Deep Research: A Systematic Survey

Zhengliang Shi[1] **Yiqun Chen**[2] **Haitao Li**[3] **Weiwei Sun**[4] **Shiyu Ni**[5] **Yougang Lyu**[6]
Run-Ze Fan[7] **Bowen Jin**[8] **Yixuan Weng**[9] **Minjun Zhu**[9] **Qiujie Xie**[9] **Xinyu Guo**[10] **Qu Yang**[11]
Jiayi Wu[11] **Jujia Zhao**[12] **Xiaqiang Tang**[11] **Xinbei Ma**[11] **Cunxiang Wang**[3] **Ruotian Ma**[11]
Jiaxin Mao[2] **Qingyao Ai**[3] **Jen-Tse Huang**[13] **Wenxuan Wang**[2] **Yue Zhang**[9] **Yiming Yang**[4]
Zhaopeng Tu[11,✉] **Zhaochun Ren**[12,✉]

[1]Shandong University [2]Renmin University of China [3]Tsinghua University
[4]Carnegie Mellon University [5]UCAS [6]University of Amsterdam
[7]University of Massachusetts Amherst [8]University of Illinois Urbana-Champaign
[9]Westlake University [10]University of Arizona [11]Tencent
[12]Leiden University [13]Johns Hopkins University

Abstract: Large language models (LLMs) have rapidly evolved from text generators into powerful problem solvers. Yet, many open tasks demand critical thinking, multi-source, and verifiable outputs, which are beyond single-shot prompting or standard retrieval-augmented generation. Recently, numerous studies have explored *Deep Research* (DR), which aims to combine the reasoning capabilities of LLMs with external tools, such as search engines, thereby empowering LLMs to act as research agents capable of completing complex, open-ended tasks. This survey presents a comprehensive and systematic overview of deep research systems, including a clear roadmap, foundational components, practical implementation techniques, important challenges, and future directions. Specifically, our main contributions are as follows: (i) we formalize a three-stage roadmap and distinguish deep research from related paradigms; (ii) we introduce four key components: query planning, information acquisition, memory management, and answer generation, each paired with fine-grained sub-taxonomies; (iii) we summarize optimization techniques, including prompting, supervised fine-tuning, and agentic reinforcement learning; and (iv) we consolidate evaluation criteria and open challenges, aiming to guide and facilitate future development. *As the field of deep research continues to evolve rapidly, we are committed to continuously updating this survey to reflect the latest progress in this area.*

✉ *Corresponding Author*

Keywords: Deep Research, Large Language Models, Information Retrieval

Date: November 13, 2025
Code Repository: https://github.com/mangopy/Deep-Research-Survey
Contact: zhengliang.shii@gmail.com chenyiqun990321@ruc.edu.cn z.ren@liacs.leidenuniv.nl

Contents

1. Introduction

Large language models (LLMs), trained on web-scale corpora, have rapidly evolved from fluent text generators into autonomous agents capable of long-horizon reasoning in practical complex applications [224, 83, 465, 288]. They have exhibited strong generalization across diverse domains, including mathematical reasoning [112, 466], creative writing [95], and practical software engineering [118, 140, 166]. Many real-world tasks are inherently open-ended, involving **critical thinking**, **factually grounded information**, and the production of **self-contained** responses. This is far beyond what single-shot prompting or static parametric knowledge can provide [122, 183, 289]. To address this gap, the **Deep Research (DR)** paradigm [237, 97, 66, 481, 125, 202] has emerged. DR frames LLMs within an end-to-end research workflow that iteratively decomposes complex problems, acquire evidence via tool use, and synthesizes validated insights into coherent long-form answers.

Despite rapid progress, there remains no comprehensive survey that systematically analyzes the key components, technical details, and open challenges of DR. Most existing work [458, 31] mainly summarizes developments in related areas such as Retrieval-Augmented Generation (RAG) and web-based agents [401, 200, 285, 456, 316]. However, in contrast to RAG [89, 72], DR adopts a more flexible, autonomous workflow that eschews handcrafted pipelines and aims to produce coherent, evidence-grounded reports. Therefore, a clear overview of its technical landscape is urgent but remains a challenge. This survey fills this gap by providing a comprehensive synthesis of DR: mapping its core components to representative system implementations, consolidating key techniques and evaluation methodologies, and establishing a foundation for consistent benchmarking and sustained progress in AI-driven research.

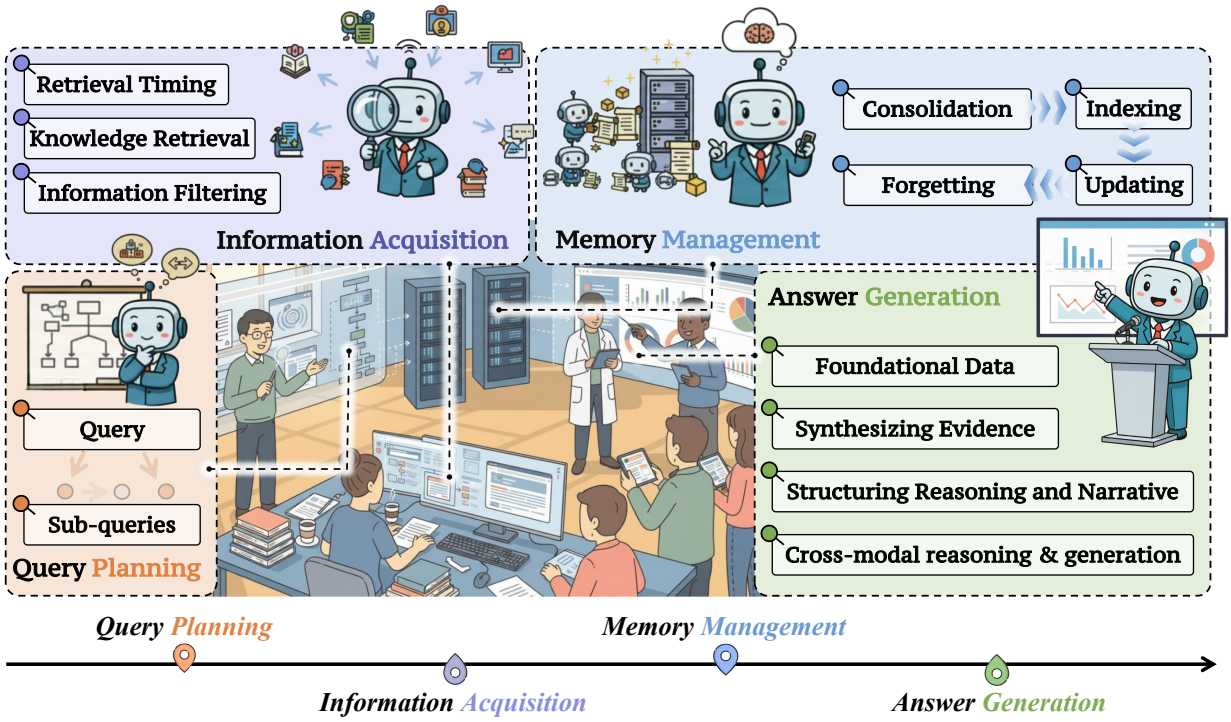

Figure 1: An overview of four key components in a general deep research system, including: Task Planning (Section 3.1). Information Acquisition (Section 3.2). Memory Management (Section 3.3) and Answer Generation (Section 3.4).

In this survey, we propose a three-stage roadmap for DR systems, illustrating their broad applications ranging from agentic information seeking to autonomous scientific discovery. Based on the roadmap, we summarize the key components of the task-solving workflow for the most commonly used DR systems. Specifically, we present four foundational components in DR: (i) *query planning*, which decomposes the initially input query into a series of simpler, sub-queries [250, 426]; (ii) *information acquisition*, which invokes external retrieval, web browsing, or various tools on demand [167, 221]; (iii) *memory management*, which ensures relevant task-solving context through controlled updating or folding [243]; (iv) *answer generation*, which produces comprehensive outputs with explicit source attribution, *e.g.,* a scientific report. This scope is distinct from standard RAG [89, 72] techniques, which typically treat retrieval as a heuristic augmentation step, without a flexible research workflow or a broader action space. We also introduce how to optimize DR systems in effectively coordinating these components, categorizing existing approaches into three types: (i) *workflow prompting*; (ii) *supervised fine-tuning* (SFT), and (iii) *end-to-end reinforcement learning* (RL).

The remainder of this paper is organized as follows: Section 2 clearly defines DR and its boundaries; Section 3 introduces four key components in DR; Section 4 introduces technique details about optimizing a DR system; Section 5 summarizes well-known evaluation datasets and resources, and Section 6 discusses challenges for future directions.

To sum up, our survey makes the following contributions: (i) We formalize a three-stage roadmap of DR and clearly distinguish it from related techniques such as standard retrieval-augmented generation; (ii) We introduce four key components of DR systems, together with fine-grained sub-taxonomies for each, to provide a comprehensive view of the research loop; (iii) We summarize detailed optimization approaches for building DR systems, offering practical insights into workflow prompting, supervised fine-tuning, and reinforcement learning; and (iv) We consolidate evaluation criteria and open challenges to enable comparable reporting and to guide future research.

2. Preliminary Concept of Deep Research

2.1. What is Deep Research

DR aims to endow LLMs with an **end-to-end research workflow**, enabling them to function as agents that generate coherent, source-grounded reports with minimal human supervision. Such systems automate the entire research loop, spanning planning, evidence acquisition, analysis, and reporting. In a DR setting, the LLM agent plans queries, acquires and filters evidence from heterogeneous sources (*e.g.,* the web, tools, and local files), maintains and revises a working memory, and synthesizes verifiable answers with explicit attribution. Below, we formally introduce a three-phase roadmap that structures the rapidly evolving, capability-oriented landscape of DR, and we compare it systematically with conventional RAG paradigms.

2.2. Understanding Deep Research from Three Phases

We view DR as a capability trajectory rather than a value hierarchy. The three phases below capture a progressive expansion of what systems can reliably do, from acquiring precise evidence, to synthesizing it into readable analyses, and finally to forming defensible insights.

Phase I: Agentic Search. Phase I systems specialize in finding the correct sources and extracting answers with minimal synthesis. They typically reformulate the user query (via rewriting or decomposition) to improve recall, retrieve and re-rank candidate documents, apply lightweight

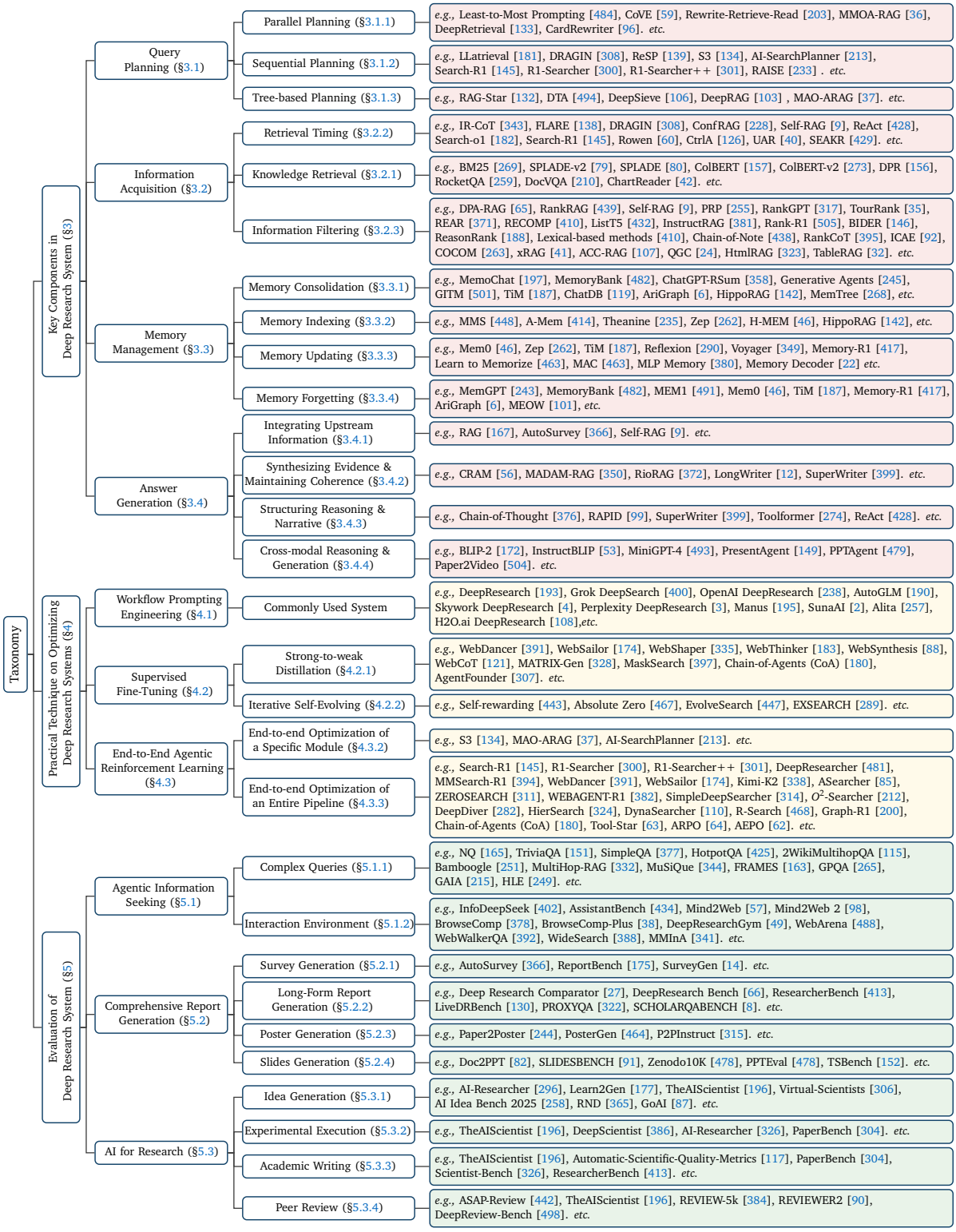

Figure 2: Taxonomy of the main content of this survey.

filtering or compression, and produce concise answers supported by explicit citations. The emphasis is on faithfulness to retrieved content and predictable runtime. Representative applications include

Capability (Key Feature)	Standard RAG	Agentic Search	Integrated Research	Full-stack AI Scientist
Search Engine Access	✓	✓	✓	✓
Use of Various Tools (e.g., Web APIs)	✗	✓	✓	✓
Code Execution for Experiment	✗	✗	✗	✓
Reflection for Action Correction	✗	✓	✓	✓
Task-solving Memory Management	✗	✓	✓	✓
Innovation and Hypothesis Proposal	✗	✗	✗	✓
Long-form Answer Generation & Validation	✓	✗	✓	✓
Action Space	Narrow	Broad	Broad	Broad
Reasoning Horizon	Single	Long-horizon	Long-horizon	Long-horizon
Workflow Organization	Fixed	Flexible	Flexible	Flexible
Output Form and Application	Short Span	Short Span	Report	Academic Paper

Table 1: Comparison between conventional RAG (leftmost column) and the three envisioned stages of Deep Research (right columns). The capabilities evolve from static retrieval and generation to adaptive, autonomous, and scientifically creative workflows.

open-domain question answering [227, 165], multi-hop question answering [425, 344, 265], and other information-seeking tasks [271, 444, 333, 70, 215] where truth is localized to a small set of sources. Evaluation prioritizes retrieval recall@k and answer exact matching, complemented by citation correctness and end-to-end latency, reflecting the phase's focus on accuracy-per-token and operational efficiency.

Phase II: Integrated Research. Phase II systems move beyond isolated facts to produce coherent, structured reports that integrate heterogeneous evidence while managing conflicts and uncertainty. The research loop becomes explicitly iterative: systems plan sub-questions, retrieve and extract key evidence from various raw content (*e.g.,* HTML [323], tables [44, 226], and charts [208, 208]), and ultimately synthesize comprehensive, narrative reports. The most commonly-used applications include market and competitive analysis [469, 347], policy briefs [356], itinerary design under constraints [331], and other long-horizon question answering [66, 434, 378, 49]. Accordingly, evaluation shifts from superficial short-form lexical matching to long-form quality, including: fine-grained factuality [43, 216], verified citations [310, 86], structural coherence [21], key points coverage [379]. Phase II thus trades a modest increase in compute and complexity for substantial gains in clarity, coverage, and decision support.

Phase III: Full-stack AI Scientist. Phase III aims at advancing scientific understanding and creation beyond mere information aggregation, representing a broader and more ambitious stage of DR In this phase, DR agents are expected not only to aggregate evidence but also to generate hypotheses [490], conduct experimental validation or ablation studies [223], critique existing claims [498], and propose novel perspectives [386]. Common applications include paper reviewing [506, 248, 498], scientific discovery [460, 292, 291], and experiment automation [362, 472]. Evaluation at this stage emphasizes the novelty and insightfulness of the findings, the argumentative coherence, the reproducibility of claims (including the ability to re-derive results from cited sources or code), and calibrated uncertainty disclosure.

