https://en.

wikipedia.org/wiki/John_Cho[fair], https://en.wikipedia.org/wiki/Magic_Johnson[fair], https://en.wikipedia.org/wiki/Princeton_University[fair], https://en.wikipedia.org/wiki/Shannon_Elizabeth[fair], https://en.wikipedia.org/wiki/Steven_Curtis_Chapman[fair], https://en.wikipedia.org/wiki/Susan_Rice[fair], https://en.wikipedia.org/wiki/Thomas_Ian_Nicholas[fair], https://en.wikipedia.org/wiki/Tim_Reid[fair], https://en.wikipedia.org/wiki/Tucker_Carlson[fair], https://en.wikipedia.org/wiki/Bradley_Whitford[low], https://en.wikipedia.org/wiki/Dan_Petronijevic[low], https://en.wikipedia.org/wiki/David_Sutherland[low], https://en.wikipedia.org/wiki/Donald_Sutherland[low], https://en.wikipedia.org/wiki/Heath_Ledger[low], https://en.wikipedia.org/wiki/John_Lithgow[low], https://en.wikipedia.org/wiki/Jonathan_Franzen[low], https://en.wikipedia.org/wiki/Joss_Whedon[low], https://en.wikipedia.org/wiki/List_of_NYU_Tisch_School_of_the_Arts_people[low], https://en.wikipedia.org/wiki/Wes_Craven[low]
[EVICTED low–importance] 1 doc(s): https://en.wikipedia.org/wiki/Eugene_Levy_(politician)

State after turn 15

pool_size: 74
curated_size: 30/30
importance: very_high=1, high=5, fair=14, low=10
top_curated:
- https://en.wikipedia.org/wiki/Eugene_Levy [very_high]
- https://en.wikipedia.org/wiki/American_Pie_(film) [high]
- https://en.wikipedia.org/wiki/American_Pie_(film_series) [high]
- https://en.wikipedia.org/wiki/Dalhousie_University [high]
- https://en.wikipedia.org/wiki/List_of_American_Pie_characters [high]
- https://en.wikipedia.org/wiki/Seann_William_Scott [high]
- https://en.wikipedia.org/wiki/Ben_Carson [fair]
- https://en.wikipedia.org/wiki/Bradley_Cooper [fair]
- https://en.wikipedia.org/wiki/Chris_Owen_(actor) [fair]
- https://en.wikipedia.org/wiki/Ellis_Paul [fair]
- https://en.wikipedia.org/wiki/Jason_Biggs [fair]
- https://en.wikipedia.org/wiki/John_Cho [fair]

... 18 more

Q.4 Case Study D: Held-Out Multi-Hop QA via Search-State Editing

HotpotQA query

Query. Which American comedian starred in the original cast of Saturday Night Live and played Stan Winters in the sitcom Martin?
Gold answer. Garrett Morris
Outcome. 11 turns, tool diversity 5, curated recall 1.0, final-answer recall 1.0.

Why this case matters. We chose a small held-out HotpotQA example so that the transfer claim can be seen in one trajectory. The policy solves a multi-hop web query outside the SFT and RL families with the same operations it learned elsewhere: search, curation, full-document reading, exact grep, and final evidence promotion.

Trajectory overview.

Turn	Tool	Policy move	Harness state / why it matters
0	`search`	Search both clues jointly	initial hit identifies Garrett Morris through the Stan Winters clue
1	`curate`	Promote candidate pages	auto-seeded web hits become an editable curated set
2	`search`	Search the SNL-original-cast clue	second hop confirms the other half of the query

Turn	Tool	Policy move	Harness state / why it matters
3	`curate`	Promote SNL cast evidence	curated state now contains both supporting paths
4	`read`	Read Garrett Morris page	full text unifies comedian, original SNL cast, and Martin role
5	`curate`	Keep Garrett Morris high	final answer evidence is retained
6	`grep`	Grep exact role phrase	exact matching checks the Stan Winters clue
7	`curate`	Add sitcom/role pages	curated set adds corroborating context

PHASE 1 - JOINT SEARCH IDENTIFIES THE LIKELY ENTITY

Exact search call

```
[
  {
    "tool": "search_corpus",
    "params": {
      "query": "Stan Winters Martin SNL original cast comedian"
    }
  }
]
```

Search return: Stan Winters points to Garrett Morris

DOCUMENT ID: https://en.wikipedia.org/wiki/Garrett_Morris (46 tokens) Garrett Morris – Wikipedia He is also known for his role in the sitcom Martin as Stan Winters, from 1992 to 1995, until he suffered an injury. Also, he made two guest appearances on The ...

