Towards Long-Horizon Agents: A Survey

Foundation, Evolution, Harness, Optimization, Application, and Frontier

Guanting Dong[1,*], **Xiaoshuai Song**[1,*], **Yuyang Hu**[1,*], **Jiajie Jin**[1,*], **Chenghao Zhang**[1,*], **Yifei Chen**[1], **Xiaoxi Li**[1], **Huaying Yuan**[1], **Xinyu Yang**[1], **Tongyu Wen**[1], **Jiejun Tan**[1], **Hongjin Qian**[2], **Shijue Huang**[5], **Junting Lu**[2], **Zhenyu Li**[3], **Wanjun Zhong**[4], **Yutao Zhu**[1], **Tat-Seng Chua**[6], **Zhicheng Dou**[1,†], **Ji-Rong Wen**[1]

∗ Core Contributors † Corresponding author

[1]**Renmin University of China**, [2]**Peking University**, [3]**Tsinghua University**, [4]**Sun Yat-Sen University**, [5]**Hong Kong University of Science and Technology**, [6]**National University of Singapore**

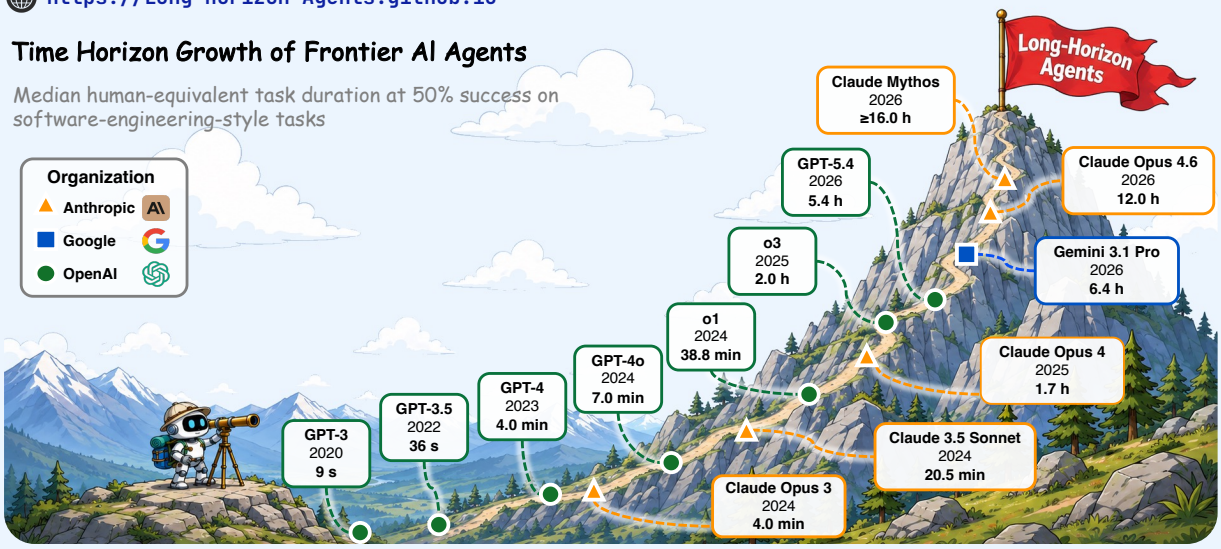
With the rapid advancement of LLM capabilities, expectations for AI agents are shifting from solving simple, single-turn tasks toward carrying out long-horizon tasks in the real world. We refer to such systems as **long-horizon agents**: agents that plan over extended horizons, interact with real-world environments, recover from their own mistakes, and adapt their strategies during execution. This capability is rapidly becoming the central bottleneck for practical agent intelligence. Despite this progress, the field still lacks a shared definition and taxonomy for long-horizon agents. Related concepts such as "self-evolving" or "autonomous" agents are often used interchangeably, none of which clearly captures what it means to build a *long-horizon agent*. This gap leaves the community without a principled way to attribute advances in long-horizon competence.

This paper offers a unified landscape by framing long-horizon agency as the co-evolution of an **externalized harness** and an **internalized optimization**, and organizes the paper around six connected perspectives: **Foundation**, **Evolution**, **Harness**, **Optimization**, **Application**, and **Frontier**. We first formalize long-horizon agency as a harness-coupled decision process and distinguish it from neighboring concepts such as long-running execution, autonomy, and self-evolution. We then trace the field's evolution from prompt-level control to runtime agent systems, classifying existing work through the complementary lenses of externalized harnesses and internalized optimization. Building on this view, we organize five application forms of long-horizon agents by their interfaces, and review the corresponding benchmarks and resources. Finally, we discuss key challenges and frontier directions. Looking ahead, we hope this paper serves not only as a reference for existing work, but also as a foundation for building the next generation of capable, reliable long-horizon agents.

dongguanting@ruc.edu.cn, dou@ruc.edu.cn

https://github.com/RUC-NLPIR/Awesome-Long-Horizon-Agents

https://Long-Horizon-Agents.github.io



*The empirical time-horizon trend of frontier agents: the task-completion horizon has climbed from seconds to over twelve hours, doubling roughly every seven months (https://metr.org/time-horizons).

Contents

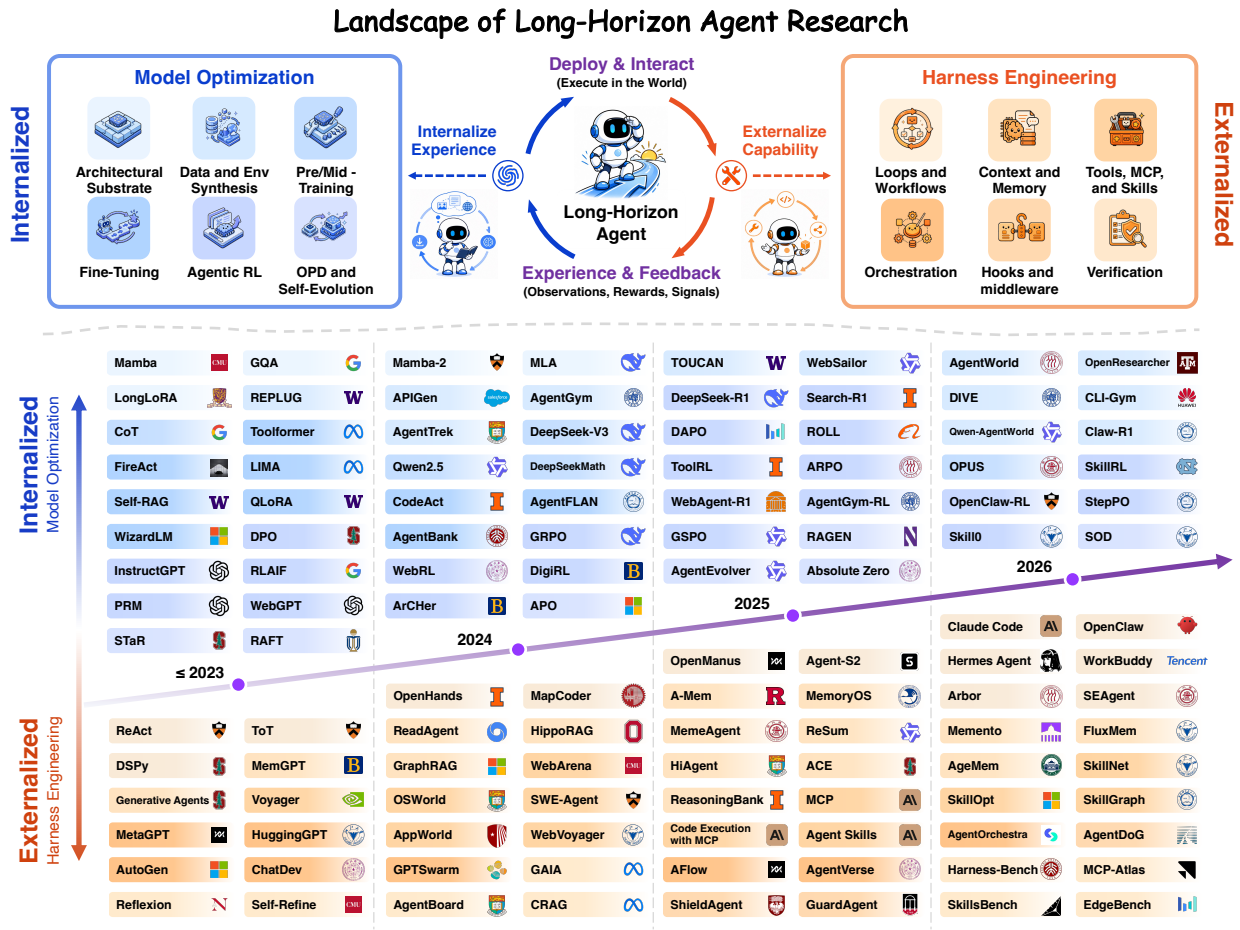


Figure 1 The landscape of long-horizon agent research. This paper adopts a co-evolutionary view of long-horizon agents, organized around *externalized harness engineering* (bottom) and *internalized model optimization* (top).

1 Introduction

Over the past few years, large language models (LLMs) have moved from single-turn chatbots to the decision-making core of autonomous agents in software engineering [260, 747], general-purpose assistance [431], scientific discovery [678], computer use [716, 885], and multimodal interaction [141]. These domains differ on the surface, but they share one decisive requirement that we call *long horizon*: they require persistent iteration across reasoning, tool use, observation, and revision over many interdependent steps—from those completable within a single context window to those that span windows, sessions, or open-ended task streams. To meet this requirement, *long-horizon agents* have emerged, expanding the boundaries of agent applications through continual, sustained interaction with real-world environments.

Long-horizon competence is now advancing rapidly and increasingly marks the practical line for delegating productive work to agents. For instance, frontier coding agents are already able to run for hours on a single project, occasionally close to a full day [14, 456]. Moreover, METR [286] reports that the time horizon of advanced agentic task execution has doubled roughly every seven months over the past several years, accelerating to about every four months for the most recent models. For example, GLM-5.1 is reported to sustain independent execution on a single task for up to eight hours, covering planning, implementation, testing, iterative refinement, and final delivery [179]. Notably, this capability cannot be obtained *simply* by optimizing and scaling the model's internal parameters or enlarging its context window alone; its capability boundary also varies sharply with the *external* harness surrounding the model [123, 378, 468, 511]. Therefore, the central thesis of this paper is that *long-horizon agency is a system-level capability jointly shaped by two*

co-evolving components: externalized harness engineering and internalized model optimization.

Why long-horizon task execution is hard for agents. The difficulty of long-horizon task execution does not reduce to per-step accuracy: it arises from the *composition* of many tightly coupled steps into a single trajectory, rather than from any individual decision. Across the systems we organize, three representative failure modes recur throughout long-horizon execution and structure much of the technical discussion.

- **Goal drift and compounding error.** Over thousands of steps an agent can slowly diverge from its original objective, and because no step is perfectly reliable, local errors accumulate along the trajectory and eventually steer it onto a wrong reasoning path [20, 549].
- **Context rot and context-limit pressure.** As the working context fills, an agent tends to degrade abruptly once it passes a certain context-utilization threshold [378, 511]; conversely, an agent that perceives its context limit may terminate prematurely and declare partial progress a success [14].
- **Sparse, delayed rewards and irreversibility.** Many long-horizon tasks return a reward only at the end, leaving the agent with an extremely sparse learning signal over its intermediate decisions [808]; moreover, the longer an agent acts, the more likely it is to take a risky, irreversible action, so monitoring its execution safety and recoverability becomes essential.

Why long-horizon agents need a new taxonomy. Recent works have begun to clarify parts of this landscape, but its taxonomies still exhibit two structural limitations that motivate a new one.

- **Limited and unsystematic coverage.** Existing papers typically examine a *single* technical component of long-horizon agents in isolation, such as memory [694], context engineering [425, 841], agent harnesses [468, 876], self-evolution [165, 709], or agentic reinforcement learning [808]. Each is a genuine piece of the long-horizon agent, but none offers a comprehensive account of what such an agent *is*, nor of how to build an efficient and trustworthy long-horizon agent that stays reliable across a full trajectory.
- **Unclear definitions and overlapping terminology.** As interest in capable agents grows, the concept itself has become broad and blurred. Related notions such as long-context, long-running, autonomous, and self-evolving agents are routinely used interchangeably, even though each emphasizes a different facet, and reported gains entangle changes in the model, the harness, the data pipeline, and the evaluation protocol. Most works cannot say precisely what counts as long-horizon agency or how these techniques relate, which leaves improvements hard to attribute and hard to transfer.

These findings highlight the need for a unified taxonomy for long-horizon agent research.

We therefore organize the long-horizon agent field in this paper through a *co-evolutionary* view (Figure 1). Our organizing principle is that long-horizon agency is realized through the co-evolution of **externalized harness engineering** and **internalized model optimization**, two complementary halves that together constitute a long-horizon agent. Concretely, **externalized harness engineering** supplies runtime structure such as loops and workflows, context and memory, tools, orchestration, hooks and middleware, and verification; **internalized model optimization** improves the agent policy through architecture, data, environments, fine-tuning, reinforcement learning, and self-evolution. Neither half is sufficient on its own; long-horizon agency emerges only through their co-evolution, as harness mechanisms are gradually internalized into the policy and a stronger policy in turn unlocks more advanced harnesses.

Building on this co-evolutionary view, this paper aims to reconcile the fragmented definitions, connect the emerging trends, and articulate the foundational principles of long-horizon agents. Concretely, we organize the discussion around the following key questions:

Key Questions

- **Q1. Foundation.** How should long-horizon agency be formalized, and how does it differ from long-running execution, autonomy, and self-evolution?
- **Q2. Evolution.** How did the control surface widen from prompts, to context, to runtime agent systems?
- **Q3. Harness.** How does externalized harness engineering keep long-horizon agent execution specifiable, steerable, verifiable, and recoverable at runtime?
- **Q4. Optimization.** How can internalized model optimization build long-horizon competence across the agent's architectural substrate, data and environment synthesis, and full training lifecycle?
- **Q5. Application.** What forms do long-horizon agents take across domains, and how should their long-horizon capability be systematically evaluated?
- **Q6. Frontier.** What open problems define the next stage as horizons grow?

To address these questions, the paper follows this co-evolutionary decomposition through six connected perspectives. **Foundation** formalizes the agent and separates it from adjacent concepts, while **Evolution** traces how the locus of control widened from prompts to context to runtime systems. The two halves of the argument are then treated in turn: **Harness** studies how long-horizon capability is *externalized* into a runtime layer, and **Optimization** studies how it is *internalized* into the policy. Finally, **Application** shows how the same long-horizon pressures recur across domains and how such systems are evaluated, and **Frontier** identifies what remains open. Crucially, these two halves are not independent: capability continually migrates between them, so throughout we treat externalized harness engineering and internalized model optimization as coupled rather than as separate tracks.

Contributions and Roadmap. This paper makes the following contributions, each answering one of the key questions and mapping to one part of the paper. Figure 2 presents an overview of our taxonomy.

1. **Foundation (Section 2).** We formalize long-horizon agency as a coupled model–harness decision process and organize its difficulty into three nested task levels (H1–H3) with matching capabilities (C1–C3): intra-context reasoning, cross-context memory, and cross-task experience. Building on METR's data, we note that the frontier time horizon is scaling with an accelerating doubling time. We further separate long-horizon agency from long-running execution, autonomy, and self-evolution, giving the field a shared vocabulary for attributing where long-horizon competence comes from.

2. **Evolution (Section 3).** We trace how the control surface migrated from prompts, to context, to runtime systems, framing today's long-horizon agents as the current endpoint of a capability-driven trajectory and clarifying why control first moved outward before it could be internalized.

3. **Harness (Section 4).** We organize externalized harness engineering by component: loops and workflows, context and memory, tools, MCP, and skills, orchestration and multi-agent control, hooks and middleware, and verification. Together these components show how a runtime layer keeps long execution specifiable, steerable, verifiable, and recoverable.

4. **Optimization (Section 5).** We organize internalized model optimization along the pipeline, from the architectural substrate and data and environment synthesis to agentic pre/mid-training, fine-tuning, reinforcement learning with long-horizon credit assignment, on-policy distillation, and self-evolution, tracing how capabilities first scaffolded through externalized harness engineering are progressively internalized into the policy.

5. **Application (Section 6).** Organized by the agent–environment interface, we map long-horizon pressures across software engineering, information seeking, computer use, multimodal agents, and general-purpose agents, revealing a common long-horizon core beneath surface-level domain differences, and consolidate horizon-aware evaluation resources.

6. **Frontier (Section 7).** We organize open problems into four axes, namely evolution, effectiveness, efficiency, and trustworthiness, spanning nine concrete directions that range from self-evolving and transferable

harnesses, to real-world environments and error robustness, to cost- and modality-aware efficiency, and to safety, governance, and embodiment, marking what must be solved as horizons keep growing.

Taken together, these contributions serve three purposes: a *clear definition* of long-horizon agency; a *high-quality taxonomy* (Figure 2) built around the paper's six perspectives; and a *roadmap* sketching the directions toward capable long-horizon agents. Beyond organizing existing work, this co-evolutionary view also offers a vantage point for reasoning about how the balance between externalized harness and internalized optimization may shift as horizons grow, and thus for anticipating the frontier forms that long-horizon agents may take.

Scope and Boundaries. We delimit our scope along four boundaries. **(1)** This is not a general investigation of LLM agents, memory, or reinforcement learning: we treat memory, reinforcement learning, and self-evolution through the co-evolution of externalized harness engineering and internalized model optimization, and only as they bear on long-horizon competence. **(2)** We exclude general ML platform infrastructure (e.g., container orchestration, GPU-cluster scheduling), and consider inference and decoding only where they materially affect long-trajectory execution. **(3)** We address model-architecture internals only for their effect on horizon, i.e., making long trajectories representable, trainable, and affordable to decode, while still treating reasoning and long-context modeling as foundational capabilities. **(4)** We draw on official documentation, open-source implementations, and emerging specifications alongside indexed academic publications, prioritizing documented engineering practice.

2 Foundation: Formalizing Long-Horizon Agents

Overview. Section 1 positioned long-horizon agency as a system-level capability, jointly shaped by the internalized model optimization and the externalized harness engineering that surrounds it. This section makes that claim precise and lays the conceptual groundwork for the rest of the paper, organized around four questions that follow the section's structure:

- **What is a long-horizon agent? (Section 2.1)** We define it by the coupled logical dependencies of a task rather than by elapsed time, and cast the agent as a base policy coupled to a surrounding harness, so long-horizon agency becomes a property of the model–harness system rather than of the foundation model alone.
- **How does its difficulty decompose? (Section 2.2)** We use how far execution stretches beyond one working context as an observable proxy for the underlying dependency burden, giving three nested levels (H1–H3) and their capabilities (C1–C3).
- **How fast is the frontier moving? (Section 2.3)** We quantify the rising time horizon of frontier agents as an empirical scaling trend and compare its full-period and recent-window doubling times.
- **How does it differ from neighboring notions? (Section 2.4)** We separate long-horizon agency from long-running execution, autonomy, and self-evolution, showing that it integrates these rather than reducing to any one.

2.1 Definition

We begin with the two objects of study: the task and the agent that seeks to complete it. As introduced in Section 1, we characterize long-horizon agency not by elapsed time or token count, but by the *coupled logical dependencies* that distribute a task across many interdependent steps. It is this chain of decisions, together with the interaction it entails, rather than duration alone, that determines the length of the horizon.

Long-horizon Tasks. A **long-horizon task** is a goal that may be under-specified at the outset and may require progressive clarification during solving, and whose solution path admits many possibilities and typically couples several complex subtasks together. As a result, it cannot be answered by a single-turn response in isolation, but only by *composing many interdependent decisions* into a single coherent trajectory [286, 811]. Concretely, a long-horizon task requires the agent to reason over many steps while interacting with the environment and, where relevant, the user: it progressively narrows down the solution path and continually tests and revises its reasoning direction until the goal is reached. Such a task may run from minutes to days

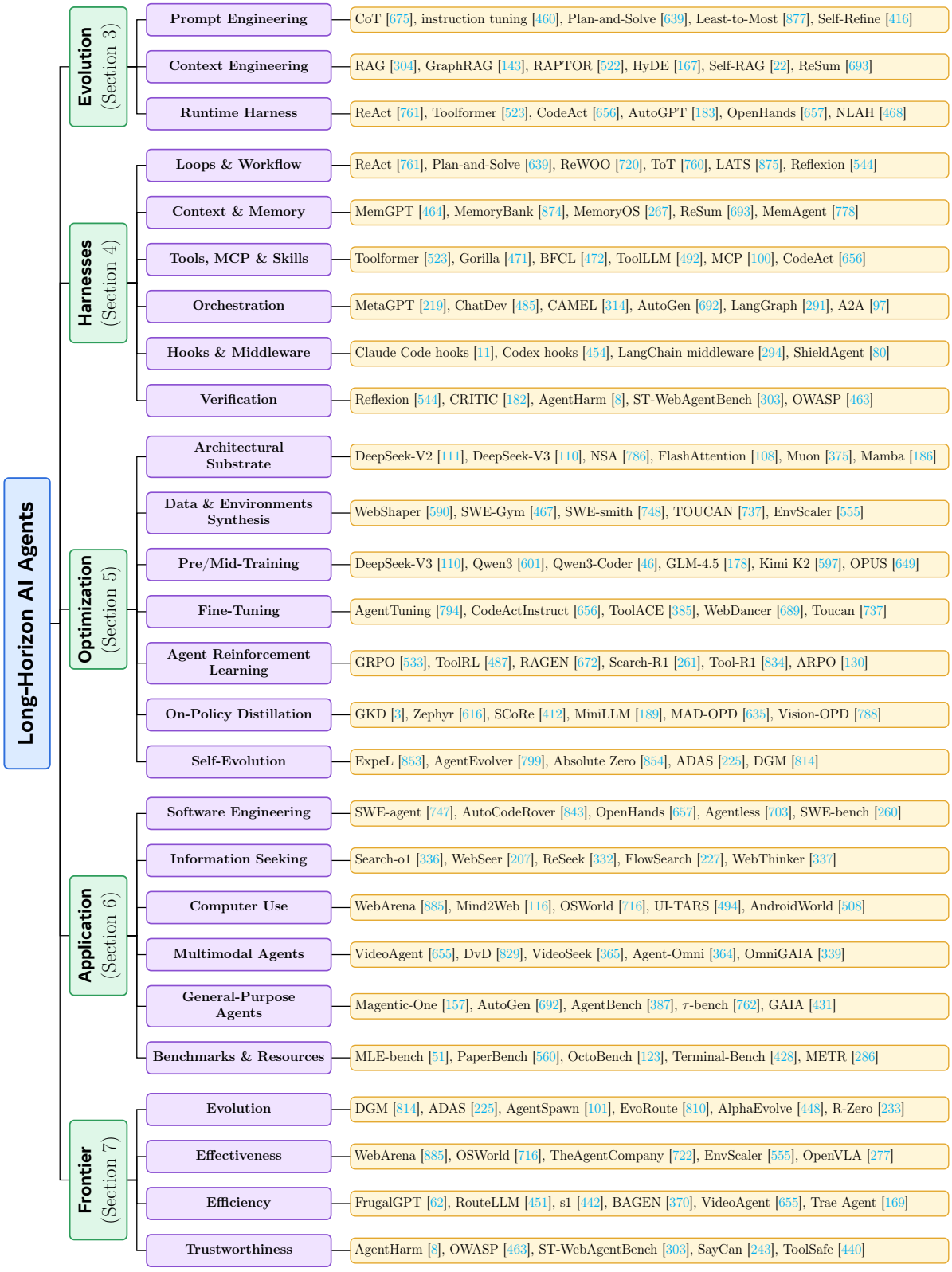


Figure 2 A taxonomy of long-horizon agent research. The paper is structured according to this taxonomy, progressing from **Evolution** (Section 3) through **Harnesses** (Section 4), **Optimization** (Section 5), and **Applications** (Section 6), to open **Frontiers** (Section 7). Each leaf lists representative works or systems discussed in the corresponding part of the paper.

and may fit within one context window or stretch far beyond it; its defining feature is not duration itself but this sustained, feedback-driven dependence among steps and the continual interaction required to manage it.

Long-horizon Agents. A **long-horizon agent** is a system designed to solve such tasks in an environment with state, feedback, and side effects [640]. Although the stated intent may be simple, successful execution requires resolving a large and tightly coupled chain of logical dependencies. Unlike a single-turn responder, it does not produce one answer and stop; instead, it must sustain reasoning through repeated cycles of planning, action, observation, and revision, progressively clarifying an under-specified goal as the trajectory unfolds. As argued in Section 1, this competence is not a property of the foundation model alone. It arises from the joint design of an **externalized harness engineering**, which supplies runtime structure such as tools, memory, orchestration, and checks, and an **internalized model optimization** process, which improves the policy through data, training, and experience.

Formalization: Long-Horizon Agent

We model a long-horizon agent as a foundation-model *base policy* π_θ coupled to a surrounding structure $\mathcal{H}$ that closes the loop around it:

$$\boxed{\text{Agent} \;=\; \pi_\theta \;\oplus\; \mathcal{H}} \tag{1}$$

Here π_θ proposes actions, while $\mathcal{H}$ comprises the loops and workflows, system prompts, tools and skills, context and memory, orchestration logic, hooks, and verification mechanisms that govern *when* the model is invoked, *how* its actions are validated and executed, and *what* state persists across steps. The operator $\oplus$ denotes runtime coupling rather than simple addition: $\mathcal{H}$ closes the action–observation–memory loop around π_θ, so long-horizon agency is a property of the coupled system and its training, not of π_θ alone. Equation (1) describes the *deployed* coupling that produces long-horizon behavior at run time; how it *evolves* over time, as π_θ internalizes competence through training and the system accumulates experience, is the co-evolving recipe we develop in Section 5 and Section 3. We refer to these two halves as **Pillar I**, the externalized harness $\mathcal{H}$ (Section 4), and **Pillar II**, the internalized optimization of π_θ (Section 5).

The Long-horizon Agent–Environment Loop. Since a long-horizon agent makes progress precisely by interacting with its environment step after step, its execution is most naturally described as a continual loop in which it acts, observes the consequences, and updates its state before acting again. Following the standard LLM-based agent formulation [228, 640, 761], we place the coupled system in an environment. Let $\mathcal{S}$ be an environment state space, $\mathcal{O}$ an observation space, and $\mathcal{A}$ an action space. The environment evolves by a transition kernel $s_{t+1} \sim \Psi(\cdot \mid s_t, a_t)$ and emits observations through an observation kernel $o_t \sim Z(\cdot \mid s_t)$, which is partial and in general stochastic. We distinguish the base policy π_θ from the *agent policy* $\pi = \pi_\theta \oplus \mathcal{H}$ that the harness closes around it: at each step the harness assembles a contextual signal c_t and the base policy acts on it,

$$c_t = \mathcal{H}(o_{0:t},\, a_{0:t-1},\, \mathcal{Q})\,, \qquad a_t \sim \pi_\theta(\cdot \mid c_t),$$

where $\mathcal{Q}$ is the task specification and c_t is the contextual signal that the harness assembles at step t from the interaction history: the prompt window with retrieved memories, prior tool outputs, reasoning traces, and other state. Execution induces a trajectory $\tau = (s_0, o_0, a_0, s_1, o_1, a_1, \ldots, s_T)$ that runs until termination, i.e. until the first step $T = \min\{t : s_t \in \mathcal{S}_{\text{term}}\}$ at which the state enters a terminal set $\mathcal{S}_{\text{term}}$ (the goal is reached, a budget is exhausted, or the agent halts). Because o_t is only a partial view of s_t and c_t is a lossy summary of history, the coupled system is naturally modeled, following standard POMDP formulations for sequential decision making under partial observability [264], as a partially observable Markov decision process (POMDP). Much of what $\mathcal{H}$ does is the machinery of acting under partial observability. The task assigns each trajectory a utility $R(\tau) \in \mathbb{R}$, typically sparse and delayed and often revealed only at T, so that credit for the intermediate decisions that actually drive success or failure must be inferred from a single terminal signal. This is the long-horizon credit-assignment problem that is central to reinforcement learning methods [579].

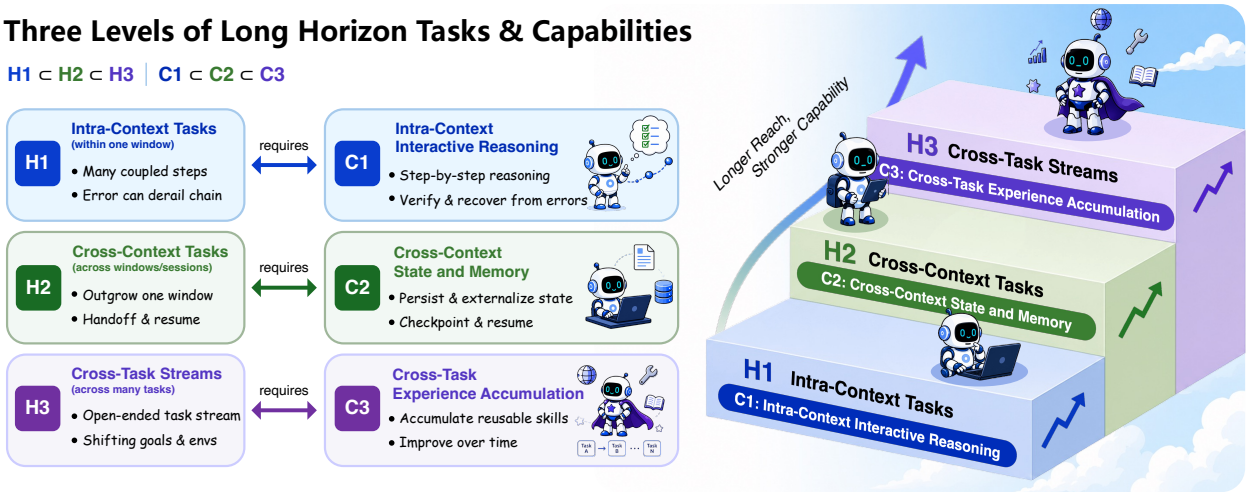


Figure 3 Three levels of long-horizon tasks and the capabilities they demand. As the required horizon widens from intra-context tasks (H1), to cross-context tasks (H2), to cross-task streams (H3), the agent must add intra-context interactive reasoning (C1), cross-context state and memory (C2), and cross-task experience accumulation (C3) capabilities.

2.2 Three Levels of Long-Horizon Tasks and Capabilities

Although the notion of a *long-horizon* agent is by now widely used, it remains conceptually vague, and prior work still lacks a principled way to partition long-horizon competence into distinct levels. Building on the definition above, we organize long-horizon tasks by an operational axis: *how far execution must stretch beyond a single working context*, ranging from intra-context tasks completable within one window (H1), to cross-context tasks that span windows or sessions (H2), to cross-task streams in which an overarching goal unfolds over many episodes (H3). This axis is a proxy for, rather than the definition of, logical dependency: a context boundary matters when coupled decisions and task state must survive it, whereas a long but decomposable batch process need not be long-horizon. The proxy is useful because no step is perfectly reliable and errors compound along the dependency graph, so trajectory-level reliability rather than per-step accuracy governs success [286]. It yields three nested operational levels of increasing reach [14, 228], and we pair each level with the capability an agent needs to succeed at it (Figure 3). Because every level adds a new failure mode on top of the one below, the corresponding demands accumulate. The time ranges below are therefore descriptive of current systems, not definitional thresholds.

Three Levels of Long-Horizon Tasks (H1 ≺ H2 ≺ H3)

H1 — Intra-context, within one window (∼minutes).
A single task can be completed within one working context window, yet it is far from a one-shot query: it requires composing many interdependent decisions into one coherent trajectory, with repeated interaction with the environment and, where relevant, the user to test and revise the reasoning direction until the goal is reached [286, 544].

H2 — Cross-context, across windows/sessions (∼hours to days).
A single task far exceeds one context window or session and must be carried across multiple context windows or by several agents working in concert. The agent(s) must therefore actively compress and externalize historical context, track interaction state in real time, and resume progress without assuming that the full history remains visible at every step [14, 378, 511].

H3 — Cross-task, open-ended task stream (∼lifelong).
Tasks arrive as an open-ended stream rather than a closed episode. A single overarching goal may therefore span many tasks, each of which may itself require multiple context windows to complete. In deployment, such settings often demand persistent, and sometimes lifelong, operation across heterogeneous environments, tools, and sessions [165, 228, 580, 808].

So far we have organized long-horizon *tasks* into three difficulty levels; we now turn from tasks to *agents* and identify, for each level, the capability that its difficulty demands. Each level exposes a distinct insufficiency. H1 shows that per-step accuracy alone is insufficient: the agent must sustain reasoning through continual interaction rather than produce a single response. H2 shows that a larger context window is insufficient: once a task spans multiple windows or agents, state must be compressed, externalized, and tracked even when the full history is not continuously in view. H3 shows that episodic success is insufficient: an agent that completes an individual task may still fail to accumulate reusable competence across an open-ended stream of subsequent tasks. Each level therefore demands a capability of matching reach, denoted C1–C3 respectively, and these capabilities compose cumulatively: C2 presupposes C1, and C3 presupposes both.

Capabilities Demanded by Each Level (C1 $\prec$ C2 $\prec$ C3)

C1 — Intra-context interactive reasoning (for H1).
Within a single context, sustain continuous reasoning through interaction with the environment and the user: take actions, absorb feedback, and revise the trajectory so that many interdependent decisions compose into one coherent path toward the goal [286, 544].

C2 — Cross-context state and memory (for H2).
Building on C1, maintain task state when execution spans windows or agents: compress and externalize historical context, read and write memory, checkpoint progress, and faithfully resume after handoffs [14, 378, 511, 694].

C3 — Cross-task experience accumulation (for H3).
Building on C2, turn experience into lasting competence across an open-ended task stream: switch among complex tasks, each of which may itself span multiple context windows, accumulate reusable skills [630], generalize from past experience [462], and continually improve as goals, environments, and data distributions shift [165, 228, 580].

Because the levels nest, so do the capabilities, in that an agent operating at H3 must still satisfy C1 and C2 within every task it touches. This is where the paper's central question enters. Each capability can be supplied either from *outside* the model, through the harness, or from *inside* it, through training. H1 pressures are often met by C1 within a single context, or absorbed into the model weights; H2 pressures push C2 out of the window into durable external state, namely compressed context, memory, and checkpoints; and H3 pressures demand C3, which turns deployment experience into lasting competence. In many H3 settings, however, neither route is likely to suffice on its own, and the highest level of long-horizon competence is therefore inherently a *system-level* property, jointly constructed by internalized model weights and the externalized harness rather than achievable by either in isolation (Figure 1). We attach each method to the level it primarily serves.

2.3 Scaling the Time Horizon of Frontier-Agent Execution

In the previous subsection, we organized long-horizon difficulty and its corresponding capabilities into a stair-stepped ladder *qualitatively*; here, we quantify how an observable proxy for the frontier scales over time. We build on the measurements of METR [286], which define the *50%-task-completion time horizon* as the human-expert completion time for the class of agentic engineering tasks that a given agent is predicted to complete with 50% reliability. These tasks span software engineering, machine learning engineering, and computer use, providing a more objective proxy for task difficulty than task labels alone.

Figure 4 plots METR's live estimates through May 8, 2026 [286, 429][1]. Because several early models were not re-evaluated under the newer TH1.1 methodology, the full historical series combines TH1.0 estimates for those models with TH1.1 estimates for later models. Under this stitched series, the time horizon of frontier agents rises from seconds for early models (e.g. GPT-2) to approximately twelve hours for Claude Opus 4.6 and at least sixteen hours for an early Claude Mythos Preview by May 2026, spanning nearly four orders of magnitude within only a few years.

[1]METR live dashboard, https://metr.org/time-horizons/, accessed May 8, 2026.

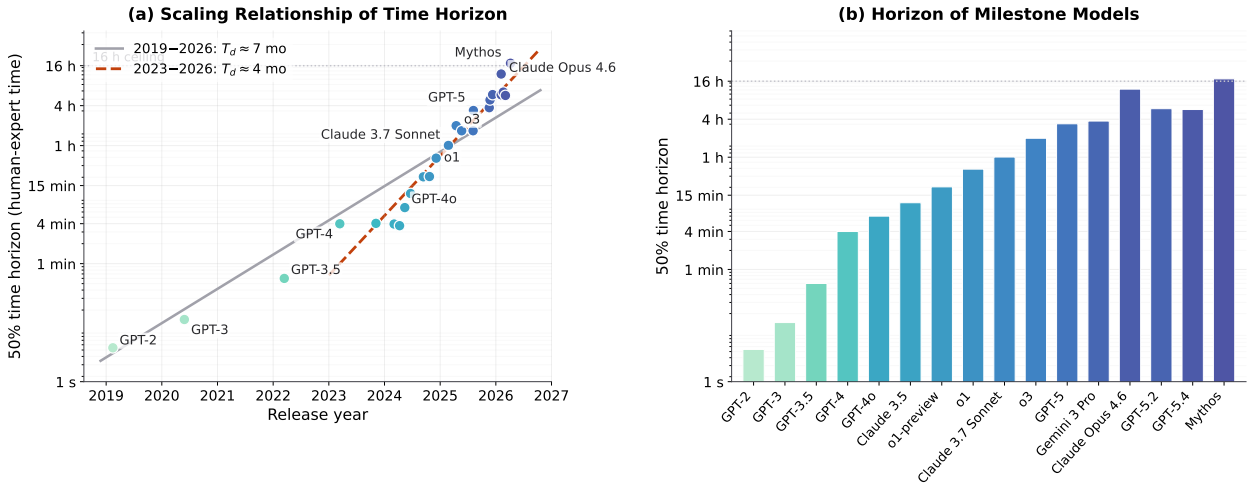


Figure 4 Empirical growth of frontier-agent time horizons. Frontier agents' 50%-task-completion time horizons over model release dates, shown for the full fitted series in **(a)** and selected milestone agents in **(b)**.

The fitted trend in Figure 4 summarizes this rapid expansion. Across the stitched full-period series, the estimated doubling time is approximately 196.5 days, or 6.5 months. Restricting the fit to models released from 2023 onward under TH1.1 yields a shorter doubling time of approximately 130.8 days, or 4.3 months [286, 429, 683]. The dotted line marks the approximately sixteen-hour region above which METR considers estimates unreliable under the current task suite. Consequently, both individual estimates and fitted trends near or beyond this boundary carry substantial uncertainty.

To summarize the trend formally, we borrow the functional spirit of the compute-optimal scaling law of Hoffmann et al. [217] and model the frontier horizon in terms of two quantities: the release date of a frontier agent and the human-expert duration of tasks that the evaluated system can complete. We write this relationship as a compact descriptive summary:

$$\log_2 H(t) \approx \log_2 H_0 + \frac{t - t_0}{T_d}, \qquad \text{equivalently} \qquad H(t) \approx H_0 \cdot 2^{(t-t_0)/T_d}, \tag{2}$$

where $H(t)$ is the estimated time horizon of a frontier agent whose base model was released at date t, t_0 is a reference date, $H_0 = H(t_0)$ is the fitted horizon at that date, and T_d is the *doubling time*. Under this functional form, the full-period and 2023-onward fits correspond to $T_d \approx 196.5$ days and $T_d \approx 130.8$ days, respectively. Although these estimates are derived from self-contained, automatically scorable software-style tasks and become less precise near the upper boundary of the benchmark, these limitations primarily constrain the exact fitted values rather than the overall direction of the trend.

The decisive signal is that the task duration frontier agents can reliably handle has increased approximately exponentially. The shorter doubling time obtained for the 2023-onward subset further suggests that this expansion may have accelerated, although differences in evaluation coverage and methodology prevent treating the two fitted values as definitive evidence of acceleration. The trend nevertheless aligns directly with our capability hierarchy: as completable task horizons expand from minutes to hours and then to tens of hours, frontier agents move beyond intra-context H1 tasks, increasingly address H2 tasks that require cross-context state and memory, and begin to approach the boundary of cross-task, open-ended H3. In this sense, the sustained expansion of the measured time horizon is an observable projection of long-horizon competence climbing the H1→H2→H3 ladder. At the current pace, many multi-hour tasks that remain at the frontier today may become routine in the foreseeable future. That said, the trend characterizes *progress over time*; it does not explain its *source*. In particular, it does not distinguish whether the gains arise from a stronger policy π_θ, a better harness $\mathcal{H}$, or the co-evolution of the two. Attributing observed progress among these factors is the central question of this paper. We therefore turn from measuring the frontier to asking what drives it.

2.4 Long-Horizon Agent vs. Neighboring Concepts

Long-horizon agency is often conflated with three adjacent notions: long-running execution, autonomy, and self-evolution. They are related because all appear in modern agent systems, but they answer different questions. Long-running asks how long execution persists, autonomy asks how much human control remains, and self-evolution asks whether the agent improves itself over time. Long-horizon agency asks a different question: can the coupled system sustain interactive reasoning within a context (C1), maintain state and memory across contexts when needed (C2), and accumulate experience across tasks when deployment demands it (C3)? Figure 5 summarizes the overlap and separation.

2.4.1 Long-Horizon Agent vs. Long-Running Agent

A long-running agent is defined by duration: it keeps making progress over hours to weeks across many windows, sessions, or sandboxes, with checkpointing and resumption [14]. It often appears in long-horizon settings, but duration and difficulty are not the same.

Overlap. Many genuinely long-horizon tasks are also long-running, because their trajectories outgrow one context window or one execution session. In these cases the same engineering mechanisms that instantiate C2 matter: compressing and externalizing historical context, durable memory, progress logs, checkpoints, and faithful resumption. Long-running engineering is therefore a common implementation requirement in H2 settings, where task state must survive resets, context boundaries, and hand-offs without assuming that full history remains visible.

Distinctions. The key distinction is that duration and *logical coupling* are independent properties. A task may run for days while consisting mostly of logically decoupled, independently checkpointable steps; a script that loops over a large dataset and applies the same operation runs for a long time, yet its steps carry few interdependencies, so it demands neither dense, sustained interactive reasoning (C1) nor live management of cross-context state (C2). Conversely, a task may complete within a single window while still requiring many interdependent decisions whose direction must be continually tested and revised through interaction with the environment. Long-running engineering thus optimizes persistence, throughput, and cost over time, whereas long-horizon agency concerns whether an agent can sustain reasoning through continual interaction and, when the task demands it, maintain the state and memory needed to carry that reasoning across contexts [286]. Accordingly, execution duration captures temporal persistence, whereas long-horizon agency denotes the capacity to sustain coupled reasoning and state management across the task's dependency structure.

2.4.2 Long-Horizon Agent vs. Autonomous Agent

An autonomous agent is defined by how much it can act without user involvement, often formalized as roles ranging from operator to observer [155]. Autonomy and long-horizon agency are strongly related, but they describe different properties of a system.

Overlap. The two overlap because long-horizon tasks make constant human supervision expensive, so most long-horizon agents are largely automated: they interact with external environments and tools on their own, and autonomously decide when to revise their strategy. If an agent must take many steps, interact with tools and the environment, absorb feedback, and revise its trajectory, then requiring approval at every step can destroy the usefulness of the system, even when the task itself fits within H1. At the same time, autonomy raises the safety stakes of long-horizon execution, since more unattended steps and more coupled side effects mean a larger blast radius, so permission gates and oversight remain central.

Distinctions. The distinction is that autonomy is only a partial, governance-oriented dimension of a system, not a measure of its capability. A strong long-horizon agent can be run with low autonomy behind approval gates, whereas a highly autonomous agent may execute only a short, single-window task, such as a single web action [452]. Moreover, long-horizon agents need not be fully autonomous: their harnesses can introduce permission gates, approval checkpoints, and escalation policies that selectively involve humans at high-impact or uncertain steps. Autonomy therefore asks who remains in the loop, whereas long-horizon agency asks

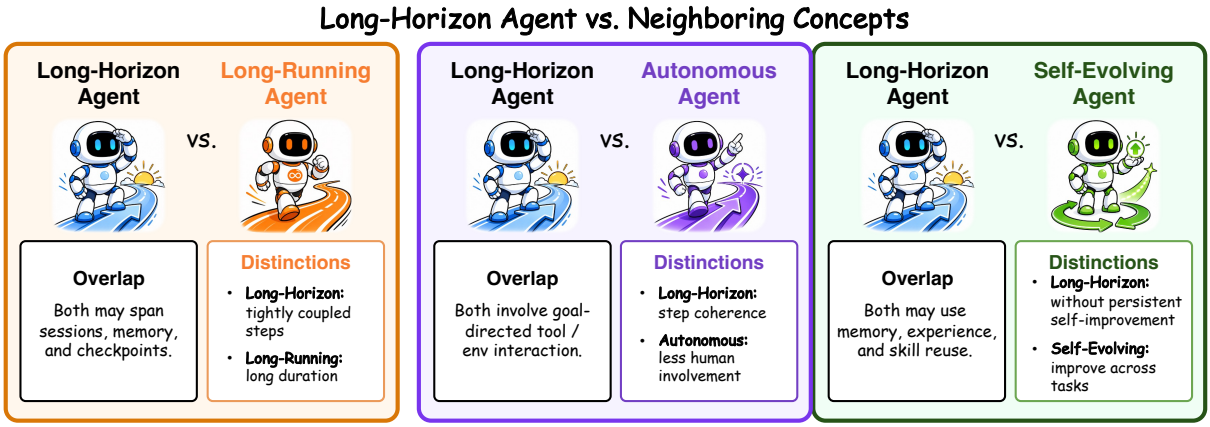


Figure 5 Long-horizon agent and three neighboring concepts. Long-running execution, autonomy, and self-evolution each emphasize a different facet; long-horizon agency asks whether the coupled model–harness system can sustain interactive reasoning (C1), maintain state and memory across contexts when needed (C2), and accumulate experience across tasks when deployment demands it (C3).

whether the coupled system can sustain the reasoning, state management, and, at H3, experience accumulation that the task horizon demands. In short, autonomy controls the degree of human involvement, whereas long-horizon agency measures whether execution can be carried through the required horizon.

2.4.3 Long-Horizon Agent vs. Self-Evolving Agent

A self-evolving agent is defined by experience-driven, *persistent* change to its own parameters, memory, tools, or topology, with the goal of improving future performance. It is the closest neighbor to long-horizon agency because both concern competence beyond a single model call, but they still differ in scope.

Overlap. At the cross-task (H3) scale the two overlap strongly. A long-horizon agent that switches among complex tasks, accumulates reusable skills, consolidates memory, and reuses experience across episodes begins to approach self-evolution. Both also fit the "era of experience" view that agent competence should grow from interaction rather than only from a frozen pretraining corpus [580]. In this paper, self-evolution is one internalized mechanism for realizing C3, namely persistent cross-task experience accumulation.

Distinctions. The distinction is whether experience persistently changes the agent for the future. Self-evolution requires such persistent change and is therefore only one facet of the C3 capability, cross-task experience accumulation, rather than the whole of long-horizon agency, which also rests on core capabilities such as interactive reasoning, tool use, and long-context reasoning. Consequently, many long-horizon agents operating at H1 or H2 complete difficult tasks with a fixed policy and temporary state, finishing a trajectory without updating their weights, tools, or lasting memory. Such agents lack any self-evolution capability, yet they remain long-horizon agents.

Self-evolution therefore concerns persistent improvement *between* tasks, while long-horizon agency additionally concerns reliably carrying each task *through* the extended interaction, state management, tool use, and iterative reasoning that its dependency horizon demands.

Overall, this section has laid the conceptual foundation on which the rest of the paper builds. We defined a long-horizon agent as a coupled model–harness system, $\pi_\theta \oplus \mathcal{H}$, rather than as the foundation model alone (Section 2.1); organized its difficulty and matching capabilities into three nested levels, H1–H3 and C1–C3 (Section 2.2); quantified the empirical growth of a frontier-agent time-horizon proxy and compared alternative fitting windows (Section 2.3); and separated long-horizon agency from long-running execution, autonomy, and self-evolution (Section 2.4). The methodological takeaway is that long-horizon agency is a high-level, system-level capability that integrates these adjacent properties rather than reducing to any one of them. An agent may exhibit sustained running, autonomy, and self-evolution and yet fail an H1 task when it cannot

Table 1 Three stages in the co-evolution of long-horizon agents. Each stage widens the *control unit*, opens a new *control surface*, and advances a set of *key technical routes*. The graded *long-horizon profile* on the right scores each stage against the three accumulating capabilities defined in Section 2.2, namely C1 intra-context interactive reasoning, C2 cross-context state and memory, and C3 cross-task experience accumulation (● full, ◗ partial, ○ absent).

Stage (era)	Control unit & surface	Key technical routes	Long-horizon profile		
			C1	C2	C3
I. Prompt engineering (2020–2023)	Single query *language of the prompt*	• Chain-of-thought reasoning • Decomposition, planning, and search • Prompt optimization and internalization	◗	○	○
II. Context engineering (2023–2025)	Conditioned call *information in the context*	• Retrieval-augmented generation • Tool use and function calling • Long context, memory, and context engineering	◗	◗	○
III. Runtime harnesses (2025–Present)	Whole trajectory *runtime around the model*	• Agent loops (reason–act–observe) • Harness engineering • Internalizing the loop	●	●	◗

sustain interactive reasoning through environment feedback, or fail an H2–H3 task when it lacks the state management or experience accumulation that those horizons require. Competence therefore shifts from the model alone to the coupled system $\pi_\theta \oplus \mathcal{H}$ and its training, and evaluation must track task completion, robustness over interaction, cross-context recovery, cross-task transfer, cost, and safety rather than static per-call accuracy. Having framed long-horizon agency as a coupled model–harness property organized along H1–H3 and C1–C3, we now turn to a historical question, namely how the field progressively moved capability from prompts, to context, to persistent runtime systems. Section 3 traces that progression.

3 Evolution: From Prompting to Runtime

As stated in the previous sections, we define a long-horizon agent as the joint product of *internalized model optimization* and *externalized harness engineering*, and reading the field through the lens of evolution shows why this coupling is not a recent design choice but a recurring necessity. Across the history of LLMs, the base model and the engineering structure around it have advanced together, each repeatedly compensating for the other's limits. A base model is a powerful but fragile next-step predictor that, on its own, does not reliably decompose a goal, ground its steps in the world, sustain reasoning through environment interaction, or maintain the cross-context state that longer horizons require (Sections 1 and 2.2), and supplying this missing structure has always been the task of the surrounding engineering. Following this co-evolutionary thread, this section traces the development of long-horizon agents along its two driving forces, model optimization and harness engineering, and organizes the representative work that defines each stage.

As shown in Figure 6, we trace this co-evolution through three nested, chronologically and thematically coherent stages, each opened by an expansion in the base model's own capability, organized around a new control surface exposed by the surrounding harness, and closed by the long-horizon limit that forces the next stage. **(1) Prompt Engineering** (2020–2023) steers behavior through the *language* of a single query, first eliciting multi-step reasoning from a fixed model; **(2) Context Engineering** (2023–2025) extends control to the *information* supplied to the model, grounding the agent in retrieved evidence, tools, and memory; and **(3) Runtime Harnesses** (2025–present) make the unit of control the whole *trajectory*, so that the long-horizon agent proper emerges as a model coupled to a runtime that sustains one goal over many dependent steps. The three are not successive replacements but nested layers, in which each new control surface absorbs the previous one rather than discarding it. Throughout this chapter, the years attached to each stage denote the period in which its control surface became the *dominant* paradigm, not the publication date of any individual work; a stage may therefore rest on foundational methods that predate it and preview ideas that mature later. Table 1 summarizes the three stages along a common set of axes.

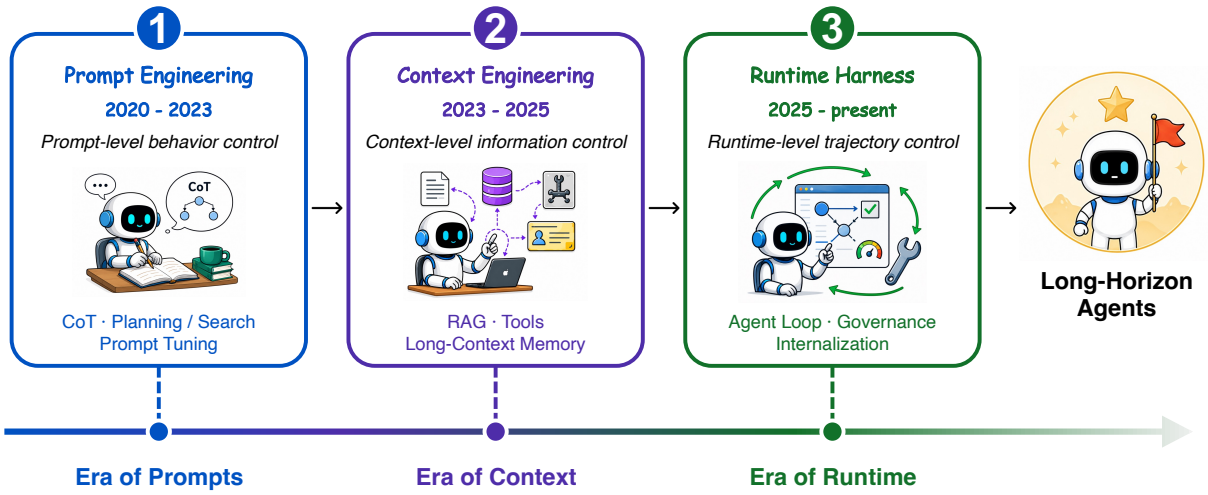


Figure 6 The co-evolution of long-horizon agents across three stages. Control widens from the *language* of a single prompt (2020–2023), to the *information* supplied to the model before it acts, to the whole *trajectory* sustained by a runtime harness (2025–present); each stage absorbs the previous rather than replacing it.

3.1 Stage I: Prompt Engineering (2020–2023)

The prompting stage is the prototype of long-horizon agency, in which the model carries most task capability in its weights and the only surface a builder can touch is the language of the query. The stage opens with *prompt learning*, made widely visible by GPT-3 [40] and consolidated by dedicated papers on prompting and in-context learning [132, 379], which let a pre-trained model perform a new task from a handful of in-prompt examples with no weight update and thereby moved the focus of system building from training the model to shaping its input at inference time. Its central discovery is that the right prompt can *elicit* latent abilities the base model does not show by default, including the embryonic forms of planning and execution that long-horizon tasks demand. Concretely, the prompting stage surfaces this latent competence along three representative threads, which we take up in turn: chain-of-thought reasoning, decomposition and planning, and the optimization and internalization of prompts themselves.

Chain-of-Thought Reasoning. The first thread showed that a fixed model reasons far better when prompted to externalize intermediate steps rather than answer directly. Initially, the scratchpad [450] emitted intermediate computation for multi-step problems; chain-of-thought (CoT) prompting [675] then turned this into a few-shot recipe that elicits a reasoning trace before the answer; and zero-shot CoT [281] further showed that the same behavior could be triggered by a single instruction, confirming that the capacity was latent in the weights and merely needed surfacing. Building on CoT, a family of variants quickly followed that automate demonstration construction, scale reasoning effort with problem complexity, and modularize the prompt into reusable sub-steps [160, 275, 851]. The shared lesson, that the same model exhibits qualitatively different competence depending on how its input is phrased, recurs throughout the paper.

Decomposition, Planning, and Search. A second thread widened the single chain into the structures that prefigure an agent, using prompting not merely to reason but to make a fixed model *decompose* a goal, *plan* a course of action, and *execute* it step by step, which is the earliest recognizable form of an agent. Self-consistency [659] samples many chains and aggregates them by majority vote, trading one fragile path for a more reliable distribution; least-to-most prompting [877] made decomposition explicit by solving ordered subproblems; and Plan-and-Solve prompting [639] split an explicit planning step from execution, producing the first clean "plan then act" template that later agent loops inherit. A parallel line grounded execution in code, with PAL [168] and Program of Thoughts [70] offloading computation to an interpreter rather than the model's own arithmetic, while Tree-of-Thoughts [760] recast reasoning as deliberate search over a tree of partial solutions with look-ahead and backtracking. A closely related line then pushed the

same plan-then-act template into *interactive* environments, where SayCan and Inner Monologue grounded language plans in affordances and environment feedback [236, 243], and AdaPlanner, ADaPT, and DEPS closed the loop with adaptive re-planning from execution feedback in embodied and text worlds such as ALFWorld [482, 545, 569, 673]. Read together, this line forms the embryonic agent skeleton, which decomposes a goal, proposes and executes steps, evaluates progress, and revises. Prompt learning alone cannot yet yield a long-horizon agent, since all of this unfolds without a persistent runtime harness, durable state, or robust recovery. What the stage does provide is a soft external constraint on model behavior and an early exploration of the harness-mediated control that later stages systematize, together with the planning-and-execution skeleton here and the elicitation-then-internalization dynamic below.

Prompt Optimization and Internalization. A third thread responded to a property this stage exposed and the rest of the paper inherits, namely that elicited capability is *fragile*, swinging with wording, example order, example choice, or output format [403, 433, 862], which made prompt-based systems hard to reproduce, test, and compose. The community answered by making prompt construction systematic rather than handcrafted. One line recast prompt design as an explicit search or optimization problem [483, 889], while another treated prompts as composable, program-like artifacts [514]. In parallel, what began as a prompting trick started to be *internalized*, as instruction tuning together with reinforcement learning from human feedback let models follow intent without elaborate templates, most visibly in InstructGPT [460], and the same dynamic would later fold full chain-of-thought traces into the weights, which we revisit in Section 5. This is the first concrete instance of the harness→policy migration the paper tracks across all three stages.

Although prompt engineering by itself does not constitute a long-horizon agent, its core idea, shaping a model's reasoning behavior through external constraints rather than weight updates, remains a guiding principle for long-horizon agents to this day. Yet prompting controls only *language* over a fixed knowledge base. A single call can use only the text in the prompt and the knowledge already stored in the parameters [266, 479, 516], so no phrasing can supply information the model never saw. As applications moved toward current events, private data, large repositories, and changing APIs, the binding question shifted from how the instruction is worded to what the model is allowed to see, pushing engineering one layer outward, from the language of the prompt to the information assembled into it.

3.2 Stage II: Context Engineering (2023–2025)

Where Stage I engineered the model's *behavior* through the language of the query, Stage II turns to its *information*, that is, the evidence, tools, and history that a single call is allowed to condition on. Broadening this input, rather than rewording the instruction, becomes the primary means of extending the model's capability boundary. The representative directions of this stage are retrieval-augmented generation, tool use and function calling, and context management and memory, which we detail in turn below.

Retrieval-Augmented Generation. Retrieval-augmented generation (RAG) loosened the boundary between the model and the world by attaching external evidence to the prompt at inference time [304], building on a line of retrieval- and memorization-augmented models that coupled parametric knowledge with an external corpus [38, 200, 246, 247, 270, 273] and organized systematically in dedicated RAG reviews. Its core result is that grounding generation in retrieved text reduces hallucination and improves factuality on knowledge-intensive tasks. Initially a single retrieve-then-read pass, the technique soon evolved into a richer family. HyDE-style hypothetical-document embeddings first improved recall [167], and hierarchical indexing summarized corpora into multi-resolution trees [522]. Retrieval was then interleaved with reasoning, through multi-hop methods that chain retrieval across steps [612] and active retrieval that lets the model decide *when* to look something up [258]. More recently, self-reflective retrieval further lets the model critique and re-retrieve its own evidence [22].

Tool Use and Function Calling. Where retrieval lets a model *read* the world, tool use lets it *query and act on* systems beyond text. Early work established the mechanics. Toolformer [523] showed that a model could teach itself, in a self-supervised way, when and how to call APIs; ToolAlpaca [585] and ToolLLM [492], with the StableToolBench testbed [198], then scaled tool use to thousands of real-world APIs; Gorilla [471] grounded API selection in documentation to curb hallucinated calls; and WebGPT [445] grounded long-form

answering in live browsing. As the practice consolidated, function calling and structured outputs standardized the interface so that model decisions could be parsed and dispatched reliably [385, 472], a maturation charted by dedicated tool-learning papers [493, 497]. The decisive bridge is ReAct [761], which interleaves reasoning with tool calls so that each observation feeds the next thought, while HuggingGPT [537] further showed how an LLM could route tasks across specialized models in a tool ecosystem. Both sit at the boundary of two stages, turning one-shot tool use into the beginnings of the loop that Stage III makes central.

Long Context, Memory, and Context Engineering. The third development enlarged the container itself. Efficient-attention kernels such as FlashAttention [108] improved the practicality of long-sequence serving, while sparse-attention architectures such as Longformer [32] and BigBird [792] explored algorithmic routes to longer inputs; later frontier systems exposed hundred-thousand- to million-token windows [593], and state-space architectures offered near-linear alternatives to attention [186]. This made it physically possible to condition on whole documents, codebases, and histories rather than summarizing them away, which shifted the question from whether the model can reach information to whether it can use the information well once present. LongBench [28] was among the first benchmarks to quantify how poorly models handle genuinely long-context tasks, a gap that dedicated long-context suites continue to make precise [7, 220, 833]. Because no window is unbounded, a complementary line externalized state into persistent memory, with MemGPT treating context as a paging hierarchy over external storage [464] and generative agents maintaining a long-running memory stream with retrieval and reflection over remembered experience [470]. This abundance, however, created its own problem, since more context is not better context. Long inputs are read unevenly, with material in the middle systematically ignored [378]; irrelevant evidence actively distracts reasoning [541]; and utilization past a threshold degrades sharply, a failure measured as *context rot* [511]. The methodological response, named by practitioners *context engineering*, is the systematic selection, organization, compression, and refresh of the model's input under hard token budgets [13, 425], with a toolkit spanning prompt compression [254, 351, 773], working-memory mechanisms [224, 268, 778], and rolling summarization for long runs [693, 890]. More recently, this curation is itself becoming agentic, as *agentic context engineering* lets the agent treat its own context as an evolving object that it revises and improves over successive steps [824]. Together, retrieval, tools, longer windows, memory, and context engineering shifted the relevant capacity from how much knowledge sits inside the model to how effectively the surrounding structure can selectively surface, organize, and use relevant information at each step; here too capability was partly internalized, as retrieval, tool use, and long-context handling increasingly became native, trainable model skills rather than bolted-on calls.

For all its reach, context engineering keeps the structural assumption it inherited from prompting, in that it optimizes the input to a *single* model call. The unit is wider, but it remains a one-shot, open-loop conditioning step that shapes what the model sees *before* it acts, not what happens after. Once deployments began chaining many such calls into consequential actions on tools, files, browsers, and operating systems, the dominant failures became properties of the *trajectory*, as errors compounded across steps, belief state drifted from the world, and side effects became irreversible. Closing that loop requires control over the runtime itself, which is the surface Stage III opens.

3.3 Stage III: Runtime Harnesses (2025–Present)

In Stage III, capability expansion manifests as a shift from answering questions to acting in environments, and the long-horizon agent takes its recognizable form. This also marks the transition of agents from chatbots toward genuinely productive settings, in which they carry out consequential work rather than merely converse. The transition was made concrete by executable environments and benchmarks such as WebShop [759], WebArena [885], SWE-bench [260], OSWorld [716], and Terminal-Bench [428], which score completed actions in web, code, desktop, and terminal settings rather than static answers, and thereby supply the trajectory-level feedback long-horizon agents are optimized against. The unit of engineering is no longer a prompt or a single conditioned call but a *trajectory*, comprising repeated cycles of reasoning, action, observation, verification, and recovery. We summarize the stage in three movements, namely the loop that makes the agent act, the harness that governs the loop, and the internalization that absorbs harness capabilities back into the model.

Agent Loops. The core engineering unit is a *trajectory* that unfolds repeatedly in an environment, in which the agent reasons, acts, reads feedback, and decides what to do next, following the widely adopted

thought–action–observation cycle. Early work such as ReAct, Reflexion, and Self-Refine established the basic pattern of interleaving reasoning with tool use and verbal reflection after failure [416, 544, 761], while language-agent tree search wrapped the loop in value-guided search with backtracking [875]; around the same time, open-source autonomous agents such as AutoGPT and BabyAGI popularized the self-directed loop for a broad audience [183, 444]. Since 2025, the loop has tightened along three axes rather than inventing a new paradigm. First, *action representation* became more executable and modular, as CodeAct unified step-level actions as code [656] and ReWOO separated planning from tool execution [720]. Second, *reliability* moved inward to the loop itself, as ReflAct grounds decisions in goal-state reflection [276] and test-and-repair loops such as CRITIC introduce tool-interactive verify-then-correct steps that catch and revise errors mid-trajectory [182], complementing the reflect-and-retry pattern of Reflexion and the value-guided search of LATS above. Third, *task form* widened from QA-style interaction to open-ended, multi-tool trajectories in benchmarks such as GAIA and AssistantBench [431, 774], to vision–language control in SeeAct [864], and to multi-stage research pipelines that interleave planning, retrieval, and writing [258, 613, 871]. Alongside these free-form loops, a complementary line re-emphasizes structured *agent workflows*, in which recurring procedures are captured as reusable graphs or memories and replayed across tasks, as in Agent Workflow Memory [674] and automated workflow generation [353]. The loop has thus become a runtime primitive that sustain one goal across many coupled steps while continuously changing external state, and the remaining question is how to govern it.

Harness Engineering. Once a loop stretches over a horizon, failures become properties of the *execution process* rather than of any single input, as errors compound, goals drift, success is declared too early, state distorts as context fills, and side effects escape authorization. The engineering response must therefore enter the runtime itself. *Harness engineering* names the organization of prompts, context, tools, state, permissions, verification, and recovery into a system-level control surface. The field's first response was to make multi-step execution explicit. Communicative multi-agent efforts led by ChatDev and MetaGPT cast long-horizon software work as role-specialized message passing [219, 485], AutoGen turned that pattern into a programmable conversation runtime [692], and Magentic-One added dual ledgers that separate outer replanning from inner dispatch [157]. As trajectories grew longer and more inspectable, control migrated from ad hoc dialogue to graph-structured runtimes exemplified by LangGraph and packaged into deep-research systems such as Deep Agents and Flash-Searcher, where state can be resumed, branched, and audited [291, 293, 491]. In parallel, the interface to the outside world was standardized, as MCP unified heterogeneous tools behind a discoverable, authorizable service layer [99], while AGENTS.md and Agent Skills externalized project rules as portable harness configuration [16, 98]. Long-running deployment then forced a third concern, what an agent is allowed to do, into the foreground through hooks, middleware, pre-action gates, sandboxes, and policy layers [17, 18, 41, 463, 617]. These layers are what product systems ultimately integrate, including repository-oriented harnesses such as OpenHands and SWE-agent, whose Agent–Computer Interface showed that interface design alone can determine long-horizon success [657, 747], local personal-agent runtimes such as Claude Code and OpenClaw [10, 457, 513], and GUI-facing systems such as Operator [12, 452]. A later turn makes the harness itself an object of search and improvement, as in workflow discovery by AFlow and ADAS [225, 383, 662, 816, 897] and self-editing harness code in the Darwin Gödel Machine [814]. Reliable long-horizon agency is therefore a system property to be engineered and learned. Practitioners increasingly describe this outward shift as *loop engineering*, a move from writing one-shot prompts to designing loops that dispatch, monitor, verify, and recover agents over a trajectory.

Internalizing the Loop. Stage III is also where the paper's two halves become visible together, as behaviors first organized by the harness are increasingly absorbed into model weights through trajectory-level SFT and RL. Post-2024 work no longer pursues a single generic agent-tuning recipe; instead, internalization *tracks harness type*, because each runtime exposes a different action grammar, feedback channel, and failure mode. Search-centric harnesses are distilled by agentic-search RL such as Search-R1 [261]; repository harnesses by software-engineering RL such as SWE-RL, increasingly trained inside neural environments like SWE-World [572, 680]; terminal harnesses by systems such as RLAnything that unify shell, tool-use, and coding rollouts [663]; GUI harnesses by UI-TARS-2 [632]; and personal runtimes by Claw-R1, which routes rich OpenClaw trajectories into RL backends through step-level middleware [457, 627]. Agent-World pushes this further by coupling environment–task discovery with continual self-evolving training, so that internalization is no longer confined to fixed benchmarks [131]. Evaluation moves in the same direction,

being less concerned with a single end-to-end score under one hand-built harness and more with whether a model natively carries the capability a harness would otherwise supply. Recent benchmarks accordingly probe cross-industry computer-use workflows [575], long-horizon tool and MCP use [322, 700], local control and multi-day collaboration [426, 764], and productive office reasoning [458]. Internalization does not make the harness disappear; it shifts the field from hand-assembling a runtime for every model toward distilling runtime experience into weights and validating transfer with harness-native benchmarks, a theme developed systematically in Section 5.

This is the co-evolution at the heart of the paper. Long-horizon capability is supplied *jointly* by an externalized harness and an internalized optimization process, with experience traces carrying capability between them. The next two chapters take up these halves in turn, treating the externalized harness in Section 4 and the internalized optimization in Section 5.

4 Harness: Externalizing Long-Horizon Capability

To analyze long-horizon capability, we start from the externalization perspective. Long-horizon capabilities need not originate in weights: before a model learns to plan, remember, or recover through training, these capacities can be *externalized*, that is, engineered into the runtime system through the harness that wraps the model at inference time and mediates every interaction it has with the world. It contains the machinery of loops, memory, tools, coordination, and oversight that extends a model into a long-horizon agent without changing what the model knows. Where Section 5 asks what training can absorb, this chapter asks what harness can supply. As shown in Table 2, we organize the harness into six components that recur across essentially every contemporary agent, ordered from the control loop outward to the systems that govern it:

The Six Components of an Agent Harness

1. **Loops and Workflows** (Section 4.1): specifies how execution moves across states, including when the model is called, how actions are taken, and how control returns after observations.
2. **Context and Memory** (Section 4.2): manages the state available to the model, from the active context of the current run to persistent memory across windows, tasks, and sessions.
3. **Tools, MCP, and Skills** (Section 4.3): defines how the agent accesses external capabilities, represents actions, and reuses acquired procedures.
4. **Orchestration** (Section 4.4): organizes work beyond a single agent loop, including task decomposition, role assignment, routing, and inter-agent coordination.
5. **Hooks and Middleware** (Section 4.5): provides predefined interception points in the runtime where execution can be inspected, modified, paused, or handed off.
6. **Verification** (Section 4.6): checks whether states, actions, outputs, or side effects satisfy task, safety, and quality constraints before execution continues or results are accepted.

4.1 Loops and Workflows

The innermost layer of a long-horizon agent harness is the workflow that advances execution from one state to the next. At this layer, the harness specifies the control pattern by which model calls, tool calls, state updates, checks, and feedback are sequenced. Recent workflow papers describe agent workflows as execution structures in which agents perceive state, reason, decide, act, receive feedback, and adapt under a workflow engine [777]. An example of a long-horizon run can be found in Figure 7.

For long-horizon agents, the central design question is how much of the trajectory is governed by this external control pattern rather than by the model's next-token policy alone. A workflow may simply repeat an observe–act loop, maintain an explicit plan across steps, or branch over multiple candidate trajectories before committing. We therefore classify harness workflows into three structures: *linear workflows*, which advance through a single trajectory; *plan-execute workflows*, which externalize task decomposition and progress tracking; and *branching workflows*, which search over alternatives using comparison, pruning, or backtracking.

Table 2 The agent harness by component. *Harness responsibility* states what the component must supply at runtime; *representative mechanisms* lists the component's sub-mechanisms, and *representative systems* pairs each sub-mechanism with two concrete systems on the same line.

Component (§)	Harness responsibility	Representative mechanisms	Representative systems
Loop & Workflow (§4.1)	Schedules model calls, action execution, and observation feedback across steps	• Linear workflows • Plan–execute • Branching	ReAct [761], Self-Refine [416] ReWOO [720], Arbor [262] Tree-of-Thoughts [760], LATS [875]
Context & Memory (§4.2)	Bounds in-run state and persists what must survive across windows and sessions	• Working context • Factual memory	ReSum [693], MemAgent [778] Mem0 [88], MemoryBank [874] ExpeL [853], ReasoningBank [462]
Tools, MCP & Skills (§4.3)	Interfaces model decisions with external capabilities and reusable procedures	• Tool interfaces & MCP • Active discovery • Skill libraries	Toolformer [523], MCP [100] AnyTool [140], ToolGen [647] Voyager [630], SkillNet [360]
Orchestration (§4.4)	Decomposes goals, assigns roles, and coordinates work beyond a single loop	• Decomposition • Topologies • Routing & protocols	MetaGPT [219], CAMEL [314] Magentic-One [157], LangGraph [291] GPTSwarm [897], A2A [97]
Hooks & Middleware (§4.5)	Enforces allow, block, or hand-off at predefined points on the action path	• Rule-based hooks • User-defined policies • Adaptive guards	AEGIS [783], IsolateGPT [698] AgentSpec [633], ShieldAgent [80] AdaptiveGuard [754], AGrail [408]
Verification (§4.6)	Scores states and outputs for correctness, safety, and task fit before continuing or accepting results	• Assessment targets • Verification levels • Verifier strategies	SelfCheckGPT [418], VeriGuard [432] PRMs [363], LLM-as-a-Judge [867] Math-Shepherd [643], RLV [521]

4.1.1 Linear Workflows

Linear workflows are the simplest way to keep an agent moving: the harness repeatedly renders the current state to the model, executes the selected action, records the resulting observation, and feeds the updated state into the next call. The workflow follows one evolving trace rather than maintaining a separate plan or search tree. This makes it suitable for interactive tasks where the agent must react to fresh evidence, such as web navigation, debugging, tool use, and exploratory search. ReAct [761] is the canonical instance of this pattern, where reasoning and acting are interleaved and each observation conditions the next model invocation.

The strength of a linear workflow is responsiveness, but its weakness is that all progress depends on a single trace. Early mistakes can contaminate later steps, repeated reasoning can exhaust the context window, and the model may loop, drift, or stop prematurely. Many variants therefore wrap feedback, critique, retrieval, memory, or self-correction around the basic chain [22, 416, 544, 567]. In this sense, a bare linear workflow is only the inner loop, while the emerging practice of *loop engineering* designs the recurring outer system that automatically prompts, monitors, verifies, persists, and resumes the agent. This outer system adds external stop conditions, persistent state, verification gates, cost limits, and recovery policies, turning the workflow into an unattended long-horizon run in which a human need not decide every next prompt.

4.1.2 Plan-Execute Workflows

Plan-execute workflows address the short-sightedness of purely reactive loops by making task structure explicit. The harness first creates or maintains a plan, then uses it to guide later actions through subgoals, dependencies, progress markers, budgets, and failure conditions. Early forms ask the model to plan before solving, while more structured variants plan tool calls, decompose tasks recursively, or revise the plan during execution [482, 639, 720]. Across these systems, the plan becomes an execution object that the harness can inspect, update, resume, invalidate, or assign to different modules.

This structure is especially useful for long-horizon loops because an unattended agent cannot safely run on a vague instruction such as "improve the project". It needs a goal that can be decomposed, monitored, and verified. A maintained plan provides this control surface: it translates an underspecified objective into concrete checkpoints and failure conditions, while allowing the model to make local decisions under persistent task commitments. Recent long-horizon systems therefore couple planning with execution monitoring, verification, replanning, or learning-based policy improvement [763, 832]. Arbor [262] extends this paradigm to autonomous research by maintaining a persistent hypothesis tree across coordinator–executor cycles, where each branch

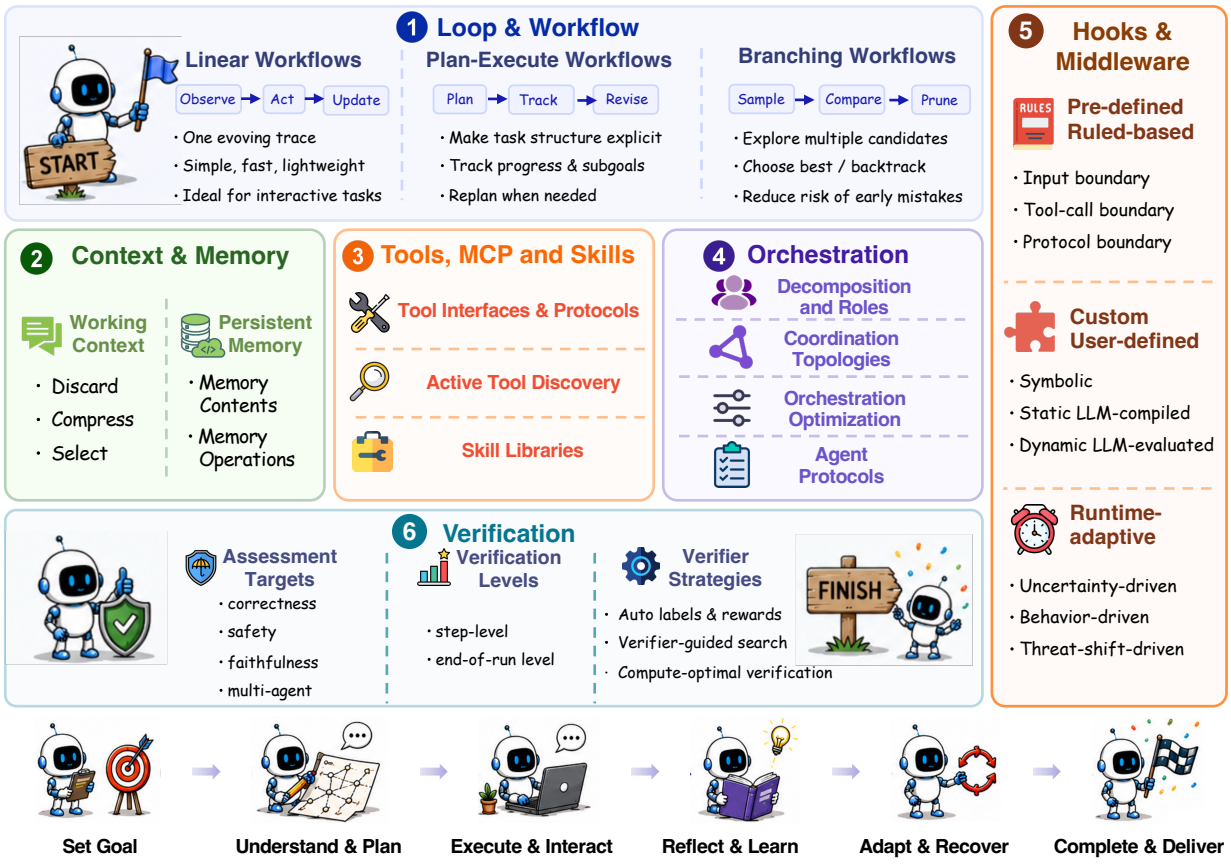


Figure 7 Overview of the agent harness. Six core harness components support the end-to-end agent lifecycle, from goal formulation and planning to execution, adaptation, verification, and delivery, enabling coherent long-horizon execution across interdependent steps.

records hypotheses, experimental artifacts, evidence, and distilled insights that propagate across iterations, turning isolated attempts into a cumulative strategy-driven search.

4.1.3 Branching Workflows

Branching workflows are useful when one wrong commitment can derail the whole task. Instead of immediately following the first plausible action, plan, or answer, the harness generates several alternatives and evaluates them before deciding which one to pursue. This pattern appears in reasoning trees, tree-search-based acting, graph-based thought, code-generation search, and memory-augmented search variants [33, 91, 760, 875]. The defining feature is evaluation before commitment: the harness controls the branching factor, scoring function, pruning rule, backtracking policy, and final selection procedure [319].

For long-horizon agents, branching reduces the risk of spending a large trajectory budget on a weak early decision [280]. It is also closely related to loop engineering, where unattended loops need reliable verification rather than repeated self-approval. A branching workflow can separate generation and checking: one branch proposes a patch or plan, another critiques it, a verifier tests constraints, and the harness decides whether to continue, revise, or stop. This maker-checker separation helps reduce self-evaluation bias and Goodhart-style failures, where the agent satisfies the visible completion condition while violating the user's underlying intent.

Branching is most useful when partial states can be evaluated cheaply and meaningfully. It is less attractive when feedback is weak, delayed, or expensive, because search can multiply token and tool costs, create incompatible partial worlds, and make provenance harder to audit [515]. Thus, branching workflows are best

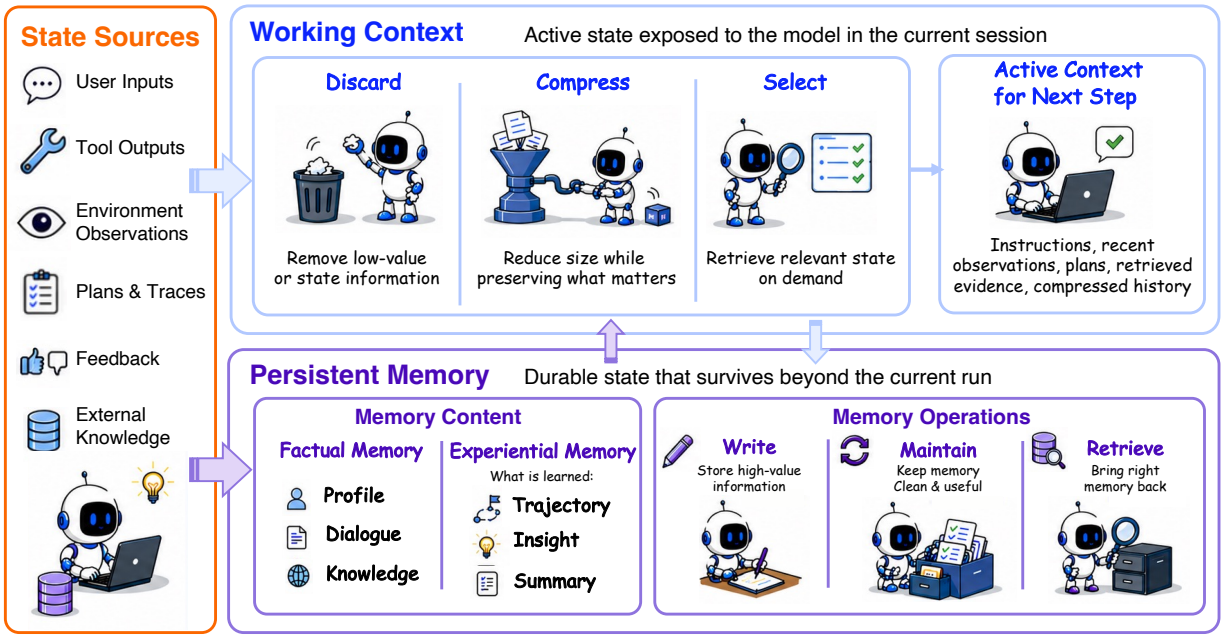


Figure 8 Context and memory in an agent harness. The harness governs the state interface between the workflow and the model: it selects, compresses, or discards active context under a token budget, and reads from and writes to persistent memory that outlives a single window, task, or session.

viewed as selective search mechanisms invoked when the cost of a bad commitment is higher than the cost of evaluating alternatives [343]. Across linear, plan-execute, and branching workflows, the common requirement is that the harness must maintain the state on which these workflows operate. We therefore turn next to the state machinery that supports long-horizon execution.

4.2 Context and Memory

If workflows define how an agent proceeds, context and memory define what state it proceeds with [228]. For long-horizon agents, the harness cannot simply pass along an ever-growing transcript. It must decide what the next model call can see, what should be compressed or discarded when the context window fills, and what information should persist beyond the current window, task, or session [876]. In this sense, context and memory are the state interface between the workflow and the model: the harness reads from prior observations, tool outputs, plans, traces and stored knowledge, then writes the selected state back into the next step.

We distinguish context and memory by their time horizon and role at runtime. *Context* is the active state exposed to the model in the current run, including instructions, recent observations, intermediate reasoning, retrieved evidence, and compressed history. *Memory* is the durable state that survives beyond the immediate window and can be retrieved or updated over time. We organize this component accordingly. Section 4.2.1 covers working state, namely the in-window or near-window state maintained within a run; Section 4.2.2 covers persistent state, namely the durable memory that survives across windows, tasks, and sessions. In Table 3, we list representative works for each sub-section. The overview of this section is shown in Figure 8.

4.2.1 Working Context

Every step of an agent run produces new state: model outputs, tool returns, environment observations, partial plans, and intermediate artifacts. Runtime context maintenance is the harness process that decides which of these states should condition the next model call. Some work refers to this in-window or near-window state as the agent's *working memory*, emphasizing its role as a limited workspace for keeping task-relevant information available within a single episode [228]. For long-horizon agents, the central challenge is not only that the context window is finite, but that useful, stale, and distracting information accumulate together over time.

Too much context can degrade performance through *context rot* and *lost-in-the-middle* effects, while too little context removes constraints, evidence, or partial progress that the agent still needs [378, 511, 541]. The harness must therefore turn an expanding trajectory into a bounded, high-signal workspace. In practice, long-running agents combine three operations: *discard*, *compress*, and *select* [13, 14, 293].

Context Discard. Discarding removes information from the active window [113]. Common policies include resetting context at task boundaries, dropping bulky tool responses after they have been parsed or summarized, and keeping only the most recent turns [597]. For short interactions, discard mainly saves tokens; for long-horizon runs, it is also a reliability mechanism. Stale observations, obsolete plans, verbose logs, and irrelevant tool outputs can continue to bias later decisions if they remain visible.

Discard can also happen before information enters the trajectory. Tool or schema pruning, for example, prevents rarely useful capabilities or verbose specifications from occupying the prompt in the first place [27, 622]. This matters because noise compounds over long runs: a small amount of irrelevant state repeated across many steps can become a first-order source of drift. Discard is therefore the most direct harness-level way to enforce forgetting, rather than relying on the model to ignore low-value context.

Context Compression. Compression preserves task-relevant information while reducing its token and attention footprint. A useful operational distinction is between compressing the current payload and compressing the accumulated trajectory. The first case handles large immediate inputs, such as retrieved passages, tool outputs, web pages, logs, or DOM trees [57, 254, 773]. In agent harnesses, similar operations selectively rewrite verbose observations into compact, task-oriented state descriptions.

The second case is trajectory consolidation: replacing a growing interaction history with a maintained state. This state may be a summary, a list of open constraints, a table of facts, a plan with completion markers, or a graph of entities and dependencies. For long-horizon agents, the goal is not just to shorten the transcript, but to preserve what still matters for future action: completed decisions, remaining subgoals, unresolved failures, and produced artifacts. Recent methods explore learned, task-aware, budget-aware, and hierarchical forms of such compression [224, 268, 573, 693, 696, 768, 778, 824, 890]. This also marks a shift from hand-written context rules toward learned policies for when and how to fold state.

Context Selection. Selection keeps richer state outside the active window and retrieves only what is relevant to the current decision. The indexed state may include documents, events, artifacts, plans, tool outputs, or previous observations. Unlike ordinary RAG, selection in an agent harness often retrieves from the agent's own trajectory, not only from an external knowledge base [202, 669]. Typically, CL-bench is designed to target and evaluate a model's long-context comprehension and selection capabilities [133]. The goal is to keep the current context small while preserving access to the evidence trail.