2.3. Comparing Deep Research with RAG

Many real-world tasks are inherently open-ended, involving **critical thinking**, **factually grounded information**, and **self-contained** responses. These present several fundamental limitations of existing approaches. Below, we summarize three key challenges that cannot be solved by conventional RAG or scaling LLM parameters alone:

- *Flexible Interaction with the Digital World.* Conventional RAG systems operate in a static retrieval loop, relying solely on pre-indexed corpora [232, 225]. However, real-world tasks often require active interaction with dynamic environments such as search engines, web APIs, or even Code executors [487, 223, 362]. DR systems extend this paradigm by enabling LLMs to perform multi-step, tool-augmented interactions, allowing agents to access up-to-date information, execute operations, and verify hypotheses within a digital ecosystem.
- *Long-horizon Planning with Autonomous Workflows.* Complex research-like problems often require agents to coordinate multiple subtasks [378], manage task-solving context [411], and iteratively refine intermediate outcomes [290]. DR addresses this limitation through closed-loop control and multi-turn reasoning, allowing agents to autonomously plan, revise, and optimize their workflows toward long-horizon objectives.
- *Reliable Language Interfaces for Open-ended Tasks.* LLMs are prone to hallucination and inconsistency [109, 471, 123, 13, 52], particularly in open-ended settings. DR systems introduce verifiable mechanisms that align natural language outputs with grounded evidence, establishing a more reliable interface between human users and autonomous research agents.

3. Key Components in Deep Research System

A DR system can be viewed as a closed-loop workflow that takes a complex research question as input and produces a structured answer, typically in the form of long-form text with citations or synthesized reports. As illustrated in Figure 1, the DR system iteratively cycles through a set of interconnected components: (i) *query planning*, which decomposes the original question into sub-queries and tool calls that guide the workflow; (ii) *knowledge acquisition*, which retrieves and filters relevant information from external corpora, tools, or APIs; (iii) *memory management*, which stores, updates, and prunes intermediate findings to maintain context over long horizons; and (iv) *answer generation*, which synthesizes the accumulated evidence into a coherent, verifiable response with citations and checks for consistency. In this work, we provide detailed definitions and functionality for each component, along with representative works.

3.1. Query Planning

Query Planning refers to the process of transforming a complex and logically intricate question into a structured sequence of executable sub-queries (*aka.,* sub-tasks), each of which can be addressed incrementally. This decomposition allows stepwise reasoning and knowledge acquisition, thereby enhancing the reliability and accuracy of the final output generated by deep research system.

Figure 3 shows three widely-used strategies for query planning: (i) *parallel planning*, which decomposes the input into independent sub-queries that may be resolved in parallel [36, 59]; (ii) *sequential planning*, which arranges sub-queries into a linear order where each step depends on intermediate outcomes [286, 145]; and (iii) *tree-based planning*, which explores branching decision spaces and selects among candidate paths through pruning, backtracking, or heuristic-

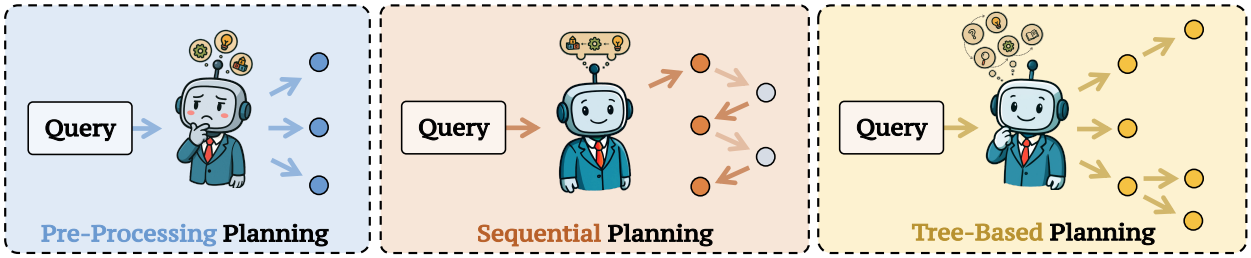

Figure 3: Three commonly-used types of query planning: (i) parallel planning; (ii) sequential planning; and (iii) tree-based planning.

guided search [427].

3.1.1. Parallel Planning

Definition. As illustrated in Figure 3(a), parallel planning operates by rewriting or decomposing the original query into multiple sub-questions in a single pass, typically without iterative interaction with downstream components. The primary advantage of this strategy lies in its efficiency: simultaneous generation enables parallel processing of sub-queries.

Representative Work. Early research typically instantiates parallel planning modules through heuristic approaches, most notably via prompt engineering [484, 59] or training on manually annotated datasets. For example, Least-to-Most Prompting [484] guides GPT-3 [19] to decompose a complex task into an ordered sequence of simpler, self-contained sub-queries in a few-shot setting. Similarly, CoVE [59] prompts LLMs to first generate multiple independent sub-questions and then ground each one with well-established evidence in parallel, a strategy widely adopted in knowledge-intensive applications.

Despite these advancements, query planning based on general heuristics or task-agnostic supervision often suffers from misalignment with end-to-end objectives in downstream applications, particularly in complex QA scenarios [425, 115, 344, 250]. To mitigate this issue, recent work has turned to end-to-end planning optimization via RL. For example, the Rewrite-Retrieve-Read framework [203] trains a query planner to maximize final answer accuracy using the Proximal Policy Optimization algorithm [276]. Crucially, the planner is reinforced only when documents retrieved by its sub-queries enable an LLM to generate a correct answer, which replaces reliance on heuristic decomposition rules. Building on this approach, subsequent efforts such as DeepRetrieval [133] and CardRewriter [96] have extended reward modeling for query planners to incorporate diverse downstream metrics (*e.g.,* evidence recall, retrieval NDCG@k). More recently, studies have also explored jointly optimizing query planning with other components in modular dense retrieval pipelines through multi-agent RL methods [36].

Advantages & Disadvantages. Despite their efficiency, parallel planning has two primary limitations. First, they typically operate in a *one-shot* fashion, interacting with other modules (*e.g.,* retriever, reasoner, aggregator) non-iteratively. As a result, they lack mechanisms to incorporate intermediate evidence, correct earlier decisions, or adaptively allocate computational resources. Second, they often *ignore data and logical dependencies* across sub-queries. Parallel execution assumes conditional independence, yet many real-world queries involve sequential reasoning in which later subtasks depend on the resolution of earlier ones. This can result in ill-posed or unanswerable sub-queries

due to missing contextual information.

3.1.2. Sequential Planning

Definition. As illustrated in Figure 3(b), the sequential planning decomposes the original query through multiple iterative steps, where each round of decomposition builds upon the outputs of previous rounds. At each stage, the sequential planning may invoke different modules or external tools to process intermediate results, enabling a dynamic, feedback-driven reasoning process. This multi-turn interaction allows the sequential planning to perform logically dependent query decompositions that are often intractable for pre-processing planning, which typically assumes conditional independence among sub-queries. By incorporating intermediate evidence and adapting the query trajectory accordingly, sequential planning is particularly well-suited for complex tasks that require stepwise inference, disambiguation, or progressive information gathering.

Representative Work. The sequential planning is often used to provide a series of sub-queries for the external knowledge needed in a step-by-step manner, which has been widely used in iterative QA systems [181, 308, 139]. For example, LLatrieval [181] introduces an iterative query planner that, whenever the current documents fail verification, leverages the LLM to pinpoint missing knowledge and generate a new query, either a question or a pseudo-passage, to retrieve supplementary evidence, repeating the cycle until the accumulated context fully supports a verifiable answer. DRAGIN [308] introduces a query planner that can utilize the self-attention scores to select the most context-relevant tokens from the entire generation history and reformulate them into a concise and focused query. This dynamic, attention-driven approach produces more accurate queries compared to the static *last sentence* or *last n tokens* strategies in previous methods, resulting in higher-quality retrieved knowledge and improved downstream generation. In ReSP [139], the query planner dynamically guides each retrieval iteration by formulating novel sub-questions explicitly targeted at identified information gaps whenever the currently accumulated evidence is deemed insufficient. By conditioning this reformulation process on both global and local memory states and by disallowing previously issued sub-questions, the approach mitigates the risks of over-planning and redundant retrieval. This design ensures that each newly generated query substantially contributes to advancing the multi-hop reasoning trajectory toward the final answer. RAISE [233] sequentially decomposes a scientific question into sub-problems, generates logic-aware queries for each, and retrieves step-specific knowledge to drive planning and reasoning. Additionally, S3 [134] and AI-SearchPlanner [213] both adopt sequential decision-making to control when and how to propose retrieval queries during multi-turn search. At each turn, the sequential planner evaluates the evolving evidence state and decides whether to retrieve additional context or to stop. Besides, more recent studies, including Search-R1 [145], R1-Searcher [300, 301] integrate a sequential planning strategy into an end-to-end, multi-turn search framework, thereby leveraging LLMs' internal reasoning for query planning.

Advantages & Disadvantages. Sequential planning enables dynamic, context-aware reasoning and fine-grained query reformulation, thereby facilitating more accurate acquisition of external knowledge. However, excessive reasoning turns or overly long reasoning chains can incur substantial computational costs and latency. In addition, an increased number of turns may introduce cumulative noise and error propagation, potentially causing instability during reinforcement learning training.

3.1.3. Tree-based Planning

Definition. As illustrated in Figure 3(c), the tree-based planning integrates features of both parallel and sequential planning by recursively treating each sub-query as a node within a structured search space, typically represented as a tree or a directed acyclic graph (DAG) [51]. This structure enables the use of advanced search algorithms, such as Monte Carlo Tree Search (MCTS) [20], to explore and refine potential reasoning paths. Compared to linear or flat decompositions, this approach supports more flexible and fine-grained decomposition of the original query, facilitating comprehensive knowledge acquisition.

Representative Work. A representative example is RAG-Star [132], which leverages MCTS in conjunction with the Upper Confidence Bound for Trees (UCT) [161] to guide a query planner in the iterative decomposition of complex questions. At each iteration, the planning model selects the most promising node using the UCT criterion, expands it by generating a sub-query and corresponding answer using a language model, evaluates the quality of the expansion via a retrieval-based reward model, and back-propagates the resulting score. This iterative process grows a reasoning tree of sub-queries until a satisfactory final answer is obtained. Other examples include DTA [494] and DeepSieve [106], which use a tree-based planner to restructure sequential reasoning traces into a DAG. This design enables the planning to aggregate intermediate answers along multiple branches and improves the model's ability to capture both hierarchical and non-linear dependencies across sub-tasks. DeepRAG [103] introduces tree-based planning via binary-tree exploration to iteratively decompose queries and decide parametric vs. retrieved reasoning, yielding large accuracy gains with fewer retrievals. More recently, MAO-ARAG [37] trains a planning agent that can dynamically orchestrate multiple, diverse query reformulation modules through a DAG structure. This adaptive workflow enables comprehensive query decomposition to enhance performance.

Advantages & Disadvantages. Tree-based planning integrates the strengths of parallel and sequential planning. It facilitates the decomposition of interdependent sub-queries and supports local parallel execution, striking an effective balance between efficiency and effectiveness. Nevertheless, training a robust Tree-based Planning module is challenging, requiring precise dependency modeling, careful trade-offs between speed and quality, addressing data scarcity, and tackling credit assignment issues in reinforcement learning.

Takeaway

This section on query planning provides a detailed overview of strategies for enhancing DR systems by decomposing complex queries into simpler, manageable subtasks. Each type of planning strategy offers unique benefits and faces specific challenges.

- *Pre-processing planning* is efficient in executing sub-queries simultaneously, though they may overlook dependencies between them.
- *Sequential planning* excels in managing dependencies through iterative processes but can incur higher computational costs.
- *Tree-based planning* strikes a balance by combining the strengths of both sequential and pre-processing approaches, allowing for adaptive and flexible query decomposition.

3.2. Information Acquisition

DR systems often acquire external information to augment LLMs' internal knowledge. However, due to the cost of retrieval and the uncertainty of document quality, it is necessary to determine when retrieval is needed [449, 396, 455]. Moreover, how to perform retrieval and manage retrieved information is key to the DR system's interaction with external knowledge. In the following, we discuss retrieval tools, retrieval timing, and information filtering in turn.

3.2.1. Retrieval Tools

Definition. In the context of DR, *retrieval tools* [299, 503, 419] are used to identify relevant information from large-scale corpora in response to a query, typically containing indexing and search techniques. Within typical DR workflows, retrieval serves as a core mechanism for bridging knowledge gaps by surfacing candidate evidence that can then be checked for accuracy, filtered for relevance, or combined into a coherent answer. Below, we systematically review widely adopted retrieval techniques, organized by modality: (i) *text-only retrieval*, and (ii) *multimodal retrieval*.

Text Retrieval. Conceptually, modern text retrieval can be organized into three families: (i) lexical retrieval, (ii) semantic retrieval, and (iii) commercial web search. Lexical and semantic retrieval are typically implemented on local resources, while commercial web search is typically accessed only via paid APIs. Specifically, *lexical retrieval* refers to methods that match documents based on exact term overlaps and statistical term weighting, including traditional approaches like TF-IDF and BM25 [269], as well as neural sparse models that learn to expand queries and documents with relevant terms while maintaining interpretable inverted-index structures [80, 79, 157, 156, 259, 354, 80, 273].

Different from the lexical retrieval, *semantic retrieval* refers to dense neural methods that encode queries and documents into continuous vector spaces to capture semantic similarity beyond exact term matching [283, 111, 284, 144], which has been widely adopted in recent works [145, 289].

More recently, *commercial web search* (like Google or Bing) has also been widely used in DR systems and web agents [334, 75, 240, 34]. It diverges from lexical and semantic retrieval models by providing access to real-time information, leveraging massive-scale web crawling and indexing, incorporating sophisticated ranking algorithms that consider authority and freshness signals, and offering built-in fact verification through cross-source validation. Previous work, such as WebGPT [221] and SearchGPT [412], demonstrates that commercial search APIs enable research agents to access current events and dynamic content that would be missing from static corpora.

Recent studies [182, 183, 329, 253] exemplify a shift towards more autonomous and capable research agents. These models feature deep web exploration capabilities, allowing them to interactively navigate beyond static search results to gather information. Overall, the evolution from lexical and semantic retrieval to commercial web search marks a shift from static, closed-corpus search toward dynamic, real-world information access, enabling DR systems to retrieve not only relevant but also timely and verifiable knowledge.

Multimodal Retrieval. Multimodal retrieval aims to mine multimodal information, including text, layout, and visuals (figures, tables, charts), and to preserve grounded pointers (spans, cells, coordinates) for verifiable citation, while maximizing recall under tight latency to support iterative DR. Multimodal information retrieval can be organized into three classes based on the primary type of information modality being indexed and retrieved: (i)*text-aware retrieval with layout*, which

indexes titles, captions, callouts, and surrounding prose and leverages document understanding models (LayoutLM [415], Donut [159], DocVQA [210]) plus layout/metadata filters; (ii) *visual retrieval via text–image similarity*, which encodes figures and chart thumbnails with CLIP [261], SigLIP [446], or BLIP [171] and performs ANN search for text-to-image matching or composed image retrieval [420]; and (iii) *structure-aware retrieval over parsed tables and charts*, which indexes axes, legends, data marks, and table schemas to support grounded lookup of numeric facts and relations (*e.g.,* ChartReader [42] or Chartformer [477]). These three approaches are typically combined: queries are searched across all indices simultaneously, with results fused using reciprocal-rank fusion [50] or cross-modal reranking to preserve grounded pointers for citations. Recent chart-focused VLMs [214, 25, 452, 209] further enhance the quality of visual-textual features.

Comparing Text Retrieval and Multimodal Retrieval. Compared to text-only retrieval, multimodal retrieval provides several key advantages. First, it captures visually encoded information and numeric trends that text-based methods often overlook, and facilitates cross-modal verification through hybrid fusion [50]. Second, it enables grounded citations using techniques such as layout parsing (*e.g.,* LayoutLM [415], Donut [159]) and chart understanding (*e.g.,* ChartReader [42] or Chartformer [477]). However, multimodal retrieval also presents several challenges, including increased computational costs for visual processing [261, 446], sensitivity to OCR errors and variations in chart formats [327, 404], and the complexity of aligning information across different modalities.

Takeaway

Knowledge retrieval for DR has evolved from traditional lexical and dense-text search to the use of real-time commercial web search engines for up-to-date information. However, text-only methods fail to capture information embedded in visual elements like charts, tables, and layouts. Multimodal retrieval addresses this gap by modeling visual and structural data. Its primary contribution is enabling grounded, verifiable citations by linking retrieved evidence back to specific data points (*e.g.,* table cells or chart coordinates), though this introduces higher computational costs and challenges in cross-modal alignment and format processing.

3.2.2. Retrieval Timing

Definition. Retrieval timing refers to determining when a model should trigger retrieval tools during information seeking, which is also known as adaptive retrieval [131, 429, 60]. Because the quality of retrieved documents is not guaranteed, blindly performing retrieval at every step is often suboptimal [206, 470, 289]. Retrieval introduces additional computational overhead, and low-quality or irrelevant documents may even mislead the model or degrade its reasoning performance [286]. Consequently, adaptive retrieval aims to invoke retrieval only when the model lacks sufficient knowledge, which requires the model to recognize its own knowledge boundaries [176, 453, 405, 266], *i.e.,* knowing what it knows and what it does not.

Prior work on adaptive retrieval follows two main directions: (i) estimating and enhancing a model's ability to *recognize its own knowledge boundaries* for a given query, and (ii) optimizing the *retrieval-trigger mode*l in multi-step settings to maximize downstream task performance.