Curation update

Curated set updated (8/30): https://en.wikipedia.org/wiki/Garrett_Morris[high], https://en.wikipedia.org/wiki/Andrea_Martin[fair], https://en.wikipedia.org/wiki/Bobby_Moynihan[fair], https://en.wikipedia.org/wiki/Don_Novello[fair], https://en.wikipedia.org/wiki/Richard_Lewis_(comedian)[fair], https://en.wikipedia.org/wiki/Second_City_Television[fair], https://en.wikipedia.org/wiki/Victoria_Jackson[fair], https://en.wikipedia.org/wiki/Will_Forte[fair]

State after turn 1

pool_size: 10
curated_size: 8/30
importance: very_high=0, high=1, fair=7, low=0
top_curated:
- https://en.wikipedia.org/wiki/Garrett_Morris [high]
- https://en.wikipedia.org/wiki/Andrea_Martin [fair]
- https://en.wikipedia.org/wiki/Bobby_Moynihan [fair]
- https://en.wikipedia.org/wiki/Don_Novello [fair]
- https://en.wikipedia.org/wiki/Richard_Lewis_(comedian) [fair]
- https://en.wikipedia.org/wiki/Second_City_Television [fair]
- https://en.wikipedia.org/wiki/Victoria_Jackson [fair]
- https://en.wikipedia.org/wiki/Will_Forte [fair]

PHASE 2 - SECOND-HOP SEARCH CHECKS ORIGINAL SNL CAST MEMBERSHIP

Exact search call

```
[
  {
    "tool": "search_corpus",
    "params": {
      "query": "Garrett Morris Saturday Night Live original cast"
    }
  }
]
```

Search return: original cast evidence

DOCUMENT ID: https://en.wikipedia.org/wiki/Saturday_Night_Live_15th_Anniversary_Special (46 tokens)
Saturday Night Live 15th Anniversary Special – Wikipedia
The special was SNL's first anniversary celebration; it featured original cast members Chevy Chase, Dan Aykroyd, Jane Curtin, Garrett Morris, and Laraine ...

DOCUMENT ID: https://en.wikipedia.org/wiki/List_of_Saturday_Night_Live_cast_members (54 tokens)
List of Saturday Night Live cast members – Wikipedia
Original cast (from left to right): Laraine Newman, John Belushi, Jane Curtin, Gilda Radner, Dan Aykroyd, Garrett Morris, and Chevy Chase. Contents. 1 List.

DOCUMENT ID: https://en.wikipedia.org/wiki/Saturday_Night_Live_season_1 (56 tokens)
Saturday Night Live season 1 – Wikipedia
Starring. Dan Aykroyd * John Belushi * Chevy Chase * George Coe * Jane Curtin * Garrett Morris * Laraine Newman * Michael O'Donoghue * Gilda Radner ; No. of ...

DOCUMENT ID: https://en.wikipedia.org/wiki/Saturday_Night_Live_season_3 (40 tokens)
Saturday Night Live season 3 – Wikipedia
Cast * Dan Aykroyd * John Belus
[... clipped after ...]

Curation update

Curated set updated (12/30): https://en.wikipedia.org/wiki/Garrett_Morris[high], https://en.wikipedia.org/wiki/List_of_Saturday_Night_Live_cast_members[high], https://en.wikipedia.org/wiki/Saturday_Night_Live_15th_Anniversary_Special[high], https://en.wikipedia.org/wiki/Andrea_Martin[fair], https://en.wikipedia.org/wiki/Bobby_Moynihan[fair], https://en.wikipedia.org/wiki/Don_Novello[fair], https://en.wikipedia.org/wiki/Richard_Lewis_(comedian)[fair], https://en.wikipedia.org/wiki/Saturday_Night_Live_season_1[fair], https://en.wikipedia.org/wiki/Saturday_Night_Live_season_3[fair], https://en.wikipedia.org/wiki/Second_City_Television[fair], https://en.wikipedia.org/wiki/Victoria_Jackson[fair], https://en.wikipedia.org/wiki/Will_Forte[fair]