From the harness perspective, selection involves four choices: when to retrieve, how to form the query, which store to search, and how to filter or aggregate the results before injection. Retrieving too often increases cost and noise; retrieving too rarely risks losing dependencies that were established many steps earlier. Long-running agents therefore usually combine selection with discard and compression: bulky outputs are removed, completed subtasks are folded, and relevant past state is retrieved on demand [13, 230, 231, 824, 846]. These operations keep the active context bounded without forcing the agent to forget the full task history.

4.2.2 Persistent Memory

Working context keeps a single run usable; persistent memory lets the harness carry state beyond the current window, trajectory, or session [228, 583, 694, 848]. This is crucial for long-horizon agents because useful information often outlives the step in which it was produced. A failed attempt may become a lesson for later recovery, a completed subtask may define constraints for downstream actions, and a discovered procedure may be reused in future tasks. Without persistent memory, the agent must repeatedly rediscover the same facts, strategies, and skills as the context window turns over. We use a simple harness-oriented view. *Memory Contents* asks what persistent state is stored; *Memory Operations* asks how that state is written, maintained, retrieved, and evaluated.

Table 3 Representative context and memory systems in the agent harness. Rows follow the split of this subsection: *working context* (Section 4.2.1) is bounded within a single run, while *persistent memory* (Section 4.2.2) survives across windows, tasks, and sessions. **Type** names what the method does with state. **State form** is the representation it maintains. **Train-free** (✔) marks methods that plug into a frozen model as pure harness machinery, versus (✘) those whose policy must be trained or RL-tuned. **Trigger** is when the operation fires. **Key mechanism** briefly describes the core operation.

Method	Type	State form	Train-free	Trigger	Key mechanism
Working Context					
Effective Harnesses [14]	Discard	Turns	✔	Boundary	Reset context, drop parsed tool output
Deep Agents [293]	Discard	Files	✔	Boundary	Offload state to files
Tool pruning [622]	Discard	Schemas	✔	Start-of-run	Prune tools/schemas before prompt
LLMLingua [254]	Compress	Summary	✘	Per-payload	Token-level prompt compression
CompAct [773]	Compress	Summary	✘	Per-payload	Actively compress retrieved docs
ReSum [693]	Compress	Summary	✔	Boundary	Learned trajectory summarization
MEM1 [890]	Compress	Summary	✘	Per-step	RL fuses memory and reasoning
MemAgent [778]	Compress	Summary	✘	Per-step	RL over fixed-size memory
ACON [268]	Compress	Summary	✔	Boundary	Optimizes context-compression policy
ContextBudget [696]	Compress	Summary	✘	Per-step	Budget-aware context allocation
HiAgent [224]	Compress	Plan	✔	Boundary	Subgoal-scoped working memory
Context-Folding [573]	Compress	Summary	✘	Per-step	Folds history under a budget
AgentFold [768]	Compress	Summary	✘	Per-step	Proactive history folding
ACE [824]	Select	NL-note	✔	On-demand	Retrieves, evolves context entries
LEGOMem [202]	Select	Summary	✔	On-demand	Modular per-subtask procedural memory
Memex(RL) [669]	Select	Vector	✘	On-demand	RL over indexed experience memory
Memory-as-Action [846]	Mixed	Summary	✘	Per-step	Memory read/write as actions
Persistent Memory					
CLAUDE.md [10]	Factual	NL-note	✔	Start-of-run	Project-rule file loaded per run
Cursor Rules [104]	Factual	NL-note	✔	Start-of-run	Persistent project rules per run
AGENTS.md [98]	Factual	NL-note	✔	Start-of-run	Portable project-knowledge spec
MemoryBank [874]	Factual	Vector	✔	On-demand	Store with forgetting-curve updates
Mem0 [88]	Factual	Graph	✔	On-demand	Linked long-term fact store
MemoryOS [267]	Factual	Mixed	✔	On-demand	OS-style tiered memory
HippoRAG [199]	Factual	Graph	✔	On-demand	Graph memory for associative recall
A-MEM [731]	Factual	Graph	✔	On-demand	Self-organizing memory notes
Reflexion [544]	Experiential	NL-note	✔	After-failure	Verbal lessons from failures
ExpeL [853]	Experiential	NL-note	✔	On-demand	Reuses insights from trajectories
Synapse [868]	Experiential	Summary	✔	On-demand	Trajectory-as-exemplar prompting
SAGE [359]	Experiential	NL-note	✔	On-demand	Reflective memory-augmented evolution
Buffer of Thoughts [750]	Experiential	NL-note	✔	On-demand	Reusable thought templates
ReasoningBank [462]	Experiential	NL-note	✔	On-demand	Distills reusable reasoning strategies
AWM [674]	Experiential	Code-skill	✔	Boundary	Induces workflows from traces
Agent KB [587]	Experiential	NL-note	✔	On-demand	Cross-domain experience knowledge base
Voyager [630]	Experiential	Code-skill	✔	On-demand	Reusable executable skill library
MemGPT [464]	Mixed	Mixed	✔	On-demand	OS-style context/memory paging
Generative Agents [470]	Mixed	Summary	✔	On-demand	Memory stream with reflection

Memory Contents. Persistent memory mainly stores two kinds of state. *Factual memory* stores declarative information that the agent should not have to rediscover: user preferences, project rules, commitments, environment state, document or codebase facts, and the capabilities of external systems [228]. For a long-horizon harness, these facts support continuity: user-level facts keep behavior consistent across sessions, while environment-level facts prevent the agent from repeatedly re-inferring stable constraints. Practical examples include project-rule files such as `CLAUDE.md`, Cursor `Rules`, and `AGENTS.md`, which provide stable project knowledge outside the immediate prompt [10, 98, 104]. User memory systems and graph-structured stores further extend this idea by maintaining profiles, preferences, and linked factual records that can be consolidated and audited over time [88, 199, 229, 267, 507, 731, 874].

Experiential memory stores what the agent has learned from prior trajectories. It may contain failure lessons, successful cases, state-action exemplars, reusable reasoning patterns, plans, workflows, or executable procedures. The common purpose is to reduce long-horizon cost by replacing repeated rediscovery with retrieval of prior structure. Some systems store verbal reflections or lessons from past failures; others maintain experience pools, reusable reasoning templates, workflow memories, or cross-domain solution paths [44, 262, 359, 462, 544, 587, 674, 750, 807, 853, 868]. At the most operational end, experiential memory can be distilled into skills, scripts, API wrappers, or tool-like procedures, as in skill-library agents such as Voyager [446, 630]. Once stored in this form, memory begins to overlap with the tool layer: the harness can retrieve, version, validate, reuse, or retire a learned procedure without changing model weights.

Memory Operations. A persistent store is useful only if the harness manages how memory is written, maintained, and read back into context. *Memory writing* decides what should be saved from a long run. Tool outputs, partial plans, reasoning traces, user feedback, observations, and intermediate artifacts are often too noisy to store as-is. The harness therefore extracts higher-value units such as facts, summaries, cases, lessons, workflows, or skills, using summarization, distillation, or structured construction [221, 407, 464, 470, 674]. The key question is what level of experience should survive: an event, a workflow, or an executable procedure.

Memory maintenance keeps the store usable as new information arrives. Over long horizons, memories can become stale, repeated, or inconsistent. The harness must merge related entries, update facts when users or environments change, and remove outdated or low-value material before retrieval becomes noisy. Existing systems use memory streams, graphs, linked notes, or other structured stores to support this process [199, 470, 731]. The model may suggest what to remember, but the harness decides whether a memory is inserted, merged, linked, archived, or forgotten.

Memory retrieval decides how stored state re-enters the active context. Some memories are loaded at the start of a run, such as project rules or user preferences [664]; others are retrieved at subtask boundaries, after failures, before tool use, or when the model asks for them. Retrieval requires choices about the query, the store to search, the ranking or filtering method, and how much content to inject [488, 568, 610, 694]. Too little retrieval makes the agent repeat mistakes or miss long-range dependencies; too much retrieval turns memory back into context overload. Recent work therefore treats memory read/write as a policy problem, learning when to remember, update, or retrieve during inference [778, 846].

Persistent memory is also hard to evaluate. End-task success often mixes memory quality with model strength, tool reliability, and benchmark variance. New benchmarks test long-term recall, cross-session consistency, and tasks that depend on earlier sessions, but recent analyses still find saturation, judge sensitivity, and limited coverage of update cost and memory maintenance [235, 253, 686]. For long-horizon harnesses, the key question is not whether an agent can recall an isolated fact, but whether it can save, update, retrieve, and inject the right memory when it changes the next action. We collect evaluation resources in Section 6.6.

4.3 Tools, MCP, and Skills

Where context and memory settle what the agent *knows*, this component settles what it can *do*. Without tools that return observations, even a capable model can only emit text; tools are what let it query a search engine, run code, or read and write a database. We organize this capability interface in three parts: the tools exposed by the runtime and the protocol layer, most prominently the Model Context Protocol (MCP), that standardizes them (Section 4.3.1); the discovery mechanisms that keep a large tool surface usable (Section 4.3.2); and the skills an agent accumulates on top of them (Section 4.3.3). This interface matters more as the horizon grows,

because extended execution becomes a long chain of interdependent calls whose state must be carried, verified, and recovered across steps.

4.3.1 Tool Interfaces and Protocols

Function Calling. Tool interfaces let a language model act beyond its parametric knowledge. Foundational work established when, what, and how to call: Toolformer self-supervises where API calls should be inserted, Gorilla maps requests onto large API surfaces while suppressing hallucinated calls, and ToolLLM scales instruction-tuned tool use to thousands of real-world REST APIs [471, 492, 523]. Papers consolidate this line into a workflow of task planning, tool selection, tool calling, and response generation within a controller–tools–environment view [497, 724]. Single-call competence is by now well covered by mature benchmarks (BFCL, StableToolBench) and data-synthesis pipelines (APIGen, ToolACE) [198, 385, 392, 472].

Tool Protocols and MCP. The dominant engineering response to a growing, heterogeneous tool surface is protocolization. The Model Context Protocol (MCP) defines a host–client–server lifecycle with capability negotiation, authorization, and tool annotations, turning ad hoc integrations into a uniform, discoverable layer; its ecosystem, architecture, and security are covered in dedicated and broader protocol papers [100, 758]. Because it lets agents compose independently maintained servers and exposes the tool space uniformly, MCP also sets up the scaling techniques of Section 4.3.2. MCP-grounded benchmarks such as MCP-Universe reveal what single-call leaderboards miss: once tasks span many servers and tools in stateful sandboxes, even the strongest models succeed on only a minority of them, with input tokens rising rapidly over steps and failures becoming increasingly cognitive rather than syntactic [31, 350, 411, 700]. Beyond MCP, a broader wave of tool-use benchmarks probes schema discipline, state mutation, conversational tool legality, and messy real-world user behavior, from constrained tool-legality suites such as τ-bench to long-horizon, cross-application, and in-the-wild benchmarks such as Toolathlon [213, 322, 399, 539, 548, 762, 779].

Multi-turn Tool Use. Function calling and MCP standardize the *interface* of an individual call, but tool use at scale is inherently multi-turn: an agent chains many dependent calls, each conditioned on the previous observation. As these sequences grow to tens or hundreds of invocations, the bottleneck migrates from isolated-call correctness to sustaining the whole chain [724], and even strong agents complete only a fraction of realistic tasks, degrading through context saturation and memory decay as the turn count rises [322, 325, 406]. The limiting constraint is increasingly the state and context accumulated across turns rather than the mechanics of any single call. This is a property of the execution process rather than the call interface; the control-flow responses to it, namely decomposing, scheduling, and coordinating many turns, are orchestration, which we treat separately, so this subsection stays on the tool surface itself.

4.3.2 Active Tool Discovery

A fixed, prompt-resident tool list does not scale. Once catalogs reach thousands of tools, their descriptions crowd out the task, token cost tracks registry size rather than step demand, and selection accuracy degrades; the tool-retrieval saturation diagnosed above motivates active discovery [350, 411]. The remedy is to stop placing the entire surface in context and instead let the agent actively discover the few tools each step needs.

Tool Retrieval. The first move is retrieval-based discovery: index tool metadata and inject only the top-ranked schemas at each step. RAG-MCP, AnyTool, and MCP-Zero develop this idea with plain, hierarchical, and agent-initiated retrievers respectively, sharply reducing token use on large toolsets [140, 153, 162]. A complementary, supply-side approach generates MCP servers directly from OpenAPI specifications, and Anthropic's Tool Search Tool demonstrates the same retrieval principle in production, scaling to thousands of tools with large token savings and higher tool-selection accuracy [9, 423]. Pushing this further, DeepAgent makes tool search an action inside a single reasoning process, dynamically discovering tools from registries of tens of thousands as the task unfolds, while ToolGen removes the separate retriever altogether by representing each tool as a vocabulary token the model emits during generation [338, 647].

Programmatic Tool Use. A second move treats the tool surface itself as something the agent can read and program against. CodeAct treats executable code as the action space, letting a single step compose

calls, branch, and loop [656]; in MCP settings, code-execution patterns present servers as importable APIs and coordinate many calls inside one code block, cutting token consumption by orders of magnitude [9, 15], and Code-Execution MCP formalizes the shift from tool-by-tool orchestration to a single model-generated program, keeping context consumption nearly constant in the call count and tool-ecosystem size at the cost of a larger executable attack surface [154].

As coding agents strengthen, tool *selection* itself becomes a workspace activity: rather than picking from a static registry, the agent inspects tool definitions, documentation, or source on demand, using progressive disclosure to keep a large ecosystem cheap in context [730]. The same insight drives agentic search. Direct corpus interaction has a strong agent search a raw corpus with terminal tools (grep, find, read) instead of a top-k retriever, beating sparse, dense, and reranking baselines [355], and agentic keyword search with only grep-style tools matches vector-database RAG [566]. Applied to code, a terminal-only RL-trained agent localizes code as well as far larger models [578], turning discovery from a static input choice into an agent-driven search. Together, code execution and workspace-mediated discovery bound a tool surface that would otherwise grow with every call.

4.3.3 Skill Libraries

Once tools and workspace state are in place, the next layer of externalized capability is the *skill*: a reusable, parameterizable procedure that packages a validated combination of tools, control flow, and domain knowledge. Recent papers cast this as the step from episodic tool use to persistent, colleague-like agents under a "Workspace + Skill" paradigm, where a persistent workspace supplies state and a curated skill library supplies reusable operational knowledge [841]. We unpack it along three axes: what a skill is, how skills are acquired and reused, and how the library evolves.

Procedural Skills. On top of tool interfaces, the *skill* abstraction packages validated combinations of tools, control flow, and domain knowledge into named, reusable procedures. Rather than re-deriving a tool sequence each time, the agent recalls a procedure that already works, shifting the interface from tools to skills [888]. Papers frame skills as procedural artifacts with a representation, acquisition, retrieval, and evolution lifecycle, and as a layer orthogonal to MCP: where MCP supplies connectivity, skills supply the procedural knowledge for interpreting outputs and recovering from failure. The current landscape converges on three representation families: natural-language or Markdown procedures that condition a frozen agent, realized as a progressively disclosed file tree by Anthropic's Agent Skills [16]; executable code the agent can call, inspect, and edit, as in Voyager [630]; and structured relational libraries, where SkillNet weaves over 200,000 skills into a typed skill graph (depend_on, compose_with, ...) [360]. The shared move is from flat, similarity-retrieved entries toward curated, relationally organized procedures owned by the harness as first-class assets.

Skill Acquisition. Work on skill acquisition and reuse shows the payoff. Voyager, the canonical instance, discovers executable Minecraft skills and reloads them on later tasks, and later systems extend the idea to inference-time tool synthesis, cumulative scientific-reasoning skills, and skills for unfamiliar software [4, 237, 400, 577, 630]. Related systems evaluate higher-order tool-composition skills and convert episodic narratives into executable skills with explicit activation and termination conditions [65, 430]. Across these systems the recurring benefit is lower token cost, higher reliability, and shorter effective horizons through reuse.

Skill Library Evolution. What distinguishes a skill from a tool is that the library is a first-class object the harness owns and curates: skills can be versioned, retrieved, refined, retired, and reused, so where tools supply flexibility, skills supply efficiency and reliability [340]. Because the library lives outside the weights, it can evolve without gradient updates: Memento treats skill acquisition as memory-based reinforcement learning, Alita generates and reuses MCP servers on demand, and observability-driven harness evolution finds most of its gains in rewriting the tool, middleware, and memory layers rather than the system prompt [456, 495, 881]. A first line optimizes the skill text directly: SkillOpt trains a frozen agent by editing its natural-language skill document, accepting a change only when it improves a held-out validation score [757]. A second structures the library as a graph rather than a flat list: the prerequisite-skill graph originates in data-efficient training (Skill-it) [63], and is imported into agent libraries by SkillGraph, which co-evolves a typed skill graph with the policy via reinforcement learning [341], by GraSP, which compiles retrieved skills into an executable

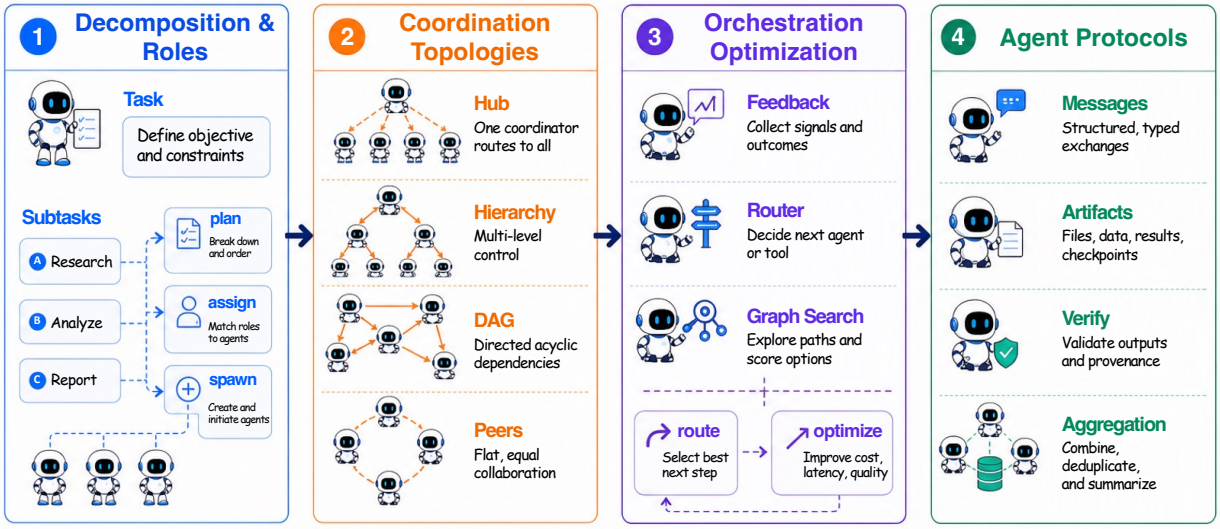


Figure 9 Overview of agent orchestration. Tasks are decomposed into roles, coordinated through different topologies, optimized via feedback-driven routing, and integrated through communication and aggregation protocols.

DAG with locality-bounded repair [707]. A growing line couples skill libraries directly with reinforcement learning: Skill0, for instance, trains with an in-context curriculum that progressively withdraws skill files until the policy runs skill-free, beginning to fold the library into the weights themselves [405]. This library-level self-improvement is the externalized form of evolution; its internalized counterpart is taken up in Section 5.

4.4 Orchestration

A single agent loop can attend to only one thing at a time; many long-horizon tasks are better served by composing several loops, whether as subtasks of one agent or as a team of specialized agents. Multi-agent collaboration can be described along a few dimensions: the participating agents, the interaction type (cooperation, competition, or a mix), the communication structure, the coordination strategy, and whether the arrangement is static or dynamic [197, 611]. We adopt a pragmatic version of this taxonomy and ask four questions: how work is decomposed and assigned, what topology the resulting computation takes, whether that topology is fixed or learned, and how the agents communicate. Orchestration is the externalized form of long-horizon control structure [468]: once a task outgrows what a single loop can sustain, how the work is split and coordinated often affects end-to-end success more than swapping the underlying model does. Composition is not free, however. A systematic study of multi-agent failures attributes most breakdowns to under-specified roles, inter-agent misalignment, and missing verification rather than to weak base models, a reminder that orchestration must be designed rather than merely assembled [48]. The overall framework of agent orchestration is shown in Figure 9.

4.4.1 Decomposition and Roles

The first decision is how a long task is split into units and assigned to roles. The earliest harnesses fix this split statically, hard-coding the roles and their interaction pattern, whether as software-team standard operating procedures, planner–executor role-play, staged role-conditioned dialogues, or a general programmable substrate in which roles and their conversations are first-class, as in frameworks such as software-team SOPs [219], role-play collaboration [314], staged dialogues [485], and a programmable substrate [692]. A predefined subtask graph is simple but brittle, since an early failure propagates downstream and a hand-coded role set scales poorly; this motivates dynamic decomposition that revises the split from feedback and can spawn a subtask-tailored subagent on the fly, as in TDAG [660]. Either way the harness must keep the split well-formed: agent-oriented planning, for example, holds the decomposing meta-agent to solvability, completeness, and non-redundancy, re-planning unresolvable or duplicated subtasks before any agent runs [305]. How fine to

decompose is then a granularity trade-off: too coarse and a subtask overwhelms its agent, too fine and coordination overhead dominates, and that overhead grows with the horizon. Much of the benefit usually attributed to role personas in fact comes from the structural decomposition they induce, namely parallelism, context isolation, and tool specialization, rather than from the personas themselves [48].

4.4.2 Coordination Topologies

Once units and roles are set, the way agents are wired together fixes the shape of the computation. These topologies fall into centralized forms built around a hub or orchestrator, decentralized or distributed forms such as peer debate and dynamic graphs, and hierarchical forms [611]. Centralized designs, exemplified by Magentic-One, place a hub or orchestrator that drives a team, in some variants stacking layers that successively refine one another or dynamically recruiting collaborators [69, 157, 638]. Decentralized designs instead let peers select and prune one another at inference time or scale collaboration over large network structures, as in DyLAN and MacNet [390, 486]. The dominant programmable abstraction across both is the directed acyclic graph: LangGraph defines workflows as DAGs whose nodes are agents or functions and whose edges carry typed state, while Agint and Flash-Searcher compile instructions into typed DAGs and recast web research as a parallel DAG to cut step counts [90, 491]. On long-horizon coding tasks this design space can matter more than the choice of model: on SWE-bench Verified, a fixed frontier model varies far more across scaffolds than different models do under a single fixed scaffold [468].

The wirings above coordinate agents toward a shared objective, but a complementary family makes the interaction itself *competitive*, pitting two roles in a GAN-like loop where one proposes ever-harder problems and the other strives to solve them, so the pair co-evolves an automatic curriculum with no external data. R-Zero instantiates a Challenger rewarded for posing tasks at the edge of a Solver's competence and a Solver rewarded for cracking them, while Absolute Zero folds both roles into a single model that proposes and solves verifiable tasks in one self-play loop [233, 854]; earlier, SPAG showed that setting a model against a copy of itself in a two-player adversarial language game steadily sharpens reasoning [84]. Because this adversarial coordination improves the participants through training rather than routing a fixed roster of models, its payoff is ultimately internalized (Section 5); yet the arrangement itself remains a distinctly competitive orchestration pattern, the counterpart to the cooperative composition that dominates the rest of this section.

4.4.3 Orchestration Optimization

A topology can be hand-designed or searched. AFlow treats workflow design as program-space search with Monte-Carlo tree search, and GPTSwarm represents a multi-agent system as an optimizable graph and learns both node prompts and edges [816, 897]. A related line learns who executes each unit. At the model level, routers learn to select among models under performance, cost, latency, and congestion constraints [447, 810, 880]; at the agent level, MasRouter jointly chooses collaboration mode, roles, and the model per role, while other systems spawn subagents on the fly or pair central planning with a tool–environment–agent protocol [519, 791, 828]. Taken to its limit, the coordination structure itself is allowed to emerge: a central orchestrator can be trained with reinforcement learning to induce compact coordination at lower cost, and related systems refine functionality and coordination from scratch or couple shared memory with asynchronous execution [107, 496, 836]; at scale, mixed self-organizing coordination can even outperform hand-designed hierarchies [127]. Like skill-library growth (Section 4.3.3), these black-box optimizations of the orchestration graph are an externalized form of evolution.

4.4.4 Agent Protocols

Finally, composed agents must exchange messages reliably. The A2A protocol foregrounds Agent Cards, Tasks, and Artifacts over JSON-RPC with streaming, IBM's Agent Communication Protocol offers a lighter REST-native alternative, and MCP, ACP, A2A, and ANP have been compared across architecture, discovery, communication, and security [97, 144, 512, 758]. Framework-level messaging conventions coexist with richer schemes that add decentralized identity and verifiable credentials, switch between structured protocols and natural language according to communication frequency, or perform decentralized discovery over encrypted peer-to-peer channels [219, 314, 422, 501, 504, 692]. A complementary harness duty is aggregation: when parallel trials or subagents run, their outputs must be reduced to a single result, through best-of-N or

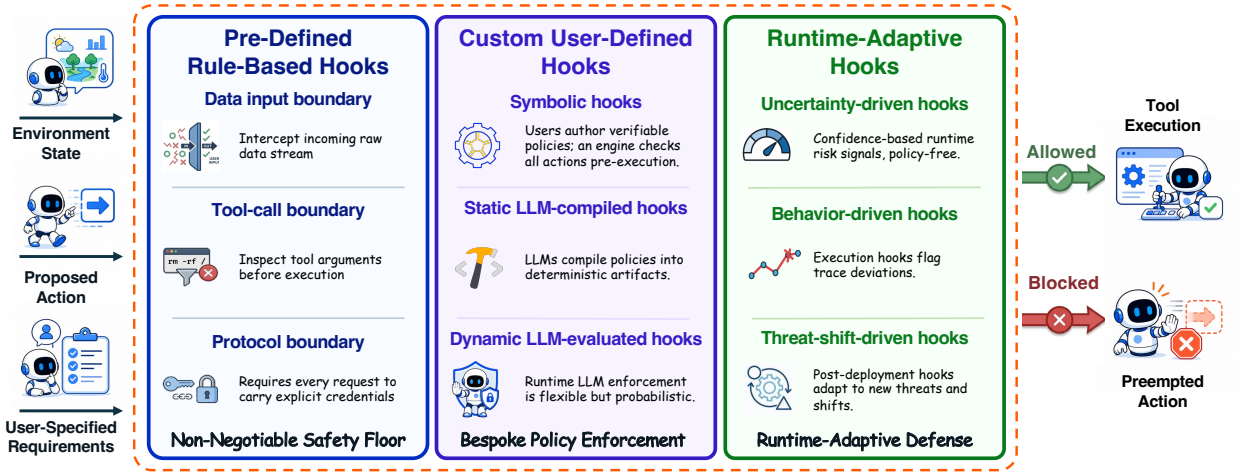


Figure 10 Overview of hooks and middleware. We categorize hooks into three paradigms based on the origin and adaptiveness of their safety criteria: pre-defined rules, user-defined policies, and runtime-adaptive signals.

voting for verifiable tasks [323, 442, 659] and through confidence-aware agentic aggregation for open-ended ones [296, 298]. This shift from prescribed hierarchies toward protocol-mediated coordination stands in productive tension with the deterministic governance we turn to next, and remains one of open problems.

4.5 Hooks and Middleware

The preceding subsections establish the full execution infrastructure of an agent harness: loops govern control flow, context and memory manage information state, tools extend capabilities, and orchestration composes agents into collaborative systems. Yet the very expressiveness that makes these mechanisms powerful also amplifies the risk surface: an agent equipped with file-system access, API credentials, and multi-step autonomy can cause irreversible harm through hallucinated commands or adversarial jailbreaks [1]. Prompt-level instructions alone are insufficient to constrain such behavior, as they remain vulnerable to the same failure modes they aim to prevent. Recent architectures [421, 529] have therefore shifted toward programmable boundaries enforced at the harness level, which we collectively term *hooks*. Hooks perform enforcement, gating the action path with allow, block, or hand-off decisions, whether through rule-based evaluation or model judgment. As illustrated in Figure 10, based on who defines the safety criterion and how the trigger condition is determined, we identify three complementary paradigms: (1) pre-defined rule-based hooks enforce universal safety invariants hard-coded by framework designers; (2) custom user-defined hooks enforce domain-specific policies authored by deployers; and (3) runtime-adaptive hooks derive their trigger conditions from runtime signals such as model uncertainty and behavioral patterns rather than from any predetermined specification.

4.5.1 Pre-defined Rule-based Hooks

At the foundation of agent safety, harnesses employ immutable, hard-coded middleware to intercept or block actions that violate universal safety invariants without requiring user configuration. These hooks are triggered at fixed lifecycle points such as immediately before a shell command or API call, and because they operate on the action path rather than the reasoning path, they run outside the LLM's context window and are therefore less directly exposed to prompt-level manipulation; their coverage, however, still depends on policy completeness and adversarial robustness, since encoded payloads, obfuscated arguments, or gaps in the policy can still slip past them. Concretely, pre-defined hooks manifest at three successive boundaries along the agent execution pipeline:

- **Input Boundary.** Operating before data ingestion by the core language model, this mechanism intercepts incoming raw data streams, including user instructions, prompt templates, and retrieved external context. Its primary objective is to filter out adversarial prompt injections, jailbreaks, and malformed

system directives designed to compromise the model's behavioral alignment. Systems implementing this boundary [529] deploy independent, token-level or semantic static filters to reject non-compliant inputs before they can influence the model's internal attention state.

- **Tool-call Boundary.** Acting as a pre-execution firewall, this mechanism inspects tool arguments before execution to detect dangerous options (e.g., `rm -rf`) and common attack patterns such as code injection or unauthorized file access. AEGIS [783] instantiates this design by interposing a policy engine on every tool call, validating arguments against a curated set of known vulnerability patterns, including path traversal, with negligible latency overhead.
- **Protocol Boundary.** Enforced at the inter-component communication layer, this boundary requires every request to carry explicit credentials and receive per-action authorization, ensuring no message is implicitly trusted regardless of its origin. Representative systems include Authenticated Workflows [503], which attaches cryptographic tokens to protocol messages for authenticity verification, and MagenticUI [441], which mandates explicit user approval before dispatching agent actions.

These three boundaries form independent, complementary defense layers: because each targets a distinct attack surface, the failure of any single layer does not compromise the others [421]. This defense-in-depth property is further strengthened by architectural isolation techniques such as IsolateGPT [698], whose hub-and-spoke design routes all tool interactions through an isolated mediator to prevent cross-tool information leakage. Together, these pre-defined mechanisms establish a non-negotiable safety floor that holds irrespective of the model's reasoning quality, the user's configuration choices, or the particular task being executed.

4.5.2 Custom User-defined Hooks

Beyond immutable rules, modern harnesses expose programmable interfaces that allow developers to define domain-specific policies enforced at runtime. While pre-defined hooks target universally dangerous operations, custom hooks address the far broader space of application-specific requirements: **workflow constraints, regulatory compliance, access control scoping, and safety boundaries that vary across deployment contexts.** The defining characteristic is the locus of policy authorship: in every approach below, it is the user or developer, not the framework designer, who specifies the substantive safety criterion. "User-defined" thus refers to who writes the policy, not how it is enforced; the enforcement mechanism ranges from deterministic symbolic engines to probabilistic model evaluators, but the policy content originates with the deployer in all cases.

Existing approaches vary along two orthogonal dimensions: (i) how the user specifies a policy (formal language vs. natural language), and (ii) what enforces it at runtime (a deterministic symbolic engine vs. a probabilistic model). The interplay of these two dimensions yields three distinct categories. Formal-language policies pair naturally with symbolic enforcement. Natural-language policies split by whether an LLM translates them into executable artifacts offline and exits the enforcement path, or remains on the critical path to evaluate compliance at every step. The resulting spectrum trades specification effort against enforcement strength.

(1) Symbolic Hooks. Users author policies in a formal, machine-verifiable language, and a deterministic engine evaluates every action against the policy before execution. Because the LLM is entirely absent from the enforcement path, compliance guarantees are hard: a compliant action is provably within policy, and no prompt injection can subvert the check.

Several systems design event-triggered DSLs whose predicate guards and enforcement actions are executed by lightweight symbolic runtimes [509, 633]. Others go further by translating temporal-logic or JSON-schema specifications into first-order constraints and using SMT solvers or constrained token generation to make non-compliant tool calls physically impossible to emit [265, 542]. For multi-agent settings, PCAS [466] models inter-agent state as a causal dependency graph and enforces Datalog-derived policies via Differential Datalog with a reference monitor intercepting every action. A complementary line draws on operating-system security primitives such as capability labels and cryptographic audit records to enforce information-flow control and provenance tracking at the tool-call boundary [109, 617]. Across all variants, the enforcement engine is a formally deterministic program whose correctness can be audited independently of the LLM it guards.

(2) Static LLM-compiled Hooks. A second paradigm accepts policies written in natural language or semi-structured documents but compiles them into executable artifacts that are then enforced deterministically

or quasi-deterministically. The LLM participates at compile time but is absent from the enforcement path, yielding guarantees bounded by compilation quality but independent of runtime model reasoning.

The compilation targets range from executable guardrail code that runs deterministically to admit or block each action [708], to probabilistic rule circuits derived from regulatory documents and verified by formal model checkers [80], to formally verified policy code generated from Hoare-triple specifications and validated by an offline verifier before deployment [432]. The common thread is a two-phase architecture: an LLM-powered compilation step that transforms human-readable intent into machine-executable policy, followed by a symbolic or quasi-symbolic enforcement step. Errors can be introduced at compile time, but once compiled, enforcement proceeds without further LLM involvement.

(3) Dynamic LLM-evaluated Hooks. A third paradigm also starts from user-authored natural-language policies but, unlike static compilation, keeps the LLM on the enforcement critical path: the model directly evaluates each action or trajectory against the user-supplied policy at runtime. The deployer must provide domain-specific safety criteria without which the system has nothing to enforce, so the "user-defined" property holds in the same sense as the preceding categories. This maximizes expressiveness and minimizes specification effort at the cost of replacing hard guarantees with probabilistic ones.

One approach deploys a guardian LLM that interprets natural-language policies with contextual reasoning across multiple control points [285]. A classifier-based alternative fine-tunes or prompts language models as configurable safety classifiers whose risk taxonomies or policy definitions can be adapted without retraining [244, 274]. A third variant composes multiple heterogeneous scanners into a layered pipeline that users can extend with custom scanners tailored to their threat model [87]. Because the enforcer is itself a language model, these approaches can handle nuanced, context-dependent policies that resist formalization, but compliance is probabilistic and susceptible to the same failure modes that motivate guardrails in the first place.

Collectively, these three categories trace a design-space frontier between specification cost and enforcement strength: symbolic hooks demand the most expertise to author but provide the hardest guarantees; LLM-compiled hooks offer a middle ground by lowering the authoring barrier while keeping the LLM off the enforcement path; and LLM-evaluated hooks are the most accessible but the least formally bounded, trading deterministic compliance for the ability to express nuanced, context-dependent policies. In practice, production deployments increasingly layer multiple paradigms (e.g., symbolic access-control checks composed with model-based content classifiers) to exploit the complementary strengths of each.

4.5.3 Runtime-adaptive Hooks

While rule-based and user-defined hooks operate on predetermined conditions, a third, runtime-adaptive paradigm derives dynamic, context-sensitive triggers from runtime signals rather than fixed rules. The defining characteristic is that neither the trigger condition nor the enforcement threshold is fully specified before deployment; instead, these hooks continuously monitor the agent's internal confidence, task progress, and external behavioral patterns in real time, activating timely protective measures when the system detects that the agent is operating outside its reliable competence range. We distinguish runtime-adaptive hooks from user-defined hooks by the locus of criterion definition: user-defined hooks enforce human-authored policies, whereas runtime-adaptive hooks let the system itself discover what to check from various data.

Existing approaches can be organized by the type of runtime signal to which the hook responds: the model's own uncertainty, deviations in the agent's behavioral patterns, or shifts in the external threat distribution.

(1) Uncertainty-driven Hooks. The most direct form of runtime adaptation monitors the model's own confidence in its reasoning. These hooks extract calibrated risk scores from the inference process and surface them to downstream systems for intervention decisions, without themselves enforcing any fixed policy. Representative methods estimate semantic entropy over sampled completions to detect confabulations [151], propagate verbalized confidence scores across reasoning steps and trigger targeted recomputation when accumulated uncertainty exceeds a threshold [815], or model situational weights via hidden Markov models to extend uncertainty propagation to multi-step agentic reasoning [858]. Because these signals are derived from the model's internal state at each inference step, the effective trigger condition adapts continuously without requiring any external training data or human-authored rules.

(2) Behavior-driven Hooks. A second class of hooks monitors the agent's observable execution behavior and intervenes when action patterns deviate from learned expectations. Unlike uncertainty-driven hooks that read internal model states, these approaches build structural models of normal agent behavior from execution traces and use them to detect anomalies at runtime. One line of work learns formal behavioral models from historical traces and applies model checking to anticipate violations before they occur, enabling proactive intervention [634]. Another line operates at the trajectory level, training compact verifiers on synthetically perturbed trajectories for step-level error localization and targeted rollback [388], or constructing dynamic execution graphs from distributed traces to detect structural irregularities such as infinite loops, prompt injections, and collusive delegation across multi-agent interactions [214]. The common thread is that the definition of "normal behavior" is learned from data rather than specified by a human, making these hooks adaptive to the particular agent and task at hand.

(3) Threat-shift-driven Hooks. The preceding two categories respond to signals that arise within a fixed threat model; a third category addresses the scenario where the threat landscape itself evolves after deployment. These hooks detect that previously unseen attack types or domain shifts have rendered the current safety model inadequate, and update their own detection criteria accordingly. AdaptiveGuard [754] identifies novel attack types as out-of-distribution inputs on guardrail-model activations and rapidly adapts via continual learning, maintaining high detection performance where static guardrails degrade sharply. AGrail [408] takes a lifelong-learning approach: two cooperative LLMs generate and iteratively refine safety checks per action type via test-time adaptation, accumulating them in a memory module that generalizes across tasks. DuoGuard [119] employs adversarial co-evolution between a generator and a classifier during training to produce a robust safety classifier; notably, its adaptiveness is a training-time property and the deployed model is static, placing it at the boundary between this category and fixed guardrail classifiers.

Benchmarks such as Agent-SafetyBench [849] and ToolEmu [520], which reveal that even frontier agents struggle to achieve satisfactory safety scores, underscore the necessity of these signal-driven mechanisms. By deriving their trigger conditions from learned models rather than human-authored rules, runtime-adaptive hooks complement user-defined hooks: the former discovers what to watch for, while the latter enforces what the developer already knows.

4.6 Verification

Hooks establish deterministic boundaries that intercept or block actions, but interception alone is not sufficient: the system must also *evaluate* the quality, correctness, and safety of agent behavior at every stage. Verification mechanisms serve this complementary role, acting as the harness's assessment engine that judges whether an intercepted action should proceed, whether intermediate reasoning is on track, and whether final outputs meet task requirements. Together with hooks, they close the governance loop by shifting the harness from a passive execution wrapper to an active controller. Table 4 provides a structured overview of representative verification and evaluation methods discussed in this subsection, organized by assessment target, verification level, and verifier strategy.

4.6.1 Assessment Targets

Verification mechanisms can be taxonomized by the property they certify: the factual correctness of outputs, the operational safety of actions, faithfulness to the user's instructions, or agreement across verifiers.

(1) Correctness Verification. Correctness verification certifies the factual accuracy of agent outputs. SelfCheck-GPT [418] detects hallucinations through sampling consistency alone, providing a zero-resource baseline for factual self-assessment without any external knowledge source.

(2) Safety Verification. Safety verification certifies that agent actions do not cause harm before they take effect. VeriGuard [432] generates both functional code and formal verification artifacts for each action, achieving near-zero attack success. A rapidly growing line builds dedicated safety verifiers and diagnostic guardrails: AgentDoG [287] organizes moderation around a three-dimensional risk taxonomy (source, failure mode, consequence) for fine-grained, trajectory-level judgments with root-cause diagnosis beyond binary labels, and its successor AgentDoG 1.5 [288] reports strong moderation from a small amount of training data while supporting real-time online deployment. ToolSafe [440] targets tool-invocation safety with a step-level guardrail

Table 4 Representative verification and evaluation methods for LLM agents. Rows follow the three dimensions of this subsection: assessment targets, verification levels, and verifier strategies. **Train-free** (✔) marks methods that work with a frozen model, versus (✘) those requiring dedicated training.

Method	Train-free	Key mechanism
Assessment Targets		
SelfCheckGPT [418]	✔	Sampling consistency for factuality verification
VeriGuard [432]	✔	Verified code generation for action safety
ToolSafe [440]	✘	RL-trained proactive tool-invocation safety guardrail
NSVIF [563]	✔	Neuro-symbolic constraint checking for instruction faithfulness
Multiagent Debate [139]	✔	Inter-model deliberation for factual agreement
Multi-Agent Verification [362]	✔	Independent aspect verifiers for cross-verifier agreement
Verification Levels		
Let's Verify Step by Step [363]	✘	Human-supervised process rewards for intermediate steps
AgentPRM [702]	✘	Per-decision process rewards for agent trajectories
CRITIC [182]	✔	Tool-interactive stepwise verify-then-correct loop
LLM-as-a-Judge [867]	✔	Model-based evaluation of final outputs
Agent-as-a-Judge [898]	✔	Tool-augmented assessment after task completion
Reflexion [544]	✔	End-of-trial verbal reflection with episodic memory
Verifier Strategies		
Math-Shepherd [643]	✘	Monte Carlo process labels for verifier training
Implicit PRM [787]	✘	Process rewards from policy logits without extra training
Self-Consistency [659]	✔	Majority voting over diverse reasoning paths
LATS [875]	✔	Tree search with LLM-generated value estimates
ReST-MCTS* [805]	✘	PRM-guided self-training via iterative tree search
Compute-optimal scaling [550]	✘	Difficulty-adaptive compute allocation with process verifiers

(TS-Guard) trained via multi-task reinforcement learning that flags unsafe calls before execution, substantially reducing harmful invocations. These verifiers are increasingly stress-tested by dedicated safety benchmarks. The need for them is underscored by AgentHarm [8], a motivating benchmark showing that leading LLMs remain surprisingly compliant with explicitly malicious multi-step tasks; complementary benchmarks then probe specific settings: R-Judge [790] measures safety-risk awareness over multi-turn interaction records, RedCode [193] probes risky code execution and generation for code agents, ST-WebAgentBench [303] evaluates policy-constrained actions for web agents, and OS-Sentinel [571] (distinct from the SentinelAgent runtime monitor of §4.5) combines formal and neural validation for mobile GUI agents in realistic workflows.

(3) Faithfulness Verification. Faithfulness verification certifies that an agent follows the given instructions and stays on task, rather than drifting toward a superficially plausible but misaligned objective. Neuro-symbolic checkers translate natural-language instructions into formal constraints that an output can be mechanically checked against [563], while instruction-following evaluations for multi-turn, tool-using web agents test whether constraints stated early in a session are still honored many steps later [118]. This concern is sharpened by evidence of *goal drift*: agents can gradually abandon their original objective under accumulated context or contextual pressure, so faithfulness verification must track adherence across the whole trajectory rather than at a single step [20, 427]. In agentic settings it is thus the natural counterpart to the maker–checker separation of Section 4.1.3: an independent checker guards against completions that satisfy the visible goal while violating the user's underlying intent.

(4) Multi-agent Cross-verification. Multi-agent cross-verification employs several models for mutual critique. Multiagent debate [139] shows that iterative inter-model deliberation substantially reduces factual errors, and Multi-Agent Verification [362] demonstrates that scaling the number of independent aspect verifiers yields weak-to-strong generalization.

4.6.2 Verification Levels

Verification mechanisms can also be characterized by the stage at which they intervene during the agent's execution lifecycle.

(1) Step-level Verification. Step-level verification evaluates intermediate reasoning quality at each step, complementing hooks that enforce binary allow-or-block decisions via deterministic rules. Where a hook asks "is this action permitted?", a step-level verifier asks "is this reasoning step productive toward the goal?", producing scalar quality signals that guide search and correction rather than gating execution. Building on earlier evidence that training answer verifiers and combining process- with outcome-based feedback improves math reasoning [96, 618], process reward models (PRMs) [363] show that per-step supervision substantially outperforms outcome-only evaluation for complex reasoning, AgentPRM [702] extends this to agentic tasks by scoring each decision on goal proximity and incremental progress, and CRITIC [182] introduces a tool-interactive verify-then-correct loop that establishes external feedback as essential for reliable self-correction. Recent work advances step-level verifiers along three axes. *Generative verification*, which recasts reward modeling as next-token prediction [822], lets GenPRM [857] and ThinkPRM [272] reason step-by-step before scoring, showing that generative verifiers trained on minimal labels can match or outperform discriminative PRMs and rival strong proprietary judges on process benchmarks. *Domain-specialized PRMs* tailor verifiers to concrete agent domains: Web-Shepherd [49] builds structured subgoal checklists for web navigation at substantially lower cost than GPT-4o-based verification, while SWE-PRM [163] and SWE-TRACE [203] deliver trajectory-level course-correction and rubric-based dense feedback for software-engineering agents. *Efficient supervision* concentrates labels where they matter: CSO [329] restricts preference learning to verified critical steps that demonstrably flip task outcomes, requiring supervision at only a small fraction of trajectory steps, and ToolPRMBench [311] contributes the first large-scale PRM benchmark for tool-using settings, showing that reinforcement learning substantially improves robustness and generalization in tool-specialized verifiers. Most recently, step-level verification is moving toward fully *agentic* verifiers that themselves reason and invoke tools while scoring each decision, rather than emitting a single scalar in one pass, as in AgentV-RL [818].

(2) End-of-run Verification. End-of-run verification evaluates agent outputs after task completion. The LLM-as-a-Judge paradigm [867] uses strong models to score final outputs, and its evolution toward agentic judges has proceeded along two lines. *Agent-as-a-Judge* augments the judge with tools and multi-step reasoning: Agent-as-a-Judge [898] equips the evaluator with tool-augmented assessment for substantially higher human agreement, and a recent paper [775] systematizes how agentic judges combine planning, tool-augmented verification, and persistent memory. *Multi-agent collaborative evaluation* leverages inter-model deliberation: CollabEval [489] runs a three-phase initial-evaluation, discussion, and consensus process that outperforms competitive debate; a complementary route instead scales test-time compute for judging itself, which we discuss under verifier strategies below. *Self-correction and environment grounding* close the loop after execution: Reflexion [544] performs verbal self-reflection with episodic memory across trials, though CorrectBench [605] finds that self-correction yields only limited gains on reasoning models at significant cost; environment-based frameworks such as WebArena [885] and SWE-bench [260] verify success through functional correctness of final states, and SWE-RM [546] complements test-suite checks with an execution-free reward model that measurably lifts SWE-bench Verified accuracy. Chain-of-Verification [121] decomposes a response into atomic claims for independent post-hoc checking. Moving beyond static post-hoc scoring, agentic verifiers increasingly probe the environment after a run: verifying a GUI agent through online, proactive interaction lets the judge actively confirm claimed outcomes rather than trust the final transcript alone [102].

4.6.3 Verifier Strategies

Orthogonal to temporal granularity, verifier quality can be improved through training-free architectural designs and learning-based methods, which we group into three families.

(1) Automated labels and implicit rewards. Math-Shepherd [643] automates process-label generation via Monte Carlo estimation, eliminating costly human annotation, and Implicit PRM [787] shows that process rewards can be extracted from outcome-trained models at no additional cost. Self-consistency [659] offers a complementary training-free signal through majority voting across diverse reasoning paths.

(2) Tree search with verifier guidance. A family of methods couples Monte Carlo tree search with LLM self-

evaluation for structured exploration. LATS [875] and rStar-Math [191] combine MCTS with model-generated value estimates, AB-MCTS [436] adds adaptive branching that decides whether to "go wider" or "go deeper," and ReST-MCTS* [805] develops reinforced self-training that harvests higher-quality reasoning traces through iterative process-reward-guided tree search.

(3) Compute-optimal Verification Scaling. Test-time compute scaling [550] shows that verifier-guided search lets smaller models match substantially larger ones, and DeepSeek-R1 [112] demonstrates that self-verification can emerge from reinforcement learning with rule-based rewards alone. RLV [521] unifies LLM reasoners with learned value functions, improving MATH accuracy while enabling markedly more efficient scaling, while MCTS-Judge [665] carries test-time scaling into the LLM-as-a-Judge setting, markedly improving code-correctness accuracy through System-2 reasoning. Extending this scaling to specialized domains, tool-integrated verification interleaves reinforcement learning with external tools so that the verifier can certify multi-step reasoning that pure model judgment cannot [812].

5 Optimization: Internalizing Long-Horizon Capability

Overview. In Section 4, we describe how long-horizon capability can be supplied by the harness that specifies, steers, and consolidates each run. This section turns to *optimization*: how such capabilities can be progressively internalized into model parameters. We organize the section along the *training pipeline*. We begin with the *architectural substrate* (Section 5.1), which determines whether long-horizon training and inference are computationally feasible in the first place: a policy that cannot represent and efficiently process a long trajectory cannot be effectively trained on one. We then examine six stages of capability development: data and environment synthesis (Section 5.2), pre-training and mid-training (Section 5.3), fine-tuning (Section 5.4), reinforcement learning (Section 5.5), on-policy distillation (Section 5.6), and self-evolution (Section 5.7). For each stage, we identify the long-horizon capabilities it develops, the mechanisms through which they are acquired, and the limitations that remain. Figure 11 provides an overview of the pipeline, while Table 5 summarizes the characteristics, mechanisms, and representative works associated with each stage.

The Architectural Substrate and Six Training Stages

1. **Architectural Substrate** (Section 5.1): makes long trajectories efficient to train on and decode from.
2. **Data and Environment Synthesis** (Section 5.2): constructs verifiable long-horizon experience, including tasks, environments, and trajectories, that later stages can use for training and feedback.
3. **Pre-/Mid-training** (Section 5.3): builds action–observation and reasoning priors, so agentic behavior becomes native rather than prompted.
4. **Fine-tuning** (Section 5.4): instills behavioral and tool-use discipline from curated agentic trajectories.
5. **Reinforcement Learning** (Section 5.5): optimizes whole trajectories and confronts long-horizon credit assignment under sparse, delayed reward.
6. **On-policy Distillation** (Section 5.6): consolidates the policy on its own state distribution, reducing distribution shift and stabilizing earlier learned behaviors.
7. **Self-evolution** (Section 5.7): turns self-generated experience, feedback, and task-environment updates into continuing policy improvement.

5.1 Architectural Substrate

A long-horizon agent can internalize only what its architecture can represent: a policy that cannot ingest, retain, and efficiently use a long interaction history, such as observations, tool results, and constraints that accrue across many steps, cannot be trained or served on long trajectories. The architectural substrate therefore sets the ceiling for every later training stage, which is why we begin here. From an architecture view, the central question is how the model keeps and uses a long history. We group the main approaches into three representational classes: explicit-context architectures, compressed-state architectures, and hybrid

Table 5 The agentic training pipeline by stage. Each row is the architectural substrate or one of the training stages. **Representative mechanisms** lists the stage's sub-mechanisms (one per line, following its subsections), and **representative works** pairs each sub-mechanism with two concrete works on the same line.

Stage (§)	Internalized capability	Representative mechanisms	Representative works
Architectural Substrate (§5.1)	Affordably representing and serving long histories	• Explicit-context • Compressed-state • Hybrid memory • High-throughput	Longformer [32], Big Bird [792] Mamba [186], RWKV [476] Jamba [300], Kimi Linear [842] EAGLE [352], GQA [5]
Data & Environment Synthesis (§5.2)	Executable, verifiable long-horizon experience	• Task synthesis • Environment synthesis • Trajectory synthesis	TaskCraft [540], WebShaper [590] SWE-Gym [467], EnvScaler [555] TOUCAN [737], AgentGym-RL [701]
Pre-/Mid-training (§5.3)	Reasoning and perception priors	• Reasoning priors • Long-context priors • Multimodal priors • Data-mixture design	DeepSeek-V3 [110], Qwen3-Coder [46] YaRN [477], LongLoRA [78] Qwen2-VL [644], InternVL3 [892] DoReMi [715], OPUS [649]
Fine-tuning (§5.4)	Behavioral and tool-use discipline	• Selection & mixing • Curriculum • Distillation	AgentTuning [794], LIMI [711] ScalingInter-RL [701], E2H Reasoner [469] Agent Distillation [269], Agent-R [789]
Reinforcement Learning (§5.5)	Feedback-driven long-horizon decisions	• Credit assignment • Policy optimization • Sampling strategy • Interaction patterns	ToolRL [487], RuscaRL [887] DAPO [780], GiGPO [156] Tree-GRPO [250], TCOD [636] GLIDER [232], Memory-R1 [743]
On-policy Distillation (§5.6)	Consolidating behavior on the policy's own states	• Teacher-guided • Self-improving	GKD [3], DAgger-LLM [309] π-Play [838], Self-Distillation Zero [216]
Self-evolution (§5.7)	Cross-task experience accumulation	• Offline bootstrap • Online interaction • Co-evolution	STaR [793], rStar-Math [191] R-Zero [233], Agent0 [704] Env Tuning [401], Agent-World [131]

architectures. These are not strict model families, but different ways to represent past information within the model. We discuss high-throughput mechanisms separately as a serving-oriented axis rather than as a fourth representation class. This view follows the main split in efficient Transformer and state-space papers [591]. We do not treat Mixture-of-Experts (MoE) as a fourth representational class. MoE routes tokens to different feed-forward experts, as in GShard and Switch Transformer [152, 301]. It primarily changes model capacity, but it does not by itself decide how the sequence history is stored. Accordingly, MoE models can still use explicit attention, latent attention, or hybrid blocks [111, 300, 601].

5.1.1 Explicit-Context Architectures

Explicit-context architectures keep history as a token sequence. The model then uses attention to route information across these tokens. Full attention gives the cleanest form of this design, because any token can attend to any other token [620]. Long-context Transformer variants reduce the cost with local attention, sliding windows, block-sparse attention, or global tokens. Longformer and Big Bird are classic examples of this direction [32, 792]. Mistral 7B uses sliding-window attention and grouped-query attention (GQA) for lower-cost decoding [252]. Gemma 2 and Gemma 3 use local-global attention patterns. Gemma 3 uses more local attention layers to reduce KV-cache cost at long context [595, 596]. Efficient Transformer papers give a broader view of this design space [591]. More recent work makes sparsity *natively trainable* rather than an inference-time approximation, making long-horizon training affordable: DeepSeek NSA couples a hardware-aligned, natively trainable sparse-attention design with efficient kernels [786], and MoBA applies a mixture-of-block-attention routing that lets each query attend to a learned subset of context blocks [398].

This class is useful for long-horizon agents because earlier states remain explicitly represented. The model can, in principle, look back at an old observation, a tool result, or a user constraint. These objects are common in reasoning-and-acting agents and web agents [761, 885]. The cost is the main weakness. As the context grows, attention compute and key-value (KV) cache size also grow [5, 620]. If the model uses a sparse pattern, it may also miss a remote token that matters. Long-context benchmarks and analyses show that locating useful evidence inside long inputs remains hard [28, 378]. Many open base model reports still fit this class. Qwen2.5 and Qwen3 mainly use decoder-only Transformer architectures; Qwen3 adds both dense and MoE variants,

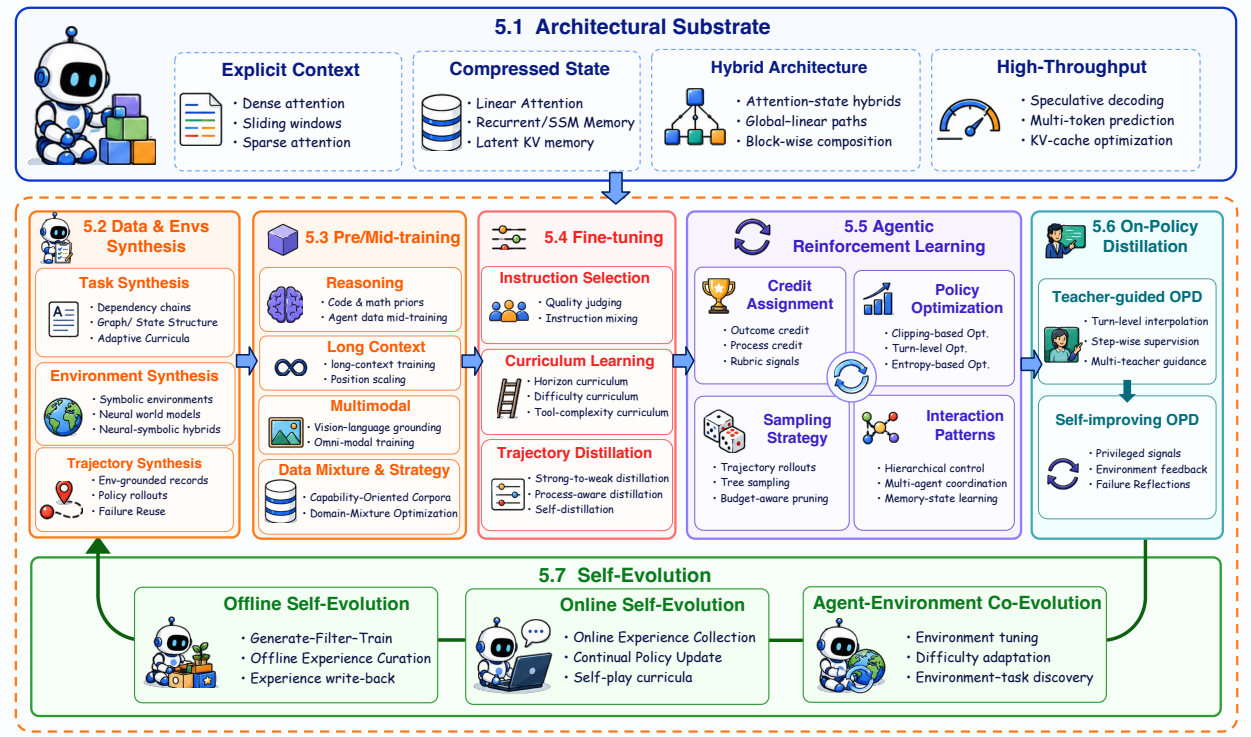


Figure 11 The agentic training pipeline for internalizing long-horizon capabilities. As the internalized counterpart to the externalized harness described in Section 4, the pipeline builds on an architectural substrate that supports long-sequence representation and proceeds through six stages. Each stage either internalizes a capability previously provided at runtime or generates the experience required for its subsequent internalization.

but the history is still accessed through attention [601, 744]. The GLM series follows the same pattern. The ChatGLM report describes the path from GLM-130B to GLM-4. GLM-4.5 is an MoE foundation model for agentic, reasoning, and coding tasks, but it is still an attention-based Transformer family [178, 795].

5.1.2 Compressed-State Architectures

Compressed-state architectures do not keep all pairwise token interactions. They instead encode history into a more compact internal representation. Some methods compress the attention map or the token set, as in low-rank attention and kernelized linear attention [92, 651]. Other methods use a recurrent or state-space form. This includes Retentive Network (RetNet), Receptance Weighted Key Value (RWKV), structured State Space Models (SSMs), and selective SSMs such as Mamba [186, 187, 476, 576].

This class fits the stream-like nature of agent work. In many agent settings, a model observes, reasons, acts, and observes again [761, 885]. A compressed state can be updated step by step, so the runtime memory can stay small [186, 187, 476, 576]. The weakness is precision. Once old tokens are folded into a state, the model may not recover a rare fact or an exact old string. This is a core trade-off between efficient recurrence and content-level access [473, 606]. This is a problem when an agent must obey a past constraint exactly, as in tool-use and web tasks that carry instructions across steps [761, 885]. A partial, and importantly different, base-model version of this idea is Multi-head Latent Attention (MLA), which compresses the KV cache into a latent form. DeepSeek-V2 introduced MLA together with DeepSeekMoE, and DeepSeek-V3 kept this design at larger scale [110, 111]; Kimi-K2 likewise pairs a large MoE model with MLA [597]. Unlike recurrent or state-space compression, however, MLA keeps a per-token latent and still attends over all past tokens, so it preserves per-token addressability and does not incur the exact-recall loss described above. It is therefore best read not as a recurrent compressed state but as a low-rank KV compression *of* explicit context, sitting between the two classes and doubling as a throughput mechanism (Section 5.1.4).

5.1.3 Hybrid Architectures

Hybrid architectures combine explicit attention with compressed or recurrent state [300, 842]. The goal is to combine precise content access with efficient long-range state tracking. Attention can keep precise access to important tokens [378, 620]. A compressed state can carry long-range signals at low cost [186, 606]. In this view, attention is used for exact evidence, while the state module is used for long-term continuity [300, 842]. Jamba is a direct example: it mixes Transformer layers, Mamba layers, and MoE layers [300]. Kimi Linear is another clear example. It combines Kimi Delta Attention units with a global MLA layer. The model therefore keeps both a linear-attention path and an explicit global attention path [842].

This makes hybrid design a plausible direction for long-horizon agents. Agent runs need both exact recall and cheap long-term tracking, as shown by web-agent and long-context tasks [28, 885]. Pure attention gives direct access but can be expensive at long context [591, 620]. Pure compressed state is cheaper but may lose details [473, 606]. A hybrid model can assign these roles to different layers, heads, or blocks, as in Jamba and Kimi Linear [300, 842]. The reports above show the same direction at system scale: strong models often combine attention, KV-cache reduction, and MoE routing [110, 597, 601]. These parts solve different problems and should not be collapsed into one label [5, 152, 301].

5.1.4 High-Throughput Mechanisms

High-throughput mechanisms form a second axis under architecture [5, 111, 180, 302, 535]. They do not decide how the model stores long history; they reduce the serial work or cache traffic needed to decode from that history [5, 111, 180, 302, 535]. A complementary, kernel-level route keeps attention *exact* but IO-aware: FlashAttention tiles the computation to avoid materializing the full attention matrix in high-bandwidth memory, cutting memory traffic and enabling longer sequences without altering the model's outputs [108]. One group reduces serial dependence in the output path through parallel generation with diffusion language models [559] or through autoregressive lookahead [43, 180, 302, 352]. Multi-token prediction trains the model to predict several future tokens at each position, so the model learns a longer local prediction horizon [180]. Speculative decoding uses a draft model to propose several tokens and lets the target model verify them in parallel [302]. Medusa keeps the draft process inside one model by adding multiple decoding heads [43]. EAGLE moves the draft signal into feature space and treats feature uncertainty as central to speculative sampling [352]. DeepSeek-V3 uses multi-token prediction as an auxiliary training objective and also presents it as a path to speculative decoding [110].

The other group reduces the cost of reading and writing the KV cache [5, 111, 535]. Multi-query attention shares key and value projections across query heads to speed decoding [535]. Grouped-query attention keeps more key-value groups than multi-query attention, but still reduces the cache cost relative to full multi-head attention [5]. MLA compresses the KV cache into a latent form, so it is both a compressed-state design and a throughput design [110, 111]. These mechanisms matter for long horizon agents because long runs often produce many intermediate tokens after a large context has already been built [180, 302, 761, 885]. They reduce the serving cost of long decoding, but they do not remove the need for explicit recall, compressed state, or hybrid memory when old evidence must be found accurately [28, 180, 302, 378].

5.2 Data and Environment Synthesis

Long-horizon behavior is learned from interaction experience that is executable, feedback-bearing, and reusable. Unlike static instruction–response pairs, such experience records how an agent decides, observes feedback, corrects errors, and accumulates information over a long trajectory. Deployment logs, demonstrations, and benchmark rollouts offer natural sources of such data, but their direct use is constrained by acquisition cost, privacy, and limited reproducibility. Controlled data synthesis has therefore become an important route for constructing long-horizon training experience. It centers on three coupled objects: *tasks* define goals and capability pressures, *environments* instantiate execution and verification, and *trajectories* turn interaction histories into trainable signals.

5.2.1 Task Synthesis

Task synthesis determines the objective distribution for long-horizon agents. A task defines the target behavior through goals, constraints, initial and terminal states, and success criteria. Task generation sits at the entry of interactive experience collection, since a task definition largely fixes horizon, dependency structure, and failure modes [238, 808]. Existing methods mainly construct tasks through dependency chains, structured information spaces, state transitions, and capability-adaptive curricula.

Dependency-chain Construction. The most direct way to lengthen horizon is to chain subgoals so that early decisions constrain the actions later available. Long-horizon diagnostics relate failures to intrinsic horizon and compositional depth [658]; horizon-length studies further show that longer tasks aggravate optimization instability and credit assignment [279]. AgentSynth composes easy-to-generate subtasks into longer computer-use tasks, while TaskCraft expands atomic tasks into multi-hop, multi-tool objectives [540, 713]. The horizon here grows through necessary intermediate dependencies rather than through mere interaction length.

Graph-structured Information Construction. When horizon arises from evidence rather than from actions, tasks are drawn as paths through web pages, documents, tools, or evidence graphs, and the difficulty shifts to selecting relevant evidence under branching, incomplete, or masked observations. WebWalker and WebDancer construct multi-step information-seeking tasks from traversal, crawling, and backward question evolution [689, 690]. WebSailor uses knowledge-graph walks and information masking, while WebShaper formulates information seeking through set and knowledge projection to control evidence structure, answer space, and verifiability [327, 590]. WebSailor-V2 and WebWeaver further extend this line to more open-ended information-seeking tasks [326, 356].

State-transition Construction. A third view frames a task as the gap between an initial and a target state, or as recovery from an erroneous one. In software settings, SWE-Gym, R2E, and SWE-smith reconstruct tasks from repositories, issues, commits, and tests [248, 467, 748]. Reverse synthesis makes this state-transition view explicit: DIVE generates tasks from executed evidence traces, TASTE evolves valid tool-sequence skeletons, and CLI-Gym derives repair tasks from abnormal command-line states and historical errors [53, 271, 369]. Execution traces and state gaps are thereby recast as tasks with verifiable outcomes.

Curriculum-driven Construction. The routes above fix a task distribution in advance; curricula instead let difficulty track the current policy, reducing the mismatch between static task sets and evolving capability. WebRL generates new web tasks from failed attempts, and Learn-by-interact reconstructs instructions and supervision from environment documents and interaction histories [484, 562]. AgentFrontier targets the model's zone of proximal development, while CuES uses exploration and curiosity signals to synthesize environment-grounded RL tasks [75, 417]. Self-evolving agent papers view such mechanisms as part of joint agent–task evolution [165].

These routes impose complementary capability pressures on plan maintenance, evidence selection, and state recovery, whereas curricula act orthogonally, tuning how fast difficulty grows rather than what it tests. Which pressure to emphasize depends on the target deployment, since a distribution poorly matched to the policy tends to either saturate or destabilize training.

5.2.2 Environment Synthesis

Environment synthesis supplies the training ground where long-horizon agents accumulate the interaction experience that static corpora cannot provide. Whereas tasks specify what to solve, environments determine how actions are executed, observed, and rewarded, closing the action–observation–feedback loop that turns a passive predictor into an active decision maker. A synthesized environment makes this loop reproducible, resettable, parallelizable, and verifiable, so that agents can act, fail safely, and receive checkable feedback at scale. Real repositories, browsers, APIs, databases, desktop systems, and game simulators provide fidelity [145], but only controlled synthesis makes such interaction cheap and reliable enough for repeated rollout. We organize environment synthesis into three families according to how their transitions and feedback are produced: symbolic environments, neural environments, and neural–symbolic hybrids.

Symbolic Environments. Symbolic environments implement transitions and feedback through code, tests, databases, or business logic, which yields stable execution and verifiable hard signals. This family has developed along several technical routes, distinguished by the substrate they make executable. A first route builds *repository-level software environments*: SWE-Gym packages real repositories, issues, dependencies, and tests into executable tasks, and follow-up systems scale repository construction, test generation, and task expansion [248, 467, 748, 797, 856]. A second route targets *terminal and command-line environments*, where Terminal-Bench standardizes hard, realistic shell tasks and later work co-derives instructions, sandboxes, and teacher trajectories to scale terminal-agent training [82, 86, 369, 428]. A third route provides *GUI, web, and game execution environments*, represented by WebArena, OSWorld, and GameWorld and extended toward broader interface and generation settings [251, 461, 695, 716, 852, 885]. A fourth route constructs *tool- and domain-specific environments*, ranging from multi-step tool substrates such as AgentScaler to grounded scientific and medical settings [148, 538, 555, 626, 729]. Across these routes, explicit rules stabilize verification but remain less effective under ambiguity, dynamic interfaces, and cross-tool side effects.

Neural Environments and World Models. Neural environments approximate transitions, observations, or feedback with LLMs, visual generators, or latent dynamics, reducing the cost of repeated interaction. These methods treat the model as a *world action model*: conditioned on the interaction history and the agent's action, it simulates the next round of environmental feedback, so that rollout no longer touches the real system. Three technical routes differ mainly in the representation over which the next state is predicted. The first route learns *pixel-level world models*, represented by DIAMOND and GameNGen, which generate action-conditioned visual futures and have since been extended to embodied and open-world settings [2, 6, 89, 170, 196, 249, 619, 769, 839]. The second route builds *text-level simulators* better suited to web, GUI, tools, and code, where WebDreamer predicts future web states and later systems structure UI, webpage, or software states [47, 149, 188, 572, 661, 712, 870]; recent language world models push this route toward general agents, learning next-state prediction across many agent domains and coupling generative simulators with agents in difficulty-aligned co-evolution [195, 905]. The third route learns *latent world models* in the style of joint-embedding prediction, which compress dynamics for cheaper long-horizon prediction [24, 25, 30, 176]. Neural environments scale cheaply, but they may introduce hallucination, drift, unreliable feedback, and sim-to-real gaps.

Neural–Symbolic Hybrid Environments. Neural–symbolic hybrid environments keep hard state and final verification in databases, file systems, programs, or APIs, while using models for user simulation, language interfaces, or hard-to-specify observations. They can be grouped by where the model is inserted into an otherwise symbolic loop. One route places models at the *user and application interface*, as in τ-bench, which combines stateful applications, user simulation, and final-state checks [399, 614, 762]. A second route standardizes *tool-server backends* through MCP [299, 411, 668, 700]. A third route pairs *generated content with symbolic constraints*, coupling generated webpages with state-machine constraints and completion checks [50, 695, 852]. A fourth route grounds *domain environments* in real data, retrieval, or computation while using models to expand tasks and feedback [538, 626, 804, 894]. The design choice throughout is where to draw the boundary between model-based simulation and symbolic verification.

Across these families, feedback and verification quality remain central. Software tasks can rely on tests and execution; tool tasks use persistent database, file, or API states; web and GUI tasks often require scripts, page-state checks, visual-semantic matching, or expert rules; and open-ended tasks depend more heavily on rubrics, LLM judges, or hybrid verifiers. Dense step-level signals can improve credit assignment, but stronger automation also raises judge bias, verifier loopholes, and reward-hacking risks [238, 402]. Synthesized environments must therefore balance correctness, diversity, complexity, and fidelity.

5.2.3 Trajectory Synthesis

For long-horizon agents, trajectory synthesis internalizes interaction as trainable behavioral experience: it records how a policy acts, observes, errs, and recovers over an environment, and converts that history into supervision, preferences, rewards, and memory. What can be recorded is bounded by the environment's observability and verification, so trajectory quality inherits the strengths and gaps of the synthesized environment. A useful trajectory keeps the cross-step dependencies, delayed feedback, and recovery decisions

that plain instruction–response data discard. Prior work develops this idea along four directions: grounding records in real environments, aligning rollouts with the trained policy, assigning step-level credit while reusing failures, and compressing long histories into reusable experience.

Environment-grounded Execution Records. Environment-grounded records capture state changes, tool side effects, and feedback produced during execution. Textual ReAct traces or function-call transcripts provide useful templates, but they often omit post-action states and failure signals. TOUCAN synthesizes large-scale tool-agentic trajectories in real MCP environments; OpenResearcher anchors deep-research trajectories in a reproducible offline corpus; and ASTRA constrains multi-turn tool trajectories with tool-call graphs [354, 604, 737]. WebSTAR, ANCHOR, and HATS further tie GUI and computer-use trajectories to interface states, branch points, action ambiguity, and step-level feedback [215, 531, 679].

Policy-aligned Rollouts. Policy-aligned rollouts reduce distribution shift between collected trajectories and the policy being trained. Expert, human, or strong-model trajectories provide useful cold-start priors, whereas on-policy or near-on-policy rollouts expose the current policy's own exploration, failures, and recovery states. AgentGym-RL, AgentRL, and ASearcher combine multi-turn rollout, horizon scheduling, and asynchronous sampling; DPEPO uses parallel trajectories for cross-path exploration; and Agent Lightning decomposes execution records into RL-consumable units [166, 409, 701, 811, 820].

Step-level Credit Assignment and Failure Reuse. Step-level credit assignment exposes local correctness, redundant actions, and critical errors that are hidden by final success labels. WebSTAR filters learnable GUI actions with a process reward model; ANCHOR expands and cleans post-branch trajectories; and HATS targets difficult segments under contextual and visual ambiguity [215, 531, 679]. CSO identifies outcome-changing critical steps, VPR constructs verifiable turn-level rewards, and GEAR reallocates trajectory-level advantages to segments [329, 333, 785]. The unit of supervision shifts from whole trajectories to steps, segments, and critical decisions. Failure trajectories provide complementary training signals for planning, evidence selection, tool use, state recovery, and harness interaction. CSO generates critical-step replacements; AgentHER relabels failed attempts under reachable alternative goals; and AOI and PIVOT use failed records for correction and trajectory refinement [125, 329, 751, 826]. HORIZON and HarnessFix further show that failures may stem from tools, context management, verifiers, or harnesses rather than model decisions alone [64, 658]. Useful trajectories therefore retain errors, control flow, state changes, and verification provenance.

History Compression and Experience Reuse. History compression addresses the context and credit-assignment difficulties caused by long interaction records. ReSum trains over summary-bounded segments; Context-Folding and AgentFold organize trajectories into foldable subprocesses; and Memory-as-Action and Memory-R2 treat memory edits as trajectory decisions with credit-assignment consequences [573, 693, 742, 768, 846]. Long-horizon trajectories can thus be segmented, summarized, folded, converted into memory states, and abstracted into reusable experience.

In summary, trajectory synthesis is moving from text-only demonstrations toward environment-grounded, verifiable, and reusable agentic experience. Task synthesis sets the capability pressure, environment synthesis supplies the interaction loop, and trajectory synthesis turns execution histories into trainable signals.

5.3 Pre-training and Mid-training

Pre-training and mid-training are the earliest stages for internalizing the capabilities required by long-horizon agents. Pre-training establishes broad linguistic, factual, coding, mathematical, and reasoning priors through large-scale self-supervised learning, while mid-training further reshapes these priors before instruction tuning or reinforcement learning. For long-horizon agents, these stages are important because later supervised fine-tuning (SFT), alignment, and policy optimization can only elicit and refine capabilities that are already sufficiently represented in the base model. We therefore discuss pre-training and mid-training from four aspects: reasoning capability, long-context state modeling, multimodal perception, and the data mixture and training strategies that support these capabilities.

5.3.1 Reasoning Priors

Reasoning is a foundation for long-horizon agency because agents must decompose goals, maintain intermediate hypotheses, evaluate partial results, and revise plans over multiple steps. Code and mathematics are representative sources for injecting such priors during pre-training and mid-training, since they expose models to structured dependencies, executable programs, multi-step derivations, and verifiable intermediate states. Recent model reports increasingly emphasize coding, mathematics, reasoning, and agentic capabilities as part of their training recipes, including Qwen2.5 [744], DeepSeek-V3 [110], GLM-4.5 and GLM-5 [178, 179], and Kimi K2 [597]. Qwen3-Coder-Next [46] makes this direction more explicit through targeted mid-training toward code reasoning, repository-level understanding, and agent-style interaction patterns. These structured domains make reasoning supervision closer to long-horizon agent workloads than ordinary short-form text, because they require cross-step consistency, error checking, and compositional problem solving.

5.3.2 Long-Context State

Long-context capability provides the substrate for maintaining task state across extended interactions. For a long-horizon agent, context is not merely a longer input window; it serves as a serialized working state that may contain goals, observations, retrieved evidence, tool outputs, failed attempts, intermediate plans, and revised decisions. A first line of work extends the usable window by rescaling rotary position embeddings: Position Interpolation linearly rescales positions to fit a target length [66], YaRN improves this with an NTK-aware, frequency-dependent schedule for cheaper extension [477], and LongLoRA makes such extension parameter-efficient through shifted sparse attention during fine-tuning [78]. A complementary line studies the training recipe rather than the positional scheme. ProLong shows that long-context capability depends on continued-training data composition, training sequence length, and instruction tuning [171]; UltraLong pushes context to the million-token regime through continued pre-training and instruction tuning [721]; and LongWriter shows that long input windows do not by themselves induce long-output behavior without long-output training examples [29]. Recent frontier models also operationalize this direction. GLM-5.2 couples a 1M-token context window with specialized training for long-horizon coding-agent scenarios [179]. What matters for agents is therefore whether extended context is actually used for state tracking and evidence integration, not the maximum length a model nominally accepts [133, 625].

5.3.3 Multimodal Perception

Long-horizon agents increasingly operate in environments where task state is distributed across webpages, screenshots, videos, and audio streams; their perceptual priors must therefore extend beyond linguistic shortcuts that can mask basic visual failures [61, 502, 608, 624]. Open vision–language models establish the basic substrate: Cambrian-1 explores vision-centric representations and visual grounding [607]; Qwen2-VL perceives images and long videos at varying scales through dynamic resolution and multimodal rotary position embedding [644]; and the InternVL3 series advances open pre-training for high-resolution, long-context visual understanding [654, 892]. Recent industrial reports push these priors toward agent-centric perception, with the common thread being that visual grounding and GUI control are trained as native capabilities rather than added after a text-only stage: Seed1.5-VL and the subsequent Seed1.6 series pre-train on large-scale interleaved vision, text, and interaction data to sustain long-video understanding and GUI control across evolving screens [42, 194], while MiMo-VL treats visual grounding and GUI localization as first-class targets on a reasoning-optimized backbone [393, 394]. On the omni-modal side, Qwen3.5-Omni and Kimi K2.5 extend joint text, vision, and audio pre-training toward multimodal agentic intelligence [498, 598]. The shared implication is that long-horizon capability depends on persistent multimodal state tracking and cross-modal evidence grounding, not textual reasoning alone.

5.3.4 Data Mixture and Training Strategies

The above capabilities are shaped by how training data are selected, mixed, and structured. Recent work shifts pre-training from generic web-scale language modeling toward capability-oriented data distribution design. DataComp-LM [316], FineWeb [475], Dolma [552], and OLMo [184] make pre-training data construction more reproducible. Beyond static corpus construction, DoReMi [715] and Data Mixing Laws [767] show that domain proportions can be optimized or predicted with proxy experiments, while OPUS [649] performs

principled data selection at each pre-training iteration by scoring sample utility in the optimizer-induced update space, shifting pre-training from more tokens to better tokens and improving both from-scratch and continued pre-training under a fixed compute budget. For long-horizon agents, such data strategies act as inductive biases for reasoning, state modeling, and compositional problem solving, rather than merely as mechanisms for scaling factual knowledge. A further strategy is to structure training data as trajectories rather than isolated text samples. AgentTuning [794], Agent-FLAN [79], and AgentBank [557] show that interaction trajectories can improve generalized agent abilities such as planning, memory, and tool use. Since high-quality long-horizon trajectories are scarce, synthetic and verifier-guided data are also used to control horizon length, task difficulty, feedback structure, and failure-recovery patterns [29, 46]. Here the decisive factor is not what the data contain but whether they preserve temporal dependencies, interaction feedback, and recovery processes.

Taken together, pre-training and mid-training form the first stage of internalizing long-horizon capability. Reasoning-oriented data provide structured problem-solving priors, long-context training supports state maintenance over extended interactions, multimodal training enables persistent observation across heterogeneous evidence, and data mixture strategies determine how these signals are balanced and selected. These latent priors are then elicited, sequenced, and optimized by SFT and RL into reliable agentic behavior.

5.4 Fine-tuning

Fine-tuning converts the latent priors acquired during pre-training and mid-training into controllable long-horizon behavior. Whereas pre-training shapes broad parameter-level capabilities, supervised fine-tuning (SFT) teaches the model how these capabilities should be expressed during interaction: how to decompose goals, call tools, interpret observations, update state, recover from errors, and terminate with a final answer. For long-horizon agents, SFT is therefore process-level imitation over trajectories that contain intermediate reasoning, actions, tool results, and state transitions, rather than instruction-following over isolated input–output pairs. It typically serves as a cold-start stage that constrains the model to valid, task-relevant interaction patterns before any online optimization.

The central issue is how to construct a supervised trajectory distribution that is diverse enough to teach general agent behavior, structured enough to ensure valid tool use, and progressive enough to prepare the model for later stages. We organize this section around three questions: which demonstrations to select and how to mix them, how to stage them through a curriculum, and how to obtain them through distillation.

5.4.1 Instruction Selection and Mixing

Trajectory Quality Over Quantity. Fine-tuning long-horizon agents requires data that expose not only correct answers, but also the intermediate decisions connecting goals, observations, actions, and recovery behaviors across many turns. AgentTuning [794] constructs AgentInstruct, a compact set of high-quality interaction trajectories, and mixes it with general instruction data to strengthen agent skills while preserving general ability. FireAct [54] shows that trajectories generated by stronger models improve agent behavior beyond prompt-only adaptation, and LIMI [711] argues that a small set of strategically curated demonstrations can elicit strong agency-oriented behavior. Rather than cloning full trajectories, ATLaS [81] identifies the *critical steps* of planning, subgoal reasoning, and strategic decision making, and fine-tunes only on them, mitigating expert bias and improving cross-environment generalization. These works reframe agent SFT as an instruction-selection problem: prioritizing trajectory quality, decision-point coverage, and task diversity over raw demonstration count.

Compact Reasoning Demonstrations. A complementary line explains why small trajectory sets can be effective. LIMO [770] elicits complex mathematical reasoning from a strong base model with only 817 curated demonstrations, treating them as cognitive templates over knowledge already encoded during pre-training; s1 [442] builds a 1K-example set under quality, difficulty, and diversity criteria; and LIMR [344] extends the less-is-more observation to reinforcement learning through learning-impact-based sample selection. Unified Data Selection [342] sharpens this idea by selecting samples around high-entropy tokens, which mark critical branching points in reasoning trajectories. This "less-is-more" view is not unchallenged: Ding et al. [122] report that, in an SFT-then-RL pipeline, data scale still governs the attainable performance ceiling while

trajectory difficulty acts as a multiplier, suggesting that selection should maximize decision coverage rather than merely shrink the dataset.

Executable Tool-use Data and Mixture. For tool-using agents, data value further depends on whether actions are syntactically valid, executable, and grounded in observations. CodeAct [656] represents actions as executable Python code, enabling code execution, observation feedback, and self-debugging; APIGen [392] generates function-calling data verified through format, execution, and semantic checks; ToolACE [385] scales synthetic tool-learning data with a self-evolving API pool and multi-agent dialogue generation; and TOUCAN [737] synthesizes large-scale multi-tool, multi-turn trajectories from real-world Model Context Protocol (MCP) environments. Selection also matters at scale: EDGE [844] picks informative multi-turn samples according to guideline effectiveness. Because such corpora remain fragmented across incompatible formats, the Agent Data Protocol [558] introduces a unified schema that standardizes heterogeneous agent datasets into a common trajectory representation; SFT on the resulting cross-domain mixture yields broad gains and avoids the negative transfer induced by single-domain tuning.

5.4.2 Curriculum Learning

Staging by Horizon. Long-horizon trajectories are difficult to imitate directly because errors compound over time and the model must maintain state across many steps before receiving any signal of success or failure. Curriculum learning addresses this by staging the training distribution rather than exposing the model to full-length tasks at once, with the interaction horizon as a natural first axis. ScalingInter-RL, introduced in AgentGym-RL [701], begins with short-horizon exploitation to establish stable foundational policies and then progressively enlarges the interaction budget, mitigating the collapse that plagues naive long-horizon optimization. Studying horizon in isolation, Kim et al. [279] construct controlled task suites that vary only the required action-sequence length and show that horizon itself is a training bottleneck: reducing the horizon first and then extending it stabilizes learning and induces *horizon generalization*, where competence acquired on short goals transfers to unseen longer ones at inference time.

Staging by Difficulty. A second axis orders tasks from easy to hard. E2H Reasoner [469] schedules problems probabilistically from easy to difficult, and shows both empirically and through approximate-policy-iteration bounds that gradually fading out easy tasks prevents overfitting and lets small models acquire reasoning skills that vanilla training cannot reach. Curricula for long-horizon agents can be graded along several dimensions at once, including horizon length, task difficulty, tool complexity, feedback density, and recovery demand, so that a staged cold-start distribution first instills basic interaction competence before later stages optimize decisions under delayed rewards.

5.4.3 Distillation

Because high-quality long-horizon demonstrations are scarce, many fine-tuning pipelines obtain supervision by distilling interaction processes from stronger models, scaffolded agent systems, or executable environments. This is better understood as trajectory distillation rather than answer distillation: the student learns not only final responses, but also planning steps, action formats, observation interpretation, and recovery strategies. We use *distillation* here in its offline sense, where supervision comes from fixed teacher or past-model trajectories consumed as static SFT data, in contrast to the on-policy distillation of Section 5.6 and the closed-loop self-evolution of Section 5.7.

Strong-to-weak Distillation. *Strong-to-weak distillation* transfers agentic behavior from powerful teacher models or expensive agent systems into smaller, deployable students, reusing costly execution as supervised traces. Because agent competence lives in full reason–act–observe trajectories, recent work supervises structure rather than final answers. Agent Distillation [269] transfers retrieval and code-execution trajectories from a large teacher into sub-3B students, using a first-thought prefix and self-consistent action generation to keep tiny agents competitive with far larger chain-of-thought-distilled baselines. Structured Agent Distillation [376] segments each trajectory into [REASON] and [ACT] spans with segment-specific losses, preserving both reasoning fidelity and action validity under compression. Pushing the teacher side further, AgentArk [410] distills a

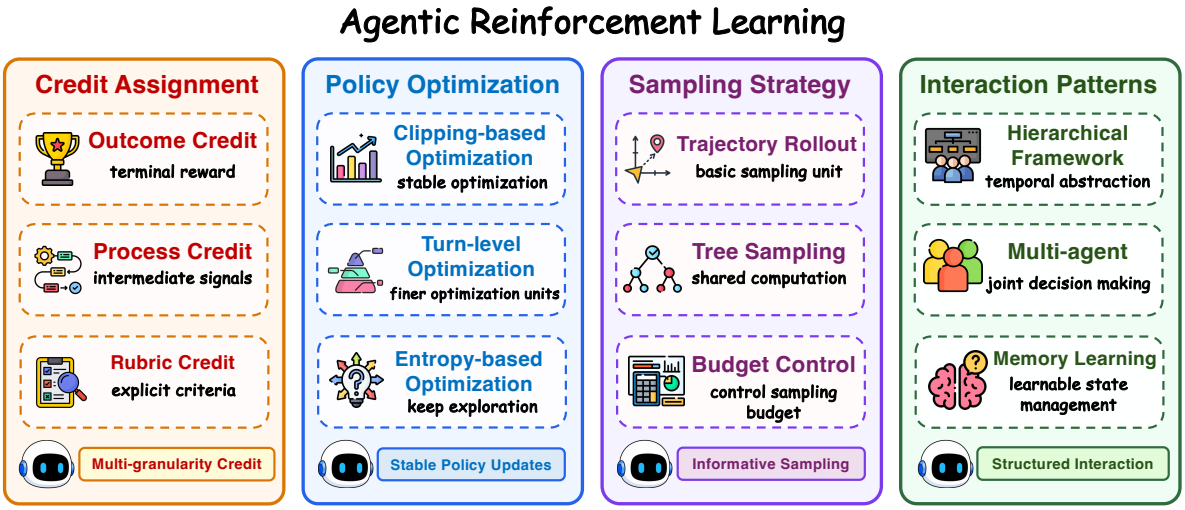


Figure 12 Agentic reinforcement learning through four key questions. These questions correspond to the four components discussed in Section 5.5: credit assignment, policy optimization, sampling strategy and interaction structure.

multi-agent debate system into a single model, turning expensive test-time interaction into implicit self-correction ability. Across these methods, effective distillation aligns supervision with trajectory structure rather than compressing outputs alone.