Confidence Estimation as a Proxy for Boundary Perception. There are extensive works that investigate LLMs' perception of their knowledge boundaries. The degree to which a model perceives its boundaries is typically measured by the alignment between its confidence and factual correctness. Since factual correctness is typically evaluated by comparing the model's generated answer with the

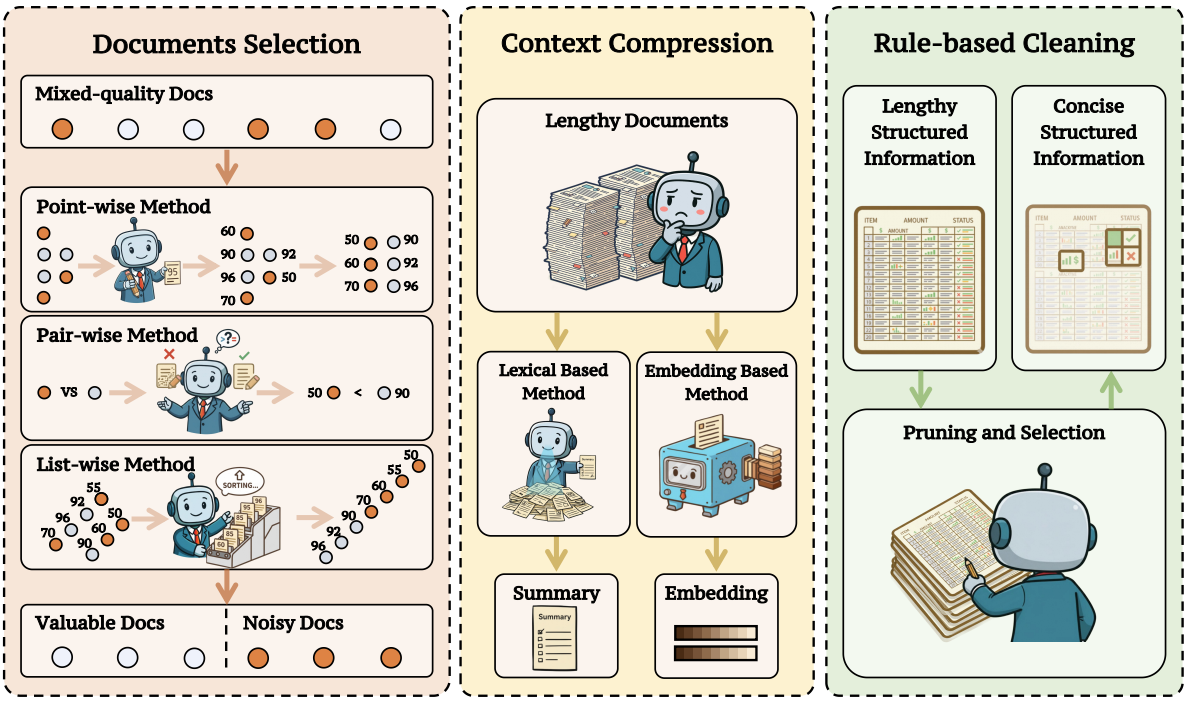

Figure 4: Existing information filtering approaches can be broadly categorized into the following types: (i) *Document Selection*; (ii) *Context Compression*; and (iii) *Rule-based Cleaning*.

ground-truth answer, existing studies focus on how to measure the model's confidence, which can be broadly divided into four categories.

- *Probabilistic Confidence*. This line of work treats a model's token-level generation probabilities as its confidence in the answer [104, 58, 137, 153, 295, 164, 69]. Prior to the emergence of LLMs, a line of work had already shown that neural networks tend to be poorly calibrated, often producing overconfident predictions even when incorrect [104, 58, 137]. More recently, some research[153, 295] reported that LLMs can be well calibrated on structured tasks such as multi-choice question answering or appropriate prompts, but for open-ended generation tasks, predicted probabilities still diverge from actual correctness. To address this gap, Duan et al. [69] proposed SAR, which computes confidence by focusing on important tokens, while Kuhn et al. [164] introduced semantic uncertainty, which estimates confidence from the consistency of outputs across multiple generations.
- *Consistency-based Confidence*. Since probabilistic confidence often fails to capture a model's semantic certainty and is inapplicable to black-box models without accessible generation probabilities, recent works represent confidence via semantic consistency across multiple responses [78, 207, 164, 451, 60]. The key idea is that a confident model should generate highly consistent answers across runs. Fomicheva et al. [78] first measured consistency through lexical similarity, while later studies used NLI (*i.e.,* natural language inference) models or LLMs to assess semantic consistency [207, 164]. To address the issue of consistent but incorrect answers, Zhang et al. [451] measure consistency across different models, as incorrect answers tend to vary between models, whereas correct ones align. Ding et al. [60] further extended this idea to multilingual settings.

- *Confidence Estimation Based on Internal States*. LLMs' internal states have been shown to capture the factuality of their generated content [10, 309, 28, 364, 230, 229]. Azaria and Mitchell [10] first discovered that internal states can signal models' judgment of textual factuality. Subsequent studies [309, 28] found that internal states after response generation reflect the factuality of self-produced answers. More recently, Wang et al. [364] and Ni et al. [230] demonstrated that factuality-related signals already exist in the pre-generation states, enabling the prediction of whether the output will be correct.
- *Verbalized Confidence*. Several studies explore enabling LLMs to express confidence in natural language, akin to humans, viewing such verbalization as a sign of intelligence [185, 431, 340, 409, 450, 424, 228]. Yin et al. [431] and Ni et al. [228] examined whether LLMs can identify unanswerable questions, finding partial ability but persistent overconfidence. Other works [340, 409] investigated fine-grained confidence expression. Xiong et al. [409] offered the first comprehensive study for black-box models, while Tian et al. [340] proposed generating multiple answers per pass for more accurate estimation. Beyond prompting, some methods explicitly train models to verbalize confidence [185, 424, 450], with Lin et al. [185] introducing this idea and using correctness-based supervision.

Representative Adaptive Retrieval Approaches. Deep research systems typically involve iterative interactions between model inference and external document retrieval, differing mainly in how they determine when to retrieve. Early works such as IR-CoT [343] enforce retrieval after every reasoning step, ensuring continual grounding in external knowledge but at the cost of efficiency. Building on insights from studies of models' perceptions of their own knowledge boundaries, recent approaches treat retrieval as a model-issued action, enabling the model to perform it dynamically only when needed. Similar to techniques in confidence estimation, these methods assess whether the model can answer a question correctly given the current context and perform retrieval when knowledge is deemed insufficient. They can be broadly categorized into four paradigms.

- *Probabilistic Strategy*. It triggers retrieval based on token-generation probabilities: when the model produces a token with low confidence, retrieval is initiated [138, 308].
- *Consistency-based Strategy*. Recognizing that both token-level probabilities and single-model self-consistency may fail to capture true semantic uncertainty, Rowen [60] evaluates consistency across responses generated by multiple models and languages, triggering retrieval when cross-model or cross-lingual agreement is low.
- *Internal States Probing*. CtrlA [126], UAR [40], and SEAKR [429] further propose that compared to generated responses, a model's internal states provide a more faithful reflection of its confidence, using them to guide adaptive retrieval decisions.
- *Verbalized Strategy*. It enables the model to directly express its confidence via natural language. These methods typically generate special tokens directly in the response to indicate the need for retrieval. ReAct [428] directly prompts the model to generate corresponding action text when retrieval is needed. Self-RAG [9] trains the model to explicitly express uncertainty through the special token (*i.e.,* `<retrieve>`), signaling the need for retrieval. With LLMs' growing reasoning capacity, recent research has shifted toward determining retrieval timing through reasoning and reflection. Search-o1 [182] introduces a Reason-in-Documents module, which prompts the model to selectively invoke search during reasoning. Search-R1 [145] further frames retrieval as part of the environment and employs reinforcement learning to jointly optimize both when and what to retrieve.

Collectively, these methods trace an evolution from fixed or per-step retrieval (*e.g.,* IR-CoT [343]) to

dynamically triggered retrieval (*e.g.,* ReAct [428], Self-RAG [9], Search-o1 [182]), and finally to RL–based systems that explicitly train retrieval policies (*e.g.,* Search-R1 [145]).

3.2.3. Information Filtering

Definition. Information filtering refers to the process of selecting, refining, or transforming retrieved documents so that only the most relevant and reliable evidence is passed to subsequent steps. Since retrieval tools are not perfect, the retrieved information often contains considerable noise [433, 393, 73]. This includes the content that is entirely irrelevant to the query or plausible-looking statements that nevertheless provide incorrect or misleading context. As shown in prior work [433, 143], LLMs are highly sensitive to such noise; without additional filtering or optimization, they can be easily misled into generating incorrect or hallucinated responses. Figure 4 summarizes three information filtering approaches: (i) *Document Selection*, (ii) *Context Compression*, and (iii) *Rule-based Cleaning*.

Document Selection. Document selection aims to rank a set of candidate documents based on their relevance and usefulness to the query, selecting the top-k helpful documents for question answering [410, 439, 381]. This selection operation reduces the impact of noisy documents on LLMs, improving the question-answering accuracy in downstream tasks. Below, we review three document selection strategies: *point-wise* selection, *pair-wise* selection, and *list-wise* selection.

- *Point-wise Selection*. Given an initially retrieved document list, **point-wise** methods independently score each candidate document. The most common approach involves fine-tuning an embedding model (*e.g.,* BGE [406]) that encodes the query and each document separately, after which their relevance is estimated via inner-product similarity [410, 128]. Another widely adopted strategy employs a cross encoder, which takes the concatenation of the query and a document as input and directly predicts a binary relevance score [65, 371]. More recently, several studies have leveraged LLMs' natural language understanding capabilities for relevance assessment. These methods train LLMs to output special tokens, such as `<ISREL>` [9] or the identifier `True` [439], to indicate whether an input document is relevant to the query.
- *Pair-wise Selection*. Unlike the point-wise approach, which assigns an absolute relevance score, the pair-wise method compares the relevance of two input candidate information snippets (typically two documents) and predicts which one is more relevant to the query. Pair-wise selection is less common than point-wise selection. A representative work is PRP [255], which adopts a pairwise-ranking-prompting approach. In PRP, the LLM receives a query and two candidate documents to decide which is more relevant, and the final ranking list is then obtained using a heapsort algorithm. To mitigate positional bias, PRP performs the comparison twice, swapping the document order each time, and aggregates the results to yield a more stable judgment.
- *List-wise methods*.Given a document list, a list-wise selection strategy directly selects the final set of relevant documents from the candidate list. A representative work is RankGPT [317], which feeds the entire candidate sequence into an LLM and leverages prompt engineering to produce a global ranking. In addition to RankGPT, other work, such as TourRank [35], uses a tournament-inspired strategy to generate a robust ranking list [35, 432]. ListT5 [432] proposes a list re-ranking method based on the Fusion-in-Decoder (FiD) [127] architecture, which independently encodes multiple documents in parallel and orders them by relevance, mitigating positional sensitivity while preserving efficiency. For large document sets, it builds m-ary tournament trees to group, rank, and merge results in parallel. Recently, more and more work has employed the reasoning model for list-wise document selection, advancing document selection by explicitly modeling a chain of thought. For example, InstructRAG [381] trains an LLM to generate detailed rationales

via instruction tuning [281], directly judging the usefulness of each document in the raw retrieved document list. Rank-R1 [505] employs the reinforcement learning algorithm GRPO [279] to train the LLM, enabling it to learn how to select the documents most relevant to a query from a list of candidates. ReasonRank [188] empowers a list-wise selection model through a proposed multi-view ranking-based GRPO [279], training an LLM on automatically synthesized multi-domain training data.

Content Compression. Content Compression aims to remove redundant or irrelevant information from retrieved knowledge, thereby increasing the density of useful content within the model's context. Existing approaches primarily fall into two categories: *lexical-based* and *embedding-based* methods.

- **Lexical-based methods** condense retrieved text into concise natural language, aiming to only include the key point related to the given query [410, 355]. Representative works such as RECOMP [410] fine-tune a smaller, open-source LLM to summarize the input retrieved documents, where the ground truth is synthesized by prompting powerful commercial LLMs like GPT-4 [1]. Chain-of-Note [438] introduces a reading-notes mechanism that compels the model to assess the relevance of retrieved documents to the query and extract the most critical information before generating an answer, with training data annotated by GPT-4 and further validated through human evaluation. Other work, like BIDER [146], eliminates reliance on external model distillation by synthesizing Key Supporting Evidence (KSE) for each document, using it for compressor SFT, and further optimizing with PPO based on gains in answer correctness. Zhu et al. [496] argue that previous compressors optimized with log-likelihood objectives failed to precisely define the scope of useful information, resulting in residual noise. They proposed a noise-filtering approach grounded in the information bottleneck principle, aiming to maximize the mutual information between the compressed content and the target output while minimizing it between the compressed content and retrieved passages. RankCoT [395] implicitly learns document reranking during information refinement. It first employs self-reflection to generate summary candidates for each document. In subsequent DPO [276] training, the compression model is encouraged to assign higher probabilities to correct summaries when all documents are fed in, thereby inducing implicit reranking in the final summarization.
- **Embedding-based methods** compress context into dense embedding sequences [219, 107, 45]. Because embedding sequences can store information flexibly, embedding-based methods can be more efficient and effective than lexical-based methods. ICAE [92] uses an encoder to compress context into fixed-length embedding sequences and designs training tasks to align the embedding space with the answer generation model. COCOM [263] jointly fine-tunes the encoder and answer generation model, enhancing the latter's ability to capture the semantics of embeddings. xRAG [41] focuses on achieving extreme compression rates. It introduces a lightweight bridging module, initialized with a two-layer MLP and trained through paraphrase pretraining and context-aware instruction tuning. This module projects the document embedding vectors originally used for initial retrieval into a single token in the answer generation model's representation space, achieving contextual compression with only a single additional token. ACC-RAG [107] adapts compression rates for different documents by employing a hierarchical compressor to produce multi-granularity embedding sequences and dynamically selecting compression rates based on query complexity. Similarly, QGC [24] adjusts compression rates based on query characteristics, dynamically selecting different rates for different documents based on their relevance to the query.

Rule-based Cleaning. Rule-based methods are effective for cleaning externally sourced information with specific structures. For example, HtmlRAG [323] applies rule-based compression to remove

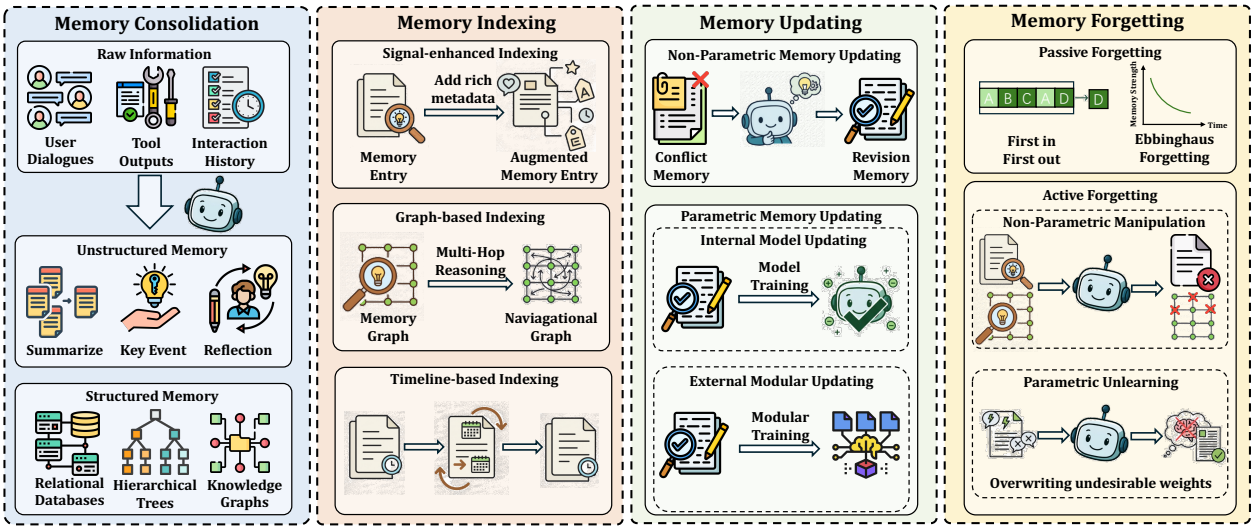

Figure 5: Memory management contains four key stages: (1) Memory Consolidation, (2) Memory Indexing, (3) Memory Updating, and (4) Memory Forgetting.

structurally present but semantically empty elements, such as CSS styling and JavaScript code, from retrieved web pages. This is combined with a two-stage block-tree pruning strategy that first uses embeddings for coarse pruning, followed by a generative model for fine-grained pruning. Separately, TableRAG [32] accurately extracts core table information through schema retrieval, which identifies key column names and data types, and cell retrieval, which locates high-frequency cell value pairs. This method addresses the challenges of context length limitations and information loss in large table understanding.

Advantages & Disadvantages. Filtering the retrieved knowledge is a simple yet effective strategy to enhance the performance of DR systems, as widely demonstrated in previous work [410, 323, 65]. However, incorporating an additional filtering module typically incurs additional computational costs and increased latency [393]. Moreover, overly filtering may remove useful or even correct information, thereby degrading model performance. Therefore, balancing filtering precision and information retention is crucial for building efficient and reliable DR systems.

Takeaway

Knowledge filtering can further process the metadata retrieved by DR systems, providing them with more concise and useful external knowledge while reducing noise interference and attention dilution caused by long context lengths. Filtering methods can be categorized into post-ranking selection, context compression, and rule-based cleaning. However, these knowledge filtering techniques often introduce additional time and computational costs. Therefore, different DR systems should choose the most suitable filtering method based on task characteristics to balance performance and resource consumption.

3.3. Memory Management

Definition. Memory management is a foundational component of advanced DR architectures, which governs the dynamic lifecycle of context used by DR agents in complex, long-horizon tasks [398, 67,

136], aiming to maintain coherent and relevant task-solving context [113, 462, 319].

Core Operation. As illustrated in Figure 5, memory management typically involves four core operations: consolidation, indexing, updating, and forgetting. Consolidation converts short-term experiences into durable representations that form the basis for later indexing. Indexing organizes these representations into retrieval structures that support efficient recall during problem solving. Updating refines or corrects stored knowledge, whereas forgetting selectively removes outdated or irrelevant content to reduce interference. In the following sections, we discuss consolidation, indexing, updating, and forgetting in detail.