State after turn 3

pool_size: 19
curated_size: 12/30
importance: very_high=0, high=3, fair=9, low=0
top_curated:
- https://en.wikipedia.org/wiki/Garrett_Morris [high]
- https://en.wikipedia.org/wiki/List_of_Saturday_Night_Live_cast_members [high]
- https://en.wikipedia.org/wiki/Saturday_Night_Live_15th_Anniversary_Special [high]
- https://en.wikipedia.org/wiki/Andrea_Martin [fair]

– https://en.wikipedia.org/wiki/Bobby_Moynihan [fair]
– https://en.wikipedia.org/wiki/Don_Novello [fair]
– https://en.wikipedia.org/wiki/Richard_Lewis_(comedian) [fair]
– https://en.wikipedia.org/wiki/Saturday_Night_Live_season_1 [fair]
– https://en.wikipedia.org/wiki/Saturday_Night_Live_season_3 [fair]
– https://en.wikipedia.org/wiki/Second_City_Television [fair]
– https://en.wikipedia.org/wiki/Victoria_Jackson [fair]
– https://en.wikipedia.org/wiki/Will_Forte [fair]

PHASE 3 - FULL-DOCUMENT READING UNIFIES BOTH CLUES

Exact read call

```
[
  {
    "tool": "read_document",
    "params": {
      "doc_id": "https://en.wikipedia.org/wiki/Garrett_Morris"
    }
  }
]
```

Full-document excerpt around answer

```
# Document (2665 tokens)
# Garrett Morris

## Summary
Garrett Isaac Morris (born February 1, 1937) is an American actor, comedian and singer. He
was part of the original cast and was the first Black cast member of the sketch comedy
program Saturday Night Live, appearing from 1975 to 1980.
He also played Jimmy on The Jeffersons (1983–1984). Morris had one of the starring roles,
as Junior "Uncle Junior" King, on the sitcom The Jamie Foxx Show, which aired from 1996 to
2001. Morris also had a regular role as Earl Washington on the CBS sitcom 2 Broke Girls,
from 2011 to 2017.
He is also known for his role in the sitcom Martin as Stan Winters, from 1992 to 1995, until he
suffered an injury. Also, he made two guest appearances on The Wayans Bros. in season one
, episode one as himself and again on episode ten as the brothers' uncle Leon (1995). He
played a concerned teacher in the film Cooley High (1975), Slide in Car Wash (1976), and
Carl in The Census Taker (1984).
In 2024, Morris was honored with his own star on the Hollywood Walk of Fame.

## Full Text
Garrett Isaac Morris (born February 1, 1937)
[... clipped after ...]
```

Exact grep call

```
[
  {
    "tool": "grep_corpus",
    "params": {
      "pattern": "Stan Winters"
    }
  }
]
```

Grep return

DOCUMENT ID: https://en.wikipedia.org/wiki/Garrett_Morris (45 tokens) Garrett Morris – Wikipedia He is also known for his role in the sitcom Martin as Stan Winters, from 1992 to 1995, until he suffered an injury. Also, he made two guest appearances on ...

DOCUMENT ID: https://en.wikipedia.org/wiki/Martin_(TV_series) (55 tokens) Martin (TV series) – Wikipedia Garrett Morris as Stan Winters (seasons 1–2; guest season 3). Supporting characters. edit ⋆ Reginald Ballard as Bruh–Man (Brother Man) (seasons 2–4); Tracy ...

State after turn 7

pool_size: 23
curated_size: 14/30
importance: very_high=0, high=3, fair=10, low=1
top_curated:
- https://en.wikipedia.org/wiki/Garrett_Morris [high]
- https://en.wikipedia.org/wiki/List_of_Saturday_Night_Live_cast_members [high]
- https://en.wikipedia.org/wiki/Saturday_Night_Live_15th_Anniversary_Special [high]
- https://en.wikipedia.org/wiki/Andrea_Martin [fair]
- https://en.wikipedia.org/wiki/Bobby_Moynihan [fair]
- https://en.wikipedia.org/wiki/Don_Novello [fair]
- https://en.wikipedia.org/wiki/Martin_(TV_series) [fair]
- https://en.wikipedia.org/wiki/Richard_Lewis_(comedian) [fair]
- https://en.wikipedia.org/wiki/Saturday_Night_Live_season_1 [fair]
- https://en.wikipedia.org/wiki/Saturday_Night_Live_season_3 [fair]
- https://en.wikipedia.org/wiki/Second_City_Television [fair]
- https://en.wikipedia.org/wiki/Victoria_Jackson [fair]