Self-distillation and Reflective Self-improvement. *Self-distillation* removes the external teacher: the model generates its own trajectories, filters or reflects on them, and trains back on the curated result. Since many long-horizon failures are local, such as choosing a wrong tool, misreading an observation, or failing to recover after an execution error, the central design choice is how to turn raw rollouts into clean, corrective supervision. Watch Every Step [719] explores along expert trajectories, collects step-level process feedback, and learns from contrastive action pairs. Agent-R [789] uses Monte Carlo Tree Search to locate the first erroneous step in a failed rollout and splices it onto an adjacent correct path, so that iterative fine-tuning teaches timely on-the-fly correction rather than end-of-episode fixes. Reflection enriches the signal further: RESD [845] converts failed rollouts into retrospective diagnoses and a reusable "playbook" of lessons, while HERO [374] turns each observation into a turn-level hindsight reflection used to re-score the student's own actions. These reflection-driven variants shade into the on-policy self-distillation of Section 5.6; used offline, their filtered traces still supply denser process supervision than final-answer imitation.

Both regimes share a lesson: distillation for long-horizon agents must preserve *process* structure, whether transferring capability from strong teachers or recycling an agent's own history into cleaner supervision. Both can also amplify biases and environment-specific habits, so execution-based verification, semantic filtering, and mixture with general instruction data remain essential for robust, transferable behavior.

Overall, fine-tuning determines how long-horizon priors surface as interactive behavior: instruction selection defines the supervised behavior distribution, curriculum learning stages it by horizon and difficulty, and distillation transfers complex interaction processes from stronger or self-generated sources. It improves controllability, tool-use validity, state tracking, and recovery, yet stays bounded by the coverage of supervised demonstrations, leaving decisions under delayed rewards and compounding errors to later online optimization.

5.5 Agentic Reinforcement Learning

Compared with supervised fine-tuning, Agentic Reinforcement Learning (Agentic RL) further optimizes the policy against feedback from the agent's own interaction. This extends the optimization targets of LLMs from single-turn responses to long-horizon interaction trajectories. As illustrated in Figure 12, the core challenges involve four aspects: sampling informative multi-turn trajectories, deriving multi-granularity credit signals

from sparse feedback, integrating these signals into stable policy updates, and organizing training interaction patterns such as hierarchy, tool use, multi-agent collaboration, and memory. In the following subsections, we elaborate on credit assignment, policy optimization, and sampling methods in Agentic RL, as well as several unique interaction patterns of Agentic RL. Representative works for each part are shown in Table 6.

5.5.1 Credit Assignment

Credit assignment is one of the core challenges for Agentic RL. It concerns how to attribute credit to heterogeneous decision units when an agent completes long-horizon tasks. This section reviews credit assignment from three perspectives: outcome credit assignment, which scores only the final result; process credit assignment, which shapes the intermediate decision process; and rubric credit assignment, which expresses quality through explicit semantic criteria.

Outcome Credit Assignment. Outcome credit assignment is the most direct form of credit assignment, where the reward is determined only by the final result, such as exact match, F1, task success, final-state checks, unit-test results, or retrieved-answer correctness. Its appeal lies in being automatable, cheap to verify, and easy to scale to large online rollouts. This paradigm was first validated by verifiable terminal rewards in mathematical and code reasoning [112, 345, 533]. It was subsequently carried into interactive settings by a line of web and information-seeking agents led by Search-R1, which reuse task success, environment-state checks, or retrieved-answer correctness as terminal rewards across long-horizon application environments [60, 256, 261, 681, 871]. However, this form of credit is coarse-grained. A single terminal score is shared across the entire trajectory, conflating key exploration, redundant actions, and early mistakes into one learning signal. As a result, the policy struggles to discriminate the quality of individual local decisions.

Process Credit Assignment. Process supervision predates agentic RL in mathematical reasoning, where process reward models (PRMs) score intermediate steps rather than only final answers [363, 643]; reinforcement learning with verifiable rewards then scaled outcome-based optimization for reasoning [112, 533]. Agentic credit assignment inherits this lineage and extends it to multi-step, tool-using trajectories. Process credit assignment extends the evaluation target from the final answer to the intermediate process. Existing methods follow two complementary routes. The first decomposes the quality of an individual decision: a group of tool-use-oriented methods represented by ToolRL break tool invocation into separately scoreable dimensions, including output format, tool selection, and parameter names and values [129, 487, 834]. The second distributes denser reward across the interaction process rather than scoring a single decision, spanning a spectrum of granularity: some works such as AdaTIR and AgentPRM assign reward to intermediate units of the trajectory rather than to its endpoint alone [77, 150, 318, 554, 682, 702, 837]. Among them, some methods fold process signals such as interaction cost and query quality into trajectory-level rewards, while others attach explicit step- or turn-level credit via process reward models or advantage redistribution. Across both routes, the shared goal is not merely to make rewards denser, but to align credit signals with the true decision structure of the agent.

Rubric Credit Assignment. Rubric credit assignment expresses answer or trajectory quality through explicit, multi-dimensional semantic criteria (e.g., factuality, completeness, reasoning soundness, evidence grounding, and safety) when a single verifiable metric is insufficient; a recent paper separates such rubrics from reward models, verifiable rewards, and LLM-as-a-judge [386]. Methods differ along two axes: how rubrics are constructed, and how they are turned into credit.

On construction, rubrics are produced along an increasingly automated spectrum: (1) strong LLMs or human experts author checklists used directly as scoring criteria or as training rewards [534], as in Rubrics as Rewards and the rubric bank of Reinforcement Learning with Rubric Anchors [192, 240]; (2) rubrics are induced from preference contrasts, where OpenRubrics extracts rules and implicit principles filtered by preference-label consistency and Chasing the Tail targets the high-reward tail to curb over-optimization [384, 819]; and (3) rubrics are made context-grounded and online, as when Agentic Rubrics lets an expert agent inspect a repository before generating criteria and DR Tulu co-evolves positive and negative rubrics with the policy during deep-research training [500, 532].

On conversion, rubrics become credit signals (1) as direct RL rewards, aggregated by weights or an LLM judge and optionally decomposed into fine-grained signals such as answer correctness, step validity, and premise

Table 6 Representative works in agentic reinforcement learning. Rows are grouped by the four parts of Section 5.5: credit assignment, policy optimization, sampling strategy, and interaction patterns; **Type** names the mechanism family, **Key Mechanism** gives its one-line idea, and **Resource** links to a public code repository where available.

Method	Type	Key Mechanism	Resource
Credit Assignment			
DeepSeekMath [533]	Outcome Reward	Terminal reward for math reasoning	GitHub
Search-R1 [261]	Outcome Reward	Answer correctness as terminal reward	GitHub
DeepRetrieval [256]	Outcome Reward	Retrieval-metric terminal reward	GitHub
ToolRL [487]	Process Reward	Decomposes tool calls into scored aspects	GitHub
Tool-Star [129]	Process Reward	Monitors answer and tool usage	GitHub
CriticSearch [837]	Process Reward	Retrospective critic for fine-grained credit	–
RaR [192]	Rubric Reward	Expert checklists used directly as rewards	–
OpenRubrics [384]	Rubric Reward	Rubrics induced from preference contrasts	–
DR Tulu [532]	Rubric Reward	Co-evolving rubrics in deep research	GitHub
RuscaRL [887]	Rubric Reward	Rubrics as decaying exploration scaffolds	GitHub
Policy Optimization			
REINFORCE++ [222]	Ratio Clipping	Global advantage normalization	GitHub
DAPO [780]	Ratio Clipping	Decoupled clip-higher/lower	GitHub
Dr.GRPO [389]	Ratio Clipping	Removes length/std normalization bias	GitHub
GSPO [865]	Ratio Clipping	Sequence-level ratio and clipping	–
Turn-PPO [321]	Credit Granularity	Turn-level advantage estimation	–
StepPO [628]	Credit Granularity	Step-aligned advantage	GitHub
GiGPO [156]	Credit Granularity	Episode- and step-level advantage	GitHub
Clip-Cov [103]	Entropy Regularization	Covariance-gated updates control entropy	GitHub
CURE [330]	Entropy Regularization	Critical-token-guided re-concatenation	GitHub
EPO [732]	Entropy Regularization	Trajectory entropy regularization	GitHub
Sampling Strategy			
R1-searcher [553]	Trajectory Rollout	On-policy search-agent trajectory rollout	GitHub
WebDancer [689]	Trajectory Rollout	Web-search trajectory pipeline	GitHub
WebRL [484]	Trajectory Rollout	Self-evolving curriculum web trajectories	GitHub
Tree of Thoughts [760]	Tree Sampling	Organizes thoughts as a search tree	GitHub
RAP [204]	Tree Sampling	Planning as tree search with world model	GitHub
LATS [875]	Tree Sampling	Language-agent tree search over actions	GitHub
TreePO [348]	Tree Sampling	Reuses inference compute across tree paths	GitHub
LiteResearcher [334]	Budget-Aware Pruning	Difficulty/pass-rate-based pruning	GitHub
TCOD [636]	Budget-Aware Pruning	Trajectory-depth curriculum	GitHub
Interaction Patterns			
GLIDER [232]	Hierarchical Framework	Grounds LLMs as decision-making agents	GitHub
SkillRL [705]	Hierarchical Framework	Evolves a skill library from failures	GitHub
THOR [52]	Hierarchical Framework	Joint episode- and step-level optimization	GitHub
AT-GRPO [861]	Multi-agent Framework	Per-agent, per-turn advantage grouping	GitHub
MATPO [439]	Multi-agent Framework	Multiple roles in one LLM over joint rollouts	GitHub
SPIRAL [371]	Multi-agent Framework	Role-conditioned advantages	GitHub
Memory-R1 [743]	Memory-state Learning	Separates memory manager and answer agent	GitHub
Memory-as-Action [846]	Memory-state Learning	Working-memory editing as policy action	GitHub
Agent Workflow Memory [674]	Memory-state Learning	Induces reusable workflows	GitHub
Agent Lightning [409]	Memory-state Learning	Traces workflows into trainable samples	GitHub

utilization [164, 192]; (2) as supervision for reward models or verifiers, including Prometheus, the OpenRubrics reward model, and the interpretable scorer R3 [19, 278, 384]; and (3) as guidance for policy behavior rather than only final scoring, as in Rubric-Scaffolded RL, which injects rubrics as decaying exploration scaffolds during rollout and as judge references during exploitation [887]. Rubric credit is especially useful in Agentic RL, where context dependence, tool interaction, and side effects are easily obscured by a single composite reward; its open challenges remain rubric quality, judge bias, criterion weighting, and reward hacking.

5.5.2 Policy Optimization

Policy optimization is where credit signals are turned into parameter updates. In Agentic RL, this step is difficult because the updates must be computed from multi-turn trajectories that mix tool calls, environmental observations, GUI states, and delayed terminal rewards [261, 321, 628, 681]. We organize the discussion around three practical questions: how to make sparse-reward updates stable, which interaction unit should be optimized, and how to preserve exploration over long trajectories.

Clipping-based Policy Optimization. The first problem is how to make PPO- or GRPO-style RLVR reliable when trajectories are long and rewards are sparse or binary [460, 524, 525, 533]. The main difficulty is the mismatch between noisy group statistics and long sequence-level gradients. This mismatch appears as length bias, local normalization bias, homogeneous all-correct or all-wrong groups, unstable token-level importance ratios, and reward-scale drift. Existing methods fall into two broad categories. One category makes group-relative updates less biased through normalization, clipping, and sampling design; representative examples include REINFORCE++, DAPO, Dr.GRPO, CISPO, and GPPO [222, 389, 435, 565, 780]. GSPO defines the importance ratio and applies clipping at the sequence level rather than per token [865]. The other category changes the estimation unit or objective to reduce variance and constrain policy drift, with representative examples such as BNPO [95, 205, 391, 710].

Turn-level Policy Optimization. The second problem is the granularity of optimization. In agent-environment interaction, the meaningful decision unit is often a trajectory, turn, step, subgoal, or branch rather than a single response. Early agentic RL systems therefore lift optimization from response-level training to full interaction trajectories, as in web and search agents such as WebAgent-R1 [261, 681]. A more fine-grained line then aligns advantages with explicit segments, steps, or turns, represented by SegPO, StepPO, and Turn-PPO [321, 628, 901]. More recent methods make the comparison set itself more local and informative. Some anchor comparisons to shared states or trajectory structures, as in GiGPO and related methods [156, 212, 318, 349]. Others use shared branches, subgoal segments, or hindsight information to ask which local choice changed the downstream trajectory, as in Tree-GRPO and its extensions [250, 478, 582, 687].

Entropy-based Policy Optimization. The third problem is exploration. Long-horizon RL can quickly over-reinforce early successful patterns, causing entropy collapse and reducing the diversity of future trajectories. Recent work therefore treats entropy as a decision- or trajectory-level control variable rather than only a token-level regularizer. One group modifies the update rule to avoid excessive probability concentration: Clip-Cov and KL-Cov gate over-concentrating updates, and OPEFO balances entropy-increasing and entropy-decreasing gradients [103, 725]. Another group injects diversity at critical decision points, for example CURE, which re-concatenates rollouts around critical (high-entropy) tokens to prevent entropy collapse [330]. Other methods, such as AEnt and AER, attempt to stabilize policy entropy within a bounded interval to alleviate entropy collapse [536, 830]. For multi-turn agents, the same idea becomes trajectory-level exploration control: EPO regularizes trajectory entropy under sparse rewards [732]. ARPO and AEPO branch from high-entropy positions after tool feedback to trade off rollout diversity against gradient distortion [128, 130].

5.5.3 Sampling Strategy

In long-horizon agent tasks, the sampling target is multi-turn interaction trajectories encompassing tool calls, environmental observations, memory operations and final rewards [808, 821]. Unlike credit assignment and policy optimization, the sampling strategy decides which trajectories are generated in the first place. We organize this from three angles: trajectory-level rollout that fixes the basic sampling unit, tree-structured

sampling that shares computation and produces relative signals, and budget-aware pruning that keeps sampling informative under a fixed cost.

Trajectory-level Rollout. The basic sampling unit in Agentic RL is a multi-step agent-environment trajectory: at each step the model acts on observations, history, and task goals, and the environment returns new observations and reshapes the future state distribution. Early systems represented by ReAct established this reasoning-acting-observing loop, together with tool use, verbal reflection, and skill reuse, mostly through prompting rather than training [523, 544, 630, 761]. Subsequent RL work turned such loops into the rollout unit: a line of information-seeking agents led by Search-R1 folds query generation, evidence acquisition, and stopping decisions into on-policy trajectories [256, 261, 553, 871], while web and tool agents organize browser operations or executable tool composition as multi-turn online trajectories [484, 653, 681, 701, 834]. At larger scale, deep-research systems represented by WebDancer and the WebSailor family build end-to-end pipelines that combine synthetic data, tool-based web environments, and trajectory-level rollout [326, 327, 689]. On top of such pipelines, WebResearcher, ReSum, and WebWeaver further address long-context research, context summarization, and evidence structuring during sampling [356, 490, 693].

Tree-structured Sampling. Independent full-trajectory rollout is expensive and, when trajectories are homogeneous, yields weak relative learning signals. Tree-structured sampling addresses both problems by sharing prefixes across rollouts and contrasting sibling branches. The idea originates from organizing thoughts or actions as trees during reasoning and planning, as in Tree of Thoughts, RAP, and LATS [204, 760, 875], and is brought into RL training by Tree-GRPO, which compares sibling branches under a fixed token or tool-call budget [250]. A subsequent group refines where and how the tree expands: AT2PO, TreePO, and PORTool guide expansion toward uncertain turns, reuse inference computation, or organize shared and divergent tool-use paths [348, 687, 900], while branching methods such as BranPO and Tool-Light branch selectively from high-entropy positions, late critical decisions, or otherwise informative states to maximize rollout contrast [76, 860].

Budget-aware and Pruning Strategies. A third angle keeps sampling informative under a fixed rollout budget, either by pruning or reshaping the trajectory distribution or by allocating the branching budget adaptively. One line prunes or reshapes the trajectory distribution: representative mechanisms include experience reuse and confidence-based filtering in Tool-R1 and WebRL, difficulty- or pass-rate-based pruning in LiteResearcher, and trajectory-depth curricula in TCOD [334, 484, 636, 834]. These strategies discard low-signal rollouts, concentrate budget on contrastive or appropriately difficult cases, and bound trajectory length. They reduce the cost of producing each informative training sample in long-horizon settings. Another research approach makes branching itself budget-aware: ARPO triggers branch sampling only at high-entropy tool-call steps, reducing the tool-use budget of trajectory-level RL, while AEPO adaptively allocates the budget between global and branch sampling [128, 130].

5.5.4 Interaction Patterns

Interaction patterns concern how to organize optimizable decision-making structures during training. We discuss them from three perspectives: hierarchical interaction frameworks, multi-agent interaction frameworks, and memory-state learning.

Hierarchical Interaction Frameworks. Hierarchical frameworks exploit the temporal abstraction that naturally exists in long-horizon tasks, and split along two threads. The first is explicit planner-executor decomposition, where the policy is separated into a high-level module that sets subgoals and a low-level module that executes them, so that subgoal boundaries become new optimization units. Representative instances include HiPER, GLIDER, and STEP-HRL, which differ mainly in whether the low-level layer is a task-agnostic skill set or a progress-compressed executor [232, 478, 863]. The second thread pushes hierarchy toward experience abstraction and capability reuse: building on the observation that procedural skills in agentic exploration originally rely on external skill libraries and prompting. SkillRL evolves a skill library from validation failures during RL and Skill0 internalizes skills into model parameters via in-context agentic RL [405, 705]. Analogous high-low decompositions appear in tool-augmented reasoning, where THOR jointly optimizes episode-level problem solving and step-level tool or code operations [52].

Multi-agent Frameworks. Multi-agent frameworks extend the training target from a single policy to a joint decision-making system. Their central difficulties are role non-stationarity, enlarged joint action spaces, and credit assignment under shared rewards. One thread formalizes LLM collaboration as cooperative MARL in Dec-POMDP settings and adapts group-relative training accordingly. MAGRPO uses centralized group-relative advantages with decentralized execution, while Stronger-MAS (AT-GRPO) restores variance reduction by grouping samples per agent and per turn [382, 861]. To assign credit across heterogeneous roles, M-GRPO computes separate group-relative advantages for a planner and its tool-executing sub-agents under decoupled cross-server training, whereas MATPO trains the planner and worker within a single LLM by normalizing over their joint rollouts [218, 439]. Dr.MAMR instead pushes credit to the step level with a Shapley-inspired causal-influence estimate to counter lazy-agent collapse [850]. A second thread covers self-play, debate, and role specialization, which inherit classical MARL principles such as centralized training with decentralized execution, counterfactual baselines, value decomposition, and MAPPO [396, 506, 776]. Within this thread, recent systems instantiate complementary System 1/System 2 policies for deep research, variance-reduced group-relative training with evolving memory, and role-conditioned advantage estimation in zero-sum language games [56, 201, 371]. Multi-agent debate adds a further variant, where competitive interaction provides additional preference, refutation, and verification signals for the final answer [139, 358].

Memory-state Learning. Memory-state learning turns runtime structures that are usually fixed by an external harness, including context, workflows, long-term memory, tool-use histories, and reusable experience, into learnable policies. This shifts Agentic RL from optimizing isolated LLM calls to training state management over long horizons. One thread makes memory operations themselves a learning target: Memory-R2 observes that memory writes place trajectories in different effective environments and uses LoGo-GRPO to combine a global objective with local comparisons from the same memory state, Memory-R1 separates a Memory Manager from an Answer Agent, and Memory as Action treats working-memory editing as an explicit policy action handled by Dynamic Context Policy Optimization [742, 743, 846]. A parallel thread lifts the same idea to the organization of the task process: Agent Workflow Memory induces reusable workflows from successful tasks, FlowSteer optimizes a workflow director with structural and task-level rewards, and Agent Lightning offers a system-level route that traces existing workflows as spans and transitions and converts them into trainable samples [409, 674, 823].

5.6 On-Policy Distillation

Alongside the emergence of Agentic RL, recent research has increasingly explored alternative training paradigms tailored to long-horizon trajectory optimization. As interaction horizons grow, agent training becomes increasingly challenged by distribution shift, error accumulation, and sparse, delayed supervision [377]. Existing training paradigms only partially address these challenges. Off-policy imitation, such as behavioral cloning and offline knowledge distillation [609], relies on static expert trajectories and therefore fails to expose agents to their own induced state distributions. Reinforcement learning [94], in contrast, optimizes on on-policy trajectories but often suffers from sparse rewards, costly exploration, and poor sample efficiency.

On-Policy Distillation (OPD), first formalized for language models by GKD [3], combines the high-quality supervision of distillation with the state coverage of on-policy learning [309, 635]. Instead of learning solely from fixed expert demonstrations or sparse outcome rewards, OPD trains the student on trajectories generated by its current policy and provides additional supervisory signals on these self-generated trajectories. As a result, OPD exposes the student to self-induced state distributions while providing richer supervision than sparse outcome rewards. This delimits OPD from its neighbors: unlike the offline trajectory distillation of Section 5.4, which learns from a fixed set of teacher or past-model traces, OPD draws supervision on the student's own current rollouts; unlike the self-evolution of Section 5.7, it keeps the task set and environment fixed and only densifies the supervision signal. These properties make OPD particularly attractive for long-horizon agents, where robustness to distribution shift, recovery from compounding errors, and access to dense supervisory signals are all critical for effective learning [697, 838].

5.6.1 Teacher-guided OPD

In teacher-guided OPD, one or more teacher models are queried on states visited by the student policy to provide supervisory signals for the generated trajectories [309, 312, 635, 697]. Unlike conventional offline

distillation that relies on fixed expert demonstrations, teacher-guided OPD generates supervision adaptively online according to the student's evolving trajectories, allowing teachers to provide guidance tailored to the student's current behaviors and mistakes. Existing methods can be characterized by two complementary design choices: how supervision is organized over trajectories and how teacher signals are constructed.

Supervision Scheduling. A first question is how teacher signals should be organized and delivered throughout long interaction trajectories. Representative methods such as DAgger-LLM introduce turn-level interpolation and step-wise supervision to provide dense corrections on states visited by the student policy [309, 873]. Other approaches further incorporate adaptive curricula and preference-based guidance, progressively adjusting the timing, granularity, and difficulty of supervision during training [313, 636, 835]. What emerges is that teacher-guided OPD hinges not only on the quality of teacher signals, but also on when and at what resolution they are injected during multi-turn interactions.

Teacher Signals. A second question is how to construct richer and more informative supervision for on-policy distillation. A series of methods represented by MAD-OPD leverage multiple agents or agent evolvers to aggregate complementary expertise and generate more reliable supervision than a single teacher policy [635, 809]. Along a related line, KAT-Coder-V2 [312] exploits specialist agents with complementary capabilities and distills their collective knowledge into a unified policy. Retrieval-augmented approaches such as LiteGUI [697] instead enrich teacher supervision with oracle trajectories and dynamically retrieved references, providing structured guidance in multi-solution interaction environments. Across these efforts, the recurring lesson is that supervision improves when teacher signals are made more diverse, complementary, and context-aware.

5.6.2 Self-improving OPD

Self-improving OPD removes direct reliance on external teacher models and instead constructs its own supervisory signals from auxiliary information available during training. Such information generally arises from two complementary sources: privileged information that is accessible only during training [838] and adaptive feedback generated through repeated interactions with the environment [216, 242, 845].

Privileged Information. One source of self-supervision is information that is accessible during training but unavailable at deployment time. Some methods leverage privileged annotations or modalities, such as outcome-aware information and fine-grained visual details, to provide dense supervision beyond what the agent's own observations reveal [788]. Others derive privileged information from the interaction trajectories themselves, treating intermediate construction processes, latent skills, or other trajectory-level abstractions as additional supervision that exposes the hidden structure of successful behaviors [631, 838]. Augmenting sparse interaction signals with training-only information in this way helps the student acquire more robust and reusable long-horizon behaviors.

Environment Feedback. The other source of self-supervision is the interaction outcomes and verifier signals returned by the environment. Some methods transform sparse environmental feedback, such as rewards, textual critiques, runtime errors, and verifier outputs, into dense self-supervision by conditioning the policy on feedback-informed predictions and distilling them back into the model [216, 242]. A complementary view treats failures as supervision, generating retrospective reflections and accumulating reusable lessons that guide future interactions [757, 845]. Beyond obtaining feedback, a further line studies how to use it well during distillation, for example by emphasizing informative signals, learning feedback-aware representations, and mitigating the instability caused by noisy or stale supervision [72, 161, 735, 782]. The decisive factor is therefore not whether environmental feedback is available, but how it is transformed, accumulated, and reused across long-horizon training.

5.7 Self-evolution

This subsection discusses self-evolution as a training mechanism for internalizing experience generated by the agent itself. The preceding stages mainly rely on prepared data, task sets, environment collections, or trajectories. Self-evolution adds a feedback loop: the current policy produces new attempts, failures,

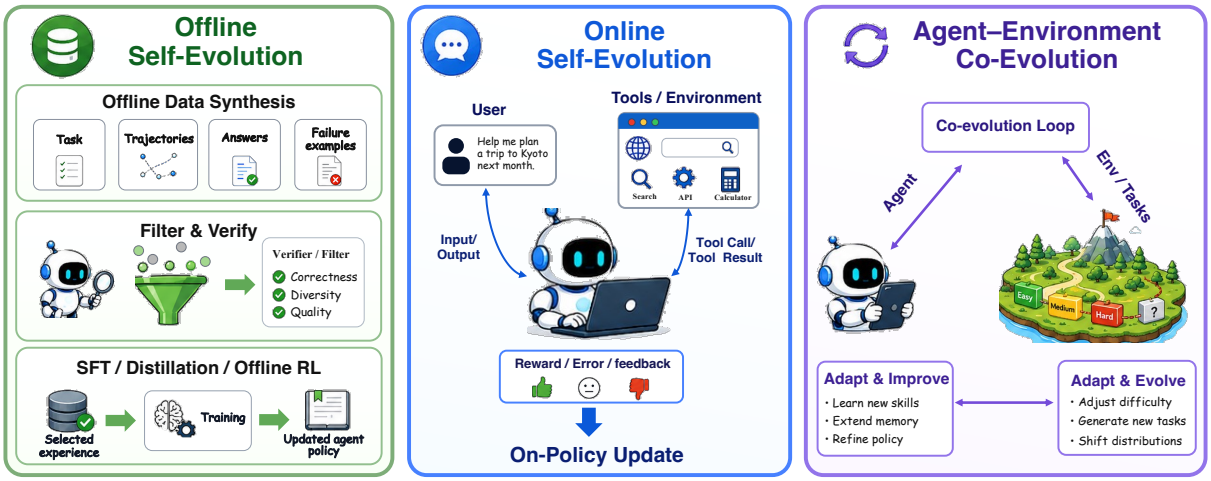


Figure 13 Three modes of self-evolution for long-horizon agents. *Offline self-evolution* learns from selected self-generated experience through a generate–filter–train pipeline; *online self-evolution* learns from live environment interaction on the agent's own induced state distribution; and *agent–environment co-evolution* jointly adapts the agent and its task/environment distribution to maintain a learnable capability boundary.

and feedback in its environment, and training writes selected records back into the model. This is what distinguishes it from the two distillation stages above: whereas fine-tuning distillation (Section 5.4) and on-policy distillation (Section 5.6) both consume a supervision source that is held fixed within a training round, self-evolution treats the data, tasks, and even the environment as objects that the loop itself updates. This has recently grown into a fast-moving research area, and dedicated papers now catalog how agents can autonomously evolve their models, memories, tools, and workflows toward lifelong adaptation [147, 165]. As illustrated in Figure 13, we organize this line of work into three forms: offline self-evolution, online self-evolution, and agent–environment co-evolution.

5.7.1 Offline Self-evolution

Offline bootstrapped self-evolution typically follows a generate–filter–train pattern. The model generates candidate tasks, answers, reasoning traces, code patches, tool calls, or interaction trajectories; verifiers, tests, reward models, teacher models, or self-evaluation mechanisms select successful samples, informative failures, and correctable fragments; training then turns these samples into SFT, distillation, preference optimization, or offline RL objectives. STaR bootstraps reasoning from successful rationales [793], and subsequent work extends the same idea to self-edits, tool-augmented reasoning, multi-agent demonstrations, and code-execution feedback [310, 517, 859, 906]. More recent work emphasizes how to turn search processes and failures into stable training assets. SOAR alternates evolutionary search with hindsight learning in program synthesis, converting search attempts into fine-tuning pairs; SAMULE synthesizes reflection data at the trajectory, task, and cross-task levels, and trains a retrospective model from failure analysis [173, 481]. A parallel search-based route runs the generate–filter–train loop for several rounds: rStar-Math uses Monte Carlo tree search to synthesize step-by-step verified reasoning trajectories and iteratively co-evolves a policy model together with a process preference model, so that each round's verified rollouts become the next round's supervision [191]. This route is attractive because data can be audited, denoised, and relabeled before an update, making it suitable for cold start and capability consolidation. Its main risk is that candidate data still largely comes from the existing model distribution, which can reinforce known behaviors while doing little to discover out-of-distribution strategies.

5.7.2 Online Self-evolution

Online interactive self-evolution interleaves experience collection with policy updates, training the model on state distributions induced by its own behavior. For long-horizon agents, difficult states are often not static

benchmark inputs; they arise from early decisions, tool side effects, recovery attempts, and environment state changes. Recent work therefore incorporates web, code, search, tool-use, and multi-agent game environments into training, using rollout, environment feedback, and reinforcement learning to continually update the policy [484, 672, 701, 799, 871]. A complementary line reduces external data seeds by using self-play, challenger–solver, or proposer–solver structures to create training pressure, so that the model obtains learning signals by proposing tasks, solving them, and verifying outcomes [233, 371, 854]. Recent agentic instantiations push this toward tool use and multi-step reasoning: Agent0 co-evolves a curriculum agent and a tool-using executor from the same base model with zero external data, so that a stronger executor pressures the curriculum agent to propose harder, tool-aware tasks [704]. Compared with offline bootstrapping, online self-evolution is more likely to move beyond the ceiling of existing data, but it is also more sensitive to the quality of environments and verifiers: delayed rewards, credit-assignment errors, policy collapse, verifier loopholes, and exploration cost are all amplified in long rollouts.

5.7.3 Agent–Environment Co-evolution

Agent–environment co-evolution further treats the training environment as an object of update. A fixed environment can quickly drift away from the model's capability boundary: if it is too easy, it only reinforces existing strategies; if it is too hard, it yields little usable learning signal. A first line makes the environment itself tunable to the policy: Environment Tuning uses curricula, environment augmentation, and progress feedback to mitigate cold start and sparse rewards in tool-use training, while RLVE and GenEnv align the difficulty of verifiable or generated environments with the policy's current capability, and Agent-World closes the loop by discovering environment–task pairs, exposing capability gaps through dynamic task synthesis, and improving the policy through multi-environment RL [131, 195, 401, 798]. In multi-agent settings, the environment may also be constituted by the policies of other agents; CoMAS uses interaction rewards to jointly update multiple agent policies, showing that peer interaction can serve as a training signal for self-evolution [738]. Socratic-Zero pushes this idea in a data-free direction, letting a Teacher, a Solver, and a Generator co-evolve so that the Teacher continually crafts questions targeting the Solver's weaknesses while the Generator distills its curriculum-design strategy, effectively turning the task distribution itself into a co-adapting counterpart [648]. From this perspective, the environment is not merely a source of samples, but a dynamic curriculum, feedback generator, and probe of the capability boundary: improvements in the agent change future reachable states and failure modes, while environment updates reshape the next round of exploration and optimization.

Overall, training self-evolution is not unconstrained self-training, but an experience loop jointly constrained by environments, verifiers, and optimization. External memories, skill libraries, and workflow traces can serve as intermediate forms of accumulated experience, but they constitute training self-evolution only when their stable patterns are absorbed through distillation, fine-tuning, reinforcement learning, or trainable modules [705]. The risks of this loop arise from its bootstrapping structure: self-generated data may shrink the distribution, weak verifiers may induce reward hacking, and dynamic curricula may reinforce flawed capabilities. Long-horizon agent self-evolution therefore requires strong verifiers, process-level feedback, and a combination of offline audit with online exploration.

6 Application: Long-Horizon Agents in Practice

The preceding sections examined the general ingredients of long-horizon agency, including persistent state, feedback-driven adaptation, and mechanisms for maintaining alignment over extended trajectories. We now turn to the settings in which these requirements arise in practice. Long-horizon agents operate across a wide range of environments, from code repositories and web documents to graphical interfaces, multimodal streams, and open-ended research workflows. These settings differ not only in task duration, but also in the interface through which an agent observes, acts, and receives feedback from the world, which in turn shapes what information is available during execution, what errors can be detected locally, and what capabilities the agent must add to compensate for what the environment does not provide. Therefore, we organize this section by the interface between the agent and its environment rather than by overall difficulty.

Figure 14 groups representative applications into five broad categories, distinguished by the interface through which each agent observes and acts:

Five Representative Applications of Long-Horizon Agents

1. **Software Engineering Agents** (Section 6.1): act over code repositories and execution environments, where edits can often be checked through tests, compilers, and runtime outputs.
2. **Information-seeking Agents** (Section 6.2): interact with web documents, search engines, and knowledge sources, where the challenge is to gather and synthesize evidence without a direct execution oracle.
3. **Computer-use Agents** (Section 6.3): operate graphical interfaces through screenshots, accessibility trees, and low-level actions, inferring and manipulating partially observable interface state.
4. **Multimodal Agents** (Section 6.4): reason over long visual, auditory, and textual streams, where relevant information is distributed across time and modalities.
5. **General-purpose Agents** (Section 6.5): combine several such interfaces within a single workflow, drawing on heterogeneous tools, permissions, and external services.

For each category, we ask three questions:

- What information and actions does the interface expose?
- What kinds of state and feedback must persist across a long trajectory?
- What capabilities must the agent or harness add to compensate for what the environment does not provide directly?

The following subsections address these questions by discussing the core capabilities required in each domain and representative systems that instantiate them.

6.1 Software Engineering

Software engineering agents are one of the most foundational and representative classes of long-horizon agents. The agent interacts with its environment through code: it acts by writing and running code, obtains feedback from how that code behaves, and uses this feedback to advance its task. A defining feature of this paradigm is unusually strong executable feedback: through tests, compilers, and runtime traces, the environment can often tell the agent automatically whether an action has succeeded, which is what distinguishes software engineering from many other long-horizon tasks. These signals nonetheless remain incomplete proxies for intended behavior: passing tests and clean compiles cannot always certify that the real issue is resolved, especially where hidden requirements, security, maintainability, and regression risk are concerned [499, 671, 801].

Turning this feedback into reliable long-horizon behavior, however, takes more than the signal itself. We organize long-horizon software engineering around three capabilities that an agent must add. The agent first has to work out *where* to act, building a usable picture of a codebase too large to fit in its context window; we call this *repository grounding*. It then has to keep the overall objective intact as it proceeds, ordering its edits and tracking progress rather than chasing one fix at a time, which is *workflow-level planning*. Finally, it has to use the feedback it receives to actually correct its mistakes, the task of *feedback-driven repair*. Together, these capabilities mark the step up from fixing isolated bugs to repository-scale work [124, 749], including implementation, migration, and pull-request preparation; commercial systems such as Devin[2] brought such end-to-end long-horizon coding agents into production. The next subsections examine each capability in turn.

6.1.1 Repository Grounding

Before any code is written, the agent must decide *where* to act. A natural-language issue typically says little about which files, dependencies, or project conventions are relevant, so long-horizon systems need a way to progressively construct and reuse a working model of the repository.

[2]Devin (Cognition): https://devin.ai.

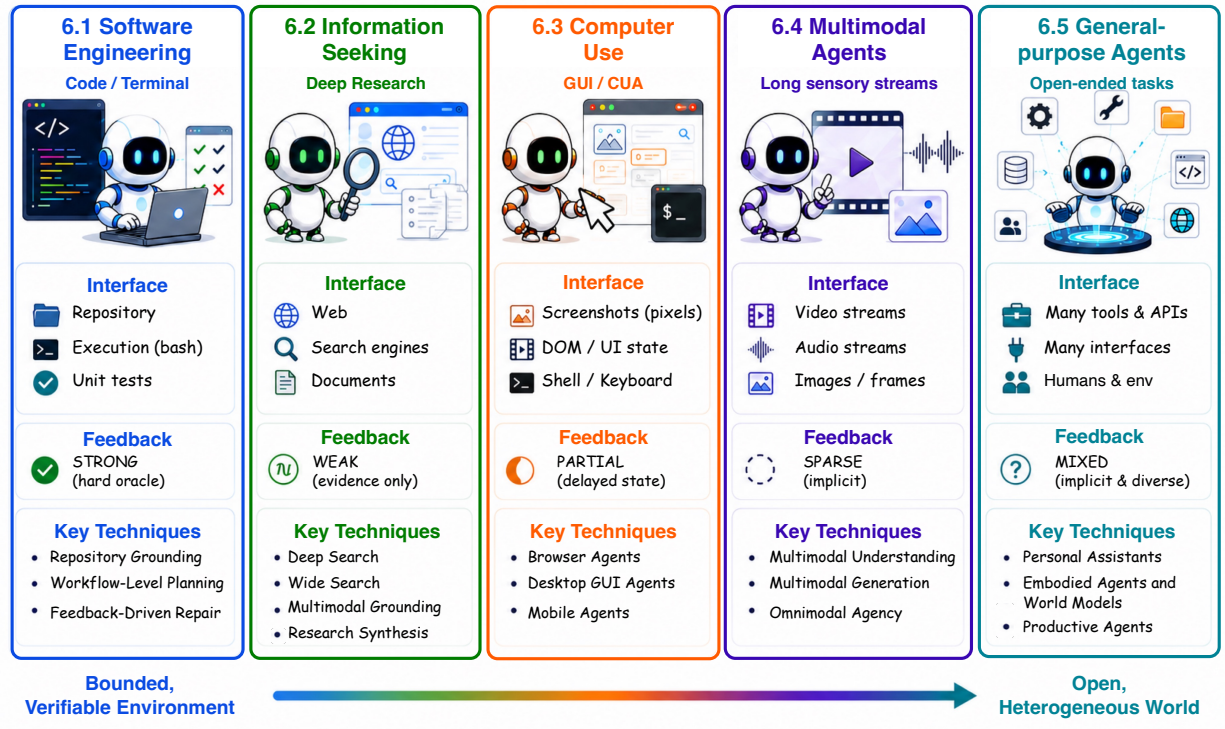


Figure 14 Representative long-horizon agent applications grouped by the structure of the agent–environment interface. Different interfaces expose different forms of observation, action, state, and feedback, leading to distinct capability requirements for long-horizon behavior.

Existing designs largely follow three strategies. The first is to improve *interactive navigation*: SWE-agent, for example, exposes search, navigation, and editing commands through an agent–computer interface tailored to LLM use [747], while structure-aware systems such as AutoCodeRover search over a program's class and method hierarchy rather than over raw text [843]. The second is to maintain a *persistent workspace*, so that file, shell, and browser state can accumulate across steps rather than being repeatedly reconstructed; OpenHands is representative of this approach [657]. The third is to externalize repository structure in compact, reusable artifacts, such as Aider's repository map,[3] IDE-level project rules,[4] and `AGENTS.md`-style instructions.[5] Although implemented differently, these approaches share a purpose: turning a large, partially observed codebase into stable task-relevant state that can guide later decisions. Related work extends this idea by constructing executable environments from arbitrary repositories [248], or by training models to internalize more repository-level structure directly [46].

6.1.2 Workflow-level Planning

Knowing where to act does not by itself determine how to sequence the work. Feature development, dependency migration, experiment reproduction, and pull-request preparation require the agent to decompose an objective, track intermediate progress, manage artifacts, and decide when available evidence is sufficient to stop [117, 142, 260, 317, 437]. The challenge here is to preserve the relationship between each local action and a global engineering objective.

Long-horizon systems increasingly externalize this relationship through explicit workflow state. One line of work structures execution around branches, plans, checkpoints, and approval gates, so that progress can be reviewed and resumed rather than remaining implicit in a conversation history.[6] A second line factors

[3] Aider: https://github.com/Aider-AI/aider.
[4] Cursor Rules: https://cursor.com/docs/rules.md.
[5] AGENTS.md: https://agents.md/.
[6] Codex, *Agent Approvals & Security*: https://developers.openai.com/codex/agent-approvals-security. Anthropic, *Claude*

recurring orchestration functions, such as planning, file-system state, tool invocation, and delegation, into reusable runtime components, as in OpenClaw and Deep Agents [293, 457]. A third line extends the workflow beyond a single coding loop by coordinating specialized agents across browsing, terminal use, file manipulation, context management and code editing [157, 381]. At deployment scale, managed execution systems further separate session-level reasoning from sandboxed execution, allowing long-running work to be supervised, audited, and resumed [18, 561]. Across these designs, the common goal is to make the task trajectory itself a persistent object of control rather than an unstructured sequence of model actions.

6.1.3 Feedback-driven Repair

Within each step of such a plan, the agent must turn execution feedback into revisions without mistaking local evidence for complete correctness. Tests and compiler errors make failures visible, but they do not fully specify the intended behavior, and an agent may repeatedly patch the most recent symptom, overfit to visible tests, or introduce a regression elsewhere in the repository.

Systems address this problem at both training and execution time. At training time, executable environments and repair trajectories teach agents to associate failures with plausible diagnosis-and-revision behavior, as in SWE-Gym and SWE-smith [467, 748]. At execution time, the central principle is to make repair *inspectable and reversible*: agents preserve diffs, logs, execution traces, and isolated workspaces so that failed attempts can be diagnosed rather than simply overwritten [657, 747]. Production-oriented systems extend this loop with branch isolation, hooks, approval boundaries, rollback, and final verification gates.[7] The result is more than iterative debugging: the agent accumulates evidence across failed attempts while retaining enough control to revise or discard earlier decisions. Not all of this requires an open-ended agent loop, however: Agentless shows that a fixed localize-then-repair pipeline can rival more elaborate agentic systems [703].

6.2 Information Seeking

Information-seeking agents are a prominent class of long-horizon agents, because answering complex information needs often requires progressively accumulating evidence over extended retrieval trajectories, where relevant information is distributed across heterogeneous sources and only partially observable at any individual search step. We organize information-seeking agents around four capabilities: *deep search* sustains a single line of inquiry across many retrieval hops; *wide search* coordinates numerous parallel investigations under a shared objective; *multimodal grounding* extends retrieval beyond text to evidence in documents, figures, and images; and *research synthesis* organizes dispersed evidence into coherent, well-attributed research reports.

Together, these capabilities define the progression from retrieval-centric systems to long-horizon information-seeking agents. Rather than locating isolated facts, such agents must continuously acquire, compress, verify, and integrate evidence across extended trajectories [284]. They coordinate multiple investigations, maintain evolving research state, synthesize findings into structured outputs, and ground conclusions in heterogeneous sources. The resulting challenge is no longer search alone, but the end-to-end management of evidence throughout the entire research lifecycle [239].

6.2.1 Deep Search

Deep search concerns sustained investigation along a single line of inquiry, where relevant evidence may only emerge after many rounds of retrieval, navigation, and reasoning [223, 883]. As search trajectories lengthen, maintaining useful state becomes increasingly important. Deep search agents can be broadly categorized into two paradigms according to how they optimize long-horizon search behaviors. The first paradigm consists of RL-based deep search agents, which explicitly optimize multi-step search policies through reinforcement learning [367]. Existing efforts mainly improve these systems from two complementary directions. One line focuses on constructing challenging long-horizon search tasks that encourage models to acquire complex search and reasoning behaviors [586, 689], while another develops more effective reinforcement learning algorithms to stabilize exploration, credit assignment, and long-term planning [120, 130, 682]. The second paradigm comprises workflow-based deep search agents, which rely on carefully designed reasoning and retrieval

Code documentation: https://code.claude.com/docs/en/overview.

[7] Anthropic, *Claude Code Hooks reference*: https://code.claude.com/docs/en/hooks.

pipelines rather than policy optimization. A series of methods represented by ReSum therefore compress accumulated interaction histories into compact summaries or latent states that can be repeatedly reused across long trajectories [530, 693], while another class of approaches focus on condensing retrieved documents into task-relevant evidence representations for downstream reasoning [106, 336]. Long search chains also make retrieval failures harder to distinguish from reasoning failures [772] and amplify the risk of propagating incorrect assumptions or low-quality evidence [480]. Representative systems such as WebSeer continuously reassess intermediate conclusions, identify unsupported claims, and trigger additional retrieval when evidence remains incomplete or inconsistent [207, 332, 746]. Through these mechanisms, retrieval becomes a persistent evidence-accumulation process that supports deep investigation over extended horizons.

6.2.2 Wide Search

Wide search addresses research objectives whose scope exceeds a single search trajectory. Rather than pursuing one chain of evidence to completion, agents must coordinate many partially independent investigations while maintaining coverage of a large information space. A common strategy is task decomposition, where broad objectives are recursively broken into manageable research questions, evidence requirements, or execution plans. Explicit planning further enables agents to revise information needs and adapt research trajectories as new evidence emerges during execution [227, 292, 650, 746]. Increasingly, decomposition is coupled with parallel exploration. Different agents or search branches investigate complementary hypotheses, domains, or source collections before aggregating findings into a shared research state. Systems such as FlowSearch demonstrate that parallel search can substantially improve coverage and reduce the effective depth of individual trajectories [227, 241, 295, 556, 736]. Beyond parallel exploration, recent methods increasingly organize the search process as structured graphs, trees, or tables, which explicitly maintain search states and support dynamic expansion and revision as exploration proceeds [227, 289, 292]. The resulting challenge shifts from discovering isolated facts to coordinating large-scale evidence collection under a unified objective.

6.2.3 Multimodal Grounding

Multimodal grounding extends information seeking beyond textual retrieval. Many long-horizon research tasks require agents to gather evidence scattered across images, videos, and visually rich documents [114, 414, 714], where multilingual visual-text comprehension, document parsing, and OCR provide the perceptual access needed when text-based retrieval alone is insufficient [159, 290, 459, 584]. Recent work has therefore expanded information-seeking agents toward image search, enabling agents to retrieve, localize, and interpret visual evidence embedded in web-pages or documents, such as screenshots, diagrams, charts, and figures, while grounding subsequent reasoning in visual observations rather than textual descriptions alone [67, 138, 175, 346, 802, 825]. Another line of work focuses on video search, where agents actively explore long videos to identify answer-critical segments through iterative reasoning and interaction, replacing exhaustive frame-by-frame parsing with adaptive evidence seeking over temporal visual content [146, 361, 372]; the underlying active-perception and video-memory systems are treated in Section 6.4. These advances extend information seeking from text-centric retrieval toward unified search over heterogeneous sources.

6.2.4 Research Synthesis

Research synthesis focuses on transforming large collections of retrieved evidence into coherent research artifacts. In long-horizon settings, information gathering alone is insufficient; agents must organize findings accumulated across many searches, identify relationships among sources, and construct narratives that remain faithful to the underlying evidence. Its central application is deep-research report generation, where agents synthesize retrieved evidence into comprehensive reports for domains such as open-domain research, finance, and business analysis [136]. Beyond research prototypes, widely deployed commercial systems such as OpenAI Deep Research, Gemini Deep Research, and Perplexity have brought this capability to broad audiences. Representative systems progressively externalize intermediate artifacts, including outlines, section plans, research notes, and persistent workspaces, before composing final reports [337, 356, 827, 891]. They further maintain explicit evidence attribution so that readers can trace individual claims back to their sources, improving both faithfulness and auditability [292, 347, 621]. The core challenge is therefore not merely to summarize retrieved documents, but to preserve coverage, narrative coherence, factual support [434, 677], and source-level traceability across a long search trajectory.

6.3 Computer Use

Computer-use agents interact with stateful software systems in which actions have persistent consequences and task completion requires sustained execution across extended interaction trajectories. These agents autonomously manipulate desktops, browsers, and applications through heterogeneous interfaces including screenshots, DOMs, accessibility trees, shells, executable code, and file systems [126, 588]. Recent systems such as Anthropic's Computer Use and OpenAI's Operator support planning, multi-tool coordination, and persistent execution state across long-horizon tasks [12, 452, 745].

We organize long-horizon computer use according to the execution environment: *browser agents*, *desktop GUI agents*, and *mobile agents*. Browser agents focus on sustained task execution within web environments through navigation and information extraction. Desktop GUI agents emphasize reliable visual grounding and interaction with operating systems and desktop applications. Mobile agents further extend long-horizon execution to smartphone applications, where agents must coordinate touch-based interaction across diverse mobile interfaces. Together, these directions characterize the evolution from browser automation to general-purpose computer use. Beyond executing actions within a single application, modern computer-use agents must coordinate perception, reasoning, and execution across heterogeneous software environments while maintaining persistent state and coherent progress toward long-running objectives.

6.3.1 Browser Agents

Browser agents investigate long-horizon interaction within web environments through navigation, information extraction, and form manipulation. Benchmarks such as WebArena, Mind2Web, and ClawBench establish browser interaction as a representative long-horizon computer-use setting, requiring agents to interpret page content, choose actions, and maintain progress across evolving interface states [116, 847, 885]. To support sustained execution, many systems expose browsers through reusable action primitives such as clicking, typing, scrolling, navigation, and form completion, allowing agents to reason over browser interaction as a structured action space rather than raw interface manipulation [210, 226, 404, 443, 781]. At the same time, accumulated observations, retrieved information, and intermediate decisions are increasingly externalized through memories, notes, summaries, or reusable knowledge stores that persist across long interaction trajectories [637, 781]. These developments progressively transform browser interaction from isolated page-level decisions into persistent task execution over extended horizons.

6.3.2 Desktop GUI Agents

Desktop GUI agents study how autonomous systems interact with operating systems and graphical desktop environments through multimodal perception and action. Benchmarks such as OSWorld and Workflow-GYM, together with representative systems including UI-TARS, have established desktop interaction as a realistic long-horizon setting, requiring agents to perceive interface state, execute multi-step workflows, and recover from execution failures across diverse applications [494, 526, 551, 716, 745]. Visual grounding plays a central role in desktop interaction by maintaining reliable correspondence between evolving interface observations and executable actions. Because graphical layouts can be both dense and dynamic across applications and execution steps, agents must identify relevant elements in high-resolution interfaces [324], interpret their functional roles, and update action targets throughout long interaction trajectories [181, 691]. Modern systems further complement multimodal perception with execution verification mechanisms based on screenshot comparison, visual change detection, semantic consistency checks, and artifact inspection [12, 259, 518], together with safety guardrails such as permission boundaries and confirmation policies [12, 455] to improve robustness under execution uncertainty.

6.3.3 Mobile Agents

Mobile agents extend long-horizon computer use to smartphones and tablets, where agents must coordinate interactions across multiple applications through touch-based interfaces [282, 373, 570]. Compared with desktop environments, mobile tasks typically involve long multi-app workflows, limited screen space, frequent context switching, and gesture-based operations, making persistent planning and state management particularly important [667, 766]. Recent systems increasingly adopt hierarchical agent architectures that separate high-level planning from low-level UI execution, allowing complex user requests to be decomposed into executable

subtasks while maintaining progress across extended interaction trajectories [543, 667, 723, 766]. At the same time, long-term memories, reusable interaction skills, and execution verification mechanisms are incorporated to improve robustness and enable continual improvement from previous experiences [105, 571, 723, 766]. These developments transform mobile agents from reactive UI operators into persistent assistants capable of accomplishing complex real-world workflows across diverse mobile applications.

6.4 Multimodal Agents

Multimodal agents are long-horizon agents that must process information from different modalities. In long-horizon scenarios, multimodal evidence may be distributed across long visual contexts or multiple videos [55, 320, 413, 666], requiring agents to locate and acquire relevant information, store it, reason over it, and integrate it into an answer or multimodal output. Recent benchmarks such as AgentVista and OmniGAIA mainly evaluate agents' ability to solve cross-modal tasks [339, 564, 589], rather than focusing only on isolated visual question answering or reasoning [45, 85, 190, 465].

We organize long-horizon multimodal agents into three categories according to their tasks and capabilities: *multimodal understanding*, *multimodal generation*, and *omnimodal agency*. Multimodal understanding focuses on how to locate useful information over extended timelines, including deciding which segments to inspect and which perception tools to invoke. Multimodal generation focuses on how to understand user intent and conduct multi-round planning, generation, and refinement, a process that relies on external evidence, verification mechanisms, and persistent memory. Omnimodal agency focuses on the coordination of different modalities so that they can jointly accomplish an end-to-end task. Overall, these three types of capabilities enable agents to handle richer multimodal information and interact more deeply with the real world.

6.4.1 Multimodal Understanding

For long-horizon multimodal understanding, agents must possess the ability of *active perception*. Under limited budgets, it is impossible to read all information streams at once; truly important information often appears only in a few frames of a long video, a small region of an image, or a single line of text in a document. Therefore, agents should be able to actively locate useful information within long information streams.

Current designs follow two main lines. The first line studies *tool-augmented active perception*. Works represented by VideoSeek treat long videos and images as explorable spaces rather than inputs that are fed into the model all at once [138, 365, 655, 825, 903]. The agent dynamically invokes browsing and retrieval tools according to the current question and existing clues, progressively locating key evidence in long videos. The second line studies *memory-augmented evidence organization*. Representative works such as DVD [510, 739, 829] construct structured video memory and then retrieve relevant evidence according to task demands. The former emphasizes active exploration strategies, while the latter emphasizes structured representations of long information streams and evidence preservation [796]. Their shared goal is to avoid blindly reading all multimodal inputs under limited budgets.

6.4.2 Multimodal Generation

In long-horizon scenarios, multimodal generation is not a one-step mapping from a prompt to an image or video clip. Agents usually need to understand user intent, plan the generation process, and conduct multiple rounds of generation and verification. Therefore, multimodal generation is a long-term task that must be completed through continuous iteration. Recent multimodal generation models, such as those in the Qwen-Image, Seedream, and SeedEdit series, have significantly improved intent understanding, multimodal creation, image editing, and video synthesis capabilities [73, 115, 172, 528, 600, 645, 855]. However, for long-horizon agents, beyond the generative capability of the model itself, the design of the surrounding harness is more important for ensuring that generated content remains consistent with user requirements.

Therefore, three mechanisms are crucial. *Evidence seeking* provides grounding for generation by retrieving images, documents, and videos. This is especially necessary in knowledge-intensive multimodal generation tasks, where sufficient task-relevant information must be collected. Recent generative agents autonomously invoke retrieval and visual analysis tools to gather task-relevant evidence before generation [68, 209, 255]. *Verification* checks generated results from multiple dimensions to ensure that they remain aligned with

the original goal [574, 741, 831]. *Memory* stores user requirements and evidence at different granularities, enabling subsequent generation to remain consistent with earlier decisions [71, 733]. Recent multimodal generation systems increasingly organize planning, evidence collection, memory, verification, and rendering into unified agentic workflows. For example, PTAH coordinates specialized agents throughout the research and writing lifecycle, maintains structured visual working memory, and enforces cross-modal consistency through verifier-guided report generation [803]. Similarly, long-form audio-video systems such as MUSE employ planning-execution-verification-revision loops, transforming multimodal generation into a long-horizon iterative process [574].

6.4.3 Omnimodal Agency

Omnimodal tasks require agents to coordinate across different modalities, because a single modality or a single model is often insufficient to complete the task. This is closer to real-world scenarios, where inputs may include text, images, speech, audio, video, and other modalities. Therefore, agents need the ability of *cross-modal orchestration*. Specifically, an agent needs to decompose the task, route subtasks to appropriate specialized models, and then verify and integrate their outputs.

Recent work has developed along two complementary directions. One direction focuses on native omnimodal models, such as those in the Qwen-Omni series, which provide unified interfaces for multimodal understanding or generation and usually support streaming interaction and speech-centered multimodal dialogue [498, 726, 727]. The other direction focuses on orchestration-based agents, such as OmniAgent, Orchestra-o1, and OmniGAIA, which explicitly assign subtasks to different modalities and coordinate their execution processes and outputs [137, 339, 364, 718, 728, 806]. In long-horizon omnimodal tasks, information from different modalities may conflict with each other, so explicit verification is required. Omnimodal agent capabilities enable agents to interact more deeply with real-world environments and broaden their range of applications.

6.5 General-purpose Agents

General-purpose agents represent one of the broadest forms of long-horizon agency, because they must pursue open-ended goals across heterogeneous interfaces rather than operate within a single fixed environment. A user request may require an agent to interact with multiple interfaces and invoke different types of tools. The central challenge for such agents is therefore *sustained coordination*: an agent must conduct overall planning for the task, maintain and transfer state across different environments, and recover from failures during execution. Benchmarks such as GAIA and τ-bench reflect a recent shift toward evaluating whether agents can consistently complete tasks in heterogeneous environments [335, 387, 431, 762, 840]; we do not restate the individual environments here, and collect them together with the domain-specific benchmarks reused throughout this section in Table 7.

We organize general-purpose agents into three representative categories according to their deployment scenarios. *Personal assistants* deploy general-purpose agent capabilities in personal computing and office environments, where agents must operate across applications and interact continuously with users. *Embodied agents and world models* extend such capabilities to physical environments, where agents must perceive, reason, and predict the consequences of actions in the world. *Productive agents* apply general-purpose agent capabilities to professional workflows, where outputs must satisfy domain-specific standards and verification requirements.

6.5.1 Personal Assistants

Personal assistants are an increasingly important class of agents designed to operate across everyday digital environments. Unlike traditional conversational assistants, these agents are expected not only to answer user questions, but also to complete continuous tasks across applications and local resources. Personal assistants therefore mark a shift from conversational assistance to persistent computer agents: the agent no longer merely generates text, but can continuously carry out tasks through multi-turn interaction with both the user and the environment.

A capability underlying all personal assistants is *general tool use*: selecting and invoking the right tools to act beyond the model's parametric knowledge. Early benchmarks assess this ability *without environment state*, testing whether an agent can choose and call the correct function from large, scalable tool sets, as

in ToolBench [492] and the Berkeley Function-Calling Leaderboard [472]. τ-bench [762] then moves tool use into *stateful* environments, where each call mutates a persistent backend and success is judged against realistic multi-turn user interactions rather than isolated call correctness. More recent benchmarks push toward mixed, cross-application settings with increasingly long-tailed tool inventories: Toolathlon [322] spans diverse realistic applications, while MCP-Atlas [31] and MCP-Mark [700] stress-test tool-use competency over large, heterogeneous MCP tool ecosystems. Finally, Claw-Eval [764] and ClawMark [426] evaluate tool use mediated by reusable *skills*, enabling more flexible invocation that better reflects modern agent deployments.

Personal assistants can be broadly divided into four categories. The first targets *office scenarios*, focusing on whether agents can complete tasks in realistic workflows involving documents, emails, calendars, and related office applications. Benchmarks such as OfficeQA, APEX-Agents, PresentBench, and SpreadsheetBench examine capabilities in information retrieval, file and presentation generation, spreadsheet manipulation, and multi-step task execution in such office environments [74, 415, 458, 623]. The second targets *personal computing scenarios*, emphasizing agents' ability to operate real device interfaces, including browsers, desktop applications, mobile apps, and local file systems; this setting largely reuses the computer-use systems and benchmarks of Section 6.3 (e.g., UI-TARS and OSWorld [494, 716]) and is not detailed again here. The third targets *software development scenarios*, where agents assist developers with coding tasks that require code understanding, modification, and validation. Systems such as Claude Code and Codex are representative examples of this direction [10, 453]. The fourth targets *persistent personal assistant scenarios*, emphasizing agents that run continuously in user-controlled devices or environments and connect local resources, long-term memory, and external tools. Systems such as OpenClaw and Hermes illustrate this direction [420, 457, 513, 893].

6.5.2 Embodied Agents and World Models

Embodied agents extend agent capabilities from digital environments to the physical world, where they must continuously perceive the environment, reason about goals and constraints, and execute actions that alter the world state. Long-horizon embodied tasks therefore require a sustained loop of perception, planning, and action, together with reliable state tracking, safety constraints, and recovery from execution errors. Their capabilities can be broadly divided into *navigation* and *manipulation*. Navigation focuses on understanding spatial layouts, planning paths, and adapting to environmental changes [135, 755, 771], whereas manipulation requires fine-grained reasoning about spatial references, object states, physical properties, and interactions such as grasping, moving, and assembling [93, 684, 872, 878]. Recent vision-language-action models, including Gemini Robotics and Qwen's robotic VLA models, integrate language instructions, visual observations, and robotic control within a unified action interface, marking a shift from task-specific controllers toward more general-purpose embodied agents [35, 245, 594, 646, 784, 817].

World models provide embodied agents with predictive representations of environment dynamics, allowing them to evaluate possible action outcomes before acting. This capability is particularly important in physical environments, where exploration is costly and execution failures may be difficult to reverse. Recent world-model research can be broadly divided into two main directions. The first is *predictive representation models*, exemplified by V-JEPA 2 and DINO-WM, which compress the environment into abstract latent representations and predict how these representations evolve over time and in response to actions, rather than directly generating complete future images [25, 879]. By omitting visual details such as texture and illumination, these models can support more efficient multi-step prediction and planning. The second direction is *generative world models*, including Cosmos, Cosmos-Predict2.5, and OSCAR, which directly generate predicted outcomes following an agent's actions to support trajectory comparison, policy training, and risk assessment before real-world execution [2, 449, 699]. Building on these two directions, *world-action models*, such as DreamZero and τ_0-WM, have recently emerged to unify world prediction and action generation by jointly modeling future states and executable actions [769, 884].

6.5.3 Productive Agents

Productive agents apply general-purpose capabilities to workflows whose outputs must satisfy domain-specific standards. They retrieve authoritative sources, invoke specialized tools, and maintain traceable intermediate and final artifacts over extended trajectories. The central challenge is therefore not simply task automation, but the integration of planning, tool use, memory, and verification into established practice. We group these

applications into two intuitive families: (1) *professional services*, where agents assist established work in finance, law, and medicine; and (2) *scientific discovery and reasoning*, where agents help generate and verify new knowledge.

Professional Services. In finance, agents integrate market data, financial statements, and related evidence to produce traceable analyses, as illustrated by FinRobot and FinSight [34, 263, 886]. Legal agents retrieve authoritative sources, connect facts to rules, and produce outputs that can be reviewed against legal doctrine and procedure; the Legal Agent Benchmarks, together with agents such as Parthenon Law and LawThinker, reflect the shift from static question answering toward long-horizon legal work [174, 185, 315, 756]. Medical agents must further combine patient records, images, and examination reports while respecting safety, privacy, and auditability requirements [21, 904]. MedAgentBench, MedAgentBoard, and PhysicianBench accordingly evaluate sustained clinical reasoning and workflow execution rather than isolated diagnosis [23, 257, 380, 895]. Across these domains, correctness cannot usually be reduced to a single executable oracle, so provenance, procedural compliance, and expert review remain central [474, 752].

Scientific Discovery and Reasoning. Scientific-discovery agents go beyond producing domain-compliant analyses: they generate new claims and test them against computational or empirical feedback. Their workflows connect literature review with expert-level and experimental scientific reasoning [642, 882], hypothesis generation, experimental design, execution, result analysis, and iterative refinement [505, 678, 869]. The AI Scientist and its successor organize machine-learning research into repeated cycles of ideation, experimentation, evaluation, and writing [397, 740]. AlphaEvolve and Kosmos extend this pattern to algorithmic and cross-domain discovery [438, 448], while NovelSeek closes the loop from hypothesis generation to verification across heterogeneous scientific tasks [800]. In empirical settings, Coscientist couples language-model planning with robotic chemical experimentation [37], and Robin and the Virtual Lab connect multi-agent reasoning to biomedical discovery and laboratory validation [177, 581]. Hypotheses, programs, experimental records, and manuscripts consequently become evolving research state that must be revised whenever new evidence arrives.

Formal mathematics represents a complementary form of discovery in which proof assistants can mechanically check each completed proof [615, 866]. Seed-Prover 1.5 combines natural-language reasoning, lemma-level sketches, recursive decomposition, and an agentic Lean prover to construct long proofs under iterative verification [59]. Prover Agent similarly coordinates informal reasoning, auxiliary-lemma generation, formal proof construction, and Lean diagnostics in a feedback-driven refinement loop [26]. Automated Proof Engineering broadens the objective beyond isolated theorem proving to repository-scale work, including feature addition, refactoring, and bug repair across evolving Lean libraries, with evaluation based on both compilation and semantic requirement satisfaction [717]. Together, empirical science and formal mathematics show how productive agents can move beyond summarizing existing knowledge to generating claims that survive domain-specific verification [629].

6.6 Benchmarks and Resources

Progress on long-horizon agents is measured and reproduced through a shared substrate of benchmarks and open resources. Because a reported score reflects the joint contribution of the model, the harness, and the training recipe, long-horizon evaluation is best read as *horizon-aware*, sensitive to how long a goal must be held, and *attribution-aware*, able to separate harness-induced from model-induced variance [123, 286, 468]. The same verifiable, long-horizon environments that serve as benchmarks are also frequently reused as training environments for the optimization methods of Section 5, so progress on evaluation and progress on internalization are tightly coupled. Beyond the domain benchmarks tabulated below, a growing family of suites deliberately stresses multi-hour autonomy. Examples range from machine-learning engineering in MLE-bench [51] to end-to-end replication of research papers in PaperBench [560], complementing the frontier time-horizon measurements of METR [286, 429]; these appear among the productive-agent entries of Table 7.

Table 7 consolidates representative resources for the application domains above, grouped by domain and, within each domain, ordered by the capability taxonomy of Section 6. Each row pairs an evaluation benchmark or a reusable system with the capability it targets and, where available, a link to its public repository or official release page (), offering a concrete entry point into the works discussed throughout this section. The multimodal-understanding entries (e.g., video and streaming comprehension benchmarks) are not long-horizon

agent tasks in the strict sense; we include them because they measure the foundational perception capabilities on which long-horizon multimodal agents build.

Table 7 Representative systems and benchmarks relevant to long-horizon agents, grouped by domain. One work per row, annotated with its capability, type, a short description, an (approximate) release date, and a code repository. *Type* distinguishes evaluation *Benchmark*s from reusable *System*s; within each domain, rows are grouped by capability (per Section 6) and ordered by release date. The list is representative rather than exhaustive.

Work	Capability	Type	Description	Date	GitHub Repo
Software Engineering					
AutoCodeRover [843]	Repository Grounding	System	Structure-aware fault localization and patching	2024.04	GitHub
SWE-agent [747]	Repository Grounding	System	Agent–computer interface for repositories	2024.05	GitHub
OpenHands [657]	Repository Grounding	System	Persistent coding workspace	2024.07	GitHub
NL2Repo-Bench [124]	Repository Grounding	Benchmark	From-scratch repository generation	2025.12	GitHub
OctoBench [123]	Repository Grounding	Benchmark	Scaffold-aware instruction following	2026.01	GitHub
SWE-bench [260]	Workflow-level Planning	Benchmark	Repository issue resolution	2023.10	GitHub
Claude Code [10]	Workflow-level Planning	System	Autonomous code editing	2025.02	GitHub
Trae Agent [169]	Workflow-level Planning	System	Ensemble agent for repository issue resolution	2025.07	GitHub
SWE-bench Pro [117]	Workflow-level Planning	Benchmark	Enterprise repository issue resolution	2025.09	GitHub
Terminal-Bench 2.0 [428]	Workflow-level Planning	Benchmark	Hard terminal-based tasks	2025.11	GitHub
Aider[8]	Feedback-driven Repair	System	Iterative pair-programming loop	2023.06	GitHub
DebugBench [602]	Feedback-driven Repair	Benchmark	Multi-language bug diagnosis	2024.01	GitHub
Agentless [703]	Feedback-driven Repair	System	Localize-then-repair pipeline	2024.07	GitHub
SWE-Gym [467]	Feedback-driven Repair	System	Executable training environments	2024.10	GitHub
SWE-smith [748]	Feedback-driven Repair	System	Repair-trajectory synthesis	2025.04	GitHub
Information Seeking					
Search-o1 [336]	Deep Search	System	Retrieval and document reading	2025.01	GitHub
BrowseComp [676]	Deep Search	Benchmark	Persistent web browsing	2025.04	GitHub
WebDancer [689]	Deep Search	System	Information-seeking policy	2025.05	GitHub
Tongyi DeepResearch [307]	Deep Search	System	Agentic deep-research model	2025.10	GitHub
MiroThinker [599]	Deep Search	System	Interactive-scaling research agent	2025.11	GitHub
Open Deep Research [292]	Wide Search	System	Evidence aggregation and synthesis	2025.06	GitHub
WideSearch [685]	Wide Search	Benchmark	Structured information collection	2025.08	GitHub
FlowSearch [227]	Wide Search	System	Knowledge-flow orchestration	2025.10	GitHub
GISA [896]	Wide Search	Benchmark	Deep reasoning and broad information aggregation	2026.02	GitHub
MMInA [603]	Multimodal Grounding	Benchmark	Multi-hop multimodal browsing	2024.04	GitHub
MM-BrowseComp [331]	Multimodal Grounding	Benchmark	Multimodal evidence seeking	2025.08	GitHub
WebWatcher [175]	Multimodal Grounding	System	Vision-language deep research	2025.08	GitHub
WebThinker [337]	Research Synthesis	System	Autonomous research workflow	2025.04	GitHub
DeepResearch Bench [136]	Research Synthesis	Benchmark	Research-report quality evaluation	2025.06	GitHub
WebWeaver [356]	Research Synthesis	System	Evidence organization and synthesis	2025.09	GitHub
DeepConsult [592]	Research Synthesis	Benchmark	Consulting and business deep research	~2025	GitHub
Computer Use					
Mind2Web [116]	Browser Agents	Benchmark	Cross-domain web demonstrations	2023.06	GitHub
WebArena [885]	Browser Agents	Benchmark	Realistic interactive websites	2023.07	GitHub
WebVoyager [208]	Browser Agents	Benchmark	End-to-end live-web agent evaluation	2024.01	GitHub
browser-use[9]	Browser Agents	System	Browser control through agent loop	2024.10	GitHub
BrowserAgent [781]	Browser Agents	System	Human-inspired native browsing	2025.10	GitHub
OSWorld [716]	Desktop GUI Agents	Benchmark	Desktop task environment	2024.04	GitHub
UI-TARS [494]	Desktop GUI Agents	System	Native visual GUI perception	2025.01	GitHub
ScreenSpot-Pro [324]	Desktop GUI Agents	Benchmark	High-resolution professional GUI grounding	2025.01	GitHub
PointArena [83]	Desktop GUI Agents	Benchmark	Language-guided visual pointing	2025.05	GitHub
Seed 2.0 [527]	Desktop GUI Agents	System	Frontier multimodal foundation model	2026.01	GitHub
AndroidWorld [508]	Mobile Agents	Benchmark	Dynamic Android environment	2024.05	GitHub
AndroidLab [734]	Mobile Agents	Benchmark	Android agent training framework	2024.10	GitHub
Mobile-Agent-v3 [766]	Mobile Agents	System	Fundamental GUI-automation agents	2025.08	GitHub

(continued on next page)

Table 7 (continued)

Work	Capability	Type	Description	Date	GitHub Repo
Mobile-Agent-v3.5 [723]	Mobile Agents	System	Multi-platform fundamental GUI agents	2026.01	GitHub
Multimodal Agents					
EgoSchema [419]	Multimodal Understanding	Benchmark	Egocentric video question answering	2023.08	GitHub
MVBench [328]	Multimodal Understanding	Benchmark	Comprehensive video understanding	2023.11	GitHub
VideoAgent [655]	Multimodal Understanding	System	Agentic evidence selection over long video	2024.03	GitHub
Video-MME [158]	Multimodal Understanding	Benchmark	Long-video understanding	2024.05	GitHub
LongVideoBench [688]	Multimodal Understanding	Benchmark	Long-video and text understanding	2024.07	GitHub
StreamingBench [366]	Multimodal Understanding	Benchmark	Streaming-video understanding	2024.11	GitHub
DVD [829]	Multimodal Understanding	System	Tool-using video deep research	2025.05	GitHub
VideoSeek [365]	Multimodal Understanding	System	Tool-augmented active perception in long video	2026.03	GitHub
GenAI-Bench [306]	Multimodal Generation	Benchmark	Text-to-visual generation evaluation	2024.06	GitHub
MM-StoryAgent [733]	Multimodal Generation	System	Text–image–audio story generation	2025.03	GitHub
MUSE [574]	Multimodal Generation	System	Plan–execute–verify story-video generation	2026.02	GitHub
Ptah [803]	Multimodal Generation	System	Verifier-guided multimodal report generation	2026.05	GitHub
CrayOtter [741]	Multimodal Generation	System	Verification-driven multimodal generation	2026.06	GitHub
OmniVideoBench [308]	Omnimodal Agency	Benchmark	Audio-visual evaluation	2025.10	GitHub
Agent-Omni [364]	Omnimodal Agency	System	Omnimodal reasoning agent	2025.11	GitHub
OmniGAIA [339]	Omnimodal Agency	Benchmark	Cross-modal tool-use reasoning	2026.02	GitHub
AgentVista [564]	Omnimodal Agency	Benchmark	Multimodal agent evaluation	2026.02	GitHub
Orchestra-o1 [806]	Omnimodal Agency	System	Cross-modal orchestration agent	2026.06	GitHub
General-Purpose Agents					
ToolBench [492]	Personal Assistants	Benchmark	Large-scale tool use without env state	2023.07	GitHub
Open Interpreter[10]	Personal Assistants	System	Local computer and code execution	2023.07	GitHub
AgentBench [387]	Personal Assistants	Benchmark	Multi-environment agent evaluation	2023.08	GitHub
GAIA [431]	Personal Assistants	Benchmark	General assistant tasks	2023.11	GitHub
τ-bench [762]	Personal Assistants	Benchmark	Stateful tool–agent–user interaction	2024.06	GitHub
AppWorld [614]	Personal Assistants	Benchmark	Stateful app-based personal tasks	2024.07	GitHub
BFCL [472]	Personal Assistants	Benchmark	Function-calling leaderboard	2024.08	GitHub
OpenManus[11]	Personal Assistants	System	General-purpose agent workflows	2025.03	GitHub
MCP-Mark [700]	Personal Assistants	Benchmark	Realistic MCP tool-use stress test	2025.09	GitHub
Toolathlon [322]	Personal Assistants	Benchmark	Diverse realistic tool environments	2025.10	GitHub
MCP-Atlas [31]	Personal Assistants	Benchmark	Large-scale tool-use competency	2026.02	GitHub
Claw-Eval [764]	Personal Assistants	Benchmark	Trustworthy autonomous-agent evaluation	2026.04	GitHub
ClawMark [426]	Personal Assistants	Benchmark	Multi-day multimodal living-world eval	2026.04	GitHub
Agents' Last Exam [575]	Personal Assistants	Benchmark	Frontier professional-workflow evaluation	2026.06	GitHub
VLN-CE [283]	Embodied & World Models	Benchmark	Continuous vision-language navigation	2020.04	GitHub
OpenVLA [277]	Embodied & World Models	System	Open vision-language-action model	2024.06	GitHub
DINO-WM [879]	Embodied & World Models	System	Feature-space world model	2024.11	GitHub
EmbodiedBench [753]	Embodied & World Models	Benchmark	Embodied task evaluation suite	2025.02	GitHub
GR00T N1 [35]	Embodied & World Models	System	Humanoid vision-language-action model	2025.03	GitHub
$\pi_{0.5}$ [245]	Embodied & World Models	System	Open-world VLA model	2025.04	GitHub
V-JEPA 2 [25]	Embodied & World Models	System	Latent video-prediction world model	2025.06	GitHub
Cosmos-Predict2.5 [2]	Embodied & World Models	System	Video world simulation	2025.10	GitHub
ChemCrow [39]	Productive Agents	System	Tool-augmented chemistry reasoning	2023.04	GitHub
FinRobot [886]	Productive Agents	System	Financial research and analysis	2024.05	GitHub
MLE-bench [51]	Productive Agents	Benchmark	Machine-learning engineering tasks	2024.10	GitHub
LegalAgentBench [315]	Productive Agents	Benchmark	Multi-step legal tasks	2024.12	GitHub
MedAgentBench [257]	Productive Agents	Benchmark	Clinical EHR workflows	2025.01	GitHub
METR [286, 429]	Productive Agents	Benchmark	Frontier time-horizon measurement	2025.03	GitHub
AI Scientist-v2 [740]	Productive Agents	System	Autonomous scientific discovery	2025.04	GitHub
PaperBench [560]	Productive Agents	Benchmark	End-to-end research-paper replication	2025.04	GitHub
AlphaEvolve [448]	Productive Agents	System	Algorithmic discovery agent	2025.05	GitHub
Finance Agent Benchmark [34]	Productive Agents	Benchmark	Real-world financial analysis tasks	2025.08	GitHub
GDPval [474]	Productive Agents	Benchmark	Real-world economic task value	2025.10	GitHub

(continued on next page)

Table 7 (continued)

Work	Capability	Type	Description	Date	GitHub Repo
Kosmos [438]	Productive Agents	System	Cross-domain discovery agent	2025.11	GitHub
APEX-Agents [623]	Productive Agents	Benchmark	Office multi-step task execution	2026.01	GitHub
OfficeQA Pro [458]	Productive Agents	Benchmark	Enterprise grounded reasoning	2026.03	GitHub
PresentBench [74]	Productive Agents	Benchmark	Rubric-based slide generation	2026.03	GitHub
OneMillion Bench [752]	Productive Agents	Benchmark	Human-level economic tasks	2026.03	GitHub
Workspace-Bench [588]	Productive Agents	Benchmark	Workspace file-dependency tasks	2026.05	GitHub
YC-Bench [211]	Productive Agents	Benchmark	Long-term planning consistency	2026.06	GitHub

Taken together, these resources show that long-horizon evaluation is consolidating around stateful, multi-hour, and economically grounded tasks, and that a reported score must be read as a joint property of the model, the harness, and the training recipe rather than of the model alone.

7 Frontier: Open Challenges and Outlooks

The previous sections have organized long-horizon agency as it currently stands, across both pillars and the experience loop that links them. We now present a set of *positions*: claims about the most consequential paradigm-level shifts that we expect to define the next several years of research, together with the open problems each shift entails. In our view, the frontiers that matter most lie in this *in-between* space: capability migrates bidirectionally between the harness and the weights. The harness itself turns from fixed scaffolding into a learnable, portable object. And in noisy, costly, real environments, long horizons invalidate the idealized premises that both pillars rely on. Our central diagnosis is that practical long-horizon progress is increasingly a property of the runtime around the model, not of the model alone. To make these pressures easier to compare, we group the frontiers into four higher-level axes: *evolution*, *effectiveness*, *efficiency*, and *trustworthiness*. Table 8 previews the nine positions under this structure. For each concrete frontier we first state where the field stands and what blocks it (*status and challenges*), then our position on where it should go and the open problems it leaves behind (*outlook*).

7.1 Evolution: Generalization and Learning Over Time

The first frontier axis is evolution: whether an agent's long-horizon competence can improve over time rather than remain locked to a hand-written scaffold and a frozen policy. This axis raises three linked questions. Can the harness itself self-evolve? Can the resulting competence transfer across harnesses? And can experience compound across episodes so that the agent keeps learning, and even learns for a lifetime, without destroying previously acquired capability?

7.1.1 Self-evolving Harness and Agents

Status and Challenges. The dominant paradigm of *hand-maintained* agents is a transitional artifact, and self-improvement offers a direct route to extending the effective horizon. We distinguish two levels as follow:

The first is the *self-evolving harness*: with little or no model training, the agent keeps rewriting the runtime structure around a fixed policy. This includes skill-driven fast adaptation, where new skills are distilled directly from failure and success trajectories [206, 705, 706]; recursive procedural-memory evolution, where memory and skills co-evolve from execution traces [813, 853]; and topology evolution, where sub-agents are spawned on demand and the multi-agent graph itself becomes a learnable object [101, 652, 670, 791]. Increasingly the harness is treated as a first-class artifact that can be automatically generated and optimized, as in Meta-Harness and AutoHarness [297, 395], and even hosted by agent operating systems such as AIOS that provide a unified, resource-managed runtime across agents and tasks [424]; industrial trace-based iteration treats run traces as the primary evolution signal [293, 765].

[8] Aider: https://github.com/Aider-AI/aider.
[9] browser-use: https://github.com/browser-use/browser-use.
[10] Open Interpreter: https://github.com/OpenInterpreter/open-interpreter.
[11] OpenManus: https://github.com/FoundationAgents/OpenManus.

Table 8 Four frontier axes and nine frontiers for long-horizon agents. For each frontier we summarize its *open challenges* and our *outlook*. A recurring thread: the harness is where much of the next advance must happen.

Frontier	Open challenges	Outlook
I. Evolution — generalize across runtimes and keep learning over time		
Self-evolving Harness & Agents	• Hand-set optimization metric • Limited gains on out-of-distribution tasks • Overfitting and drift over long runs	• Adaptively choose optimization objective and scope • Autonomous evolution guided by feedback • Lifelong, non-overfitting evolution
Harness Generalization and Transferability	• Models tied to specific harnesses • Performance varies across model–harness pairings • Portable standards still ad hoc	• Multi-harness (hybrid-harness) training • Portable skills / experience at inference time • A standard harness protocol
Continual & Lifelong Learning	• External experience is noisy and unverified • Catastrophic forgetting during weight updates	• Learning only from validated experience • Safe online learning from billion-user logs
II. Effectiveness — act reliably in realistic, verifiable environments		
Real-world Environment Interaction	• Slow, costly, and irreversible interactions with real-world systems • Synthesized Envs & world-models may be unfaithful	• Measure, audit, and optimize sim-to-real faithfulness • Fusion of symbolic and neural simulation environments
From Digital to Embodied Agents	• Deep planning vs. real-time control • Limited modeling of physical dynamics & laws • Coarse vs. fine-grained feedback	• Async, hierarchical harness (deliberation vs. reflex) • Predict physical outcomes with world models • Integrate feedback across timescales
III. Efficiency — spend compute, context, and modality budgets		
Cost- & Budget-aware Agency	• Model selection is poorly matched to task difficulty • No calibrated cost / infeasibility sense • No runtime enforcement of ceilings • No law linking budget to success	• Perceive cost, difficulty, infeasibility • Act mid-run: shorten, re-route, or stop • Harness-enforced ceilings & cost-aware routing • A budget-vs-success scaling law
Multimodal & Omni Harness	• Multimodality bolted on; no vision-native harness • Visual-token budgeting is heuristic • Cross-modal verification is unreliable	• A vision-native, unified omni harness substrate • Per-modality granularity control • Cross-modal verification & omni-modal routing
IV. Trustworthiness — stay robust and governable as the horizon grows		
Reflection & Error Robustness	• Failures detected late under noisy, delayed feedback • Intrinsic self-correction is unreliable • Errors compound into goal drift	• Early failure detection & path-switching • External-anchored, calibrated self-knowledge • Align to human course-correction
Safety & Governance	• Injected error & hazardous experience reuse • No unified safety-verification standard • Self-evolution erodes invariants; biased self-checks	• Untrusted external input; gate experience reuse • Score action safety beside effectiveness • Independent, invariant-preserving verification

The second, more ambitious level is the *self-evolving agent*, whose goal is to fold harness structure and skills back into the weights: the Darwin Gödel Machine rewrites its own code to improve a coding agent [814], and a small but growing line couples skill discovery with reinforcement learning or routes rich runtime trajectories into training, as in SkillRL and OpenClaw-style RL such as Claw-R1 [457, 627, 705].

The shared limitations are threefold, mirroring the outlook below: **(1)** the optimization objective is still a hand-set benchmark metric rather than something the agent chooses for itself; **(2)** gains remain largely within-distribution while the evolution history stays opaque about *what* changed and *why*; and **(3)** over long runs such systems risk overfitting a single benchmark or drifting.

Outlook. We expect self-evolution to mature into a complex *automated-optimization* problem pursued with minimal human supervision, and we highlight three requirements. **(1)** The agent should *autonomously discover its own optimization objective*. Today this is typically a single task's benchmark metric or a training-loss level; in the future it will more likely be a joint objective over many tasks. The agent should locate where its own competence is weak on each task and decide accordingly where to optimize. Sometimes it is enough to

adjust one part of the harness (orchestration logic, a memory module, the tool set), while in other cases the harness and the model weights should be optimized together. **(2)** Self-evolution should proceed *autonomously*: the agent should acquire external information, distill reusable skills (for instance, high-value strategies for deep web search), and continuously update its optimization strategy in response to environmental feedback. **(3)** This process should be able to run *continually, even for a lifetime*: driven by a stream of sparse external signals, it should neither overfit to a single benchmark nor drift into random change, which makes stable long-run self-evolution a genuinely long-term goal. Crucially, if self-evolution is to be more than per-benchmark overfitting, the structure it produces must survive being moved to a different runtime. This is the transferability problem we turn to next.

7.1.2 Harness Generalization and Transferability

Status and Challenges. Harness paradigms keep multiplying, spanning early patterns such as ReAct and CodeAct, the Model Context Protocol, and today's workstation-style harnesses like OpenClaw, Hermes, and Claude Code. As they do, the field is drifting toward a regime in which each strong model is implicitly co-trained with, and bound to, a single harness. Yet what users expect from a capable model is the opposite: that they can drive it from their own agent, and even extend it to a harness they design themselves. We regard this binding as the central scalability risk of the field: a model competitive only inside the harness it was tuned for must be retrained for every new one. It cannot keep pace with a runtime ecosystem that turns over every few months.

Three concrete pressures make the risk tangible. **(1)** Harness-specific tuning effects are large: identical harnesses change rank when ported across providers, sometimes by tens of points, so a model's measured "ability" is partly an artifact of how well its idioms match the harness [468]. **(2)** Multi-model orchestration is already standard: large models plan while small models execute. Learned preference-based routing [451] and layered model composition [638] serve as building blocks, with systems such as EvoRoute, Chimera, and ORCH offering early systematic studies [447, 810, 880]; any single-harness assumption thus breaks immediately. **(3)** Portable cross-provider standards such as AGENTS.md and Agent Skills are emerging organically rather than from any single vendor [16, 98]. This development elevates *transferability*, the extent to which a harness configuration remains effective when the base model is swapped, to a first-class concern.