3.3.1. Memory Consolidation

Definition. Memory consolidation is the process of transforming transient, short-term information, such as user dialogues or tool execution outputs, into stable, long-term representations [303, 67, 398]. Drawing an analogy to cognitive neuroscience, this process encodes and abstracts raw inputs to create durable memory engrams, laying the groundwork for efficient long-term storage and retrieval [398].

Memory consolidation involves transforming interaction histories into durable formats, including but not limited to model parameters [370], structured graphs [474], or knowledge bases [197, 67]. Distinct from memory indexing, which creates navigable access pathways over existing memories, consolidation is fundamentally concerned with the initial transformation and structural organization of raw experience. Two primary paradigms for this process have emerged: (i) *unstructured memory consolidation* and (ii) *structured memory consolidation*.

Unstructured Memory Consolidation. This paradigm distills lengthy interaction histories or raw texts into high-level, concise summaries or key event logs. For example, MemoryBank [482] processes and distills conversations into a high-level summary of daily events, which helps in constructing a long-term user profile. Similarly, MemoChat [197] summarizes conversation segments by abstracting the main topics discussed, while ChatGPT-RSum [358] adopts a recursive summarization strategy to manage extended conversations. Other approaches focus on abstracting experiences; Generative Agents [245] utilize a reflection mechanism triggered by sufficient event accumulation to generate more abstract thoughts as new, consolidated memories. To create generalizable plans, GITM [501] summarizes key actions from multiple successful plans into a common reference memory.

Structured Memory Consolidation. This paradigm transforms unstructured information into highly organized formats such as databases, graphs, or trees. This structural encoding is the primary act of consolidation, designed to capture complex inter-entity relationships and create an organized memory corpus. For instance, TiM [187] extracts entity relationships from raw information and stores them as tuples in a structured database. ChatDB [119] leverages a database as a form of symbolic memory, transforming raw inputs into a queryable, relational format. AriGraph [6] implements a memory graph where knowledge is represented as vertices and their interconnections as edges. Similarly, HippoRAG [142] constructs knowledge graphs over entities, phrases, and summaries to form an interconnected web of fragmented knowledge units. MemTree [268] builds and updates a tree structure by traversing from the root and deciding whether to deepen the tree with new information or create new leaf nodes based on semantic similarity. This hierarchical organization is the core of its consolidation strategy, enabling structured storage of memories.

3.3.2. Memory Indexing

Definition. Memory indexing involves constructing a navigational map over a DR agent's consolidated memories, analogous to a library's catalog or a book's index for efficient information retrieval [204]. Unlike memory consolidation, which focuses on the initial transformation of raw data into a durable format, indexing operates on already consolidated memories to create efficient, semantically rich retrieval pathways. This process builds auxiliary access structures that enhance retrieval not only in efficiency but also in relevance.

Effective indexing goes beyond simple keyword matching by encoding temporal [211] and relational [142] dependencies among memories. This is typically achieved by generating auxiliary codes, such as vector embeddings, summaries, or entity tags, which serve as retrieval entry points into the memory store. Given the vast, high-dimensional spaces these codes inhabit, specialized search techniques are required, such as Locality-Sensitive Hashing (LSH) [54], Hierarchical Navigable Small World (HNSW) graphs [205], or libraries like FAISS [150] for high-speed similarity search. These access mechanisms are commonly organized through three established paradigms:

- **Signal-enhanced Indexing.** This paradigm augments consolidated memory entries with auxiliary metadata, including emotional context, topics, and keywords, which function as granular pivots for context-aware retrieval [312, 448]. For instance, LongMemEval [390] enhances memory keys by integrating temporal and semantic signals to improve retrieval precision. Similarly, the Multiple Memory System (MMS) [448] decomposes experiences into discrete components, such as cognitive perspectives and semantic facts, thereby facilitating multifaceted retrieval strategies.
- **Graph-based Indexing.** This paradigm leverages a graph structure, where memories are nodes and their relationships are edges, as a sophisticated index. By representing memory networks in this way, agents can perform complex multi-hop reasoning by traversing chains of connections to locate information that is not explicitly linked to the initial query [46, 194]. For instance, HippoRAG [142] uses lightweight knowledge graphs to explicitly model inter-memory relations, enabling structured, interpretable access. A-Mem [414] adopts a dynamic strategy where the agent autonomously links related memory notes, progressively growing a flexible access network.
- **Timeline-based Indexing.** This paradigm creates a temporal index by organizing memory entries along chronological or causal sequences. Such structuring provides a historical access pathway, which is essential for understanding progression, maintaining conversational coherence, and supporting lifelong learning [353]. For example, the Theanine system [235] arranges memories along evolving timelines to facilitate retrieval based on both relevance and temporal dynamics. Zep [262] introduces a bi-temporal model for its knowledge graph, indexing each fact with t_{valid} and $t_{invalid}$ timestamps, which allows the agent to navigate the memory based on temporal validity.

3.3.3. Memory Updating

Definition. Memory updating is a core capability of DR agents, involving the reactivation and modification of existing knowledge in response to new information or environmental feedback [361, 321, 369]. This process is essential for maintaining the consistency, accuracy, and relevance of the agent's internal world model, thereby enabling continual learning and adaptive behavior in dynamic environments [349, 357].

Memory updating governs how an agent corrects factual inaccuracies, incorporates new information, and gradually improves its knowledge base [290, 55, 218]. Although related to memory forgetting, which focuses on removing outdated or incorrect content, memory updating centers on

modifying and refining existing knowledge to increase its fidelity. In the following, we introduce two updating strategies, depending on whether the memory is external (non-parametric) or internal (parametric) to the model [357].

Non-Parametric Memory Updating. Non-parametric memory, stored in external formats such as vector databases or structured files, is updated via explicit, discrete operations on the data itself. This approach offers flexibility and transparency. Key operations include:

- *Integration and Conflict Updating.* This operation focuses on incorporating new information and refining existing entries to maintain logical consistency. For example, the Mem0 framework employs an LLM to manage its knowledge base through explicit operations, such as adding new facts (`ADD`) or modifying existing entries with new details (`UPDATE`) to resolve inconsistencies [46]. To handle temporal conflicts, Zep updates its knowledge graph by modifying an existing fact's effective time range, setting an invalidation timestamp ($t_{invalid}$) to reflect that a newer fact has superseded it [262]. Similarly, the TiM framework curates its memory by using `MERGE` operations to combine related facts into a more coherent representation [187]
- *Self-Reflection Updating.* Inspired by human memory reconsolidation, this paradigm enables agents to iteratively refine their knowledge by reflecting on past experiences [290, 486]. Early systems like Reflexion [290] and Voyager [349] implement this through verbal self-correction and updates to a skill library. More dynamically, A-Mem [414] triggers a Memory Evolution process that re-evaluates and autonomously refines previously linked memories based on new contextual information.

Parametric Memory Updating. Parametric memory, encoded directly in a model's weights, is updated by modifying internal representations. This is typically more complex and computationally intensive. Three main approaches have emerged:

- *Global Updating.* This approach integrates new knowledge by continuing model training on additional datasets [278]. While effective for large-scale adaptation, it is computationally expensive and prone to catastrophic forgetting [361]. To address this, instead of simply injecting factual knowledge, Memory-R1 trains a dedicated Memory Manager agent to learn an optimal policy for modification operations such as `ADD` and `UPDATE`, moving beyond heuristic rules [417]. Additionally, a recent framework refines this process by employing methods such as Direct Preference Optimization to fine-tune the model's memory utilization strategy [463].
- *Localized Updating.* This technique modifies specific facts in the model's parameters without requiring full retraining [55, 218]. It is especially suited for online settings where rapid adaptation is needed, such as updating a user's preference [321]. Methods typically follow a *locate-and-edit* strategy or use meta-learning to predict weight adjustments while preserving unrelated knowledge [218, 321].
- *Modular Updating.* This emerging paradigm avoids the risks of continual weight modification by distilling knowledge into a dedicated, plug-and-play parametric module. Frameworks such as MLP Memory [380] and Memory Decoder [22] train a lightweight external module to imitate the output distribution of a non-parametric kNN retriever. This process effectively compiles a large corpus of external knowledge into the compact weights of the module. The resulting module can then be attached to any compatible LLM to provide specialized knowledge without modifying the base model's parameters, thereby avoiding catastrophic forgetting and reducing the latency of real-time retrieval [380, 22].

3.3.4. Memory Forgetting

Definition. Forgetting constitutes a fundamental mechanism in advanced agent architectures, enabling the selective removal or suppression of outdated, irrelevant, or potentially erroneous memory content. Rather than a system defect, forgetting is a functional process critical for filtering noise, reclaiming finite storage resources, and mitigating interference between conflicting information. In contrast to memory updating, which modifies existing knowledge to improve its accuracy, forgetting is a subtractive process that streamlines the memory store by eliminating specific content. This process can be broadly categorized into passive and active mechanisms.

Passive Forgetting. This simulates the natural decay of human memory, in which infrequently accessed or temporally irrelevant memories gradually lose prominence. This mechanism is particularly critical for managing the agent's immediate working memory or context window. Implementations are typically governed by automated, time-based rules rather than explicit content analysis. For instance, MemGPT [243] employs a First-In-First-Out (FIFO) queue for recent interactions, automatically moving the oldest messages from the main context into long-term storage. MemoryBank [482] draws inspiration from the Ebbinghaus forgetting curve, in which memory traces decay over time unless reinforced, allowing the agent to naturally prioritize recent content. A more aggressive approach, MEM1 [491], employs a *use-and-discard* policy: after each interaction, the agent synthesizes essential information into a compact state and immediately discards all prior contextual data to maintain constant memory consumption.

Active Forgetting. Active forgetting involves the intentional and targeted removal or invalidation of specific memory content. This process is a deliberate action, often triggered by the detection of contradictions or the need to correct inaccurate information, and its implementation varies depending on the memory type.

- *Non-Parametric Memory.* Active forgetting in external memory stores involves direct data manipulation. For example, Mem0 [46] implements an explicit `DELETE` command to remove outdated or contradictory facts. Similarly, TiM [187] introduces a dedicated `FORGET` operation to actively purge irrelevant or incorrect thoughts from its memory cache. Reinforcement learning can also be used to train a specialized Memory Manager agent to autonomously decide when to execute a `DELETE` command, as seen in the Memory-R1 framework [417]. AriGraph [6] maintains a structured memory graph by removing outdated vertices and edges. Some systems employ non-destructive forgetting; the Zep architecture [262], for example, uses *edge invalidation* to assign an invalid timestamp to an outdated entry, effectively retiring it without permanent deletion.
- *Parametric Memory.* In this context, active forgetting is typically achieved through machine unlearning techniques that modify a model's internal parameters to erase specific knowledge without full retraining. Approaches include locating and deactivating specific neurons or adjusting training objectives to promote the removal of targeted information. For example, MEOW [101] facilitates efficient forgetting by fine-tuning an LLM on generated contradictory facts, effectively overwriting undesirable memories stored in its weights.

Takeaway

Memory management is a cornerstone of the DR paradigm, enabling agents to transcend single-turn interactions and conduct complex, long-horizon investigations by governing the information lifecycle. Through the interdependent operations of consolidation, indexing, updating, and forgetting, the DR system maintains the context and coherence essential for an iterative research loop. Consequently, a sophisticated memory framework is what fundamentally distinguishes a DR agent from a simple RAG system, equipping it with the consistency, adaptability, and self-evolution necessary to autonomously synthesize comprehensive, trustworthy, and verifiable reports from a vast and dynamic information landscape.

3.4. Answer Generation

Definition. Answer generation typically represents the culminating stage of a DR system. It synthesizes information from upstream components, such as query planning (Section 3.1), information acquisition (Section 3.3.1), and memory systems (Section 3.3.1), and generates a coherent, comprehensive, and well-supported response that accurately reflects the user's original intent.

Unlike traditional text generation, the answer generation within an advanced DR workflow addresses complex challenges such as reconciling conflicting evidence, maintaining long-range coherence, and structuring outputs with transparent reasoning and proper citations. It has evolved from template-based generation [160] to sophisticated synthesis shown in Figure 6, which reflects the growing demand for trustworthy, explainable, and multimodal research outputs [167, 16]. To deconstruct this process, we will explore it across four progressive stages: beginning with the integration of diverse information sources, moving to the synthesis of evidence and maintenance of coherence, then structuring the reasoning and narrative, and finally, advancing to the frontier of cross-modal generation.

3.4.1. Integrating Upstream Information

Definition. The main principle of trustworthy answer generation is to ensure that every statement is grounded in verifiable external evidence. Thus, the first stage of answer generation is integrating information from its upstream components, including: the sub-queries from the query planning, the ranked and potentially conflicting evidence, and the evolving contextual state stored in memory.

Recent developments in this area demonstrate sophisticated strategies for integrating upstream information, moving from simple evidence-feeding to dynamic, state-aware synthesis. The most common approach involves integrating ranked evidence from the retrieval module. Frameworks like Self-RAG [9], for example, employ a more dynamic integration by adaptively retrieving passages on demand. It then generates reflection tokens to assess the relevance of the retrieved information and its own generation, effectively integrating an internal self-correction mechanism to steer the synthesis. Moving beyond static evidence, more advanced systems integrate the query plan with an evolving memory state to ensure long-range coherence, a paradigm known as Stateful Query Planning. For instance, graph-centric frameworks like Plan-on-Graph (PoG) [30] explicitly integrate the plan with a dynamic memory (storing sub-goal status, explored paths, and retrieved entities). This memory is then actively used during a *reflection* step to guide and self-correct subsequent planning, tightly coupling the reasoning state with the generation process. Similarly, search-based frameworks like MCTS-OPS [435] formalize this by treating the MCTS tree itself as the state of the evolving query

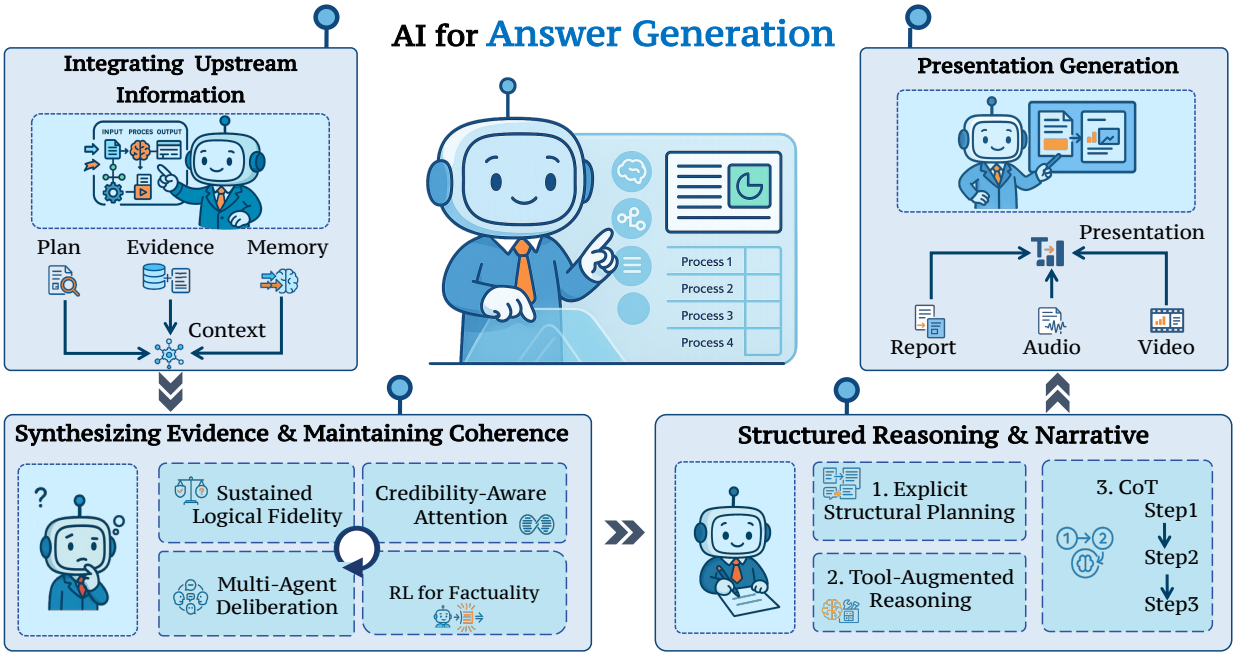

Figure 6: Illustrating the schematic of the answer generation process in DR. The workflow begins by integrating upstream information, moves to synthesizing evidence and ensuring coherence, then constructs a structured narrative via reasoning, and culminates in a multimodal presentation output.

plan. Here, the system integrates its experiential memory (node values from past rollouts) to guide the `SELECTION` and `EXPANSION` of the next planning step, ensuring the final answer synthesizes the full context of the problem-solving process.

While these architectures provide a robust foundation, the core challenges of synthesizing contradictory evidence and maintaining long-form coherence remain the next frontier.

3.4.2. Synthesizing Evidence and Maintaining Coherence

Producing answers to research-level questions requires resolving informational conflicts and sustaining coherent, information-dense narration across extended outputs.

Resolving Conflicting Evidence. Research queries frequently surface contradictory sources, requiring the model to discriminate among varying levels of reliability. Building on fact-verification paradigms [339], recent systems adopt three major strategies.