... 2 more

Q.5 Case Study E: Backtracking after Rejecting the Right Entity

HotpotQA query

Query. When was the game, in which Drew Gehling voices Gord, released for PlayStation 2?
Gold answer. October 17, 2006
Outcome. 40 turns, tool diversity 5, curated recall 1.0, final-answer recall 1.0.

Why this case is interesting. The policy first rejects the correct entity, carries that mistake through several turns, and then reverses it after reading a stronger source. The first search surfaces *Bully*, but the policy calls that hit likely wrong and pursues a separate "Gord" interpretation. Several turns later, `read_document` on Drew Gehling's page proves that the discarded game was correct. The recovery does not happen between two adjacent search turns. It is a delayed correction of the state, after which the policy removes stale foreign-language variants and runs a new search for the PlayStation 2 release date.

Trajectory overview.

Turn	Tool	Policy move	Harness state / why it matters
0	`search`	Search Drew Gehling, Gord, and PS2 together	auto-seed includes *Bully* pages, including the English game page
2	`search`	Mistakenly reject *Bully*	the policy says the correct game is "likely wrong" and starts a separate Gord-character path

Turn	Tool	Policy move	Harness state / why it matters
3	`curate`	Keep Drew Gehling, demote noisy pages	the mistaken hypothesis remains visible because *Bully* is still retained at low importance
4–5	`search`	Pursue the wrong Gord interpretation	Duck Tales and generic Gord-game searches do not resolve the query
7	`read`	Read Drew Gehling's page	full text contradicts the earlier rejection: Gehling is the voice of Gord in *Bully*
8	`curate`	Backtrack by editing memory	foreign-language variants are removed while the English *Bully* page is retained
9	`search`	Search the corrected title plus PS2 release date	the search now targets the right game instead of the wrong Gord branch
10	`curate`	Promote release-date evidence	the simple *Bully* page and *Canis Canem Edit* page become high-importance evidence
11	`verify`	Check the release-date claim	verification is attempted over the curated candidate docs
13–14	`grep`	Exact-date grep is tried and rejected	date-only matching returns unrelated October 17 pages, so the agent does not treat the date string alone as evidence

PHASE 1 - THE CORRECT GAME APPEARS FIRST, BUT THE POLICY REJECTS IT

Initial reasoning excerpt

The query asks for the PlayStation 2 release date of the game in which Drew Gehling voices Gord. The first decomposition is sensible: search the actor, the role name, and the platform together.

Exact tool call

```
[
  {
    "tool": "search_corpus",
    "params": {
      "query": "Drew Gehling Gord PS2 game"
    }
  }
]
```

Search return: the right entity is present immediately

DOCUMENT ID: https://en.wikipedia.org/wiki/Bully_(video_game)
Bully (video game) – Wikipedia
Bully is a 2006 action–adventure video game developed by Rockstar Vancouver and published by Rockstar Games ...

DOCUMENT ID: https://es.wikipedia.org/wiki/Bully_(videojuego)
Bully es un videojuego ... desarrollado por Rockstar Vancouver para la consola PlayStation 2.

[AUTO–POPULATED] Top 3 docs from this search have been added to your curated set at 'fair' importance.

The policy wrongly rules out the right hit

The search for Drew Gehling Gord PS2 returned Bully, which is likely wrong given that Bully is not voiced by Gehling. My current search angle is exhausted. Let me rethink. A missed facet: ”Gord” is probably the character from the ”Duck Tales” game series voiced by Gehling.