Outlook. We see three complementary routes toward harness-robust competence. **(1)** *External transfer*: package experience, and ideally skills, into portable artifacts so that a model can read externally mounted skills at inference time and gradually raise its competence without retraining. **(2)** *Internal generalization*, in our view the more fundamental route: it exposes the model to many harnesses during training by mixing multi-harness trajectories and even running hybrid-harness RL, so that harness-robust behavior is built into the weights rather than bolted on. **(3)** *Standardizing a harness protocol*: just as the Model Context Protocol connected the world's tools behind one discoverable, authorizable interface and thereby made tool use broadly transferable [100], a standard harness protocol that supports flexible design and registration could become the starting point for world-wide harness generalization.

The concrete open problems are: the minimal portable contract that preserves performance across providers; training recipes that explicitly optimize for cross-harness transfer rather than single-harness fit; robustness under harness drift, when a new server or tool-calling convention silently changes behavior; and portable observability, so that traces from different runtimes can be analyzed in one stack. Transferability concerns whether competence transfers across runtimes at a single point in time; the complementary question is whether competence *accumulates* across time, which is the problem of continual learning.

7.1.3 Continual and Lifelong Learning

Status and Challenges. Continual and lifelong learning are central to long-horizon agents because such agents operate across many tasks, sessions, environments, and users rather than within a single isolated episode. A capable agent should not only solve the current task, but also accumulate experience that improves later behavior while preserving previously acquired abilities. Existing work pursues this along two complementary routes. **(1)** *External* lifelong learning stores experience outside the parameters: Voyager maintains an expanding library of executable skills for open-ended exploration [630], ExpeL extracts natural-language lessons from past

trajectories [853], and Reflexion keeps verbal feedback in episodic memory to improve later trials without weight updates [544]; such methods are cheap, reversible, and quick to adapt. **(2)** *Internal* continual learning folds new knowledge or behavior into the model itself; this supports broader generalization, but frequent updates remain costly and risk catastrophic forgetting, especially under noisy, shifting deployment data [58, 641].

Outlook. The key challenge is to connect these two forms of learning into a stable lifelong-learning loop: most new experience should first be kept externally as memory, trajectories, or skills, where it can be inspected, revised, and tested across future tasks, and only experience that proves reliable and broadly useful should later be consolidated into weights. This staged process makes lifelong learning less about fine-tuning after every interaction and more about deciding what to remember, reuse, generalize, or forget.

A distinctive new opportunity is *online learning from deployment at scale*: as billion-user agent products such as Cursor[12], Claude Code[13], and Doubao[14] mature, they accumulate vast streams of user logs and feedback. Turning these traces into safe improvements is a difficult but consequential long-term research challenge. Doing so requires metrics that identify high-value episodes and online-learning algorithms that incorporate them into the agent with minimal human intervention and without regressions. The open problems include how to prevent forgetting during periodic updates, how to extract trustworthy learning signals from noisy trajectories and user feedback, how to transfer lessons across tasks and environments, and how to evaluate improvement over many sessions rather than within a single episode. In this sense, lifelong learning is a necessary condition for long-horizon agency: without it, an agent may act for a long time, yet never truly become better over time.

7.2 Effectiveness: Acting in Realistic Environments

Evolution matters only if the resulting agent remains effective when the environment becomes slow, noisy, partially irreversible, and hard to verify. The second axis concerns whether long-horizon agents can operate effectively in realistic environments. We first consider the digital environments used for training and evaluation, and then turn to the physical world, where challenges arise from real-time perception and control.

7.2.1 Real-world Environment Interaction

Status and Challenges. A great many agent applications are defined by sustained interaction with real-world environments: ordering food or booking a flight, drafting a Feishu or Notion document, running real-time financial analysis and forecasting, or communicating on platforms such as WeChat[15] and X[16]. For training long-horizon agents, however, we usually cannot act directly in these live environments because interaction is slow, costly, and often irreversible. The binding constraint becomes the *environment* rather than the model: where an agent can act, whether it can fail safely, and whether it can receive a checkable reward.

Two construction routes are emerging. **(1)** *Verifiable, executable synthesis* builds environments that behave like the real systems they stand in for and mechanically decide success and failure: realistic testbeds such as WebArena, OSWorld, and TheAgentCompany ground agents in real browsers, operating systems, and multi-day company workflows [716, 722, 885], and automated environment synthesis pushes this toward scale [131, 238, 555]. **(2)** *Model-simulated environments* treat a large model itself as an implicit world model, as in GenEnv, Qwen-AgentWorld, and reasoning-model simulators such as Simia-SFT [195, 905], trading mechanical guarantees for flexibility and coverage.

Both routes face the same test of *faithfulness*: an environment that is inexpensive to evaluate but unfaithful may train agents that perform well in controlled settings yet fail in deployment. Faithfulness is thus best treated not as one property but as several separately auditable axes: whether interfaces and available actions match the real system; whether identical actions produce the same state change; whether automatic validators agree with real acceptance criteria; whether simulated users behave like real ones; and whether a reset returns the system to a functionally equivalent state rather than one that merely looks similar.

[12] https://cursor.com
[13] https://www.anthropic.com/claude-code
[14] https://www.doubao.com
[15] https://www.wechat.com
[16] https://x.com

Outlook. We hold that environment construction, not model scale, is the rate-limiting step for the next phase of agent capability, and that "faithfulness" deserves to be measured as explicitly as task success is today. The most promising direction is to *fuse* the two routes above by combining the verifiability of synthesized environments with the flexibility of model-simulated ones. This fusion would allow agents to train broadly while still receiving trustworthy rewards. On the learning side the binding constraint is sample efficiency: real-world rollouts are slow, costly, and sometimes irreversible, so the algorithms that matter extract the most signal per interaction through off-policy reuse, experience replay over returned traces, and curricula in which environment difficulty co-evolves with the policy.

The open problems are: automated construction of environments that are provably checkable and measurably faithful; metrics and audits for the sim-to-real faithfulness of synthetic and model-simulated environments; sample-efficient RL for slow, costly, partially irreversible interaction; and curricula that co-evolve environment difficulty with agent competence. The most demanding realistic environment of all, however, is the physical world rather than the digital one. In this setting, perception and control must run in real time, and many actions cannot be undone. These constraints define the frontier of embodiment.

7.2.2 From Digital to Embodied Agents

Status and Challenges. A natural extension of long-horizon digital agency is embodiment: equipping an agent with real-world perception and control so that it can provide high-level deliberation for an embodied system. The externalization logic of this paper extends directly to embodiment. The digital harness becomes a high-level controller that orchestrates low-level physical skills, recapitulating the harness pattern as a division of labor between the cerebrum and cerebellum. In this arrangement, a high-level LLM/VLM agent interprets goals, decomposes tasks, and replans based on feedback [243, 357], while Vision-Language-Action (VLA) models serve as callable real-time skill modules for atomic manipulation [35, 36, 134, 277, 899]. But embodiment sharpens three challenges specific to the physical world. **(1)** A *timescale conflict*: long-horizon planning is deep but slow, and its deliberation clashes with the millisecond reactive control an embodied system demands. **(2)** A *dimensionality and physics gap*: today's long-horizon agents reason mostly over language and, increasingly, other digital modalities, whereas the physical world has far higher effective dimensionality, so better modeling of physical laws and dynamics is a key hurdle to migrating a digital long-horizon agent into a body. **(3)** A *feedback-granularity gap*: real environments provide feedback at multiple spatial and temporal scales, from sparse task-level signals to dense control-level signals (pose deviations, contact forces, or tactile mismatches), and an embodied agent must integrate feedback across these scales rather than rely only on the coarse plan-level signals that text harnesses handle well.

Outlook. We expect embodied and digital agent architectures to converge on a shared engineering stack with the long-horizon harness as the deliberative core, and we read the three challenges above as its research agenda. **(1)** For the timescale conflict, the promising route is an asynchronous, hierarchical harness that decouples slow deliberation from a fast reactive control loop, letting the deliberative core replan out-of-band while cached skills and reflexes keep the body responsive. **(2)** For the physics gap, embodied harnesses will likely lean on action-conditioned world models and simulation, so that the agent can predict physical consequences and close part of the sim-to-real gap before acting. **(3)** For the feedback-granularity gap, memory, skill, and protocol designs must carry both coarse subgoals and fine-grained continuous corrections through one interface, with verifiers that operate at both scales.

The open problems are: real-time, safety-critical action verification under latency budgets text harnesses can ignore; memory, skill, and protocol designs for irreversible physical action; closing the sim-to-real gap between simulated and physical deployment; and a unified deliberative harness that drives both digital tools and physical effectors. Whether an agent acts through digital tools or physical effectors, each step also draws down a finite budget of computation, context, and time. Therefore, how these resources should be allocated becomes a first-class frontier.

7.3 Efficiency: Spending Compute, Context, and Modality Budgets

Long horizons turn resources into first-class design variables. An agent must decide not only what to do, but how much computation, context, modality bandwidth, and verification effort each part of the run deserves.

The efficiency frontier spans both classical compute budgeting and the newer problem of scheduling high-cost multimodal evidence.

7.3.1 Cost- and Budget-aware Agency

Status and Challenges. As long-horizon agents move from research demos into ordinary users' everyday workflows, such as hours-long coding sessions and personal-assistance workflows spanning an entire day, cost and budget cease to be implementation details and become first-class concerns for both usability and economics. Yet today's agents are largely *budget-blind*: they carry almost no working sense of what a token costs, no notion that context and compute are a finite budget to be spent deliberately, and no felt pressure to finish within a price or time ceiling. In practice they default to the most capable model at every step, re-read context they could have summarized, and keep spending on runs that are already doomed. This is merely wasteful at demo scale, but once such agents run continuously for millions of users it translates into real cost and a degraded day-to-day experience.

The mechanisms to spend more wisely already exist, including compute-optimal test-time allocation [550], adaptive reasoning budgets [442], tunable best-of-N and multi-agent aggregation [323], and cost-aware model routing and cascades [62, 237, 447, 451, 810, 880]. However, agents do not yet use these mechanisms with any awareness of their own spending. Budget-awareness is best seen not as a single capability but as a chain: an agent must perceive what a task will cost, act on that estimate, run inside a harness that can enforce the limits it would otherwise ignore, and ideally be guided by a law that predicts the trade-off.

Outlook. We frame the way forward as four open problems along this chain. **(1)** *Perceiving cost, difficulty, and infeasibility.* An agent should forecast, and keep re-forecasting as a run unfolds, how hard a task is, how much budget remains, and whether the task has already become infeasible; even a coarse predicted-infeasibility signal can save a large fraction of the tokens otherwise burned on doomed runs [370]. The quantity is multi-dimensional and coupled (tokens, money, wall-clock time, sometimes real resources), and the estimate must be *calibrated* rather than optimistic, since only a calibrated forecast tells the agent when and by how much to change course. **(2)** *Acting on the estimate.* A cost estimate is worthless unless it changes behavior: the agent should shorten reasoning on easy subtasks, avoid redundant re-reading, reserve depth for where it changes the outcome, and escalate, stop, or re-route when the forecast turns bad, while avoiding a measurable loss in quality. The difficulty is that these choices are made mid-trajectory under uncertainty, where a false economy early can cost far more later. **(3)** *Enforcing budgets from the harness.* Not all budget discipline should live in the model's head; the harness should enforce frugality through budget-aware routing and cascades, context compression and eviction, caching and reuse of intermediate results, and scheduling primitives that hold the agent to a hard price, token, or time ceiling even when the model would overspend. Because the harness sees the whole run, it is also the natural place to account for cost across a long, multi-tool trajectory. **(4)** *A scaling law for budget and success.* Further out, is there a predictable relationship linking compute budget, tool calls, and run length to success rate, analogous to a Chinchilla-style law for agency? Without one, budget allocation stays a craft; the first credible candidates should emerge from the long-running-task regime, where budgets are large enough that depth-versus-breadth trade-offs dominate.

In text-only agents these budgets are already difficult; in multimodal agents the dominant cost may be not reasoning depth but which sensory stream to keep, compress, or discard.

7.3.2 Multimodal and Omni Harnesses

Status and Challenges. Current harnesses are still designed primarily around textual inputs and command-line interaction, whereas future long-horizon agents will operate over far richer multimodal information, including images, video, audio, GUI states, and generated media. Current multimodal agents typically treat multimodality as an auxiliary capability rather than as a native substrate. Doubao's task-assistant mode, for example, uses a text-centric main agent that invokes external tools for image and video generation and perception. Recent work has begun to address this challenge. Examples include efforts toward on-policy data evolution for visual-native multimodal deep-search agents, as well as verifiable multimodal deep research through multi-agent harnesses for interleaved report generation. However, a unified definition and architecture for a multimodal or omni harness is still missing. The technical obstacles are concrete. **(1)** *The cost of visual*

tokens: video and image streams cannot be placed into context in full in the same way as text logs, so the agent must decide when to observe, what evidence to retain, and how to compress, retrieve, and memorize multimodal information; today this allocation across text, images, and video remains largely heuristic. **(2)** *Cross-modal verification*: multimodal evidence often must be transcribed, aligned, or abstracted into another representation before it can be judged, and these intermediate transformations can themselves fail or hallucinate, so adding more verification stages need not improve reliability linearly. **(3)** *A wider externalization surface*: multimodal externalization changes the design assumptions underlying skill specification, memory indexing, and protocol schemas, opening a substantially wider frontier for what can be externalized at all.

Outlook. The most consequential shift is that a unified, *vision-native* omni harness must emerge, and we frame the agenda as four requirements. **(1)** *A vision-native substrate.* It should play for real-world visual capability a role analogous to that of Claude Code and OpenClaw for code and the workstation, giving the whole research program a vision-native harness to build on rather than a text harness with modalities attached. **(2)** *Per-modality, budget-aware granularity.* Future omni harnesses must choose an appropriate information granularity per modality (raw frames preserve the most information at the highest cost, while sampling, clip summarization, or feature compression trade detail for savings) and decide autonomously at what granularity multimodal evidence is retained, compressed, and retrieved. **(3)** *Early error detection before rendering.* Because the outputs of image and video generators are no longer cheap textual artifacts, harnesses must detect errors and reduce rework *before* full artifacts are produced, through intermediate representations, low-fidelity previews, staged inspection, or constrained generation. **(4)** *Omni-modal routing.* The harness should decide not only which model and how deep to search, but which modality, temporal segment, and expert model deserve the budget [339, 364]. As agents extend perception and action beyond text, efficiency and trustworthiness become inseparable: every decision to retain, compress, discard, or regenerate multimodal evidence is at once a cost trade-off and a reliability risk.

7.4 Trustworthiness: Robust and Governed Autonomy

The final axis asks how autonomy remains trustworthy as the horizon grows. Long runs make failures cumulative rather than isolated: small early mistakes compound into goal drift, and the more authority a harness is granted, the larger the blast radius of a single wrong action. Accordingly, trustworthiness joins two problems that are usually studied apart: whether an agent can notice and recover from its own errors under noisy feedback, and whether its autonomy can be governed, contained, and audited as it acts on real systems.

7.4.1 Reflection and Error Robustness

Status and Challenges. Error robustness is the first pillar of trustworthy long-horizon autonomy: an agent that runs unattended over many steps can be trusted only insofar as it can tell when it is going wrong and change course, since an undetected early error is silently amplified over the rest of the run. Real tasks rarely succeed on the first attempt, and the feedback from a real environment is delayed, partial, and noisy, so robustness is a property of the *whole trajectory*, not of any single step: the decisive capability is to act sensibly under noisy feedback, detect mid-execution that a path is failing, and stop or switch reasoning paths early rather than committing to one track to the bitter end. The self-correction literature now provides a more qualified account of when these methods are effective. Reflexion and Self-Refine show that iterative critique-and-revise loops can raise reliability [416, 544], and CRITIC shows that tool-grounded critique, which verifies against an external signal rather than the model's own confidence, is markedly more trustworthy than ungrounded self-reflection [182]. The crucial caveat is that models frequently *cannot* reliably self-correct reasoning from intrinsic signals alone and may even degrade when they try [234], which is why external, harness-supplied verification must anchor reflection, and why over long horizons undetected errors compound into goal drift and inherited error from weaker prior trajectories [20, 427].

Outlook. We see three priorities. **(1)** *Early detection and path-switching*: under noisy, delayed environmental feedback, the harness should detect a failing trajectory as early and cheaply as possible and abandon it to try an alternative, rather than spending the budget before discovering the run was doomed. **(2)** *Aligning to human correction*: in practice, a human often steps in partway through a run to redirect the agent, and a central open problem is how to internalize that signal; possible routes include external, human-aligned validation

checkpoints and incorporating human course-correction trajectories into training so that the agent learns *when* and *how* people intervene. **(3)** *Calibrated self-knowledge*: reflection should be grounded in external verification rather than the model's own often-miscalibrated confidence, and the agent should carry a calibrated sense of what it does not know, so that it escalates or asks for help instead of confidently compounding an error. But detecting and recovering from one's own errors is only half of trustworthy autonomy; the other half is ensuring that an agent, however robust, stays governed, contained, and auditable as the authority it wields over real systems grows. This is the problem of safety and governance.

7.4.2 Safety and Governance

Status and Challenges. In many respects, safety is the highest-stakes frontier in long-horizon agency, and three pressures make this challenge particularly acute.

(1) *The environment itself can inject error and risk*: a real environment may return manipulated or adversarial observations. More subtly, an agent that summarizes its own experience can distill a hazardous heuristic and then reuse it; for example, having once observed that a successful refund elicits a grateful reply, an agent may over-generalize and become predisposed to issue refunds in later tasks. Such experience-reuse effects compound as harness artifacts (skills, servers, shared memory) are shared across teams, creating attack surfaces analogous to a software supply chain [463, 902]. **(2)** *No unified standard for safety verification*: harness-level verifiers and benchmarks still overwhelmingly treat task success as the primary measure of *effectiveness*, while largely ignoring whether the actions taken along the way were *safe*, so local verifiers can all pass even when the run violates real-world constraints. **(3)** *Safety under self-evolution*: self-modifying harnesses make it hard to preserve initial safety invariants, and because a self-evolving agent that audits its own modifications is prone to the same biases that produced them, self-checking alone is not a trustworthy guarantee.

Underlying all three is a structural fact: most deployed safety mechanisms (least privilege, approval gates, capability scoping, audit, sandbox isolation) are *harness-level* controls living above the model and outside its weights, so treating safety as a model-only property miscounts the true attack surface.

Outlook. We sketch a technical path for each pressure. **(1)** Against injected error and hazardous experience reuse, the harness should treat external observations as untrusted input by sandboxing them and tracking their provenance, and should gate which distilled experiences may be reused, so that a heuristic must clear a safety review before it can shape future behavior. **(2)** Against the verification gap, the community needs a *unified safety-verification standard* that scores action safety as a first-class axis alongside effectiveness, with external acceptance tests and red-team suites that certify what an agent did, not just whether it succeeded. **(3)** Against self-evolution risk, invariant-preserving evolution should be enforced by *independent* verifiers rather than the evolving agent's own judgment, ideally with provable guarantees that a modification cannot weaken a fixed set of safety invariants.

More broadly, each safety layer adds latency, token cost, and friction, so the recurring engineering challenge is to quantify the *alignment tax* of a harness design and minimize it without compromising safety [1, 41, 303, 368, 520, 547, 617, 783]. Our position is that the harness, not the model, is the principal alignment and privilege surface for production agents: safety properties that hold robustly in production are almost always harness-imposed, while model-side alignment is necessary but rarely sufficient. This argument only sharpens once the harness leaves the screen: as embodied agents turn digital decisions into physical action, governance stops being a runtime policy and becomes a real-time safety constraint, which makes the harness, not the model, the decisive locus of trust for long-horizon autonomy.

8 Conclusion

In this paper, we review long-horizon AI agents through a central thesis: long-horizon agency emerges from the *co-evolution of externalized harness engineering and internalized model optimization*. From this view, we develop the discussion through six aspects. **Foundation** formalizes the agent as a coupling of a base policy and a surrounding harness, organizes long-horizon tasks into three nested levels (H1 to H3) together with the capabilities they demand (C1 to C3), and characterizes the scaling of the execution time horizon.

Evolution traces how the control surface has progressively widened across three stages: prompt engineering, context engineering, and runtime harnesses. **Harness** takes a full lifecycle view, organizing the runtime by component: the agent loop and workflow, context and memory, tools and skills, orchestration, hooks, and verification. **Optimization** lays out a complete training recipe for internalizing competence, proceeding from an architectural and data–environment substrate through pre-/mid-training, fine-tuning, reinforcement learning, and on-policy distillation to self-evolution. **Application** organizes practice by the interface between the agent and its environment, spanning software engineering, information seeking, computer use, multimodal, and general-purpose agents, paired with horizon-aware evaluation. textbfFrontier consolidates the open problems into four axes: evolution, effectiveness, efficiency, and trustworthiness.

Beyond organizing prior work, we distill several emerging directions that define the next stage of long-horizon agent research. We hope this paper lays a solid foundation for future work and serves as a useful reference for researchers and practitioners alike. As agentic systems continue to mature, the long-horizon agent will remain a central and open problem, one likely to play a decisive role in building robust, general, and enduring artificial intelligence.

References

[1] Sahar Abdelnabi, Kai Greshake, Shailesh Mishra, Christoph Endres, Thorsten Holz, and Mario Fritz. Not what you've signed up for: Compromising real-world llm-integrated applications with indirect prompt injection. In Maura Pintor, Xinyun Chen, and Florian Tramèr, editors, *Proceedings of the 16th ACM Workshop on Artificial Intelligence and Security, AISec 2023, Copenhagen, Denmark, 30 November 2023*, pages 79–90. ACM, 2023. doi: 10.1145/3605764.3623985. URL https://doi.org/10.1145/3605764.3623985.

[2] Niket Agarwal, Arslan Ali, Maciej Bala, Yogesh Balaji, Erik Barker, Tiffany Cai, Prithvijit Chattopadhyay, Yongxin Chen, Yin Cui, Yifan Ding, Daniel Dworakowski, Jiaojiao Fan, Michele Fenzi, Francesco Ferroni, Sanja Fidler, Dieter Fox, Songwei Ge, Yunhao Ge, Jinwei Gu, Siddharth Gururani, Ethan He, Jiahui Huang, Jacob Samuel Huffman, Pooya Jannaty, Jingyi Jin, Seung Wook Kim, Gergely Klár, Grace Lam, Shiyi Lan, Laura Leal-Taixé, Anqi Li, Zhaoshuo Li, Chen-Hsuan Lin, Tsung-Yi Lin, Huan Ling, Ming-Yu Liu, Xian Liu, Alice Luo, Qianli Ma, Hanzi Mao, Kaichun Mo, Arsalan Mousavian, Seungjun Nah, Sriharsha Niverty, David Page, Despoina Paschalidou, Zeeshan Patel, Lindsey Pavao, Morteza Ramezanali, Fitsum Reda, Xiaowei Ren, Vasanth Rao Naik Sabavat, Ed Schmerling, Stella Shi, Bartosz Stefaniak, Shitao Tang, Lyne Tchapmi, Przemek Tredak, Wei-Cheng Tseng, Jibin Varghese, Hao Wang, Haoxiang Wang, Heng Wang, Ting-Chun Wang, Fangyin Wei, Xinyue Wei, Jay Zhangjie Wu, Jiashu Xu, Wei Yang, Yen-Chen Lin, Xiaohui Zeng, Yu Zeng, Jing Zhang, Qinsheng Zhang, Yuxuan Zhang, Qingqing Zhao, and Artur Zólkowski. Cosmos world foundation model platform for physical AI. *CoRR*, abs/2501.03575, 2025. doi: 10.48550/ARXIV.2501.03575. URL https://doi.org/10.48550/arXiv.2501.03575.

[3] Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos Garea, Matthieu Geist, and Olivier Bachem. On-policy distillation of language models: Learning from self-generated mistakes. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=3zKtaqxLhW.

[4] Saaket Agashe, Kyle Wong, Vincent Tu, Jiachen Yang, Ang Li, and Xin Eric Wang. Agent S2: A compositional generalist-specialist framework for computer use agents. *CoRR*, abs/2504.00906, 2025. doi: 10.48550/ARXIV.2504.00906. URL https://doi.org/10.48550/arXiv.2504.00906.

[5] Joshua Ainslie, James Lee-Thorp, Michiel de Jong, Yury Zemlyanskiy, Federico Lebrón, and Sumit Sanghai. GQA: training generalized multi-query transformer models from multi-head checkpoints. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 4895–4901. Association for Computational Linguistics, 2023. doi: 10.18653/V1/2023.EMNLP-MAIN.298. URL https://doi.org/10.18653/v1/2023.emnlp-main.298.

[6] Eloi Alonso, Adam Jelley, Vincent Micheli, Anssi Kanervisto, Amos J. Storkey, Tim Pearce, and François Fleuret. Diffusion for world modeling: Visual details matter in atari. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang, editors, *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. URL http://papers.nips.cc/paper_files/paper/2024/hash/6bdde0373d53d4a501249547084bed43-Abstract-Conference.html.

[7] Chenxin An, Shansan Gong, Ming Zhong, Xingjian Zhao, Mukai Li, Jun Zhang, Lingpeng Kong, and Xipeng Qiu. L-eval: Instituting standardized evaluation for long context language models. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pages 14388–14411. Association for Computational Linguistics, 2024. doi: 10.18653/V1/2024.ACL-LONG.776. URL https://doi.org/10.18653/v1/2024.acl-long.776.

[8] Maksym Andriushchenko, Alexandra Souly, Mateusz Dziemian, Derek Duenas, Maxwell Lin, Justin Wang, Dan Hendrycks, Andy Zou, J. Zico Kolter, Matt Fredrikson, Yarin Gal, and Xander Davies. Agentharm: A benchmark for measuring harmfulness of LLM agents. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=AC5n7xHuR1.

[9] Anthropic. Introducing advanced tool use on the claude developer platform, 2025. URL https://www.anthropic.com/engineering/advanced-tool-use.

[10] Anthropic. Overview - claude code docs, 2025. URL https://code.claude.com/docs/en/overview.

[11] Anthropic. Hooks reference - claude code docs, 2025. URL https://code.claude.com/docs/en/hooks.

[12] Anthropic. Computer use tool — claude api docs, 2025. URL https://platform.claude.com/docs/en/agents-and-tools/tool-use/computer-use-tool.

[13] Anthropic. Effective context engineering for ai agents, 2025. URL https://www.anthropic.com/engineering/effective-context-engineering-for-ai-agents.

[14] Anthropic. Effective harnesses for long-running agents, 2025. URL https://www.anthropic.com/engineering/effective-harnesses-for-long-running-agents.

[15] Anthropic. Code execution with mcp: building more efficient ai agents, 2025. URL https://www.anthropic.com/engineering/code-execution-with-mcp.

[16] Anthropic. Equipping agents for the real world with agent skills, 2025. URL https://www.anthropic.com/engineering/equipping-agents-for-the-real-world-with-agent-skills.

[17] Anthropic. Harness design for long-running application development, 2026. URL https://www.anthropic.com/engineering/harness-design-long-running-apps.

[18] Anthropic. Scaling managed agents: Decoupling the brain from the hands, 2026. URL https://www.anthropic.com/engineering/managed-agents.

[19] David Anugraha, Zilu Tang, Lester James V. Miranda, Hanyang Zhao, Mohammad Rifqi Farhansyah, Garry Kuwanto, Derry Wijaya, and Genta Indra Winata. R3: robust rubric-agnostic reward models. *CoRR*, abs/2505.13388, 2025. doi: 10.48550/ARXIV.2505.13388. URL https://doi.org/10.48550/arXiv.2505.13388.

[20] Rauno Arike, Elizabeth Donoway, Henning Bartsch, and Marius Hobbhahn. Technical report: Evaluating goal drift in language model agents. *CoRR*, abs/2505.02709, 2025. doi: 10.48550/ARXIV.2505.02709. URL https://doi.org/10.48550/arXiv.2505.02709.

[21] Rahul K. Arora, Jason Wei, Rebecca Soskin Hicks, Preston Bowman, Joaquin Quiñonero Candela, Foivos Tsimpourlas, Michael Sharman, Meghan Shah, Andrea Vallone, Alex Beutel, Johannes Heidecke, and Karan Singhal. Healthbench: Evaluating large language models towards improved human health. *CoRR*, abs/2505.08775, 2025. doi: 10.48550/ARXIV.2505.08775. URL https://doi.org/10.48550/arXiv.2505.08775.

[22] Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. Self-rag: Learning to retrieve, generate, and critique through self-reflection. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=hSyW5go0v8.

[23] Tajamul Ashraf, Hyewon Jeong, Fida Mohammad Thoker, and Bernard Ghanem. Medcta: A benchmark for clinical tool agents. *CoRR*, abs/2606.11702, 2026. doi: 10.48550/ARXIV.2606.11702. URL https://doi.org/10.48550/arXiv.2606.11702.

[24] Mahmoud Assran, Quentin Duval, Ishan Misra, Piotr Bojanowski, Pascal Vincent, Michael G. Rabbat, Yann LeCun, and Nicolas Ballas. Self-supervised learning from images with a joint-embedding predictive architecture. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023*, pages 15619–15629. IEEE, 2023. doi: 10.1109/CVPR52729.2023.01499. URL https://doi.org/10.1109/CVPR52729.2023.01499.

[25] Mido Assran, Adrien Bardes, David Fan, Quentin Garrido, Russell Howes, Mojtaba Komeili, Matthew J. Muckley, Ammar Rizvi, Claire Roberts, Koustuv Sinha, Artem Zholus, Sergio Arnaud, Abha Gejji, Ada Martin, Francois Robert Hogan, Daniel Dugas, Piotr Bojanowski, Vasil Khalidov, Patrick Labatut, Francisco Massa, Marc Szafraniec, Kapil Krishnakumar, Yong Li, Xiaodong Ma, Sarath Chandar, Franziska Meier, Yann LeCun, Michael Rabbat, and Nicolas Ballas. V-JEPA 2: Self-supervised video models enable understanding, prediction and planning. *CoRR*, abs/2506.09985, 2025. doi: 10.48550/ARXIV.2506.09985. URL https://doi.org/10.48550/arXiv.2506.09985.

[26] Kaito Baba, Chaoran Liu, Shuhei Kurita, and Akiyoshi Sannai. Prover agent: An agent-based framework for formal mathematical proofs. *CoRR*, abs/2506.19923, 2025. doi: 10.48550/ARXIV.2506.19923. URL https://doi.org/10.48550/arXiv.2506.19923.

[27] S. Bai, L. Bing, L. Lei, R. Li, X. Li, X. Lin, E. Min, L. Su, B. Wang, L. Wang, L. Wang, S. Wang, X. Wang, Y. Zhang, Z. Zhang, G. Chen, L. Chen, Z. Cheng, Y. Deng, Z. Huang, D. Ng, J. Ni, Q. Ren, X. Tang, B. L. Wang, H. Wang, N. Wang, C. Wei, Q. Wu, J. Xia, Y. Xiao, H. Xu, X. Xu, C. Xue, Z. Yang, Z. Yang,

F. Ye, H. Ye, J. Yu, C. Zhang, W. Zhang, H. Zhao, and P. Zhu. Mirothinker-1.7 & H1: towards heavy-duty research agents via verification. *CoRR*, abs/2603.15726, 2026. doi: 10.48550/ARXIV.2603.15726. URL https://doi.org/10.48550/arXiv.2603.15726.

[28] Yushi Bai, Xin Lv, Jiajie Zhang, Hongchang Lyu, Jiankai Tang, Zhidian Huang, Zhengxiao Du, Xiao Liu, Aohan Zeng, Lei Hou, Yuxiao Dong, Jie Tang, and Juanzi Li. Longbench: A bilingual, multitask benchmark for long context understanding. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pages 3119–3137. Association for Computational Linguistics, 2024. doi: 10.18653/V1/2024.ACL-LONG.172. URL https://doi.org/10.18653/v1/2024.acl-long.172.

[29] Yushi Bai, Jiajie Zhang, Xin Lv, Linzhi Zheng, Siqi Zhu, Lei Hou, Yuxiao Dong, Jie Tang, and Juanzi Li. Longwriter: Unleashing 10, 000+ word generation from long context llms. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=kQ5s9Yh0WI.

[30] Federico Baldassarre, Marc Szafraniec, Basile Terver, Vasil Khalidov, Francisco Massa, Yann LeCun, Patrick Labatut, Maximilian Seitzer, and Piotr Bojanowski. Back to the features: DINO as a foundation for video world models. *CoRR*, abs/2507.19468, 2025. doi: 10.48550/ARXIV.2507.19468. URL https://doi.org/10.48550/arXiv.2507.19468.

[31] Chaithanya Bandi, Ben Hertzberg, Geobio Boo, Tejas Polakam, Jeff Da, Sami Hassaan, Manasi Sharma, Andrew Park, Ernesto Hernandez, Dan Rambado, Ivan Salazar, Rafael Cruz, Chetan Rane, Ben Levin, Brad Kenstler, and Bing Liu. Mcp-atlas: A large-scale benchmark for tool-use competency with real MCP servers. *CoRR*, abs/2602.00933, 2026. doi: 10.48550/ARXIV.2602.00933. URL https://doi.org/10.48550/arXiv.2602.00933.

[32] Iz Beltagy, Matthew E. Peters, and Arman Cohan. Longformer: The long-document transformer. *CoRR*, abs/2004.05150, 2020. URL https://arxiv.org/abs/2004.05150.

[33] Maciej Besta, Nils Blach, Ales Kubicek, Robert Gerstenberger, Michal Podstawski, Lukas Gianinazzi, Joanna Gajda, Tomasz Lehmann, Hubert Niewiadomski, Piotr Nyczyk, and Torsten Hoefler. Graph of thoughts: Solving elaborate problems with large language models. In Michael J. Wooldridge, Jennifer G. Dy, and Sriraam Natarajan, editors, *Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2024, Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence, IAAI 2024, Fourteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2024, February 20-27, 2024, Vancouver, Canada*, pages 17682–17690. AAAI Press, 2024. doi: 10.1609/AAAI.V38I16.29720. URL https://doi.org/10.1609/aaai.v38i16.29720.

[34] Antoine Bigeard, Langston Nashold, Rayan Krishnan, and Shirley Wu. Finance agent benchmark: Benchmarking llms on real-world financial research tasks. *CoRR*, abs/2508.00828, 2025. doi: 10.48550/ARXIV.2508.00828. URL https://doi.org/10.48550/arXiv.2508.00828.

[35] Johan Bjorck, Fernando Castañeda, Nikita Cherniadev, Xingye Da, Runyu Ding, Linxi Fan, Yu Fang, Dieter Fox, Fengyuan Hu, Spencer Huang, Joel Jang, Zhenyu Jiang, Jan Kautz, Kaushil Kundalia, Lawrence Lao, Zhiqi Li, Zongyu Lin, Kevin Lin, Guilin Liu, Edith LLontop, Loic Magne, Ajay Mandlekar, Avnish Narayan, Soroush Nasiriany, Scott Reed, You Liang Tan, Guanzhi Wang, Zu Wang, Jing Wang, Qi Wang, Jiannan Xiang, Yuqi Xie, Yinzhen Xu, Zhenjia Xu, Seonghyeon Ye, Zhiding Yu, Ao Zhang, Hao Zhang, Yizhou Zhao, Ruijie Zheng, and Yuke Zhu. GR00T N1: an open foundation model for generalist humanoid robots. *CoRR*, abs/2503.14734, 2025. doi: 10.48550/ARXIV.2503.14734. URL https://doi.org/10.48550/arXiv.2503.14734.

[36] Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Lucy Xiaoyang Shi, James Tanner, Quan Vuong, Anna Walling, Haohuan Wang, and Ury Zhilinsky. π_0: A vision-language-action flow model for general robot control. *CoRR*, abs/2410.24164, 2024. doi: 10.48550/ARXIV.2410.24164. URL https://doi.org/10.48550/arXiv.2410.24164.

[37] Daniil A. Boiko, Robert MacKnight, Ben Kline, and Gabe Gomes. Autonomous chemical research with large language models. *Nat.*, 624(7992):570–578, 2023. doi: 10.1038/S41586-023-06792-0. URL https://doi.org/10.1038/s41586-023-06792-0.

[38] Sebastian Borgeaud, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, George van den Driessche, Jean-Baptiste Lespiau, Bogdan Damoc, Aidan Clark, Diego de Las Casas, Aurelia Guy, Jacob Menick, Roman Ring, Tom Hennigan, Saffron Huang, Loren Maggiore, Chris Jones, Albin Cassirer, Andy Brock, Michela Paganini, Geoffrey Irving, Oriol Vinyals, Simon Osindero, Karen Simonyan, Jack W. Rae, Erich Elsen, and Laurent Sifre. Improving language models by retrieving from trillions of tokens. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvári, Gang Niu, and Sivan Sabato, editors, *International Conference on Machine*

Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA, volume 162 of *Proceedings of Machine Learning Research*, pages 2206–2240. PMLR, 2022. URL https://proceedings.mlr.press/v162/borgeaud22a.html.

[39] Andres M. Bran, Sam Cox, Oliver Schilter, Carlo Baldassari, Andrew D. White, and Philippe Schwaller. Augmenting large language models with chemistry tools. *Nat. Mac. Intell.*, 6(5):525–535, 2024. doi: 10.1038/S42256-024-00832-8. URL https://doi.org/10.1038/s42256-024-00832-8.

[40] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/1457c0d6bfcb4967418bfb8ac142f64a-Abstract.html.

[41] Christoph Bühler, Matteo Biagiola, Luca Di Grazia, and Guido Salvaneschi. Securing AI agent execution. *CoRR*, abs/2510.21236, 2025. doi: 10.48550/ARXIV.2510.21236. URL https://doi.org/10.48550/arXiv.2510.21236.

[42] ByteDance Seed Team. Seed1.6 tech introduction, 2025. URL https://seed.bytedance.com/en/seed1_6.

[43] Tianle Cai, Yuhong Li, Zhengyang Geng, Hongwu Peng, Jason D. Lee, Deming Chen, and Tri Dao. Medusa: Simple LLM inference acceleration framework with multiple decoding heads. In Ruslan Salakhutdinov, Zico Kolter, Katherine A. Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp, editors, *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*, volume 235 of *Proceedings of Machine Learning Research*, pages 5209–5235. PMLR / OpenReview.net, 2024. URL https://proceedings.mlr.press/v235/cai24b.html.

[44] Yishuo Cai, Xingyu Guo, Xuancheng Huang, Jinhua Du, Can Huang, Wenxuan Huang, Wenhan Ma, Yuyang Hu, Aohan Zeng, Jie Tang, and Xu Sun. From player to master: Enhancing test-time learning of LLM agents via reinforcement learning over memory. *CoRR*, abs/2606.08656, 2026. doi: 10.48550/ARXIV.2606.08656. URL https://doi.org/10.48550/arXiv.2606.08656.

[45] Meng Cao, Pengfei Hu, Yingyao Wang, Jihao Gu, Haoran Tang, Haoze Zhao, Chen Wang, Jiahua Dong, Wangbo Yu, Ge Zhang, Xiang Li, Ian Reid, and Xiaodan Liang. Video simpleqa: Towards factuality evaluation in large video language models. In Sven Koenig, Chad Jenkins, and Matthew E. Taylor, editors, *Fortieth AAAI Conference on Artificial Intelligence, Thirty-Eighth Conference on Innovative Applications of Artificial Intelligence, Sixteenth Symposium on Educational Advances in Artificial Intelligence, AAAI 2026, Singapore, January 20-27, 2026*, pages 2616–2624. AAAI Press, 2026. doi: 10.1609/AAAI.V40I4.37249. URL https://doi.org/10.1609/aaai.v40i4.37249.

[46] Ruisheng Cao, Mouxiang Chen, Jiawei Chen, Zeyu Cui, Yunlong Feng, Binyuan Hui, Yuheng Jing, Kaixin Li, Mingze Li, Junyang Lin, Zeyao Ma, Kashun Shum, Xuwu Wang, Jinxi Wei, Jiaxi Yang, Jiajun Zhang, Lei Zhang, Zongmeng Zhang, Wenting Zhao, and Fan Zhou. Qwen3-coder-next technical report. *CoRR*, abs/2603.00729, 2026. doi: 10.48550/ARXIV.2603.00729. URL https://doi.org/10.48550/arXiv.2603.00729.

[47] Yilin Cao, Yufeng Zhong, Zhixiong Zeng, Liming Zheng, Jing Huang, Haibo Qiu, Peng Shi, Wenji Mao, and Wan Guanglu. Mobiledreamer: Generative sketch world model for GUI agent. *CoRR*, abs/2601.04035, 2026. doi: 10.48550/ARXIV.2601.04035. URL https://doi.org/10.48550/arXiv.2601.04035.

[48] Mert Cemri, Melissa Z. Pan, Shuyi Yang, Lakshya A. Agrawal, Bhavya Chopra, Rishabh Tiwari, Kurt Keutzer, Aditya G. Parameswaran, Dan Klein, Kannan Ramchandran, Matei A. Zaharia, Joseph E. Gonzalez, and Ion Stoica. Why do multi-agent LLM systems fail? In Danielle Belgrave, Cheng Zhang, Laura N. Montoya, Hsuan-Tien Lin, Razvan Pascanu, Piotr Koniusz, Marzyeh Ghassemi, Nancy Chen, Iván Vladimir Meza Ruíz, and Arturo Loaiza-Bonilla, editors, *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2025, NeurIPS 2025, San Diego, CA, USA, December 2-7, 2025 / Mexico City, Mexico, November 30 - December 5, 2025*, 2025. URL http://papers.nips.cc/paper_files/paper/2025/hash/b1041e52d3be19f0a9bc491657488e4a-Abstract-Datasets_and_Benchmarks_Track.html.

[49] Hyungjoo Chae, Sunghwan Kim, Junhee Cho, Seungone Kim, Seungjun Moon, Gyeom Hwangbo, Dongha Lim, Minjin Kim, Yeonjun Hwang, Minju Gwak, Dongwook Choi, Minseok Kang, Gwanhoon Im, ByeongUng Cho, Hyojun Kim, Jun Hee Han, Taeyoon Kwon, Minju Kim, Beong-woo Kwak, Dongjin Kang, and Jinyoung Yeo. Web-shepherd: Advancing prms for reinforcing web agents. *CoRR*, abs/2505.15277, 2025. doi: 10.48550/ARXIV.2505.15277. URL https://doi.org/10.48550/arXiv.2505.15277.

[50] Hyungjoo Chae, Jungsoo Park, and Alan Ritter. Safe and scalable web agent learning via recreated websites. *CoRR*, abs/2603.10505, 2026. doi: 10.48550/ARXIV.2603.10505. URL https://doi.org/10.48550/arXiv.2603.10505.

[51] Jun Shern Chan, Neil Chowdhury, Oliver Jaffe, James Aung, Dane Sherburn, Evan Mays, Giulio Starace, Kevin Liu, Leon Maksin, Tejal Patwardhan, Aleksander Madry, and Lilian Weng. Mle-bench: Evaluating machine learning agents on machine learning engineering. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=6s5uXNWGIh.

[52] Qikai Chang, Zhenrong Zhang, Pengfei Hu, Jiefeng Ma, Yicheng Pan, Jianshu Zhang, Jun Du, Quan Liu, and Jianqing Gao. THOR: tool-integrated hierarchical optimization via RL for mathematical reasoning. *CoRR*, abs/2509.13761, 2025. doi: 10.48550/ARXIV.2509.13761. URL https://doi.org/10.48550/arXiv.2509.13761.

[53] Aili Chen, Chi Zhang, Junteng Liu, Jiangjie Chen, Chengyu Du, Yunji Li, Ming Zhong, Qin Wang, Zhengmao Zhu, Jiayuan Song, Ke Ji, Junxian He, Pengyu Zhao, and Yanghua Xiao. DIVE: scaling diversity in agentic task synthesis for generalizable tool use. *CoRR*, abs/2603.11076, 2026. doi: 10.48550/ARXIV.2603.11076. URL https://doi.org/10.48550/arXiv.2603.11076.

[54] Baian Chen, Chang Shu, Ehsan Shareghi, Nigel Collier, Karthik Narasimhan, and Shunyu Yao. FireAct: Toward language agent fine-tuning. *CoRR*, abs/2310.05915, 2023. doi: 10.48550/ARXIV.2310.05915. URL https://doi.org/10.48550/arXiv.2310.05915.

[55] Guo Chen, Yicheng Liu, Yifei Huang, Baoqi Pei, Jilan Xu, Yuping He, Tong Lu, Yali Wang, and Limin Wang. Cg-bench: Clue-grounded question answering benchmark for long video understanding. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=le4IoZZHy1.

[56] Guoxin Chen, Zile Qiao, Wenqing Wang, Donglei Yu, Xuanzhong Chen, Hao Sun, Minpeng Liao, Kai Fan, Yong Jiang, Penguin Xie, Wayne Xin Zhao, Ruihua Song, and Fei Huang. MARS: optimizing dual-system deep research via multi-agent reinforcement learning. *CoRR*, abs/2510.04935, 2025. doi: 10.48550/ARXIV.2510.04935. URL https://doi.org/10.48550/arXiv.2510.04935.

[57] Guoxin Chen, Zile Qiao, Xuanzhong Chen, Donglei Yu, Haotian Xu, Wayne Xin Zhao, Ruihua Song, Wenbiao Yin, Huifeng Yin, Liwen Zhang, Kuan Li, Minpeng Liao, Yong Jiang, Pengjun Xie, Fei Huang, and Jingren Zhou. Iterresearch: Rethinking long-horizon agents with interaction scaling, 2026. URL https://arxiv.org/abs/2511.07327.

[58] Hongyang Chen, Zhongwu Sun, Hongfei Ye, Kunchi Li, and Xuemin Lin. Continual learning in large language models: Methods, challenges, and opportunities. *CoRR*, abs/2603.12658, 2026. doi: 10.48550/ARXIV.2603.12658. URL https://doi.org/10.48550/arXiv.2603.12658.

[59] Jiangjie Chen, Wenxiang Chen, Jiacheng Du, Jinyi Hu, Zhicheng Jiang, Allan Jie, Xiaoran Jin, Xing Jin, Chenggang Li, Wenlei Shi, Zhihong Wang, Mingxuan Wang, Chenrui Wei, Shufa Wei, Huajian Xin, Fan Yang, Weihao Gao, Zheng Yuan, Tianyang Zhan, Zeyu Zheng, Tianxi Zhou, and Thomas Hanwen Zhu. Seed-prover 1.5: Mastering undergraduate-level theorem proving via learning from experience. *CoRR*, abs/2512.17260, 2025. doi: 10.48550/ARXIV.2512.17260. URL https://doi.org/10.48550/arXiv.2512.17260.

[60] Kevin Chen, Marco F. Cusumano-Towner, Brody Huval, Aleksei Petrenko, Jackson Hamburger, Vladlen Koltun, and Philipp Krähenbühl. Reinforcement learning for long-horizon interactive LLM agents. *CoRR*, abs/2502.01600, 2025. doi: 10.48550/ARXIV.2502.01600. URL https://doi.org/10.48550/arXiv.2502.01600.

[61] Liang Chen, Weichu Xie, Yiyan Liang, Hongfeng He, Hans Zhao, Zhibo Yang, Zhiqi Huang, Haoning Wu, Haoyu Lu, Y. Charles, Yiping Bao, Yuantao Fan, Guopeng Li, Haiyang Shen, Xuanzhong Chen, Wendong Xu, Shuzheng Si, Zefan Cai, Wenhao Chai, Ziqi Huang, Fangfu Liu, Tianyu Liu, Baobao Chang, Xiaobo Hu, Kaiyuan Chen, Yixin Ren, Yang Liu, Yuan Gong, and Kuan Li. Babyvision: Visual reasoning beyond language. *CoRR*, abs/2601.06521, 2026. doi: 10.48550/ARXIV.2601.06521. URL https://doi.org/10.48550/arXiv.2601.06521.

[62] Lingjiao Chen, Matei Zaharia, and James Zou. Frugalgpt: How to use large language models while reducing cost and improving performance. *Trans. Mach. Learn. Res.*, 2024, 2024. URL https://openreview.net/forum?id=cSimKw5p6R.

[63] Mayee F. Chen, Nicholas Roberts, Kush Bhatia, Jue Wang, Ce Zhang, Frederic Sala, and Christopher Ré. Skill-it! A data-driven skills framework for understanding and training language models. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine, editors, *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New*

Orleans, LA, USA, December 10 - 16, 2023, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/70b8505ac79e3e131756f793cd80eb8d-Abstract-Conference.html.

[64] Mengzhuo Chen, Junjie Wang, Zhe Liu, Yawen Wang, and Qing Wang. From failed trajectories to reliable LLM agents: Diagnosing and repairing harness flaws. *CoRR*, abs/2606.06324, 2026. doi: 10.48550/ARXIV.2606.06324. URL https://doi.org/10.48550/arXiv.2606.06324.

[65] Shiqi Chen, Jingze Gai, Ruochen Zhou, Jinghan Zhang, Tongyao Zhu, Junlong Li, Kangrui Wang, Zihan Wang, Zhengyu Chen, Klara Kaleb, Ning Miao, Siyang Gao, Cong Lu, Manling Li, Junxian He, and Yee Whye Teh. Skillcraft: Can LLM agents learn to use tools skillfully? *CoRR*, abs/2603.00718, 2026. doi: 10.48550/ARXIV.2603.00718. URL https://doi.org/10.48550/arXiv.2603.00718.

[66] Shouyuan Chen, Sherman Wong, Liangjian Chen, and Yuandong Tian. Extending context window of large language models via positional interpolation. *CoRR*, abs/2306.15595, 2023. doi: 10.48550/ARXIV.2306.15595. URL https://doi.org/10.48550/arXiv.2306.15595.

[67] Shuang Chen, Kaituo Feng, Hangting Chen, Wenxuan Huang, Dasen Dai, Quanxin Shou, Yunlong Lin, Xiangyu Yue, Shenghua Gao, and Tianyu Pang. Opensearch-vl: An open recipe for frontier multimodal search agents. *CoRR*, abs/2605.05185, 2026. doi: 10.48550/ARXIV.2605.05185. URL https://doi.org/10.48550/arXiv.2605.05185.

[68] Shuang Chen, Quanxin Shou, Hangting Chen, Yucheng Zhou, Kaituo Feng, Wenbo Hu, Yi-Fan Zhang, Yunlong Lin, Wenxuan Huang, Mingyang Song, Dasen Dai, Bolin Jiang, Manyuan Zhang, Shi-Xue Zhang, Zhengkai Jiang, Lucas Wang, Zhao Zhong, Yu Cheng, and Nanyun Peng. Unify-agent: A unified multimodal agent for world-grounded image synthesis. *CoRR*, abs/2603.29620, 2026. doi: 10.48550/ARXIV.2603.29620. URL https://doi.org/10.48550/arXiv.2603.29620.

[69] Weize Chen, Yusheng Su, Jingwei Zuo, Cheng Yang, Chenfei Yuan, Chi-Min Chan, Heyang Yu, Yaxi Lu, Yi-Hsin Hung, Chen Qian, Yujia Qin, Xin Cong, Ruobing Xie, Zhiyuan Liu, Maosong Sun, and Jie Zhou. Agentverse: Facilitating multi-agent collaboration and exploring emergent behaviors. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=EHg5GDnyq1.

[70] Wenhu Chen, Xueguang Ma, Xinyi Wang, and William W. Cohen. Program of thoughts prompting: Disentangling computation from reasoning for numerical reasoning tasks. *Trans. Mach. Learn. Res.*, 2023, 2023. URL https://openreview.net/forum?id=YfZ4ZPt8zd.

[71] Wenshuo Chen, Kuimou Yu, Bowen Tian, Jianfei Song, Shaofeng Liang, Haozhe Jia, Kan Cheng, Haosen Li, Kaishen Yuan, Lei Wang, Jiemin Wu, Songning Lai, and Yutao Yue. Memogen: Can past experience improve future text-to-image generation? *CoRR*, abs/2606.03243, 2026. doi: 10.48550/ARXIV.2606.03243. URL https://doi.org/10.48550/arXiv.2606.03243.

[72] Xianwei Chen, Shimin Zhang, and Jibin Wu. f-opd: Stabilizing long-horizon on-policy distillation with freshness-aware control. *CoRR*, abs/2605.17862, 2026. doi: 10.48550/ARXIV.2605.17862. URL https://doi.org/10.48550/arXiv.2605.17862.

[73] Xiaokang Chen, Zhiyu Wu, Xingchao Liu, Zizheng Pan, Wen Liu, Zhenda Xie, Xingkai Yu, and Chong Ruan. Janus-pro: Unified multimodal understanding and generation with data and model scaling. *CoRR*, abs/2501.17811, 2025. doi: 10.48550/ARXIV.2501.17811. URL https://doi.org/10.48550/arXiv.2501.17811.

[74] Xin-Sheng Chen, Jiayu Zhu, Pei-lin Li, Hanzheng Wang, Shuojin Yang, and Menghao Guo. Presentbench: A fine-grained rubric-based benchmark for slide generation. *CoRR*, abs/2603.07244, 2026. doi: 10.48550/ARXIV.2603.07244. URL https://doi.org/10.48550/arXiv.2603.07244.

[75] Xuanzhong Chen, Zile Qiao, Guoxin Chen, Liangcai Su, Zhen Zhang, Xinyu Wang, Pengjun Xie, Fei Huang, Jingren Zhou, and Yong Jiang. Agentfrontier: Expanding the capability frontier of LLM agents with zpd-guided data synthesis. *CoRR*, abs/2510.24695, 2025. doi: 10.48550/ARXIV.2510.24695. URL https://doi.org/10.48550/arXiv.2510.24695.

[76] Yifei Chen, Guanting Dong, and Zhicheng Dou. Toward effective tool-integrated reasoning via self-evolved preference learning. *CoRR*, abs/2509.23285, 2025. doi: 10.48550/ARXIV.2509.23285. URL https://doi.org/10.48550/arXiv.2509.23285.

[77] Yifei Chen, Guanting Dong, and Zhicheng Dou. Et-agent: Incentivizing effective tool-integrated reasoning agent via behavior calibration. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens, editors, *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long*

Papers), ACL 2026, San Diego, California, United States, July 2-7, 2026, pages 7337–7359. Association for Computational Linguistics, 2026. URL https://aclanthology.org/2026.acl-long.333/.

[78] Yukang Chen, Shengju Qian, Haotian Tang, Xin Lai, Zhijian Liu, Song Han, and Jiaya Jia. Longlora: Efficient fine-tuning of long-context large language models. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=6PmJoRfdaK.

[79] Zehui Chen, Kuikun Liu, Qiuchen Wang, Wenwei Zhang, Jiangning Liu, Dahua Lin, Kai Chen, and Feng Zhao. Agent-flan: Designing data and methods of effective agent tuning for large language models. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Findings of the Association for Computational Linguistics, ACL 2024, Bangkok, Thailand and virtual meeting, August 11-16, 2024*, volume ACL 2024 of *Findings of ACL*, pages 9354–9366. Association for Computational Linguistics, 2024. doi: 10.18653/V1/2024.FINDINGS-ACL.557. URL https://doi.org/10.18653/v1/2024.findings-acl.557.

[80] Zhaorun Chen, Mintong Kang, and Bo Li. Shieldagent: Shielding agents via verifiable safety policy reasoning. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu, editors, *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*, volume 267 of *Proceedings of Machine Learning Research*. PMLR / OpenReview.net, 2025. URL https://proceedings.mlr.press/v267/chen25ae.html.

[81] Zhixun Chen, Ming Li, Yuxuan Huang, Yali Du, Meng Fang, and Tianyi Zhou. ATLAS: agent tuning via learning critical steps. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar, editors, *Findings of the Association for Computational Linguistics, ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, volume ACL 2025 of *Findings of ACL*, pages 25334–25349. Association for Computational Linguistics, 2025. doi: 10.18653/V1/2025.FINDINGS-ACL.1299. URL https://doi.org/10.18653/v1/2025.findings-acl.1299.

[82] Daixuan Cheng, Shaohan Huang, Yuxian Gu, Huatong Song, Guoxin Chen, Li Dong, Wayne Xin Zhao, Ji-Rong Wen, and Furu Wei. Computer environments elicit general agentic intelligence in llms, 2026. URL https://arxiv.org/abs/2601.16206.

[83] Long Cheng, Jiafei Duan, Yi Ru Wang, Haoquan Fang, Boyang Li, Yushan Huang, Elvis Wang, Ainaz Eftekhar, Jason Lee, Wentao Yuan, Rose Hendrix, Noah A. Smith, Fei Xia, Dieter Fox, and Ranjay Krishna. Pointarena: Probing multimodal grounding through language-guided pointing. *CoRR*, abs/2505.09990, 2025. doi: 10.48550/ARXIV.2505.09990. URL https://doi.org/10.48550/arXiv.2505.09990.

[84] Pengyu Cheng, Tianhao Hu, Han Xu, Zhisong Zhang, Yong Dai, Lei Han, Nan Du, and Xiaolong Li. Self-playing adversarial language game enhances LLM reasoning. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang, editors, *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. URL http://papers.nips.cc/paper_files/paper/2024/hash/e4be7e9867ef163563f4a5e90cec478f-Abstract-Conference.html.

[85] Xianfu Cheng, Wei Zhang, Shiwei Zhang, Jian Yang, Xiangyuan Guan, Xianjie Wu, Xiang Li, Ge Zhang, Jiaheng Liu, Yuying Mai, Yutao Zeng, Zhoufutu Wen, Ke Jin, Baorui Wang, Weixiao Zhou, Yunhong Lu, Hangyuan Ji, Tongliang Li, Wenhao Huang, and Zhoujun Li. Simplevqa: Multimodal factuality evaluation for multimodal large language models. In *IEEE/CVF International Conference on Computer Vision, ICCV 2025, Honolulu, HI, USA, October 19-25, 2025*, pages 4637–4646. IEEE, 2025. doi: 10.1109/ICCV51701.2025.00441. URL https://doi.org/10.1109/ICCV51701.2025.00441.

[86] Zihao Cheng, Hongru Wang, Zeming Liu, Xinyi Wang, Xiangrong Zhu, Yuhang Guo, Wei Lin, Jeff Z. Pan, and Yunhong Wang. Terminal-world: Scaling terminal-agent environments via agent skills. *CoRR*, abs/2605.20876, 2026. doi: 10.48550/ARXIV.2605.20876. URL https://doi.org/10.48550/arXiv.2605.20876.

[87] Sahana Chennabasappa, Cyrus Nikolaidis, Daniel Song, David Molnar, Stephanie Ding, Shengye Wan, Spencer Whitman, Lauren Deason, Nicholas Doucette, Abraham Montilla, Alekhya Gampa, Beto de Paola, Dominik Gabi, James Crnkovich, Jean-Christophe Testud, Kat He, Rashnil Chaturvedi, Wu Zhou, and Joshua Saxe. Llamafirewall: An open source guardrail system for building secure AI agents. *CoRR*, abs/2505.03574, 2025. doi: 10.48550/ARXIV.2505.03574. URL https://doi.org/10.48550/arXiv.2505.03574.

[88] Prateek Chhikara, Dev Khant, Saket Aryan, Taranjeet Singh, and Deshraj Yadav. Mem0: Building production-ready AI agents with scalable long-term memory. In Inês Lynce, Nello Murano, Mauro Vallati, Serena Villata, Federico Chesani, Michela Milano, Andrea Omicini, and Mehdi Dastani, editors, *ECAI 2025 - 28th European Conference on Artificial Intelligence, 25-30 October 2025, Bologna, Italy - Including 14th Conference on Prestigious*

Applications of Intelligent Systems (PAIS 2025), volume 413 of *Frontiers in Artificial Intelligence and Applications*, pages 2993–3000. IOS Press, 2025. doi: 10.3233/FAIA251160. URL https://doi.org/10.3233/FAIA251160.

[89] Xiaowei Chi, Chun-Kai Fan, Hengyuan Zhang, Xingqun Qi, Rongyu Zhang, Anthony Chen, Chi-Min Chan, Wei Xue, Qifeng Liu, Shanghang Zhang, and Yike Guo. Empowering world models with reflection for embodied video prediction. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu, editors, *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*, volume 267 of *Proceedings of Machine Learning Research*. PMLR / OpenReview.net, 2025. URL https://proceedings.mlr.press/v267/chi25b.html.

[90] Abhi Chivukula, Jay Somasundaram, and Vijay Somasundaram. Agint: Agentic graph compilation for software engineering agents. *CoRR*, abs/2511.19635, 2025. doi: 10.48550/ARXIV.2511.19635. URL https://doi.org/10.48550/arXiv.2511.19635.

[91] Jae-Woo Choi, Hyungmin Kim, Hyobin Ong, Minsu Jang, Dohyung Kim, Jaehong Kim, and Youngwoo Yoon. Reactree: Hierarchical LLM agent trees with control flow for long-horizon task planning. *CoRR*, abs/2511.02424, 2025. doi: 10.48550/ARXIV.2511.02424. URL https://doi.org/10.48550/arXiv.2511.02424.

[92] Krzysztof Marcin Choromanski, Valerii Likhosherstov, David Dohan, Xingyou Song, Andreea Gane, Tamás Sarlós, Peter Hawkins, Jared Quincy Davis, Afroz Mohiuddin, Lukasz Kaiser, David Benjamin Belanger, Lucy J. Colwell, and Adrian Weller. Rethinking attention with performers. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. URL https://openreview.net/forum?id=Ua6zuk0WRH.

[93] Wei Chow, Jiageng Mao, Boyi Li, Daniel Seita, Vitor Campagnolo Guizilini, and Yue Wang. Physbench: Benchmarking and enhancing vision-language models for physical world understanding. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=Q6a9W6kzv5.

[94] Paul F. Christiano, Jan Leike, Tom B. Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 4299–4307, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/d5e2c0adad503c91f91df240d0cd4e49-Abstract.html.

[95] Xiangxiang Chu, Hailang Huang, Xiao Zhang, Fei Wei, and Yong Wang. GPG: A simple and strong reinforcement learning baseline for model reasoning. *CoRR*, abs/2504.02546, 2025. doi: 10.48550/ARXIV.2504.02546. URL https://doi.org/10.48550/arXiv.2504.02546.

[96] Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. *CoRR*, abs/2110.14168, 2021. URL https://arxiv.org/abs/2110.14168.

[97] A2A Contributors. Agent2agent (a2a) protocol specification, 2025. URL https://a2a-protocol.org/latest/specification/.

[98] AGENTS.md Contributors. Agents.md community specification, 2025. URL https://agents.md/.

[99] Model Context Protocol Contributors. Model Context Protocol (MCP), 2025. URL https://modelcontextprotocol.io/docs/getting-started/intro.

[100] Model Context Protocol Contributors. Specification - model context protocol, 2025. URL https://modelcontextprotocol.io/specification/2025-11-25.

[101] Igor Costa. Agentspawn: Adaptive multi-agent collaboration through dynamic spawning for long-horizon code generation. *CoRR*, abs/2602.07072, 2026. doi: 10.48550/ARXIV.2602.07072. URL https://doi.org/10.48550/arXiv.2602.07072.

[102] Chaoqun Cui, Jing Huang, Shijing Wang, Liming Zheng, Qingchao Kong, and Zhixiong Zeng. Agentic reward modeling: Verifying GUI agent via online proactive interaction. *CoRR*, abs/2602.00575, 2026. doi: 10.48550/ARXIV.2602.00575. URL https://doi.org/10.48550/arXiv.2602.00575.

[103] Ganqu Cui, Yuchen Zhang, Jiacheng Chen, Lifan Yuan, Zhi Wang, Yuxin Zuo, Haozhan Li, Yuchen Fan, Huayu Chen, Weize Chen, Zhiyuan Liu, Hao Peng, Lei Bai, Wanli Ouyang, Yu Cheng, Bowen Zhou, and Ning Ding.

The entropy mechanism of reinforcement learning for reasoning language models. *CoRR*, abs/2505.22617, 2025. doi: 10.48550/ARXIV.2505.22617. URL https://doi.org/10.48550/arXiv.2505.22617.

[104] Cursor. Cursor docs — rules, 2025. URL https://cursor.com/docs/rules.md.

[105] Gaole Dai, Shiqi Jiang, Ting Cao, Yuanchun Li, Yuqing Yang, Rui Tan, Mo Li, and Lili Qiu. Advancing mobile GUI agents: A verifier-driven approach to practical deployment. *CoRR*, abs/2503.15937, 2025. doi: 10.48550/ARXIV.2503.15937. URL https://doi.org/10.48550/arXiv.2503.15937.

[106] Yuqin Dai, Guoqing Wang, Yuan Wang, Kairan Dou, Kaichen Zhou, Zhanwei Zhang, Shuo Yang, Fei Tang, Jun Yin, Pengyu Zeng, Zhenzhe Ying, Can Yi, Changhua Meng, Yuchen Zhou, Yongliang Shen, and Shuai Lu. Evinote-rag: Enhancing RAG models via answer-supportive evidence notes. *CoRR*, abs/2509.00877, 2025. doi: 10.48550/ARXIV.2509.00877. URL https://doi.org/10.48550/arXiv.2509.00877.

[107] Yufan Dang, Chen Qian, Xueheng Luo, Jingru Fan, Zihao Xie, Ruijie Shi, Weize Chen, Cheng Yang, Xiaoyin Che, Ye Tian, Xuantang Xiong, Lei Han, Zhiyuan Liu, and Maosong Sun. Multi-agent collaboration via evolving orchestration. In Danielle Belgrave, Cheng Zhang, Laura N. Montoya, Hsuan-Tien Lin, Razvan Pascanu, Piotr Koniusz, Marzyeh Ghassemi, Nancy Chen, Iván Vladimir Meza Ruíz, and Arturo Loaiza-Bonilla, editors, *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2025, NeurIPS 2025, San Diego, CA, USA, December 2-7, 2025 / Mexico City, Mexico, November 30 - December 5, 2025*, 2025. URL http://papers.nips.cc/paper_files/paper/2025/hash/f1320d2e2842169c6fc89dcbd80e94d0-Abstract-Conference.html.

[108] Tri Dao, Daniel Y. Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. Flashattention: Fast and memory-efficient exact attention with io-awareness. In Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022. URL http://papers.nips.cc/paper_files/paper/2022/hash/67d57c32e20fd0a7a302cb81d36e40d5-Abstract-Conference.html.

[109] Edoardo Debenedetti, Ilia Shumailov, Tianqi Fan, Jamie Hayes, Nicholas Carlini, Daniel Fabian, Christoph Kern, Chongyang Shi, Andreas Terzis, and Florian Tramèr. Defeating prompt injections by design. *CoRR*, abs/2503.18813, 2025. doi: 10.48550/ARXIV.2503.18813. URL https://doi.org/10.48550/arXiv.2503.18813.

[110] DeepSeek-AI. Deepseek-v3 technical report. *CoRR*, abs/2412.19437, 2024. doi: 10.48550/ARXIV.2412.19437. URL https://doi.org/10.48550/arXiv.2412.19437.

[111] DeepSeek-AI. Deepseek-v2: A strong, economical, and efficient mixture-of-experts language model. *CoRR*, abs/2405.04434, 2024. doi: 10.48550/ARXIV.2405.04434. URL https://doi.org/10.48550/arXiv.2405.04434.

[112] DeepSeek-AI. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *CoRR*, abs/2501.12948, 2025. doi: 10.48550/ARXIV.2501.12948. URL https://doi.org/10.48550/arXiv.2501.12948.

[113] DeepSeek-AI. Deepseek-v3.2: Pushing the frontier of open large language models. *CoRR*, abs/2512.02556, 2025. doi: 10.48550/ARXIV.2512.02556. URL https://doi.org/10.48550/arXiv.2512.02556.

[114] Chao Deng, Jiale Yuan, Pi Bu, Peijie Wang, Zhong-Zhi Li, Jian Xu, Xiao-Hui Li, Yuan Gao, Jun Song, Bo Zheng, and Cheng-Lin Liu. Longdocurl: a comprehensive multimodal long document benchmark integrating understanding, reasoning, and locating. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar, editors, *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, pages 1135–1159. Association for Computational Linguistics, 2025. doi: 10.18653/V1/2025.ACL-LONG.57. URL https://doi.org/10.18653/v1/2025.acl-long.57.

[115] Chaorui Deng, Deyao Zhu, Kunchang Li, Chenhui Gou, Feng Li, Zeyu Wang, Shu Zhong, Weihao Yu, Xiaonan Nie, Ziang Song, Shi Guang, and Haoqi Fan. Emerging properties in unified multimodal pretraining. *CoRR*, abs/2505.14683, 2025. doi: 10.48550/ARXIV.2505.14683. URL https://doi.org/10.48550/arXiv.2505.14683.

[116] Xiang Deng, Yu Gu, Boyuan Zheng, Shijie Chen, Samual Stevens, Boshi Wang, Huan Sun, and Yu Su. Mind2web: Towards a generalist agent for the web. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine, editors, *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/5950bf290a1570ea401bf98882128160-Abstract-Datasets_and_Benchmarks.html.

[117] Xiang Deng, Jeff Da, Edwin Pan, Yannis Yiming He, Charles Ide, Kanak Garg, Niklas Lauffer, Andrew Park, Nitin Pasari, Chetan Rane, Karmini Sampath, Maya Krishnan, Srivatsa Kundurthy, Sean Hendryx, Zifan Wang, Chen Bo Calvin Zhang, Noah Jacobson, Bing Liu, and Brad Kenstler. Swe-bench pro: Can AI agents solve long-horizon software engineering tasks? *CoRR*, abs/2509.16941, 2025. doi: 10.48550/ARXIV.2509.16941. URL https://doi.org/10.48550/arXiv.2509.16941.

[118] Yang Deng, Xuan Zhang, Wenxuan Zhang, Yifei Yuan, See-Kiong Ng, and Tat-Seng Chua. On the multi-turn instruction following for conversational web agents. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pages 8795–8812. Association for Computational Linguistics, 2024. doi: 10.18653/V1/2024.ACL-LONG.477. URL https://doi.org/10.18653/v1/2024.acl-long.477.

[119] Yihe Deng, Yu Yang, Junkai Zhang, Wei Wang, and Bo Li. Enhancing llm safety through a theoretical minimax game lens, 2026. URL https://arxiv.org/abs/2502.05163.

[120] 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. *CoRR*, abs/2508.12800, 2025. doi: 10.48550/ARXIV.2508.12800. URL https://doi.org/10.48550/arXiv.2508.12800.

[121] Shehzaad Dhuliawala, Mojtaba Komeili, Jing Xu, Roberta Raileanu, Xian Li, Asli Celikyilmaz, and Jason Weston. Chain-of-verification reduces hallucination in large language models. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Findings of the Association for Computational Linguistics, ACL 2024, Bangkok, Thailand and virtual meeting, August 11-16, 2024*, volume ACL 2024 of *Findings of ACL*, pages 3563–3578. Association for Computational Linguistics, 2024. doi: 10.18653/V1/2024.FINDINGS-ACL.212. URL https://doi.org/10.18653/v1/2024.findings-acl.212.

[122] Bowen Ding, Yuhan Chen, Jiayang Lyu, Jiyao Yuan, Qi Zhu, Shuangshuang Tian, Dantong Zhu, Futing Wang, Heyuan Deng, Fei Mi, Lifeng Shang, and Tao Lin. Rethinking expert trajectory utilization in LLM post-training for mathematical reasoning. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens, editors, *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2026, San Diego, California, United States, July 2-7, 2026*, pages 33081–33106. Association for Computational Linguistics, 2026. URL https://aclanthology.org/2026.acl-long.1528/.

[123] Deming Ding, Shichun Liu, Enhui Yang, Jiahang Lin, Ziying Chen, Shihan Dou, Honglin Guo, Weiyu Cheng, Pengyu Zhao, Chengjun Xiao, Qunhong Zeng, Qi Zhang, Xuanjing Huang, Qidi Xu, and Tao Gui. Octobench: Benchmarking scaffold-aware instruction following in repository-grounded agentic coding. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens, editors, *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2026, San Diego, California, United States, July 2-7, 2026*, pages 5958–5978. Association for Computational Linguistics, 2026. URL https://aclanthology.org/2026.acl-long.269/.

[124] Jingzhe Ding, Shengda Long, Changxin Pu, Huan Zhou, Hongwan Gao, Xiang Gao, Chao He, Yue Hou, Fei Hu, Zhaojian Li, Weiran Shi, Zaiyuan Wang, Daoguang Zan, Chenchen Zhang, Xiaoxu Zhang, Qizhi Chen, Xianfu Cheng, Bo Deng, Qingshui Gu, Kai Hua, Juntao Lin, Pai Liu, Mingchen Li, Xuanguang Pan, Zifan Peng, Yujia Qin, Yong Shan, Zhewen Tan, Weihao Xie, Zihan Wang, Yishuo Yuan, Jiayu Zhang, Enduo Zhao, Yunfei Zhao, He Zhu, Chenyang Zou, Ming Ding, Jianpeng Jiao, Jiaheng Liu, Minghao Liu, Qian Liu, Chongyao Tao, Jian Yang, Tong Yang, Zhaoxiang Zhang, Xinjie Chen, Wenhao Huang, and Ge Zhang. Nl2repo-bench: Towards long-horizon repository generation evaluation of coding agents. *CoRR*, abs/2512.12730, 2025. doi: 10.48550/ARXIV.2512.12730. URL https://doi.org/10.48550/arXiv.2512.12730.

[125] Liang Ding. Agenther: Hindsight experience replay for LLM agent trajectory relabeling. *CoRR*, abs/2603.21357, 2026. doi: 10.48550/ARXIV.2603.21357. URL https://doi.org/10.48550/arXiv.2603.21357.

[126] Shuangrui Ding, Xuanlang Dai, Long Xing, Shengyuan Ding, Ziyu Liu, Yang JingYi, Penghui Yang, Zhixiong Zhang, Xilin Wei, Xinyu Fang, Yubo Ma, Haodong Duan, Jing Shao, Jiaqi Wang, Dahua Lin, Kai Chen, and Yuhang Zang. Wildclawbench: A benchmark for real-world, long-horizon agent evaluation. *CoRR*, abs/2605.10912, 2026. doi: 10.48550/ARXIV.2605.10912. URL https://doi.org/10.48550/arXiv.2605.10912.

[127] Victoria Dochkina. Drop the hierarchy and roles: How self-organizing LLM agents outperform designed structures. *CoRR*, abs/2603.28990, 2026. doi: 10.48550/ARXIV.2603.28990. URL https://doi.org/10.48550/arXiv.2603.28990.

[128] Guanting Dong, Licheng Bao, Zhongyuan Wang, Kangzhi Zhao, Xiaoxi Li, Jiajie Jin, Jinghan Yang, Hangyu Mao, Fuzheng Zhang, Kun Gai, Guorui Zhou, Yutao Zhu, Ji-Rong Wen, and Zhicheng Dou. Agentic entropy-

balanced policy optimization. *CoRR*, abs/2510.14545, 2025. doi: 10.48550/ARXIV.2510.14545. URL https://doi.org/10.48550/arXiv.2510.14545.

[129] Guanting Dong, Yifei Chen, Xiaoxi Li, Jiajie Jin, Hongjin Qian, Yutao Zhu, Hangyu Mao, Guorui Zhou, Zhicheng Dou, and Ji-Rong Wen. Tool-star: Empowering llm-brained multi-tool reasoner via reinforcement learning. *CoRR*, abs/2505.16410, 2025. doi: 10.48550/ARXIV.2505.16410. URL https://doi.org/10.48550/arXiv.2505.16410.

[130] Guanting Dong, Hangyu Mao, Kai Ma, Licheng Bao, Yifei Chen, Zhongyuan Wang, Zhongxia Chen, Jiazhen Du, Huiyang Wang, Fuzheng Zhang, Guorui Zhou, Yutao Zhu, Ji-Rong Wen, and Zhicheng Dou. Agentic reinforced policy optimization. *CoRR*, abs/2507.19849, 2025. doi: 10.48550/ARXIV.2507.19849. URL https://doi.org/10.48550/arXiv.2507.19849.

[131] Guanting Dong, Junting Lu, Junjie Huang, Wanjun Zhong, Longxiang Liu, Shijue Huang, Zhenyu Li, Yang Zhao, Xiaoshuai Song, Xiaoxi Li, Jiajie Jin, Yutao Zhu, Hanbin Wang, Fangyu Lei, Qinyu Luo, Mingyang Chen, Zehui Chen, Jiazhan Feng, Ji-Rong Wen, and Zhicheng Dou. Agent-world: Scaling real-world environment synthesis for evolving general agent intelligence. *CoRR*, abs/2604.18292, 2026. doi: 10.48550/ARXIV.2604.18292. URL https://doi.org/10.48550/arXiv.2604.18292.

[132] Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Jingyuan Ma, Rui Li, Heming Xia, Jingjing Xu, Zhiyong Wu, Baobao Chang, Xu Sun, Lei Li, and Zhifang Sui. A survey on in-context learning. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen, editors, *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, EMNLP 2024, Miami, FL, USA, November 12-16, 2024*, pages 1107–1128. Association for Computational Linguistics, 2024. doi: 10.18653/V1/2024.EMNLP-MAIN.64. URL https://doi.org/10.18653/v1/2024.emnlp-main.64.

[133] Shihan Dou, Ming Zhang, Zhangyue Yin, Chenhao Huang, Yujiong Shen, Junzhe Wang, Jiayi Chen, Yuchen Ni, Junjie Ye, Cheng Zhang, Huaibing Xie, Jianglu Hu, Shaolei Wang, Weichao Wang, Yanling Xiao, Yiting Liu, Zenan Xu, Zhen Guo, Pluto Zhou, Tao Gui, Zuxuan Wu, Xipeng Qiu, Qi Zhang, Xuanjing Huang, Yu-Gang Jiang, Di Wang, and Shunyu Yao. Cl-bench: A benchmark for context learning. *CoRR*, abs/2602.03587, 2026. doi: 10.48550/ARXIV.2602.03587. URL https://doi.org/10.48550/arXiv.2602.03587.

[134] Danny Driess, Fei Xia, Mehdi S. M. Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, Wenlong Huang, Yevgen Chebotar, Pierre Sermanet, Daniel Duckworth, Sergey Levine, Vincent Vanhoucke, Karol Hausman, Marc Toussaint, Klaus Greff, Andy Zeng, Igor Mordatch, and Pete Florence. Palm-e: An embodied multimodal language model. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett, editors, *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pages 8469–8488. PMLR, 2023. URL https://proceedings.mlr.press/v202/driess23a.html.

[135] Mengfei Du, Binhao Wu, Zejun Li, Xuanjing Huang, and Zhongyu Wei. Embspatial-bench: Benchmarking spatial understanding for embodied tasks with large vision-language models. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pages 346–355. Association for Computational Linguistics, 2024. doi: 10.18653/V1/2024.ACL-SHORT.33. URL https://doi.org/10.18653/v1/2024.acl-short.33.

[136] Mingxuan Du, Benfeng Xu, Chiwei Zhu, Xiaorui Wang, and Zhendong Mao. Deepresearch bench: A comprehensive benchmark for deep research agents. *CoRR*, abs/2506.11763, 2025. doi: 10.48550/ARXIV.2506.11763. URL https://doi.org/10.48550/arXiv.2506.11763.

[137] Pengfei Du. Omninova:a general multimodal agent framework. *CoRR*, abs/2503.20028, 2025. doi: 10.48550/ARXIV.2503.20028. URL https://doi.org/10.48550/arXiv.2503.20028.

[138] Yifan Du, Zikang Liu, Jinbiao Peng, Jie Wu, Junyi Li, Jinyang Li, Wayne Xin Zhao, and Ji-Rong Wen. Towards long-horizon agentic multimodal search. *CoRR*, abs/2604.12890, 2026. doi: 10.48550/ARXIV.2604.12890. URL https://doi.org/10.48550/arXiv.2604.12890.

[139] Yilun Du, Shuang Li, Antonio Torralba, Joshua B. Tenenbaum, and Igor Mordatch. Improving factuality and reasoning in language models through multiagent debate. In Ruslan Salakhutdinov, Zico Kolter, Katherine A. Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp, editors, *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*, volume 235 of *Proceedings of Machine Learning Research*, pages 11733–11763. PMLR / OpenReview.net, 2024. URL https://proceedings.mlr.press/v235/du24e.html.

[140] Yu Du, Fangyun Wei, and Hongyang Zhang. Anytool: Self-reflective, hierarchical agents for large-scale API calls. In Ruslan Salakhutdinov, Zico Kolter, Katherine A. Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp, editors, *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*, volume 235 of *Proceedings of Machine Learning Research*, pages 11812–11829. PMLR / OpenReview.net, 2024. URL https://proceedings.mlr.press/v235/du24h.html.

[141] Zane Durante, Qiuyuan Huang, Naoki Wake, Ran Gong, Jae Sung Park, Bidipta Sarkar, Rohan Taori, Yusuke Noda, Demetri Terzopoulos, Yejin Choi, Katsushi Ikeuchi, Hoi Vo, Li Fei-Fei, and Jianfeng Gao. Agent AI: surveying the horizons of multimodal interaction. *CoRR*, abs/2401.03568, 2024. doi: 10.48550/ARXIV.2401.03568. URL https://doi.org/10.48550/arXiv.2401.03568.

[142] Titouan Duston, Shuo Xin, Yang Sun, Daoguang Zan, Aoyan Li, Shulin Xin, Kai Shen, Yixiao Chen, Qiming Sun, Ge Zhang, Jiashuo Liu, Huan Zhou, Jingkai Liu, Zhichen Pu, Yuanheng Wang, Bo-Xuan Ge, Xin Tong, Fei Ye, Zhi-Chao Zhao, Wen-Biao Han, Zhoujian Cao, Yueran Zhao, Weiluo Ren, Qingshen Long, Yuxiao Liu, Anni Huang, Yidi Du, Yuanyuan Rong, and Jiahao Peng. Ainsteinbench: Benchmarking coding agents on scientific repositories. *CoRR*, abs/2512.21373, 2025. doi: 10.48550/ARXIV.2512.21373. URL https://doi.org/10.48550/arXiv.2512.21373.

[143] Darren Edge, Ha Trinh, Newman Cheng, Joshua Bradley, Alex Chao, Apurva Mody, Steven Truitt, and Jonathan Larson. From local to global: A graph RAG approach to query-focused summarization. *CoRR*, abs/2404.16130, 2024. doi: 10.48550/ARXIV.2404.16130. URL https://doi.org/10.48550/arXiv.2404.16130.

[144] Abul Ehtesham, Aditi Singh, Gaurav Kumar Gupta, and Saket Kumar. A survey of agent interoperability protocols: Model context protocol (mcp), agent communication protocol (acp), agent-to-agent protocol (a2a), and agent network protocol (ANP). *CoRR*, abs/2505.02279, 2025. doi: 10.48550/ARXIV.2505.02279. URL https://doi.org/10.48550/arXiv.2505.02279.

[145] Linxi Fan, Guanzhi Wang, Yunfan Jiang, Ajay Mandlekar, Yuncong Yang, Haoyi Zhu, Andrew Tang, De-An Huang, Yuke Zhu, and Anima Anandkumar. Minedojo: Building open-ended embodied agents with internet-scale knowledge. In Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022. URL http://papers.nips.cc/paper_files/paper/2022/hash/74a67268c5cc5910f64938cac4526a90-Abstract-Datasets_and_Benchmarks.html.

[146] Sunqi Fan, Meng-Hao Guo, and Shuojin Yang. Agentic keyframe search for video question answering. *CoRR*, abs/2503.16032, 2025. doi: 10.48550/ARXIV.2503.16032. URL https://doi.org/10.48550/arXiv.2503.16032.

[147] Jinyuan Fang, Yanwen Peng, Xi Zhang, Yingxu Wang, Xinhao Yi, Guibin Zhang, Yi Xu, Bin Wu, Siwei Liu, Zihao Li, Zhaochun Ren, Nikos Aletras, Xi Wang, Han Zhou, and Zaiqiao Meng. A comprehensive survey of self-evolving AI agents: A new paradigm bridging foundation models and lifelong agentic systems. *CoRR*, abs/2508.07407, 2025. doi: 10.48550/ARXIV.2508.07407. URL https://doi.org/10.48550/arXiv.2508.07407.

[148] Runnan Fang, Shihao Cai, Baixuan Li, Jialong Wu, Guangyu Li, Wenbiao Yin, Xinyu Wang, Xiaobin Wang, Liangcai Su, Zhen Zhang, Shibin Wu, Zhengwei Tao, Yong Jiang, Pengjun Xie, Ningyu Zhang, Fei Huang, Wentao Zhang, and Jingren Zhou. Towards general agentic intelligence via environment scaling. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens, editors, *Findings of the Association for Computational Linguistics, ACL 2026, San Diego, California, United States, July 2-7, 2026*, pages 17610–17621. Association for Computational Linguistics, 2026. URL https://aclanthology.org/2026.findings-acl.872/.

[149] Tianqing Fang, Hongming Zhang, Zhisong Zhang, Kaixin Ma, Wenhao Yu, Haitao Mi, and Dong Yu. Webevolver: Enhancing web agent self-improvement with co-evolving world model. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng, editors, *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, EMNLP 2025, Suzhou, China, November 4-9, 2025*, pages 8959–8975. Association for Computational Linguistics, 2025. doi: 10.18653/V1/2025.EMNLP-MAIN.454. URL https://doi.org/10.18653/v1/2025.emnlp-main.454.

[150] Zhaiyu Fang and Ruipeng Sun. Adatir: Adaptive tool-integrated reasoning via difficulty-aware policy optimization. *CoRR*, abs/2601.14696, 2026. doi: 10.48550/ARXIV.2601.14696. URL https://doi.org/10.48550/arXiv.2601.14696.

[151] Sebastian Farquhar, Jannik Kossen, Lorenz Kuhn, and Yarin Gal. Detecting hallucinations in large language models using semantic entropy. *Nat.*, 630(8017):625–630, 2024. doi: 10.1038/S41586-024-07421-0. URL https://doi.org/10.1038/s41586-024-07421-0.

[152] William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *J. Mach. Learn. Res.*, 23:120:1–120:39, 2022. URL https://jmlr.org/papers/v23/21-0998.html.

[153] Xiang Fei, Xiawu Zheng, and Hao Feng. Mcp-zero: Active tool discovery for autonomous LLM agents. *CoRR*, abs/2506.01056, 2025. doi: 10.48550/ARXIV.2506.01056. URL https://doi.org/10.48550/arXiv.2506.01056.

[154] Yuval Felendler, Parth Atulbhai Gandhi, Idan Habler, Yuval Elovici, and Asaf Shabtai. From tool orchestration to code execution: A study of MCP design choices. *CoRR*, abs/2602.15945, 2026. doi: 10.48550/ARXIV.2602.15945. URL https://doi.org/10.48550/arXiv.2602.15945.