- *Credibility-Aware Attention*: Instead of treating all retrieved information equally, this approach intelligently weighs evidence based on its source. The system assigns a higher score to information coming from more credible sources (*e.g.,* a top-tier scientific journal) compared to less reliable ones (*e.g.,* an unverified blog) [56]. This allows the model to prioritize trustworthy information while still considering relevant insights from a wider range of sources [94].
- *Multi-Agent Deliberation*: This strategy simulates an expert committee meeting to debate the evidence. Frameworks like MADAM-RAG [350] employ multiple independent AI agents, each tasked with analyzing the retrieved documents from a different perspective. Each agent forms its own assessment and conclusion. Afterwards, a final *meta-reasoning* step synthesizes these

diverse viewpoints to forge a more robust and nuanced final answer, much like a panel of experts reaching a consensus [351].

- *Reinforcement Learning for Factuality*: This method trains the generator through a trial-and-error process that rewards factual accuracy [313]. A representative approach is RioRAG [372], in which an LLM receives a positive reward when it generates statements that are strongly and consistently supported by the provided evidence. Conversely, it is penalized for making unsubstantiated claims or statements that contradict the source material, shaping the model to inherently prefer generating factually grounded and reliable answers.

Long-form Coherence and Information Density. Another key challenge is ensuring **Sustained Informational Accuracy**. Research answers are often lengthy, and maintaining a logical thread while avoiding repetition or verbosity is non-trivial. Let L_{model} denote the maximum coherent length of a model's output, and L_{SFT} represent the average length of examples in its supervised fine-tuning dataset. SFT offers an intuitive approach to enhancing the long-form generation capabilities of large language models. However, LongWriter [12] empirically demonstrates that the maximum coherent length of a model's output often scales with the average length of its fine-tuning samples, which can be formally expressed as $L_{\text{model}} \propto L_{\text{SFT}}$ [12]. To address this, LongWriter focuses on systematic training for extended generation, while others use reflection-driven processes to iteratively improve consistency [399]. Additionally, RioRAG [372] introduces a length-adaptive reward function to promote information density, which penalizes verbosity that fails to add informational value, preventing reward hacking through verbosity. Together, these techniques shift the focus of generation from mere content aggregation toward credible, concise, and coherent synthesis, laying the groundwork for structured reasoning.

3.4.3. Structuring Reasoning and Narrative

The research community's focus is shifting from the mere factual accuracy of DR systems to the crucial need for explainability and logical rigor in their answers. An opaque answer, which prevents users from tracing the underlying reasoning process, has significantly diminished utility in critical domains like scientific research [116, 201, 283]. Consequently, a significant line of work has emerged to enable models to generate structured reasoning processes rather than just monolithic final answers [376, 484, 99]. This trend is reflected in the design of most modern DeepResearch systems, which increasingly favor the explicit presentation of this structural information [418, 486].

Prompt-based Chain-of-Thought. This foundational approach focuses on eliciting intermediate reasoning steps before producing a final answer. The most prominent technique is Chain-of-Thought (CoT) prompting [376], which can be formally expressed as $\mathcal{R} = \text{LLM}(\text{CoT-Prompt} + \mathcal{Q} + \text{Evidence})$. This method enhances both interpretability and multi-step reasoning performance. Its applicability has been broadened by extensions such as zero-shot CoT [162] and Least-to-Most prompting [484].

Explicit Structural Planning. More advanced systems move beyond simple linear chains to formalize the structure of the entire answer. For instance, RAPID [99] formalizes this process into three stages: (i) *outline generation*; (ii) *outline refinement through evidence discovery*; and (iii) *plan-guided writing*, where the outline forms a directed acyclic graph to support complex, non-linear argumentation. Similarly, SuperWriter [399] extends this idea by decoupling the reasoning and text-production phases and optimizing the entire process via hierarchical Direct Preference Optimization.

Tool-Augmented Reasoning. This line of work enhances reasoning by dynamically interfacing with

external resources. Representative work allows models to invoke external computational or retrieval tools dynamically, ensuring both analytic rigor and factual grounding [274, 254, 120, 385, 287].

3.4.4. Presentation Generation

The frontier of answer generation extends beyond text, encompassing the integration of multimodal and structured information. Research questions increasingly demand answers that combine textual reasoning with visual, tabular, or auditory data, maintaining semantic and presentational coherence. Early breakthroughs such as BLIP-2 [172] and InstructBLIP [53] enable multimodal instruction-following by aligning vision-language embeddings. MiniGPT-4 [493] advances this by leveraging cross-modal attention to seamlessly integrate visual and textual evidence.

Recently, a series of works have demonstrated higher presentation capabilities, signaling an evolution from content generation to presentation generation [430, 407, 504]. Existing work like MedConQA [403], LIDA [346], ChartGPT [441], and Urania [430] can synthesize data analyses into dynamic, interactive visualizations. Others work, including PresentAgent [149], Qwen2.5-Omni [147], and AnyToAny [507], generates synchronized audio narrations alongside text. More recently, PPTAgent [479] and Paper2Video [504] even extend to editable presentation generation, where full analytical reports are automatically transformed into slide decks with coordinated text, figures, and layout elements. At the leading edge, video-grounded agents [178, 383] retrieve or generate relevant visual footage, delivering answers through multimodal storytelling. As summarized in Table 2, while most DR systems still focus on textual synthesis with citations, only a handful, such as OpenAI DeepResearch [238] and H2O.ai DeepResearch [108], currently support comprehensive multimodal output. The emerging consensus suggests that rich, multi-format answer generation will soon become a standard expectation [408], bridging the gap between knowledge synthesis and human-centered presentation.

Takeaway

Answer generation represents the synthesis core of DR systems, integrating upstream information, reconciling conflicting evidence, and structuring coherent, evidence-grounded narratives. Recent advances, from credibility-aware attention and multi-agent deliberation to reinforcement learning for factuality, have enhanced both factual reliability and interpretability. Systems now move beyond content aggregation toward concise, logically structured synthesis supported by transparent reasoning frameworks such as Chain-of-Thought and plan-guided writing. Moreover, the frontier of answer generation extends into multimodal generation, where text, visuals, tables, and audio coalesce into rich, human-centered outputs. These developments mark a paradigm shift from generating text to generating explainable, trustworthy, and presentation-ready knowledge.

4. Practical Techniques for Optimizing Deep Research Systems

So far, we have introduced the core components that constitute a typical DR system. Building on these foundation, we now delve into practical techniques for improving such DR systems in real-world settings. These techniques focus on how to flexibly coordinate and enhance the key components, with the goal of achieving more reliable and effective task completion. Below, we discuss three commonly used paradigms: workflow prompting, supervised fine-tuning, and agentic reinforcement learning. Workflow prompting typically relies on a carefully designed pipeline (*aka.,* prompting engineering)

Table 2: Comparing output capabilities of contemporary representative DR systems, where the ■ indicates **supported capability**

System	Content Generation					Structured Output			Advanced		
	Text	Image	Audio	Video	Pres.	Table	JSON	Code	Chart	GUI	Cite
Gemini DeepResearch [193]	■	■				■			■		■
Grok DeepSearch [400]	■										■
OpenAI DeepResearch [238]	■	■				■		■	■	■	■
AutoGLM [190]	■										■
H2O.ai DeepResearch [108]	■	■	■		■	■	■	■	■		■
Skywork DeepResearch [4]	■			■	■						■
Perplexity DeepResearch [3]	■										■
Manus [195]	■				■	■			■		■
OpenManus [239]	■								■		■
OWL (CAMEL-AI) [120]	■				■	■			■		■
SunaAI [2]	■				■	■			■		
Alita [257]	■								■		

that guides the agents. The latter two paradigms aim to train a specific DR agent capable of reasoning, retrieving information, and generating high-quality answers.

4.1. Workflow Prompt Engineering

Definition. A simple yet effective way to build a DR system is to construct a complex workflow that enables collaboration among multiple agents. In the most common setting, an orchestration agent coordinates a team of specialized *worker agents*, allowing them to operate in parallel on different aspects of a complex research task. To illustrate the key principles and design considerations behind such a DR workflow, we introduce Anthropic Deep Research [337] as a representative example.

4.1.1. Deep Research System of Anthropic

Anthropic proposes a multi-agent Deep Research (DR) framework where a **lead orchestrator** coordinates multiple **worker agents** through structured, auditable interactions. The system transforms an open-ended research query into a complete workflow, from planning and delegation to synthesis and citation, under an explicit research budget controlling agent count, tool usage, and reasoning depth. We highlight several **core points** that enable the system's efficiency and reliability:

- *Query Stratification and Planning.* The orchestrator first analyzes the semantic type and difficulty of the input query (*e.g.,* depth-first vs. breadth-first) to determine research strategy and allocate a corresponding budget of agents, tool calls, and synthesis passes.
- *Delegation and Scaling.* Effort scales with complexity: from 1–2 agents for factual lookups to up to 10 or more for multi-perspective analyses, each assigned with clear quotas and stopping criteria to enable dynamic budget reallocation.
- *Task Decomposition and Prompt Specification.* The main query is decomposed into modular subtasks, each encoded as a structured prompt specifying objectives, output schema, citation policy, and fallback actions to ensure autonomy with accountability.
- *Tool Selection and Evidence Logging.* A central tool registry (*e.g.,* web fetch, PDF parsing, calculators) is used following freshness, verifiability, and latency rules. Agents record all tool provenance in an evidence ledger for traceable attribution.
- *Parallel Gathering and Interim Synthesis.* Worker agents operate concurrently while the orchestrator monitors coverage, resolves conflicts, and launches micro-delegations to close residual gaps or

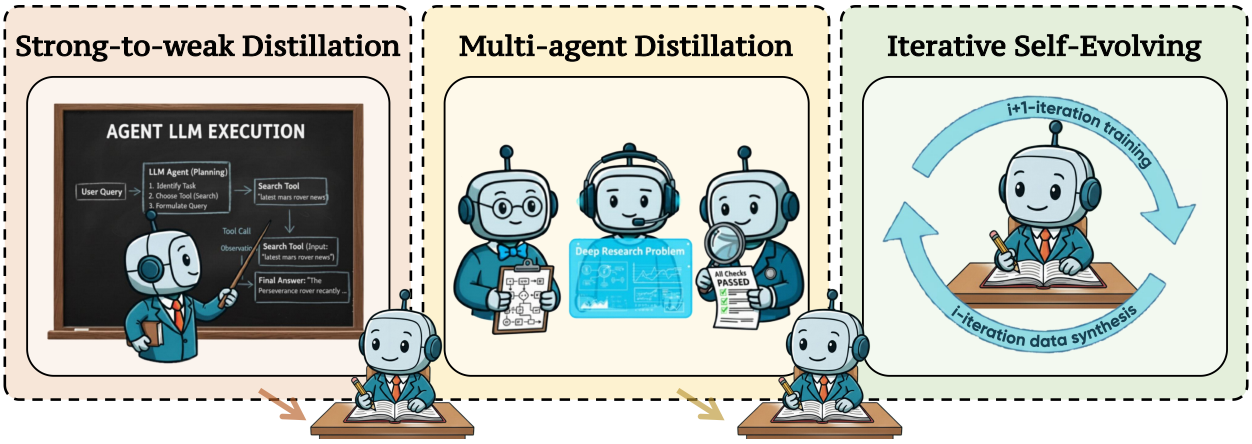

Figure 7: Comparisons among three types of data synthesis approaches, including: (i) Strong-to-Weak Distillation, (ii) Multi-Agent Distillation, and (iii) Iterative Self-Evolving. Each type is illustrated through the process of how agents perform tasks, learn, and refine their abilities.

trigger deeper reasoning where needed.

- *Final Report and Attribution.* The orchestrator integrates verified findings into a coherent report, programmatically linking claims to sources and ensuring schema compliance, factual grounding, and transparent citation.

Overall, Anthropic's system exemplifies a scalable, interpretable multi-agent research paradigm that achieves high-quality synthesis through modular delegation, explicit budgeting, and verifiable reasoning.

4.2. Supervised Fine-Tuning

Definition. Supervised fine-tuning (SFT) is a widely adopted approach that trains models to imitate desired behaviors using input–output pairs under a supervised learning objective. Within DR, SFT is commonly employed as the *cold start*, *e.g.,* a warm-up process, before online reinforcement learning [145, 182, 397, 300, 173]. It aims to endow agents with basic task-solving skills [252, 382].

Since manual collection of expert trajectories is labor-intensive, costly, and difficult to scale, a key challenge lies in automatically constructing high-quality SFT datasets. This has been widely explored by prior work [367, 483, 336, 48]. Below, we categorize representative work into two main paradigms: (i) *strong-to-weak distillation*, distilling correct task-solving trajectories from powerful LLMs (*e.g.,* GPT-5 and DeepSeek-V3.1) into smaller, weak models; and (ii) *iterative self-evolution*, iteratively fine-tuning the model on the dataset produced by itself, leading to a progressive improvement.

4.2.1. Strong-to-weak Distillation

Definition. Strong-to-weak distillation transfers high-quality decision trajectories from a powerful *teacher* system to smaller, weaker *student* models. Early work predominantly uses a single LLM-based agent to synthesize trajectories; more recent research employs multi-agent teacher systems to elicit more diverse, higher-complexity trajectories. We detail these two lines of work below.

Single-agent distillation. Representative systems instantiate this pipeline in various ways. WebDancer [391] provides the agent with search and click tools. A strong non-reasoning model generates short CoT, while a large reasoning model (LRM) generates long CoT. The agent learns from both, using rejection sampling for quality control. WebSailor [174] uses an expert LRM to generate action-observation trajectories, then reconstructs short CoT with a non-reasoning model, ensuring the final reasoning chain is compact enough for long-horizon tasks. WebShaper [335] uses search and visit tools in a ReAct-style trajectory. It performs 5 rollouts per task and filters out repeated or speculative answers using a reviewing LLM. WebThinker [183] augments SFT with policy-gradient refinement and WebSynthesis [88] leverages a learned world model to simulate virtual web environments and employs MCTS to synthesize diverse, controllable web interaction trajectories entirely offline.

Multi-agent distillation. Multi-agent distillation synthesizes training data using an agentic teacher system composed of specialized, collaborating agents (*e.g.,* a planner, a tool caller, and a verifier), with the goal of transferring emergent problem-solving behaviors into a single end-to-end student model [93, 328]. This paradigm tends to produce diverse trajectories, richer tool-use patterns, and explicit self-correction signals.

A representative work is MaskSearch [397], which constructs a multi-agent pipeline that includes a planner, a query rewriter, and an observer, generating 58k verified chain-of-thought trajectories. Similarly, Chain-of-Agents [180] builds on the expert multi-agent system OAgents [495] to synthesize task-solving trajectories, and after a four-stage filtering pipeline that removes trivial or incorrect cases, it yields 16,433 high-quality trajectories for agent training. More recently, AgentFounder [307] propose the agentic continual pre-training, which scales up the data generation process by constructing large-scale planning traces, tool-invocation sequences, and step-by-step reasoning data.

Comparing Two Types of Distillation. Single-agent distillation provides a simple and easy-to-deploy pipeline, but it is limited by the bias of a single teacher model and the relatively shallow nature of its synthesized trajectories [234, 199, 502, 220]. Such trajectories often emphasize token-level action sequences rather than higher-level reasoning, which can restrict the student model's generalization ability in complex tasks. In contrast, multi-agent distillation generates longer and more diverse trajectories that expand the action space to include strategic planning, task decomposition, iterative error correction, and self-reflection [489, 29]. This broader behavioral coverage equips student models with stronger capabilities for multi-step and knowledge-intensive reasoning [180].

Despite these advantages, multi-agent distillation introduces notable trade-offs. The pipelines require careful system design, substantial inference cost, and dedicated infrastructure for logging and verification. Data quality can also be brittle as the system's sensitivity to prompting [256, 264, 280].

4.2.2. Iterative Self-Evolving

Definition. Iterative self-evolving data generation is an autonomous, cyclic process in which a model continuously generates new training data to fine-tune itself, progressively enhancing its capabilities [367, 349, 447, 467].

Representative Work. Early evidence that large language models can improve themselves comes from self-training methods [367, 443, 39], where a model bootstraps from a small set of seed tasks to synthesize instruction–input–output triples, filters the synthetic data, and then fine-tunes itself on the resulting corpus. These approaches deliver substantial gains in instruction following with minimal human supervision. Yuan et al. [443] further introduces self-rewarding language models,

in which the model generates its own rewards through LLM-as-a-Judge prompting. More recently, Zhao et al. [467] extends this idea to the zero-data regime by framing self-play as an autonomous curriculum. The model creates code-style reasoning tasks, solves them, and relies on an external code executor as a verifiable environment to validate both tasks and solutions. In the context of DR, EvolveSearch [447] iteratively selects high-performing rollouts (*i.e.,* task-solving trajectories) and re-optimizes the model on these data via supervised fine-tuning.

Advantages & Disadvantages. A key advantage of iterative self-evolving frameworks is their closed-loop design, where the model progressively improves its capabilities by tightly interleaving data generation with training. This autonomy enables scalable training without heavy reliance on external models or human annotations, and it allows exploration of data distributions that extend beyond handcrafted knowledge [367, 445, 443, 298, 76].

However, self-improvement also introduces significant risks. Previous studies have shown that, as iterations progress, distributional drift, reward hacking, and self-reinforcing errors may accumulate and degrade data quality, potentially leading to training collapse [293, 294, 114]. In addition, without robust validation mechanisms, the process may converge prematurely to narrow modes with limited performance ceilings [5, 15, 61, 18].

4.3. End-to-End Agentic Reinforcement Learning

Definition. In this section, we dive into the application of end-to-end agentic reinforcement learning (RL) in DR, *i.e.,* using RL algorithms to incentivize DR agents that can flexibly plan, act, and generate a final answer. We start with a brief overview, including commonly used RL algorithms and reward design for optimizing DR systems. For a clear explanation, we provide a glossary table in Table 3 to formally introduce the key variable in this section 4.3. Then we discuss two training practices: (i) *specific module optimization* and (ii) *entire pipeline optimization*.