Memory after the mistaken rejection

Curated set updated (4/30):
https://en.wikipedia.org/wiki/Drew_Gehling[high],
https://en.wikipedia.org/wiki/Bully_(video_game)[low],
https://es.wikipedia.org/wiki/Bully_(videojuego)[low],
https://pt.wikipedia.org/wiki/Bully_(jogo_eletr%C3%B4nico)[low]

Phase 2 - A delayed read overturns the wrong branch

The wrong branch persists for several turns

Turn 4 search:
Duck Tales Gord voiced by Drew Gehling
–> no results

Turn 5 search:
Gord PlayStation 2 game
–> generic ”Gord” pages, planned PS2 titles, Ford/Gordon distractors

Turn 7 reasoning:
Current search angle on Gord PlayStation 2 is stuck and likely wrong ... Another angle: check the Drew Gehling page for credited games and character list for ”Gord.”

Backtracking read

```
[
  {
    ”tool”: ”read_document”,
    ”params”: {
      ”doc_id”: ”https://en.wikipedia.org/wiki/Drew_Gehling”
    }
  }
]
```

Drew Gehling
Andrew Shearer Gehling ... is an American stage and screen actor ... and as the voice of Gord in the 2006 video game Bully from Rockstar Games.

State edit: remove stale variants, keep the corrected entity

```
[
  {
    ”tool”: ”curate”,
    ”params”: {
      ”add_ids”: [
        ”https://en.wikipedia.org/wiki/Drew_Gehling”
      ],
      ”remove_ids”: [
```

```
        "https://es.wikipedia.org/wiki/Bully_(videojuego)",
        "https://pt.wikipedia.org/wiki/Bully_(jogo_eletr%C3%B4nico)"
      ],
      "importance": {
        "https://en.wikipedia.org/wiki/Drew_Gehling": "high"
      }
    }
  }
]
```


Curated set updated (3/30):
https://en.wikipedia.org/wiki/Drew_Gehling[high],
https://en.wikipedia.org/wiki/Bully_(video_game)[low],
https://en.wikipedia.org/wiki/Gord_(video_game)[low]

PHASE 3 - THE CORRECTED ENTITY DRIVES THE ANSWER SEARCH

Recovered strategy

The identity facet is solved: Gord is from \emph{Bully}. The policy now stops searching generic Gord pages and searches the corrected title with the platform/date facet.

Corrected search


```
[
  {
    "tool": "search_corpus",
    "params": {
      "query": "Bully PlayStation 2 release date"
    }
  }
]
```


DOCUMENT ID: https://simple.wikipedia.org/wiki/Bully_(video_game)
Bully ... was released on 17 October 2006 ...

DOCUMENT ID: https://en.wikipedia.org/wiki/
List_of_video_games_published_by_Rockstar_Games
Bully – PlayStation 2, October 17, 2006.

DOCUMENT ID: https://en.wikipedia.org/wiki/List_of_best–
selling_PlayStation_2_video_games
Bully, 1.5 million, October 17, 2006, Action–adventure ...

Promoting the answer evidence


```
[
  {
    "tool": "curate",
    "params": {
      "add_ids": [
        "https://en.wikipedia.org/wiki/Simple_Wikipedia/Bully_(video_game)",
        "https://en.wikipedia.org/wiki/Canis_Canem_Edit"
      ],
      "importance": {
        "https://en.wikipedia.org/wiki/Simple_Wikipedia/Bully_(video_game)": "high",
        "https://en.wikipedia.org/wiki/Canis_Canem_Edit": "high"
```

```
      }
    }
  }
]
```

Curated set updated (5/30):
https://en.wikipedia.org/wiki/Canis_Canem_Edit[high],
https://en.wikipedia.org/wiki/Drew_Gehling[high],
https://en.wikipedia.org/wiki/Simple_Wikipedia/Bully_(video_game)[high],
https://en.wikipedia.org/wiki/Bully_(video_game)[low],
https://en.wikipedia.org/wiki/Gord_(video_game)[low]

A final exact-date check is not enough by itself

Grep on ”October 17, 2006|2006–10–17” returns date–only false positives such as current–events pages, films released on the same day, and unrelated titles. The agent therefore keeps the title–and–platform evidence as the answer path rather than treating the date string alone as proof.

Why the state edit matters. What matters here is the delayed correction. The policy's early hypothesis calls *Bully* probably wrong, but the harness still keeps it at low importance. When a later read contradicts the hypothesis, the policy can edit the curated set to keep the corrected entity and discard stale variants, and it can then redirect search toward the release-date facet. The backtracking is thus visible as a change to the external state. It is not confined to private reasoning between two adjacent searches.