[155] K. J. Kevin Feng, David W. McDonald, and Amy X. Zhang. Levels of autonomy for AI agents. *CoRR*, abs/2506.12469, 2025. doi: 10.48550/ARXIV.2506.12469. URL https://doi.org/10.48550/arXiv.2506.12469.

[156] Lang Feng, Zhenghai Xue, Tingcong Liu, and Bo An. Group-in-group policy optimization for LLM agent training. *CoRR*, abs/2505.10978, 2025. doi: 10.48550/ARXIV.2505.10978. URL https://doi.org/10.48550/arXiv.2505.10978.

[157] Adam Fourney, Gagan Bansal, Hussein Mozannar, Cheng Tan, Eduardo Salinas, Erkang Zhu, Friederike Niedtner, Grace Proebsting, Griffin Bassman, Jack Gerrits, Jacob Alber, Peter Chang, Ricky Loynd, Robert West, Victor Dibia, Ahmed Awadallah, Ece Kamar, Rafah Hosn, and Saleema Amershi. Magentic-one: A generalist multi-agent system for solving complex tasks. *CoRR*, abs/2411.04468, 2024. doi: 10.48550/ARXIV.2411.04468. URL https://doi.org/10.48550/arXiv.2411.04468.

[158] Chaoyou Fu, Yuhan Dai, Yongdong Luo, Lei Li, Shuhuai Ren, Renrui Zhang, Zihan Wang, Chenyu Zhou, Yunhang Shen, Mengdan Zhang, Peixian Chen, Yanwei Li, Shaohui Lin, Sirui Zhao, Ke Li, Tong Xu, Xiawu Zheng, Enhong Chen, Caifeng Shan, Ran He, and Xing Sun. Video-mme: The first-ever comprehensive evaluation benchmark of multi-modal llms in video analysis. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2025, Nashville, TN, USA, June 11-15, 2025*, pages 24108–24118, 2025. doi: 10.1109/CVPR52734.2025.02245. URL https://openaccess.thecvf.com/content/CVPR2025/html/Fu_Video-MME_The_First-Ever_Comprehensive_Evaluation_Benchmark_of_Multi-modal_LLMs_in_CVPR_2025_paper.html.

[159] Ling Fu, Zhebin Kuang, Jiajun Song, Mingxin Huang, Biao Yang, Yuzhe Li, Linghao Zhu, Qidi Luo, Xinyu Wang, Hao Lu, Zhang Li, Guozhi Tang, Bin Shan, Chunhui Lin, Qi Liu, Binghong Wu, Hao Feng, Hao Liu, Can Huang, Jingqun Tang, Wei Chen, Lianwen Jin, Yuliang Liu, and Xiang Bai. Ocrbench v2: An improved benchmark for evaluating large multimodal models on visual text localization and reasoning. In Danielle Belgrave, Cheng Zhang, Laura N. Montoya, Hsuan-Tien Lin, Razvan Pascanu, Piotr Koniusz, Marzyeh Ghassemi, Nancy Chen, Iván Vladimir Meza Ruíz, and Arturo Loaiza-Bonilla, editors, *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2025, NeurIPS 2025, San Diego, CA, USA, December 2-7, 2025 / Mexico City, Mexico, November 30 - December 5, 2025*, 2025. URL http://papers.nips.cc/paper_files/paper/2025/hash/8c2e6bb15be1894b8fb4e0f9bcad1739-Abstract-Datasets_and_Benchmarks_Track.html.

[160] Yao Fu, Hao Peng, Ashish Sabharwal, Peter Clark, and Tushar Khot. Complexity-based prompting for multi-step reasoning. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023. URL https://openreview.net/forum?id=yf1icZHC-l9.

[161] Yuqian Fu, Haohuan Huang, Kaiwen Jiang, Yuanheng Zhu, and Dongbin Zhao. Revisiting on-policy distillation: Empirical failure modes and simple fixes. *CoRR*, abs/2603.25562, 2026. doi: 10.48550/ARXIV.2603.25562. URL https://doi.org/10.48550/arXiv.2603.25562.

[162] Tiantian Gan and Qiyao Sun. RAG-MCP: mitigating prompt bloat in LLM tool selection via retrieval-augmented generation. *CoRR*, abs/2505.03275, 2025. doi: 10.48550/ARXIV.2505.03275. URL https://doi.org/10.48550/arXiv.2505.03275.

[163] Shubham Gandhi, Jason Tsay, Jatin Ganhotra, Kiran Kate, and Yara Rizk. When agents go astray: Course-correcting SWE agents with prms. *CoRR*, abs/2509.02360, 2025. doi: 10.48550/ARXIV.2509.02360. URL https://doi.org/10.48550/arXiv.2509.02360.

[164] Ishaan Gangwani and Aayam Bansal. Rubric as reward: Decomposing verification signals for logical reasoning in GRPO. In *ICLR 2026 Workshop on Logical Reasoning of Large Language Models*, 2026. URL https://openreview.net/forum?id=LFBnQEGh23.

[165] Huan-ang Gao, Jiayi Geng, Wenyue Hua, Mengkang Hu, Xinzhe Juan, Hongzhang Liu, Shilong Liu, Jiahao Qiu, Xuan Qi, Qihan Ren, Yiran Wu, Hongru Wang, Han Xiao, Yuhang Zhou, Shaokun Zhang, Jiayi Zhang,

Jinyu Xiang, Yixiong Fang, Qiwen Zhao, Dongrui Liu, Cheng Qian, Zhenhailong Wang, Minda Hu, Huazheng Wang, Qingyun Wu, Heng Ji, and Mengdi Wang. A survey of self-evolving agents: What, when, how, and where to evolve on the path to artificial super intelligence. *Trans. Mach. Learn. Res.*, 2026, 2026. URL https://openreview.net/forum?id=CTr3bovS5F.

[166] 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. *CoRR*, abs/2508.07976, 2025. doi: 10.48550/ARXIV.2508.07976. URL https://doi.org/10.48550/arXiv.2508.07976.

[167] Luyu Gao, Xueguang Ma, Jimmy Lin, and Jamie Callan. Precise zero-shot dense retrieval without relevance labels. In Anna Rogers, Jordan L. Boyd-Graber, and Naoaki Okazaki, editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 1762–1777. Association for Computational Linguistics, 2023. doi: 10.18653/V1/2023.ACL-LONG.99. URL https://doi.org/10.18653/v1/2023.acl-long.99.

[168] Luyu Gao, Aman Madaan, Shuyan Zhou, Uri Alon, Pengfei Liu, Yiming Yang, Jamie Callan, and Graham Neubig. PAL: program-aided language models. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett, editors, *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pages 10764–10799. PMLR, 2023. URL https://proceedings.mlr.press/v202/gao23f.html.

[169] Pengfei Gao, Zhao Tian, Xiangxin Meng, Xinchen Wang, Ruida Hu, Yuanan Xiao, Yizhou Liu, Zhao Zhang, Junjie Chen, Cuiyun Gao, Yun Lin, Yingfei Xiong, Chao Peng, and Xia Liu. Trae agent: An llm-based agent for software engineering with test-time scaling. *CoRR*, abs/2507.23370, 2025. doi: 10.48550/ARXIV.2507.23370. URL https://doi.org/10.48550/arXiv.2507.23370.

[170] Shenyuan Gao, Siyuan Zhou, Yilun Du, Jun Zhang, and Chuang Gan. Adaworld: Learning adaptable world models with latent actions. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu, editors, *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*, volume 267 of *Proceedings of Machine Learning Research*. PMLR / OpenReview.net, 2025. URL https://proceedings.mlr.press/v267/gao25u.html.

[171] Tianyu Gao, Alexander Wettig, Howard Yen, and Danqi Chen. How to train long-context language models (effectively). In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar, editors, *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, pages 7376–7399. Association for Computational Linguistics, 2025. doi: 10.18653/V1/2025.ACL-LONG.366. URL https://doi.org/10.18653/v1/2025.acl-long.366.

[172] Yu Gao, Lixue Gong, Qiushan Guo, Xiaoxia Hou, Zhichao Lai, Fanshi Li, Liang Li, Xiaochen Lian, Chao Liao, Liyang Liu, Wei Liu, Yichun Shi, Shiqi Sun, Yu Tian, Zhi Tian, Peng Wang, Rui Wang, Xuanda Wang, Xun Wang, Ye Wang, Guofeng Wu, Jie Wu, Xin Xia, Xuefeng Xiao, Zhonghua Zhai, Xinyu Zhang, Qi Zhang, Yuwei Zhang, Shijia Zhao, Jianchao Yang, and Weilin Huang. Seedream 3.0 technical report. *CoRR*, abs/2504.11346, 2025. doi: 10.48550/ARXIV.2504.11346. URL https://doi.org/10.48550/arXiv.2504.11346.

[173] Yubin Ge, Salvatore Romeo, Jason Cai, Monica Sunkara, and Yi Zhang. SAMULE: self-learning agents enhanced by multi-level reflection. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng, editors, *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, EMNLP 2025, Suzhou, China, November 4-9, 2025*, pages 16591–16610. Association for Computational Linguistics, 2025. doi: 10.18653/V1/2025.EMNLP-MAIN.839. URL https://doi.org/10.18653/v1/2025.emnlp-main.839.

[174] Hejia Geng and Leo Liu. Parthenon law: A self-evolving legal-agent framework. *CoRR*, abs/2606.04602, 2026. doi: 10.48550/ARXIV.2606.04602. URL https://doi.org/10.48550/arXiv.2606.04602.

[175] Xinyu Geng, Peng Xia, Zhen Zhang, Xinyu Wang, Qiuchen Wang, Ruixue Ding, Chenxi Wang, Jialong Wu, Yida Zhao, Kuan Li, Yong Jiang, Pengjun Xie, Fei Huang, and Jingren Zhou. Webwatcher: Breaking new frontier of vision-language deep research agent. *CoRR*, abs/2508.05748, 2025. doi: 10.48550/ARXIV.2508.05748. URL https://doi.org/10.48550/arXiv.2508.05748.

[176] Hafez Ghaemi, Eilif B. Muller, and Shahab Bakhtiari. seq-jepa: Autoregressive predictive learning of invariant-equivariant world models. *CoRR*, abs/2505.03176, 2025. doi: 10.48550/ARXIV.2505.03176. URL https://doi.org/10.48550/arXiv.2505.03176.

[177] Ali Essa Ghareeb, Benjamin Chang, Ludovico Mitchener, Angela Yiu, Caralyn J. Szostkiewicz, Jon M. Laurent, Muhammed T. Razzak, Andrew D. White, Michaela M. Hinks, and Samuel G. Rodriques. Robin: A multi-agent

system for automating scientific discovery. *CoRR*, abs/2505.13400, 2025. doi: 10.48550/ARXIV.2505.13400. URL https://doi.org/10.48550/arXiv.2505.13400.

[178] GLM. GLM-4.5: agentic, reasoning, and coding (ARC) foundation models. *CoRR*, abs/2508.06471, 2025. doi: 10.48550/ARXIV.2508.06471. URL https://doi.org/10.48550/arXiv.2508.06471.

[179] GLM. GLM-5: from vibe coding to agentic engineering. *CoRR*, abs/2602.15763, 2026. doi: 10.48550/ARXIV.2602.15763. URL https://doi.org/10.48550/arXiv.2602.15763.

[180] Fabian Gloeckle, Badr Youbi Idrissi, Baptiste Rozière, David Lopez-Paz, and Gabriel Synnaeve. Better & faster large language models via multi-token prediction. In Ruslan Salakhutdinov, Zico Kolter, Katherine A. Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp, editors, *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*, volume 235 of *Proceedings of Machine Learning Research*, pages 15706–15734. PMLR / OpenReview.net, 2024. URL https://proceedings.mlr.press/v235/gloeckle24a.html.

[181] Boyu Gou, Ruohan Wang, Boyuan Zheng, Yanan Xie, Cheng Chang, Yiheng Shu, Huan Sun, and Yu Su. Navigating the digital world as humans do: Universal visual grounding for GUI agents. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=kxnoqaisCT.

[182] Zhibin Gou, Zhihong Shao, Yeyun Gong, Yelong Shen, Yujiu Yang, Nan Duan, and Weizhu Chen. CRITIC: large language models can self-correct with tool-interactive critiquing. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=Sx038qxjek.

[183] Significant Gravitas. Autogpt: Build, deploy, and run ai agents, 2023. URL https://github.com/Significant-Gravitas/AutoGPT.

[184] Dirk Groeneveld, Iz Beltagy, Evan Pete Walsh, Akshita Bhagia, Rodney Kinney, Oyvind Tafjord, Ananya Harsh Jha, Hamish Ivison, Ian Magnusson, Yizhong Wang, Shane Arora, David Atkinson, Russell Authur, Khyathi Raghavi Chandu, Arman Cohan, Jennifer Dumas, Yanai Elazar, Yuling Gu, Jack Hessel, Tushar Khot, William Merrill, Jacob Morrison, Niklas Muennighoff, Aakanksha Naik, Crystal Nam, Matthew E. Peters, Valentina Pyatkin, Abhilasha Ravichander, Dustin Schwenk, Saurabh Shah, Will Smith, Emma Strubell, Nishant Subramani, Mitchell Wortsman, Pradeep Dasigi, Nathan Lambert, Kyle Richardson, Luke Zettlemoyer, Jesse Dodge, Kyle Lo, Luca Soldaini, Noah A. Smith, and Hannaneh Hajishirzi. Olmo: Accelerating the science of language models. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pages 15789–15809. Association for Computational Linguistics, 2024. doi: 10.18653/V1/2024.ACL-LONG.841. URL https://doi.org/10.18653/v1/2024.acl-long.841.

[185] Niko Grupen, Gabe Pereyra, and Julio Pereyra. Introducing harvey's legal agent benchmark. Harvey Blog, 2026. URL https://www.harvey.ai/blog/introducing-harveys-legal-agent-benchmark.

[186] Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. *CoRR*, abs/2312.00752, 2023. doi: 10.48550/ARXIV.2312.00752. URL https://doi.org/10.48550/arXiv.2312.00752.

[187] Albert Gu, Karan Goel, and Christopher Ré. Efficiently modeling long sequences with structured state spaces. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022. URL https://openreview.net/forum?id=uYLFoz1vlAC.

[188] Yu Gu, Kai Zhang, Yuting Ning, Boyuan Zheng, Boyu Gou, Tianci Xue, Cheng Chang, Sanjari Srivastava, Yanan Xie, Peng Qi, Huan Sun, and Yu Su. Is your LLM secretly a world model of the internet? model-based planning for web agents. *Trans. Mach. Learn. Res.*, 2025, 2025. URL https://openreview.net/forum?id=c6l7yA0HSq.

[189] Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. Minillm: On-policy distillation of large language models, 2026. URL https://arxiv.org/abs/2306.08543.

[190] Tianrui Guan, Fuxiao Liu, Xiyang Wu, Ruiqi Xian, Zongxia Li, Xiaoyu Liu, Xijun Wang, Lichang Chen, Furong Huang, Yaser Yacoob, Dinesh Manocha, and Tianyi Zhou. Hallusionbench: An advanced diagnostic suite for entangled language hallucination and visual illusion in large vision-language models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024*, pages 14375–14385. IEEE, 2024. doi: 10.1109/CVPR52733.2024.01363. URL https://doi.org/10.1109/CVPR52733.2024.01363.

[191] Xinyu Guan, Li Lyna Zhang, Yifei Liu, Ning Shang, Youran Sun, Yi Zhu, Fan Yang, and Mao Yang. rstar-math: Small llms can master math reasoning with self-evolved deep thinking. In Aarti Singh, Maryam Fazel,

Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu, editors, *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*, volume 267 of *Proceedings of Machine Learning Research*. PMLR / OpenReview.net, 2025. URL https://proceedings.mlr.press/v267/guan25f.html.

[192] Anisha Gunjal, Anthony Wang, Elaine Lau, Vaskar Nath, Bing Liu, and Sean Hendryx. Rubrics as rewards: Reinforcement learning beyond verifiable domains. *CoRR*, abs/2507.17746, 2025. doi: 10.48550/ARXIV.2507.17746. URL https://doi.org/10.48550/arXiv.2507.17746.

[193] Chengquan Guo, Xun Liu, Chulin Xie, Andy Zhou, Yi Zeng, Zinan Lin, Dawn Song, and Bo Li. Redcode: Risky code execution and generation benchmark for code agents. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang, editors, *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. URL http://papers.nips.cc/paper_files/paper/2024/hash/bfd082c452dffb450d5a5202b0419205-Abstract-Datasets_and_Benchmarks_Track.html.

[194] Dong Guo, Faming Wu, Feida Zhu, Fuxing Leng, Guang Shi, Haobin Chen, Haoqi Fan, Jian Wang, Jianyu Jiang, Jiawei Wang, Jingji Chen, Jingjia Huang, Kang Lei, Liping Yuan, Lishu Luo, Pengfei Liu, Qinghao Ye, Rui Qian, Shen Yan, Shixiong Zhao, Shuai Peng, Shuangye Li, Sihang Yuan, Sijin Wu, Tianheng Cheng, Weiwei Liu, Wenqian Wang, Xianhan Zeng, Xiao Liu, Xiaobo Qin, Xiaohan Ding, Xiaojun Xiao, Xiaoying Zhang, Xuanwei Zhang, Xuehan Xiong, Yanghua Peng, Yangrui Chen, Yanwei Li, Yanxu Hu, Yi Lin, Yiyuan Hu, Yiyuan Zhang, Youbin Wu, Yu Li, Yudong Liu, Yue Ling, Yujia Qin, Zanbo Wang, Zhiwu He, Aoxue Zhang, Bairen Yi, Bencheng Liao, Can Huang, Can Zhang, Chaorui Deng, Chaoyi Deng, Cheng Lin, Cheng Yuan, Chenggang Li, Chenhui Gou, Chenwei Lou, Chengzhi Wei, Chundian Liu, Chunyuan Li, Deyao Zhu, Donghong Zhong, Feng Li, Feng Zhang, Gang Wu, Guodong Li, Guohong Xiao, Haibin Lin, Haihua Yang, Haoming Wang, Heng Ji, Hongxiang Hao, Hui Shen, Huixia Li, Jiahao Li, Jialong Wu, Jianhua Zhu, Jianpeng Jiao, Jiashi Feng, Jiaze Chen, Jianhui Duan, Jihao Liu, Jin Zeng, Jingqun Tang, Jingyu Sun, Joya Chen, Jun Long, Junda Feng, Junfeng Zhan, Junjie Fang, Junting Lu, Kai Hua, Kai Liu, Kai Shen, Kaiyuan Zhang, Ke Shen, Ke Wang, Keyu Pan, Kun Zhang, Kunchang Li, Lanxin Li, Lei Li, Lei Shi, Li Han, Liang Xiang, Liangqiang Chen, Lin Chen, Lin Li, Lin Yan, Liying Chi, Longxiang Liu, Mengfei Du, Mingxuan Wang, Ningxin Pan, Peibin Chen, Pengfei Chen, Pengfei Wu, Qingqing Yuan, Qingyao Shuai, Qiuyan Tao, Renjie Zheng, Renrui Zhang, Ru Zhang, Rui Wang, Rui Yang, Rui Zhao, Shaoqiang Xu, Shihao Liang, Shipeng Yan, Shu Zhong, Shuaishuai Cao, Shuangzhi Wu, Shufan Liu, Shuhan Chang, Songhua Cai, Tenglong Ao, Tianhao Yang, Tingting Zhang, Wanjun Zhong, Wei Jia, Wei Weng, Weihao Yu, Wenhao Huang, Wenjia Zhu, Wenli Yang, Wenzhi Wang, Xiang Long, XiangRui Yin, Xiao Li, Xiaolei Zhu, Xiaoying Jia, Xijin Zhang, Xin Liu, Xinchen Zhang, Xinyu Yang, Xiongcai Luo, Xiuli Chen, Xuantong Zhong, Xuefeng Xiao, Xujing Li, Yan Wu, Yawei Wen, Yifan Du, Yihao Zhang, Yining Ye, Yonghui Wu, Yu Liu, Yu Yue, Yufeng Zhou, Yufeng Yuan, Yuhang Xu, Yuhong Yang, Yun Zhang, Yunhao Fang, Yuntao Li, Yurui Ren, Yuwen Xiong, Zehua Hong, Zehua Wang, Zewei Sun, Zeyu Wang, Zhao Cai, Zhaoyue Zha, Zhecheng An, Zhehui Zhao, Zhengzhuo Xu, Zhipeng Chen, Zhiyong Wu, Zhuofan Zheng, Zihao Wang, Zilong Huang, Ziyu Zhu, and Zuquan Song. Seed1.5-vl technical report, 2025. URL https://arxiv.org/abs/2505.07062.

[195] Jiacheng Guo, Ling Yang, Peter Chen, Qixin Xiao, Yinjie Wang, Xinzhe Juan, Jiahao Qiu, Ke Shen, and Mengdi Wang. Genenv: Difficulty-aligned co-evolution between LLM agents and environment simulators. *CoRR*, abs/2512.19682, 2025. doi: 10.48550/ARXIV.2512.19682. URL https://doi.org/10.48550/arXiv.2512.19682.

[196] Junliang Guo, Yang Ye, Tianyu He, Haoyu Wu, Yushu Jiang, Tim Pearce, and Jiang Bian. Mineworld: a real-time and open-source interactive world model on minecraft. *CoRR*, abs/2504.08388, 2025. doi: 10.48550/ARXIV.2504.08388. URL https://doi.org/10.48550/arXiv.2504.08388.

[197] Taicheng Guo, Xiuying Chen, Yaqi Wang, Ruidi Chang, Shichao Pei, Nitesh V. Chawla, Olaf Wiest, and Xiangliang Zhang. Large language model based multi-agents: A survey of progress and challenges. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI 2024, Jeju, South Korea, August 3-9, 2024*, pages 8048–8057. ijcai.org, 2024. URL https://www.ijcai.org/proceedings/2024/890.

[198] Zhicheng Guo, Sijie Cheng, Hao Wang, Shihao Liang, Yujia Qin, Peng Li, Zhiyuan Liu, Maosong Sun, and Yang Liu. Stabletoolbench: Towards stable large-scale benchmarking on tool learning of large language models. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Findings of the Association for Computational Linguistics, ACL 2024, Bangkok, Thailand and virtual meeting, August 11-16, 2024*, volume ACL 2024 of *Findings of ACL*, pages 11143–11156. Association for Computational Linguistics, 2024. doi: 10.18653/V1/2024.FINDINGS-ACL.664. URL https://doi.org/10.18653/v1/2024.findings-acl.664.

[199] Bernal Jimenez Gutierrez, Yiheng Shu, Yu Gu, Michihiro Yasunaga, and Yu Su. Hipporag: Neurobiologically inspired long-term memory for large language models. In Amir Globersons, Lester Mackey, Danielle Belgrave,

Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang, editors, *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. URL http://papers.nips.cc/paper_files/paper/2024/hash/6ddc001d07ca4f319af96a3024f6dbd1-Abstract-Conference.html.

[200] Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. REALM: retrieval-augmented language model pre-training. *CoRR*, abs/2002.08909, 2020. URL https://arxiv.org/abs/2002.08909.

[201] Ai Han, Junxing Hu, Pu Wei, Zhiqian Zhang, Yuhang Guo, Jiawei Lu, Zhen Chen, and Zicheng Zhang. Joyagents-r1: Accelerating multi-agent evolution dynamics with variance-reduction group relative policy optimization, 2026. URL https://openreview.net/forum?id=U7n8gZGyAu.

[202] Dongge Han, Camille Couturier, Daniel Madrigal Díaz, Xuchao Zhang, Victor Rühle, and Saravan Rajmohan. Legomem: Modular procedural memory for multi-agent LLM systems for workflow automation. *CoRR*, abs/2510.04851, 2025. doi: 10.48550/ARXIV.2510.04851. URL https://doi.org/10.48550/arXiv.2510.04851.

[203] Hao Han, Jin Xie, Xuehao Ma, Weiquan Zhu, Ziyao Zhang, ZhiLiang Long, Hongkai Chen, and Qingwen Ye. SWE-TRACE: optimizing long-horizon SWE agents through rubric process reward models and heuristic test-time scaling. *CoRR*, abs/2604.14820, 2026. doi: 10.48550/ARXIV.2604.14820. URL https://doi.org/10.48550/arXiv.2604.14820.

[204] Shibo Hao, Yi Gu, Haodi Ma, Joshua Jiahua Hong, Zhen Wang, Daisy Zhe Wang, and Zhiting Hu. Reasoning with language model is planning with world model. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 8154–8173. Association for Computational Linguistics, 2023. doi: 10.18653/V1/2023.EMNLP-MAIN.507. URL https://doi.org/10.18653/v1/2023.emnlp-main.507.

[205] Yaru Hao, Li Dong, Xun Wu, Shaohan Huang, Zewen Chi, and Furu Wei. On-policy RL with optimal reward baseline. *CoRR*, abs/2505.23585, 2025. doi: 10.48550/ARXIV.2505.23585. URL https://doi.org/10.48550/arXiv.2505.23585.

[206] Zhezheng Hao, Hong Wang, Jian Luo, Jianqing Zhang, Yuyan Zhou, Qiang Lin, Can Wang, Hande Dong, and Jiawei Chen. Recreate: Reasoning and creating domain agents driven by experience. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens, editors, *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2026, San Diego, California, United States, July 2-7, 2026*, pages 31018–31046. Association for Computational Linguistics, 2026. URL https://aclanthology.org/2026.acl-long.1432/.

[207] Guanzhong He, Zhen Yang, Jinxin Liu, Bin Xu, Lei Hou, and Juanzi Li. Webseer: Training deeper search agents through reinforcement learning with self-reflection. *CoRR*, abs/2510.18798, 2025. doi: 10.48550/ARXIV.2510.18798. URL https://doi.org/10.48550/arXiv.2510.18798.

[208] Hongliang He, Wenlin Yao, Kaixin Ma, Wenhao Yu, Yong Dai, Hongming Zhang, Zhenzhong Lan, and Dong Yu. Webvoyager: Building an end-to-end web agent with large multimodal models. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pages 6864–6890. Association for Computational Linguistics, 2024. doi: 10.18653/V1/2024.ACL-LONG.371. URL https://doi.org/10.18653/v1/2024.acl-long.371.

[209] Jun He, Junyan Ye, Zilong Huang, Dongzhi Jiang, Chenjue Zhang, Leqi Zhu, Renrui Zhang, Xiang Zhang, and Weijia Li. Mind-brush: Integrating agentic cognitive search and reasoning into image generation. *CoRR*, abs/2602.01756, 2026. doi: 10.48550/ARXIV.2602.01756. URL https://doi.org/10.48550/arXiv.2602.01756.

[210] Kaiwen He, Zhiwei Wang, Chenyi Zhuang, and Jinjie Gu. Recon-act: A self-evolving multi-agent browser-use system via web reconnaissance, tool generation, and task execution. *CoRR*, abs/2509.21072, 2025. doi: 10.48550/ARXIV.2509.21072. URL https://doi.org/10.48550/arXiv.2509.21072.

[211] Muyu He, Adit Jain, Anand Kumar, Vincent Tu, Soumyadeep Bakshi, Sachin Patro, and Nazneen Rajani. Yc-bench: Benchmarking AI agents for long-term planning and consistent execution. *CoRR*, abs/2604.01212, 2026. doi: 10.48550/ARXIV.2604.01212. URL https://doi.org/10.48550/arXiv.2604.01212.

[212] Shuo He, Lang Feng, Qi Wei, Xin Cheng, Lei Feng, and Bo An. Hierarchy-of-groups policy optimization for long-horizon agentic tasks. *CoRR*, abs/2602.22817, 2026. doi: 10.48550/ARXIV.2602.22817. URL https://doi.org/10.48550/arXiv.2602.22817.

[213] Wei He, Yueqing Sun, Hongyan Hao, Xueyuan Hao, Zhikang Xia, Qi Gu, Chengcheng Han, Dengchang Zhao, Hui Su, Kefeng Zhang, Man Gao, Xi Su, Xiaodong Cai, Xunliang Cai, Yu Yang, and Yunke Zhao. Vitabench: Benchmarking LLM agents with versatile interactive tasks in real-world applications. *CoRR*, abs/2509.26490, 2025. doi: 10.48550/ARXIV.2509.26490. URL https://doi.org/10.48550/arXiv.2509.26490.

[214] Xu He, Di Wu, Yan Zhai, and Kun Sun. Sentinelagent: Graph-based anomaly detection in multi-agent systems. *CoRR*, abs/2505.24201, 2025. doi: 10.48550/ARXIV.2505.24201. URL https://doi.org/10.48550/arXiv.2505.24201.

[215] Yifei He, Pranit Chawla, Yaser Souri, Subhojit Som, and Xia Song. Webstar: Scalable data synthesis for computer use agents with step-level filtering. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens, editors, *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2026, San Diego, California, United States, July 2-7, 2026*, pages 516–533. Association for Computational Linguistics, 2026. URL https://aclanthology.org/2026.acl-long.21/.

[216] Yinghui He, Simran Kaur, Adithya Bhaskar, Yongjin Yang, Jiarui Liu, Narutatsu Ri, Liam Fowl, Abhishek Panigrahi, Danqi Chen, and Sanjeev Arora. Self-distillation zero: Self-revision turns binary rewards into dense supervision. *CoRR*, abs/2604.12002, 2026. doi: 10.48550/ARXIV.2604.12002. URL https://doi.org/10.48550/arXiv.2604.12002.

[217] Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Jack W. Rae, Oriol Vinyals, and Laurent Sifre. Training compute-optimal large language models. *CoRR*, abs/2203.15556, 2022. doi: 10.48550/ARXIV.2203.15556. URL https://doi.org/10.48550/arXiv.2203.15556.

[218] Haoyang Hong, Jiajun Yin, Yuan Wang, Jingnan Liu, Zhe Chen, Ailing Yu, Ji Li, Zhiling Ye, Hansong Xiao, Yefei Chen, Hualei Zhou, Yun Yue, Minghui Yang, Chunxiao Guo, Junwei Liu, Peng Wei, and Jinjie Gu. Multi-agent deep research: Training multi-agent systems with M-GRPO. *CoRR*, abs/2511.13288, 2025. doi: 10.48550/ARXIV.2511.13288. URL https://doi.org/10.48550/arXiv.2511.13288.

[219] Sirui Hong, Mingchen Zhuge, Jonathan Chen, Xiawu Zheng, Yuheng Cheng, Jinlin Wang, Ceyao Zhang, Zili Wang, Steven Ka Shing Yau, Zijuan Lin, Liyang Zhou, Chenyu Ran, Lingfeng Xiao, Chenglin Wu, and Jürgen Schmidhuber. Metagpt: Meta programming for A multi-agent collaborative framework. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=VtmBAGCN7o.

[220] Cheng-Ping Hsieh, Simeng Sun, Samuel Kriman, Shantanu Acharya, Dima Rekesh, Fei Jia, Yang Zhang, and Boris Ginsburg. RULER: what's the real context size of your long-context language models? *CoRR*, abs/2404.06654, 2024. doi: 10.48550/ARXIV.2404.06654. URL https://doi.org/10.48550/arXiv.2404.06654.

[221] Chuanrui Hu, Xingze Gao, Zuyi Zhou, Dannong Xu, Yi Bai, Xintong Li, Hui Zhang, Tong Li, Chong Zhang, Lidong Bing, and Yafeng Deng. EverMemOS: A self-organizing memory operating system for structured long-horizon reasoning. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens, editors, *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 45836–45853, San Diego, California, United States, July 2026. Association for Computational Linguistics. ISBN 979-8-89176-390-6. doi: 10.18653/v1/2026.acl-long.2125. URL https://aclanthology.org/2026.acl-long.2125/.

[222] Jian Hu. REINFORCE++: A simple and efficient approach for aligning large language models. *CoRR*, abs/2501.03262, 2025. doi: 10.48550/ARXIV.2501.03262. URL https://doi.org/10.48550/arXiv.2501.03262.

[223] Liang Hu, Jianpeng Jiao, Jiashuo Liu, Yanle Ren, Zhoufutu Wen, Kaiyuan Zhang, Xuanliang Zhang, Xiang Gao, Tianci He, Fei Hu, Yali Liao, Zaiyuan Wang, Chenghao Yang, Qianyu Yang, Mingren Yin, Zhiyuan Zeng, Ge Zhang, Xinyi Zhang, Xiying Zhao, Zhenwei Zhu, Hongseok Namkoong, Wenhao Huang, and Yuwen Tang. Finsearchcomp: Towards a realistic, expert-level evaluation of financial search and reasoning. *CoRR*, abs/2509.13160, 2025. doi: 10.48550/ARXIV.2509.13160. URL https://doi.org/10.48550/arXiv.2509.13160.

[224] Mengkang Hu, Tianxing Chen, Qiguang Chen, Yao Mu, Wenqi Shao, and Ping Luo. Hiagent: Hierarchical working memory management for solving long-horizon agent tasks with large language model. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar, editors, *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, pages 32779–32798. Association for Computational Linguistics, 2025. doi: 10.18653/V1/2025.ACL-LONG.1575. URL https://doi.org/10.18653/v1/2025.acl-long.1575.

[225] Shengran Hu, Cong Lu, and Jeff Clune. Automated design of agentic systems. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=t9U3LW7JVX.

[226] Xueyu Hu, Tao Xiong, Biao Yi, Zishu Wei, Ruixuan Xiao, Yurun Chen, Jiasheng Ye, Meiling Tao, Xiangxin Zhou, Ziyu Zhao, Yuhuai Li, Shengze Xu, Shenzhi Wang, Xinchen Xu, Shuofei Qiao, Zhaokai Wang, Kun Kuang, Tieyong Zeng, Liang Wang, Jiwei Li, Yuchen Eleanor Jiang, Wangchunshu Zhou, Guoyin Wang, Keting Yin, Zhou Zhao, Hongxia Yang, Fan Wu, Shengyu Zhang, and Fei Wu. OS agents: A survey on mllm-based agents for computer, phone and browser use. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar, editors, *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, pages 7436–7465. Association for Computational Linguistics, 2025. doi: 10.18653/V1/2025.ACL-LONG.369. URL https://doi.org/10.18653/v1/2025.acl-long.369.

[227] Yusong Hu, Runmin Ma, Yue Fan, Jinxin Shi, Zongsheng Cao, Yuhao Zhou, Jiakang Yuan, Shuaiyu Zhang, Shiyang Feng, Xiangchao Yan, Shufei Zhang, Wenlong Zhang, Lei Bai, and Bo Zhang. Flowsearch: Advancing deep research with dynamic structured knowledge flow. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens, editors, *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2026, San Diego, California, United States, July 2-7, 2026*, pages 21212–21245. Association for Computational Linguistics, 2026. URL https://aclanthology.org/2026.acl-long.971/.

[228] Yuyang Hu, Shichun Liu, Yanwei Yue, Guibin Zhang, Boyang Liu, Fangyi Zhu, Jiahang Lin, Honglin Guo, Shihan Dou, Zhiheng Xi, Senjie Jin, Jiejun Tan, Yanbin Yin, Jiongnan Liu, Zeyu Zhang, Zhongxiang Sun, Yutao Zhu, Hao Sun, Boci Peng, Zhenrong Cheng, Xuanbo Fan, Jiaxin Guo, Xinlei Yu, Zhenhong Zhou, Zewen Hu, Jiahao Huo, Junhao Wang, Yuwei Niu, Yu Wang, Zhenfei Yin, Xiaobin Hu, Yue Liao, Qiankun Li, Kun Wang, Wangchunshu Zhou, Yixin Liu, Dawei Cheng, Qi Zhang, Tao Gui, Shirui Pan, Yan Zhang, Philip Torr, Zhicheng Dou, Ji-Rong Wen, Xuanjing Huang, Yu-Gang Jiang, and Shuicheng Yan. Memory in the age of AI agents. *CoRR*, abs/2512.13564, 2025. doi: 10.48550/ARXIV.2512.13564. URL https://doi.org/10.48550/arXiv.2512.13564.

[229] Yuyang Hu, Jiongnan Liu, Jiejun Tan, Yutao Zhu, and Zhicheng Dou. Memory matters more: Event-centric memory as a logic map for agent searching and reasoning. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens, editors, *Findings of the Association for Computational Linguistics, ACL 2026, San Diego, California, United States, July 2-7, 2026*, pages 22389–22407. Association for Computational Linguistics, 2026. URL https://aclanthology.org/2026.findings-acl.1123/.

[230] Yuyang Hu, Hongjin Qian, Shuting Wang, Jiongnan Liu, Tong Zhao, Xiaoxi Li, Zheng Liu, and Zhicheng Dou. Agentfugue: Agent scaling for long-horizon tasks through collective reasoning. *CoRR*, abs/2605.24486, 2026. doi: 10.48550/ARXIV.2605.24486. URL https://doi.org/10.48550/arXiv.2605.24486.

[231] Yuyang Hu, Hongjin Qian, Shuting Wang, Jiongnan Liu, Ziliang Zhao, Jiejun Tan, Zheng Liu, and Zhicheng Dou. SAM: state-adaptive memory for long-horizon reasoning agent. *CoRR*, abs/2605.24468, 2026. doi: 10.48550/ARXIV.2605.24468. URL https://doi.org/10.48550/arXiv.2605.24468.

[232] Zican Hu, Wei Liu, Xiaoye Qu, Xiangyu Yue, Chunlin Chen, Zhi Wang, and Yu Cheng. Divide and conquer: Grounding llms as efficient decision-making agents via offline hierarchical reinforcement learning. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu, editors, *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*, volume 267 of *Proceedings of Machine Learning Research*. PMLR / OpenReview.net, 2025. URL https://proceedings.mlr.press/v267/hu25q.html.

[233] Chengsong Huang, Wenhao Yu, Xiaoyang Wang, Hongming Zhang, Zongxia Li, Ruosen Li, Jiaxin Huang, Haitao Mi, and Dong Yu. R-zero: Self-evolving reasoning LLM from zero data. *CoRR*, abs/2508.05004, 2025. doi: 10.48550/ARXIV.2508.05004. URL https://doi.org/10.48550/arXiv.2508.05004.

[234] Jie Huang, Xinyun Chen, Swaroop Mishra, Huaixiu Steven Zheng, Adams Wei Yu, Xinying Song, and Denny Zhou. Large language models cannot self-correct reasoning yet. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=IkmD3fKBPQ.

[235] Wei-Chieh Huang, Weizhi Zhang, Yueqing Liang, Yuanchen Bei, Yankai Chen, Tao Feng, Xinyu Pan, Zhen Tan, Yu Wang, Tianxin Wei, Shanglin Wu, Ruiyao Xu, Liangwei Yang, Rui Yang, Wooseong Yang, Chin-Yuan Yeh, Hanrong Zhang, Haozhen Zhang, Siqi Zhu, Henry Peng Zou, Wanjia Zhao, Song Wang, Wujiang Xu, Zixuan Ke, Zheng Hui, Dawei Li, Yaozu Wu, Langzhou He, Chen Wang, Xiongxiao Xu, Baixiang Huang, Juntao Tan, Shelby Heinecke, Huan Wang, Caiming Xiong, Ahmed A. Metwally, Jun Yan, Chen-Yu Lee, Hanqing Zeng,

Yinglong Xia, Xiaokai Wei, Ali Payani, Yu Wang, Haitong Ma, Wenya Wang, Chenguang Wang, Yu Zhang, Xin Wang, Yongfeng Zhang, Jiaxuan You, Hanghang Tong, Xiao Luo, Xue Liu, Yizhou Sun, Wei Wang, Julian J. McAuley, James Zou, Jiawei Han, Philip S. Yu, and Kai Shu. Rethinking memory mechanisms of foundation agents in the second half: A survey. *CoRR*, abs/2602.06052, 2026. doi: 10.48550/ARXIV.2602.06052. URL https://doi.org/10.48550/arXiv.2602.06052.

[236] Wenlong Huang, Fei Xia, Ted Xiao, Harris Chan, Jacky Liang, Pete Florence, Andy Zeng, Jonathan Tompson, Igor Mordatch, Yevgen Chebotar, Pierre Sermanet, Tomas Jackson, Noah Brown, Linda Luu, Sergey Levine, Karol Hausman, and Brian Ichter. Inner monologue: Embodied reasoning through planning with language models. In Karen Liu, Dana Kulic, and Jeffrey Ichnowski, editors, *Conference on Robot Learning, CoRL 2022, 14-18 December 2022, Auckland, New Zealand*, volume 205 of *Proceedings of Machine Learning Research*, pages 1769–1782. PMLR, 2022. URL https://proceedings.mlr.press/v205/huang23c.html.

[237] Xu Huang, Junwu Chen, Yuxing Fei, Zhuohan Li, Philippe Schwaller, and Gerbrand Ceder. CASCADE: cumulative agentic skill creation through autonomous development and evolution. *CoRR*, abs/2512.23880, 2025. doi: 10.48550/ARXIV.2512.23880. URL https://doi.org/10.48550/arXiv.2512.23880.

[238] Yuchen Huang, Sijia Li, Minghao Liu, Wei Liu, Shijue Huang, Zhiyuan Fan, Hou Pong Chan, and Yi R. Fung. Environment scaling for interactive agentic experience collection: A survey. *CoRR*, abs/2511.09586, 2025. doi: 10.48550/ARXIV.2511.09586. URL https://doi.org/10.48550/arXiv.2511.09586.

[239] Yuxuan Huang, Yihang Chen, Haozheng Zhang, Kang Li, Meng Fang, Linyi Yang, Xiaoguang Li, Lifeng Shang, Songcen Xu, Jianye Hao, Kun Shao, and Jun Wang. Deep research agents: A systematic examination and roadmap. *CoRR*, abs/2506.18096, 2025. doi: 10.48550/ARXIV.2506.18096. URL https://doi.org/10.48550/arXiv.2506.18096.

[240] Zenan Huang, Yihong Zhuang, Guoshan Lu, Zeyu Qin, Haokai Xu, Tianyu Zhao, Ru Peng, Jiaqi Hu, Zhanming Shen, Xiaomeng Hu, Xijun Gu, Peiyi Tu, Jiaxin Liu, Wenyu Chen, Yuzhuo Fu, Zhiting Fan, Yanmei Gu, Yuanyuan Wang, Zhengkai Yang, Jianguo Li, and Junbo Zhao. Reinforcement learning with rubric anchors. *CoRR*, abs/2508.12790, 2025. doi: 10.48550/ARXIV.2508.12790. URL https://doi.org/10.48550/arXiv.2508.12790.

[241] Ziyang Huang, Haolin Ren, Xiaowei Yuan, Jiawei Wang, Zhongtao Jiang, Kun Xu, Shizhu He, Jun Zhao, and Kang Liu. Wideseek: Advancing wide research via multi-agent scaling. *CoRR*, abs/2602.02636, 2026. doi: 10.48550/ARXIV.2602.02636. URL https://doi.org/10.48550/arXiv.2602.02636.

[242] Jonas Hübotter, Frederike Lübeck, Lejs Behric, Anton Baumann, Marco Bagatella, Daniel Marta, Ido Hakimi, Idan Shenfeld, Thomas Kleine Buening, Carlos Guestrin, and Andreas Krause. Reinforcement learning via self-distillation. *CoRR*, abs/2601.20802, 2026. doi: 10.48550/ARXIV.2601.20802. URL https://doi.org/10.48550/arXiv.2601.20802.

[243] Brian Ichter, Anthony Brohan, Yevgen Chebotar, Chelsea Finn, Karol Hausman, Alexander Herzog, Daniel Ho, Julian Ibarz, Alex Irpan, Eric Jang, Ryan Julian, Dmitry Kalashnikov, Sergey Levine, Yao Lu, Carolina Parada, Kanishka Rao, Pierre Sermanet, Alexander Toshev, Vincent Vanhoucke, Fei Xia, Ted Xiao, Peng Xu, Mengyuan Yan, Noah Brown, Michael Ahn, Omar Cortes, Nicolas Sievers, Clayton Tan, Sichun Xu, Diego Reyes, Jarek Rettinghouse, Jornell Quiambao, Peter Pastor, Linda Luu, Kuang-Huei Lee, Yuheng Kuang, Sally Jesmonth, Nikhil J. Joshi, Kyle Jeffrey, Rosario Jauregui Ruano, Jasmine Hsu, Keerthana Gopalakrishnan, Byron David, Andy Zeng, and Chuyuan Kelly Fu. Do as I can, not as I say: Grounding language in robotic affordances. In Karen Liu, Dana Kulic, and Jeffrey Ichnowski, editors, *Conference on Robot Learning, CoRL 2022, 14-18 December 2022, Auckland, New Zealand*, volume 205 of *Proceedings of Machine Learning Research*, pages 287–318. PMLR, 2022. URL https://proceedings.mlr.press/v205/ichter23a.html.

[244] Hakan Inan, Kartikeya Upasani, Jianfeng Chi, Rashi Rungta, Krithika Iyer, Yuning Mao, Michael Tontchev, Qing Hu, Brian Fuller, Davide Testuggine, and Madian Khabsa. Llama guard: Llm-based input-output safeguard for human-ai conversations. *CoRR*, abs/2312.06674, 2023. doi: 10.48550/ARXIV.2312.06674. URL https://doi.org/10.48550/arXiv.2312.06674.

[245] Physical Intelligence, Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Manuel Y. Galliker, Dibya Ghosh, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Devin LeBlanc, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Allen Z. Ren, Lucy Xiaoyang Shi, Laura Smith, Jost Tobias Springenberg, Kyle Stachowicz, James Tanner, Quan Vuong, Homer Walke, Anna Walling, Haohuan Wang, Lili Yu, and Ury Zhilinsky. $\pi_{0.5}$: a vision-language-action model with open-world generalization. *CoRR*, abs/2504.16054, 2025. doi: 10.48550/ARXIV.2504.16054. URL https://doi.org/10.48550/arXiv.2504.16054.

[246] Gautier Izacard and Edouard Grave. Leveraging passage retrieval with generative models for open domain question answering. In Paola Merlo, Jörg Tiedemann, and Reut Tsarfaty, editors, *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, EACL 2021, Online, April 19 - 23, 2021*, pages 874–880. Association for Computational Linguistics, 2021. doi: 10.18653/V1/2021.EACL-MAIN.74. URL https://doi.org/10.18653/v1/2021.eacl-main.74.

[247] Gautier Izacard, Patrick Lewis, Maria Lomeli, Lucas Hosseini, Fabio Petroni, Timo Schick, Jane Dwivedi-Yu, Armand Joulin, Sebastian Riedel, and Edouard Grave. Atlas: Few-shot learning with retrieval augmented language models. *J. Mach. Learn. Res.*, 24:251:1–251:43, 2023. URL https://jmlr.org/papers/v24/23-0037.html.

[248] Naman Jain, Manish Shetty, Tianjun Zhang, King Han, Koushik Sen, and Ion Stoica. R2E: turning any github repository into a programming agent environment. In Ruslan Salakhutdinov, Zico Kolter, Katherine A. Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp, editors, *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*, volume 235 of *Proceedings of Machine Learning Research*, pages 21196–21224. PMLR / OpenReview.net, 2024. URL https://proceedings.mlr.press/v235/jain24c.html.

[249] Joel Jang, Seonghyeon Ye, Zongyu Lin, Jiannan Xiang, Johan Bjorck, Yu Fang, Fengyuan Hu, Spencer Huang, Kaushil Kundalia, Yen-Chen Lin, Loic Magne, Ajay Mandlekar, Avnish Narayan, You Liang Tan, Guanzhi Wang, Jing Wang, Qi Wang, Yinzhen Xu, Xiaohui Zeng, Kaiyuan Zheng, Ruijie Zheng, Ming-Yu Liu, Luke Zettlemoyer, Dieter Fox, Jan Kautz, Scott Reed, Yuke Zhu, and Linxi Fan. Dreamgen: Unlocking generalization in robot learning through video world models, 2025. URL https://arxiv.org/abs/2505.12705.

[250] Yuxiang Ji, Ziyu Ma, Yong Wang, Guanhua Chen, Xiangxiang Chu, and Liaoni Wu. Tree search for LLM agent reinforcement learning. *CoRR*, abs/2509.21240, 2025. doi: 10.48550/ARXIV.2509.21240. URL https://doi.org/10.48550/arXiv.2509.21240.

[251] Hongrui Jia, Jitong Liao, Xi Zhang, Haiyang Xu, Tianbao Xie, Chaoya Jiang, Ming Yan, Si Liu, Wei Ye, and Fei Huang. Osworld-mcp: Benchmarking MCP tool invocation in computer-use agents. *CoRR*, abs/2510.24563, 2025. doi: 10.48550/ARXIV.2510.24563. URL https://doi.org/10.48550/arXiv.2510.24563.

[252] Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de Las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mistral 7b. *CoRR*, abs/2310.06825, 2023. doi: 10.48550/ARXIV.2310.06825. URL https://doi.org/10.48550/arXiv.2310.06825.

[253] Dongming Jiang, Yi Li, Songtao Wei, Jinxin Yang, Ayushi Kishore, Alysa Zhao, Dingyi Kang, Xu Hu, Feng Chen, Qiannan Li, and Bingzhe Li. Anatomy of agentic memory: Taxonomy and empirical analysis of evaluation and system limitations. *CoRR*, abs/2602.19320, 2026. doi: 10.48550/ARXIV.2602.19320. URL https://doi.org/10.48550/arXiv.2602.19320.

[254] Huiqiang Jiang, Qianhui Wu, Chin-Yew Lin, Yuqing Yang, and Lili Qiu. Llmlingua: Compressing prompts for accelerated inference of large language models. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 13358–13376. Association for Computational Linguistics, 2023. doi: 10.18653/V1/2023.EMNLP-MAIN.825. URL https://doi.org/10.18653/v1/2023.emnlp-main.825.

[255] Kaixun Jiang, Yuzheng Wang, Junjie Zhou, Pandeng Li, Zhihang Liu, Chen-Wei Xie, Zhaoyu Chen, Yun Zheng, and Wenqiang Zhang. Genagent: Scaling text-to-image generation via agentic multimodal reasoning. *CoRR*, abs/2601.18543, 2026. doi: 10.48550/ARXIV.2601.18543. URL https://doi.org/10.48550/arXiv.2601.18543.

[256] 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. *CoRR*, abs/2503.00223, 2025. doi: 10.48550/ARXIV.2503.00223. URL https://doi.org/10.48550/arXiv.2503.00223.

[257] Yixing Jiang, Kameron C. Black, Gloria Geng, Danny Park, James Zou, Andrew Y. Ng, and Jonathan H. Chen. Medagentbench: A realistic virtual ehr environment to benchmark medical llm agents, 2025. URL https://arxiv.org/abs/2501.14654.

[258] 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 Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP*

2023, Singapore, December 6-10, 2023, pages 7969–7992. Association for Computational Linguistics, 2023. doi: 10.18653/V1/2023.EMNLP-MAIN.495. URL https://doi.org/10.18653/v1/2023.emnlp-main.495.

[259] Ziyan Jiang, Li An, Yujian Liu, Jiabao Ji, Qiucheng Wu, Jacob Andreas, Yang Zhang, and Shiyu Chang. VISUALSKILL: multimodal skills for computer-use agents. *CoRR*, abs/2606.18448, 2026. doi: 10.48550/ARXIV.2606.18448. URL https://doi.org/10.48550/arXiv.2606.18448.

[260] Carlos E. Jimenez, John Yang, Alexander Wettig, Shunyu Yao, Kexin Pei, Ofir Press, and Karthik R. Narasimhan. Swe-bench: Can language models resolve real-world github issues? In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=VTF8yNQM66.

[261] Bowen Jin, Hansi Zeng, Zhenrui Yue, Dong Wang, Hamed Zamani, and Jiawei Han. Search-r1: Training llms to reason and leverage search engines with reinforcement learning. *CoRR*, abs/2503.09516, 2025. doi: 10.48550/ARXIV.2503.09516. URL https://doi.org/10.48550/arXiv.2503.09516.

[262] Jiajie Jin, Yuyang Hu, Kai Qiu, Qi Dai, Chong Luo, Guanting Dong, Xiaoxi Li, Tong Zhao, Xiaolong Ma, Gongrui Zhang, Zhirong Wu, Bei Liu, Zhengyuan Yang, Linjie Li, Lijuan Wang, Hongjin Qian, Yutao Zhu, and Zhicheng Dou. Toward generalist autonomous research via hypothesis-tree refinement. *CoRR*, abs/2606.11926, 2026. doi: 10.48550/ARXIV.2606.11926. URL https://doi.org/10.48550/arXiv.2606.11926.

[263] Jiajie Jin, Yuyao Zhang, Yimeng Xu, Hongjin Qian, Yutao Zhu, and Zhicheng Dou. Finsight: Towards real-world financial deep research. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens, editors, *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2026, San Diego, California, United States, July 2-7, 2026*, pages 5868–5894. Association for Computational Linguistics, 2026. URL https://aclanthology.org/2026.acl-long.265/.

[264] Leslie Pack Kaelbling, Michael L. Littman, and Anthony R. Cassandra. Planning and acting in partially observable stochastic domains. *Artif. Intell.*, 101(1-2):99–134, 1998. doi: 10.1016/S0004-3702(98)00023-X. URL https://doi.org/10.1016/S0004-3702(98)00023-X.

[265] Adharsh Kamath, Sishen Zhang, Calvin Xu, Shubham Ugare, Gagandeep Singh, and Sasa Misailovic. Enforcing temporal constraints for LLM agents. *CoRR*, abs/2512.23738, 2025. doi: 10.48550/ARXIV.2512.23738. URL https://doi.org/10.48550/arXiv.2512.23738.

[266] Nikhil Kandpal, Haikang Deng, Adam Roberts, Eric Wallace, and Colin Raffel. Large language models struggle to learn long-tail knowledge. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett, editors, *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pages 15696–15707. PMLR, 2023. URL https://proceedings.mlr.press/v202/kandpal23a.html.

[267] Jiazheng Kang, Mingming Ji, Zhe Zhao, and Ting Bai. Memory OS of AI agent. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng, editors, *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, EMNLP 2025, Suzhou, China, November 4-9, 2025*, pages 25961–25970. Association for Computational Linguistics, 2025. doi: 10.18653/V1/2025.EMNLP-MAIN.1318. URL https://doi.org/10.18653/v1/2025.emnlp-main.1318.

[268] Minki Kang, Wei-Ning Chen, Dongge Han, Huseyin A. Inan, Lukas Wutschitz, Yanzhi Chen, Robert Sim, and Saravan Rajmohan. ACON: optimizing context compression for long-horizon LLM agents. *CoRR*, abs/2510.00615, 2025. doi: 10.48550/ARXIV.2510.00615. URL https://doi.org/10.48550/arXiv.2510.00615.

[269] Minki Kang, Jongwon Jeong, Seanie Lee, Jaewoong Cho, and Sung Ju Hwang. Distilling LLM agent into small models with retrieval and code tools. In Danielle Belgrave, Cheng Zhang, Laura N. Montoya, Hsuan-Tien Lin, Razvan Pascanu, Piotr Koniusz, Marzyeh Ghassemi, Nancy Chen, Iván Vladimir Meza Ruíz, and Arturo Loaiza-Bonilla, editors, *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2025, NeurIPS 2025, San Diego, CA, USA, December 2-7, 2025 / Mexico City, Mexico, November 30 - December 5, 2025*, 2025. URL http://papers.nips.cc/paper_files/paper/2025/hash/99263f9bf46874e2b8b7f104b5063864-Abstract-Conference.html.

[270] Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 6769–6781. Association for Computational Linguistics, 2020. doi: 10.18653/V1/2020.EMNLP-MAIN.550. URL https://doi.org/10.18653/v1/2020.emnlp-main.550.

[271] Tomer Keren, Nitay Calderon, Asaf Yehudai, Yotam Perlitz, Michal Shmueli-Scheuer, and Roi Reichert. A matter of TASTE: improving coverage and difficulty of agent benchmarks. *CoRR*, abs/2605.28556, 2026. doi: 10.48550/ARXIV.2605.28556. URL https://doi.org/10.48550/arXiv.2605.28556.

[272] Muhammad Khalifa, Rishabh Agarwal, Lajanugen Logeswaran, Jaekyeom Kim, Hao Peng, Moontae Lee, Honglak Lee, and Lu Wang. Process reward models that think. *Trans. Mach. Learn. Res.*, 2026, 2026. URL https://openreview.net/forum?id=FPVCb0WMuN.

[273] Urvashi Khandelwal, Omer Levy, Dan Jurafsky, Luke Zettlemoyer, and Mike Lewis. Generalization through memorization: Nearest neighbor language models. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020. URL https://openreview.net/forum?id=HklBjCEKvH.

[274] Gauri Kholkar and Ratinder Ahuja. Policy-as-prompt: Turning ai governance rules into guardrails for ai agents, 2025. URL https://arxiv.org/abs/2509.23994.

[275] Tushar Khot, Harsh Trivedi, Matthew Finlayson, Yao Fu, Kyle Richardson, Peter Clark, and Ashish Sabharwal. Decomposed prompting: A modular approach for solving complex tasks. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023. URL https://openreview.net/forum?id=_nGgzQjzaRy.

[276] Jeonghye Kim, Sojeong Rhee, Minbeom Kim, Dohyung Kim, Sangmook Lee, Youngchul Sung, and Kyomin Jung. Reflact: World-grounded decision making in LLM agents via goal-state reflection. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng, editors, *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, EMNLP 2025, Suzhou, China, November 4-9, 2025*, pages 33433–33465. Association for Computational Linguistics, 2025. doi: 10.18653/V1/2025.EMNLP-MAIN.1697. URL https://doi.org/10.18653/v1/2025.emnlp-main.1697.

[277] Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Paul Foster, Pannag R. Sanketi, Quan Vuong, Thomas Kollar, Benjamin Burchfiel, Russ Tedrake, Dorsa Sadigh, Sergey Levine, Percy Liang, and Chelsea Finn. Openvla: An open-source vision-language-action model. In Pulkit Agrawal, Oliver Kroemer, and Wolfram Burgard, editors, *Conference on Robot Learning, 6-9 November 2024, Munich, Germany*, volume 270 of *Proceedings of Machine Learning Research*, pages 2679–2713. PMLR, 2024. URL https://proceedings.mlr.press/v270/kim25c.html.

[278] Seungone Kim, Jamin Shin, Yejin Choi, Joel Jang, Shayne Longpre, Hwaran Lee, Sangdoo Yun, Seongjin Shin, Sungdong Kim, James Thorne, and Minjoon Seo. Prometheus: Inducing fine-grained evaluation capability in language models. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=8euJaTveKw.

[279] Sunghwan Kim, Junhee Cho, Beong-woo Kwak, Taeyoon Kwon, Liang Wang, Nan Yang, Xingxing Zhang, Furu Wei, and Jinyoung Yeo. On training large language models for long-horizon tasks: An empirical study of horizon length. *CoRR*, abs/2605.02572, 2026. doi: 10.48550/ARXIV.2605.02572. URL https://doi.org/10.48550/arXiv.2605.02572.

[280] Jing Yu Koh, Stephen Marcus McAleer, Daniel Fried, and Ruslan Salakhutdinov. Tree search for language model agents. *Trans. Mach. Learn. Res.*, 2025, 2025. URL https://openreview.net/forum?id=QF0N3x2XVm.

[281] Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners. In Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022. URL http://papers.nips.cc/paper_files/paper/2022/hash/8bb0d291acd4acf06ef112099c16f326-Abstract-Conference.html.

[282] Quyu Kong, Xu Zhang, Zhenyu Yang, Nolan Gao, Chen Liu, Panrong Tong, Chenglin Cai, Hanzhang Zhou, Jianan Zhang, Liangyu Chen, Zhidan Liu, Steven Hoi, and Yue Wang. Mobileworld: Benchmarking autonomous mobile agents in agent-user interactive and mcp-augmented environments. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens, editors, *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2026, San Diego, California, United States, July 2-7, 2026*, pages 6142–6167. Association for Computational Linguistics, 2026. URL https://aclanthology.org/2026.acl-long.278/.

[283] Jacob Krantz, Erik Wijmans, Arjun Majumdar, Dhruv Batra, and Stefan Lee. Beyond the nav-graph: Vision-and-language navigation in continuous environments. In *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part XXVIII*, pages 104–120, 2020. doi: 10.1007/978-3-030-58604-1_7. URL https://doi.org/10.1007/978-3-030-58604-1_7.

[284] 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 Luis Chiruzzo, Alan Ritter, and Lu Wang, editors, *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL 2025 - Volume 1: Long Papers, Albuquerque, New Mexico, USA, April 29 - May 4, 2025*, pages 4745–4759. Association for Computational Linguistics, 2025. doi: 10.18653/V1/2025.NAACL-LONG.243. URL https://doi.org/10.18653/v1/2025.naacl-long.243.

[285] Tarun Kumar, Aalap Tripathy, Gayathri Saranathan, Martin Foltin, Suparna Bhattacharya, Scott Hinchley, Donald M. Bahls, David Brookshire, Larry Kaplan, and Robert W. Wisniewski. Infrastructuresentinel: Policy enforced guardrails for secure mcp-driven infrastructure agents. In Sven Koenig, Chad Jenkins, and Matthew E. Taylor, editors, *Fortieth AAAI Conference on Artificial Intelligence, Thirty-Eighth Conference on Innovative Applications of Artificial Intelligence, Sixteenth Symposium on Educational Advances in Artificial Intelligence, AAAI 2026, Singapore, January 20-27, 2026*, pages 40295–40301. AAAI Press, 2026. doi: 10.1609/AAAI.V40I47.41468. URL https://doi.org/10.1609/aaai.v40i47.41468.

[286] Thomas Kwa, Ben West, Joel Becker, Amy Deng, Katharyn Garcia, Max Hasin, Sami Jawhar, Megan Kinniment, Nate Rush, Sydney von Arx, Ryan Bloom, Thomas Broadley, Haoxing Du, Brian Goodrich, Nikola Jurkovic, Luke Harold Miles, Seraphina Nix, Tao Lin, Neev Parikh, David Rein, Lucas Jun Koba Sato, Hjalmar Wijk, Daniel M. Ziegler, Elizabeth Barnes, and Lawrence Chan. Measuring AI ability to complete long software tasks. In Danielle Belgrave, Cheng Zhang, Laura N. Montoya, Hsuan-Tien Lin, Razvan Pascanu, Piotr Koniusz, Marzyeh Ghassemi, Nancy Chen, Iván Vladimir Meza Ruíz, and Arturo Loaiza-Bonilla, editors, *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2025, NeurIPS 2025, San Diego, CA, USA, December 2-7, 2025 / Mexico City, Mexico, November 30 - December 5, 2025*, 2025. URL http://papers.nips.cc/paper_files/paper/2025/hash/85069585133c4c168c865e65d72e9775-Abstract-Conference.html.

[287] Shanghai Artificial Intelligence Laboratory. Agentdog: A diagnostic guardrail framework for AI agent safety and security. *CoRR*, abs/2601.18491, 2026. doi: 10.48550/ARXIV.2601.18491. URL https://doi.org/10.48550/arXiv.2601.18491.

[288] Shanghai Artificial Intelligence Laboratory. Agentdog 1.5: A lightweight and scalable alignment framework for AI agent safety and security. *CoRR*, abs/2605.29801, 2026. doi: 10.48550/ARXIV.2605.29801. URL https://doi.org/10.48550/arXiv.2605.29801.

[289] Tian Lan, Felix Henry, Bin Zhu, Qianghuai Jia, Junyang Ren, Qihang Pu, Haijun Li, Longyue Wang, Zhao Xu, and Weihua Luo. Table-as-search: Agentic information seeking is table completion. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens, editors, *Findings of the Association for Computational Linguistics, ACL 2026, San Diego, California, United States, July 2-7, 2026*, pages 15088–15107. Association for Computational Linguistics, 2026. URL https://aclanthology.org/2026.findings-acl.743/.

[290] Jordy Van Landeghem, Rafal Powalski, Rubèn Tito, Dawid Jurkiewicz, Matthew B. Blaschko, Lukasz Borchmann, Mickaël Coustaty, Sien Moens, Michal Pietruszka, Bertrand Anckaert, Tomasz Stanislawek, Pawel Józiak, and Ernest Valveny. Document understanding dataset and evaluation (DUDE). In *IEEE/CVF International Conference on Computer Vision, ICCV 2023, Paris, France, October 1-6, 2023*, pages 19471–19483. IEEE, 2023. doi: 10.1109/ICCV51070.2023.01789. URL https://doi.org/10.1109/ICCV51070.2023.01789.

[291] LangChain. langgraph: Build resilient agents., 2024. URL https://github.com/langchain-ai/langgraph.

[292] LangChain. Open deep research, 2025. URL https://www.langchain.com/blog/open-deep-research.

[293] LangChain. Improving deep agents with harness engineering, 2026. URL https://www.langchain.com/blog/improving-deep-agents-with-harness-engineering.

[294] LangChain. How middleware lets you customize your agent harness, 2026. URL https://www.langchain.com/blog/how-middleware-lets-you-customize-your-agent-harness.

[295] Ka Yiu Lee, Yuxuan Huang, Zhiyuan He, Huichi Zhou, Weilin Luo, Kun Shao, Meng Fang, and Jun Wang. Infoseeker: A scalable hierarchical parallel agent framework for web information seeking. *CoRR*, abs/2604.02971, 2026. doi: 10.48550/ARXIV.2604.02971. URL https://doi.org/10.48550/arXiv.2604.02971.

[296] Nicholas Lee, Lutfi Eren Erdogan, Chris Joseph John, Surya Krishnapillai, Michael W. Mahoney, Kurt Keutzer, and Amir Gholami. Agentic test-time scaling for webagents. *CoRR*, abs/2602.12276, 2026. doi: 10.48550/ARXIV.2602.12276. URL https://doi.org/10.48550/arXiv.2602.12276.

[297] Yoonho Lee, Roshen Nair, Qizheng Zhang, Kangwook Lee, Omar Khattab, and Chelsea Finn. Meta-harness: End-to-end optimization of model harnesses. *CoRR*, abs/2603.28052, 2026. doi: 10.48550/ARXIV.2603.28052. URL https://doi.org/10.48550/arXiv.2603.28052.

[298] Yoonsang Lee, Howard Yen, Xi Ye, and Danqi Chen. Agentic aggregation for parallel scaling of long-horizon agentic tasks. *CoRR*, abs/2604.11753, 2026. doi: 10.48550/ARXIV.2604.11753. URL https://doi.org/10.48550/arXiv.2604.11753.

[299] Fei Lei, Yibo Yang, Wenxiu Sun, and Dahua Lin. Mcpverse: An expansive, real-world benchmark for agentic tool use. *CoRR*, abs/2508.16260, 2025. doi: 10.48550/ARXIV.2508.16260. URL https://doi.org/10.48550/arXiv.2508.16260.

[300] Barak Lenz, Opher Lieber, Alan Arazi, Amir Bergman, Avshalom Manevich, Barak Peleg, Ben Aviram, Chen Almagor, Clara Fridman, Dan Padnos, Daniel Gissin, Daniel Jannai, Dor Muhlgay, Dor Zimberg, Edden M. Gerber, Elad Dolev, Eran Krakovsky, Erez Safahi, Erez Schwartz, Gal Cohen, and et al. Jamba: Hybrid transformer-mamba language models. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=JFPaD7lpBD.

[301] Dmitry Lepikhin, HyoukJoong Lee, Yuanzhong Xu, Dehao Chen, Orhan Firat, Yanping Huang, Maxim Krikun, Noam Shazeer, and Zhifeng Chen. Gshard: Scaling giant models with conditional computation and automatic sharding. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. URL https://openreview.net/forum?id=qrwe7XHTmYb.

[302] Yaniv Leviathan, Matan Kalman, and Yossi Matias. Fast inference from transformers via speculative decoding. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett, editors, *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pages 19274–19286. PMLR, 2023. URL https://proceedings.mlr.press/v202/leviathan23a.html.

[303] Ido Levy, Ben Wiesel, Sami Marreed, Alon Oved, Avi Yaeli, and Segev Shlomov. St-webagentbench: A benchmark for evaluating safety and trustworthiness in web agents. *CoRR*, abs/2410.06703, 2024. doi: 10.48550/ARXIV.2410.06703. URL https://doi.org/10.48550/arXiv.2410.06703.

[304] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive NLP tasks. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/6b493230205f780e1bc26945df7481e5-Abstract.html.

[305] Ao Li, Yuexiang Xie, Songze Li, Fugee Tsung, Bolin Ding, and Yaliang Li. Agent-oriented planning in multi-agent systems. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=EqcLAU6gyU.

[306] Baiqi Li, Zhiqiu Lin, Deepak Pathak, Jiayao Li, Yixin Fei, Kewen Wu, Tiffany Ling, Xide Xia, Pengchuan Zhang, Graham Neubig, and Deva Ramanan. Genai-bench: Evaluating and improving compositional text-to-visual generation. *CoRR*, abs/2406.13743, 2024. doi: 10.48550/ARXIV.2406.13743. URL https://doi.org/10.48550/arXiv.2406.13743.

[307] Baixuan Li, Bo Zhang, Dingchu Zhang, Fei Huang, Guangyu Li, Guoxin Chen, Huifeng Yin, Jialong Wu, Jingren Zhou, Kuan Li, Liangcai Su, Litu Ou, Liwen Zhang, Pengjun Xie, Rui Ye, Wenbiao Yin, Xinmiao Yu, Xinyu Wang, Xixi Wu, Xuanzhong Chen, Yida Zhao, Zhen Zhang, Zhengwei Tao, Zhongwang Zhang, Zile Qiao, Chenxi Wang, Donglei Yu, Gang Fu, Haiyang Shen, Jiayin Yang, Jun Lin, Junkai Zhang, Kui Zeng, Li Yang, Hailong Yin, Maojia Song, Ming Yan, Peng Xia, Qian Xiao, Rui Min, Ruixue Ding, Runnan Fang, Shaowei Chen, Shen Huang, Shihang Wang, Shihao Cai, Weizhou Shen, Xiaobin Wang, Xin Guan, Xinyu Geng, Yingcheng Shi, Yuning Wu, Zhuo Chen, Zijian Li, and Yong Jiang. Tongyi deepresearch technical report. *CoRR*, abs/2510.24701, 2025. doi: 10.48550/ARXIV.2510.24701. URL https://doi.org/10.48550/arXiv.2510.24701.

[308] Caorui Li, Yu Chen, Yiyan Ji, Jin Xu, Zhenyu Cui, Shihao Li, Yuanxing Zhang, Jiafu Tang, Zhenghao Song, Dingling Zhang, Ying He, Haoxiang Liu, Yuxuan Wang, Qiufeng Wang, Zhenhe Wu, Jiehui Luo, Zhiyu Pan, Weihao Xie, Chenchen Zhang, Zhaohui Wang, Jiayi Tian, Yanghai Wang, Zhe Cao, Minxin Dai, Ke Wang, Runzhe Wen, Yinghao Ma, Yaning Pan, Sungkyun Chang, Termeh Taheri, Haiwen Xia, Christos Plachouras, Emmanouil Benetos, Yizhi Li, Ge Zhang, Jian Yang, Tianhao Peng, Zili Wang, Minghao Liu, Junran Peng, Zhaoxiang Zhang, and Jiaheng Liu. Omnivideobench: Towards audio-visual understanding evaluation for omni mllms. *CoRR*, abs/2510.10689, 2025. doi: 10.48550/ARXIV.2510.10689. URL https://doi.org/10.48550/arXiv.2510.10689.

[309] Changhao Li, Rushi Qiang, Jiawei Huang, Chenxiao Gao, Chao Zhang, Niao He, and Bo Dai. Revisiting dagger in the era of llm-agents. *CoRR*, abs/2605.12913, 2026. doi: 10.48550/ARXIV.2605.12913. URL https://doi.org/10.48550/arXiv.2605.12913.

[310] Chengpeng Li, Mingfeng Xue, Zhenru Zhang, Jiaxi Yang, Beichen Zhang, Bowen Yu, Binyuan Hui, Junyang Lin, Xiang Wang, and Dayiheng Liu. START: self-taught reasoner with tools. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng, editors, *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, EMNLP 2025, Suzhou, China, November 4-9, 2025*, pages 13512–13553. Association for Computational Linguistics, 2025. doi: 10.18653/V1/2025.EMNLP-MAIN.683. URL https://doi.org/10.18653/v1/2025.emnlp-main.683.

[311] Dawei Li, Yuguang Yao, Zhen Tan, Huan Liu, and Ruocheng Guo. Toolprmbench: Evaluating and advancing process reward models for tool-using agents. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens, editors, *Findings of the Association for Computational Linguistics, ACL 2026, San Diego, California, United States, July 2-7, 2026*, pages 12378–12391. Association for Computational Linguistics, 2026. URL https://aclanthology.org/2026.findings-acl.602/.

[312] Fengxiang Li, Han Zhang, Haoyang Huang, Jinghui Wang, Jinhua Hao, Kun Yuan, Mengtong Li, Minglei Zhang, Pengcheng Xu, Wenhao Zhuang, Yizhen Shao, Zongxian Feng, Can Tang, Chao Wang, Chengxiao Tong, Fan Yang, Gang Xiong, Haixuan Gao, Han Gao, Hao Wang, Haochen Liu, Hongliang Sun, Jiabao Li, Jingwen Chang, Jun Du, Junyi Peng, Leizhen Cui, Meimei Jing, Mingqi Wu, Shangpeng Yan, Shaotong Qi, Suzhe Xu, Wenxuan Zhao, Xianda Sun, Xuan Xie, Yanbo Wang, Yao Xia, Yinghan Cui, Yingpeng Chen, Yong Wang, Yuze Shi, Zhiwei Shen, Ziyu Wang, Ming Sun, Lin Ye, and Bin Chen. Kat-coder-v2 technical report. *CoRR*, abs/2603.27703, 2026. doi: 10.48550/ARXIV.2603.27703. URL https://doi.org/10.48550/arXiv.2603.27703.

[313] Gengsheng Li, Mao Zheng, Mingyang Song, Ruiqi Liu, Tianyu Yang, Jie Sun, Qiyong Zhong, Haiyun Guo, Junfeng Fang, Dan Zhang, and Jinqiao Wang. On-policy distillation with curriculum turn-level guidance for multi-turn agents. *CoRR*, abs/2606.15912, 2026. doi: 10.48550/ARXIV.2606.15912. URL https://doi.org/10.48550/arXiv.2606.15912.

[314] Guohao Li, Hasan Hammoud, Hani Itani, Dmitrii Khizbullin, and Bernard Ghanem. CAMEL: communicative agents for "mind" exploration of large language model society. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine, editors, *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/a3621ee907def47c1b952ade25c67698-Abstract-Conference.html.

[315] Haitao Li, Junjie Chen, Jingli Yang, Qingyao Ai, Wei Jia, Youfeng Liu, Kai Lin, Yueyue Wu, Guozhi Yuan, Yiran Hu, Wuyue Wang, Yiqun Liu, and Minlie Huang. Legalagentbench: Evaluating LLM agents in legal domain. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar, editors, *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, pages 2322–2344. Association for Computational Linguistics, 2025. doi: 10.18653/V1/2025.ACL-LONG.116. URL https://doi.org/10.18653/v1/2025.acl-long.116.

[316] Jeffrey Li, Alex Fang, Georgios Smyrnis, Maor Ivgi, Matt Jordan, Samir Yitzhak Gadre, Hritik Bansal, Etash Kumar Guha, Sedrick Scott Keh, Kushal Arora, Saurabh Garg, Rui Xin, Niklas Muennighoff, Reinhard Heckel, Jean Mercat, Mayee F. Chen, Suchin Gururangan, Mitchell Wortsman, Alon Albalak, Yonatan Bitton, Marianna Nezhurina, Amro Abbas, Cheng-Yu Hsieh, Dhruba Ghosh, Josh Gardner, Maciej Kilian, Hanlin Zhang, Rulin Shao, Sarah M. Pratt, Sunny Sanyal, Gabriel Ilharco, Giannis Daras, Kalyani Marathe, Aaron Gokaslan, Jieyu Zhang, Khyathi Raghavi Chandu, Thao Nguyen, Igor Vasiljevic, Sham M. Kakade, Shuran Song, Sujay Sanghavi, Fartash Faghri, Sewoong Oh, Luke Zettlemoyer, Kyle Lo, Alaaeldin El-Nouby, Hadi Pouransari, Alexander Toshev, Stephanie Wang, Dirk Groeneveld, Luca Soldaini, Pang Wei Koh, Jenia Jitsev, Thomas Kollar, Alex Dimakis, Yair Carmon, Achal Dave, Ludwig Schmidt, and Vaishaal Shankar. Datacomp-lm: In search of the next generation of training sets for language models. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang, editors, *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. URL http://papers.nips.cc/paper_files/paper/2024/hash/19e4ea30dded58259665db375885e412-Abstract-Datasets_and_Benchmarks_Track.html.

[317] Jia Li, Ge Li, Yunfei Zhao, Yongmin Li, Huanyu Liu, Hao Zhu, Lecheng Wang, Kaibo Liu, Zheng Fang, Lanshen Wang, Jiazheng Ding, Xuanming Zhang, Yuqi Zhu, Yihong Dong, Zhi Jin, Binhua Li, Fei Huang, Yongbin Li, Bin Gu, and Mengfei Yang. Deveval: A manually-annotated code generation benchmark aligned with real-world code repositories. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Findings of the*

Association for Computational Linguistics, ACL 2024, Bangkok, Thailand and virtual meeting, August 11-16, 2024, volume ACL 2024 of *Findings of ACL*, pages 3603–3614. Association for Computational Linguistics, 2024. doi: 10.18653/V1/2024.FINDINGS-ACL.214. URL https://doi.org/10.18653/v1/2024.findings-acl.214.

[318] Jiazheng Li, Yawei Wang, Qiaojing Yan, Yijun Tian, Zhichao Xu, Huan Song, Panpan Xu, and Lin Lee Cheong. SALT: step-level advantage assignment for long-horizon agents via trajectory graph. In Vera Demberg, Kentaro Inui, and Lluís Marquez, editors, *Findings of the Association for Computational Linguistics: EACL 2026, Rabat, Morocco, March 24-29, 2026*, Findings of ACL, pages 4709–4725. Association for Computational Linguistics, 2026. doi: 10.18653/V1/2026.FINDINGS-EACL.247. URL https://doi.org/10.18653/v1/2026.findings-eacl.247.

[319] Jierui Li, Hung Le, Yingbo Zhou, Caiming Xiong, Silvio Savarese, and Doyen Sahoo. Codetree: Agent-guided tree search for code generation with large language models. In Luis Chiruzzo, Alan Ritter, and Lu Wang, editors, *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL 2025 - Volume 1: Long Papers, Albuquerque, New Mexico, USA, April 29 - May 4, 2025*, pages 3711–3726. Association for Computational Linguistics, 2025. doi: 10.18653/V1/2025.NAACL-LONG.189. URL https://doi.org/10.18653/v1/2025.naacl-long.189.

[320] Jingyao Li, Jingyun Wang, Molin Tan, Haochen Wang, Cilin Yan, Likun Shi, Jiayin Cai, Xiaolong Jiang, and Yao Hu. Crossvid: A comprehensive benchmark for evaluating cross-video reasoning in multimodal large language models. In Sven Koenig, Chad Jenkins, and Matthew E. Taylor, editors, *Fortieth AAAI Conference on Artificial Intelligence, Thirty-Eighth Conference on Innovative Applications of Artificial Intelligence, Sixteenth Symposium on Educational Advances in Artificial Intelligence, AAAI 2026, Singapore, January 20-27, 2026*, pages 6244–6252. AAAI Press, 2026. doi: 10.1609/AAAI.V40I8.37550. URL https://doi.org/10.1609/aaai.v40i8.37550.

[321] Junbo Li, Peng Zhou, Rui Meng, Meet P. Vadera, Lihong Li, and Yang Li. Turn-ppo: Turn-level advantage estimation with PPO for improved multi-turn RL in agentic llms. In Vera Demberg, Kentaro Inui, and Lluís Marquez, editors, *Findings of the Association for Computational Linguistics: EACL 2026, Rabat, Morocco, March 24-29, 2026*, Findings of ACL, pages 6227–6243. Association for Computational Linguistics, 2026. doi: 10.18653/V1/2026.FINDINGS-EACL.328. URL https://doi.org/10.18653/v1/2026.findings-eacl.328.

[322] Junlong Li, Wenshuo Zhao, Jian Zhao, Weihao Zeng, Haoze Wu, Xiaochen Wang, Rui Ge, Yuxuan Cao, Yuzhen Huang, Wei Liu, Junteng Liu, Zhaochen Su, Yiyang Guo, Fan Zhou, Lueyang Zhang, Juan Michelini, Xingyao Wang, Xiang Yue, Shuyan Zhou, Graham Neubig, and Junxian He. The tool decathlon: Benchmarking language agents for diverse, realistic, and long-horizon task execution. *CoRR*, abs/2510.25726, 2025. doi: 10.48550/ARXIV.2510.25726. URL https://doi.org/10.48550/arXiv.2510.25726.

[323] Junyou Li, Qin Zhang, Yangbin Yu, Qiang Fu, and Deheng Ye. More agents is all you need. *Trans. Mach. Learn. Res.*, 2024, 2024. URL https://openreview.net/forum?id=bgzUSZ8aeg.

[324] Kaixin Li, Ziyang Meng, Hongzhan Lin, Ziyang Luo, Yuchen Tian, Jing Ma, Zhiyong Huang, and Tat-Seng Chua. Screenspot-pro: GUI grounding for professional high-resolution computer use. In Cathal Gurrin, Klaus Schoeffmann, Min Zhang, Luca Rossetto, Stevan Rudinac, Duc-Tien Dang-Nguyen, Wen-Huang Cheng, Phoebe Chen, and Jenny Benois-Pineau, editors, *Proceedings of the 33rd ACM International Conference on Multimedia, MM 2025, Dublin, Ireland, October 27-31, 2025*, pages 8778–8786. ACM, 2025. doi: 10.1145/3746027.3755688. URL https://doi.org/10.1145/3746027.3755688.

[325] Keyu Li, Junhao Shi, Yang Xiao, Mohan Jiang, Jie Sun, Yunze Wu, Dayuan Fu, Shijie Xia, Xiaojie Cai, Tianze Xu, Weiye Si, Wenjie Li, Dequan Wang, and Pengfei Liu. Agencybench: Benchmarking the frontiers of autonomous agents in 1m-token real-world contexts. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens, editors, *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2026, San Diego, California, United States, July 2-7, 2026*, pages 7422–7440. Association for Computational Linguistics, 2026. URL https://aclanthology.org/2026.acl-long.337/.

[326] Kuan Li, Zhongwang Zhang, Huifeng Yin, Rui Ye, Yida Zhao, Liwen Zhang, Litu Ou, Dingchu Zhang, Xixi Wu, Jialong Wu, Xinyu Wang, Zile Qiao, Zhen Zhang, Yong Jiang, Pengjun Xie, Fei Huang, and Jingren Zhou. Websailor-v2: Bridging the chasm to proprietary agents via synthetic data and scalable reinforcement learning. *CoRR*, abs/2509.13305, 2025. doi: 10.48550/ARXIV.2509.13305. URL https://doi.org/10.48550/arXiv.2509.13305.

[327] Kuan Li, Zhongwang Zhang, Huifeng Yin, Liwen Zhang, Litu Ou, Jialong Wu, Wenbiao Yin, Baixuan Li, Zhengwei Tao, Xinyu Wang, Weizhou Shen, Junkai Zhang, Dingchu Zhang, Xixi Wu, Yong Jiang, Ming Yan, Pengjun Xie, Fei Huang, and Jingren Zhou. Websailor: Navigating super-human reasoning for web agent. *CoRR*, abs/2507.02592, 2025. doi: 10.48550/ARXIV.2507.02592. URL https://doi.org/10.48550/arXiv.2507.02592.

[328] Kunchang Li, Yali Wang, Yinan He, Yizhuo Li, Yi Wang, Yi Liu, Zun Wang, Jilan Xu, Guo Chen, Ping Lou, Limin Wang, and Yu Qiao. Mvbench: A comprehensive multi-modal video understanding benchmark. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024*, pages 22195–22206. IEEE, 2024. doi: 10.1109/CVPR52733.2024.02095. URL https://doi.org/10.1109/CVPR52733.2024.02095.

[329] Mukai Li, Qingcheng Zeng, Tianqing Fang, Zhenwen Liang, Linfeng Song, Qi Liu, Haitao Mi, and Dong Yu. Verified critical step optimization for LLM agents. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens, editors, *Findings of the Association for Computational Linguistics, ACL 2026, San Diego, California, United States, July 2-7, 2026*, pages 39627–39639. Association for Computational Linguistics, 2026. URL https://aclanthology.org/2026.findings-acl.1974/.

[330] Qingbin Li, Rongkun Xue, Jie Wang, Ming Zhou, Zhi Li, Xiaofeng Ji, Yongqi Wang, Miao Liu, Zheming Yang, Minghui Qiu, and Jing Yang. CURE: critical-token-guided re-concatenation for entropy-collapse prevention. *CoRR*, abs/2508.11016, 2025. doi: 10.48550/ARXIV.2508.11016. URL https://doi.org/10.48550/arXiv.2508.11016.

[331] Shilong Li, Xingyuan Bu, Wenjie Wang, Jiaheng Liu, Jun Dong, Haoyang He, Hao Lu, Haozhe Zhang, Chenchen Jing, Zhen Li, Chuanhao Li, Jiayi Tian, Chenchen Zhang, Tianhao Peng, Yancheng He, Jihao Gu, Yuanxing Zhang, Jian Yang, Ge Zhang, Wenhao Huang, Wangchunshu Zhou, Zhaoxiang Zhang, Ruizhe Ding, and Shilei Wen. Mm-browsecomp: A comprehensive benchmark for multimodal browsing agents. *CoRR*, abs/2508.13186, 2025. doi: 10.48550/ARXIV.2508.13186. URL https://doi.org/10.48550/arXiv.2508.13186.

[332] Shiyu Li, Yang Tang, Yifan Wang, Peiming Li, and Xi Chen. Reseek: A self-correcting framework for search agents with instructive rewards. *CoRR*, abs/2510.00568, 2025. doi: 10.48550/ARXIV.2510.00568. URL https://doi.org/10.48550/arXiv.2510.00568.

[333] Sijia Li, Yuchen Huang, Zifan Liu, Yanping Li, Jingjing Fu, Li Zhao, Jiang Bian, Ling Zhang, Jun Zhang, and Rui Wang. GEAR: granularity-adaptive advantage reweighting for LLM agents via self-distillation. *CoRR*, abs/2605.11853, 2026. doi: 10.48550/ARXIV.2605.11853. URL https://doi.org/10.48550/arXiv.2605.11853.

[334] Wanli Li, Bince Qu, Bo Pan, Jianyu Zhang, Zheng Liu, Pan Zhang, Wei Chen, and Bo Zhang. Literesearcher: A scalable agentic RL training framework for deep research agent. *CoRR*, abs/2604.17931, 2026. doi: 10.48550/ARXIV.2604.17931. URL https://doi.org/10.48550/arXiv.2604.17931.

[335] Xiaochuan Li, Ryan Ming, Pranav Setlur, Abhijay Paladugu, Andy Tang, Hao Kang, Shuai Shao, Rong Jin, and Chenyan Xiong. Benchmark test-time scaling of general LLM agents. *CoRR*, abs/2602.18998, 2026. doi: 10.48550/ARXIV.2602.18998. URL https://doi.org/10.48550/arXiv.2602.18998.