4.3.1. Preliminary

RL algorithms in Deep Research. In DR, LLMs are trained to act as autonomous agents that generate comprehensive reports through complex query decomposition, multi-step reasoning, and extensive tool use. The primary RL algorithms used to train these agents include Proximal Policy Optimization (PPO) from OpenAI [276, 236], Group Relative Policy Optimization (GRPO) from DeepSeek [279, 105], and their variants [437].

Proximal Policy Optimization. PPO [276] is a clipped policy-gradient method that constrains updates within a trust region [242]. Given a current policy π_θ and a old policy $\pi_{\theta_{\text{old}}}$, the objective is to maximize the clipped surrogate:

$$L^{\text{PPO}}(\theta) = \mathbb{E}_t\left[\min\left(r_t(\theta)\hat{A}_t,\ \text{clip}\left(r_t(\theta), 1-\epsilon, 1+\epsilon\right)\hat{A}_t\right)\right], \tag{1}$$

$$r_t(\theta) = \frac{\pi_\theta(o^t \mid s_t)}{\pi_{\theta_{\text{old}}}(o^t \mid s_t)}. \tag{2}$$

where ϵ bounds the policy update and $\hat{A}_t$ is the estimated advantage. The advantage is computed using discounted returns or generalized advantage estimation (GAE) [275] as:

$$\hat{A}_t = \sum_{l=0}^{T-t} \gamma^l \cdot r_{t+l} + \gamma^{T-t+1} \cdot V_\phi(s_{T+1}) - V_\phi(s_t). \tag{3}$$

Table 3: Summary of key notations used in proximal policy optimization and group-relative policy optimization algorithms.

Symbol	Definition	Description
π_θ	Current policy	Parameterized LLM policy that generates actions (tokens or sequences) conditioned on a given state.
$\pi_{\theta_{\text{old}}}$	Reference (old) policy	A frozen snapshot of the policy before the current update, used for computing probability ratios and ensuring stable optimization.
q	Input query	Input question or prompt to the agent.
o	Model output	Final answer produced by the policy model.
o^t	Action at step t	The token generated by the policy model conditioned on state s_t.
s_t	State at step t	Context of the policy model at time step t.
$\mathcal{R}(o\|\cdot)$	Reward function	Scalar score assigned to output o for the input query q.
$r_t(\theta)$	Probability ratio	Ratio between current and reference policy probabilities, computed as $\frac{\pi_\theta(o^t\|s_t)}{\pi_{\theta_{\text{old}}}(o^t\|s_t)}$.
ϵ	Clipping threshold	Stability constant that limits update magnitude in PPO or adds numerical robustness in GRPO.
$\mathcal{G}$	Response group	A collection of multiple sampled responses corresponding to the same query s_t in GRPO.
m	Group size	The number of candidate responses in a response group $\mathcal{G}$.
o_j	j-th response in group	The j-th sampled output candidate among the m responses in group $\mathcal{G}$.

Here r_{t+l} denotes the immediate reward at time step $t+l$, $\gamma \in [0,1)$ is the discount factor balancing the importance of long-term and short-term returns; T is the terminal time step of the current trajectory (episode); s_{T+1} is the next state used for bootstrapping after termination, $V_\phi(s_t)$ is the value function predicted by the value network parameterized by ϕ. We define the empirical return $\hat{R}_t$ purely from rewards as:

$$\hat{R}_t = \sum_{l=0}^{T-t} \gamma^l r_{t+l}, \tag{4}$$

which represents the cumulative discounted rewards from time step t until the end of the episode. In PPO, the value function parameters ϕ are updated by minimizing the squared error between the predicted value and the empirical return:

$$\mathcal{L}^{\text{value}}(\phi) = \frac{1}{2}\,\mathbb{E}_t\left[\left(V_\phi(s_t) - \hat{R}_t\right)^2\right]. \tag{5}$$

Group Relative Policy Optimization. Group Relative Policy Optimization (GRPO) [279] extends PPO by normalizing rewards *within groups of responses* to the same query. Formally, given a group $\mathcal{G}$ of m responses $\{o_1, o_2, \ldots, o_m\}$ sampled for the same query s_t, each response is assigned a scalar reward R_j. The *group-relative advantage* for the j-th response is:

$$\hat{A}_j^{\mathcal{G}} = \frac{\mathcal{R}_j - \text{mean}(\{\mathcal{R}_i \mid i \in [m]\})}{\text{std}(\{\mathcal{R}_i \mid i \in [m]\}) + \epsilon}, \tag{6}$$

where $\text{mean}_{\mathcal{G}}$ and $\text{std}_{\mathcal{G}}$ denote the mean and standard deviation of rewards within group $\mathcal{G}$, and ϵ prevents numerical instability when the variance is small. The GRPO objective mirrors PPO's clipping mechanism but replaces $\hat{A}_t^{\mathcal{G}}$ with the group-relative advantage $\hat{A}_j^{\mathcal{G}}$:

$$\mathcal{L}^{\text{GRPO}}(\theta) = \mathbb{E}\left[\frac{1}{|\mathcal{G}|}\sum_{j=1}^{|\mathcal{G}|}\min\Big\{\frac{\pi_\theta(o_j \mid q)}{\pi_{\theta_{\text{old}}}(o_j \mid q)}\hat{A}_j^{\mathcal{G}},\ \text{clip}\Big(\frac{\pi_\theta(o_j \mid q)}{\pi_{\theta_{\text{old}}}(o_j \mid q)}, 1-\epsilon, 1+\epsilon\Big)\hat{A}_j^{\mathcal{G}}\Big\}\right]. \tag{7}$$

Comparison between PPO and GRPO in Deep Research. In PPO, each sampled output is optimized using an advantage signal derived from a value model. While this approach is effective, its performance is highly reliant on accurate value estimation and requires additional resources for training the value model. In contrast, GRPO optimizes by contrasting each response against others within the same group. This shifts the focus to a relative-quality comparison among competing hypotheses, simplifying implementation while maintaining strong performance.

Reward Design in Deep Research Agents. During the RL training of DR agents, the reward model, denoted as $\mathcal{R}(\cdot)$, assesses the quality (*e.g.,* correctness) of the agents' outputs and produces scalar signals to enable policy optimization algorithms such as PPO and GRPO. Reward design takes a critical role in training LLM. There are two common reward design paradigms in DR systems, *i.e., rule-based rewards* and *LLM-as-judge rewards*.

- *Rule-based Rewards* $\mathcal{R}_{rule}(\cdot)$. Rule-based rewards are derived from deterministic, task-specific metrics such as Exact Match (EM) and F1 score [277]. In the context of research agents, EM is a commonly used binary score that indicates whether a generated answer perfectly matches a ground-truth string [156, 145, 144]. Alternatively, the F1 score (*i.e.,* the harmonic mean of precision and recall calculated over token overlap) is also used to reward outputs [36, 37]. However, a key limitation of rule-based rewards is that they are primarily suited for tasks with well-defined, short-span ground truths (*e.g.,* a specific entity name) and struggle to evaluate multi-answer or open-ended questions effectively.
- *LLM-as-judge Rewards* $\mathcal{R}_{LLMs}(\cdot)$. The LLM-as-judge approach uses an external LLM to evaluate the quality of an agent's output and assign a scalar score based on a predefined rubric [7]. Formally, for an output o to an input query q, the reward assigned by an LLM judge ϕ can be formulated as:

  $$\mathcal{R}_{\text{LLMs}}(o \mid q) = \mathbb{E}_{\text{criteria}\in\mathcal{C}}\left[\phi(o, q, \text{criteria})\right]$$

  where $\mathcal{C}$ is the set of evaluation criteria (*e.g.,* accuracy, completeness, citation quality, clarity, etc) and $\phi(\cdot)$ returns a scalar score for each criterion.

4.3.2. End-to-end Optimization of a Specific Module

Definition. End-to-end optimization of a specific module focuses on applying RL techniques to improve individual components within a DR system, such as the query planning, document ranking, or planning modules.

Representative work. Within DR, most existing work trains the query planner [134, 499, 492, 192] while freezing the parameters, leaving components such as retrieval. MAO-ARAG [37] treats DR as a multi-turn process where a planning agent orchestrates sub-agents for information seeking. PPO propagates a holistic reward (*e.g.,* final F1 minus token and latency penalties) across all steps,

enabling end-to-end learning of the trade-offs between answer quality and computational cost. AI-SearchPlanner [213] decouples a lightweight search planner from a frozen QA generator. PPO optimizes the planner with dual rewards: an outcome reward for improving answer quality and a process reward for reasoning rationality. A Pareto-regularized objective balances utility with real-world cost, guiding the planner on when to query or stop.

Advantages & Disadvantages. Single-module optimization usually focuses on training a single core component (*e.g.,* the planning module) while keeping the others fixed. Optimizing this critical module can improve the performance of a DR system by enabling more accurate credit assignment, more sophisticated algorithm design for the target module, and reduced training data and computational costs. However, this approach restricts the optimization space and may be inadequate when other frozen modules contain significant design or performance flaws.

4.3.3. End-to-end Optimization of an Entire Pipeline

Definition. End-to-end pipeline optimization involves jointly optimizing all components and processes from input to output (*e.g.,* query decomposition, search, reading, and report generation) to achieve the best overall performance across the DR workflow.

Representative work on Multi-Hop Search. Some work focuses on enhancing the capability of multi-hop search by training the entire DR systems end-to-end [247, 300, 301, 481, 394]. For example, Jin et al. [145, 144] present Search-R1, the first work to formulate search-augmented reasoning as a fully observable Markov Decision Process and to optimize the entire pipeline via RL, containing query planning, retrieval, and extracting the final answer. By masking retrieved tokens in the policy-gradient loss, the model learns to autonomously decide when and what to search while keeping the training signal on its own generated tokens. Meanwhile, Song et al. [300] introduces R1-Searcher, a two-stage RL method in which the DR agent learns when to invoke external searches and how to use retrieved knowledge via outcome rewards. However, it has been observed that pure RL training often leads to over-reliance on external retrieval, resulting in over-searching [301]. To mitigate this issue, R1-Searcher++ [301] first cold-starts the DR agent via an SFT, then applies a knowledge-assimilation RL process to encourage the agent to internalize previously retrieved documents and avoid redundant retrievals.

Besides the above early effort, recent work extends the naive Search-R1 by integrating multi-reward signals or improving the training environment. *For the reward design*, R-Search [468] trains models to decide when to retrieve and how to integrate external knowledge in both single-hop and multi-hop question answering. The framework improves answer quality and evidence reliability by optimizing reasoning–search trajectories under a multi-objective reward design. *For the training environment*, ZeroSearch [311] and O^2-Searcher [212] simulate a search engine to develop retrieval capabilities without accessing the actual web, providing a more controllable setting for RL training. In contrast, DeepResearcher [481] operates directly in real-world web environments, learning to plan, search, verify, and autonomously answer open-domain questions.

Besides a basic document retrieval tool, some work also integrates additional information-seeking tools, teaching the DR agent to flexibly combine them. MMSearch-R1 [394] stands out as the first RL-trained multimodal model that learns when and how to search the text or image from the web on demand. HierSearch [324] introduces a hierarchical DR framework for enterprise scenarios that involve both local and web knowledge sources. Other work [200, 247, 110] integrates knowledge

graphs into DR agents to achieve efficient multi-hop reasoning.

Representative work on Long-Chain Web Search. Besides the relatively simple multi-hop QA tasks, more recent work also applies end-to-end pipeline optimization to address longer-chain web search problems. Prior works, such as WebDancer [391], WebSailor [174], and Kimi-K2 [338], have focused on advancing more intricate multi-hop tasks, including GAIA [215] and BrowseComp [378]. These approaches combine data synthesis with end-to-end reinforcement learning training, thereby enabling more extensive iterations in the DR process.

Furthermore, Gao et al. [85] presents ASearcher, which scales end-to-end RL to extreme long-horizon search. A fully asynchronous RL engine removes the 10-turn ceiling that plagued earlier systems, allowing trajectories of 40+ turns and 150k tokens to be optimized without blocking GPU updates. Coupled with an autonomous QA-synthesis agent that injects noise and fuzzes questions for difficulty, the whole pipeline is operated end-to-end, from synthetic data creation to multi-turn policy optimization. SimpleDeepSearcher [314] leverages real-web simulation and distilled SFT to deliver agentic search capability without heavy RL, yet stays fully compatible with lightweight RL refinement. WebAgent-R1 [382] and DeepDiver [282] are training web agents through end-to-end multi-turn RL algorithms.

In addition, some works [180, 63, 64, 62] have studied Deep Research systems across multiple tool-calling scenarios. For example, Li et al. [180] introduces Chain-of-Agents (CoA), a novel paradigm that distills the capabilities of multi-agent systems into a single LLM. CoA enables native, end-to-end complex problem-solving by dynamically orchestrating multiple role-playing and tool agents within one model. Through multi-agent distillation and agentic RL, the authors train Agent Foundation Models (AFMs) in an end-to-end approach that achieves excellent performance on diverse web search benchmarks, while significantly reducing computational overhead compared to traditional multi-agent systems. Tool-Star [63] and ARPO [64] investigate how to effectively leverage tools in long-horizon tasks such as Deep Research, and use the GRPO algorithm to optimize the entire pipeline end-to-end. Additionally, AEPO [62] further improves rollout efficiency and, based on ARPO, optimizes both performance and efficiency for the end-to-end tool-use pipeline.

Advantages & Disadvantages. These end-to-end methods model the entire DR system as a multi-turn search process, achieving comprehensive optimization across reasoning, query rewriting, knowledge retrieval, tool invocation, and answer generation. This modeling and optimization approach is not only flexible but also allows for different objectives to be emphasized through the design of reward functions. However, these methods also have drawbacks, including sparse rewards, excessively long responses, and unstable training. Continuous optimization is needed to further enhance the effectiveness, stability, and efficiency of DR systems.

Takeaway

- **RL Algorithms**: PPO provides stable updates based on absolute rewards, while GRPO leverages group-relative advantages to reduce resource requirements.
- **Specific Module End-to-End Optimization**: Targets a critical component (*e.g.,* planner or searcher) for RL training, improving overall performance at lower cost, though limitations in other frozen modules cannot be addressed.
- **Entire Pipeline End-to-End Optimization**: Optimizes the full DR workflow, including retrieval, reasoning, tool use, and answer generation, yielding holistic gains but facing sparse rewards, long outputs, and training instability.

5. Evaluation of Deep Research System

DR techniques have been applied to a wide range of downstream tasks, including healthcare [7], financial report generation [342], and survey generation [366]. In this section, we systematically review common benchmarks and evaluation protocols for DR systems across three representative scenarios: (i) *information seeking*, (ii) *report generation*, and (iii) *AI for research*. These scenarios reflect the most prevalent applications of DR agents. Each category poses distinct challenges, illuminating the limitations of current systems while offering practical insights to guide future advances.

5.1. Agentic Information Seeking

Evaluating the effectiveness of agentic information-seeking is a critical component of assessing DR systems. In DR scenarios, information seeking is not a single QA task but a multi-stage, iterative, and cross-domain process in which agents must continuously explore, reformulate, and synthesize information from diverse sources. To capture this complexity, benchmark design has evolved from early static single-hop retrieval tasks such as Natural Questions (NQ) [165] to dynamic web environments requiring multi-hop reasoning and complex interactions, *e.g.,* BrowseComp [378] and HotpotQA [425]. In this section, we review representative benchmarks and evaluation frameworks along two dimensions: query complexity and interaction environment complexity.

Table 4: Comprehensive overview of existing and emerging benchmarks for Deep Research Systems that focus on question answering scenarios.

Benchmark (with link)	Date	Aspect	Data size (train/dev/test)	Evaluation metrics
NQ	2019	QA	307373/7830/7842	Exact Match / F1 / Accuracy
SimpleQA	2024	QA	4,326	Exact Match / F1 / Accuracy
HotpotQA	2019	QA	90124 / 5617 / 5813	Exact Match / F1 / Accuracy
2WikiMultihopQA	2020	QA	167454/12576/12576	Exact Match / F1 / Accuracy
Bamboogle	2023	QA	8600	Exact Match / F1 / Accuracy
MultiHop-RAG	2024	QA	2556	Exact Match / F1 / Accuracy
MuSiQue	2022	QA	25K	Exact Match / F1 / Accuracy
GPQA	2023	QA	448	Accuracy
GAIA	2023	QA	450	Exact Match
BrowseComp	2025	QA	1266	Exact Match
BrowseComp-Plus	2025	QA	830	Accuracy / Recall / Search Call / Calibration Error
HLE	2025	QA	2500	Exact Match / Accuracy

5.1.1. Complex Queries

The evolution of benchmarks for agentic information seeking has closely followed the increasing complexity of query demands. Early benchmarks such as NQ [165], TriviaQA [151], and SimpleQA [377] established the foundation for question answering research. These datasets focused on single-hop queries, where answers could be retrieved with a single lookup or were already contained within the LLM's parameters. While such tasks provided a controlled starting point, they could not capture the reasoning and synthesis required in DR.

Table 5: Comprehensive overview of existing and emerging benchmarks for Deep Research Systems that focus on more boarder scenarios.

Benchmark (with link)	Date	Aspect	Data size (train/dev/test)	Evaluation metrics
FRAMES	2024	QA	824	Exact Match / F1 / Accuracy
InfoDeepSeek	2025	QA	245	Accuracy / Utilization / Compactness
AssistantBench	2025	QA	214	F1 / Similarity
Mind2Web	2025	QA	2350	Accuracy / F1 / Step Success Rate
Mind2Web 2	2025	QA	130	Agent-as-a-Judge
Deep Research Bench	2025	QA	89	Precision / Recall / F1
DeepResearchGym	2025	QA	96,000	Report Relevance / Retrieval Faithfulness / Report Quality
WebArena	2024	Complex Task	812	Correctness
WebWalkerQA	2025	QA	680	Accuracy / Action Count
WideSearch	2025	QA	200	LLM Judge
MMInA	2025	Complex Task	1050	Success Rate
AutoSurvey	2024	Survey Generation	530,000	Citation Quality / Content Quality
ReportBench	2025	Survey Generation	600	Content Quality / Cited Statement / Non-Cited Statements
SurveyGen	2025	Survey Generation	4200	Topical Relevance / Academic Impact / Content Diversity
Deep Research Comparator	2025	Report Generation	176	BradleyTerry Score
DeepResearch Bench	2025	Report Generation	100	LLM Judge
ResearcherBench	2025	Report Generation	65	Rubric Assessment / Factual Assessment
LiveDRBench	2025	Report Generation	100	Precision / Recall / F1
PROXYQA	2025	Report Generation	100	LLM Judge
SCHOLARQABENCH	2025	Report Generation	2967	Accuracy / Citations / Rubrics
Paper2Poster	2025	Poster Generation	100	Visual Quality / Textual Coherence / VLM Judge
PosterGen	2025	Poster Generation	10	Poster Content / Poster Design
P2PInstruct	2025	Poster Generation	121	LLM Judge
Doc2PPT	2022	Slides Generation	6000	ROUGE / Figure Subsequence / Text-Figure Relevance
SLIDESBENCH	2025	Slides Generation	7000/0/585	Text / Image / Layout / Color / LLM Judge
Zenodo10K	2025	Slides Generation	10,448	Content / Design / Coherence
TSBench	2025	Slides Generation	379	Editing Success / Efficiency
AI Idea Bench	2025	Idea Generation	0/0/3495	LLM Judge
Scientist-Bench	2025	Idea Generation, Experimental Execution	0/0/52	LLM Judge, Human Judge
PaperBench	2025	Experimental Execution	0/0/20	LLM Judge
ASAP-Review	2021	Peer Review	0/0/8877	Human / ROUGE / BERTScore
DeepReview	2025	Peer Review	13378/0/1286	LLM Judge
SWE-Bench	2023	Software Engineering	0/0/500	Environment
ScienceWorld	2022	Scientific Discovery	3600/1800/1800	Environment
GPT-Simulator	2024	Scientific Discovery	0/0/76369	LLM Judge
DiscoveryWorld	2024	Scientific Discovery	0/0/120	LLM Judge
CORE-Bench	2024	Scientific Discovery	0/0/270	Environment
MLE	2024	Machine Learning Engineering	0/0/75	Environment
RE-Bench	2024	Machine Learning Engineering	0/0/7	Environment
DSBench	2024	Data Science	0/0/540	Environment
Spider2-V	2024	Data Science	0/0/494	Environment
DSEval	2024	Data Science	0/0/513	LLM Judge
UnivEARTH	2025	Earth Observation	0/0/140	Exact Match
Commit0	2024	Software Engineering	0/0/54	Unit test

As research questions grew more complex, benchmarks evolved from simple fact retrieval to multi-step reasoning challenges. Multi-hop QA datasets assess an agent's ability to reformulate queries and build reasoning chains across documents. HotpotQA [425], one of the earliest and most widely used multi-hop datasets, requires reasoning across multiple Wikipedia articles using supporting facts to derive the answer. 2WikiMultihopQA [115] extends this by integrating information from two

separate Wikipedia pages per question, emphasizing cross-document reasoning. Bamboogle [251] consists of 125 two-hop questions generated from random Wikipedia articles, testing the ability to decompose and reason over complex queries. MultiHop-RAG [332] is the RAG dataset designed specifically for multi-hop queries, categorizing questions into four types: inference, comparison, temporal, and null queries. MuSiQue [344] adopts a bottom-up approach, systematically pairing composable single-hop questions where one reasoning step depends on another. FRAMES [163] simulates realistic multi-document queries to evaluate an LLM's ability to retrieve relevant facts, reason accurately, and synthesize information into coherent responses. However, most of these benchmarks rely on structured, linear reasoning paths, which fall short of reflecting the inherent ambiguity and branching, non-linear exploration required in real-world research scenarios.

Recent benchmarks have begun to capture this growing complexity, placing greater emphasis on the in-depth and progressive exploration of complex topics. For instance, GPQA [265] is a graduate-level dataset in physics, chemistry, and biology that tests both domain experts and skilled non-experts, requiring extensive reasoning and problem-solving. Similarly, GAIA [215] provides 466 carefully designed questions that require multi-step reasoning, real-world knowledge retrieval, and complex generation. HLE [249] aims to be a comprehensive, fully closed academic benchmark across dozens of disciplines, including mathematics, humanities, and natural sciences, designed to advance reasoning skills. Its questions cannot be quickly answered through an online search. These recent datasets challenge agents to operate in environments that better reflect the ambiguity, branching evidence paths, and iterative synthesis characteristic of real-world DR systems.

5.1.2. Interaction Environment

As agent capabilities have advanced, evaluation based solely on static environments and fixed corpora is no longer sufficient. Consequently, a series of benchmarks has been developed to reflect the scale and dynamics of real-world web environments, requiring agents to interact with, navigate, and creatively explore web pages to obtain complex or hard-to-find information.

Some studies have incorporated browsing tools such as Google and Bing into benchmarks, enabling agents to directly retrieve and extract information from the live web. For example, InfoDeepSeek [402] and AssistantBench [434] present challenging tasks that require agents to integrate multiple search and browsing tools in real-time web environments, testing their ability to operate dynamically. Mind2Web [57] replaces the overly simplified, simulated environments common in other datasets with authentic, dynamic, and unpredictable real-world websites, providing complete records of user interactions, webpage snapshots, and network traffic. Its successor, Mind2Web 2 [98], was subsequently introduced to more rigorously evaluate agent-based search systems on realistic, long-horizon tasks that involve live web search and browsing. BrowseComp [378] and BrowseComp-Plus [38] demand persistent navigation to locate hard-to-find, entangled information across multiple sites. Moreover, DeepResearchBench [17] offers a large-scale RetroSearch environment that reduces task degradation and network randomness while evaluating LLM agents on complex real-world web research tasks. DeepResearchGym [49] complements this by providing an open-source sandbox with a reproducible search API and a rigorous evaluation protocol, promoting transparency and reproducibility in DR area.

Building on this trend, subsequent datasets have increasingly emphasized the authenticity and complexity of interactive environments. WebArena [488] provides a highly realistic and reproducible environment for language-guided agents, built from fully functional websites across four domains. WebWalkerQA [392] assesses LLMs' ability to systematically traverse website subpages and extract

high-quality data through interactive actions such as clicking, specifically testing complex, multi-step web interactions. WideSearch [388] focuses on a critical yet under-evaluated task: requiring agents to thoroughly and accurately acquire all large-scale atomic information that meets a set of criteria and organize it into a structured output. MMInA [341] extends these challenges by providing a multi-hop, multi-modal benchmark for embodied agents performing integrated internet tasks on realistic, evolving websites, ensuring high realism and applicability to natural user tasks. Together, these benchmarks illustrate a clear trend: web-oriented evaluation environments are becoming increasingly human-like, visually grounded, diverse, complex, and realistic, pushing the limits of agentic information seeking and DR in dynamic, real-world settings.

5.2. Comprehensive Report Generation

Another critical dimension in evaluating DR systems is their capacity to generate comprehensive reports. Unlike single-point answers or brief summaries, comprehensive reports require systems to integrate information from multiple sources and modalities into structured, logically coherent, and broadly informative outputs. This process involves information aggregation, content organization, factual consistency verification, and clarity of expression, and is therefore regarded as a core indicator of a DR system's overall capability. Below, we introduce the relevant benchmarks by task type.

5.2.1. Survey Generation

A closely related task is survey generation, which involves producing structured overviews or syntheses of a specific scientific topic by aggregating information from diverse sources. Thanks to the clear citation structure provided by gold-standard references, survey generation has been widely used to evaluate the capabilities of DR systems. AutoSurvey [366] gathers arXiv articles of varying lengths and uses a multi-LLM-as-judge framework to evaluate survey generation in terms of speed, citation quality, and content quality. Moreover, ReportBench [175] is a systematic benchmark for evaluating research reports generated by large language models. It focuses on two key aspects: the relevance of citations and the reliability and accuracy of the report's statements. The evaluation corpus is constructed using high-quality survey papers published on arXiv as the gold standard. SurveyGen [14] is another survey-generation dataset, containing over 4,200 human-written surveys with chapter-level structure, cited references, and rich metadata. It enables comprehensive evaluation of content quality, citation accuracy, and structural consistency.

5.2.2. Long-Form Report Generation

Other benchmarks focus on different types of report generation tasks and introduce alternative evaluation frameworks. For example, Deep Research Comparator [27] provides a unified evaluation platform for DR agents, enabling systematic assessment of long-form reports and their intermediate reasoning processes through side-by-side comparison, fine-grained human feedback, and ranking mechanisms. DeepResearch Bench [66] is a benchmark of 100 PhD-level research tasks, introducing two evaluation methods for generated reports: a reference-based assessment of overall quality and a citation-based evaluation of retrieval accuracy. ResearcherBench [413] comprises 65 research questions focused on evaluating the capabilities of advanced agent systems on cutting-edge AI science problems, using an evaluation framework that combines rubric assessment and factual evaluation. LiveDRBench [130] is a benchmark for DR tasks, offering challenging science and world-event queries and evaluating systems via intermediate reasoning steps and factual sub-propositions. PROXYQA [322] uses human-designed meta-questions and corresponding proxy questions to indirectly

assess knowledge coverage and information richness, providing an objective measure of long-form text generation quality. SCHOLARQABENCH [8] is a benchmark for scientific literature synthesis tasks in multiple formats, comprising 2,967 expert-written queries and 208 long-form answers across the field of computer science. Evaluating research reports is particularly challenging because there is no single gold-standard answer, and multiple valid perspectives exist for assessing quality. The diversity of acceptable content, reasoning approaches, and presentation styles makes it difficult to define objective metrics. As a result, most benchmarks rely on LLM-as-judge methods [169], leveraging large language models' reasoning and knowledge to provide scalable, consistent, and nuanced evaluations of content quality, factual accuracy, citation relevance, and structural coherence.

5.2.3. Poster Generation

Poster generation can be viewed as a highly condensed and visually structured variant of the comprehensive report generation task. Unlike multi-page reports, a scientific poster is typically a single-page, high-density summary that concisely presents the research motivation, methodology, results, and conclusions in a format that is both navigable and visually engaging. For DR systems, this task imposes unique challenges: not only must the system aggregate and synthesize information from multiple heterogeneous sources, such as research papers, notes, and presentation slides, but it must also transform that content into an effective visual layout. Evaluation of poster generation typically focuses on three main aspects: factual completeness, visual communication effectiveness, and readability. For example, Paper2Poster [244] focuses exclusively on AI research papers. The dataset comprises 100 paper-poster pairs, covering 280 distinct topics across subfields. A comprehensive evaluation framework is introduced, comprising four key dimensions: visual quality, text coherence, VLM-based quality judgment, and PaperQuiz, which is a novel metric designed to assess how effectively a poster communicates the core knowledge of the original paper. PosterGen [464] adopts a two-dimensional evaluation protocol, dividing the assessment into poster content and poster design. It introduces a VLM-based metric to evaluate key design aspects, including layout balance, readability, and aesthetic consistency. P2PInstruct [315] is a large-scale instruction dataset for paper-to-poster generation, containing over 30,000 high-quality instruction–response pairs. It covers the full pipeline from image element processing and text generation to final layout formatting.

5.2.4. Slides Generation

Slide generation represents another critical pathway for evaluating the comprehensive capabilities of DR systems. This task challenges a system not only to comprehend and summarize large volumes of heterogeneous information sources, but also to transform the distilled content into a structured, slide-based presentation format. The objectives of slide generation encompass information distillation, logical structuring, and presentation-oriented expression, making it a strong indicator of a system's ability to coordinate across multiple dimensions. Common benchmark tasks and datasets for this evaluation often involve generating slides from meeting transcripts, research papers, long-form reports, or collections of web documents. These benchmarks typically assess whether a model can maintain content integrity and factual accuracy while performing high-quality information selection and organization. Doc2PPT [82] collects paired documents and their corresponding slide decks from academic proceedings. It conducts detailed post-processing for evaluation, using metrics such as Slide-Level ROUGE, Longest Common Figure Subsequence, Text-Figure Relevance, and Mean Intersection over Union. For Example, SLIDESBENCH [91] is a benchmark comprising 7,000 training examples and 585 test examples, derived from 310 slide decks across 10 distinct domains. It supports two types of evaluation: reference-based evaluation, which measures similarity to target (gold) slides,

and reference-free evaluation, which assesses the design quality of the generated slides independently. Zhang et al. introduced Zenodo10K [478], a new dataset collected from Zenodo, a platform hosting diverse, openly licensed artifacts across various domains. They also proposed PPTEval [478], an evaluation framework leveraging GPT-4.0 as the judge to assess presentation quality along three dimensions: content, design, and coherence. TSBench [152] is a benchmark dataset specifically designed to evaluate the slide editing capabilities of models and frameworks. It includes 379 distinct editing instructions along with the corresponding slide modifications.

5.3. AI for Research

AI for Research seeks to harness artificial intelligence to advance scientific discovery, either by automating processes or by assisting researchers in accelerating their work [31]. Its applications and corresponding benchmarks include (i) *idea generation*, (ii) *experimental execution*, (iii) *academic writing*, and (iv) *peer review*. Unlike report generation, research goes beyond producing extended outputs; it requires the creation of new perspectives, conclusions, and knowledge, thereby necessitating mechanisms for evaluating novelty.

5.3.1. Idea Generation

A key challenge in research lies in generating genuinely novel ideas and, more importantly, in reliably assessing their novelty. Such evaluation is typically conducted by human experts, but it remains difficult and resource-intensive. Existing approaches generally fall into two categories. The first is human- or LLM-based evaluation. Si et al. [296] recruited over 100 NLP researchers to evaluate the novelty of ideas generated by humans, LLMs, and human–LLM collaboration. However, this process is not easily scalable and proves difficult even for domain experts. Moreover, they investigated LLMs' ability to assess novelty, finding that LLM judgments show lower agreement with expert reviewers than human evaluations. Li et al. [177] and Gao et al. [87] leverage LLMs through direct prompting to evaluate novelty. To enhance the reliability of LLMs' judgments, Lu et al. [196] and Su et al. [306] integrate LLMs with the Semantic Scholar API and web access, enabling them to evaluate ideas against related literature. AI Idea Bench 2025 [258] provides a benchmark for quantifying and comparing ideas generated by LLMs. It incorporates 3,495 representative papers published in AI-related conferences after October 10, 2023, together with their corresponding inspiration papers. Furthermore, it introduces an evaluation framework to assess whether the ideas derived from inspiration papers are consistent with the ground-truth papers. The second category is density-based evaluation, which relies on the absolute local density in the semantic embedding space to measure novelty. Wang et al. [365] introduces the Relative Neighbor Density (RND) algorithm, which evaluates novelty by examining the distributional patterns of semantic neighbors rather than relying solely on absolute local density. Moreover, they construct large-scale semantic embedding databases for novelty assessment, encompassing more than 30 million publications across two distinct domains.

5.3.2. Experimental Execution

Evaluation of experimental execution typically involves both objective and subjective assessments. Objective evaluation generally emphasizes the outcomes produced in specific environments, such as benchmark performance or compiler outputs. For example, Lu et al. [196] and Weng et al. [386] directly adopt the results of downstream tasks as evaluation metrics, while Tang et al. [326] leverages compiler outputs to assist LLMs in refining experiments and correcting code errors. Subjective evaluation involves leveraging either humans or LLMs to assess the quality of experimental designs.

For example, Tang et al. [326] employs LLMs to compare code implementations with atomic research ideas, thereby verifying whether the code satisfies the intended requirements of the ideas. Starace et al. [304] employs LLMs to evaluate source code, documentation, and configuration files against human-designed rubrics to derive a final grade.

5.3.3. Academic Writing

The evaluation of academic writing differs substantially from general report generation. It requires not only factual accuracy, logical coherence, and clarity, but also alignment with underlying ideas and experimental results, proper integration of citations from related work, and effective visualization of findings. To capture these multifaceted criteria, Lu et al. [196] employ LLMs to assess writing along dimensions such as originality, quality, clarity, and significance. Similarly, Höpner et al. [117] train a domain-specific LLM to predict citation counts and review scores as proxies for paper quality, addressing the limitations of generic LLMs in academic evaluation. Building on this direction, Starace et al. [304] introduce PaperBench, a benchmark designed to assess AI agents' ability to replicate AI papers. The benchmark includes 20 ICML 2024 Spotlight and Oral papers, and evaluates replication quality using LLMs guided by manually constructed rubrics that hierarchically decompose each task into graded subtasks. Tang et al. [326] propose Scientist-Bench, a comprehensive benchmark built from top-cited papers published between 2022 and 2024 across 16 research areas and multiple expertise levels. It evaluates dimensions such as technical novelty, methodological rigor, empirical validation, and potential impact, closely reflecting the criteria used in the ICLR review process. To further push the boundaries of scientific evaluation, Xu et al. [413] present ResearcherBench, a more challenging benchmark for evaluating DR systems, which consists of 65 research questions carefully curated from real-world scientific contexts across 35 AI subfields. They also propose a dual evaluation framework that combines rubric-based assessment to evaluate the quality of insights with factual evaluation that measures citation faithfulness and evidence coverage.