[336] 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, editors, *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, EMNLP 2025, Suzhou, China, November 4-9, 2025*, pages 5420–5438. Association for Computational Linguistics, 2025. doi: 10.18653/V1/2025.EMNLP-MAIN.276. URL https://doi.org/10.18653/v1/2025.emnlp-main.276.

[337] Xiaoxi Li, Jiajie Jin, Guanting Dong, Hongjin Qian, Yongkang Wu, Ji-Rong Wen, Yutao Zhu, and Zhicheng Dou. Webthinker: Empowering large reasoning models with deep research capability. In Danielle Belgrave, Cheng Zhang, Laura N. Montoya, Hsuan-Tien Lin, Razvan Pascanu, Piotr Koniusz, Marzyeh Ghassemi, Nancy Chen, Iván Vladimir Meza Ruíz, and Arturo Loaiza-Bonilla, editors, *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2025, NeurIPS 2025, San Diego, CA, USA, December 2-7, 2025 / Mexico City, Mexico, November 30 - December 5, 2025*, 2025. URL http://papers.nips.cc/paper_files/paper/2025/hash/ae03bdef276132fae089692445725635-Abstract-Conference.html.

[338] Xiaoxi Li, Wenxiang Jiao, Jiarui Jin, Guanting Dong, Jiajie Jin, Yinuo Wang, Hao Wang, Yutao Zhu, Ji-Rong Wen, Yuan Lu, and Zhicheng Dou. Deepagent: A general reasoning agent with scalable toolsets. In Hakim Hacid, Yoelle Maarek, Francesco Bonchi, Ido Guy, and Emine Yilmaz, editors, *Proceedings of the ACM Web Conference 2026, WWW 2026, Dubai, United Arab Emirates, originally scheduled for April 13-17, 2026, rescheduled for June 29 - July 3, 2026*, pages 2219–2230. ACM, 2026. doi: 10.1145/3774904.3792460. URL https://doi.org/10.1145/3774904.3792460.

[339] Xiaoxi Li, Wenxiang Jiao, Jiarui Jin, Shijian Wang, Guanting Dong, Jiajie Jin, Hao Wang, Yinuo Wang, Ji-Rong Wen, Yuan Lu, and Zhicheng Dou. Omnigaia: Towards native omni-modal AI agents. *CoRR*, abs/2602.22897, 2026. doi: 10.48550/ARXIV.2602.22897. URL https://doi.org/10.48550/arXiv.2602.22897.

[340] Xiaoxiao Li. When single-agent with skills replace multi-agent systems and when they fail. *CoRR*, abs/2601.04748, 2026. doi: 10.48550/ARXIV.2601.04748. URL https://doi.org/10.48550/arXiv.2601.04748.

[341] Xiaoyuan Li, Moxin Li, Keqin Bao, Yubo Ma, Wenjie Wang, Dayiheng Liu, and Fuli Feng. Skillgraph: Skill-augmented reinforcement learning for agents via evolving skill graphs. *CoRR*, abs/2605.12039, 2026. doi: 10.48550/ARXIV.2605.12039. URL https://doi.org/10.48550/arXiv.2605.12039.

[342] Xiaoyuan Li, Yubo Ma, Chengpeng Li, Fengbin Zhu, Yiyao Yu, Keqin Bao, Wenjie Wang, Fuli Feng, and Dayiheng Liu. Unified data selection for LLM reasoning. *CoRR*, abs/2605.22389, 2026. doi: 10.48550/ARXIV.2605.22389. URL https://doi.org/10.48550/arXiv.2605.22389.

[343] Xinzhe Li. Chain-in-tree: Back to sequential reasoning in LLM tree search. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens, editors, *Findings of the Association for Computational Linguistics, ACL 2026, San Diego, California, United States, July 2-7, 2026*, pages 4365–4392. Association for Computational Linguistics, 2026. URL https://aclanthology.org/2026.findings-acl.214/.

[344] Xuefeng Li, Haoyang Zou, and Pengfei Liu. Limr: Less is more for rl scaling. *CoRR*, abs/2502.11886, 2025. doi: 10.48550/ARXIV.2502.11886. URL https://doi.org/10.48550/arXiv.2502.11886.

[345] Xuefeng Li, Haoyang Zou, and Pengfei Liu. Torl: Scaling tool-integrated RL. *CoRR*, abs/2503.23383, 2025. doi: 10.48550/ARXIV.2503.23383. URL https://doi.org/10.48550/arXiv.2503.23383.

[346] Yangning Li, Yinghui Li, Xinyu Wang, Yong Jiang, Zhen Zhang, Xinran Zheng, Hui Wang, Hai-Tao Zheng, Fei Huang, Jingren Zhou, and Philip S. Yu. Benchmarking multimodal retrieval augmented generation with dynamic VQA dataset and self-adaptive planning agent. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=VvDEuyVXkG.

[347] Yishan Li, Wentong Chen, Yukun Yan, Mingwei Li, Sen Mei, Xiaorong Wang, Kunpeng Liu, Xin Cong, Shuo Wang, Zhong Zhang, Yaxi Lu, Zhenghao Liu, Yankai Lin, Zhiyuan Liu, and Maosong Sun. Agentcpm-report: Interleaving drafting and deepening for open-ended deep research. *CoRR*, abs/2602.06540, 2026. doi: 10.48550/ARXIV.2602.06540. URL https://doi.org/10.48550/arXiv.2602.06540.

[348] Yizhi Li, Qingshui Gu, Zhoufutu Wen, Ziniu Li, Tianshun Xing, Shuyue Guo, Tianyu Zheng, Xin Zhou, Xingwei Qu, Wangchunshu Zhou, Zheng Zhang, Wei Shen, Qian Liu, Chenghua Lin, Jian Yang, Ge Zhang, and Wenhao Huang. Treepo: Bridging the gap of policy optimization and efficacy and inference efficiency with heuristic tree-based modeling. *CoRR*, abs/2508.17445, 2025. doi: 10.48550/ARXIV.2508.17445. URL https://doi.org/10.48550/arXiv.2508.17445.

[349] Yuanfan Li, Qi Zhou, Wenjing Duan, and Lu Chen. When denser credit is not enough: Evidence-calibrated policy optimization for long-horizon LLM agent training. *CoRR*, abs/2606.05885, 2026. doi: 10.48550/ARXIV.2606.05885. URL https://doi.org/10.48550/arXiv.2606.05885.

[350] Yuanyang Li, Xue Yang, Longyue Wang, Weihua Luo, and Hongyang Chen. Complexmcp: Evaluation of LLM agents in dynamic, interdependent, and large-scale tool sandbox. *CoRR*, abs/2605.10787, 2026. doi: 10.48550/ARXIV.2605.10787. URL https://doi.org/10.48550/arXiv.2605.10787.

[351] Yucheng Li, Bo Dong, Frank Guerin, and Chenghua Lin. Compressing context to enhance inference efficiency of large language models. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 6342–6353. Association for Computational Linguistics, 2023. doi: 10.18653/V1/2023.EMNLP-MAIN.391. URL https://doi.org/10.18653/v1/2023.emnlp-main.391.

[352] Yuhui Li, Fangyun Wei, Chao Zhang, and Hongyang Zhang. EAGLE: speculative sampling requires rethinking feature uncertainty. In Ruslan Salakhutdinov, Zico Kolter, Katherine A. Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp, editors, *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*, volume 235 of *Proceedings of Machine Learning Research*, pages 28935–28948. PMLR / OpenReview.net, 2024. URL https://proceedings.mlr.press/v235/li24bt.html.

[353] Zelong Li, Shuyuan Xu, Kai Mei, Wenyue Hua, Balaji Rama, Om Raheja, Hao Wang, He Zhu, and Yongfeng Zhang. Autoflow: Automated workflow generation for large language model agents. *CoRR*, abs/2407.12821, 2024. doi: 10.48550/ARXIV.2407.12821. URL https://doi.org/10.48550/arXiv.2407.12821.

[354] Zhuofeng Li, Dongfu Jiang, Xueguang Ma, Haoxiang Zhang, Ping Nie, Yuyu Zhang, Kai Zou, Jianwen Xie, Yu Zhang, and Wenhu Chen. Openresearcher: A fully open pipeline for long-horizon deep research trajectory

synthesis. *CoRR*, abs/2603.20278, 2026. doi: 10.48550/ARXIV.2603.20278. URL https://doi.org/10.48550/arXiv.2603.20278.

[355] Zhuofeng Li, Haoxiang Zhang, Cong Wei, Pan Lu, Ping Nie, Yi Lu, Yuyang Bai, Shangbin Feng, Hangxiao Zhu, Ming Zhong, Yuyu Zhang, Jianwen Xie, Yejin Choi, James Zou, Jiawei Han, Wenhu Chen, Jimmy Lin, Dongfu Jiang, and Yu Zhang. Beyond semantic similarity: Rethinking retrieval for agentic search via direct corpus interaction. *CoRR*, abs/2605.05242, 2026. doi: 10.48550/ARXIV.2605.05242. URL https://doi.org/10.48550/arXiv.2605.05242.

[356] Zijian Li, Xin Guan, Bo Zhang, Shen Huang, Houquan Zhou, Shaopeng Lai, Ming Yan, Yong Jiang, Pengjun Xie, Fei Huang, Jun Zhang, and Jingren Zhou. Webweaver: Structuring web-scale evidence with dynamic outlines for open-ended deep research. *CoRR*, abs/2509.13312, 2025. doi: 10.48550/ARXIV.2509.13312. URL https://doi.org/10.48550/arXiv.2509.13312.

[357] Jacky Liang, Wenlong Huang, Fei Xia, Peng Xu, Karol Hausman, Brian Ichter, Pete Florence, and Andy Zeng. Code as policies: Language model programs for embodied control. In *IEEE International Conference on Robotics and Automation, ICRA 2023, London, UK, May 29 - June 2, 2023*, pages 9493–9500. IEEE, 2023. doi: 10.1109/ICRA48891.2023.10160591. URL https://doi.org/10.1109/ICRA48891.2023.10160591.

[358] Tian Liang, Zhiwei He, Wenxiang Jiao, Xing Wang, Yan Wang, Rui Wang, Yujiu Yang, Shuming Shi, and Zhaopeng Tu. Encouraging divergent thinking in large language models through multi-agent debate. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen, editors, *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, EMNLP 2024, Miami, FL, USA, November 12-16, 2024*, pages 17889–17904. Association for Computational Linguistics, 2024. doi: 10.18653/V1/2024.EMNLP-MAIN.992. URL https://doi.org/10.18653/v1/2024.emnlp-main.992.

[359] Xuechen Liang, Meiling Tao, Yinghui Xia, Jianhui Wang, Kun Li, Yijin Wang, Yangfan He, Jingsong Yang, Tianyu Shi, Yuantao Wang, Miao Zhang, and Xueqian Wang. SAGE: self-evolving agents with reflective and memory-augmented abilities. *Neurocomputing*, 647:130470, 2025. doi: 10.1016/J.NEUCOM.2025.130470. URL https://doi.org/10.1016/j.neucom.2025.130470.

[360] Yuan Liang, Ruobin Zhong, Haoming Xu, Chen Jiang, Yi Zhong, Runnan Fang, Jia-Chen Gu, Shumin Deng, Yunzhi Yao, Mengru Wang, Shuofei Qiao, Xin Xu, Tongtong Wu, Kun Wang, Yang Liu, Zhen Bi, Jungang Lou, Yuchen Eleanor Jiang, Hangcheng Zhu, Gang Yu, Haiwen Hong, Longtao Huang, Hui Xue, Chenxi Wang, Yijun Wang, Zifei Shan, Xi Chen, Zhaopeng Tu, Feiyu Xiong, Xin Xie, Peng Zhang, Zhengke Gui, Lei Liang, Jun Zhou, Chiyu Wu, Jin Shang, Yu Gong, Junyu Lin, Changliang Xu, Hongjie Deng, Wen Zhang, Keyan Ding, Qiang Zhang, Fei Huang, Ningyu Zhang, Jeff Z. Pan, Guilin Qi, Haofen Wang, and Huajun Chen. Skillnet: Create, evaluate, and connect AI skills. *CoRR*, abs/2603.04448, 2026. doi: 10.48550/ARXIV.2603.04448. URL https://doi.org/10.48550/arXiv.2603.04448.

[361] Zhengyang Liang, Yan Shu, Xiangrui Liu, Minghao Qin, Kaixin Liang, Nicu Sebe, Zheng Liu, and Lizi Liao. Video-browser: Towards agentic open-web video browsing, 2026. URL https://arxiv.org/abs/2512.23044.

[362] Shalev Lifshitz, Sheila A. McIlraith, and Yilun Du. Multi-agent verification: Scaling test-time compute with multiple verifiers. *CoRR*, abs/2502.20379, 2025. doi: 10.48550/ARXIV.2502.20379. URL https://doi.org/10.48550/arXiv.2502.20379.

[363] Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=v8L0pN6EOi.

[364] Huawei Lin, Yunzhi Shi, Tong Geng, Weijie Zhao, Wei Wang, and Ravender Pal Singh. Agent-omni: Test-time multimodal reasoning via model coordination for understanding anything. *CoRR*, abs/2511.02834, 2025. doi: 10.48550/ARXIV.2511.02834. URL https://doi.org/10.48550/arXiv.2511.02834.

[365] Jingyang Lin, Jialian Wu, Jiang Liu, Ximeng Sun, Ze Wang, Xiaodong Yu, Jiebo Luo, Zicheng Liu, and Emad Barsoum. Videoseek: Long-horizon video agent with tool-guided seeking. *CoRR*, abs/2603.20185, 2026. doi: 10.48550/ARXIV.2603.20185. URL https://doi.org/10.48550/arXiv.2603.20185.

[366] Junming Lin, Zheng Fang, Chi Chen, Zihao Wan, Fuwen Luo, Peng Li, Yang Liu, and Maosong Sun. Streamingbench: Assessing the gap for mllms to achieve streaming video understanding. *CoRR*, abs/2411.03628, 2024. doi: 10.48550/ARXIV.2411.03628. URL https://doi.org/10.48550/arXiv.2411.03628.

[367] Minhua Lin, Zongyu Wu, Zhichao Xu, Hui Liu, Xianfeng Tang, Qi He, Charu C. Aggarwal, Hui Liu, Xiang Zhang, and Suhang Wang. A comprehensive survey on reinforcement learning-based agentic search: Foundations, roles, optimizations, evaluations, and applications. *CoRR*, abs/2510.16724, 2025. doi: 10.48550/ARXIV.2510.16724. URL https://doi.org/10.48550/arXiv.2510.16724.

[368] Xixun Lin, Yang Liu, Yancheng Chen, Yongxuan Wu, Yucheng Ning, Yilong Liu, Nan Sun, Shun Zhang, Bin Chong, Chuan Zhou, Yanan Cao, and Li Guo. Safeharness: Lifecycle-integrated security architecture for llm-based agent deployment. *CoRR*, abs/2604.13630, 2026. doi: 10.48550/ARXIV.2604.13630. URL https://doi.org/10.48550/arXiv.2604.13630.

[369] Yusong Lin, Haiyang Wang, Shuzhe Wu, Lue Fan, Feiyang Pan, Sanyuan Zhao, and Dandan Tu. Cli-gym: Scalable CLI task generation via agentic environment inversion. *CoRR*, abs/2602.10999, 2026. doi: 10.48550/ARXIV.2602.10999. URL https://doi.org/10.48550/arXiv.2602.10999.

[370] Yuxiang Lin, Zihan Wang, Mengyang Liu, Yuxuan Shan, Longju Bai, Junyao Zhang, Xing Jin, Boshan Chen, Jinyan Su, Xingyao Wang, Jiaxin Pei, and Manling Li. BAGEN: are LLM agents budget-aware? *CoRR*, abs/2606.00198, 2026. doi: 10.48550/ARXIV.2606.00198. URL https://doi.org/10.48550/arXiv.2606.00198.

[371] Bo Liu, Leon Guertler, Simon Yu, Zichen Liu, Penghui Qi, Daniel Balcells, Mickel Liu, Cheston Tan, Weiyan Shi, Min Lin, Wee Sun Lee, and Natasha Jaques. SPIRAL: self-play on zero-sum games incentivizes reasoning via multi-agent multi-turn reinforcement learning. *CoRR*, abs/2506.24119, 2025. doi: 10.48550/ARXIV.2506.24119. URL https://doi.org/10.48550/arXiv.2506.24119.

[372] Chengwen Liu, Xiaomin Yu, Zhuoyue Chang, Zhe Huang, Shuo Zhang, Heng Lian, Kunyi Wang, Rui Xu, Sen Hu, Jianheng Hou, Hao Peng, Chengwei Qin, Xiaobin Hu, Hong Peng, Ronghao Chen, and Huacan Wang. Watching, reasoning, and searching: A video deep research benchmark on open web for agentic video reasoning. *CoRR*, abs/2601.06943, 2026. doi: 10.48550/ARXIV.2601.06943. URL https://doi.org/10.48550/arXiv.2601.06943.

[373] Guangyi Liu, Pengxiang Zhao, Liang Liu, Zhiming Chen, Yuxiang Chai, Yaozhen Liang, WenHao Wang, Siheng Chen, Zhengxi Lu, Shuai Ren, Hao Wang, Shibo He, Yong Liu, and Wenchao Meng. Learnact: Few-shot mobile GUI agent with a unified demonstration benchmark. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens, editors, *Findings of the Association for Computational Linguistics, ACL 2026, San Diego, California, United States, July 2-7, 2026*, pages 29820–29843. Association for Computational Linguistics, 2026. URL https://aclanthology.org/2026.findings-acl.1491/.

[374] Haoran Liu, Yuwei Zhang, Xiyao Li, Bohan Lyu, and Jingbo Shang. HERO: hindsight-enhanced reflection from environment observations for agentic self-distillation. *CoRR*, abs/2606.11559, 2026. doi: 10.48550/ARXIV.2606.11559. URL https://doi.org/10.48550/arXiv.2606.11559.

[375] Jingyuan Liu, Jianlin Su, Xingcheng Yao, Zhejun Jiang, Guokun Lai, Yulun Du, Yidao Qin, Weixin Xu, Enzhe Lu, Junjie Yan, Yanru Chen, Huabin Zheng, Yibo Liu, Shaowei Liu, Bohong Yin, Weiran He, Han Zhu, Yuzhi Wang, Jianzhou Wang, Mengnan Dong, Zheng Zhang, Yongsheng Kang, Hao Zhang, Xinran Xu, Yutao Zhang, Yuxin Wu, Xinyu Zhou, and Zhilin Yang. Muon is scalable for LLM training. *CoRR*, abs/2502.16982, 2025. doi: 10.48550/ARXIV.2502.16982. URL https://doi.org/10.48550/arXiv.2502.16982.

[376] Jun Liu, Zhenglun Kong, Peiyan Dong, Changdi Yang, Tianqi Li, Hao Tang, Geng Yuan, Wei Niu, Wenbin Zhang, Pu Zhao, Xue Lin, Dong Huang, and Yanzhi Wang. Structured agent distillation for large language model. *CoRR*, abs/2505.13820, 2025. doi: 10.48550/ARXIV.2505.13820. URL https://doi.org/10.48550/arXiv.2505.13820.

[377] 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. *CoRR*, abs/2509.06501, 2025. doi: 10.48550/ARXIV.2509.06501. URL https://doi.org/10.48550/arXiv.2509.06501.

[378] Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. Lost in the middle: How language models use long contexts. *Trans. Assoc. Comput. Linguistics*, 12:157–173, 2024. doi: 10.1162/TACL_A_00638. URL https://doi.org/10.1162/tacl_a_00638.

[379] Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Comput. Surv.*, 55(9):195:1–195:35, 2023. doi: 10.1145/3560815. URL https://doi.org/10.1145/3560815.

[380] Ruoqi Liu, Imran Q. Mohiuddin, Austin J. Schoeffler, Kavita Renduchintala, Ashwin Nayak, Prasantha L. Vemu, Shivam Vedak, Kameron C. Black, John L. Havlik, Isaac Ogunmola, Stephen P. Ma, Roopa Dhatt,

and Jonathan H. Chen. Physicianbench: Evaluating LLM agents in real-world EHR environments. *CoRR*, abs/2605.02240, 2026. doi: 10.48550/ARXIV.2605.02240. URL https://doi.org/10.48550/arXiv.2605.02240.

[381] Shukai Liu, Bo Jiang, Jian Yang, Yizhi Li, Jinyang Guo, Xianglong Liu, and Bryan Dai. Context as a tool: Context management for long-horizon swe-agents. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens, editors, *Findings of the Association for Computational Linguistics, ACL 2026, San Diego, California, United States, July 2-7, 2026*, pages 20604–20617. Association for Computational Linguistics, 2026. URL https://aclanthology.org/2026.findings-acl.1032/.

[382] Shuo Liu, Zeyu Liang, Xueguang Lyu, and Christopher Amato. LLM collaboration with multi-agent reinforcement learning. In Sven Koenig, Chad Jenkins, and Matthew E. Taylor, editors, *Fortieth AAAI Conference on Artificial Intelligence, Thirty-Eighth Conference on Innovative Applications of Artificial Intelligence, Sixteenth Symposium on Educational Advances in Artificial Intelligence, AAAI 2026, Singapore, January 20-27, 2026*, pages 32150–32158. AAAI Press, 2026. doi: 10.1609/AAAI.V40I38.40487. URL https://doi.org/10.1609/aaai.v40i38.40487.

[383] Siwei Liu, Jinyuan Fang, Han Zhou, Yingxu Wang, and Zaiqiao Meng. SEW: self-evolving agentic workflows for automated code generation. *CoRR*, abs/2505.18646, 2025. doi: 10.48550/ARXIV.2505.18646. URL https://doi.org/10.48550/arXiv.2505.18646.

[384] Tianci Liu, Ran Xu, Tony Yu, Ilgee Hong, Carl Yang, Tuo Zhao, and Haoyu Wang. Openrubrics: Towards scalable synthetic rubric generation for reward modeling and LLM alignment. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens, editors, *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2026, San Diego, California, United States, July 2-7, 2026*, pages 17417–17437. Association for Computational Linguistics, 2026. URL https://aclanthology.org/2026.acl-long.791/.

[385] Weiwen Liu, Xu Huang, Xingshan Zeng, Xinlong Hao, Shuai Yu, Dexun Li, Shuai Wang, Weinan Gan, Zhengying Liu, Yuanqing Yu, Zezhong Wang, Yuxian Wang, Wu Ning, Yutai Hou, Bin Wang, Chuhan Wu, Xinzhi Wang, Yong Liu, Yasheng Wang, Duyu Tang, Dandan Tu, Lifeng Shang, Xin Jiang, Ruiming Tang, Defu Lian, Qun Liu, and Enhong Chen. Toolace: Winning the points of LLM function calling. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=8EB8k6DdCU.

[386] Wenhan Liu, Jiajie Jin, Zhaoheng Huang, Tongyu Wen, Guanting Dong, Ziliang Zhao, Yutao Zhu, Zhicheng Dou, and Ji-Rong Wen. The rules of the game: A survey of rubrics for large language models, 2026. URL https://openreview.net/pdf?id=FnSimngGYk.

[387] Xiao Liu, Hao Yu, Hanchen Zhang, Yifan Xu, Xuanyu Lei, Hanyu Lai, Yu Gu, Hangliang Ding, Kaiwen Men, Kejuan Yang, Shudan Zhang, Xiang Deng, Aohan Zeng, Zhengxiao Du, Chenhui Zhang, Sheng Shen, Tianjun Zhang, Yu Su, Huan Sun, Minlie Huang, Yuxiao Dong, and Jie Tang. Agentbench: Evaluating llms as agents. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=zAdUB0aCTQ.

[388] Yibing Liu, Chong Zhang, Zhongyi Han, Hansong Liu, Yong Wang, Yang Yu, Xiaoyan Wang, and Yilong Yin. Trajad: Trajectory anomaly detection for trustworthy LLM agents. *CoRR*, abs/2602.06443, 2026. doi: 10.48550/ARXIV.2602.06443. URL https://doi.org/10.48550/arXiv.2602.06443.

[389] Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. Understanding r1-zero-like training: A critical perspective. *CoRR*, abs/2503.20783, 2025. doi: 10.48550/ARXIV.2503.20783. URL https://doi.org/10.48550/arXiv.2503.20783.

[390] Zijun Liu, Yanzhe Zhang, Peng Li, Yang Liu, and Diyi Yang. Dynamic llm-agent network: An llm-agent collaboration framework with agent team optimization. *CoRR*, abs/2310.02170, 2023. doi: 10.48550/ARXIV.2310.02170. URL https://doi.org/10.48550/arXiv.2310.02170.

[391] Zongkai Liu, Fanqing Meng, Lingxiao Du, Zhixiang Zhou, Chao Yu, Wenqi Shao, and Qiaosheng Zhang. CPGD: toward stable rule-based reinforcement learning for language models. *CoRR*, abs/2505.12504, 2025. doi: 10.48550/ARXIV.2505.12504. URL https://doi.org/10.48550/arXiv.2505.12504.

[392] Zuxin Liu, Thai Hoang, Jianguo Zhang, Ming Zhu, Tian Lan, Shirley Kokane, Juntao Tan, Weiran Yao, Zhiwei Liu, Yihao Feng, Rithesh R. N., Liangwei Yang, Silvio Savarese, Juan Carlos Niebles, Huan Wang, Shelby Heinecke, and Caiming Xiong. Apigen: Automated pipeline for generating verifiable and diverse function-calling datasets. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang, editors, *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver,*

BC, Canada, December 10 - 15, 2024, 2024. URL http://papers.nips.cc/paper_files/paper/2024/hash/61cce86d180b1184949e58939c4f983d-Abstract-Datasets_and_Benchmarks_Track.html.

[393] LLM-Core Team, Xiaomi. Mimo: Unlocking the reasoning potential of language model – from pretraining to posttraining. *CoRR*, abs/2505.07608, 2025. doi: 10.48550/ARXIV.2505.07608. URL https://doi.org/10.48550/arXiv.2505.07608.

[394] LLM-Core Team, Xiaomi. Mimo-vl technical report. *CoRR*, abs/2506.03569, 2025. doi: 10.48550/ARXIV.2506.03569. URL https://doi.org/10.48550/arXiv.2506.03569.

[395] 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. *CoRR*, abs/2603.03329, 2026. doi: 10.48550/ARXIV.2603.03329. URL https://doi.org/10.48550/arXiv.2603.03329.

[396] Ryan Lowe, Yi Wu, Aviv Tamar, Jean Harb, Pieter Abbeel, and Igor Mordatch. Multi-agent actor-critic for mixed cooperative-competitive environments. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 6379–6390, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/68a9750337a418a86fe06c1991a1d64c-Abstract.html.

[397] Chris Lu, Cong Lu, Robert Tjarko Lange, Jakob N. Foerster, Jeff Clune, and David Ha. The AI scientist: Towards fully automated open-ended scientific discovery. *CoRR*, abs/2408.06292, 2024. doi: 10.48550/ARXIV.2408.06292. URL https://doi.org/10.48550/arXiv.2408.06292.

[398] Enzhe Lu, Zhejun Jiang, Jingyuan Liu, Yulun Du, Tao Jiang, Chao Hong, Shaowei Liu, Weiran He, Enming Yuan, Yuzhi Wang, Zhiqi Huang, Huan Yuan, Suting Xu, Xinran Xu, Guokun Lai, Yanru Chen, Huabin Zheng, Junjie Yan, Jianlin Su, Yuxin Wu, Neo Y. Zhang, Zhilin Yang, Xinyu Zhou, Mingxing Zhang, and Jiezhong Qiu. Moba: Mixture of block attention for long-context llms. *CoRR*, abs/2502.13189, 2025. doi: 10.48550/ARXIV.2502.13189. URL https://doi.org/10.48550/arXiv.2502.13189.

[399] Jiarui Lu, Thomas Holleis, Yizhe Zhang, Bernhard Aumayer, Feng Nan, Haoping Bai, Shuang Ma, Shen Ma, Mengyu Li, Guoli Yin, Zirui Wang, and Ruoming Pang. Toolsandbox: A stateful, conversational, interactive evaluation benchmark for LLM tool use capabilities. In Luis Chiruzzo, Alan Ritter, and Lu Wang, editors, *Findings of the Association for Computational Linguistics: NAACL 2025, Albuquerque, New Mexico, USA, April 29 - May 4, 2025*, volume NAACL 2025 of *Findings of ACL*, pages 1160–1183. Association for Computational Linguistics, 2025. doi: 10.18653/V1/2025.FINDINGS-NAACL.65. URL https://doi.org/10.18653/v1/2025.findings-naacl.65.

[400] Jiaxuan Lu, Ziyu Kong, Yemin Wang, Rong Fu, Haiyuan Wan, Cheng Yang, Wenjie Lou, Haoran Sun, Lilong Wang, Yankai Jiang, Xiaosong Wang, Xiao Sun, and Dongzhan Zhou. Beyond static tools: Test-time tool evolution for scientific reasoning. *CoRR*, abs/2601.07641, 2026. doi: 10.48550/ARXIV.2601.07641. URL https://doi.org/10.48550/arXiv.2601.07641.

[401] Siyuan Lu, Zechuan Wang, Hongxuan Zhang, Qintong Wu, Leilei Gan, Chenyi Zhuang, Jinjie Gu, and Tao Lin. Don't just fine-tune the agent, tune the environment. *CoRR*, abs/2510.10197, 2025. doi: 10.48550/ARXIV.2510.10197. URL https://doi.org/10.48550/arXiv.2510.10197.

[402] Xing Han Lù, Amirhossein Kazemnejad, Nicholas Meade, Arkil Patel, Dongchan Shin, Alejandra Zambrano, Karolina Stanczak, Peter Shaw, Christopher J. Pal, and Siva Reddy. Agentrewardbench: Evaluating automatic evaluations of web agent trajectories. *CoRR*, abs/2504.08942, 2025. doi: 10.48550/ARXIV.2504.08942. URL https://doi.org/10.48550/arXiv.2504.08942.

[403] Yao Lu, Max Bartolo, Alastair Moore, Sebastian Riedel, and Pontus Stenetorp. Fantastically ordered prompts and where to find them: Overcoming few-shot prompt order sensitivity. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio, editors, *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 8086–8098. Association for Computational Linguistics, 2022. doi: 10.18653/V1/2022.ACL-LONG.556. URL https://doi.org/10.18653/v1/2022.acl-long.556.

[404] Yuxuan Lu, Ziyi Wang, Jing Huang, Hui Liu, Jiri Gesi, Yan Han, Shihan Fu, Tianqi Zheng, Xianfeng Tang, Chen Luo, Yisi Sang, Jin Lai, and Dakuo Wang. Webserv: A full-stack and rl-ready web environment for training web agents at scale, 2026. URL https://arxiv.org/abs/2510.16252.

[405] Zhengxi Lu, Zhiyuan Yao, Jinyang Wu, Chengcheng Han, Qi Gu, Xunliang Cai, Weiming Lu, Jun Xiao, Yueting

Zhuang, and Yongliang Shen. SKILL0: in-context agentic reinforcement learning for skill internalization. *CoRR*, abs/2604.02268, 2026. doi: 10.48550/ARXIV.2604.02268. URL https://doi.org/10.48550/arXiv.2604.02268.

[406] Haotian Luo, Huaisong Zhang, Xuelin Zhang, Haoyu Wang, Zeyu Qin, Wenjie Lu, Guozheng Ma, Haiying He, Yingsha Xie, Qiyang Zhou, Zixuan Hu, Hongze Mi, Yibo Wang, Naiqiang Tan, Hong Chen, Yi R. Fung, Chun Yuan, and Li Shen. Ultrahorizon: Benchmarking agent capabilities in ultra long-horizon scenarios. *CoRR*, abs/2509.21766, 2025. doi: 10.48550/ARXIV.2509.21766. URL https://doi.org/10.48550/arXiv.2509.21766.

[407] Jinghao Luo, Yuchen Tian, Chuxue Cao, Ziyang Luo, Hongzhan Lin, Kaixin Li, Chuyi Kong, Ruichao Yang, and Jing Ma. From storage to experience: A survey on the evolution of LLM agent memory mechanisms. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens, editors, *Findings of the Association for Computational Linguistics, ACL 2026, San Diego, California, United States, July 2-7, 2026*, pages 41622–41652. Association for Computational Linguistics, 2026. URL https://aclanthology.org/2026.findings-acl.2069/.

[408] Weidi Luo, Shenghong Dai, Xiaogeng Liu, Suman Banerjee, Huan Sun, Muhao Chen, and Chaowei Xiao. Agrail: A lifelong agent guardrail with effective and adaptive safety detection. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar, editors, *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, pages 8104–8139. Association for Computational Linguistics, 2025. doi: 10.18653/V1/2025.ACL-LONG.399. URL https://doi.org/10.18653/v1/2025.acl-long.399.

[409] Xufang Luo, Yuge Zhang, Zhiyuan He, Zilong Wang, Siyun Zhao, Dongsheng Li, Luna K. Qiu, and Yuqing Yang. Agent lightning: Train ANY AI agents with reinforcement learning. *CoRR*, abs/2508.03680, 2025. doi: 10.48550/ARXIV.2508.03680. URL https://doi.org/10.48550/arXiv.2508.03680.

[410] Yinyi Luo, Yiqiao Jin, Weichen Yu, Mengqi Zhang, Srijan Kumar, Xiaoxiao Li, Weijie Xu, Xin Chen, and Jindong Wang. Agentark: Distilling multi-agent intelligence into a single LLM agent. *CoRR*, abs/2602.03955, 2026. doi: 10.48550/ARXIV.2602.03955. URL https://doi.org/10.48550/arXiv.2602.03955.

[411] Ziyang Luo, Zhiqi Shen, Wenzhuo Yang, Zirui Zhao, Prathyusha Jwalapuram, Amrita Saha, Doyen Sahoo, Silvio Savarese, Caiming Xiong, and Junnan Li. Mcp-universe: Benchmarking large language models with real-world model context protocol servers. *CoRR*, abs/2508.14704, 2025. doi: 10.48550/ARXIV.2508.14704. URL https://doi.org/10.48550/arXiv.2508.14704.

[412] Yuanjie Lyu, Chengyu Wang, Jun Huang, and Tong Xu. From correction to mastery: Reinforced distillation of large language model agents. *CoRR*, abs/2509.14257, 2025. doi: 10.48550/ARXIV.2509.14257. URL https://doi.org/10.48550/arXiv.2509.14257.

[413] Wentao Ma, Weiming Ren, Yiming Jia, Zhuofeng Li, Ping Nie, Ge Zhang, and Wenhu Chen. Videoeval-pro: Robust and realistic long video understanding evaluation. *CoRR*, abs/2505.14640, 2025. doi: 10.48550/ARXIV.2505.14640. URL https://doi.org/10.48550/arXiv.2505.14640.

[414] Yubo Ma, Yuhang Zang, Liangyu Chen, Meiqi Chen, Yizhu Jiao, Xinze Li, Xinyuan Lu, Ziyu Liu, Yan Ma, Xiaoyi Dong, Pan Zhang, Liangming Pan, Yu-Gang Jiang, Jiaqi Wang, Yixin Cao, and Aixin Sun. MMLONGBENCH-DOC: benchmarking long-context document understanding with visualizations. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang, editors, *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. URL http://papers.nips.cc/paper_files/paper/2024/hash/ae0e43289bffea0c1fa34633fc608e92-Abstract-Datasets_and_Benchmarks_Track.html.

[415] Zeyao Ma, Bohan Zhang, Jing Zhang, Jifan Yu, Xiaokang Zhang, Xiaohan Zhang, Sijia Luo, Xi Wang, and Jie Tang. Spreadsheetbench: Towards challenging real world spreadsheet manipulation. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang, editors, *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. URL http://papers.nips.cc/paper_files/paper/2024/hash/ac840df270ac537dd74530a15c332684-Abstract-Datasets_and_Benchmarks_Track.html.

[416] Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. Self-refine: Iterative refinement with self-feedback. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine, editors, *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/91edff07232fb1b55a505a9e9f6c0ff3-Abstract-Conference.html.

[417] Shinji Mai, Yunpeng Zhai, Ziqian Chen, Cheng Chen, Anni Zou, Shuchang Tao, Zhaoyang Liu, and Bolin Ding. Cues: A curiosity-driven and environment-grounded synthesis framework for agentic RL. *CoRR*, abs/2512.01311, 2025. doi: 10.48550/ARXIV.2512.01311. URL https://doi.org/10.48550/arXiv.2512.01311.

[418] Potsawee Manakul, Adian Liusie, and Mark J. F. Gales. Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 9004–9017. Association for Computational Linguistics, 2023. doi: 10.18653/V1/2023.EMNLP-MAIN.557. URL https://doi.org/10.18653/v1/2023.emnlp-main.557.

[419] Karttikeya Mangalam, Raiymbek Akshulakov, and Jitendra Malik. Egoschema: A diagnostic benchmark for very long-form video language understanding. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/90ce332aff156b910b002ce4e6880dec-Abstract-Datasets_and_Benchmarks.html.

[420] Manus Team. Manus: Hands on ai, 2025. URL https://manus.im/.

[421] Rahul Marchand, Art Ó Catháin, Jerome Wynne, Philippos Maximos Giavridis, Sam Deverett, John Wilkinson, Jason Gwartz, and Harry Coppock. Quantifying frontier LLM capabilities for container sandbox escape. *CoRR*, abs/2603.02277, 2026. doi: 10.48550/ARXIV.2603.02277. URL https://doi.org/10.48550/arXiv.2603.02277.

[422] Samuele Marro, Emanuele La Malfa, Jesse Wright, Guohao Li, Nigel Shadbolt, Michael J. Wooldridge, and Philip Torr. A scalable communication protocol for networks of large language models. *CoRR*, abs/2410.11905, 2024. doi: 10.48550/ARXIV.2410.11905. URL https://doi.org/10.48550/arXiv.2410.11905.

[423] Meriem Mastouri, Emna Ksontini, and Wael Kessentini. Making REST apis agent-ready: From openapi to model context protocol servers for tool-augmented llms. *CoRR*, abs/2507.16044, 2025. doi: 10.48550/ARXIV.2507.16044. URL https://doi.org/10.48550/arXiv.2507.16044.

[424] Kai Mei, Zelong Li, Shuyuan Xu, Ruosong Ye, Yingqiang Ge, and Yongfeng Zhang. AIOS: LLM agent operating system. *CoRR*, abs/2403.16971, 2024. doi: 10.48550/ARXIV.2403.16971. URL https://doi.org/10.48550/arXiv.2403.16971.

[425] Lingrui Mei, Jiayu Yao, Yuyao Ge, Yiwei Wang, Baolong Bi, Yujun Cai, Jiazhi Liu, Mingyu Li, Zhong-Zhi Li, Duzhen Zhang, Chenlin Zhou, Jiayi Mao, Tianze Xia, Jiafeng Guo, and Shenghua Liu. A survey of context engineering for large language models. *CoRR*, abs/2507.13334, 2025. doi: 10.48550/ARXIV.2507.13334. URL https://doi.org/10.48550/arXiv.2507.13334.

[426] Fanqing Meng, Lingxiao Du, Zijian Wu, Guanzheng Chen, Xiangyan Liu, Jiaqi Liao, Chonghe Jiang, Zhenglin Wan, Jiawei Gu, Pengfei Zhou, Rui Huang, Ziqi Zhao, Shengyuan Ding, Ailing Yu, Bo Peng, Bowei Xia, Hao Sun, Haotian Liang, Ji Xie, Jiajun Chen, Jiajun Song, Liu Yang, Ming Xu, Qionglin Qiu, Runhao Fu, Shengfang Zhai, Shijian Wang, Tengfei Ma, Tianyi Wu, Weiyang Jin, Yan Wang, Yang Dai, Yao Lai, Youwei Shu, Yue Liu, Yunzhuo Hao, Yuwei Niu, Jinkai Huang, Jiayuan Zhuo, Zhennan Shen, Linyu Wu, Cihang Xie, Yuyin Zhou, Jiaheng Zhang, Zeyu Zheng, Mengkang Hu, and Michael Qizhe Shieh. Clawmark: A living-world benchmark for multi-turn, multi-day, multimodal coworker agents. *CoRR*, abs/2604.23781, 2026. doi: 10.48550/ARXIV.2604.23781. URL https://doi.org/10.48550/arXiv.2604.23781.

[427] Achyutha Menon, Magnus Saebo, Tyler Crosse, Spencer Gibson, Eyon Jang, and Diogo Cruz. Inherited goal drift: Contextual pressure can undermine agentic goals. *CoRR*, abs/2603.03258, 2026. doi: 10.48550/ARXIV.2603.03258. URL https://doi.org/10.48550/arXiv.2603.03258.

[428] Mike A. Merrill, Alexander Glenn Shaw, Nicholas Carlini, Boxuan Li, Harsh Raj, Ivan Bercovich, Lin Shi, Jeong Yeon Shin, Thomas Walshe, Estefany Kelly Buchanan, Junhong Shen, Guanghao Ye, Haowei Lin, Jason Poulos, Maoyu Wang, Marianna Nezhurina, Jenia Jitsev, Di Lu, Orfeas Menis-Mastromichalakis, Zhiwei Xu, Zizhao Chen, Yue Liu, Robert Zhang, Leon Liangyu Chen, Anurag Kashyap, Jan-Lucas Uslu, Jeffrey Li, Jianbo Wu, Minghao Yan, Song Bian, Vedang Sharma, Ke Sun, Steven Dillmann, Akshay Anand, Andrew Lanpouthakoun, Bardia Koopah, Changran Hu, Etash Kumar Guha, Gabriel H. S. Dreiman, Jiacheng Zhu, Karl Krauth, Li Zhong, Niklas Muennighoff, Robert Amanfu, Shangyin Tan, Shreyas Pimpalgaonkar, Tushar Aggarwal, Xiangning Lin, Xin Lan, Xuandong Zhao, Yiqing Liang, Yuanli Wang, Zilong Wang, Changzhi Zhou, David Heineman, Hange Liu, Harsh Trivedi, John Yang, Junhong Lin, Manish Shetty, Michael Yang, Nabil Omi, Negin Raoof, Shanda Li, Terry Yue Zhuo, Wuwei Lin, Yiwei Dai, Yuxin Wang, Wenhao Chai, Shang Zhou, Dariush Wahdany, Ziyu She, Jiaming Hu, Zhikang Dong, Yuxuan Zhu, Sasha Cui, Ahson Saiyed, Arinbjörn Kolbeinsson, Jesse Hu, Christopher Michael Rytting, Ryan Marten, Yixin Wang, Alex Dimakis, Andy Konwinski, and Ludwig

Schmidt. Terminal-bench: Benchmarking agents on hard, realistic tasks in command line interfaces. *CoRR*, abs/2601.11868, 2026. doi: 10.48550/ARXIV.2601.11868. URL https://doi.org/10.48550/arXiv.2601.11868.

[429] METR. Time horizon 1.1, 2026. URL https://metr.org/blog/2026-1-29-time-horizon-1-1/.

[430] Qirui Mi, Zhijian Ma, Mengyue Yang, Haoxuan Li, Yisen Wang, Haifeng Zhang, and Jun Wang. Procmem: Learning reusable procedural memory from experience via non-parametric PPO for LLM agents. *CoRR*, abs/2602.01869, 2026. doi: 10.48550/ARXIV.2602.01869. URL https://doi.org/10.48550/arXiv.2602.01869.

[431] Grégoire Mialon, Clémentine Fourrier, Thomas Wolf, Yann LeCun, and Thomas Scialom. GAIA: a benchmark for general AI assistants. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=fibxvahvs3.

[432] Lesly Miculicich, Mihir Parmar, Hamid Palangi, Krishnamurthy Dj Dvijotham, Mirko Montanari, Tomas Pfister, and Long T. Le. Veriguard: Enhancing LLM agent safety via verified code generation. *CoRR*, abs/2510.05156, 2025. doi: 10.48550/ARXIV.2510.05156. URL https://doi.org/10.48550/arXiv.2510.05156.

[433] Sewon Min, Xinxi Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. Rethinking the role of demonstrations: What makes in-context learning work? In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang, editors, *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*, pages 11048–11064. Association for Computational Linguistics, 2022. doi: 10.18653/V1/2022.EMNLP-MAIN.759. URL https://doi.org/10.18653/v1/2022.emnlp-main.759.

[434] Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Wei Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. Factscore: Fine-grained atomic evaluation of factual precision in long form text generation. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 12076–12100. Association for Computational Linguistics, 2023. doi: 10.18653/V1/2023.EMNLP-MAIN.741. URL https://doi.org/10.18653/v1/2023.emnlp-main.741.

[435] MiniMax. Minimax-m1: Scaling test-time compute efficiently with lightning attention. *CoRR*, abs/2506.13585, 2025. doi: 10.48550/ARXIV.2506.13585. URL https://doi.org/10.48550/arXiv.2506.13585.

[436] Kou Misaki, Yuichi Inoue, Yuki Imajuku, So Kuroki, Taishi Nakamura, and Takuya Akiba. Wider or deeper? scaling LLM inference-time compute with adaptive branching tree search. *CoRR*, abs/2503.04412, 2025. doi: 10.48550/ARXIV.2503.04412. URL https://doi.org/10.48550/arXiv.2503.04412.

[437] Samuel Miserendino, Michele Wang, Tejal Patwardhan, and Johannes Heidecke. Swe-lancer: Can frontier llms earn $1 million from real-world freelance software engineering? In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu, editors, *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*, volume 267 of *Proceedings of Machine Learning Research*. PMLR / OpenReview.net, 2025. URL https://proceedings.mlr.press/v267/miserendino25a.html.

[438] Ludovico Mitchener, Angela Yiu, Benjamin Chang, Mathieu Bourdenx, Tyler Nadolski, Arvis Sulovari, Eric C. Landsness, Daniel L. Barabasi, Siddharth Narayanan, Nicky Evans, Shriya Reddy, Martha Foiani, Aizad Kamal, Leah P. Shriver, Fang Cao, Asmamaw T. Wassie, Jon M. Laurent, Edwin Melville-Green, Mayk Caldas Ramos, Albert Bou, Kaleigh F. Roberts, Sladjana Zagorac, Timothy C. Orr, Miranda E. Orr, Kevin J. Zwezdaryk, Ali Essa Ghareeb, Laurie McCoy, Bruna Gomes, Euan A. Ashley, Karen E. Duff, Tonio Buonassisi, Tom Rainforth, Randall J. Bateman, Michael D. Skarlinski, Samuel G. Rodriques, Michaela M. Hinks, and Andrew D. White. Kosmos: An AI scientist for autonomous discovery. *CoRR*, abs/2511.02824, 2025. doi: 10.48550/ARXIV.2511.02824. URL https://doi.org/10.48550/arXiv.2511.02824.

[439] Zhanfeng Mo, Xingxuan Li, Yuntao Chen, and Lidong Bing. Multi-agent tool-integrated policy optimization. *CoRR*, abs/2510.04678, 2025. doi: 10.48550/ARXIV.2510.04678. URL https://doi.org/10.48550/arXiv.2510.04678.

[440] Yutao Mou, Zhangchi Xue, Lijun Li, Peiyang Liu, Shikun Zhang, Wei Ye, and Jing Shao. Toolsafe: Enhancing tool invocation safety of llm-based agents via proactive step-level guardrail and feedback. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens, editors, *Findings of the Association for Computational Linguistics, ACL 2026, San Diego, California, United States, July 2-7, 2026*, pages 37125–37153. Association for Computational Linguistics, 2026. URL https://aclanthology.org/2026.findings-acl.1850/.

[441] Hussein Mozannar, Gagan Bansal, Cheng Tan, Adam Fourney, Victor Dibia, Jingya Chen, Jack Gerrits, Tyler Payne, Matheus Kunzler Maldaner, Madeleine Grunde-McLaughlin, Eric Zhu, Griffin Bassman, Jacob Alber, Peter Chang, Ricky Loynd, Friederike Niedtner, Ece Kamar, Maya Murad, Rafah Hosn, and Saleema Amershi. Magentic-ui: Towards human-in-the-loop agentic systems. *CoRR*, abs/2507.22358, 2025. doi: 10.48550/ARXIV.2507.22358. URL https://doi.org/10.48550/arXiv.2507.22358.

[442] Niklas Muennighoff, Zitong Yang, Weijia Shi, Xiang Lisa Li, Li Fei-Fei, Hannaneh Hajishirzi, Luke Zettlemoyer, Percy Liang, Emmanuel J. Candès, and Tatsunori Hashimoto. s1: Simple test-time scaling. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng, editors, *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, EMNLP 2025, Suzhou, China, November 4-9, 2025*, pages 20275–20321. Association for Computational Linguistics, 2025. doi: 10.18653/V1/2025.EMNLP-MAIN.1025. URL https://doi.org/10.18653/v1/2025.emnlp-main.1025.

[443] Shikhar Murty, Hao Zhu, Dzmitry Bahdanau, and Christopher D. Manning. Nnetnav: Unsupervised learning of browser agents through environment interaction in the wild, 2025. URL https://arxiv.org/abs/2410.02907.

[444] Yohei Nakajima. Babyagi, 2023. URL https://github.com/yoheinakajima/babyagi.

[445] Reiichiro Nakano, Jacob Hilton, Suchir Balaji, Jeff Wu, Long Ouyang, Christina Kim, Christopher Hesse, Shantanu Jain, Vineet Kosaraju, William Saunders, Xu Jiang, Karl Cobbe, Tyna Eloundou, Gretchen Krueger, Kevin Button, Matthew Knight, Benjamin Chess, and John Schulman. Webgpt: Browser-assisted question-answering with human feedback. *CoRR*, abs/2112.09332, 2021. URL https://arxiv.org/abs/2112.09332.

[446] Jingwei Ni, Yihao Liu, Xinpeng Liu, Yutao Sun, Mengyu Zhou, Pengyu Cheng, Dexin Wang, Erchao Zhao, Xiaoxi Jiang, and Guanjun Jiang. Trace2skill: Distill trajectory-local lessons into transferable agent skills. *CoRR*, abs/2603.25158, 2026. doi: 10.48550/ARXIV.2603.25158. URL https://doi.org/10.48550/arXiv.2603.25158.

[447] Kangqi Ni, Wenyue Hua, Xiaoxiang Shi, Jiang Guo, Shiyu Chang, and Tianlong Chen. Chimera: Latency- and performance-aware multi-agent serving for heterogeneous llms. *CoRR*, abs/2603.22206, 2026. doi: 10.48550/ARXIV.2603.22206. URL https://doi.org/10.48550/arXiv.2603.22206.

[448] Alexander Novikov, Ngân Vu, Marvin Eisenberger, Emilien Dupont, Po-Sen Huang, Adam Zsolt Wagner, Sergey Shirobokov, Borislav Kozlovskii, Francisco J. R. Ruiz, Abbas Mehrabian, M. Pawan Kumar, Abigail See, Swarat Chaudhuri, George Holland, Alex Davies, Sebastian Nowozin, Pushmeet Kohli, and Matej Balog. Alphaevolve: A coding agent for scientific and algorithmic discovery. *CoRR*, abs/2506.13131, 2025. doi: 10.48550/ARXIV.2506.13131. URL https://doi.org/10.48550/arXiv.2506.13131.

[449] NVIDIA. World simulation with video foundation models for physical AI. *CoRR*, abs/2511.00062, 2025. doi: 10.48550/ARXIV.2511.00062. URL https://doi.org/10.48550/arXiv.2511.00062.

[450] Maxwell I. Nye, Anders Johan Andreassen, Guy Gur-Ari, Henryk Michalewski, Jacob Austin, David Bieber, David Dohan, Aitor Lewkowycz, Maarten Bosma, David Luan, Charles Sutton, and Augustus Odena. Show your work: Scratchpads for intermediate computation with language models. *CoRR*, abs/2112.00114, 2021. URL https://arxiv.org/abs/2112.00114.

[451] Isaac Ong, Amjad Almahairi, Vincent Wu, Wei-Lin Chiang, Tianhao Wu, Joseph E. Gonzalez, M. Waleed Kadous, and Ion Stoica. Routellm: Learning to route llms with preference data. *CoRR*, abs/2406.18665, 2024. doi: 10.48550/ARXIV.2406.18665. URL https://doi.org/10.48550/arXiv.2406.18665.

[452] OpenAI. Introducing operator, 2025. URL https://openai.com/index/introducing-operator/.

[453] OpenAI. Introducing codex, 2025. URL https://openai.com/index/introducing-codex/.

[454] OpenAI. Hooks | chatgpt learn, 2025. URL https://learn.chatgpt.com/docs/hooks.

[455] OpenAI. From model to agent: Equipping the responses api with a computer environment, 2026. URL https://openai.com/index/equip-responses-api-computer-environment/.

[456] OpenAI. Harness engineering: leveraging codex in an agent-first world, 2026. URL https://openai.com/index/harness-engineering/.

[457] OpenClaw. openclaw: Your own personal ai assistant., 2025. URL https://github.com/openclaw/openclaw.

[458] Krista Opsahl-Ong, Arnav Singhvi, Jasmine Collins, Ivan Zhou, Cindy Wang, Ashutosh Baheti, Owen Oertell, Jacob Portes, Sam Havens, Erich Elsen, Michael Bendersky, Matei Zaharia, and Xing Chen. Officeqa pro: An enterprise benchmark for end-to-end grounded reasoning. *CoRR*, abs/2603.08655, 2026. doi: 10.48550/ARXIV.2603.08655. URL https://doi.org/10.48550/arXiv.2603.08655.

[459] Linke Ouyang, Yuan Qu, Hongbin Zhou, Jiawei Zhu, Rui Zhang, Qunshu Lin, Bin Wang, Zhiyuan Zhao, Man Jiang, Xiaomeng Zhao, Jin Shi, Fan Wu, Pei Chu, Minghao Liu, Zhenxiang Li, Chao Xu, Bo Zhang, Botian Shi, Zhongying Tu, and Conghui He. Omnidocbench: Benchmarking diverse PDF document parsing with comprehensive annotations. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2025, Nashville, TN, USA, June 11-15, 2025*, pages 24838–24848. Computer Vision Foundation / IEEE, 2025. doi: 10.1109/CVPR52734.2025.02313. URL https://openaccess.thecvf.com/content/CVPR2025/html/Ouyang_OmniDocBench_Benchmarking_Diverse_PDF_Document_Parsing_with_Comprehensive_Annotations_CVPR_2025_paper.html.

[460] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022. URL http://papers.nips.cc/paper_files/paper/2022/hash/b1efde53be364a73914f58805a001731-Abstract-Conference.html.

[461] Mingyu Ouyang, Siyuan Hu, Kevin Qinghong Lin, Hwee Tou Ng, and Mike Zheng Shou. Gameworld: Towards standardized and verifiable evaluation of multimodal game agents. *CoRR*, abs/2604.07429, 2026. doi: 10.48550/ARXIV.2604.07429. URL https://doi.org/10.48550/arXiv.2604.07429.

[462] Siru Ouyang, Jun Yan, I-Hung Hsu, Yanfei Chen, Ke Jiang, Zifeng Wang, Rujun Han, Long T. Le, Samira Daruki, Xiangru Tang, Vishy Tirumalashetty, George Lee, Mahsan Rofouei, Hangfei Lin, Jiawei Han, Chen-Yu Lee, and Tomas Pfister. Reasoningbank: Scaling agent self-evolving with reasoning memory. *CoRR*, abs/2509.25140, 2025. doi: 10.48550/ARXIV.2509.25140. URL https://doi.org/10.48550/arXiv.2509.25140.

[463] OWASP GenAI Security Project. Owasp top 10 for agentic applications for 2026, 2025. URL https://genai.owasp.org/resource/owasp-top-10-for-agentic-applications-for-2026/.

[464] Charles Packer, Vivian Fang, Shishir G. Patil, Kevin Lin, Sarah Wooders, and Joseph E. Gonzalez. Memgpt: Towards llms as operating systems. *CoRR*, abs/2310.08560, 2023. doi: 10.48550/ARXIV.2310.08560. URL https://doi.org/10.48550/arXiv.2310.08560.

[465] Piotr Padlewski, Max Bain, Matthew Henderson, Zhongkai Zhu, Nishant Relan, Hai Pham, Donovan Ong, Kaloyan Aleksiev, Aitor Ormazabal, Samuel Phua, Ethan Yeo, Eugenie Lamprecht, Qi Liu, Yuqi Wang, Eric Chen, Deyu Fu, Lei Li, Che Zheng, Cyprien de Masson d'Autume, Dani Yogatama, Mikel Artetxe, and Yi Tay. Vibe-eval: A hard evaluation suite for measuring progress of multimodal language models. *CoRR*, abs/2405.02287, 2024. doi: 10.48550/ARXIV.2405.02287. URL https://doi.org/10.48550/arXiv.2405.02287.

[466] Nils Palumbo, Sarthak Choudhary, Jihye Choi, Guy Amir, Prasad Chalasani, and Somesh Jha. Formal policy enforcement for real-world agentic systems, 2026. URL https://arxiv.org/abs/2602.16708.

[467] Jiayi Pan, Xingyao Wang, Graham Neubig, Navdeep Jaitly, Heng Ji, Alane Suhr, and Yizhe Zhang. Training software engineering agents and verifiers with swe-gym. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu, editors, *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*, volume 267 of *Proceedings of Machine Learning Research*. PMLR / OpenReview.net, 2025. URL https://proceedings.mlr.press/v267/pan25g.html.

[468] Linyue Pan, Lexiao Zou, Shuo Guo, Jingchen Ni, and Haitao Zheng. Natural-language agent harnesses. *CoRR*, abs/2603.25723, 2026. doi: 10.48550/ARXIV.2603.25723. URL https://doi.org/10.48550/arXiv.2603.25723.

[469] Shubham Parashar, Shurui Gui, Xiner Li, Hongyi Ling, Sushil Vemuri, Blake Olson, Eric Li, Yu Zhang, James Caverlee, Dileep Kalathil, and Shuiwang Ji. Curriculum reinforcement learning from easy to hard tasks improves llm reasoning. *CoRR*, abs/2506.06632, 2025. doi: 10.48550/ARXIV.2506.06632. URL https://doi.org/10.48550/arXiv.2506.06632.

[470] Joon Sung Park, Joseph C. O'Brien, Carrie Jun Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. Generative agents: Interactive simulacra of human behavior. In Sean Follmer, Jeff Han, Jürgen Steimle, and Nathalie Henry Riche, editors, *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology, UIST 2023, San Francisco, CA, USA, 29 October 2023- 1 November 2023*, pages 2:1–2:22. ACM, 2023. doi: 10.1145/3586183.3606763. URL https://doi.org/10.1145/3586183.3606763.

[471] Shishir G. Patil, Tianjun Zhang, Xin Wang, and Joseph E. Gonzalez. Gorilla: Large language model connected with massive apis. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang, editors, *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. URL http://papers.nips.cc/paper_files/paper/2024/hash/e4c61f578ff07830f5c37378dd3ecb0d-Abstract-Conference.html.

[472] Shishir G. Patil, Huanzhi Mao, Fanjia Yan, Charlie Cheng-Jie Ji, Vishnu Suresh, Ion Stoica, and Joseph E. Gonzalez. The berkeley function calling leaderboard (BFCL): from tool use to agentic evaluation of large language models. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu, editors, *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*, volume 267 of *Proceedings of Machine Learning Research*. PMLR / OpenReview.net, 2025. URL https://proceedings.mlr.press/v267/patil25a.html.

[473] Badri Narayana Patro and Vijay Srinivas Agneeswaran. Mamba-360: Survey of state space models as transformer alternative for long sequence modelling: Methods, applications, and challenges. *Eng. Appl. Artif. Intell.*, 159: 111279, 2025. doi: 10.1016/J.ENGAPPAI.2025.111279. URL https://doi.org/10.1016/j.engappai.2025.111279.

[474] Tejal Patwardhan, Rachel Dias, Elizabeth Proehl, Grace Kim, Michele Wang, Olivia Watkins, Simón Posada Fishman, Marwan Aljubeh, Phoebe Thacker, Laurance Fauconnet, Natalie S. Kim, Patrick Chao, Samuel Miserendino, Gildas Chabot, David Li, Michael Sharman, Alexandra Barr, Amelia Glaese, and Jerry Tworek. Gdpval: Evaluating ai model performance on real-world economically valuable tasks. *CoRR*, abs/2510.04374, 2025. doi: 10.48550/ARXIV.2510.04374. URL https://doi.org/10.48550/arXiv.2510.04374.

[475] Guilherme Penedo, Hynek Kydlícek, Loubna Ben Allal, Anton Lozhkov, Margaret Mitchell, Colin A. Raffel, Leandro von Werra, and Thomas Wolf. The fineweb datasets: Decanting the web for the finest text data at scale. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang, editors, *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. URL http://papers.nips.cc/paper_files/paper/2024/hash/370df50ccfdf8bde18f8f9c2d9151bda-Abstract-Datasets_and_Benchmarks_Track.html.

[476] Bo Peng, Eric Alcaide, Quentin Anthony, Alon Albalak, Samuel Arcadinho, Stella Biderman, Huanqi Cao, Xin Cheng, Michael Chung, Leon Derczynski, Xingjian Du, Matteo Grella, Kranthi Kiran GV, Xuzheng He, Haowen Hou, Przemyslaw Kazienko, Jan Kocon, Jiaming Kong, Bartlomiej Koptyra, Hayden Lau, Jiaju Lin, Krishna Sri Ipsit Mantri, Ferdinand Mom, Atsushi Saito, Guangyu Song, Xiangru Tang, Johan S. Wind, Stanislaw Wozniak, Zhenyuan Zhang, Qinghua Zhou, Jian Zhu, and Rui-Jie Zhu. RWKV: reinventing rnns for the transformer era. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Findings of the Association for Computational Linguistics: EMNLP 2023, Singapore, December 6-10, 2023*, volume EMNLP 2023 of *Findings of ACL*, pages 14048–14077. Association for Computational Linguistics, 2023. doi: 10.18653/V1/2023.FINDINGS-EMNLP.936. URL https://doi.org/10.18653/v1/2023.findings-emnlp.936.

[477] Bowen Peng, Jeffrey Quesnelle, Honglu Fan, and Enrico Shippole. Yarn: Efficient context window extension of large language models. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=wHBfxhZu1u.

[478] Jiangweizhi Peng, Yuanxin Liu, Ruida Zhou, Charles Fleming, Zhaoran Wang, Alfredo García, and Mingyi Hong. Hiper: Hierarchical reinforcement learning with explicit credit assignment for large language model agents. *CoRR*, abs/2602.16165, 2026. doi: 10.48550/ARXIV.2602.16165. URL https://doi.org/10.48550/arXiv.2602.16165.

[479] Fabio Petroni, Tim Rocktäschel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, Alexander H. Miller, and Sebastian Riedel. Language models as knowledge bases?, 2019. URL https://arxiv.org/abs/1909.01066.

[480] 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. *CoRR*, abs/2506.01062, 2025. doi: 10.48550/ARXIV.2506.01062. URL https://doi.org/10.48550/arXiv.2506.01062.

[481] Julien Pourcel, Cédric Colas, and Pierre-Yves Oudeyer. Self-improving language models for evolutionary program synthesis: A case study on ARC-AGI. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu, editors, *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*, volume 267 of *Proceedings of Machine Learning Research*. PMLR / OpenReview.net, 2025. URL https://proceedings.mlr.press/v267/pourcel25a.html.

[482] Archiki Prasad, Alexander Koller, Mareike Hartmann, Peter Clark, Ashish Sabharwal, Mohit Bansal, and Tushar Khot. Adapt: As-needed decomposition and planning with language models. In Kevin Duh, Helena

Gómez-Adorno, and Steven Bethard, editors, *Findings of the Association for Computational Linguistics: NAACL 2024, Mexico City, Mexico, June 16-21, 2024*, volume NAACL 2024 of *Findings of ACL*, pages 4226–4252. Association for Computational Linguistics, 2024. doi: 10.18653/V1/2024.FINDINGS-NAACL.264. URL https://doi.org/10.18653/v1/2024.findings-naacl.264.

[483] Reid Pryzant, Dan Iter, Jerry Li, Yin Tat Lee, Chenguang Zhu, and Michael Zeng. Automatic prompt optimization with "gradient descent" and beam search. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 7957–7968. Association for Computational Linguistics, 2023. doi: 10.18653/V1/2023.EMNLP-MAIN.494. URL https://doi.org/10.18653/v1/2023.emnlp-main.494.

[484] Zehan Qi, Xiao Liu, Iat Long Iong, Hanyu Lai, Xueqiao Sun, Jiadai Sun, Xinyue Yang, Yu Yang, Shuntian Yao, Wei Xu, Jie Tang, and Yuxiao Dong. Webrl: Training LLM web agents via self-evolving online curriculum reinforcement learning. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=oVKEAFjEqv.

[485] Chen Qian, Wei Liu, Hongzhang Liu, Nuo Chen, Yufan Dang, Jiahao Li, Cheng Yang, Weize Chen, Yusheng Su, Xin Cong, Juyuan Xu, Dahai Li, Zhiyuan Liu, and Maosong Sun. Chatdev: Communicative agents for software development. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pages 15174–15186. Association for Computational Linguistics, 2024. doi: 10.18653/V1/2024.ACL-LONG.810. URL https://doi.org/10.18653/v1/2024.acl-long.810.

[486] Chen Qian, Zihao Xie, Yifei Wang, Wei Liu, Kunlun Zhu, Hanchen Xia, Yufan Dang, Zhuoyun Du, Weize Chen, Cheng Yang, Zhiyuan Liu, and Maosong Sun. Scaling large language model-based multi-agent collaboration. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=K3n5jPkrU6.

[487] Cheng Qian, Emre Can Acikgoz, Qi He, Hongru Wang, Xiusi Chen, Dilek Hakkani-Tur, Gokhan Tur, and Heng Ji. Toolrl: Reward is all tool learning needs. In Danielle Belgrave, Cheng Zhang, Laura N. Montoya, Hsuan-Tien Lin, Razvan Pascanu, Piotr Koniusz, Marzyeh Ghassemi, Nancy Chen, Iván Vladimir Meza Ruíz, and Arturo Loaiza-Bonilla, editors, *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2025, NeurIPS 2025, San Diego, CA, USA, December 2-7, 2025 / Mexico City, Mexico, November 30 - December 5, 2025*, 2025. URL http://papers.nips.cc/paper_files/paper/2025/hash/97c5b2707228e7e3fb67e4ecc2e0e607-Abstract-Conference.html.

[488] Hongjin Qian, Zheng Liu, Peitian Zhang, Kelong Mao, Defu Lian, Zhicheng Dou, and Tiejun Huang. Memorag: Boosting long context processing with global memory-enhanced retrieval augmentation. In *Proceedings of the ACM on Web Conference 2025*, WWW '25, page 2366–2377, New York, NY, USA, 2025. Association for Computing Machinery. ISBN 9798400712746. doi: 10.1145/3696410.3714805. URL https://doi.org/10.1145/3696410.3714805.

[489] Yiyue Qian, Shinan Zhang, Yun Zhou, Haibo Ding, Diego Socolinsky, and Yi Zhang. Collabeval: Enhancing llm-as-a-judge via multi-agent collaboration. *CoRR*, abs/2603.00993, 2026. doi: 10.48550/ARXIV.2603.00993. URL https://doi.org/10.48550/arXiv.2603.00993.

[490] Zile Qiao, Guoxin Chen, Xuanzhong Chen, Donglei Yu, Wenbiao Yin, Xinyu Wang, Zhen Zhang, Baixuan Li, Huifeng Yin, Kuan Li, Rui Min, Minpeng Liao, Yong Jiang, Pengjun Xie, Fei Huang, and Jingren Zhou. Webresearcher: Unleashing unbounded reasoning capability in long-horizon agents. *CoRR*, abs/2509.13309, 2025. doi: 10.48550/ARXIV.2509.13309. URL https://doi.org/10.48550/arXiv.2509.13309.

[491] Tianrui Qin, Qianben Chen, Sinuo Wang, He Xing, King Zhu, He Zhu, Dingfeng Shi, Xinxin Liu, Ge Zhang, Jiaheng Liu, Yuchen Eleanor Jiang, Xitong Gao, and Wangchunshu Zhou. Flash-searcher: Fast and effective web agents via dag-based parallel execution. *CoRR*, abs/2509.25301, 2025. doi: 10.48550/ARXIV.2509.25301. URL https://doi.org/10.48550/arXiv.2509.25301.

[492] 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. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=dHng2O0Jjr.

[493] Yujia Qin, Shengding Hu, Yankai Lin, Weize Chen, Ning Ding, Ganqu Cui, Zheni Zeng, Xuanhe Zhou, Yufei Huang, Chaojun Xiao, Chi Han, Yi R. Fung, Yusheng Su, Huadong Wang, Cheng Qian, Runchu Tian, Kunlun Zhu, Shihao Liang, Xingyu Shen, Bokai Xu, Zhen Zhang, Yining Ye, Bowen Li, Ziwei Tang, Jing Yi, Yuzhang

Zhu, Zhenning Dai, Lan Yan, Xin Cong, Yaxi Lu, Weilin Zhao, Yuxiang Huang, Junxi Yan, Xu Han, Xian Sun, Dahai Li, Jason Phang, Cheng Yang, Tongshuang Wu, Heng Ji, Guoliang Li, Zhiyuan Liu, and Maosong Sun. Tool learning with foundation models. *ACM Comput. Surv.*, 57(4):101:1–101:40, 2025. doi: 10.1145/3704435. URL https://doi.org/10.1145/3704435.

[494] Yujia Qin, Yining Ye, Junjie Fang, Haoming Wang, Shihao Liang, Shizuo Tian, Junda Zhang, Jiahao Li, Yunxin Li, Shijue Huang, Wanjun Zhong, Kuanye Li, Jiale Yang, Yu Miao, Woyu Lin, Longxiang Liu, Xu Jiang, Qianli Ma, Jingyu Li, Xiaojun Xiao, Kai Cai, Chuang Li, Yaowei Zheng, Chaolin Jin, Chen Li, Xiao Zhou, Minchao Wang, Haoli Chen, Zhaojian Li, Haihua Yang, Haifeng Liu, Feng Lin, Tao Peng, Xin Liu, and Guang Shi. UI-TARS: pioneering automated GUI interaction with native agents. *CoRR*, abs/2501.12326, 2025. doi: 10.48550/ARXIV.2501.12326. URL https://doi.org/10.48550/arXiv.2501.12326.

[495] Jiahao Qiu, Xuan Qi, Tongcheng Zhang, Xinzhe Juan, Jiacheng Guo, Yifu Lu, Yimin Wang, Zixin Yao, Qihan Ren, Xun Jiang, Xing Zhou, Dongrui Liu, Ling Yang, Yue Wu, Kaixuan Huang, Shilong Liu, Hongru Wang, and Mengdi Wang. Alita: Generalist agent enabling scalable agentic reasoning with minimal predefinition and maximal self-evolution. *CoRR*, abs/2505.20286, 2025. doi: 10.48550/ARXIV.2505.20286. URL https://doi.org/10.48550/arXiv.2505.20286.

[496] Ao Qu, Han Zheng, Zijian Zhou, Yihao Yan, Yihong Tang, Shao Yong Ong, Fenglu Hong, Kaichen Zhou, Chonghe Jiang, Minwei Kong, Jiacheng Zhu, Xuan Jiang, Sirui Li, Cathy Wu, Bryan Kian Hsiang Low, Jinhua Zhao, and Paul Pu Liang. CORAL: towards autonomous multi-agent evolution for open-ended discovery. *CoRR*, abs/2604.01658, 2026. doi: 10.48550/ARXIV.2604.01658. URL https://doi.org/10.48550/arXiv.2604.01658.

[497] Changle Qu, Sunhao Dai, Xiaochi Wei, Hengyi Cai, Shuaiqiang Wang, Dawei Yin, Jun Xu, and Ji-Rong Wen. Tool learning with large language models: a survey. *Frontiers Comput. Sci.*, 19(8):198343, 2025. doi: 10.1007/S11704-024-40678-2. URL https://doi.org/10.1007/s11704-024-40678-2.

[498] Qwen Team. Qwen3.5-Omni technical report. *CoRR*, abs/2604.15804, 2026. doi: 10.48550/ARXIV.2604.15804. URL https://doi.org/10.48550/arXiv.2604.15804.

[499] Mohit Raghavendra, Soham Dan, Miguel Romero Calvo, Yannis Yiming He, Johannes Baptist Mols, Gautam Anand, Cole McCollum, Edgar Arakelyan, Vijay Bharadwaj, Andrew Park, Jeff Da, MohammadHossein Rezaei, Bing Liu, Brad Kenstler, and Yunzhong He. SWE atlas: Benchmarking coding agents beyond issue resolution. *CoRR*, abs/2605.08366, 2026. doi: 10.48550/ARXIV.2605.08366. URL https://doi.org/10.48550/arXiv.2605.08366.

[500] Mohit Raghavendra, Anisha Gunjal, Bing Liu, and Yunzhong He. Agentic rubrics as contextual verifiers for SWE agents. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens, editors, *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2026, San Diego, California, United States, July 2-7, 2026*, pages 15265–15290. Association for Computational Linguistics, 2026. URL https://aclanthology.org/2026.acl-long.697/.

[501] Lokman Rahmani, David Minarsch, and Jonathan Ward. Peer-to-peer autonomous agent communication network. In Frank Dignum, Alessio Lomuscio, Ulle Endriss, and Ann Nowé, editors, *AAMAS '21: 20th International Conference on Autonomous Agents and Multiagent Systems, Virtual Event, United Kingdom, May 3-7, 2021*, pages 1037–1045. ACM, 2021. doi: 10.5555/3463952.3464073. URL https://www.ifaamas.org/Proceedings/aamas2021/pdfs/p1037.pdf.

[502] Pooyan Rahmanzadehgervi, Logan Bolton, Mohammad Reza Taesiri, and Anh Totti Nguyen. Vision language models are blind: Failing to translate detailed visual features into words, 2025. URL https://arxiv.org/abs/2407.06581.

[503] Mohan Rajagopalan and Vinay Rao. Authenticated workflows: A systems approach to protecting agentic AI. *CoRR*, abs/2602.10465, 2026. doi: 10.48550/ARXIV.2602.10465. URL https://doi.org/10.48550/arXiv.2602.10465.

[504] Rajesh Ranjan, Shailja Gupta, and Surya Narayan Singh. LOKA protocol: A decentralized framework for trustworthy and ethical AI agent ecosystems. *CoRR*, abs/2504.10915, 2025. doi: 10.48550/ARXIV.2504.10915. URL https://doi.org/10.48550/arXiv.2504.10915.

[505] Ben Rank, Hardik Bhatnagar, Ameya Prabhu, Shira Eisenberg, Karina Nguyen, Matthias Bethge, and Maksym Andriushchenko. Posttrainbench: Can LLM agents automate LLM post-training? *CoRR*, abs/2603.08640, 2026. doi: 10.48550/ARXIV.2603.08640. URL https://doi.org/10.48550/arXiv.2603.08640.

[506] Tabish Rashid, Mikayel Samvelyan, Christian Schröder de Witt, Gregory Farquhar, Jakob N. Foerster, and Shimon Whiteson. QMIX: monotonic value function factorisation for deep multi-agent reinforcement learning. In Jennifer G. Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pages 4292–4301. PMLR, 2018. URL http://proceedings.mlr.press/v80/rashid18a.html.

[507] Preston Rasmussen, Pavlo Paliychuk, Travis Beauvais, Jack Ryan, and Daniel Chalef. Zep: A temporal knowledge graph architecture for agent memory. *CoRR*, abs/2501.13956, 2025. doi: 10.48550/ARXIV.2501.13956. URL https://doi.org/10.48550/arXiv.2501.13956.

[508] Christopher Rawles, Sarah Clinckemaillie, Yifan Chang, Jonathan Waltz, Gabrielle Lau, Marybeth Fair, Alice Li, William E. Bishop, Wei Li, Folawiyo Campbell-Ajala, Daniel Kenji Toyama, Robert James Berry, Divya Tyamagundlu, Timothy P. Lillicrap, and Oriana Riva. Androidworld: A dynamic benchmarking environment for autonomous agents. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=il5yUQsrjC.

[509] Traian Rebedea, Razvan Dinu, Makesh Narsimhan Sreedhar, Christopher Parisien, and Jonathan Cohen. Nemo guardrails: A toolkit for controllable and safe LLM applications with programmable rails. In Yansong Feng and Els Lefever, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023 - System Demonstrations, Singapore, December 6-10, 2023*, pages 431–445. Association for Computational Linguistics, 2023. doi: 10.18653/V1/2023.EMNLP-DEMO.40. URL https://doi.org/10.18653/v1/2023.emnlp-demo.40.

[510] Xubin Ren, Lingrui Xu, Long Xia, Shuaiqiang Wang, Dawei Yin, and Chao Huang. Videorag: Retrieval-augmented generation with extreme long-context videos. In Srinivasan Parthasarathy, David F. Gleich, Xiangliang Zhang, Wee Hyong Tok, Faisal Farooq, Qi He, Ambuj K. Singh, Haixun Wang, and Yan Liu, editors, *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.1, KDD 2026, Jeju Island, Korea, August 9-13, 2026*, pages 2390–2401. ACM, 2026. doi: 10.1145/3770854.3783944. URL https://doi.org/10.1145/3770854.3783944.

[511] Chroma Research. Context rot: When long contexts hurt llm performance, 2025. URL https://www.trychroma.com/research/context-rot.

[512] IBM Research. Agent communication protocol (acp), 2025. URL https://research.ibm.com/projects/agent-communication-protocol.

[513] Nous Research. hermes-agent: The agent that grows with you, 2025. URL https://github.com/nousresearch/hermes-agent.

[514] Laria Reynolds and Kyle McDonell. Prompt programming for large language models: Beyond the few-shot paradigm. In Yoshifumi Kitamura, Aaron Quigley, Katherine Isbister, and Takeo Igarashi, editors, *CHI '21: CHI Conference on Human Factors in Computing Systems, Virtual Event / Yokohama Japan, May 8-13, 2021, Extended Abstracts*, pages 314:1–314:7. ACM, 2021. doi: 10.1145/3411763.3451760. URL https://doi.org/10.1145/3411763.3451760.

[515] Yeonju Ro, Haoran Qiu, Íñigo Goiri, Rodrigo Fonseca, Ricardo Bianchini, Aditya Akella, Zhangyang Wang, Mattan Erez, and Esha Choukse. Sherlock: Reliable and efficient agentic workflow execution. *CoRR*, abs/2511.00330, 2025. doi: 10.48550/ARXIV.2511.00330. URL https://doi.org/10.48550/arXiv.2511.00330.

[516] Adam Roberts, Colin Raffel, and Noam Shazeer. How much knowledge can you pack into the parameters of a language model? In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 5418–5426. Association for Computational Linguistics, 2020. doi: 10.18653/V1/2020.EMNLP-MAIN.437. URL https://doi.org/10.18653/v1/2020.emnlp-main.437.

[517] Maxime Robeyns, Martin Szummer, and Laurence Aitchison. A self-improving coding agent. *CoRR*, abs/2504.15228, 2025. doi: 10.48550/ARXIV.2504.15228. URL https://doi.org/10.48550/arXiv.2504.15228.

[518] Bojie Rong, Zheyu Shen, Qiaoping Wang, Pengfei Kang, Yang Xu, Yawen Wei, Hanyu Wu, Zhi Zhao, Leihao Pei, and Linquan Jiang. Aliyunconsoleagent: Training web agents in real-world cloud environments via distillation and reinforcement learning. *CoRR*, abs/2606.09447, 2026. doi: 10.48550/ARXIV.2606.09447. URL https://doi.org/10.48550/arXiv.2606.09447.

[519] Jianhao Ruan, Zhihao Xu, Yiran Peng, Fashen Ren, Zhaoyang Yu, Xinbing Liang, Jinyu Xiang, Yongru Chen, Bang Liu, Chenglin Wu, Yuyu Luo, and Jiayi Zhang. Aorchestra: Automating sub-agent creation for agentic orchestration. *CoRR*, abs/2602.03786, 2026. doi: 10.48550/ARXIV.2602.03786. URL https://doi.org/10.48550/arXiv.2602.03786.

[520] Yangjun Ruan, Honghua Dong, Andrew Wang, Silviu Pitis, Yongchao Zhou, Jimmy Ba, Yann Dubois, Chris J. Maddison, and Tatsunori Hashimoto. Identifying the risks of LM agents with an lm-emulated sandbox. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=GEcwtMk1uA.

[521] Kusha Sareen, Morgane M. Moss, Alessandro Sordoni, Rishabh Agarwal, and Arian Hosseini. Putting the value back in RL: better test-time scaling by unifying LLM reasoners with verifiers. *CoRR*, abs/2505.04842, 2025. doi: 10.48550/ARXIV.2505.04842. URL https://doi.org/10.48550/arXiv.2505.04842.

[522] Parth Sarthi, Salman Abdullah, Aditi Tuli, Shubh Khanna, Anna Goldie, and Christopher D. Manning. RAPTOR: recursive abstractive processing for tree-organized retrieval. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=GN921JHCRw.

[523] Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language models can teach themselves to use tools. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine, editors, *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/d842425e4bf79ba039352da0f658a906-Abstract-Conference.html.

[524] John Schulman, Philipp Moritz, Sergey Levine, Michael I. Jordan, and Pieter Abbeel. High-dimensional continuous control using generalized advantage estimation. In Yoshua Bengio and Yann LeCun, editors, *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*, 2016. URL http://arxiv.org/abs/1506.02438.

[525] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *CoRR*, abs/1707.06347, 2017. URL http://arxiv.org/abs/1707.06347.

[526] ByteDance Seed. Workflow-gym: Towards long-horizon evaluation of computer-use agentic tasks in real-world professional fields. *CoRR*, abs/2606.11042, 2026. doi: 10.48550/ARXIV.2606.11042. URL https://doi.org/10.48550/arXiv.2606.11042.

[527] Bytedance Seed. Seed2.0 model card: Towards intelligence frontier for real-world complexity, 2026. URL https://arxiv.org/abs/2607.00248.

[528] Seedance Team. Seedance 2.0: Advancing video generation for world complexity. *arXiv preprint arXiv:2604.14148*, 2026.

[529] Md. Shamsujjoha, Qinghua Lu, Dehai Zhao, and Liming Zhu. Swiss cheese model for AI safety: A taxonomy and reference architecture for multi-layered guardrails of foundation model based agents. In *22nd IEEE International Conference on Software Architecture, ICSA 2025, Odense, Denmark, March 31 - April 4, 2025*, pages 37–48. IEEE, 2025. doi: 10.1109/ICSA65012.2025.00014. URL https://doi.org/10.1109/ICSA65012.2025.00014.

[530] Jiaqi Shao, Yufeng Miao, Wei Zhang, and Bing Luo. Foldact: Efficient and stable context folding for long-horizon search agents. *CoRR*, abs/2512.22733, 2025. doi: 10.48550/ARXIV.2512.22733. URL https://doi.org/10.48550/arXiv.2512.22733.

[531] Rui Shao, Ruize Gao, Bin Xie, Yixing Li, Kaiwen Zhou, Shuai Wang, Weili Guan, and Gongwei Chen. HATS: hardness-aware trajectory synthesis for GUI agents. *CoRR*, abs/2603.12138, 2026. doi: 10.48550/ARXIV.2603.12138. URL https://doi.org/10.48550/arXiv.2603.12138.

[532] Rulin Shao, Akari Asai, Shannon Zejiang Shen, Hamish Ivison, Varsha Kishore, Jingming Zhuo, Xinran Zhao, Molly Park, Samuel G. Finlayson, David A. Sontag, Tyler Murray, Sewon Min, Pradeep Dasigi, Luca Soldaini, Faeze Brahman, Wen-tau Yih, Tongshuang Wu, Luke Zettlemoyer, Yoon Kim, Hannaneh Hajishirzi, and Pang Wei Koh. DR tulu: Reinforcement learning with evolving rubrics for deep research. *CoRR*, abs/2511.19399, 2025. doi: 10.48550/ARXIV.2511.19399. URL https://doi.org/10.48550/arXiv.2511.19399.

[533] Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *CoRR*, abs/2402.03300, 2024. doi: 10.48550/ARXIV.2402.03300. URL https://doi.org/10.48550/arXiv.2402.03300.

[534] Manasi Sharma, Chen Bo Calvin Zhang, Chaithanya Bandi, Clinton Wang, Ankit Aich, Huy Nghiem, Tahseen Rabbani, Ye Htet, Brian Jang, Sumana Basu, Aishwarya Balwani, Denis Peskoff, Marcos Ayestaran, Sean M. Hendryx, Brad Kenstler, and Bing Liu. Researchrubrics: A benchmark of prompts and rubrics for evaluating deep research agents. *CoRR*, abs/2511.07685, 2025. doi: 10.48550/ARXIV.2511.07685. URL https://doi.org/10.48550/arXiv.2511.07685.

[535] Noam Shazeer. Fast transformer decoding: One write-head is all you need. *CoRR*, abs/1911.02150, 2019. URL http://arxiv.org/abs/1911.02150.

[536] Han Shen. On entropy control in LLM-RL algorithms. *CoRR*, abs/2509.03493, 2025. doi: 10.48550/ARXIV.2509.03493. URL https://doi.org/10.48550/arXiv.2509.03493.

[537] Yongliang Shen, Kaitao Song, Xu Tan, Dongsheng Li, Weiming Lu, and Yueting Zhuang. Hugginggpt: Solving AI tasks with chatgpt and its friends in hugging face. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine, editors, *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/77c33e6a367922d003ff102ffb92b658-Abstract-Conference.html.

[538] Yujiong Shen, Yajie Yang, Zhiheng Xi, Binze Hu, Huayu Sha, Jiazheng Zhang, Qiyuan Peng, Junlin Shang, Jixuan Huang, Yutao Fan, Jingqi Tong, Shihan Dou, Ming Zhang, Lei Bai, Zhenfei Yin, Tao Gui, Xingjun Ma, Qi Zhang, Xuanjing Huang, and Yu-Gang Jiang. Sciagentgym: Benchmarking multi-step scientific tool-use in LLM agents. *CoRR*, abs/2602.12984, 2026. doi: 10.48550/ARXIV.2602.12984. URL https://doi.org/10.48550/arXiv.2602.12984.

[539] Daniel Shepard and Robin Salimans. Automationbench. *CoRR*, abs/2604.18934, 2026. doi: 10.48550/ARXIV.2604.18934. URL https://doi.org/10.48550/arXiv.2604.18934.

[540] Dingfeng Shi, Jingyi Cao, Qianben Chen, Weichen Sun, Weizhen Li, Hongxuan Lu, Fangchen Dong, Tianrui Qin, King Zhu, Minghao Liu, Jian Yang, Ge Zhang, Jiaheng Liu, Changwang Zhang, Jun Wang, Yuchen Eleanor Jiang, and Wangchunshu Zhou. Taskcraft: Automated generation of agentic tasks. *CoRR*, abs/2506.10055, 2025. doi: 10.48550/ARXIV.2506.10055. URL https://doi.org/10.48550/arXiv.2506.10055.

[541] Freda Shi, Xinyun Chen, Kanishka Misra, Nathan Scales, David Dohan, Ed H. Chi, Nathanael Schärli, and Denny Zhou. Large language models can be easily distracted by irrelevant context. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett, editors, *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pages 31210–31227. PMLR, 2023. URL https://proceedings.mlr.press/v202/shi23a.html.

[542] Tianneng Shi, Jingxuan He, Zhun Wang, Hongwei Li, Linyu Wu, Wenbo Guo, and Dawn Song. Progent: Securing ai agents with privilege control, 2026. URL https://arxiv.org/abs/2504.11703.

[543] Yucheng Shi, Wenhao Yu, Zaitang Li, Yonglin Wang, Hongming Zhang, Ninghao Liu, Haitao Mi, and Dong Yu. Mobilegui-rl: Advancing mobile GUI agent through reinforcement learning in online environment. *CoRR*, abs/2507.05720, 2025. doi: 10.48550/ARXIV.2507.05720. URL https://doi.org/10.48550/arXiv.2507.05720.

[544] Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: language agents with verbal reinforcement learning. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine, editors, *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/1b44b878bb782e6954cd888628510e90-Abstract-Conference.html.

[545] Mohit Shridhar, Xingdi Yuan, Marc-Alexandre Côté, Yonatan Bisk, Adam Trischler, and Matthew J. Hausknecht. Alfworld: Aligning text and embodied environments for interactive learning. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. URL https://openreview.net/forum?id=0IOX0YcCdTn.

[546] KaShun Shum, Binyuan Hui, Jiawei Chen, Lei Zhang, X. W., Jiaxi Yang, Yuzhen Huang, Junyang Lin, and Junxian He. SWE-RM: execution-free feedback for software engineering agents. *CoRR*, abs/2512.21919, 2025. doi: 10.48550/ARXIV.2512.21919. URL https://doi.org/10.48550/arXiv.2512.21919.

[547] Akshey Sigdel and Rista Baral. Guardrails as infrastructure: Policy-first control for tool-orchestrated workflows. *CoRR*, abs/2603.18059, 2026. doi: 10.48550/ARXIV.2603.18059. URL https://doi.org/10.48550/arXiv.2603.18059.

[548] Akshey Sigdel and Rista Baral. Schema first tool apis for LLM agents: A controlled study of tool misuse, recovery, and budgeted performance. *CoRR*, abs/2603.13404, 2026. doi: 10.48550/ARXIV.2603.13404. URL https://doi.org/10.48550/arXiv.2603.13404.

[549] Akshit Sinha, Arvindh Arun, Shashwat Goel, Steffen Staab, and Jonas Geiping. The illusion of diminishing returns: Measuring long horizon execution in LLMs. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=3lm8lWYxiq.

[550] Charlie Victor Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. Scaling LLM test-time compute optimally can be more effective than scaling parameters for reasoning. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=4FWAwZtd2n.

[551] Theta Software. Computer use benchmark (cub), 2025. URL https://thetasoftware.com/blog/introducing-cub/.

[552] Luca Soldaini, Rodney Kinney, Akshita Bhagia, Dustin Schwenk, David Atkinson, Russell Authur, Ben Bogin, Khyathi Raghavi Chandu, Jennifer Dumas, Yanai Elazar, Valentin Hofmann, Ananya Harsh Jha, Sachin Kumar, Li Lucy, Xinxi Lyu, Nathan Lambert, Ian Magnusson, Jacob Morrison, Niklas Muennighoff, Aakanksha Naik, Crystal Nam, Matthew E. Peters, Abhilasha Ravichander, Kyle Richardson, Zejiang Shen, Emma Strubell, Nishant Subramani, Oyvind Tafjord, Pete Walsh, Luke Zettlemoyer, Noah A. Smith, Hannaneh Hajishirzi, Iz Beltagy, Dirk Groeneveld, Jesse Dodge, and Kyle Lo. Dolma: an open corpus of three trillion tokens for language model pretraining research. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pages 15725–15788. Association for Computational Linguistics, 2024. doi: 10.18653/V1/2024.ACL-LONG.840. URL https://doi.org/10.18653/v1/2024.acl-long.840.

[553] 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. *CoRR*, abs/2503.05592, 2025. doi: 10.48550/ARXIV.2503.05592. URL https://doi.org/10.48550/arXiv.2503.05592.

[554] Siyao Song, Cong Ma, Zhihao Cheng, Shiye Lei, Minghao Li, Ying Zeng, Huaixiao Tou, and Kai Jia. EAPO: enhancing policy optimization with on-demand expert assistance. *CoRR*, abs/2509.23730, 2025. doi: 10.48550/ARXIV.2509.23730. URL https://doi.org/10.48550/arXiv.2509.23730.

[555] Xiaoshuai Song, Haofei Chang, Guanting Dong, Yutao Zhu, Ji-Rong Wen, and Zhicheng Dou. Envscaler: Scaling tool-interactive environments for LLM agent via programmatic synthesis. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens, editors, *Findings of the Association for Computational Linguistics, ACL 2026, San Diego, California, United States, July 2-7, 2026*, pages 8326–8357. Association for Computational Linguistics, 2026. URL https://aclanthology.org/2026.findings-acl.407/.

[556] Xiaoshuai Song, Liancheng Zhang, Kangzhi Zhao, Yutao Zhu, Zhongyuan Wang, Guanting Dong, Jinghan Yang, Han Li, Kun Gai, Ji-Rong Wen, and Zhicheng Dou. Webswarm: Recursive multi-agent orchestration for deep-and-wide web search, 2026. URL https://arxiv.org/abs/2607.08662.

[557] Yifan Song, Weimin Xiong, Xiutian Zhao, Dawei Zhu, Wenhao Wu, Ke Wang, Cheng Li, Wei Peng, and Sujian Li. Agentbank: Towards generalized LLM agents via fine-tuning on 50000+ interaction trajectories. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen, editors, *Findings of the Association for Computational Linguistics: EMNLP 2024, Miami, Florida, USA, November 12-16, 2024*, volume EMNLP 2024 of *Findings of ACL*, pages 2124–2141. Association for Computational Linguistics, 2024. doi: 10.18653/V1/2024.FINDINGS-EMNLP.116. URL https://doi.org/10.18653/v1/2024.findings-emnlp.116.

[558] Yueqi Song, Ketan Ramaneti, Zaid Sheikh, Ziru Chen, Boyu Gou, Tianbao Xie, Yiheng Xu, Danyang Zhang, Apurva Gandhi, Fan Yang, Joseph Liu, Tianyue Ou, Zhihao Yuan, Frank Xu, Shuyan Zhou, Xingyao Wang, Xiang Yue, Tao Yu, Huan Sun, Yu Su, and Graham Neubig. Agent data protocol: Unifying datasets for diverse, effective fine-tuning of LLM agents. *CoRR*, abs/2510.24702, 2025. doi: 10.48550/ARXIV.2510.24702. URL https://doi.org/10.48550/arXiv.2510.24702.

[559] Yuxuan Song, Zheng Zhang, Cheng Luo, Pengyang Gao, Fan Xia, Hao Luo, Zheng Li, Yuehang Yang, Hongli Yu, Xingwei Qu, Yuwei Fu, Jing Su, Ge Zhang, Wenhao Huang, Mingxuan Wang, Lin Yan, Xiaoying Jia, Jingjing Liu, Wei-Ying Ma, Ya-Qin Zhang, Yonghui Wu, and Hao Zhou. Seed diffusion: A large-scale diffusion language model with high-speed inference. *CoRR*, abs/2508.02193, 2025. doi: 10.48550/ARXIV.2508.02193. URL https://doi.org/10.48550/arXiv.2508.02193.

[560] Giulio Starace, Oliver Jaffe, Dane Sherburn, James Aung, Jun Shern Chan, Leon Maksin, Rachel Dias, Evan Mays, Benjamin Kinsella, Wyatt Thompson, Johannes Heidecke, Amelia Glaese, and Tejal Patwardhan. Paperbench:

Evaluating ai's ability to replicate AI research. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu, editors, *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*, volume 267 of *Proceedings of Machine Learning Research*. PMLR / OpenReview.net, 2025. URL https://proceedings.mlr.press/v267/starace25a.html.

[561] Stripe. Minions: Stripe's one-shot, end-to-end coding agents, 2026. URL https://stripe.dev/blog/minions-stripes-one-shot-end-to-end-coding-agents.

[562] Hongjin Su, Ruoxi Sun, Jinsung Yoon, Pengcheng Yin, Tao Yu, and Sercan Ö. Arik. Learn-by-interact: A data-centric framework for self-adaptive agents in realistic environments. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=3UKOzGWCVY.

[563] Yiming Su, Kunzhao Xu, Yanjie Gao, Fan Yang, Cheng Liu, Mao Yang, and Tianyin Xu. Neuro-symbolic verification on instruction following of llms. *CoRR*, abs/2601.17789, 2026. doi: 10.48550/ARXIV.2601.17789. URL https://doi.org/10.48550/arXiv.2601.17789.

[564] Zhaochen Su, Jincheng Gao, Hangyu Guo, Zhenhua Liu, Lueyang Zhang, Xinyu Geng, Shijue Huang, Peng Xia, Guanyu Jiang, Cheng Wang, Yue Zhang, Yi R. (May) Fung, and Junxian He. Agentvista: Evaluating multimodal agents in ultra-challenging realistic visual scenarios. *CoRR*, abs/2602.23166, 2026. doi: 10.48550/ARXIV.2602.23166. URL https://doi.org/10.48550/arXiv.2602.23166.

[565] Zhenpeng Su, Leiyu Pan, Minxuan Lv, Yuntao Li, Wenping Hu, Fuzheng Zhang, Kun Gai, and Guorui Zhou. CE-GPPO: coordinating entropy via gradient-preserving clipping policy optimization in reinforcement learning. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens, editors, *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2026, San Diego, California, United States, July 2-7, 2026*, pages 37747–37760. Association for Computational Linguistics, 2026. URL https://aclanthology.org/2026.acl-long.1752/.

[566] Shreyas Subramanian, Adewale Akinfaderin, Yanyan Zhang, Ishan Singh, Mani Khanuja, Sandeep Singh, and Maira Ladeira Tanke. Keyword search is all you need: Achieving rag-level performance without vector databases using agentic tool use. *CoRR*, abs/2602.23368, 2026. doi: 10.48550/ARXIV.2602.23368. URL https://doi.org/10.48550/arXiv.2602.23368.

[567] Theodore R. Sumers, Shunyu Yao, Karthik Narasimhan, and Thomas L. Griffiths. Cognitive architectures for language agents. *Trans. Mach. Learn. Res.*, 2024, 2024. URL https://openreview.net/forum?id=1i6ZCvflQJ.

[568] Haoran Sun, Shaoning Zeng, and Bob Zhang. H-MEM: Hierarchical memory for high-efficiency long-term reasoning in LLM agents. In Vera Demberg, Kentaro Inui, and Lluís Marquez, editors, *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 341–350, Rabat, Morocco, March 2026. Association for Computational Linguistics. ISBN 979-8-89176-380-7. doi: 10.18653/v1/2026.eacl-long.15. URL https://aclanthology.org/2026.eacl-long.15/.

[569] Haotian Sun, Yuchen Zhuang, Lingkai Kong, Bo Dai, and Chao Zhang. Adaplanner: Adaptive planning from feedback with language models. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine, editors, *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/b5c8c1c117618267944b2617add0a766-Abstract-Conference.html.

[570] Jiahui Sun, Zhichao Hua, and Yubin Xia. Autoeval: A practical framework for autonomous evaluation of mobile agents. *CoRR*, abs/2503.02403, 2025. doi: 10.48550/ARXIV.2503.02403. URL https://doi.org/10.48550/arXiv.2503.02403.

[571] Qiushi Sun, Mukai Li, Zhoumianze Liu, Zhihui Xie, Fangzhi Xu, Zhangyue Yin, Kanzhi Cheng, Zehao Li, Zichen Ding, Qi Liu, Zhiyong Wu, Zhuosheng Zhang, Ben Kao, and Lingpeng Kong. Os-sentinel: Towards safety-enhanced mobile GUI agents via hybrid validation in realistic workflows. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens, editors, *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2026, San Diego, California, United States, July 2-7, 2026*, pages 9529–9553. Association for Computational Linguistics, 2026. URL https://aclanthology.org/2026.acl-long.431/.

[572] Shuang Sun, Huatong Song, Lisheng Huang, Jinhao Jiang, Ran Le, Zhihao Lv, Zongchao Chen, Yiwen Hu, Wenyang Luo, Wayne Xin Zhao, Yang Song, Hongteng Xu, Tao Zhang, and Ji-Rong Wen. Swe-world: Building

software engineering agents in docker-free environments. *CoRR*, abs/2602.03419, 2026. doi: 10.48550/ARXIV.2602.03419. URL https://doi.org/10.48550/arXiv.2602.03419.

[573] Weiwei Sun, Miao Lu, Zhan Ling, Kang Liu, Xuesong Yao, Yiming Yang, and Jiecao Chen. Scaling long-horizon LLM agent via context-folding. *CoRR*, abs/2510.11967, 2025. doi: 10.48550/ARXIV.2510.11967. URL https://doi.org/10.48550/arXiv.2510.11967.

[574] Wenzhang Sun, Zhenyu Wang, Zhangchi Hu, Chunfeng Wang, Hao Li, and Wei Chen. MUSE: A multi-agent framework for unconstrained story envisioning via closed-loop cognitive orchestration. *CoRR*, abs/2602.03028, 2026. doi: 10.48550/ARXIV.2602.03028. URL https://doi.org/10.48550/arXiv.2602.03028.

[575] Yiyou Sun, Xinyang Han, Weichen Zhang, Yuanbo Pang, Tianyu Wang, Yuhan Cao, Yixiao Huang, Chris Duroiu, Haoyun Zhang, Jeffrey Lin, Weishu Zhang, Tyler Zeng, Ying Yan, Bo Liu, Hanson Wen, Mingyang Xu, Xiaoyuan Liu, Zimeng Chen, Weiyan Shi, Amanda Dsouza, Vincent Sunn Chen, Patrick Bryant, Carl Boettiger, Yamini Rangan, Bradley Rothenberg, Kyle Steinfeld, Arvind Rao, Tapio Schneider, Georgios Yannakakis, Laure Zanna, Kaan Ozbay, Ida Sim, Tarek Zohdi, George Em Karniadakis, Jack Gallant, Teresa Head-Gordon, Yushan Li, Wenxi Deng, Tao Sun, Huiqi Wang, Zhun Wang, Justin Xu, Chris Yuhao Liu, Yafei Cheng, Rongwang Hu, Aras Bacho, Shengcao Cao, Zengyi Qin, Yixiong Chen, Hengduan Fan, Hao Liu, Lin Zeng, Shashank Muralidhar Bharadwaj, Litian Gong, Yingxuan Yang, Maojia Song, Ruheng Wang, Zongzheng Zhang, Honglin Bao, Shuo Lu, Jianhong Tu, Zhonghua Wang, Zheng Zhang, Zijiao Chen, Yanqiong Jiang, Zhendong Li, Bohan Lyu, Chang Ma, Peiran Xu, Benran Zhang, Shangding Gu, Haoyue Hua, Haoyang Li, Wanzhe Liao, Chengzhi Liu, Junbo Peng, Haoran Sun, Zechen Xu, Bo Chen, Jiayi Cheng, Yi Jiang, Keying Kuang, Yuan Li, Youbang Pan, Ziyan Rao, Alexander Schubert, Yifan Shen, Vincent Siu, Xiatao Sun, Kangqi Zhang, Xiaopan Zhang, Yuchen Zhu, Ishaan Singh Chandok, Lei Ding, Jingxuan Fan, Andrew Glover, Jiaming Hu, Yiran Hu, Wenbo Huang, and et al. Agents' last exam. *CoRR*, abs/2606.05405, 2026. doi: 10.48550/ARXIV.2606.05405. URL https://doi.org/10.48550/arXiv.2606.05405.

[576] Yutao Sun, Li Dong, Shaohan Huang, Shuming Ma, Yuqing Xia, Jilong Xue, Jianyong Wang, and Furu Wei. Retentive network: A successor to transformer for large language models. *CoRR*, abs/2307.08621, 2023. doi: 10.48550/ARXIV.2307.08621. URL https://doi.org/10.48550/arXiv.2307.08621.

[577] Zeyi Sun, Ziyu Liu, Yuhang Zang, Yuhang Cao, Xiaoyi Dong, Tong Wu, Dahua Lin, and Jiaqi Wang. Seagent: Self-evolving computer use agent with autonomous learning from experience. *CoRR*, abs/2508.04700, 2025. doi: 10.48550/ARXIV.2508.04700. URL https://doi.org/10.48550/arXiv.2508.04700.

[578] Lintang Sutawika, Aditya Bharat Soni, Bharath Sriraam R. R, Apurva Gandhi, Taha Yassine, Sanidhya Vijayvargiya, Yuchen Li, Xuhui Zhou, Yilin Zhang, Leander Melroy Maben, and Graham Neubig. Codescout: An effective recipe for reinforcement learning of code search agents. *CoRR*, abs/2603.17829, 2026. doi: 10.48550/ARXIV.2603.17829. URL https://doi.org/10.48550/arXiv.2603.17829.

[579] Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, Cambridge, MA, 2nd edition, 2018.

[580] Richard S. Sutton and David Silver. Welcome to the Era of Experience, April 2025.

[581] Kyle Swanson, Wesley Wu, Nash L. Bulaong, John E. Pak, and James Zou. The virtual lab of AI agents designs new sars-cov-2 nanobodies. *Nat.*, 646(8085):716–723, 2025. doi: 10.1038/S41586-025-09442-9. URL https://doi.org/10.1038/s41586-025-09442-9.

[582] Hui-Ze Tan, Xiao-Wen Yang, Hao Chen, Jie-Jing Shao, Yi Wen, Yuteng Shen, Weihong Luo, Xiku Du, Lan-Zhe Guo, and Yu-Feng Li. Hindsight credit assignment for long-horizon LLM agents. *CoRR*, abs/2603.08754, 2026. doi: 10.48550/ARXIV.2603.08754. URL https://doi.org/10.48550/arXiv.2603.08754.

[583] Jiejun Tan, Zhicheng Dou, Liancheng Zhang, Yuyang Hu, Yiruo Cheng, and Ji-Rong Wen. Memsifter: Offloading LLM memory retrieval via outcome-driven proxy reasoning. *CoRR*, abs/2603.03379, 2026. doi: 10.48550/ARXIV.2603.03379. URL https://doi.org/10.48550/arXiv.2603.03379.

[584] Jingqun Tang, Qi Liu, Yongjie Ye, Jinghui Lu, Shu Wei, An-Lan Wang, Chunhui Lin, Hao Feng, Zhen Zhao, Yanjie Wang, Yuliang Liu, Hao Liu, Xiang Bai, and Can Huang. MTVQA: benchmarking multilingual text-centric visual question answering. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar, editors, *Findings of the Association for Computational Linguistics, ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, volume ACL 2025 of *Findings of ACL*, pages 7748–7763. Association for Computational Linguistics, 2025. doi: 10.18653/V1/2025.FINDINGS-ACL.404. URL https://doi.org/10.18653/v1/2025.findings-acl.404.

[585] Qiaoyu Tang, Ziliang Deng, Hongyu Lin, Xianpei Han, Qiao Liang, and Le Sun. Toolalpaca: Generalized tool learning for language models with 3000 simulated cases. *CoRR*, abs/2306.05301, 2023. doi: 10.48550/ARXIV.2306.05301. URL https://doi.org/10.48550/arXiv.2306.05301.

[586] Qiaoyu Tang, Hao Xiang, Le Yu, Bowen Yu, Yaojie Lu, Xianpei Han, Le Sun, WenJuan Zhang, Pengbo Wang, Shixuan Liu, Zhenru Zhang, Jianhong Tu, Hongyu Lin, and Junyang Lin. Beyond turn limits: Training deep search agents with dynamic context window. *CoRR*, abs/2510.08276, 2025. doi: 10.48550/ARXIV.2510.08276. URL https://doi.org/10.48550/arXiv.2510.08276.

[587] Xiangru Tang, Tianrui Qin, Tianhao Peng, Ziyang Zhou, Daniel Shao, Tingting Du, Xinming Wei, Peng Xia, Fang Wu, He Zhu, Ge Zhang, Jiaheng Liu, Xingyao Wang, Sirui Hong, Chenglin Wu, Hao Cheng, Chi Wang, and Wangchunshu Zhou. Agent KB: leveraging cross-domain experience for agentic problem solving. *CoRR*, abs/2507.06229, 2025. doi: 10.48550/ARXIV.2507.06229. URL https://doi.org/10.48550/arXiv.2507.06229.

[588] Zirui Tang, Xuanhe Zhou, Yumou Liu, Linchun Li, Yukai Wu, Weizheng Wang, Hongzhang Huang, Wei Zhou, Jun Zhou, Jiachen Song, Shaoli Yu, Jinqi Wang, Zihang Zhou, Hongyi Zhou, Yuting Lv, Jinyang Li, Jiashuo Liu, Ruoyu Chen, Chunwei Liu, Guoliang Li, Jihua Kang, and Fan Wu. Workspace-bench 1.0: Benchmarking AI agents on workspace tasks with large-scale file dependencies. *CoRR*, abs/2605.03596, 2026. doi: 10.48550/ARXIV.2605.03596. URL https://doi.org/10.48550/arXiv.2605.03596.

[589] Keda Tao, Yuhua Zheng, Xu Jia, Wenjie Du, Kele Shao, Hesong Wang, Xueyi Chen, Xin Jin, Junhan Zhu, Bohan Yu, Weiqiang Wang, Jian Liu, Can Qin, Yulun Zhang, Ming-Hsuan Yang, and Huan Wang. Lvomnibench: Pioneering long audio-video understanding evaluation for omnimodal llms. *CoRR*, abs/2603.19217, 2026. doi: 10.48550/ARXIV.2603.19217. URL https://doi.org/10.48550/arXiv.2603.19217.

[590] Zhengwei Tao, Jialong Wu, Wenbiao Yin, Junkai Zhang, Baixuan Li, Haiyang Shen, Kuan Li, Liwen Zhang, Xinyu Wang, Yong Jiang, Pengjun Xie, Fei Huang, and Jingren Zhou. Webshaper: Agentically data synthesizing via information-seeking formalization. *CoRR*, abs/2507.15061, 2025. doi: 10.48550/ARXIV.2507.15061. URL https://doi.org/10.48550/arXiv.2507.15061.

[591] Yi Tay, Mostafa Dehghani, Dara Bahri, and Donald Metzler. Efficient transformers: A survey. *ACM Comput. Surv.*, 55(6):109:1–109:28, 2023. doi: 10.1145/3530811. URL https://doi.org/10.1145/3530811.

[592] DeepConsult Team. DeepConsult: A deep research benchmark for consulting and business queries. https://github.com/youdotcom-oss/ydc-deep-research-evals, 2025. GitHub repository.

[593] Gemini Team, Petko Georgiev, Ving Ian Lei, Ryan Burnell, Libin Bai, Anmol Gulati, Garrett Tanzer, Damien Vincent, Zhufeng Pan, Shibo Wang, et al. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*, 2024.

[594] Gemini Robotics Team. Gemini robotics: Bringing AI into the physical world. *CoRR*, abs/2503.20020, 2025. doi: 10.48550/ARXIV.2503.20020. URL https://doi.org/10.48550/arXiv.2503.20020.

[595] Gemma Team. Gemma 2: Improving open language models at a practical size. *CoRR*, abs/2408.00118, 2024. doi: 10.48550/ARXIV.2408.00118. URL https://doi.org/10.48550/arXiv.2408.00118.

[596] Gemma Team. Gemma 3 technical report. *CoRR*, abs/2503.19786, 2025. doi: 10.48550/ARXIV.2503.19786. URL https://doi.org/10.48550/arXiv.2503.19786.

[597] Kimi Team. Kimi k2: Open agentic intelligence. *CoRR*, abs/2507.20534, 2025. doi: 10.48550/ARXIV.2507.20534. URL https://doi.org/10.48550/arXiv.2507.20534.

[598] Kimi Team. Kimi K2.5: Visual Agentic Intelligence. *CoRR*, abs/2602.02276, 2026. doi: 10.48550/ARXIV.2602.02276. URL https://doi.org/10.48550/arXiv.2602.02276.

[599] MiroMind Team, Song Bai, Lidong Bing, Carson Chen, Guanzheng Chen, Yuntao Chen, Zhe Chen, Ziyi Chen, Jifeng Dai, Xuan Dong, Wenhan Dou, Yue Deng, Yunjie Fu, Junqi Ge, Chenxia Han, Tammy Huang, Zhenhang Huang, Jerry Jiao, Shilei Jiang, Tianyu Jiao, Xiaoqi Jian, Lei Lei, Ruilin Li, Ryan Luo, Tiantong Li, Xiang Lin, Ziyuan Liu, Zhiqi Li, Jie Ni, Qiang Ren, Pax Sun, Shiqian Su, Chenxin Tao, Bin Wang, Hellen Wang, Haonan Wang, James Wang, Jin Wang, Jojo Wang, Letian Wang, Shizun Wang, Weizhi Wang, Zixuan Wang, Jinfan Xu, Sen Xing, Chenyu Yang, Hai Ye, Jiaheng Yu, Yue Yu, Muyan Zhong, Tianchen Zhao, Xizhou Zhu, Yanpeng Zhou, Yifan Zhang, and Zhi Zhu. Mirothinker: Pushing the performance boundaries of open-source research agents via model, context, and interactive scaling. *CoRR*, abs/2511.11793, 2025. doi: 10.48550/ARXIV.2511.11793. URL https://doi.org/10.48550/arXiv.2511.11793.

[600] Qwen Team. Qwen vlo: From "understanding" the world to "depicting" it, 2025. URL https://qwenlm.github.io/blog/qwen-vlo/.

[601] Qwen Team. Qwen3 technical report. *CoRR*, abs/2505.09388, 2025. doi: 10.48550/ARXIV.2505.09388. URL https://doi.org/10.48550/arXiv.2505.09388.

[602] Runchu Tian, Yining Ye, Yujia Qin, Xin Cong, Yankai Lin, Yinxu Pan, Yesai Wu, Haotian Hui, Weichuan Liu, Zhiyuan Liu, and Maosong Sun. Debugbench: Evaluating debugging capability of large language models. In *Findings of the Association for Computational Linguistics, ACL 2024, Bangkok, Thailand and virtual meeting, August 11-16, 2024*, pages 4173–4198, 2024. doi: 10.18653/V1/2024.FINDINGS-ACL.247. URL https://doi.org/10.18653/v1/2024.findings-acl.247.

[603] Shulin Tian, Ziniu Zhang, Liangyu Chen, and Ziwei Liu. Mmina: Benchmarking multihop multimodal internet agents. In *Findings of the Association for Computational Linguistics, ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, pages 13682–13697, 2025. doi: 10.18653/V1/2025.FINDINGS-ACL.703. URL https://doi.org/10.18653/v1/2025.findings-acl.703.

[604] Xiaoyu Tian, Haotian Wang, Shuaiting Chen, Hao Zhou, Kaichi Yu, Yudian Zhang, Jade Ouyang, Junxi Yin, Jiong Chen, Baoyan Guo, Lei Zhang, Junjie Tao, Yuansheng Song, Ming Cui, and Chengwei Liu. ASTRA: automated synthesis of agentic trajectories and reinforcement arenas. *CoRR*, abs/2601.21558, 2026. doi: 10.48550/ARXIV.2601.21558. URL https://doi.org/10.48550/arXiv.2601.21558.

[605] Guiyao Tie, Zenghui Yuan, Zeli Zhao, Chaoran Hu, Tianhe Gu, Ruihang Zhang, Sizhe Zhang, Junran Wu, Xiaoyue Tu, Ming Jin, Qingsong Wen, Lixing Chen, Pan Zhou, and Lichao Sun. Can llms correct themselves? A benchmark of self-correction in llms. In Danielle Belgrave, Cheng Zhang, Laura N. Montoya, Hsuan-Tien Lin, Razvan Pascanu, Piotr Koniusz, Marzyeh Ghassemi, Nancy Chen, Iván Vladimir Meza Ruíz, and Arturo Loaiza-Bonilla, editors, *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2025, NeurIPS 2025, San Diego, CA, USA, December 2-7, 2025 / Mexico City, Mexico, November 30 - December 5, 2025*, 2025. URL http://papers.nips.cc/paper_files/paper/2025/hash/ec07904adc847a45f53dceb44078f8f0-Abstract-Datasets_and_Benchmarks_Track.html.

[606] Matteo Tiezzi, Michele Casoni, Alessandro Betti, Marco Gori, and Stefano Melacci. State-space modeling in long sequence processing: a survey on recurrence in the transformer era. *Neural Networks*, 193:108039, 2026. doi: 10.1016/J.NEUNET.2025.108039. URL https://doi.org/10.1016/j.neunet.2025.108039.

[607] Peter Tong, Ellis Brown, Penghao Wu, Sanghyun Woo, Adithya Iyer, Sai Charitha Akula, Shusheng Yang, Jihan Yang, Manoj Middepogu, Ziteng Wang, Xichen Pan, Rob Fergus, Yann LeCun, and Saining Xie. Cambrian-1: A fully open, vision-centric exploration of multimodal llms. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang, editors, *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. URL http://papers.nips.cc/paper_files/paper/2024/hash/9ee3a664ccfeabc0da16ac6f1f1cfe59-Abstract-Conference.html.

[608] Shengbang Tong, Zhuang Liu, Yuexiang Zhai, Yi Ma, Yann LeCun, and Saining Xie. Eyes wide shut? exploring the visual shortcomings of multimodal llms. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024*, pages 9568–9578. IEEE, 2024. doi: 10.1109/CVPR52733.2024.00914. URL https://doi.org/10.1109/CVPR52733.2024.00914.

[609] Faraz Torabi, Garrett Warnell, and Peter Stone. Behavioral cloning from observation. In Jérôme Lang, editor, *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden*, pages 4950–4957. ijcai.org, 2018. doi: 10.24963/IJCAI.2018/687. URL https://doi.org/10.24963/ijcai.2018/687.

[610] Hieu Tran, Zonghai Yao, Nguyen Luong Tran, Zhichao Yang, Feiyun Ouyang, Shuo Han, Razieh Rahimi, and Hong Yu. PRIME: planning and retrieval-integrated memory for enhanced reasoning. In Sven Koenig, Chad Jenkins, and Matthew E. Taylor, editors, *Fortieth AAAI Conference on Artificial Intelligence, Thirty-Eighth Conference on Innovative Applications of Artificial Intelligence, Sixteenth Symposium on Educational Advances in Artificial Intelligence, AAAI 2026, Singapore, January 20-27, 2026*, pages 33268–33276. AAAI Press, 2026. doi: 10.1609/AAAI.V40I39.40612. URL https://doi.org/10.1609/aaai.v40i39.40612.

[611] Khanh-Tung Tran, Dung Dao, Minh-Duong Nguyen, Quoc-Viet Pham, Barry O'Sullivan, and Hoang D. Nguyen. Multi-agent collaboration mechanisms: A survey of llms. *CoRR*, abs/2501.06322, 2025. doi: 10.48550/ARXIV.2501.06322. URL https://doi.org/10.48550/arXiv.2501.06322.

[612] Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. Musique: Multihop questions via single-hop question composition, 2022. URL https://arxiv.org/abs/2108.00573.

[613] Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. Interleaving retrieval with chain-of-thought reasoning for knowledge-intensive multi-step questions. In Anna Rogers, Jordan L. Boyd-Graber, and Naoaki Okazaki, editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 10014–10037. Association for Computational Linguistics, 2023. doi: 10.18653/V1/2023.ACL-LONG.557. URL https://doi.org/10.18653/v1/2023.acl-long.557.

[614] Harsh Trivedi, Tushar Khot, Mareike Hartmann, Ruskin Manku, Vinty Dong, Edward Li, Shashank Gupta, Ashish Sabharwal, and Niranjan Balasubramanian. Appworld: A controllable world of apps and people for benchmarking interactive coding agents. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pages 16022–16076. Association for Computational Linguistics, 2024. doi: 10.18653/V1/2024.ACL-LONG.850. URL https://doi.org/10.18653/v1/2024.acl-long.850.

[615] George Tsoukalas, Jasper Lee, John Jennings, Jimmy Xin, Michelle Ding, Michael Jennings, Amitayush Thakur, and Swarat Chaudhuri. Putnambench: Evaluating neural theorem-provers on the putnam mathematical competition. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang, editors, *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. URL http://papers.nips.cc/paper_files/paper/2024/hash/1582eaf9e0cf349e1e5a6ee453100aa1-Abstract-Datasets_and_Benchmarks_Track.html.

[616] Lewis Tunstall, Edward Beeching, Nathan Lambert, Nazneen Rajani, Kashif Rasul, Younes Belkada, Shengyi Huang, Leandro von Werra, Clémentine Fourrier, Nathan Habib, Nathan Sarrazin, Omar Sanseviero, Alexander M. Rush, and Thomas Wolf. Zephyr: Direct distillation of LM alignment. *CoRR*, abs/2310.16944, 2023. doi: 10.48550/ARXIV.2310.16944. URL https://doi.org/10.48550/arXiv.2310.16944.

[617] Uchi Uchibeke. Before the tool call: Deterministic pre-action authorization for autonomous AI agents. *CoRR*, abs/2603.20953, 2026. doi: 10.48550/ARXIV.2603.20953. URL https://doi.org/10.48550/arXiv.2603.20953.

[618] Jonathan Uesato, Nate Kushman, Ramana Kumar, H. Francis Song, Noah Y. Siegel, Lisa Wang, Antonia Creswell, Geoffrey Irving, and Irina Higgins. Solving math word problems with process- and outcome-based feedback. *CoRR*, abs/2211.14275, 2022. doi: 10.48550/ARXIV.2211.14275. URL https://doi.org/10.48550/arXiv.2211.14275.

[619] Dani Valevski, Yaniv Leviathan, Moab Arar, and Shlomi Fruchter. Diffusion models are real-time game engines. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=P8pqeEkn1H.

[620] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html.

[621] Pranav Narayanan Venkit, Philippe Laban, Yilun Zhou, Kung-Hsiang Huang, Yixin Mao, and Chien-Sheng Wu. Deeptrace: Auditing deep research AI systems for tracking reliability across citations and evidence. *CoRR*, abs/2509.04499, 2025. doi: 10.48550/ARXIV.2509.04499. URL https://doi.org/10.48550/arXiv.2509.04499.

[622] Vercel. We removed 80% of our agent's tools, 2025. URL https://vercel.com/blog/we-removed-80-percent-of-our-agents-tools.

[623] Bertie Vidgen, Austin Mann, Abby Fennelly, John Wright Stanly, Lucas Rothman, Marco Burstein, Julien Benchek, David Ostrofsky, Anirudh Ravichandran, Debnil Sur, Neel Venugopal, Alannah Hsia, Isaac Robinson, Calix Huang, Olivia Varones, Daniyal Khan, Michael Haines, Zach Richards, Chirag Mahapatra, Brendan Foody, and Osvald Nitski. Apex-agents. *CoRR*, abs/2601.14242, 2026. doi: 10.48550/ARXIV.2601.14242. URL https://doi.org/10.48550/arXiv.2601.14242.

[624] An Vo, Khai-Nguyen Nguyen, Mohammad Reza Taesiri, Vy Tuong Dang, Anh Totti Nguyen, and Daeyoung Kim. Vision language models are biased. *CoRR*, abs/2505.23941, 2025. doi: 10.48550/ARXIV.2505.23941. URL https://doi.org/10.48550/arXiv.2505.23941.

[625] Kiran Vodrahalli, Santiago Ontanon, Nilesh Tripuraneni, Kelvin Xu, Sanil Jain, Rakesh Shivanna, Jeffrey Hui, Nishanth Dikkala, Mehran Kazemi, Bahare Fatemi, Rohan Anil, Ethan Dyer, Siamak Shakeri, Roopali Vij, Harsh Mehta, Vinay V. Ramasesh, Quoc Le, Ed H. Chi, Yifeng Lu, Orhan Firat, Angeliki Lazaridou, Jean-Baptiste Lespiau, Nithya Attaluri, and Kate Olszewska. Michelangelo: Long context evaluations beyond haystacks via latent structure queries. *CoRR*, abs/2409.12640, 2024. doi: 10.48550/ARXIV.2409.12640. URL https://doi.org/10.48550/arXiv.2409.12640.

[626] Daoyu Wang, Mingyue Cheng, Qi Liu, Shuo Yu, Zirui Liu, and Ze Guo. Paperarena: An evaluation benchmark for tool-augmented agentic reasoning on scientific literature. *CoRR*, abs/2510.10909, 2025. doi: 10.48550/ARXIV.2510.10909. URL https://doi.org/10.48550/arXiv.2510.10909.

[627] Daoyu Wang, Mingyue Cheng, Qingchuan Li, Shuo Yu, Jie Ouyang, and Qi Liu. Claw-r1: A step-level data middleware system for agentic reinforcement learning. *CoRR*, abs/2606.09138, 2026. doi: 10.48550/ARXIV.2606.09138. URL https://doi.org/10.48550/arXiv.2606.09138.

[628] Daoyu Wang, Qingchuan Li, Mingyue Cheng, Jie Ouyang, Shuo Yu, Qi Liu, and Enhong Chen. Steppo: Step-aligned policy optimization for agentic reinforcement learning. *CoRR*, abs/2604.18401, 2026. doi: 10.48550/ARXIV.2604.18401. URL https://doi.org/10.48550/arXiv.2604.18401.

[629] Erik Y. Wang, Sumeet Ramesh Motwani, James V. Roggeveen, Eliot Hodges, Dulhan Jayalath, Charles London, Kalyan Ramakrishnan, Flaviu Cipcigan, Philip Torr, and Alessandro Abate. Horizonmath: Measuring AI progress toward mathematical discovery with automatic verification. *CoRR*, abs/2603.15617, 2026. doi: 10.48550/ARXIV.2603.15617. URL https://doi.org/10.48550/arXiv.2603.15617.

[630] Guanzhi Wang, Yuqi Xie, Yunfan Jiang, Ajay Mandlekar, Chaowei Xiao, Yuke Zhu, Linxi Fan, and Anima Anandkumar. Voyager: An open-ended embodied agent with large language models. *Trans. Mach. Learn. Res.*, 2024, 2024. URL https://openreview.net/forum?id=ehfRiF0R3a.

[631] Hao Wang, Guozhi Wang, Han Xiao, Yufeng Zhou, Yue Pan, Jichao Wang, Ke Xu, Yafei Wen, Xiaohu Ruan, Xiaoxin Chen, and Honggang Qi. Skill-sd: Skill-conditioned self-distillation for multi-turn LLM agents. *CoRR*, abs/2604.10674, 2026. doi: 10.48550/ARXIV.2604.10674. URL https://doi.org/10.48550/arXiv.2604.10674.

[632] Haoming Wang, Haoyang Zou, Huatong Song, Jiazhan Feng, Junjie Fang, Junting Lu, Longxiang Liu, Qinyu Luo, Shihao Liang, Shijue Huang, Wanjun Zhong, Yining Ye, Yujia Qin, Yuwen Xiong, Yuxin Song, Zhiyong Wu, Aoyan Li, Bo Li, Chen Dun, Chong Liu, Daoguang Zan, Fuxing Leng, Hanbin Wang, Hao Yu, Haobin Chen, Hongyi Guo, Jing Su, Jingjia Huang, Kai Shen, Kaiyu Shi, Lin Yan, Peiyao Zhao, Pengfei Liu, Qinghao Ye, Renjie Zheng, Shulin Xin, Wayne Xin Zhao, Wen Heng, Wenhao Huang, Wenqian Wang, Xiaobo Qin, Yi Lin, Youbin Wu, Zehui Chen, Zihao Wang, Baoquan Zhong, Xinchun Zhang, Xujing Li, Yuanfan Li, Zhongkai Zhao, Chengquan Jiang, Faming Wu, Haotian Zhou, Jinlin Pang, Li Han, Qi Liu, Qianli Ma, Siyao Liu, Songhua Cai, Wenqi Fu, Xin Liu, Yaohui Wang, Zhi Zhang, Bo Zhou, Guoliang Li, Jiajun Shi, Jiale Yang, Jie Tang, Li Li, Qihua Han, Taoran Lu, Woyu Lin, Xiaokang Tong, Xinyao Li, Yichi Zhang, Yu Miao, Zhengxuan Jiang, Zili Li, Ziyuan Zhao, Chenxin Li, Dehua Ma, Feng Lin, Ge Zhang, Haihua Yang, Hangyu Guo, Hongda Zhu, Jiaheng Liu, Junda Du, Kai Cai, Kuanye Li, Lichen Yuan, Meilan Han, Minchao Wang, Shuyue Guo, Tianhao Cheng, Xiaobo Ma, Xiaojun Xiao, Xiaolong Huang, Xinjie Chen, Yidi Du, Yilin Chen, Yiwen Wang, Zhaojian Li, Zhenzhu Yang, Zhiyuan Zeng, Chaolin Jin, Chen Li, Hao Chen, Haoli Chen, Jian Chen, Qinghao Zhao, and Guang Shi. Ui-tars-2 technical report: Advancing gui agent with multi-turn reinforcement learning, 2025. URL https://arxiv.org/abs/2509.02544.

[633] Haoyu Wang, Christopher M. Poskitt, and Jun Sun. Agentspec: Customizable runtime enforcement for safe and reliable LLM agents. *CoRR*, abs/2503.18666, 2025. doi: 10.48550/ARXIV.2503.18666. URL https://doi.org/10.48550/arXiv.2503.18666.

[634] Haoyu Wang, Christopher M. Poskitt, Jiali Wei, and Jun Sun. Probguard: Probabilistic runtime monitoring for llm agent safety, 2026. URL https://arxiv.org/abs/2508.00500.

[635] Jianze Wang, Ying Liu, Jinlong Chen, Xuchun Hu, Qilong Zhang, Yu Cao, Jun Wang, Hua Yang, Yong Xie, and Qianglong Chen. MAD-OPD: breaking the ceiling in on-policy distillation via multi-agent debate. *CoRR*, abs/2605.01347, 2026. doi: 10.48550/ARXIV.2605.01347. URL https://doi.org/10.48550/arXiv.2605.01347.

[636] Jiaqi Wang, Wenhao Zhang, Weijie Shi, Yaliang Li, and James Cheng. TCOD: exploring temporal curriculum in on-policy distillation for multi-turn autonomous agents. *CoRR*, abs/2604.24005, 2026. doi: 10.48550/ARXIV.2604.24005. URL https://doi.org/10.48550/arXiv.2604.24005.

[637] Jihong Wang, Jiamu Zhou, Weiming Zhang, Teng Wang, Weiwen Liu, Zhuosheng Zhang, Xingyu Lou, Weinan

Zhang, Huarong Deng, and Jun Wang. Colorbrowseragent: Complex long-horizon browser agent with adaptive knowledge evolution, 2026. URL https://arxiv.org/abs/2601.07262.

[638] Junlin Wang, Jue Wang, Ben Athiwaratkun, Ce Zhang, and James Zou. Mixture-of-agents enhances large language model capabilities. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=h0ZfDIrj7T.

[639] Lei Wang, Wanyu Xu, Yihuai Lan, Zhiqiang Hu, Yunshi Lan, Roy Ka-Wei Lee, and Ee-Peng Lim. Plan-and-solve prompting: Improving zero-shot chain-of-thought reasoning by large language models. In Anna Rogers, Jordan L. Boyd-Graber, and Naoaki Okazaki, editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 2609–2634. Association for Computational Linguistics, 2023. doi: 10.18653/V1/2023.ACL-LONG.147. URL https://doi.org/10.18653/v1/2023.acl-long.147.

[640] Lei Wang, Chen Ma, Xueyang Feng, Zeyu Zhang, Hao Yang, Jingsen Zhang, Zhiyuan Chen, Jiakai Tang, Xu Chen, Yankai Lin, Wayne Xin Zhao, Zhewei Wei, and Jirong Wen. A survey on large language model based autonomous agents. *Frontiers Comput. Sci.*, 18(6):186345, 2024. doi: 10.1007/S11704-024-40231-1. URL https://doi.org/10.1007/s11704-024-40231-1.

[641] Liyuan Wang, Xingxing Zhang, Hang Su, and Jun Zhu. A comprehensive survey of continual learning: Theory, method and application. *IEEE Trans. Pattern Anal. Mach. Intell.*, 46(8):5362–5383, 2024. doi: 10.1109/TPAMI.2024.3367329. URL https://doi.org/10.1109/TPAMI.2024.3367329.

[642] Miles Wang, Robi Lin, Kat Hu, Joy Jiao, Neil Chowdhury, Ethan Chang, and Tejal Patwardhan. Frontierscience: Evaluating ai's ability to perform expert-level scientific tasks. *CoRR*, abs/2601.21165, 2026. doi: 10.48550/ARXIV.2601.21165. URL https://doi.org/10.48550/arXiv.2601.21165.

[643] Peiyi Wang, Lei Li, Zhihong Shao, Runxin Xu, Damai Dai, Yifei Li, Deli Chen, Yu Wu, and Zhifang Sui. Math-shepherd: Verify and reinforce llms step-by-step without human annotations. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pages 9426–9439. Association for Computational Linguistics, 2024. doi: 10.18653/V1/2024.ACL-LONG.510. URL https://doi.org/10.18653/v1/2024.acl-long.510.

[644] Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Yang Fan, Kai Dang, Mengfei Du, Xuancheng Ren, Rui Men, Dayiheng Liu, Chang Zhou, Jingren Zhou, and Junyang Lin. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *CoRR*, abs/2409.12191, 2024. doi: 10.48550/ARXIV.2409.12191. URL https://doi.org/10.48550/arXiv.2409.12191.

[645] Peng Wang, Yichun Shi, Xiaochen Lian, Zhonghua Zhai, Xin Xia, Xuefeng Xiao, Weilin Huang, and Jianchao Yang. Seededit 3.0: Fast and high-quality generative image editing. *CoRR*, abs/2506.05083, 2025. doi: 10.48550/ARXIV.2506.05083. URL https://doi.org/10.48550/arXiv.2506.05083.

[646] Qiuyue Wang, Mingsheng Li, Jian Guan, Jinhui Ye, Sicheng Xie, Yitao Liu, Junhao Chen, Zhixuan Liang, Jie Zhang, Xintong Hu, Xuhong Huang, Pei Lin, Junyang Lin, Dayiheng Liu, Shuai Bai, Jingren Zhou, Jiazhao Zhang, Haoqi Yuan, Gengze Zhou, Hang Yin, Ye Wang, Yiyang Huang, Zixing Lei, Wujian Peng, Delin Chen, Yingming Zheng, Jingyang Fan, Xianwei Zhuang, Xin Zhou, Haoyang Li, Anzhe Chen, Tong Zhang, Xuejing Liu, Yuchong Sun, Ruizhe Chen, Zhaohai Li, Chenxu Lü, Zhibo Yang, Tao Yu, and Xionghui Chen. Qwen-vla: Unifying vision-language-action modeling across tasks, environments, and robot embodiments. *CoRR*, abs/2605.30280, 2026. doi: 10.48550/ARXIV.2605.30280. URL https://doi.org/10.48550/arXiv.2605.30280.

[647] Renxi Wang, Xudong Han, Lei Ji, Shu Wang, Timothy Baldwin, and Haonan Li. Toolgen: Unified tool retrieval and calling via generation. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=XLMAMmowdY.

[648] Shaobo Wang, Zhengbo Jiao, Zifan Zhang, Yilang Peng, Xu Ze, Boyu Yang, Wei Wang, Hu Wei, and Linfeng Zhang. Socratic-zero : Bootstrapping reasoning via data-free agent co-evolution. *CoRR*, abs/2509.24726, 2025. doi: 10.48550/ARXIV.2509.24726. URL https://doi.org/10.48550/arXiv.2509.24726.

[649] Shaobo Wang, Xuan Ouyang, Tianyi Xu, Yuzheng Hu, Jialin Liu, Guo Chen, Tianyu Zhang, Junhao Zheng, Kexin Yang, Xingzhang Ren, Dayiheng Liu, and Linfeng Zhang. OPUS: towards efficient and principled data selection in large language model pre-training in every iteration. *CoRR*, abs/2602.05400, 2026. doi: 10.48550/ARXIV.2602.05400. URL https://doi.org/10.48550/arXiv.2602.05400.

[650] Shuting Wang, Qiaolin Xia, Vich Wang, Herberttli, Bobsimons, and Zhicheng Dou. Laser: Governing long-horizon agentic search via structured protocol and context register. *CoRR*, abs/2512.20458, 2025. doi: 10.48550/ARXIV.2512.20458. URL https://doi.org/10.48550/arXiv.2512.20458.

[651] Sinong Wang, Belinda Z. Li, Madian Khabsa, Han Fang, and Hao Ma. Linformer: Self-attention with linear complexity. *CoRR*, abs/2006.04768, 2020. URL https://arxiv.org/abs/2006.04768.

[652] Song Wang, Zhen Tan, Zihan Chen, Shuang Zhou, Tianlong Chen, and Jundong Li. Anymac: Cascading flexible multi-agent collaboration via next-agent prediction. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng, editors, *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, EMNLP 2025, Suzhou, China, November 4-9, 2025*, pages 11555–11567. Association for Computational Linguistics, 2025. doi: 10.18653/V1/2025.EMNLP-MAIN.584. URL https://doi.org/10.18653/v1/2025.emnlp-main.584.

[653] Weixun Wang, Shaopan Xiong, Gengru Chen, Wei Gao, Sheng Guo, Yancheng He, Ju Huang, Jiaheng Liu, Zhendong Li, Xiaoyang Li, Zichen Liu, Haizhou Zhao, Dakai An, Lunxi Cao, Qiyang Cao, Wanxi Deng, Feilei Du, Yiliang Gu, Jiahe Li, Xiang Li, Mingjie Liu, Yijia Luo, Zihe Liu, Yadao Wang, Pei Wang, Tianyuan Wu, Yanan Wu, Yuheng Zhao, Shuaibing Zhao, Jin Yang, Siran Yang, Yingshui Tan, Huimin Yi, Yuchi Xu, Yujin Yuan, Xingyao Zhang, Lin Qu, Wenbo Su, Wei Wang, Jiamang Wang, and Bo Zheng. Reinforcement learning optimization for large-scale learning: An efficient and user-friendly scaling library. *CoRR*, abs/2506.06122, 2025. doi: 10.48550/ARXIV.2506.06122. URL https://doi.org/10.48550/arXiv.2506.06122.

[654] Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, Zhaokai Wang, Zhe Chen, Hongjie Zhang, Ganlin Yang, Haomin Wang, Qi Wei, Jinhui Yin, Wenhao Li, Erfei Cui, Guanzhou Chen, Zichen Ding, Changyao Tian, Zhenyu Wu, JingJing Xie, Zehao Li, Bowen Yang, Yuchen Duan, Xuehui Wang, Zhi Hou, Haoran Hao, Tianyi Zhang, Songze Li, Xiangyu Zhao, Haodong Duan, Nianchen Deng, Bin Fu, Yinan He, Yi Wang, Conghui He, Botian Shi, Junjun He, Yingtong Xiong, Han Lv, Lijun Wu, Wenqi Shao, Kaipeng Zhang, Huipeng Deng, Biqing Qi, Jiaye Ge, Qipeng Guo, Wenwei Zhang, Songyang Zhang, Maosong Cao, Junyao Lin, Kexian Tang, Jianfei Gao, Haian Huang, Yuzhe Gu, Chengqi Lyu, Huanze Tang, Rui Wang, Haijun Lv, Wanli Ouyang, Limin Wang, Min Dou, Xizhou Zhu, Tong Lu, Dahua Lin, Jifeng Dai, Weijie Su, Bowen Zhou, Kai Chen, Yu Qiao, Wenhai Wang, and Gen Luo. Internvl3.5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *CoRR*, abs/2508.18265, 2025. doi: 10.48550/ARXIV.2508.18265. URL https://doi.org/10.48550/arXiv.2508.18265.

[655] Xiaohan Wang, Yuhui Zhang, Orr Zohar, and Serena Yeung-Levy. Videoagent: Long-form video understanding with large language model as agent. In Ales Leonardis, Elisa Ricci, Stefan Roth, Olga Russakovsky, Torsten Sattler, and Gül Varol, editors, *Computer Vision - ECCV 2024 - 18th European Conference, Milan, Italy, September 29-October 4, 2024, Proceedings, Part LXXX*, volume 15138 of *Lecture Notes in Computer Science*, pages 58–76. Springer, 2024. doi: 10.1007/978-3-031-72989-8_4. URL https://doi.org/10.1007/978-3-031-72989-8_4.

[656] Xingyao Wang, Yangyi Chen, Lifan Yuan, Yizhe Zhang, Yunzhu Li, Hao Peng, and Heng Ji. Executable code actions elicit better LLM agents. In Ruslan Salakhutdinov, Zico Kolter, Katherine A. Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp, editors, *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*, volume 235 of *Proceedings of Machine Learning Research*, pages 50208–50232. PMLR / OpenReview.net, 2024. URL https://proceedings.mlr.press/v235/wang24h.html.

[657] Xingyao Wang, Boxuan Li, Yufan Song, Frank F. Xu, Xiangru Tang, Mingchen Zhuge, Jiayi Pan, Yueqi Song, Bowen Li, Jaskirat Singh, Hoang H. Tran, Fuqiang Li, Ren Ma, Mingzhang Zheng, Bill Qian, Yanjun Shao, Niklas Muennighoff, Yizhe Zhang, Binyuan Hui, Junyang Lin, and et al. Openhands: An open platform for AI software developers as generalist agents. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=OJd3ayDDoF.

[658] Xinyu Jessica Wang, Haoyue Bai, Yiyou Sun, Haorui Wang, Shuibai Zhang, Wenjie Hu, Mya Schroder, Bilge Mutlu, Dawn Song, and Robert D. Nowak. The long-horizon task mirage? diagnosing where and why agentic systems break. *CoRR*, abs/2604.11978, 2026. doi: 10.48550/ARXIV.2604.11978. URL https://doi.org/10.48550/arXiv.2604.11978.

[659] Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V. Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023. URL https://openreview.net/forum?id=1PL1NIMMrw.

[660] Yaoxiang Wang, Zhiyong Wu, Junfeng Yao, and Jinsong Su. TDAG: A multi-agent framework based on dynamic

task decomposition and agent generation. *Neural Networks*, 185:107200, 2025. doi: 10.1016/J.NEUNET.2025.107200. URL https://doi.org/10.1016/j.neunet.2025.107200.

[661] Yiming Wang, Da Yin, Yuedong Cui, Ruichen Zheng, Zhiqian Li, Zongyu Lin, Di Wu, Xueqing Wu, Chenchen Ye, Yu Zhou, and Kai-Wei Chang. Llms as scalable, general-purpose simulators for evolving digital agent training. *CoRR*, abs/2510.14969, 2025. doi: 10.48550/ARXIV.2510.14969. URL https://doi.org/10.48550/arXiv.2510.14969.

[662] Yingxu Wang, Siwei Liu, Jinyuan Fang, and Zaiqiao Meng. Evoagentx: An automated framework for evolving agentic workflows. In Ivan Habernal, Peter Schulam, and Jörg Tiedemann, editors, *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, EMNLP 2025 - System Demonstrations, Suzhou, China, November 4-9, 2025*, pages 643–655. Association for Computational Linguistics, 2025. doi: 10.18653/V1/2025.EMNLP-DEMOS.47. URL https://doi.org/10.18653/v1/2025.emnlp-demos.47.

[663] Yinjie Wang, Tianbao Xie, Ke Shen, Mengdi Wang, and Ling Yang. Rlanything: Forge environment, policy, and reward model in completely dynamic RL system. *CoRR*, abs/2602.02488, 2026. doi: 10.48550/ARXIV.2602.02488. URL https://doi.org/10.48550/arXiv.2602.02488.

[664] Yu Wang and Xi Chen. Mirix: Multi-agent memory system for llm-based agents, 2025. URL https://arxiv.org/abs/2507.07957.

[665] Yutong Wang, Pengliang Ji, Chaoqun Yang, Kaixin Li, Ming Hu, Jiaoyang Li, and Guillaume Sartoretti. Mcts-judge: Test-time scaling in llm-as-a-judge for code correctness evaluation. *CoRR*, abs/2502.12468, 2025. doi: 10.48550/ARXIV.2502.12468. URL https://doi.org/10.48550/arXiv.2502.12468.

[666] Zhaowei Wang, Wenhao Yu, Xiyu Ren, Jipeng Zhang, Yu Zhao, Rohit Saxena, Liang Cheng, Ginny Y. Wong, Simon See, Pasquale Minervini, Yangqiu Song, and Mark Steedman. Mmlongbench: Benchmarking long-context vision-language models effectively and thoroughly. In Danielle Belgrave, Cheng Zhang, Laura N. Montoya, Hsuan-Tien Lin, Razvan Pascanu, Piotr Koniusz, Marzyeh Ghassemi, Nancy Chen, Iván Vladimir Meza Ruíz, and Arturo Loaiza-Bonilla, editors, *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2025, NeurIPS 2025, San Diego, CA, USA, December 2-7, 2025 / Mexico City, Mexico, November 30 - December 5, 2025*, 2025. URL http://papers.nips.cc/paper_files/paper/2025/hash/9c8712a9e6d34d60edb8c4c980d4a0f2-Abstract-Datasets_and_Benchmarks_Track.html.

[667] Zhenhailong Wang, Haiyang Xu, Junyang Wang, Xi Zhang, Ming Yan, Ji Zhang, Fei Huang, and Heng Ji. Mobile-agent-e: Self-evolving mobile assistant for complex tasks. *CoRR*, abs/2501.11733, 2025. doi: 10.48550/ARXIV.2501.11733. URL https://doi.org/10.48550/arXiv.2501.11733.

[668] Zhenting Wang, Qi Chang, Hemani Patel, Shashank Biju, Cheng-En Wu, Quan Liu, Aolin Ding, Alireza Rezazadeh, Ankit Shah, Yujia Bao, and Eugene Siow. Mcp-bench: Benchmarking tool-using LLM agents with complex real-world tasks via MCP servers. *CoRR*, abs/2508.20453, 2025. doi: 10.48550/ARXIV.2508.20453. URL https://doi.org/10.48550/arXiv.2508.20453.

[669] Zhenting Wang, Huancheng Chen, Jiayun Wang, and Wei Wei. Memex(rl): Scaling long-horizon LLM agents via indexed experience memory. *CoRR*, abs/2603.04257, 2026. doi: 10.48550/ARXIV.2603.04257. URL https://doi.org/10.48550/arXiv.2603.04257.

[670] Zhexuan Wang, Yutong Wang, Xuebo Liu, Liang Ding, Miao Zhang, Jie Liu, and Min Zhang. Agentdropout: Dynamic agent elimination for token-efficient and high-performance llm-based multi-agent collaboration. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar, editors, *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, pages 24013–24035. Association for Computational Linguistics, 2025. doi: 10.18653/V1/2025.ACL-LONG.1170. URL https://doi.org/10.18653/v1/2025.acl-long.1170.

[671] Zhun Wang, Tianneng Shi, Jingxuan He, Matthew Cai, Jialin Zhang, and Dawn Song. Cybergym: Evaluating AI agents' cybersecurity capabilities with real-world vulnerabilities at scale. *CoRR*, abs/2506.02548, 2025. doi: 10.48550/ARXIV.2506.02548. URL https://doi.org/10.48550/arXiv.2506.02548.

[672] Zihan Wang, Kangrui Wang, Qineng Wang, Pingyue Zhang, Linjie Li, Zhengyuan Yang, Xing Jin, Kefan Yu, Minh Nhat Nguyen, Licheng Liu, Eli Gottlieb, Yiping Lu, Kyunghyun Cho, Jiajun Wu, Li Fei-Fei, Lijuan Wang, Yejin Choi, and Manling Li. RAGEN: understanding self-evolution in LLM agents via multi-turn reinforcement learning. *CoRR*, abs/2504.20073, 2025. doi: 10.48550/ARXIV.2504.20073. URL https://doi.org/10.48550/arXiv.2504.20073.

[673] Zihao Wang, Shaofei Cai, Anji Liu, Xiaojian Ma, and Yitao Liang. Describe, explain, plan and select: Interactive planning with large language models enables open-world multi-task agents. *CoRR*, abs/2302.01560, 2023. doi: 10.48550/ARXIV.2302.01560. URL https://doi.org/10.48550/arXiv.2302.01560.

[674] Zora Zhiruo Wang, Jiayuan Mao, Daniel Fried, and Graham Neubig. Agent workflow memory, 2024. URL https://arxiv.org/abs/2409.07429.

[675] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022. URL http://papers.nips.cc/paper_files/paper/2022/hash/9d5609613524ecf4f15af0f7b31abca4-Abstract-Conference.html.

[676] Jason Wei, Zhiqing Sun, Spencer Papay, Scott McKinney, Jeffrey Han, Isa Fulford, Hyung Won Chung, Alex Tachard Passos, William Fedus, and Amelia Glaese. Browsecomp: A simple yet challenging benchmark for browsing agents. *CoRR*, abs/2504.12516, 2025. doi: 10.48550/ARXIV.2504.12516. URL https://doi.org/10.48550/arXiv.2504.12516.

[677] Jerry Wei, Chengrun Yang, Xinying Song, Yifeng Lu, Nathan Hu, Jie Huang, Dustin Tran, Daiyi Peng, Ruibo Liu, Da Huang, Cosmo Du, and Quoc V. Le. Long-form factuality in large language models. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang, editors, *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. URL http://papers.nips.cc/paper_files/paper/2024/hash/937ae0e83eb08d2cb8627fe1def8c751-Abstract-Conference.html.

[678] Jiaqi Wei, Yuejin Yang, Xiang Zhang, Yuhan Chen, Xiang Zhuang, Zhangyang Gao, Dongzhan Zhou, Guangshuai Wang, Zhiqiang Gao, Juntai Cao, Zijie Qiu, Xuming He, Qiang Zhang, Chenyu You, Shuangjia Zheng, Ning Ding, Wanli Ouyang, Nanqing Dong, Yu Cheng, Siqi Sun, Lei Bai, and Bowen Zhou. From AI for science to agentic science: A survey on autonomous scientific discovery. *CoRR*, abs/2508.14111, 2025. doi: 10.48550/ARXIV.2508.14111. URL https://doi.org/10.48550/arXiv.2508.14111.

[679] Jinbiao Wei, Yilun Zhao, Kangqi Ni, and Arman Cohan. Anchor: Branch-point data generation for GUI agents. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens, editors, *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2026, San Diego, California, United States, July 2-7, 2026*, pages 17031–17047. Association for Computational Linguistics, 2026. URL https://aclanthology.org/2026.acl-long.774/.

[680] Yuxiang Wei, Olivier Duchenne, Jade Copet, Quentin Carbonneaux, Lingming Zhang, Daniel Fried, Gabriel Synnaeve, Rishabh Singh, and Sida I. Wang. SWE-RL: advancing LLM reasoning via reinforcement learning on open software evolution. In Danielle Belgrave, Cheng Zhang, Laura N. Montoya, Hsuan-Tien Lin, Razvan Pascanu, Piotr Koniusz, Marzyeh Ghassemi, Nancy Chen, Iván Vladimir Meza Ruíz, and Arturo Loaiza-Bonilla, editors, *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2025, NeurIPS 2025, San Diego, CA, USA, December 2-7, 2025 / Mexico City, Mexico, November 30 - December 5, 2025*, 2025. URL http://papers.nips.cc/paper_files/paper/2025/hash/7107d4d2e837bde2171c6b71b5bde954-Abstract-Conference.html.

[681] Zhepei Wei, Wenlin Yao, Yao Liu, Weizhi Zhang, Qin Lu, Liang Qiu, Changlong Yu, Puyang Xu, Chao Zhang, Bing Yin, Hyokun Yun, and Lihong Li. Webagent-r1: Training web agents via end-to-end multi-turn reinforcement learning. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng, editors, *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, EMNLP 2025, Suzhou, China, November 4-9, 2025*, pages 7909–7928. Association for Computational Linguistics, 2025. doi: 10.18653/V1/2025.EMNLP-MAIN.401. URL https://doi.org/10.18653/v1/2025.emnlp-main.401.

[682] Tongyu Wen, Guanting Dong, and Zhicheng Dou. Smartsearch: Process reward-guided query refinement for search agents. *CoRR*, abs/2601.04888, 2026. doi: 10.48550/ARXIV.2601.04888. URL https://doi.org/10.48550/arXiv.2601.04888.

[683] Parker Whitfill, Ben Snodin, and Joel Becker. Forecasting AI time horizon under compute slowdowns. *CoRR*, abs/2511.19492, 2025. doi: 10.48550/ARXIV.2511.19492. URL https://doi.org/10.48550/arXiv.2511.19492.

[684] Lik Hang Kenny Wong, Xueyang Kang, Kaixin Bai, and Jianwei Zhang. A survey of robotic navigation and manipulation with physics simulators in the era of embodied AI. *CoRR*, abs/2505.01458, 2025. doi: 10.48550/ARXIV.2505.01458. URL https://doi.org/10.48550/arXiv.2505.01458.

[685] Ryan Wong, Jiawei Wang, Junjie Zhao, Li Chen, Yan Gao, Long Zhang, Xuan Zhou, Zuo Wang, Kai Xiang, Ge Zhang, Wenhao Huang, Yang Wang, and Ke Wang. Widesearch: Benchmarking agentic broad info-seeking. *CoRR*, abs/2508.07999, 2025. doi: 10.48550/ARXIV.2508.07999. URL https://doi.org/10.48550/arXiv.2508.07999.

[686] Di Wu, Hongwei Wang, Wenhao Yu, Yuwei Zhang, Kai-Wei Chang, and Dong Yu. Longmemeval: Benchmarking chat assistants on long-term interactive memory. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=pZiyCaVuti.

[687] Feijie Wu, Weiwu Zhu, Yuxiang Zhang, Soumya Chatterjee, Jiarong Zhu, Fan Mo, Rodin Luo, and Jing Gao. Portool: Tool-use LLM training with rewarded tree. *CoRR*, abs/2510.26020, 2025. doi: 10.48550/ARXIV.2510.26020. URL https://doi.org/10.48550/arXiv.2510.26020.

[688] Haoning Wu, Dongxu Li, Bei Chen, and Junnan Li. Longvideobench: A benchmark for long-context interleaved video-language understanding. In *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. URL http://papers.nips.cc/paper_files/paper/2024/hash/329ad516cf7a6ac306f29882e9c77558-Abstract-Datasets_and_Benchmarks_Track.html.

[689] Jialong Wu, Baixuan Li, Runnan Fang, Wenbiao Yin, Liwen Zhang, Zhenglin Wang, Zhengwei Tao, Dingchu Zhang, Zekun Xi, Robert Tang, Yong Jiang, Pengjun Xie, Fei Huang, and Jingren Zhou. Webdancer: Towards autonomous information seeking agency. In Danielle Belgrave, Cheng Zhang, Laura N. Montoya, Hsuan-Tien Lin, Razvan Pascanu, Piotr Koniusz, Marzyeh Ghassemi, Nancy Chen, Iván Vladimir Meza Ruíz, and Arturo Loaiza-Bonilla, editors, *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2025, NeurIPS 2025, San Diego, CA, USA, December 2-7, 2025 / Mexico City, Mexico, November 30 - December 5, 2025*, 2025. URL http://papers.nips.cc/paper_files/paper/2025/hash/af043aee9cb137785d93195c6cf4cd96-Abstract-Conference.html.

[690] Jialong Wu, Wenbiao Yin, Yong Jiang, Zhenglin Wang, Zekun Xi, Runnan Fang, Linhai Zhang, Yulan He, Deyu Zhou, Pengjun Xie, and Fei Huang. Webwalker: Benchmarking llms in web traversal. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar, editors, *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, pages 10290–10305. Association for Computational Linguistics, 2025. doi: 10.18653/V1/2025.ACL-LONG.508. URL https://doi.org/10.18653/v1/2025.acl-long.508.

[691] Qianhui Wu, Kanzhi Cheng, Rui Yang, Chaoyun Zhang, Jianwei Yang, Huiqiang Jiang, Jian Mu, Baolin Peng, Bo Qiao, Reuben Tan, Si Qin, Lars Liden, Qingwei Lin, Huan Zhang, Tong Zhang, Jianbing Zhang, Dongmei Zhang, and Jianfeng Gao. Gui-actor: Coordinate-free visual grounding for GUI agents. *CoRR*, abs/2506.03143, 2025. doi: 10.48550/ARXIV.2506.03143. URL https://doi.org/10.48550/arXiv.2506.03143.

[692] Qingyun Wu, Gagan Bansal, Jieyu Zhang, Yiran Wu, Beibin Li, Erkang Zhu, Li Jiang, Xiaoyun Zhang, Shaokun Zhang, Jiale Liu, Ahmed Hassan Awadallah, Ryen W White, Doug Burger, and Chi Wang. Autogen: Enabling next-gen llm applications via multi-agent conversation, 2023. URL https://arxiv.org/abs/2308.08155.

[693] Xixi Wu, Kuan Li, Yida Zhao, Liwen Zhang, Litu Ou, Huifeng Yin, Zhongwang Zhang, Yong Jiang, Pengjun Xie, Fei Huang, Minhao Cheng, Shuai Wang, Hong Cheng, and Jingren Zhou. Resum: Unlocking long-horizon search intelligence via context summarization. *CoRR*, abs/2509.13313, 2025. doi: 10.48550/ARXIV.2509.13313. URL https://doi.org/10.48550/arXiv.2509.13313.

[694] Yaxiong Wu, Sheng Liang, Chen Zhang, Yichao Wang, Yongyue Zhang, Huifeng Guo, Ruiming Tang, and Yong Liu. From human memory to AI memory: A survey on memory mechanisms in the era of llms. *CoRR*, abs/2504.15965, 2025. doi: 10.48550/ARXIV.2504.15965. URL https://doi.org/10.48550/arXiv.2504.15965.

[695] Yifan Wu, Yiran Peng, Yiyu Chen, Jianhao Ruan, Zijie Zhuang, Cheng Yang, Jiayi Zhang, Man Chen, Yenchi Tseng, Zhaoyang Yu, Liang Chen, Yuyao Zhai, Bang Liu, Chenglin Wu, and Yuyu Luo. Autowebworld: Synthesizing infinite verifiable web environments via finite state machines. *CoRR*, abs/2602.14296, 2026. doi: 10.48550/ARXIV.2602.14296. URL https://doi.org/10.48550/arXiv.2602.14296.

[696] Yong Wu, Yanzhao Zheng, Tianze Xu, ZhenTao Zhang, YuanQiang Yu, JiHuai Zhu, Chao Ma, BinBin Lin, Baohua Dong, Hangcheng Zhu, Ruohui Huang, and Gang Yu. Contextbudget: Budget-aware context management for long-horizon search agents. *CoRR*, abs/2604.01664, 2026. doi: 10.48550/ARXIV.2604.01664. URL https://doi.org/10.48550/arXiv.2604.01664.

[697] Yubin Wu, Zicheng Cai, Liping Ning, Hua Wang, Zhi Chen, Yaohua Tang, and Hao Chen. Litegui: Distilling compact GUI agents with reinforcement learning. *CoRR*, abs/2605.07505, 2026. doi: 10.48550/ARXIV.2605.07505. URL https://doi.org/10.48550/arXiv.2605.07505.

[698] Yuhao Wu, Franziska Roesner, Tadayoshi Kohno, Ning Zhang, and Umar Iqbal. Isolategpt: An execution isolation architecture for llm-based agentic systems. In *32nd Annual Network and Distributed System Security Symposium, NDSS 2025, San Diego, California, USA, February 24-28, 2025*. The Internet Society, 2025. URL https://www.ndss-symposium.org/ndss-paper/isolategpt-an-execution-isolation-architecture-for-llm-based-agentic-systems/.

[699] Zhuoyuan Wu and Jun Gao. OSCAR: omni-embodiment action-conditioned world model for robotics. *CoRR*, abs/2606.04463, 2026. doi: 10.48550/ARXIV.2606.04463. URL https://doi.org/10.48550/arXiv.2606.04463.

[700] Zijian Wu, Xiangyan Liu, Xinyuan Zhang, Lingjun Chen, Fanqing Meng, Lingxiao Du, Yiran Zhao, Fanshi Zhang, Yaoqi Ye, Jiawei Wang, Zirui Wang, Jinjie Ni, Yufan Yang, Arvin Xu, and Michael Qizhe Shieh. Mcpmark: A benchmark for stress-testing realistic and comprehensive MCP use. *CoRR*, abs/2509.24002, 2025. doi: 10.48550/ARXIV.2509.24002. URL https://doi.org/10.48550/arXiv.2509.24002.

[701] Zhiheng Xi, Jixuan Huang, Chenyang Liao, Baodai Huang, Honglin Guo, Jiaqi Liu, Rui Zheng, Junjie Ye, Jiazheng Zhang, Wenxiang Chen, Wei He, Yiwen Ding, Guanyu Li, Zehui Chen, Zhengyin Du, Xuesong Yao, Yufei Xu, Jiecao Chen, Tao Gui, Zuxuan Wu, Qi Zhang, Xuanjing Huang, and Yu-Gang Jiang. AgentGym-RL: Training LLM agents for long-horizon decision making through multi-turn reinforcement learning. *CoRR*, abs/2509.08755, 2025. doi: 10.48550/ARXIV.2509.08755. URL https://doi.org/10.48550/arXiv.2509.08755.

[702] Zhiheng Xi, Chenyang Liao, Guanyu Li, Zhihao Zhang, Wenxiang Chen, Binghai Wang, Senjie Jin, Yuhao Zhou, Jian Guan, Wei Wu, Tao Ji, Tao Gui, Qi Zhang, and Xuanjing Huang. Agentprm: Process reward models for LLM agents via step-wise promise and progress. In Hakim Hacid, Yoelle Maarek, Francesco Bonchi, Ido Guy, and Emine Yilmaz, editors, *Proceedings of the ACM Web Conference 2026, WWW 2026, Dubai, United Arab Emirates, originally scheduled for April 13-17, 2026, rescheduled for June 29 - July 3, 2026*, pages 4184–4195. ACM, 2026. doi: 10.1145/3774904.3792551. URL https://doi.org/10.1145/3774904.3792551.

[703] Chunqiu Steven Xia, Yinlin Deng, Soren Dunn, and Lingming Zhang. Agentless: Demystifying llm-based software engineering agents. *CoRR*, abs/2407.01489, 2024. doi: 10.48550/ARXIV.2407.01489. URL https://doi.org/10.48550/arXiv.2407.01489.

[704] Peng Xia, Kaide Zeng, Jiaqi Liu, Can Qin, Fang Wu, Yiyang Zhou, Caiming Xiong, and Huaxiu Yao. Agent0: Unleashing self-evolving agents from zero data via tool-integrated reasoning. *CoRR*, abs/2511.16043, 2025. doi: 10.48550/ARXIV.2511.16043. URL https://doi.org/10.48550/arXiv.2511.16043.

[705] Peng Xia, Jianwen Chen, Hanyang Wang, Jiaqi Liu, Kaide Zeng, Yu Wang, Siwei Han, Yiyang Zhou, Xujiang Zhao, Haifeng Chen, Zeyu Zheng, Cihang Xie, and Huaxiu Yao. Skillrl: Evolving agents via recursive skill-augmented reinforcement learning. *CoRR*, abs/2602.08234, 2026. doi: 10.48550/ARXIV.2602.08234. URL https://doi.org/10.48550/arXiv.2602.08234.

[706] Peng Xia, Jianwen Chen, Xinyu Yang, Haoqin Tu, Jiaqi Liu, Kaiwen Xiong, Siwei Han, Shi Qiu, Haonian Ji, Yuyin Zhou, Zeyu Zheng, Cihang Xie, and Huaxiu Yao. Metaclaw: Just talk - an agent that meta-learns and evolves in the wild. *CoRR*, abs/2603.17187, 2026. doi: 10.48550/ARXIV.2603.17187. URL https://doi.org/10.48550/arXiv.2603.17187.

[707] Tianle Xia, Lingxiang Hu, Yiding Sun, Ming Xu, Lan Xu, Siying Wang, Wei Xu, and Jie Jiang. Grasp: Graph-structured skill compositions for LLM agents. *CoRR*, abs/2604.17870, 2026. doi: 10.48550/ARXIV.2604.17870. URL https://doi.org/10.48550/arXiv.2604.17870.

[708] Zhen Xiang, Linzhi Zheng, Yanjie Li, Junyuan Hong, Qinbin Li, Han Xie, Jiawei Zhang, Zidi Xiong, Chulin Xie, Carl Yang, Dawn Song, and Bo Li. Guardagent: Safeguard LLM agents via knowledge-enabled reasoning. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu, editors, *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*, volume 267 of *Proceedings of Machine Learning Research*. PMLR / OpenReview.net, 2025. URL https://proceedings.mlr.press/v267/xiang25a.html.

[709] Zhiyi Xiang, Chengyi Yang, Zerui Chen, Zhimin Wei, Yunbo Tang, Zongpei Teng, Zexi Peng, Zongxia Li, Chengsong Huang, Yicheng He, et al. A systematic survey of self-evolving agents: From model-centric to environment-driven co-evolution. https://doi.org/10.36227/techrxiv.177203250.05832634/v2, 2026. Technical report, TechRxiv (no arXiv version available).

[710] Changyi Xiao, Mengdi Zhang, and Yixin Cao. BNPO: beta normalization policy optimization. *CoRR*, abs/2506.02864, 2025. doi: 10.48550/ARXIV.2506.02864. URL https://doi.org/10.48550/arXiv.2506.02864.

[711] Yang Xiao, Mohan Jiang, Jie Sun, Keyu Li, Jifan Lin, Yumin Zhuang, Ji Zeng, Shijie Xia, Qishuo Hua, Xuefeng Li, Xiaojie Cai, Tongyu Wang, Yue Zhang, Liming Liu, Xia Wu, Jinlong Hou, Yuan Cheng, Wenjie Li, Xiang Wang, Dequan Wang, and Pengfei Liu. LIMI: less is more for agency. *CoRR*, abs/2509.17567, 2025. doi: 10.48550/ARXIV.2509.17567. URL https://doi.org/10.48550/arXiv.2509.17567.

[712] Zikai Xiao, Jianhong Tu, Chuhang Zou, Yuxin Zuo, Zhi Li, Peng Wang, Bowen Yu, Fei Huang, Junyang Lin, and Zuozhu Liu. Webworld: A large-scale world model for web agent training. *CoRR*, abs/2602.14721, 2026. doi: 10.48550/ARXIV.2602.14721. URL https://doi.org/10.48550/arXiv.2602.14721.

[713] Jingxu Xie, Dylan Xu, Xuandong Zhao, and Dawn Song. Agentsynth: Scalable task generation for generalist computer-use agents. *CoRR*, abs/2506.14205, 2025. doi: 10.48550/ARXIV.2506.14205. URL https://doi.org/10.48550/arXiv.2506.14205.

[714] Junlin Xie, Zhihong Chen, Ruifei Zhang, Xiang Wan, and Guanbin Li. Large multimodal agents: A survey, 2024. URL https://arxiv.org/abs/2402.15116.

[715] Sang Michael Xie, Hieu Pham, Xuanyi Dong, Nan Du, Hanxiao Liu, Yifeng Lu, Percy Liang, Quoc V. Le, Tengyu Ma, and Adams Wei Yu. Doremi: Optimizing data mixtures speeds up language model pretraining. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine, editors, *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/dcba6be91359358c2355cd920da3fcbd-Abstract-Conference.html.

[716] Tianbao Xie, Danyang Zhang, Jixuan Chen, Xiaochuan Li, Siheng Zhao, Ruisheng Cao, Toh Jing Hua, Zhoujun Cheng, Dongchan Shin, Fangyu Lei, Yitao Liu, Yiheng Xu, Shuyan Zhou, Silvio Savarese, Caiming Xiong, Victor Zhong, and Tao Yu. Osworld: Benchmarking multimodal agents for open-ended tasks in real computer environments. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang, editors, *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. URL http://papers.nips.cc/paper_files/paper/2024/hash/5d413e48f84dc61244b6be550f1cd8f5-Abstract-Datasets_and_Benchmarks_Track.html.

[717] Huajian Xin, Luming Li, Xiaoran Jin, Jacques Fleuriot, and Wenda Li. Ape-bench: Evaluating automated proof engineering for formal math libraries, 2026. URL https://arxiv.org/abs/2504.19110.

[718] Zhenghao Xing, Ruiyang Xu, Yuxuan Wang, Jinzheng He, Ziyang Ma, Qize Yang, Yunfei Chu, Jin Xu, Junyang Lin, Chi-Wing Fu, and Pheng-Ann Heng. Native active perception as reasoning for omni-modal understanding. *CoRR*, abs/2606.19341, 2026. doi: 10.48550/ARXIV.2606.19341. URL https://doi.org/10.48550/arXiv.2606.19341.

[719] Weimin Xiong, Yifan Song, Xiutian Zhao, Wenhao Wu, Xun Wang, Ke Wang, Cheng Li, Wei Peng, and Sujian Li. Watch every step! LLM agent learning via iterative step-level process refinement. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen, editors, *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, EMNLP 2024, Miami, FL, USA, November 12-16, 2024*, pages 1556–1572. Association for Computational Linguistics, 2024. doi: 10.18653/V1/2024.EMNLP-MAIN.93. URL https://doi.org/10.18653/v1/2024.emnlp-main.93.

[720] Binfeng Xu, Zhiyuan Peng, Bowen Lei, Subhabrata Mukherjee, Yuchen Liu, and Dongkuan Xu. Rewoo: Decoupling reasoning from observations for efficient augmented language models. *CoRR*, abs/2305.18323, 2023. doi: 10.48550/ARXIV.2305.18323. URL https://doi.org/10.48550/arXiv.2305.18323.

[721] Chejian Xu, Wei Ping, Peng Xu, Zihan Liu, Boxin Wang, Mohammad Shoeybi, Bo Li, and Bryan Catanzaro. From 128k to 4m: Efficient training of ultra-long context large language models. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens, editors, *Findings of the Association for Computational Linguistics, ACL 2026, San Diego, California, United States, July 2-7, 2026*, pages 13122–13133. Association for Computational Linguistics, 2026. URL https://aclanthology.org/2026.findings-acl.640/.

[722] Frank F. Xu, Yufan Song, Boxuan Li, Yuxuan Tang, Kritanjali Jain, Mengxue Bao, Zora Zhiruo Wang, Xuhui Zhou, Zhitong Guo, Murong Cao, Mingyang Yang, Hao Yang Lu, Amaad Martin, Zhe Su, Leander Maben, Raj Mehta, Wayne Chi, Lawrence Keunho Jang, Yiqing Xie, Shuyan Zhou, and Graham Neubig. Theagentcompany: Benchmarking LLM agents on consequential real world tasks. *CoRR*, abs/2412.14161, 2024. doi: 10.48550/ARXIV.2412.14161. URL https://doi.org/10.48550/arXiv.2412.14161.

[723] Haiyang Xu, Xi Zhang, Haowei Liu, Junyang Wang, Zhaozai Zhu, Shengjie Zhou, Xuhao Hu, Feiyu Gao, Junjie Cao, Zihua Wang, Zhiyuan Chen, Jitong Liao, Qi Zheng, Jiahui Zeng, Ze Xu, Shuai Bai, Junyang Lin, Jingren Zhou, and Ming Yan. Mobile-agent-v3.5: Multi-platform fundamental GUI agents. *CoRR*, abs/2602.16855, 2026. doi: 10.48550/ARXIV.2602.16855. URL https://doi.org/10.48550/arXiv.2602.16855.

[724] Haoyuan Xu, Chang Li, Xinyan Ma, Xianhao Ou, Zihan Zhang, Tao He, Xiangyu Liu, Zixiang Wang, Jiafeng Liang, Zheng Chu, Runxuan Liu, Rongchuan Mu, Dandan Tu, Ming Liu, and Bing Qin. The evolution of tool use in LLM agents: From single-tool call to multi-tool orchestration. *CoRR*, abs/2603.22862, 2026. doi: 10.48550/ARXIV.2603.22862. URL https://doi.org/10.48550/arXiv.2603.22862.

[725] Huimin Xu, Shuai Zhao, Xiaobao Wu, and Anh Tuan Luu. Understanding and preventing entropy collapse in RLVR with on-policy entropy flow optimization. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens, editors, *Findings of the Association for Computational Linguistics, ACL 2026, San Diego, California, United States, July 2-7, 2026*, pages 17759–17771. Association for Computational Linguistics, 2026. URL https://aclanthology.org/2026.findings-acl.879/.

[726] Jin Xu, Zhifang Guo, Jinzheng He, Hangrui Hu, Ting He, Shuai Bai, Keqin Chen, Jialin Wang, Yang Fan, Kai Dang, Bin Zhang, Xiong Wang, Yunfei Chu, and Junyang Lin. Qwen2.5-omni technical report. *CoRR*, abs/2503.20215, 2025. doi: 10.48550/ARXIV.2503.20215. URL https://doi.org/10.48550/arXiv.2503.20215.

[727] Jin Xu, Zhifang Guo, Hangrui Hu, Yunfei Chu, Xiong Wang, Jinzheng He, Yuxuan Wang, Xian Shi, Ting He, Xinfa Zhu, Yuanjun Lv, Yongqi Wang, Dake Guo, He Wang, Linhan Ma, Pei Zhang, Xinyu Zhang, Hongkun Hao, Zishan Guo, Baosong Yang, Bin Zhang, Ziyang Ma, Xipin Wei, Shuai Bai, Keqin Chen, Xuejing Liu, Peng Wang, Mingkun Yang, Dayiheng Liu, Xingzhang Ren, Bo Zheng, Rui Men, Fan Zhou, Bowen Yu, Jianxin Yang, Le Yu, Jingren Zhou, and Junyang Lin. Qwen3-omni technical report, 2025. URL https://arxiv.org/abs/2509.17765.

[728] Ke Xu, Yuhao Wang, Ziyang Cheng, Hongcheng Liu, Yanfeng Wang, and Yu Wang. Agentic active omni-modal perception for multi-hop audio-visual reasoning. *CoRR*, abs/2605.28192, 2026. doi: 10.48550/ARXIV.2605.28192. URL https://doi.org/10.48550/arXiv.2605.28192.

[729] Ran Xu, Yuchen Zhuang, Yishan Zhong, Yue Yu, Zifeng Wang, Xiangru Tang, Hang Wu, May D. Wang, Peifeng Ruan, Donghan Yang, Tao Wang, Guanghua Xiao, Xin Liu, Carl Yang, Yang Xie, and Wenqi Shi. Medagentgym: A scalable agentic training environment for code-centric reasoning in biomedical data science, 2025. URL https://arxiv.org/abs/2506.04405.

[730] Renjun Xu and Yang Yan. Agent skills for large language models: Architecture, acquisition, security, and the path forward. *CoRR*, abs/2602.12430, 2026. doi: 10.48550/ARXIV.2602.12430. URL https://doi.org/10.48550/arXiv.2602.12430.

[731] Wujiang Xu, Zujie Liang, Kai Mei, Hang Gao, Juntao Tan, and Yongfeng Zhang. A-MEM: agentic memory for LLM agents. *CoRR*, abs/2502.12110, 2025. doi: 10.48550/ARXIV.2502.12110. URL https://doi.org/10.48550/arXiv.2502.12110.

[732] Wujiang Xu, Wentian Zhao, Zhenting Wang, Yu-Jhe Li, Can Jin, Mingyu Jin, Kai Mei, Kun Wang, and Dimitris N. Metaxas. EPO: entropy-regularized policy optimization for LLM agents reinforcement learning. *CoRR*, abs/2509.22576, 2025. doi: 10.48550/ARXIV.2509.22576. URL https://doi.org/10.48550/arXiv.2509.22576.

[733] Xuenan Xu, Jiahao Mei, Chenliang Li, Yuning Wu, Ming Yan, Shaopeng Lai, Ji Zhang, and Mengyue Wu. Mm-storyagent: Immersive narrated storybook video generation with a multi-agent paradigm across text, image and audio. *CoRR*, abs/2503.05242, 2025. doi: 10.48550/ARXIV.2503.05242. URL https://doi.org/10.48550/arXiv.2503.05242.

[734] Yifan Xu, Xiao Liu, Xueqiao Sun, Siyi Cheng, Hao Yu, Hanyu Lai, Shudan Zhang, Dan Zhang, Jie Tang, and Yuxiao Dong. Androidlab: Training and systematic benchmarking of android autonomous agents. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, pages 2144–2166, 2025. doi: 10.18653/V1/2025.ACL-LONG.107. URL https://doi.org/10.18653/v1/2025.acl-long.107.

[735] Yuanda Xu, Hejian Sang, Zhengze Zhou, Ran He, Zhipeng Wang, and Alborz Geramifard. TIP: token importance in on-policy distillation. *CoRR*, abs/2604.14084, 2026. doi: 10.48550/ARXIV.2604.14084. URL https://doi.org/10.48550/arXiv.2604.14084.

[736] Zelai Xu, Zhexuan Xu, Ruize Zhang, Chunyang Zhu, Shi Yu, Weilin Liu, Quanlu Zhang, Wenbo Ding, Chao Yu, and Yu Wang. Wideseek-r1: Exploring width scaling for broad information seeking via multi-agent reinforcement

learning. *CoRR*, abs/2602.04634, 2026. doi: 10.48550/ARXIV.2602.04634. URL https://doi.org/10.48550/arXiv.2602.04634.

[737] Zhangchen Xu, Adriana Meza Soria, Shawn Tan, Anurag Roy, Ashish Sunil Agrawal, Radha Poovendran, and Rameswar Panda. TOUCAN: synthesizing 1.5m tool-agentic data from real-world MCP environments. *CoRR*, abs/2510.01179, 2025. doi: 10.48550/ARXIV.2510.01179. URL https://doi.org/10.48550/arXiv.2510.01179.

[738] Xiangyuan Xue, Yifan Zhou, Guibin Zhang, Zaibin Zhang, Yijiang Li, Chen Zhang, Zhenfei Yin, Philip Torr, Wanli Ouyang, and Lei Bai. Comas: Co-evolving multi-agent systems via interaction rewards. *CoRR*, abs/2510.08529, 2025. doi: 10.48550/ARXIV.2510.08529. URL https://doi.org/10.48550/arXiv.2510.08529.

[739] Zhucun Xue, Jiangning Zhang, Xurong Xie, Yuxuan Cai, Yong Liu, Xiangtai Li, and Dacheng Tao. Adavideorag: Omni-contextual adaptive retrieval-augmented efficient long video understanding. *CoRR*, abs/2506.13589, 2025. doi: 10.48550/ARXIV.2506.13589. URL https://doi.org/10.48550/arXiv.2506.13589.

[740] Yutaro Yamada, Robert Tjarko Lange, Cong Lu, Shengran Hu, Chris Lu, Jakob N. Foerster, Jeff Clune, and David Ha. The AI scientist-v2: Workshop-level automated scientific discovery via agentic tree search. *CoRR*, abs/2504.08066, 2025. doi: 10.48550/ARXIV.2504.08066. URL https://doi.org/10.48550/arXiv.2504.08066.

[741] Lecheng Yan, Yichong Zhang, Ben Pan, Xiaoyu Zheng, Jiawei Qian, Anqi Wu, Wenxi Li, and Chenyang Lyu. Crayotter: Traceable multi-agent workflows for long-form video editing. *CoRR*, abs/2606.07636, 2026. doi: 10.48550/ARXIV.2606.07636. URL https://doi.org/10.48550/arXiv.2606.07636.

[742] Sikuan Yan, Ahmed Bahloul, Ercong Nie, Susanna Schwarzmann, Riccardo Trivisonno, Volker Tresp, and Yunpu Ma. Memory-r2: Fair credit assignment for long-horizon memory-augmented LLM agents. *CoRR*, abs/2605.21768, 2026. doi: 10.48550/ARXIV.2605.21768. URL https://doi.org/10.48550/arXiv.2605.21768.

[743] Sikuan Yan, Xiufeng Yang, Zuchao Huang, Ercong Nie, Zifeng Ding, Zonggen Li, Xiaowen Ma, Jinhe Bi, Kristian Kersting, Jeff Z. Pan, Hinrich Schütze, Volker Tresp, and Yunpu Ma. Memory-r1: Enhancing large language model agents to manage and utilize memories via reinforcement learning. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens, editors, *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2026, San Diego, California, United States, July 2-7, 2026*, pages 12805–12825. Association for Computational Linguistics, 2026. URL https://aclanthology.org/2026.acl-long.583/.

[744] An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report. *CoRR*, abs/2412.15115, 2024. doi: 10.48550/ARXIV.2412.15115. URL https://doi.org/10.48550/arXiv.2412.15115.

[745] Bowen Yang, Kaiming Jin, Zhenyu Wu, Zhaoyang Liu, Qiushi Sun, Zehao Li, JingJing Xie, Zhoumianze Liu, Fangzhi Xu, Kanzhi Cheng, Yian Wang, Qingyun Li, Yu Qiao, Zun Wang, and Zichen Ding. Os-symphony: A holistic framework for robust and generalist computer-using agents. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens, editors, *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2026, San Diego, California, United States, July 2-7, 2026*, pages 22300–22330. Association for Computational Linguistics, 2026. URL https://aclanthology.org/2026.acl-long.1021/.

[746] Cehao Yang, Xiaojun Wu, Xueyuan Lin, Chengjin Xu, Xuhui Jiang, Yuanliang Sun, Jia Li, Hui Xiong, and Jian Guo. Graphsearch: An agentic deep searching workflow for graph retrieval-augmented generation. *CoRR*, abs/2509.22009, 2025. doi: 10.48550/ARXIV.2509.22009. URL https://doi.org/10.48550/arXiv.2509.22009.

[747] 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. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang, editors, *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. URL http://papers.nips.cc/paper_files/paper/2024/hash/5a7c947568c1b1328ccc5230172e1e7c-Abstract-Conference.html.

[748] John Yang, Kilian Lieret, Carlos E. Jimenez, Alexander Wettig, Kabir Khandpur, Yanzhe Zhang, Binyuan Hui, Ofir Press, Ludwig Schmidt, and Diyi Yang. Swe-smith: Scaling data for software engineering agents. In Danielle Belgrave, Cheng Zhang, Laura N. Montoya, Hsuan-Tien Lin, Razvan Pascanu, Piotr Koniusz, Marzyeh Ghassemi, Nancy Chen, Iván Vladimir Meza Ruíz, and Arturo Loaiza-Bonilla, editors, *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information*

Processing Systems 2025, NeurIPS 2025, San Diego, CA, USA, December 2-7, 2025 / Mexico City, Mexico, November 30 - December 5, 2025, 2025. URL http://papers.nips.cc/paper_files/paper/2025/hash/8b86cf5ace600c48fd188efbb8dedec8-Abstract-Datasets_and_Benchmarks_Track.html.

[749] John Yang, Kilian Lieret, Jeffrey Ma, Parth Thakkar, Dmitrii Pedchenko, Sten Sootla, Emily McMilin, Pengcheng Yin, Rui Hou, Gabriel Synnaeve, Diyi Yang, and Ofir Press. Programbench: Can language models rebuild programs from scratch? *CoRR*, abs/2605.03546, 2026. doi: 10.48550/ARXIV.2605.03546. URL https://doi.org/10.48550/arXiv.2605.03546.

[750] Ling Yang, Zhaochen Yu, Tianjun Zhang, Shiyi Cao, Minkai Xu, Wentao Zhang, Joseph E. Gonzalez, and Bin Cui. Buffer of thoughts: Thought-augmented reasoning with large language models. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang, editors, *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. URL http://papers.nips.cc/paper_files/paper/2024/hash/cde328b7bf6358f5ebb91fe9c539745e-Abstract-Conference.html.

[751] Pei Yang, Wanyi Chen, Asuka Yuxi Zheng, Xueqian Li, Xiang Li, Haoqin Tu, Jie Xiao, Yifan Pang, Dongdong Zhang, Fuqiang Li, Alfred Long, Lynn Ai, Eric Yang, and Bill Shi. AOI: turning failed trajectories into training signals for autonomous cloud diagnosis. *CoRR*, abs/2603.03378, 2026. doi: 10.48550/ARXIV.2603.03378. URL https://doi.org/10.48550/arXiv.2603.03378.

[752] Qianyu Yang, Yang Liu, Jiaqi Li, Jun Bai, Hao Chen, Kaiyuan Chen, Tiliang Duan, Jiayun Dong, Xiaobo Hu, Zixia Jia, Yang Liu, Tao Peng, Yixin Ren, Ran Tian, Zaiyuan Wang, Yanglihong Xiao, Gang Yao, Lingyue Yin, Ge Zhang, Chun Zhang, Jianpeng Jiao, Zilong Zheng, and Yuan Gong. $onemillion-bench: How far are language agents from human experts? *CoRR*, abs/2603.07980, 2026. doi: 10.48550/ARXIV.2603.07980. URL https://doi.org/10.48550/arXiv.2603.07980.

[753] Rui Yang, Hanyang Chen, Junyu Zhang, Mark Zhao, Cheng Qian, Kangrui Wang, Qineng Wang, Teja Venkat Koripella, Marziyeh Movahedi, Manling Li, Heng Ji, Huan Zhang, and Tong Zhang. Embodiedbench: Comprehensive benchmarking multi-modal large language models for vision-driven embodied agents. *CoRR*, abs/2502.09560, 2025. doi: 10.48550/ARXIV.2502.09560. URL https://doi.org/10.48550/arXiv.2502.09560.

[754] Rui Yang, Michael Fu, Chakkrit Tantithamthavorn, Chetan Arora, Gunel Gulmammadova, and Joselito Joey Chua. Adaptiveguard: Towards adaptive runtime safety for llm-powered software. In *40th IEEE/ACM International Conference on Automated Software Engineering, ASE 2025, Seoul, Korea, Republic of, November 16-20, 2025*, pages 3381–3391. IEEE, 2025. doi: 10.1109/ASE63991.2025.00279. URL https://doi.org/10.1109/ASE63991.2025.00279.

[755] Sihan Yang, Runsen Xu, Yiman Xie, Sizhe Yang, Mo Li, Jingli Lin, Chenming Zhu, Xiaochen Chen, Haodong Duan, Xiangyu Yue, Dahua Lin, Tai Wang, and Jiangmiao Pang. Mmsi-bench: A benchmark for multi-image spatial intelligence. *CoRR*, abs/2505.23764, 2025. doi: 10.48550/ARXIV.2505.23764. URL https://doi.org/10.48550/arXiv.2505.23764.

[756] Xinyu Yang, Chenlong Deng, Tongyu Wen, Binyu Xie, and Zhicheng Dou. Lawthinker: A deep research legal agent in dynamic environments. *CoRR*, abs/2602.12056, 2026. doi: 10.48550/ARXIV.2602.12056. URL https://doi.org/10.48550/arXiv.2602.12056.

[757] Yifan Yang, Ziyang Gong, Weiquan Huang, Qihao Yang, Ziwei Zhou, Zisu Huang, Yan Li, Xuemei Gao, Qi Dai, Bei Liu, Kai Qiu, Yuqing Yang, Dongdong Chen, Xue Yang, and Chong Luo. Skillopt: Executive strategy for self-evolving agent skills. *CoRR*, abs/2605.23904, 2026. doi: 10.48550/ARXIV.2605.23904. URL https://doi.org/10.48550/arXiv.2605.23904.

[758] Yingxuan Yang, Huacan Chai, Yuanyi Song, Siyuan Qi, Muning Wen, Ning Li, Junwei Liao, Haoyi Hu, Jianghao Lin, Gaowei Chang, Weiwen Liu, Ying Wen, Yong Yu, and Weinan Zhang. A survey of AI agent protocols. *CoRR*, abs/2504.16736, 2025. doi: 10.48550/ARXIV.2504.16736. URL https://doi.org/10.48550/arXiv.2504.16736.

[759] Shunyu Yao, Howard Chen, John Yang, and Karthik Narasimhan. Webshop: Towards scalable real-world web interaction with grounded language agents. In Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022. URL http://papers.nips.cc/paper_files/paper/2022/hash/82ad13ec01f9fe44c01cb91814fd7b8c-Abstract-Conference.html.

[760] Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Tom Griffiths, Yuan Cao, and Karthik Narasimhan. Tree of thoughts: Deliberate problem solving with large language models. In Alice Oh, Tristan Naumann,

Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine, editors, *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/271db9922b8d1f4dd7aaef84ed5ac703-Abstract-Conference.html.

[761] Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R. Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023. URL https://openreview.net/forum?id=WE_vluYUL-X.

[762] Shunyu Yao, Noah Shinn, Pedram Razavi, and Karthik Narasimhan. τ-bench: A benchmark for tool-agent-user interaction in real-world domains. *CoRR*, abs/2406.12045, 2024. doi: 10.48550/ARXIV.2406.12045. URL https://doi.org/10.48550/arXiv.2406.12045.

[763] Yi Yao, He Zhu, Piaohong Wang, Jincheng Ren, Xinlong Yang, Qianben Chen, Xiaowan Li, Dingfeng Shi, Jiaxian Li, Qiexiang Wang, Sinuo Wang, Xinpeng Liu, Jiaqi Wu, Minghao Liu, and Wangchunshu Zhou. O-researcher: An open ended deep research model via multi-agent distillation and agentic RL. *CoRR*, abs/2601.03743, 2026. doi: 10.48550/ARXIV.2601.03743. URL https://doi.org/10.48550/arXiv.2601.03743.

[764] Bowen Ye, Rang Li, Qibin Yang, Yuanxin Liu, Linli Yao, Hanglong Lv, Zhihui Xie, Chenxin An, Lei Li, Lingpeng Kong, Qi Liu, Zhifang Sui, and Tong Yang. Claw-eval: Towards trustworthy evaluation of autonomous agents, 2026. URL https://arxiv.org/abs/2604.06132.

[765] Haoran Ye, Xuning He, Vincent Arak, Haonan Dong, and Guojie Song. Meta context engineering via agentic skill evolution. *CoRR*, abs/2601.21557, 2026. doi: 10.48550/ARXIV.2601.21557. URL https://doi.org/10.48550/arXiv.2601.21557.

[766] Jiabo Ye, Xi Zhang, Haiyang Xu, Haowei Liu, Junyang Wang, Zhaoqing Zhu, Ziwei Zheng, Feiyu Gao, Junjie Cao, Zhengxi Lu, Jitong Liao, Qi Zheng, Fei Huang, Jingren Zhou, and Ming Yan. Mobile-agent-v3: Fundamental agents for GUI automation. *CoRR*, abs/2508.15144, 2025. doi: 10.48550/ARXIV.2508.15144. URL https://doi.org/10.48550/arXiv.2508.15144.

[767] Jiasheng Ye, Peiju Liu, Tianxiang Sun, Jun Zhan, Yunhua Zhou, and Xipeng Qiu. Data mixing laws: Optimizing data mixtures by predicting language modeling performance. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=jjCB27TMK3.

[768] Rui Ye, Zhongwang Zhang, Kuan Li, Huifeng Yin, Zhengwei Tao, Yida Zhao, Liangcai Su, Liwen Zhang, Zile Qiao, Xinyu Wang, Pengjun Xie, Fei Huang, Siheng Chen, Jingren Zhou, and Yong Jiang. Agentfold: Long-horizon web agents with proactive context management. *CoRR*, abs/2510.24699, 2025. doi: 10.48550/ARXIV.2510.24699. URL https://doi.org/10.48550/arXiv.2510.24699.

[769] Seonghyeon Ye, Yunhao Ge, Kaiyuan Zheng, Shenyuan Gao, Sihyun Yu, George Kurian, Suneel Indupuru, You Liang Tan, Chuning Zhu, Jiannan Xiang, Ayaan Malik, Kyungmin Lee, William Liang, Nadun Ranawaka, Jiasheng Gu, Yinzhen Xu, Guanzhi Wang, Fengyuan Hu, Avnish Narayan, Johan Bjorck, Jing Wang, Gwanghyun Kim, Dantong Niu, Ruijie Zheng, Yuqi Xie, Jimmy Wu, Qi Wang, Ryan Julian, Danfei Xu, Yilun Du, Yevgen Chebotar, Scott Reed, Jan Kautz, Yuke Zhu, Linxi "Jim" Fan, and Joel Jang. World action models are zero-shot policies. *CoRR*, abs/2602.15922, 2026. doi: 10.48550/ARXIV.2602.15922. URL https://doi.org/10.48550/arXiv.2602.15922.

[770] Yixin Ye, Zhen Huang, Yang Xiao, Ethan Chern, Shijie Xia, and Pengfei Liu. Limo: Less is more for reasoning. *CoRR*, abs/2502.03387, 2025. doi: 10.48550/ARXIV.2502.03387. URL https://doi.org/10.48550/arXiv.2502.03387.

[771] Chun-Hsiao Yeh, Chenyu Wang, Shengbang Tong, Ta Ying Cheng, Ruoyu Wang, Tianzhe Chu, Yuexiang Zhai, Yubei Chen, Shenghua Gao, and Yi Ma. Seeing from another perspective: Evaluating multi-view understanding in mllms. In Sven Koenig, Chad Jenkins, and Matthew E. Taylor, editors, *Fortieth AAAI Conference on Artificial Intelligence, Thirty-Eighth Conference on Innovative Applications of Artificial Intelligence, Sixteenth Symposium on Educational Advances in Artificial Intelligence, AAAI 2026, Singapore, January 20-27, 2026*, pages 12000–12008. AAAI Press, 2026. doi: 10.1609/AAAI.V40I14.38188. URL https://doi.org/10.1609/aaai.v40i14.38188.

[772] Shuangshuang Ying, Zheyu Wang, Yunjian Peng, Jin Chen, Yuhao Wu, Hongbin Lin, Dingyu He, Siyi Liu, Gengchen Yu, YinZhu Piao, Yuchen Wu, Xin Gui, Zhongyuan Peng, Xin Li, Xeron Du, Libo Qin, Yixin Cao, Ge Zhang, and Stephen Huang. Retrieval-infused reasoning sandbox: A benchmark for decoupling

retrieval and reasoning capabilities. *CoRR*, abs/2601.21937, 2026. doi: 10.48550/ARXIV.2601.21937. URL https://doi.org/10.48550/arXiv.2601.21937.

[773] Chanwoong Yoon, Taewhoo Lee, Hyeon Hwang, Minbyul Jeong, and Jaewoo Kang. Compact: Compressing retrieved documents actively for question answering. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen, editors, *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, EMNLP 2024, Miami, FL, USA, November 12-16, 2024*, pages 21424–21439. Association for Computational Linguistics, 2024. doi: 10.18653/V1/2024.EMNLP-MAIN.1194. URL https://doi.org/10.18653/v1/2024.emnlp-main.1194.

[774] Ori Yoran, Samuel Joseph Amouyal, Chaitanya Malaviya, Ben Bogin, Ofir Press, and Jonathan Berant. Assistantbench: Can web agents solve realistic and time-consuming tasks? In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen, editors, *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, EMNLP 2024, Miami, FL, USA, November 12-16, 2024*, pages 8938–8968. Association for Computational Linguistics, 2024. doi: 10.18653/V1/2024.EMNLP-MAIN.505. URL https://doi.org/10.18653/v1/2024.emnlp-main.505.

[775] Runyang You, Hongru Cai, Caiqi Zhang, Qiancheng Xu, Meng Liu, Tiezheng Yu, Yongqi Li, and Wenjie Li. Agent-as-a-judge. *CoRR*, abs/2601.05111, 2026. doi: 10.48550/ARXIV.2601.05111. URL https://doi.org/10.48550/arXiv.2601.05111.

[776] Chao Yu, Akash Velu, Eugene Vinitsky, Jiaxuan Gao, Yu Wang, Alexandre M. Bayen, and Yi Wu. The surprising effectiveness of PPO in cooperative multi-agent games. In Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022. URL http://papers.nips.cc/paper_files/paper/2022/hash/9c1535a02f0ce079433344e14d910597-Abstract-Datasets_and_Benchmarks.html.

[777] Chaojia Yu, Zihan Cheng, Hanwen Cui, Yishuo Gao, Zexu Luo, Yijin Wang, Hangbin Zheng, and Yong Zhao. A survey on agent workflow - status and future. *CoRR*, abs/2508.01186, 2025. doi: 10.48550/ARXIV.2508.01186. URL https://doi.org/10.48550/arXiv.2508.01186.

[778] Hongli Yu, Tinghong Chen, Jiangtao Feng, Jiangjie Chen, Weinan Dai, Qiying Yu, Ya-Qin Zhang, Wei-Ying Ma, Jingjing Liu, Mingxuan Wang, and Hao Zhou. Memagent: Reshaping long-context LLM with multi-conv rl-based memory agent. *CoRR*, abs/2507.02259, 2025. doi: 10.48550/ARXIV.2507.02259. URL https://doi.org/10.48550/arXiv.2507.02259.

[779] Peijie Yu, Wei Liu, Yifan Yang, Jinjian Li, Zelong Zhang, Xiao Feng, and Feng Zhang. Benchmarking LLM tool-use in the wild. *CoRR*, abs/2604.06185, 2026. doi: 10.48550/ARXIV.2604.06185. URL https://doi.org/10.48550/arXiv.2604.06185.

[780] Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Juncai Liu, Lingjun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, Ru 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, Yonghui Wu, and Mingxuan Wang. DAPO: an open-source LLM reinforcement learning system at scale. In Danielle Belgrave, Cheng Zhang, Laura N. Montoya, Hsuan-Tien Lin, Razvan Pascanu, Piotr Koniusz, Marzyeh Ghassemi, Nancy Chen, Iván Vladimir Meza Ruíz, and Arturo Loaiza-Bonilla, editors, *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2025, NeurIPS 2025, San Diego, CA, USA, December 2-7, 2025 / Mexico City, Mexico, November 30 - December 5, 2025*, 2025. URL http://papers.nips.cc/paper_files/paper/2025/hash/a4277440d50f1f15d2cb4c14f7e0c0d2-Abstract-Conference.html.

[781] Tao Yu, Zhengbo Zhang, Zhiheng Lyu, Junhao Gong, Hongzhu Yi, Xinming Wang, Yuxuan Zhou, Jiabing Yang, Ping Nie, Yan Huang, and Wenhu Chen. Browseragent: Building web agents with human-inspired web browsing actions. *Trans. Mach. Learn. Res.*, 2026, 2026. URL https://openreview.net/forum?id=X4CfZPSEHE.

[782] Xinqiang Yu, Chuanguang Yang, Chengqing Yu, Libo Huang, Zhulin An, and Yongjun Xu. Online policy distillation with decision-attention. In *International Joint Conference on Neural Networks, IJCNN 2024, Yokohama, Japan, June 30 - July 5, 2024*, pages 1–8. IEEE, 2024. doi: 10.1109/IJCNN60899.2024.10651412. URL https://doi.org/10.1109/IJCNN60899.2024.10651412.

[783] Aojie Yuan, Zhiyuan Su, and Yue Zhao. AEGIS: no tool call left unchecked - A pre-execution firewall and audit layer for AI agents. *CoRR*, abs/2603.12621, 2026. doi: 10.48550/ARXIV.2603.12621. URL https://doi.org/10.48550/arXiv.2603.12621.

[784] Haoqi Yuan, Zhixuan Liang, Anzhe Chen, Ye Wang, Haoyang Li, Pei Lin, Yiyang Huang, Zixing Lei, Tong Zhang, Jiazhao Zhang, Jie Zhang, Jingyang Fan, Gengze Zhou, Qihang Peng, Chenxu Lv, Xiaoyue Chen, An Yang, Fei Huang, Junyang Lin, Dayiheng Liu, Jingren Zhou, Chenfei Wu, and Xiong-Hui Chen. Qwen-robotmanip technical report: Alignment unlocks scale for robotic manipulation foundation models. *CoRR*, abs/2606.17846, 2026. doi: 10.48550/ARXIV.2606.17846. URL https://doi.org/10.48550/arXiv.2606.17846.

[785] Huining Yuan, Zelai Xu, Huaijie Wang, Xiangmin Yi, Jiaxuan Gao, Xiao-Ping Zhang, Yu Wang, Chao Yu, and Yi Wu. Verifiable process rewards for agentic reasoning. *CoRR*, abs/2605.10325, 2026. doi: 10.48550/ARXIV.2605.10325. URL https://doi.org/10.48550/arXiv.2605.10325.

[786] Jingyang Yuan, Huazuo Gao, Damai Dai, Junyu Luo, Liang Zhao, Zhengyan Zhang, Zhenda Xie, Yuxing Wei, Lean Wang, Zhiping Xiao, Yuqing Wang, Chong Ruan, Ming Zhang, Wenfeng Liang, and Wangding Zeng. Native sparse attention: Hardware-aligned and natively trainable sparse attention. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar, editors, *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, pages 23078–23097. Association for Computational Linguistics, 2025. doi: 10.18653/V1/2025.ACL-LONG.1126. URL https://doi.org/10.18653/v1/2025.acl-long.1126.

[787] Lifan Yuan, Wendi Li, Huayu Chen, Ganqu Cui, Ning Ding, Kaiyan Zhang, Bowen Zhou, Zhiyuan Liu, and Hao Peng. Free process rewards without process labels. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu, editors, *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*, volume 267 of *Proceedings of Machine Learning Research*. PMLR / OpenReview.net, 2025. URL https://proceedings.mlr.press/v267/yuan25c.html.

[788] Qianhao Yuan, Jie Lou, Xing Yu, Hongyu Lin, Le Sun, Xianpei Han, and Yaojie Lu. Vision-opd: Learning to see fine details for multimodal llms via on-policy self-distillation. *CoRR*, abs/2605.18740, 2026. doi: 10.48550/ARXIV.2605.18740. URL https://doi.org/10.48550/arXiv.2605.18740.

[789] Siyu Yuan, Zehui Chen, Zhiheng Xi, Junjie Ye, Zhengyin Du, and Jiecao Chen. Agent-r: Training language model agents to reflect via iterative self-training. *CoRR*, abs/2501.11425, 2025. doi: 10.48550/ARXIV.2501.11425. URL https://doi.org/10.48550/arXiv.2501.11425.

[790] Tongxin Yuan, Zhiwei He, Lingzhong Dong, Yiming Wang, Ruijie Zhao, Tian Xia, Lizhen Xu, Binglin Zhou, Fangqi Li, Zhuosheng Zhang, Rui Wang, and Gongshen Liu. R-judge: Benchmarking safety risk awareness for LLM agents. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen, editors, *Findings of the Association for Computational Linguistics: EMNLP 2024, Miami, Florida, USA, November 12-16, 2024*, volume EMNLP 2024 of *Findings of ACL*, pages 1467–1490. Association for Computational Linguistics, 2024. doi: 10.18653/V1/2024.FINDINGS-EMNLP.79. URL https://doi.org/10.18653/v1/2024.findings-emnlp.79.

[791] Yanwei Yue, Guibin Zhang, Boyang Liu, Guancheng Wan, Kun Wang, Dawei Cheng, and Yiyan Qi. Masrouter: Learning to route llms for multi-agent systems. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar, editors, *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, pages 15549–15572. Association for Computational Linguistics, 2025. doi: 10.18653/V1/2025.ACL-LONG.757. URL https://doi.org/10.18653/v1/2025.acl-long.757.

[792] Manzil Zaheer, Guru Guruganesh, Kumar Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontañón, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, and Amr Ahmed. Big bird: Transformers for longer sequences. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/c8512d142a2d849725f31a9a7a361ab9-Abstract.html.

[793] Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah D. Goodman. Star: Bootstrapping reasoning with reasoning. In Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022. URL http://papers.nips.cc/paper_files/paper/2022/hash/639a9a172c044fbb64175b5fad42e9a5-Abstract-Conference.html.

[794] Aohan Zeng, Mingdao Liu, Rui Lu, Bowen Wang, Xiao Liu, Yuxiao Dong, and Jie Tang. Agenttuning: Enabling generalized agent abilities for llms. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Findings of the Association for Computational Linguistics, ACL 2024, Bangkok, Thailand and virtual meeting, August 11-16,*

2024, volume ACL 2024 of *Findings of ACL*, pages 3053–3077. Association for Computational Linguistics, 2024. doi: 10.18653/V1/2024.FINDINGS-ACL.181. URL https://doi.org/10.18653/v1/2024.findings-acl.181.

[795] Aohan Zeng, Bin Xu, Bowen Wang, Chenhui Zhang, Da Yin, Diego Rojas, Guanyu Feng, Hanlin Zhao, Hanyu Lai, Hao Yu, Hongning Wang, Jiadai Sun, Jiajie Zhang, Jiale Cheng, Jiayi Gui, Jie Tang, Jing Zhang, Juanzi Li, Lei Zhao, Lindong Wu, Lucen Zhong, Mingdao Liu, Minlie Huang, Peng Zhang, Qinkai Zheng, Rui Lu, Shuaiqi Duan, Shudan Zhang, Shulin Cao, Shuxun Yang, Weng Lam Tam, Wenyi Zhao, Xiao Liu, Xiao Xia, Xiaohan Zhang, Xiaotao Gu, Xin Lv, Xinghan Liu, Xinyi Liu, Xinyue Yang, Xixuan Song, Xunkai Zhang, Yifan An, Yifan Xu, Yilin Niu, Yuantao Yang, Yueyan Li, Yushi Bai, Yuxiao Dong, Zehan Qi, Zhaoyu Wang, Zhen Yang, Zhengxiao Du, Zhenyu Hou, and Zihan Wang. Chatglm: A family of large language models from GLM-130B to GLM-4 all tools. *CoRR*, abs/2406.12793, 2024. doi: 10.48550/ARXIV.2406.12793. URL https://doi.org/10.48550/arXiv.2406.12793.

[796] Xiangyu Zeng, Kefan Qiu, Qingyu Zhang, Xinhao Li, Jing Wang, Jiaxin Li, Ziang Yan, Kun Tian, Meng Tian, Xinhai Zhao, Yi Wang, and Limin Wang. Streamforest: Efficient online video understanding with persistent event memory. In Danielle Belgrave, Cheng Zhang, Laura N. Montoya, Hsuan-Tien Lin, Razvan Pascanu, Piotr Koniusz, Marzyeh Ghassemi, Nancy Chen, Iván Vladimir Meza Ruíz, and Arturo Loaiza-Bonilla, editors, *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2025, NeurIPS 2025, San Diego, CA, USA, December 2-7, 2025 / Mexico City, Mexico, November 30 - December 5, 2025*, 2025. URL http://papers.nips.cc/paper_files/paper/2025/hash/6dd91fec726dbed8915a1fbadd91d1d2-Abstract-Conference.html.

[797] Yucheng Zeng, Shupeng Li, Daxiang Dong, Ruijie Xu, Zimo Chen, Liwei Zheng, Yuxuan Li, Zhe Zhou, Haotian Zhao, Lun Tian, Heng Xiao, Tianshu Zhu, Longkun Hao, and Jianmin Wu. Swe-hub: A unified production system for scalable, executable software engineering tasks. *CoRR*, abs/2603.00575, 2026. doi: 10.48550/ARXIV.2603.00575. URL https://doi.org/10.48550/arXiv.2603.00575.

[798] Zhiyuan Zeng, Hamish Ivison, Yiping Wang, Lifan Yuan, Shuyue Stella Li, Zhuorui Ye, Siting Li, Jacqueline He, Runlong Zhou, Tong Chen, Chenyang Zhao, Yulia Tsvetkov, Simon Shaolei Du, Natasha Jaques, Hao Peng, Pang Wei Koh, and Hannaneh Hajishirzi. RLVE: scaling up reinforcement learning for language models with adaptive verifiable environments. *CoRR*, abs/2511.07317, 2025. doi: 10.48550/ARXIV.2511.07317. URL https://doi.org/10.48550/arXiv.2511.07317.

[799] Yunpeng Zhai, Shuchang Tao, Cheng Chen, Anni Zou, Ziqian Chen, Qingxu Fu, Shinji Mai, Li Yu, Jiaji Deng, Zouying Cao, Zhaoyang Liu, Bolin Ding, and Jingren Zhou. Agentevolver: Towards efficient self-evolving agent system. *CoRR*, abs/2511.10395, 2025. doi: 10.48550/ARXIV.2511.10395. URL https://doi.org/10.48550/arXiv.2511.10395.

[800] Bo Zhang, Shiyang Feng, Xiangchao Yan, Jiakang Yuan, Zhiyin Yu, Xiaohan He, Songtao Huang, Shaowei Hou, Zheng Nie, Zhilong Wang, Jinyao Liu, Runmin Ma, Tianshuo Peng, Peng Ye, Dongzhan Zhou, Shufei Zhang, Xiaosong Wang, Yilan Zhang, Meng Li, Zhongying Tu, Xiangyu Yue, Wangli Ouyang, Bowen Zhou, and Lei Bai. Novelseek: When agent becomes the scientist - building closed-loop system from hypothesis to verification. *CoRR*, abs/2505.16938, 2025. doi: 10.48550/ARXIV.2505.16938. URL https://doi.org/10.48550/arXiv.2505.16938.

[801] Chenchen Zhang, Yuhang Li, Can Xu, Jiaheng Liu, Ao Liu, Shihui Hu, Dengpeng Wu, Guanhua Huang, Kejiao Li, Qi Yi, Ruibin Xiong, Haotian Zhu, Yuanxing Zhang, Yuhao Jiang, Yue Zhang, Zenan Xu, Bohui Zhai, Guoxiang He, Hebin Li, Jie Zhao, Le Zhang, Lingyun Tan, Pengyu Guo, Xianshu Pang, Yang Ruan, Zhifeng Zhang, Zhonghu Wang, Ziyan Xu, Zuopu Yin, Wiggin Zhou, Chayse Zhou, and Fengzong Lian. Artifactsbench: Bridging the visual-interactive gap in LLM code generation evaluation. *CoRR*, abs/2507.04952, 2025. doi: 10.48550/ARXIV.2507.04952. URL https://doi.org/10.48550/arXiv.2507.04952.

[802] Chenghao Zhang, Guanting Dong, Xinyu Yang, and Zhicheng Dou. Towards mixed-modal retrieval for universal retrieval-augmented generation. *CoRR*, abs/2510.17354, 2025. doi: 10.48550/ARXIV.2510.17354. URL https://doi.org/10.48550/arXiv.2510.17354.

[803] Chenghao Zhang, Guanting Dong, Yufan Liu, Tong Zhao, and Zhicheng Dou. Towards verifiable multimodal deep research: A multi-agent harness for interleaved report generation. *CoRR*, abs/2605.29861, 2026. doi: 10.48550/ARXIV.2605.29861. URL https://doi.org/10.48550/arXiv.2605.29861.

[804] Chenxi Zhang, Ziliang Gan, Liyun Zhu, Youwei Pang, Qing Zhang, and Rongjunchen Zhang. Finmtm: A multi-turn multimodal benchmark for financial reasoning and agent evaluation. *CoRR*, abs/2602.03130, 2026. doi: 10.48550/ARXIV.2602.03130. URL https://doi.org/10.48550/arXiv.2602.03130.

[805] Dan Zhang, Sining Zhoubian, Ziniu Hu, Yisong Yue, Yuxiao Dong, and Jie Tang. Rest-mcts*: LLM self-training via process reward guided tree search. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan,

Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang, editors, *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. URL http://papers.nips.cc/paper_files/paper/2024/hash/76ec4dc30e9faaf0e4b6093eaa377218-Abstract-Conference.html.

[806] Fan Zhang, Vireo Zhang, Shengju Qian, Haoxuan Li, Hao Wu, Jinyang Wu, Donghao Zhou, Zhihong Zhu, Zheng Lian, Xin Wang, and Pheng-Ann Heng. Orchestra-o1: Omnimodal agent orchestration. *CoRR*, abs/2606.13707, 2026. doi: 10.48550/ARXIV.2606.13707. URL https://doi.org/10.48550/arXiv.2606.13707.

[807] Guibin Zhang, Muxin Fu, Guancheng Wan, Miao Yu, Kun Wang, and Shuicheng Yan. G-memory: Tracing hierarchical memory for multi-agent systems. *CoRR*, abs/2506.07398, 2025. doi: 10.48550/ARXIV.2506.07398. URL https://doi.org/10.48550/arXiv.2506.07398.

[808] Guibin Zhang, Hejia Geng, Xiaohang Yu, Zhenfei Yin, Zaibin Zhang, Zelin Tan, Heng Zhou, Zhong-Zhi Li, Xiangyuan Xue, Yijiang Li, Yifan Zhou, Yang Chen, Chen Zhang, Yutao Fan, Zihu Wang, Songtao Huang, Francisco Piedrahita Velez, Yue Liao, Hongru Wang, Mengyue Yang, Heng Ji, Jun Wang, Shuicheng Yan, Philip Torr, and Lei Bai. The landscape of agentic reinforcement learning for llms: A survey. *Trans. Mach. Learn. Res.*, 2026, 2026. URL https://openreview.net/forum?id=RY19y2RI1O.

[809] Guibin Zhang, Xun Xu, Yanwei Yue, Zikun Su, Wangchunshu Zhou, Xiaohu You, and Shuicheng Yan. Opd-evolver: Cultivating holistic agent evolver via on-policy distillation. *CoRR*, abs/2606.17628, 2026. doi: 10.48550/ARXIV.2606.17628. URL https://doi.org/10.48550/arXiv.2606.17628.

[810] Guibin Zhang, Haiyang Yu, Kaiming Yang, Bingli Wu, Fei Huang, Yongbin Li, and Shuicheng Yan. Evoroute: Experience-driven self-routing LLM agent systems. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens, editors, *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2026, San Diego, California, United States, July 2-7, 2026*, pages 38213–38225. Association for Computational Linguistics, 2026. URL https://aclanthology.org/2026.acl-long.1771/.

[811] Hanchen Zhang, Xiao Liu, Bowen Lv, Xueqiao Sun, Bohao Jing, Iat Long Iong, Zhenyu Hou, Zehan Qi, Hanyu Lai, Yifan Xu, Rui Lu, Hongning Wang, Jie Tang, and Yuxiao Dong. Agentrl: Scaling agentic reinforcement learning with a multi-turn, multi-task framework. *CoRR*, abs/2510.04206, 2025. doi: 10.48550/ARXIV.2510.04206. URL https://doi.org/10.48550/arXiv.2510.04206.

[812] Hang Zhang, Ruheng Wang, Yuelyu Ji, Min Gu Kwak, Xizhi Wu, Chenyu Li, Li Zhang, Wenqi Shi, Yifan Peng, and Yanshan Wang. Scaling medical reasoning verification via tool-integrated reinforcement learning. *CoRR*, abs/2601.20221, 2026. doi: 10.48550/ARXIV.2601.20221. URL https://doi.org/10.48550/arXiv.2601.20221.

[813] Haozhen Zhang, Quanyu Long, Jianzhu Bao, Tao Feng, Weizhi Zhang, Haodong Yue, and Wenya Wang. Memskill: Learning and evolving memory skills for self-evolving agents. *CoRR*, abs/2602.02474, 2026. doi: 10.48550/ARXIV.2602.02474. URL https://doi.org/10.48550/arXiv.2602.02474.

[814] Jenny Zhang, Shengran Hu, Cong Lu, Robert T. Lange, and Jeff Clune. Darwin godel machine: Open-ended evolution of self-improving agents. *CoRR*, abs/2505.22954, 2025. doi: 10.48550/ARXIV.2505.22954. URL https://doi.org/10.48550/arXiv.2505.22954.

[815] Jiaxin Zhang, Prafulla Kumar Choubey, Kung-Hsiang Huang, Caiming Xiong, and Chien-Sheng Wu. Agentic uncertainty quantification. *CoRR*, abs/2601.15703, 2026. doi: 10.48550/ARXIV.2601.15703. URL https://doi.org/10.48550/arXiv.2601.15703.

[816] Jiayi Zhang, Jinyu Xiang, Zhaoyang Yu, Fengwei Teng, Xionghui Chen, Jiaqi Chen, Mingchen Zhuge, Xin Cheng, Sirui Hong, Jinlin Wang, Bingnan Zheng, Bang Liu, Yuyu Luo, and Chenglin Wu. Aflow: Automating agentic workflow generation. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=z5uVAKwmjf.

[817] Jiazhao Zhang, Gengze Zhou, Hale Yin, Yiyang Huang, Zixing Lei, Qihang Peng, Haoqi Yuan, Jie Zhang, Xudong Guo, Xiaoyue Chen, An Yang, Fei Huang, Zhibo Yang, Junyang Lin, Dayiheng Liu, Jingren Zhou, Zhuoyuan Yu, Jingyang Fan, Zhixuan Liang, Pei Lin, Ye Wang, Haoyang Li, Anzhe Chen, Kun Yan, Xiao Xu, Jiahao Li, Lulu Hu, Minying Zhang, Shurui Li, Wenhu Xiao, Shuai Bai, Xuancheng Ren, Chenxu Lv, Chenfei Wu, and Xiong-Hui Chen. Qwen-robotnav technical report: A scalable navigation model designed for an agentic navigation system. *CoRR*, abs/2606.18112, 2026. doi: 10.48550/ARXIV.2606.18112. URL https://doi.org/10.48550/arXiv.2606.18112.

[818] Jiazheng Zhang, Ziche Fu, Zhiheng Xi, Wenqing Jing, Mingxu Chai, Wei He, Guoqiang Zhang, Chenghao Fan, Chenxin An, Wenxiang Chen, Zhicheng Liu, Haojie Pan, Dingwei Zhu, Tao Gui, Qi Zhang, and Xuanjing Huang.

Agentv-rl: Scaling reward modeling with agentic verifier. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens, editors, *Findings of the Association for Computational Linguistics, ACL 2026, San Diego, California, United States, July 2-7, 2026*, pages 23078–23100. Association for Computational Linguistics, 2026. URL https://aclanthology.org/2026.findings-acl.1156/.

[819] Junkai Zhang, Zihao Wang, Lin Gui, Swarnashree Mysore Sathyendra, Jaehwan Jeong, Victor Veitch, Wei Wang, Yunzhong He, Bing Liu, and Lifeng Jin. Chasing the tail: Effective rubric-based reward modeling for large language model post-training. *CoRR*, abs/2509.21500, 2025. doi: 10.48550/ARXIV.2509.21500. URL https://doi.org/10.48550/arXiv.2509.21500.

[820] JunShuo Zhang, Chengrui Huang, Feng Guo, Zihan Li, Ke Shi, Menghua Jiang, Jiguo Yu, Shuo Shang, and Shen Gao. DPEPO: diverse parallel exploration policy optimization for llm-based agents. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens, editors, *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2026, San Diego, California, United States, July 2-7, 2026*, pages 46367–46389. Association for Computational Linguistics, 2026. URL https://aclanthology.org/2026.acl-long.2151/.

[821] Kaiyan Zhang, Yuxin Zuo, Bingxiang He, Youbang Sun, Runze Liu, Che Jiang, Yuchen Fan, Kai Tian, Guoli Jia, Pengfei Li, Yu Fu, Xingtai Lv, Yuchen Zhang, Sihang Zeng, Shang Qu, Haozhan Li, Shijie Wang, Yuru Wang, Xinwei Long, Fangfu Liu, Xiang Xu, Jiaze Ma, Xuekai Zhu, Ermo Hua, Yihao Liu, Zonglin Li, Huayu Chen, Xiaoye Qu, Yafu Li, Weize Chen, Zhenzhao Yuan, Junqi Gao, Dong Li, Zhiyuan Ma, Ganqu Cui, Zhiyuan Liu, Biqing Qi, Ning Ding, and Bowen Zhou. A survey of reinforcement learning for large reasoning models. *CoRR*, abs/2509.08827, 2025. doi: 10.48550/ARXIV.2509.08827. URL https://doi.org/10.48550/arXiv.2509.08827.

[822] Lunjun Zhang, Arian Hosseini, Hritik Bansal, Mehran Kazemi, Aviral Kumar, and Rishabh Agarwal. Generative verifiers: Reward modeling as next-token prediction. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=Ccwp4tFEtE.

[823] Mingda Zhang, Haoran Luo, Tiesunlong Shen, Qika Lin, Xiaoying Tang, Rui Mao, and Erik Cambria. Flowsteer: Interactive agentic workflow orchestration via end-to-end reinforcement learning. *CoRR*, abs/2602.01664, 2026. doi: 10.48550/ARXIV.2602.01664. URL https://doi.org/10.48550/arXiv.2602.01664.

[824] Qizheng Zhang, Changran Hu, Shubhangi Upasani, Boyuan Ma, Fenglu Hong, Vamsidhar Kamanuru, Jay Rainton, Chen Wu, Mengmeng Ji, Hanchen Li, Urmish Thakker, James Zou, and Kunle Olukotun. Agentic context engineering: Evolving contexts for self-improving language models. *CoRR*, abs/2510.04618, 2025. doi: 10.48550/ARXIV.2510.04618. URL https://doi.org/10.48550/arXiv.2510.04618.

[825] Ruiyang Zhang, Qianguo Sun, Chao Song, Yiyan Qi, and Zhedong Zheng. Vsearcher: Long-horizon multimodal search agent via reinforcement learning. *CoRR*, abs/2603.02795, 2026. doi: 10.48550/ARXIV.2603.02795. URL https://doi.org/10.48550/arXiv.2603.02795.

[826] Tuo Zhang, Alin-Ionut Popa, Yan Xu, Rui Song, and Dimitrios Dimitriadis. PIVOT: bridging planning and execution in LLM agents via trajectory refinement. *CoRR*, abs/2605.11225, 2026. doi: 10.48550/ARXIV.2605.11225. URL https://doi.org/10.48550/arXiv.2605.11225.

[827] Wenlin Zhang, Xiaopeng Li, Yingyi Zhang, Pengyue Jia, Yichao Wang, Huifeng Guo, Yong Liu, and Xiangyu Zhao. Deep research: A survey of autonomous research agents. *CoRR*, abs/2508.12752, 2025. doi: 10.48550/ARXIV.2508.12752. URL https://doi.org/10.48550/arXiv.2508.12752.

[828] Wentao Zhang, Ce Cui, Yilei Zhao, Rui Hu, Yang Liu, Yahui Zhou, and Bo An. Agentorchestra: A hierarchical multi-agent framework for general-purpose task solving. *CoRR*, abs/2506.12508, 2025. doi: 10.48550/ARXIV.2506.12508. URL https://doi.org/10.48550/arXiv.2506.12508.

[829] Xiaoyi Zhang, Zhaoyang Jia, Zongyu Guo, Jiahao Li, Bin Li, Houqiang Li, and Yan Lu. Deep video discovery: Agentic search with tool use for long-form video understanding. In Danielle Belgrave, Cheng Zhang, Laura N. Montoya, Hsuan-Tien Lin, Razvan Pascanu, Piotr Koniusz, Marzyeh Ghassemi, Nancy Chen, Iván Vladimir Meza Ruíz, and Arturo Loaiza-Bonilla, editors, *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2025, NeurIPS 2025, San Diego, CA, USA, December 2-7, 2025 / Mexico City, Mexico, November 30 - December 5, 2025*, 2025. URL http://papers.nips.cc/paper_files/paper/2025/hash/8190b210e9808e54ee16263b673a847d-Abstract-Conference.html.

[830] Xiaoyun Zhang, Xiaojian Yuan, Di Huang, Wang You, Chen Hu, Jingqing Ruan, Kejiang Chen, and Xing Hu. Revisiting entropy regularization: Adaptive coefficient unlocks its potential for LLM reinforcement learning. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens, editors, *Findings of the Association for*

Computational Linguistics, ACL 2026, San Diego, California, United States, July 2-7, 2026, pages 18005–18020. Association for Computational Linguistics, 2026. URL https://aclanthology.org/2026.findings-acl.895/.

[831] Xinchen Zhang, Xiaoying Zhang, Youbin Wu, Yanbin Cao, Renrui Zhang, Ruihang Chu, Ling Yang, and Yujiu Yang. Generative universal verifier as multimodal meta-reasoner. *CoRR*, abs/2510.13804, 2025. doi: 10.48550/ARXIV.2510.13804. URL https://doi.org/10.48550/arXiv.2510.13804.

[832] Xing Zhang, Yanwei Cui, Guanghui Wang, Wei Qiu, Ziyuan Li, Fangwei Han, Yajing Huang, Hengzhi Qiu, Bing Zhu, and Peiyang He. Verified multi-agent orchestration: A plan-execute-verify-replan framework for complex query resolution. *CoRR*, abs/2603.11445, 2026. doi: 10.48550/ARXIV.2603.11445. URL https://doi.org/10.48550/arXiv.2603.11445.

[833] Xinrong Zhang, Yingfa Chen, Shengding Hu, Zihang Xu, Junhao Chen, Moo Khai Hao, Xu Han, Zhen Leng Thai, Shuo Wang, Zhiyuan Liu, and Maosong Sun. ∞bench: Extending long context evaluation beyond 100k tokens. *CoRR*, abs/2402.13718, 2024. doi: 10.48550/ARXIV.2402.13718. URL https://doi.org/10.48550/arXiv.2402.13718.

[834] Yabo Zhang, Yihan Zeng, Qingyun Li, Zhen Hu, Kavin Han, and Wangmeng Zuo. Tool-r1: Sample-efficient reinforcement learning for agentic tool use. *CoRR*, abs/2509.12867, 2025. doi: 10.48550/ARXIV.2509.12867. URL https://doi.org/10.48550/arXiv.2509.12867.

[835] Yanfei Zhang, Xu Lin, and Chenglin Wu. Stepopsd: Step-aware online preference distillation for agent reinforcement learning. *CoRR*, abs/2605.27140, 2026. doi: 10.48550/ARXIV.2605.27140. URL https://doi.org/10.48550/arXiv.2605.27140.

[836] Yao Zhang, Chenyang Lin, Shijie Tang, Haokun Chen, Shijie Zhou, Yunpu Ma, and Volker Tresp. Swarmagentic: Towards fully automated agentic system generation via swarm intelligence. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng, editors, *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, EMNLP 2025, Suzhou, China, November 4-9, 2025*, pages 1778–1818. Association for Computational Linguistics, 2025. doi: 10.18653/V1/2025.EMNLP-MAIN.93. URL https://doi.org/10.18653/v1/2025.emnlp-main.93.

[837] Yaocheng Zhang, Haohuan Huang, Zijun Song, Zijie Zhao, Qichao Zhang, Yuanheng Zhu, and Dongbin Zhao. Criticsearch: Fine-grained credit assignment for search agents via a retrospective critic. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens, editors, *Findings of the Association for Computational Linguistics, ACL 2026, San Diego, California, United States, July 2-7, 2026*, pages 12272–12290. Association for Computational Linguistics, 2026. URL https://aclanthology.org/2026.findings-acl.596/.

[838] Yaocheng Zhang, Yuanheng Zhu, Wenyue Chong, Songjun Tu, Qichao Zhang, Jiajun Chai, Xiaohan Wang, Wei Lin, Guojun Yin, and Dongbin Zhao. π-play: Multi-agent self-play via privileged self-distillation without external data. *CoRR*, abs/2604.14054, 2026. doi: 10.48550/ARXIV.2604.14054. URL https://doi.org/10.48550/arXiv.2604.14054.

[839] Yifan Zhang, Chunli Peng, Boyang Wang, Puyi Wang, Qingcheng Zhu, Fei Kang, Biao Jiang, Zedong Gao, Eric Li, Yang Liu, and Yahui Zhou. Matrix-game: Interactive world foundation model. *CoRR*, abs/2506.18701, 2025. doi: 10.48550/ARXIV.2506.18701. URL https://doi.org/10.48550/arXiv.2506.18701.

[840] Yinger Zhang, Shutong Jiang, Renhao Li, Jianhong Tu, Yang Su, Lianghao Deng, Xudong Guo, Chenxu Lv, and Junyang Lin. Deepplanning: Benchmarking long-horizon agentic planning with verifiable constraints. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens, editors, *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2026, San Diego, California, United States, July 2-7, 2026*, pages 7377–7407. Association for Computational Linguistics, 2026. URL https://aclanthology.org/2026.acl-long.335/.

[841] Yongheng Zhang, Ziang Liu, Jiaxuan Zhu, Shuai Wang, Xiangqi Chen, Haojing Huang, Jiayi Kuang, Siyu Chen, Ao Shen, Hao Wu, Qiufeng Wang, Qian-Wen Zhang, Junnan Dong, Wenhao Jiang, Ying Shen, Hai-Tao Zheng, Yinghui Li, Di Yin, Xing Sun, and Philip S. Yu. From chatbot to digital colleague: The paradigm shift toward persistent autonomous AI. *CoRR*, abs/2606.14502, 2026. doi: 10.48550/ARXIV.2606.14502. URL https://doi.org/10.48550/arXiv.2606.14502.

[842] Yu Zhang, Zongyu Lin, Xingcheng Yao, Jiaxi Hu, Fanqing Meng, Chengyin Liu, Xin Men, Songlin Yang, Zhiyuan Li, Wentao Li, Enzhe Lu, Weizhou Liu, Yanru Chen, Weixin Xu, Longhui Yu, Yejie Wang, Yu Fan, Longguang Zhong, Enming Yuan, Dehao Zhang, Yizhi Zhang, T. Y. Liu, Haiming Wang, Shengjun Fang, Weiran He, Shaowei Liu, Yiwei Li, Jianlin Su, Jiezhong Qiu, Bo Pang, Junjie Yan, Zhejun Jiang, Weixiao Huang, Bohong Yin, Jiacheng You, Chu Wei, Zhengtao Wang, Chao Hong, Yutian Chen, Guanduo Chen, Yucheng Wang, Huabin

Zheng, Feng Wang, Yibo Liu, Mengnan Dong, Zheng Zhang, Siyuan Pan, Wenhao Wu, Yuhao Wu, Longyu Guan, Jiawen Tao, Guohong Fu, Xinran Xu, Yuzhi Wang, Guokun Lai, Yuxin Wu, Xinyu Zhou, Zhilin Yang, and Yulun Du. Kimi linear: An expressive, efficient attention architecture. *CoRR*, abs/2510.26692, 2025. doi: 10.48550/ARXIV.2510.26692. URL https://doi.org/10.48550/arXiv.2510.26692.

[843] Yuntong Zhang, Haifeng Ruan, Zhiyu Fan, and Abhik Roychoudhury. Autocoderover: Autonomous program improvement. In Maria Christakis and Michael Pradel, editors, *Proceedings of the 33rd ACM SIGSOFT International Symposium on Software Testing and Analysis, ISSTA 2024, Vienna, Austria, September 16-20, 2024*, pages 1592–1604. ACM, 2024. doi: 10.1145/3650212.3680384. URL https://doi.org/10.1145/3650212.3680384.

[844] Yunxiao Zhang, Guanming Xiong, Haochen Li, and Wen Zhao. EDGE: efficient data selection for LLM agents via guideline effectiveness. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2025, Montreal, Canada, August 16-22, 2025*, pages 8384–8392. ijcai.org, 2025. doi: 10.24963/IJCAI.2025/932. URL https://doi.org/10.24963/ijcai.2025/932.

[845] Yuwei Zhang, Sha Li, Changlong Yu, Qin Lu, Shuowei Jin, Chengyu Dong, Haoran Liu, Ilgee Hong, Xintong Li, Zhenyu Shi, Bing Yin, and Jingbo Shang. Learning with rare success but rich feedback via reflection-enhanced self-distillation. *CoRR*, abs/2605.12741, 2026. doi: 10.48550/ARXIV.2605.12741. URL https://doi.org/10.48550/arXiv.2605.12741.

[846] Yuxiang Zhang, Jiangming Shu, Ye Ma, Xueyuan Lin, Shangxi Wu, and Jitao Sang. Memory as action: Autonomous context curation for long-horizon agentic tasks. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens, editors, *Findings of the Association for Computational Linguistics, ACL 2026, San Diego, California, United States, July 2-7, 2026*, pages 19149–19164. Association for Computational Linguistics, 2026. URL https://aclanthology.org/2026.findings-acl.956/.

[847] Yuxuan Zhang, Yubo Wang, Yipeng Zhu, Penghui Du, Junwen Miao, Xuan Lu, Wendong Xu, Yunzhuo Hao, Songcheng Cai, Xiaochen Wang, Huaisong Zhang, Xian Wu, Yi Lu, Minyi Lei, Kai Zou, Huifeng Yin, Ping Nie, Liang Chen, Dongfu Jiang, Wenhu Chen, and Kelsey R. Allen. Clawbench: Can AI agents complete everyday online tasks? *CoRR*, abs/2604.08523, 2026. doi: 10.48550/ARXIV.2604.08523. URL https://doi.org/10.48550/arXiv.2604.08523.

[848] Zeyu Zhang, Quanyu Dai, Xiaohe Bo, Chen Ma, Rui Li, Xu Chen, Jieming Zhu, Zhenhua Dong, and Ji-Rong Wen. A survey on the memory mechanism of large language model-based agents. *ACM Trans. Inf. Syst.*, 43(6): 155:1–155:47, 2025. doi: 10.1145/3748302. URL https://doi.org/10.1145/3748302.

[849] Zhexin Zhang, Shiyao Cui, Yida Lu, Jingzhuo Zhou, Junxiao Yang, Hongning Wang, and Minlie Huang. Agent-safetybench: Evaluating the safety of LLM agents. *CoRR*, abs/2412.14470, 2024. doi: 10.48550/ARXIV.2412.14470. URL https://doi.org/10.48550/arXiv.2412.14470.

[850] Zhiwei Zhang, Xiaomin Li, Yudi Lin, Hui Liu, Ramraj Chandradevan, Linlin Wu, Minhua Lin, Fali Wang, Xianfeng Tang, Qi He, and Suhang Wang. Unlocking the power of multi-agent LLM for reasoning: From lazy agents to deliberation. *CoRR*, abs/2511.02303, 2025. doi: 10.48550/ARXIV.2511.02303. URL https://doi.org/10.48550/arXiv.2511.02303.

[851] Zhuosheng Zhang, Aston Zhang, Mu Li, and Alex Smola. Automatic chain of thought prompting in large language models. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023. URL https://openreview.net/forum?id=5NTt8GFjUHkr.

[852] Ziyun Zhang, Zezhou Wang, Xiaoyi Zhang, Zongyu Guo, Jiahao Li, Bin Li, and Yan Lu. Infiniteweb: Scalable web environment synthesis for GUI agent training. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens, editors, *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2026, San Diego, California, United States, July 2-7, 2026*, pages 28465–28492. Association for Computational Linguistics, 2026. URL https://aclanthology.org/2026.acl-long.1313/.

[853] Andrew Zhao, Daniel Huang, Quentin Xu, Matthieu Lin, Yong-Jin Liu, and Gao Huang. Expel: LLM agents are experiential learners. In Michael J. Wooldridge, Jennifer G. Dy, and Sriraam Natarajan, editors, *Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2024, Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence, IAAI 2024, Fourteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2024, February 20-27, 2024, Vancouver, Canada*, pages 19632–19642. AAAI Press, 2024. doi: 10.1609/AAAI.V38I17.29936. URL https://doi.org/10.1609/aaai.v38i17.29936.

[854] Andrew Zhao, Yiran Wu, Tong Wu, Quentin Xu, Yang Yue, Matthieu Lin, Shenzhi Wang, Qingyun Wu, Zilong Zheng, and Gao Huang. Absolute zero: Reinforced self-play reasoning with zero data. In Danielle Belgrave, Cheng Zhang, Laura N. Montoya, Hsuan-Tien Lin, Razvan Pascanu, Piotr Koniusz, Marzyeh Ghassemi, Nancy

Chen, Iván Vladimir Meza Ruíz, and Arturo Loaiza-Bonilla, editors, *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2025, NeurIPS 2025, San Diego, CA, USA, December 2-7, 2025 / Mexico City, Mexico, November 30 - December 5, 2025*, 2025. URL http://papers.nips.cc/paper_files/paper/2025/hash/9837dc00ff67d176373268ed48042d49-Abstract-Conference.html.

[855] Bing Zhao, Chenfei Wu, Deqing Li, Hao Meng, Jiahao Li, Jie Zhang, Jingren Zhou, Junyang Lin, Kaiyuan Gao, Kuan Cao, Kun Yan, Liang Peng, Lihan Jiang, Niantong Li, Ningyuan Tang, Shengming Yin, Tianhe Wu, Xiao Xu, Xiaoyue Chen, Xihua Wang, Yan Shu, Yanran Zhang, Yi Wang, Yilei Chen, Ying Ba, Yixian Xu, Yujia Wu, Yuxiang Chen, Zecheng Tang, Zekai Zhang, Zhendong Wang, Zihao Liu, Zikai Zhou, An Yang, Chen Cheng, Chenxu Lv, Dayiheng Liu, Fan Zhou, Hantian Xiong, Hongzhu Shi, Hu Wei, Huihong Zhao, Ivy Liu, Jianwei Zhang, Jiawei Zhang, Kai Chen, Kang He, Levon Xue, Lin Qu, Linhan Tang, Luwen Feng, Minggang Wu, Minmin Sun, Na Ni, Rui Men, Shuai Bai, Sishou Zheng, Tao Lan, Tianqi Zhang, Tingkun Wen, Wei Wang, Weixu Qiao, Weiyi Lu, Wenmeng Zhou, Xiaodong Deng, Xiaoxiao Xu, Xinlei Fang, Xionghui Chen, Yanan Wang, Yang Fan, Yichang Zhang, Yixuan Xu, Yu Wu, Zhiyuan Ma, and Zhizhi Cai. Qwen-image-2.0 technical report, 2026. URL https://arxiv.org/abs/2605.10730.

[856] Jiale Zhao, Guoxin Chen, Fanzhe Meng, Minghao Li, Jie Chen, Hui Xu, Yongshuai Sun, Xin Zhao, Ruihua Song, Yuan Zhang, Peng Wang, Cheng Chen, Jirong Wen, and Kai Jia. Immersion in the github universe: Scaling coding agents to mastery. *CoRR*, abs/2602.09892, 2026. doi: 10.48550/ARXIV.2602.09892. URL https://doi.org/10.48550/arXiv.2602.09892.

[857] Jian Zhao, Runze Liu, Kaiyan Zhang, Zhimu Zhou, Junqi Gao, Dong Li, Jiafei Lyu, Zhouyi Qian, Biqing Qi, Xiu Li, and Bowen Zhou. Genprm: Scaling test-time compute of process reward models via generative reasoning. In Sven Koenig, Chad Jenkins, and Matthew E. Taylor, editors, *Fortieth AAAI Conference on Artificial Intelligence, Thirty-Eighth Conference on Innovative Applications of Artificial Intelligence, Sixteenth Symposium on Educational Advances in Artificial Intelligence, AAAI 2026, Singapore, January 20-27, 2026*, pages 34932–34940. AAAI Press, 2026. doi: 10.1609/AAAI.V40I41.40797. URL https://doi.org/10.1609/aaai.v40i41.40797.

[858] Qiwei Zhao, Xujiang Zhao, Yanchi Liu, Wei Cheng, Yiyou Sun, Mika Oishi, Takao Osaki, Katsushi Matsuda, Huaxiu Yao, and Haifeng Chen. SAUP: situation awareness uncertainty propagation on LLM agent. *CoRR*, abs/2412.01033, 2024. doi: 10.48550/ARXIV.2412.01033. URL https://doi.org/10.48550/arXiv.2412.01033.

[859] Wanjia Zhao, Mert Yüksekgönül, Shirley Wu, and James Y. Zou. Sirius: Self-improving multi-agent systems via bootstrapped reasoning. In Danielle Belgrave, Cheng Zhang, Laura N. Montoya, Hsuan-Tien Lin, Razvan Pascanu, Piotr Koniusz, Marzyeh Ghassemi, Nancy Chen, Iván Vladimir Meza Ruíz, and Arturo Loaiza-Bonilla, editors, *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2025, NeurIPS 2025, San Diego, CA, USA, December 2-7, 2025 / Mexico City, Mexico, November 30 - December 5, 2025*, 2025. URL http://papers.nips.cc/paper_files/paper/2025/hash/b45279ac82cb017a5f55ea7d3653193a-Abstract-Conference.html.

[860] Yubao Zhao, Weiquan Huang, Sudong Wang, Ruochen Zhao, Chen Chen, Yao Shu, and Chengwei Qin. Training multi-turn search agent via contrastive dynamic branch sampling. *CoRR*, abs/2602.03719, 2026. doi: 10.48550/ARXIV.2602.03719. URL https://doi.org/10.48550/arXiv.2602.03719.

[861] Yujie Zhao, Lanxiang Hu, Yang Wang, Minmin Hou, Hao Zhang, Ke Ding, and Jishen Zhao. Stronger together: On-policy reinforcement learning for collaborative llms. *CoRR*, abs/2510.11062, 2025. doi: 10.48550/ARXIV.2510.11062. URL https://doi.org/10.48550/arXiv.2510.11062.

[862] Zihao Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. Calibrate before use: Improving few-shot performance of language models. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 12697–12706. PMLR, 2021. URL http://proceedings.mlr.press/v139/zhao21c.html.

[863] Shuai Zhen, Yanhua Yu, Ruopei Guo, Nan Cheng, and Yang Deng. Hierarchical reinforcement learning with augmented step-level transitions for LLM agents. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens, editors, *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2026, San Diego, California, United States, July 2-7, 2026*, pages 7035–7053. Association for Computational Linguistics, 2026. URL https://aclanthology.org/2026.acl-long.318/.

[864] Boyuan Zheng, Boyu Gou, Jihyung Kil, Huan Sun, and Yu Su. Gpt-4v(ision) is a generalist web agent, if grounded. In Ruslan Salakhutdinov, Zico Kolter, Katherine A. Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp, editors, *Forty-first International Conference on Machine Learning, ICML 2024,*

Vienna, Austria, July 21-27, 2024, volume 235 of *Proceedings of Machine Learning Research*, pages 61349–61385. PMLR / OpenReview.net, 2024. URL https://proceedings.mlr.press/v235/zheng24e.html.

[865] Chujie Zheng, Shixuan Liu, Mingze Li, Xiong-Hui Chen, Bowen Yu, Chang Gao, Kai Dang, Yuqiong Liu, Rui Men, An Yang, Jingren Zhou, and Junyang Lin. Group sequence policy optimization. *CoRR*, abs/2507.18071, 2025. doi: 10.48550/ARXIV.2507.18071. URL https://doi.org/10.48550/arXiv.2507.18071.

[866] Kunhao Zheng, Jesse Michael Han, and Stanislas Polu. minif2f: a cross-system benchmark for formal olympiad-level mathematics. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022. URL https://openreview.net/forum?id=9ZPegFuFTFv.

[867] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging llm-as-a-judge with mt-bench and chatbot arena. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine, editors, *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/91f18a1287b398d378ef22505bf41832-Abstract-Datasets_and_Benchmarks.html.

[868] Longtao Zheng, Rundong Wang, Xinrun Wang, and Bo An. Synapse: Trajectory-as-exemplar prompting with memory for computer control. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=Pc8AU1aF5e.

[869] Tianshi Zheng, Zheye Deng, Hong Ting Tsang, Weiqi Wang, Jiaxin Bai, Zihao Wang, and Yangqiu Song. From automation to autonomy: A survey on large language models in scientific discovery. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng, editors, *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, EMNLP 2025, Suzhou, China, November 4-9, 2025*, pages 17733–17750. Association for Computational Linguistics, 2025. doi: 10.18653/V1/2025.EMNLP-MAIN.895. URL https://doi.org/10.18653/v1/2025.emnlp-main.895.

[870] Yuhao Zheng, Li'an Zhong, Yi Wang, Rui Dai, Kaikui Liu, Xiangxiang Chu, Linyuan Lv, Philip Torr, and Kevin Qinghong Lin. Code2world: A GUI world model via renderable code generation. *CoRR*, abs/2602.09856, 2026. doi: 10.48550/ARXIV.2602.09856. URL https://doi.org/10.48550/arXiv.2602.09856.

[871] 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, editors, *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, EMNLP 2025, Suzhou, China, November 4-9, 2025*, pages 414–431. Association for Computational Linguistics, 2025. doi: 10.18653/V1/2025.EMNLP-MAIN.22. URL https://doi.org/10.18653/v1/2025.emnlp-main.22.

[872] Zhicheng Zheng, Xin Yan, Zhenfang Chen, Jingzhou Wang, Qin Zhi Eddie Lim, Joshua B. Tenenbaum, and Chuang Gan. Contphy: Continuum physical concept learning and reasoning from videos. In Ruslan Salakhutdinov, Zico Kolter, Katherine A. Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp, editors, *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*, volume 235 of *Proceedings of Machine Learning Research*, pages 61526–61558. PMLR / OpenReview.net, 2024. URL https://proceedings.mlr.press/v235/zheng24l.html.

[873] Qiyong Zhong, Mao Zheng, Mingyang Song, Xin Lin, Jie Sun, Houcheng Jiang, Xiang Wang, and Junfeng Fang. SOD: step-wise on-policy distillation for small language model agents. *CoRR*, abs/2605.07725, 2026. doi: 10.48550/ARXIV.2605.07725. URL https://doi.org/10.48550/arXiv.2605.07725.

[874] Wanjun Zhong, Lianghong Guo, Qiqi Gao, He Ye, and Yanlin Wang. Memorybank: Enhancing large language models with long-term memory. In Michael J. Wooldridge, Jennifer G. Dy, and Sriraam Natarajan, editors, *Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2024, Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence, IAAI 2024, Fourteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2024, February 20-27, 2024, Vancouver, Canada*, pages 19724–19731. AAAI Press, 2024. doi: 10.1609/AAAI.V38I17.29946. URL https://doi.org/10.1609/aaai.v38i17.29946.

[875] Andy Zhou, Kai Yan, Michal Shlapentokh-Rothman, Haohan Wang, and Yu-Xiong Wang. Language agent tree search unifies reasoning, acting, and planning in language models. In Ruslan Salakhutdinov, Zico Kolter, Katherine A. Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp, editors, *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*, volume 235 of *Proceedings of Machine Learning Research*, pages 62138–62160. PMLR / OpenReview.net, 2024. URL https://proceedings.mlr.press/v235/zhou24r.html.

[876] Chenyu Zhou, Huacan Chai, Wenteng Chen, Zihan Guo, Rong Shan, Yuanyi Song, Tianyi Xu, Yingxuan Yang, Aofan Yu, Weiming Zhang, Congming Zheng, Jiachen Zhu, Zeyu Zheng, Zhuosheng Zhang, Xingyu Lou, Changwang Zhang, Zhihui Fu, Jun Wang, Weiwen Liu, Jianghao Lin, and Weinan Zhang. Externalization in LLM agents: A unified review of memory, skills, protocols and harness engineering. *CoRR*, abs/2604.08224, 2026. doi: 10.48550/ARXIV.2604.08224. URL https://doi.org/10.48550/arXiv.2604.08224.

[877] Denny Zhou, Nathanael Schärli, Le Hou, Jason Wei, Nathan Scales, Xuezhi Wang, Dale Schuurmans, Claire Cui, Olivier Bousquet, Quoc V. Le, and Ed H. Chi. Least-to-most prompting enables complex reasoning in large language models. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023. URL https://openreview.net/forum?id=WZH7099tgfM.

[878] Enshen Zhou, Jingkun An, Cheng Chi, Yi Han, Shanyu Rong, Chi Zhang, Pengwei Wang, Zhongyuan Wang, Tie-Jun Huang, Lu Sheng, and Shanghang Zhang. Roborefer: Towards spatial referring with reasoning in vision-language models for robotics. *CoRR*, abs/2506.04308, 2025. doi: 10.48550/ARXIV.2506.04308. URL https://doi.org/10.48550/arXiv.2506.04308.

[879] Gaoyue Zhou, Hengkai Pan, Yann LeCun, and Lerrel Pinto. DINO-WM: world models on pre-trained visual features enable zero-shot planning. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu, editors, *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*, volume 267 of *Proceedings of Machine Learning Research*. PMLR / OpenReview.net, 2025. URL https://proceedings.mlr.press/v267/zhou25t.html.

[880] Hanlin Zhou and Huah Yong Chan. ORCH: many analyses, one merge - a deterministic multi-agent orchestrator for discrete-choice reasoning with ema-guided routing. *Frontiers Artif. Intell.*, 9, 2026. doi: 10.3389/FRAI.2026.1748735. URL https://doi.org/10.3389/frai.2026.1748735.

[881] Huichi Zhou, Yihang Chen, Siyuan Guo, Xue Yan, Kin Hei Lee, Zihan Wang, Ka Yiu Lee, Guchun Zhang, Kun Shao, Linyi Yang, and Jun Wang. Memento: Fine-tuning LLM agents without fine-tuning llms. *CoRR*, abs/2508.16153, 2025. doi: 10.48550/ARXIV.2508.16153. URL https://doi.org/10.48550/arXiv.2508.16153.

[882] Junting Zhou, Jin Chen, Linfeng Hao, Denghui Cao, Zheyu Wang, Qiguang Chen, Chaoyou Fu, Jiaze Chen, Yuchen Wu, Ge Zhang, Mingxuan Wang, Wenhao Huang, and Tong Yang. BABE: biology arena benchmark. *CoRR*, abs/2602.05857, 2026. doi: 10.48550/ARXIV.2602.05857. URL https://doi.org/10.48550/arXiv.2602.05857.

[883] Peilin Zhou, Bruce Leon, Xiang Ying, Can Zhang, Yifan Shao, Qichen Ye, Dading Chong, Zhiling Jin, Chenxuan Xie, Meng Cao, Yuxin Gu, Sixin Hong, Jing Ren, Jian Chen, Chao Liu, and Yining Hua. Browsecomp-zh: Benchmarking web browsing ability of large language models in chinese. *CoRR*, abs/2504.19314, 2025. doi: 10.48550/ARXIV.2504.19314. URL https://doi.org/10.48550/arXiv.2504.19314.

[884] Pengfei Zhou, Shengcong Chen, Di Chen, Jiaxu Wang, Rongjun Jin, Bingwen Zhu, Yike Pan, Songen Gu, Kuanning Wang, Shufeng Nan, Xingyu Qiu, Chenhao Qiu, Pu Yang, Yunuo Cai, Jianxiong Gao, Yifan Li, Yanwei Fu, Xiangyu Yue, Zhi Chen, and Jianlan Luo. τ_0-wm: A unified video-action world model for robotic manipulation. *CoRR*, abs/2606.01027, 2026. doi: 10.48550/ARXIV.2606.01027. URL https://doi.org/10.48550/arXiv.2606.01027.

[885] Shuyan Zhou, Frank F. Xu, Hao Zhu, Xuhui Zhou, Robert Lo, Abishek Sridhar, Xianyi Cheng, Tianyue Ou, Yonatan Bisk, Daniel Fried, Uri Alon, and Graham Neubig. Webarena: A realistic web environment for building autonomous agents. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=oKn9c6ytLx.

[886] Tianyu Zhou, Pinqiao Wang, Yilin Wu, and Hongyang Yang. Finrobot: AI agent for equity research and valuation with large language models. *CoRR*, abs/2411.08804, 2024. doi: 10.48550/ARXIV.2411.08804. URL https://doi.org/10.48550/arXiv.2411.08804.

[887] Yang Zhou, Sunzhu Li, Shunyu Liu, Wenkai Fang, Jiale Zhao, Jingwen Yang, Jianwei Lv, Kongcheng Zhang, Yihe Zhou, Hengtong Lu, Wei Chen, Yan Xie, and Mingli Song. Breaking the exploration bottleneck: Rubric-scaffolded reinforcement learning for general LLM reasoning. *CoRR*, abs/2508.16949, 2025. doi: 10.48550/ARXIV.2508.16949. URL https://doi.org/10.48550/arXiv.2508.16949.

[888] Yingli Zhou, Wang Shu, Yaodong Su, Wenchuan Du, Yixiang Fang, and Xuemin Lin. A comprehensive survey on agent skills: Taxonomy, techniques, and applications. *CoRR*, abs/2605.07358, 2026. doi: 10.48550/ARXIV.2605.07358. URL https://doi.org/10.48550/arXiv.2605.07358.

[889] Yongchao Zhou, Andrei Ioan Muresanu, Ziwen Han, Keiran Paster, Silviu Pitis, Harris Chan, and Jimmy Ba. Large language models are human-level prompt engineers. In *The Eleventh International Conference*

on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023. OpenReview.net, 2023. URL https://openreview.net/forum?id=92gvk82DE-.

[890] 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. *CoRR*, abs/2506.15841, 2025. doi: 10.48550/ARXIV.2506.15841. URL https://doi.org/10.48550/arXiv.2506.15841.

[891] Chiwei Zhu, Benfeng Xu, Mingxuan Du, Shaohan Wang, Xiaorui Wang, Zhendong Mao, and Yongdong Zhang. Fs-researcher: Test-time scaling for long-horizon research tasks with file-system-based agents. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens, editors, *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2026, San Diego, California, United States, July 2-7, 2026*, pages 6353–6373. Association for Computational Linguistics, 2026. URL https://aclanthology.org/2026.acl-long.288/.

[892] Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Hao Tian, Yuchen Duan, Weijie Su, Jie Shao, Zhangwei Gao, Erfei Cui, Xuehui Wang, Yue Cao, Yangzhou Liu, Xingguang Wei, Hongjie Zhang, Haomin Wang, Weiye Xu, Hao Li, Jiahao Wang, Nianchen Deng, Songze Li, Yinan He, Tan Jiang, Jiapeng Luo, Yi Wang, Conghui He, Botian Shi, Xingcheng Zhang, Wenqi Shao, Junjun He, Yingtong Xiong, Wenwen Qu, Peng Sun, Penglong Jiao, Han Lv, Lijun Wu, Kaipeng Zhang, Huipeng Deng, Jiaye Ge, Kai Chen, Limin Wang, Min Dou, Lewei Lu, Xizhou Zhu, Tong Lu, Dahua Lin, Yu Qiao, Jifeng Dai, and Wenhai Wang. Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models. *CoRR*, abs/2504.10479, 2025. doi: 10.48550/ARXIV.2504.10479. URL https://doi.org/10.48550/arXiv.2504.10479.

[893] Ningyan Zhu, Huacan Wang, Jie Zhou, Feiyu Chen, Shuo Zhang, Ge Chen, Chen Liu, Jiarou Wu, Wangyi Chen, Xiaofeng Mou, and Yi Xu. Semaclaw: A step towards general-purpose personal AI agents through harness engineering. *CoRR*, abs/2604.11548, 2026. doi: 10.48550/ARXIV.2604.11548. URL https://doi.org/10.48550/arXiv.2604.11548.

[894] Yakun Zhu, Yutong Huang, Shengqian Qin, Zhongzhen Huang, Shaoting Zhang, and Xiaofan Zhang. Medmcp-calc: Benchmarking llms for realistic medical calculator scenarios via MCP integration. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens, editors, *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2026, San Diego, California, United States, July 2-7, 2026*, pages 4845–4873. Association for Computational Linguistics, 2026. URL https://aclanthology.org/2026.acl-long.221/.

[895] Yinghao Zhu, Ziyi He, Haoran Hu, Xiaochen Zheng, Xichen Zhang, Jiyao Wang, Junyi Gao, Liantao Ma, and Lequan Yu. Medagentboard: Benchmarking multi-agent collaboration with conventional methods for diverse medical tasks. In Danielle Belgrave, Cheng Zhang, Laura N. Montoya, Hsuan-Tien Lin, Razvan Pascanu, Piotr Koniusz, Marzyeh Ghassemi, Nancy Chen, Iván Vladimir Meza Ruíz, and Arturo Loaiza-Bonilla, editors, *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2025, NeurIPS 2025, San Diego, CA, USA, December 2-7, 2025 / Mexico City, Mexico, November 30 - December 5, 2025*, 2025. URL http://papers.nips.cc/paper_files/paper/2025/hash/d59aa09699530c00d4b875a883876641-Abstract-Datasets_and_Benchmarks_Track.html.

[896] Yutao Zhu, Xingshuo Zhang, Maosen Zhang, Jiajie Jin, Liancheng Zhang, Xiaoshuai Song, Kangzhi Zhao, Wencong Zeng, Ruiming Tang, Han Li, Ji-Rong Wen, and Zhicheng Dou. GISA: A benchmark for general information-seeking assistant. *CoRR*, abs/2602.08543, 2026. doi: 10.48550/ARXIV.2602.08543. URL https://doi.org/10.48550/arXiv.2602.08543.

[897] Mingchen Zhuge, Wenyi Wang, Louis Kirsch, Francesco Faccio, Dmitrii Khizbullin, and Jürgen Schmidhuber. Gptswarm: Language agents as optimizable graphs. In Ruslan Salakhutdinov, Zico Kolter, Katherine A. Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp, editors, *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*, volume 235 of *Proceedings of Machine Learning Research*, pages 62743–62767. PMLR / OpenReview.net, 2024. URL https://proceedings.mlr.press/v235/zhuge24a.html.

[898] Mingchen Zhuge, Changsheng Zhao, Dylan R. Ashley, Wenyi Wang, Dmitrii Khizbullin, Yunyang Xiong, Zechun Liu, Ernie Chang, Raghuraman Krishnamoorthi, Yuandong Tian, Yangyang Shi, Vikas Chandra, and Jürgen Schmidhuber. Agent-as-a-judge: Evaluate agents with agents. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu, editors, *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*, volume 267 of *Proceedings of Machine Learning Research*. PMLR / OpenReview.net, 2025. URL https://proceedings.mlr.press/v267/zhuge25a.html.

[899] Brianna Zitkovich, Tianhe Yu, Sichun Xu, Peng Xu, Ted Xiao, Fei Xia, Jialin Wu, Paul Wohlhart, Stefan Welker, Ayzaan Wahid, Quan Vuong, Vincent Vanhoucke, Huong T. Tran, Radu Soricut, Anikait Singh, Jaspiar Singh, Pierre Sermanet, Pannag R. Sanketi, Grecia Salazar, Michael S. Ryoo, Krista Reymann, Kanishka Rao, Karl Pertsch, Igor Mordatch, Henryk Michalewski, Yao Lu, Sergey Levine, Lisa Lee, Tsang-Wei Edward Lee, Isabel Leal, Yuheng Kuang, Dmitry Kalashnikov, Ryan Julian, Nikhil J. Joshi, Alex Irpan, Brian Ichter, Jasmine Hsu, Alexander Herzog, Karol Hausman, Keerthana Gopalakrishnan, Chuyuan Fu, Pete Florence, Chelsea Finn, Kumar Avinava Dubey, Danny Driess, Tianli Ding, Krzysztof Marcin Choromanski, Xi Chen, Yevgen Chebotar, Justice Carbajal, Noah Brown, Anthony Brohan, Montserrat Gonzalez Arenas, and Kehang Han. RT-2: vision-language-action models transfer web knowledge to robotic control. In Jie Tan, Marc Toussaint, and Kourosh Darvish, editors, *Conference on Robot Learning, CoRL 2023, 6-9 November 2023, Atlanta, GA, USA*, volume 229 of *Proceedings of Machine Learning Research*, pages 2165–2183. PMLR, 2023. URL https://proceedings.mlr.press/v229/zitkovich23a.html.

[900] Zefang Zong, Dingwei Chen, Yang Li, Qi Yi, Bo Zhou, Chengming Li, Bo Qian, Peng Chen, and Jie Jiang. At^{2}po: Agentic turn-based policy optimization via tree search, 2026. URL https://arxiv.org/abs/2601.04767.

[901] Tao Zou, Xinghua Zhang, Haiyang Yu, Minzheng Wang, Fei Huang, and Yongbin Li. EIFBENCH: extremely complex instruction following benchmark for large language models. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng, editors, *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, EMNLP 2025, Suzhou, China, November 4-9, 2025*, pages 20930–20953. Association for Computational Linguistics, 2025. doi: 10.18653/V1/2025.EMNLP-MAIN.1059. URL https://doi.org/10.18653/v1/2025.emnlp-main.1059.

[902] Wei Zou, Mingwen Dong, Miguel Romero Calvo, Shuaichen Chang, Jiang Guo, Dongkyu Lee, Xing Niu, Xiaofei Ma, Yanjun Qi, and Jiarong Jiang. Poison once, exploit forever: Environment-injected memory poisoning attacks on web agents. *CoRR*, abs/2604.02623, 2026. doi: 10.48550/ARXIV.2604.02623. URL https://doi.org/10.48550/arXiv.2604.02623.

[903] Yuanhao Zou, Shengji Jin, Andong Deng, Youpeng Zhao, Jun Wang, and Chen Chen. A.I.R.: enabling adaptive, iterative, and reasoning-based frame selection for video question answering. *CoRR*, abs/2510.04428, 2025. doi: 10.48550/ARXIV.2510.04428. URL https://doi.org/10.48550/arXiv.2510.04428.

[904] Yuxin Zuo, Shang Qu, Yifei Li, Zhang-Ren Chen, Xuekai Zhu, Ermo Hua, Kaiyan Zhang, Ning Ding, and Bowen Zhou. Medxpertqa: Benchmarking expert-level medical reasoning and understanding. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu, editors, *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*, volume 267 of *Proceedings of Machine Learning Research*. PMLR / OpenReview.net, 2025. URL https://proceedings.mlr.press/v267/zuo25a.html.

[905] Yuxin Zuo, Zikai Xiao, Li Sheng, Fei Huang, Jianhong Tu, Yuxuan Liu, Tianyi Tang, Xiaomeng Hu, Yang Su, Qingfeng Lan, Yantao Liu, Qin Zhu, Yinger Zhang, Bowen Yu, Haiquan Zhao, Haiyang Xu, Jianxin Yang, Jiayang Cheng, Junyang Wang, Lianghao Deng, Mingfeng Xue, Tianyi Bai, Yang Fan, Yubo Ma, Yucheng Li, Zeyu Cui, Zhihai Wang, Zhihui Xie, Zhuorui Ye, An Yang, Dayiheng Liu, Jingren Zhou, and Ning Ding. Qwen-agentworld: Language world models for general agents. *CoRR*, abs/2606.24597, 2026. doi: 10.48550/ARXIV.2606.24597. URL https://doi.org/10.48550/arXiv.2606.24597.

[906] Adam Zweiger, Jyothish Pari, Han Guo, Ekin Akyürek, Yoon Kim, and Pulkit Agrawal. Self-adapting language models. *CoRR*, abs/2506.10943, 2025. doi: 10.48550/ARXIV.2506.10943. URL https://doi.org/10.48550/arXiv.2506.10943.