5.3.4. Peer Review

AI for peer review seeks to leverage an AI agent to generate feedback on scientific papers. However, evaluating such feedback is challenging, as reviews are typically lengthy and inherently subjective. Yuan et al. [442] introduced ASAP-Review, a large-scale benchmark that collects 8,877 AI papers from ICLR (2017–2022) via OpenReview and NeurIPS papers (2016–2019) via the official proceedings, along with their corresponding reviews. The dataset is annotated across multiple dimensions, including Motivation, Originality, Soundness, Substance, Replicability, Clarity, and Comparison. To evaluate generated reviews, they employ automatic metrics such as ROUGE and BERTScore [454], as well as human judgments. Lu et al. [196] collect 500 ICLR 2022 papers from OpenReview to establish a benchmark for the peer review task, subsequently employing self-reflection, few-shot examples, and response ensembling with LLMs to assess the quality of the generated reviews. Weng et al. [384] introduces the REVIEW-5k dataset, which contains 782 test samples collected from ICLR 2024. Each sample includes the paper title, abstract, LaTeX or Markdown source, and the corresponding review comments. The dataset also provides structured review information, including summaries, strengths and weaknesses, clarification questions, and review scores. For evaluation, they employ Proxy Mean Squared Error (Proxy MSE) and Proxy Mean Absolute Error (Proxy MAE), which leverage multiple independent reviews of the same submission as unbiased estimators of its true rating [305]. Similarly, Gao et al. [90] constructed REVIEWER2, a dataset comprising 27,805 papers and reviews collected from CONLL-16, ACL-17, COLING-20, ARR-22, ICLR-17–23, and NeurIPS-16–22. Zhu et al. [498] introduces DeepReview-Bench, a dataset of 1.2K ICLR 2024–2025 submissions collected

from OpenReview. The dataset includes textual reviewer assessments, interactive rebuttal-stage discussions, and standardized scoring information. For quantitative evaluation, they employ MAE, MSE, accuracy, F1, and Spearman correlation, while qualitative evaluation is conducted under the LLM-as-a-judge paradigm [169, 100, 170] across five dimensions: constructive value, analytical depth, plausibility, technical accuracy, and overall quality.

5.4. Software Engineering

In addition to the above scenarios, DR agents can also be applied to software engineering, representing a shift from assisting with isolated code snippets to autonomously executing complex software development tasks [186]. A pioneering work is SWE-Bench [141], a benchmark designed to evaluate whether AI agents can resolve real-world GitHub issues. Although SWE-Bench does not yet cover full end-to-end software development, it marks an important step toward bridging idealized benchmarks with practical scenarios. Meanwhile, DR agents have been deployed in a wide range of complex software engineering domains, including scientific discovery [359, 360, 129, 71, 297, 270, 318], machine learning experimentation [124, 26, 387, 179], data science [148, 23, 461], earth observation [154], and software library completion [472, 241].

6. Challenges and Outlook

6.1. Retrieval Timing

Although determining when to retrieve has become a standard feature of various DR systems, several fundamental challenges remain. Existing DR systems, such as Search-R1 Jin et al. [145], rely too heavily on answer correctness to guide the entire search pipeline and lack fine-grained guidance on when to retrieve, leading to both over-retrieval and under-retrieval Wu et al. [396]. Moreover, even with continued retrieval, the model may still produce an incorrect answer, and when no relevant evidence can be retrieved, generating an answer regardless risks misleading users, particularly in safety-critical domains such as healthcare and finance.

Future research could explore fine-grained reward designs that assess, at each step, whether the model lacks the knowledge needed to answer the question [396] and whether relevant documents can be retrieved [374]. Such signals would help determine when retrieval is necessary. Beyond deciding when to retrieve, the system should also evaluate whether the model's post-retrieval answer is correct and, after completing the entire process, estimate the uncertainty of the final output to avoid misleading users.

6.2. Memory Evolution

DR systems aim to mimic the research process of human experts by integrating autonomous planning, multi-source information acquisition, dynamic memory management, and deep knowledge synthesis. However, existing memory modules face significant challenges in fulfilling this vision. To develop more capable and DR systems, it is essential to re-examine the role of memory and identify future directions in personalization, structurization, adaptivity, and goal-driven optimization [84, 136, 74].

6.2.1. *Proactive Personalization Memory Evolution*

Recap of previous work. Personalized memory in current systems often serves as a *passive knowledge buffer*, primarily designed to record user interaction histories and preferences for enhancing retrieval-based responses [457, 231]. While effective for maintaining conversational consistency, this potentially limits the agent to a reactive stance [231]. The memory's primary function is to serve as a repository of past events, such as the fine-grained, timestamped interactions stored in episodic memory or the consolidated user traits in semantic memory [368, 457]. Even advanced management techniques, such as the reflective mechanisms proposed by RMM [325], are chiefly focused on optimizing the organization and retrieval of this historical data to improve the relevance of future responses, rather than enabling forward-looking planning.

A necessary paradigm shift is emerging, moving from memory as a historical archive to memory as a dynamic, predictive user model [231]. To transition from mere assistants to true collaborators, future agents must leverage memory to engage in proactive reasoning. The foundation for such a model is a comprehensive, multi-dimensional user profile, as conceptualized in benchmarks like PersonaLens, which integrates demographics, detailed cross-domain preferences, and summaries of past interactions to form a holistic view of the user [473]. Early steps in this direction can be seen in goal-oriented systems like MemGuide, which employs proactive reasoning by using the user's task intent and analyzing missing information *slots* to strategically filter memories [68]. The ultimate vision is for future memory modules to empower agents as proactive partners by capturing not only explicit preferences but also implicit signals, such as communication styles and latent intents. The PaRT framework exemplifies this future, using its dynamic user profile to actively guide conversations by generating personalized new topics and retrieving real-time external information [231]. A blueprint for the underlying architecture can be found in systems like MIRIX, whose multi-component design could support diverse proactive functions; for instance, its Procedural Memory could store workflows to anticipate a user's next steps in a complex task [368]. By integrating these capabilities, the system can anticipate user needs, proactively acquire and present relevant information, and adapt its interaction style in real time, thus shifting from reactive responses to proactive planning for more effective, intuitive, personalized support [231].

6.2.2. *Cognitive-Inspired Structured Memory Evolution*

The predominant memory architecture in current systems (*e.g.,* vector stores of text chunks) follows a *flat* storage paradigm, which lacks the capacity to capture deep logical or relational structures between knowledge elements. This architectural deficiency fundamentally hinders complex multi-hop reasoning, as the system cannot traverse explicit relationships between concepts. Recent work has begun to address this by moving towards structured representations like knowledge graphs, where entities are explicitly linked by semantic relationships, thereby providing a scaffold for more sophisticated inference [46, 411, 262]. Moreover, memory is often treated as a static snapshot, making it incapable of addressing the temporal dynamics of knowledge. This is a critical failure point in real-world scenarios where information evolves. Pioneering work has introduced bi-temporal models into knowledge graphs [262], allowing memory to track not only when a fact was recorded but also the period during which it was valid in the real world, using non-destructive updates that preserve historical context [46, 262].

A key future direction is to integrate these structured memory representations with dynamic, autonomous update mechanisms, drawing inspiration from cognitive science. Agents should be capable of autonomously transforming unstructured inputs into structured representations (*e.g.,*

knowledge graphs [46, 262], operator trees [47], or multi-faceted memory fragments [448]) in real time during interaction. Importantly, this is not a one-time conversion, but a continuous *stream-processing* procedure. As new information arrives, the memory structure must dynamically expand, prune, and reorganize itself [46, 414, 187]. This vision is partially realized in systems that employ agentic, cognitive-inspired operations such as `INSERT`, `FORGET`, and `MERGE` to refine memory content [187], or processes like *memory evolution*, in which new memories trigger updates and recontextualization of existing, linked memories [414]. The ultimate goal is to create a unified cognitive framework that addresses the dual challenges of representational depth and timeliness of knowledge. This framework would likely emulate the distinction between human episodic and semantic memory, a principle already explored in several advanced architectures [222, 262, 448, 423], allowing an agent to both ground its knowledge in specific experiences and evolve a generalized, abstract understanding of the world.

6.2.3. Goal-Driven Reinforced Memory Evolution

Existing strategies for memory retention are primarily heuristic-based, relying on static signals such as recency or semantic relevance [417, 463]. However, these heuristics fail to guarantee that preserved memories are truly useful for achieving the final task goal, as they often ignore the interconnected memory cycle effect of storage, retrieval, and utilization [463]. A more powerful paradigm is to formulate memory management as a decision-making problem within a RL framework [436, 491, 194, 485, 417]. In this approach, the agent learns an optimal policy for memory operations, such as updating a fixed-length internal state [436, 491] or executing structured commands like `ADD`, `UPDATE`, and `DELETE` on a memory store [417]. The learning process is guided solely by the reward from the final task outcome, forcing memory management to emerge as a goal-aligned, adaptive capability [436, 491, 417].

A key direction lies in extending this RL paradigm to jointly optimize the entire memory cycle, where agents learn not just to store information but to dynamically retrieve and utilize it through sophisticated strategies, such as multi-round reasoning [194] and experience reuse [485, 463]. This goal is becoming increasingly practical due to two key advances. First, emerging frameworks enable policy learning for memory management at low cost and in real time, without requiring expensive LLM fine-tuning [485]. Second, the data efficiency of RL training makes this approach viable even in data-scarce domains [417]. However, despite these promising developments, a fundamental obstacle remains: the long-term credit assignment problem, which involves developing reliable algorithms to attribute a final outcome to a long sequence of intermediate memory decisions [436].

6.3. Instability in Training Algorithms

In DR systems, multiple rounds of interaction with the environment are required. Although RL algorithms such as PPO [276] and GRPO [279] exhibit stable behavior in single-turn scenarios, they often become unstable when extended to multi-turn settings. This instability typically appears as a gradual or abrupt drop in reward, the generation of invalid responses, and symptoms such as entropy collapse and gradient explosion [145, 416, 373, 320]. These issues remain persistent challenges for training agentic RL systems. Below, we examine two newly emerging solutions and outline future directions for further study.

6.3.1. *Existing Solutions*

Filtering void turns. The first representative solution is proposed by Xue et al. [416], who identify *void turns* as a major cause of collapse in multi-turn RL. Void turns refer to responses that do not advance the task, such as fragmented text, repetitive content, or premature termination; and once produced, they propagate through later turns, creating a harmful feedback loop. These errors largely stem from the distribution shift between pre-training and multi-turn inference, where the model must process external tool outputs or intermediate signals that were not present during pre-training, increasing the chance of malformed generations. To address this, SimpleTIR [416] filters out trajectories containing void turns, effectively removing corrupted supervision and stabilizing multi-turn RL training.

Mitigating the Echo Trap. Wang et al. [373] identify the *Echo Trap* as a central cause of collapse in multi-turn RL. The Echo Trap refers to rapid policy homogenization, where the model abandons exploration and repeatedly produces conservative outputs that yield short-term rewards. Once this happens, reward variance and policy entropy drop sharply, forming a self-reinforcing degenerative loop. The root cause is a misalignment between reward-driven optimization and reasoning quality. In multi-turn settings, sparse binary rewards cannot distinguish coincidental success from genuine high-quality reasoning, encouraging reward hacking behaviors such as hallucinated reasoning or skipping essential steps. To address this, the proposed StarPO-S [373] uses uncertainty-based trajectory filtering to retain trajectories exhibiting meaningful exploration. This breaks the Echo Trap cycle and stabilizes multi-turn RL training.

6.3.2. *Future Directions*

Beyond the solutions discussed above, we highlight two additional directions for achieving more stable agentic RL training.

Cold-start methods that preserve exploration. SFT is a practical cold-start strategy for multi-turn RL, yet it introduces a significant drawback: it rapidly reduces output entropy, constraining the model's ability to explore and develop new reasoning strategies [500]. A promising research direction is to design cold-start methods that improve initial task performance while maintaining exploratory behavior. Such techniques should aim to avoid early entropy collapse and preserve the model's capacity for innovation in multi-turn reasoning.

Denser and smoother reward design. Although StarPO-S [373] effectively mitigates training collapse in PPO-based multi-turn RL, its benefit is more limited for GRPO. The critic module inherent to PPO algorithm [276] naturally smooths reward signals, while GRPO relies on group-wise normalization, which makes it more sensitive to reward variance and extreme values. Developing denser, smoother, and more informative reward functions for multi-turn scenarios, especially for GRPO-style algorithms, remains an important direction for future research.

6.4. Evaluation of Deep Research System

Evaluation of DR generally falls into two complementary aspects: (i) the evaluation of agentic information-seeking capabilities and (ii) the evaluation of long-form generation [175, 425]. Considerable progress has been made in the former, with benchmarks such as HotpotQA [425], GAIA [215]. The more recent Deep Research Bench [66, 352] further provides increasingly complex and in-

teractive settings for evaluating agents' abilities to retrieve, navigate, and synthesize information across dynamic web environments. However, reliably evaluating model-generated long-form outputs, especially research-style reports in response to open-ended and high-level queries, remains an open and pressing challenge [379]. Most existing approaches rely on LLM-as-a-Judge to directly evaluate general dimensions such as content factuality, structural coherence, and readability [413]. While effective for scalable comparisons, these evaluation strategies ignore crucial dimensions and are subject to several limitations.

6.4.1. Logical Evaluation

Since DR typically requires long-form context generation, maintaining logical coherence throughout the text is essential [476]. Existing studies suggest that while LLMs demonstrate strong capabilities for recognizing logical patterns, such as in summarization tasks or the detection of inconsistencies in short passages, their ability to create rigorous logical chains during DR remains uncertain [168]. In particular, when required to synthesize insights from multiple retrieved supporting documents, models often fail to consistently transform fragmented evidence into a logically connected narrative. The generated reasoning may contain gaps, abrupt leaps, or even circular justifications, compromising the argument's fidelity [260]. This limitation underscores a key challenge for DR: the task is not merely to produce fluent and factually plausible text, but to articulate insights that are logically well-founded and epistemically defensible [189].

Accordingly, robust logical evaluation emerges as a central challenge. However, most existing research on logical assessment remains narrowly scoped. Current benchmarks typically address limited logical tasks, such as solving symbolic logic puzzles, identifying entailment in short sentences, or handling deductive reasoning in synthetic settings [375, 246]. While these tasks provide valuable insights into basic reasoning abilities, they fall short of capturing the complexities of long-form logical consistency. Specifically, they do not address whether models can sustain coherent argumentative structures across extended contexts, reconcile conflicting sources, or systematically avoid introducing unsupported claims. One potential approach is to design evaluation frameworks that assess coherence across multiple granularities (*e.g.,* sentence-level, paragraph-level, and document-level), capturing both local and global logical dependencies [102].

6.4.2. Boundary between Novelty and Hallucination

In DR, progressing beyond faithful summarization toward the generation of genuinely novel hypotheses or perspectives is a central goal [81]. However, in practice, outputs that appear original may embed unverifiable claims, fabricated connections between sources, or spurious inferences lacking epistemic grounding [184]. This challenge is exacerbated in open-ended settings, where no single ground truth exists and retrieval broadens the hypothesis space, increasing the likelihood that superficially plausible but unsupported statements evade detection, especially by surface-level or style-sensitive evaluation methods [135]. Current practices often depend on density-based novelty scores or LLM-as-judge assessments of originality [365, 459], yet these alone do not ensure verifiability or differentiate between creative recombination and unfounded speculation.

A potential solution is to differentiate between two types of novelty. Generative novelty refers to new combinations or perspectives, while deductive novelty refers to conclusions logically derived from known facts. To achieve this, novelty scoring can be combined with validity-checking mechanisms [11, 440, 267]. For example, researchers can pre-register testable claims along with verification plans, ensure that each claim is linked to clear sources, and systematically ablate sources to determine

which are necessary or sufficient [480, 345]. Additionally, inserting control examples or testing the system with false information can reveal how often it generates incorrect but seemingly original results [217]. Another useful method is to restrict the system to documents published before a certain cutoff date and then examine whether its outputs are later validated by subsequently published sources—providing insight into the independence and robustness of novel ideas [191, 155].

6.4.3. Bias and Efficiency of LLM-as-Judge

LLM-as-Judge has become a mainstream approach for evaluating long-form model outputs. However, this practice introduces two major challenges. **The first challenge is bias.** LLM judges may prefer longer responses, be affected by answer ordering, reward particular writing styles, or favor systems that resemble themselves [100, 169]. Such biases may reduce the robustness and fairness of existing evaluation protocols. **The second challenge is efficiency.** Large-scale pairwise evaluation is resource-intensive, especially when relying on paid APIs and applying costly comparison methods to long outputs [497]. These limitations motivate two directions for improvement: mitigating bias and improving efficiency.

Mitigating bias. Bias can be reduced by incorporating human evaluators for critical or ambiguous cases, providing a grounded reference for calibration [169]. Another direction is to fine-tune judge models using datasets that highlight diverse reasoning styles and explicit debiasing signals [497]. Such training may lessen systematic preferences for particular formats or linguistic patterns.

Improving efficiency. Efficiency can be improved by adopting open-source, general-purpose judge models, which reduce evaluation cost while offering greater transparency and reproducibility [272]. Further improvements may come from smarter candidate selection algorithms that focus on the most informative comparisons [475]. By lowering the number of required pairwise evaluations without sacrificing quality, such methods enable LLM-based evaluation in more resource-constrained settings.

7. Open Discussion: Deep Research to General Intelligence

As DR systems advance, they must navigate key challenges that bridge specialized task-solving with broader cognitive capabilities. This subsection examines three pivotal areas (*i.e.,* creativity, fairness, safety, and reliability) that may shape the development from current DR paradigms to AGI-level autonomy, ensuring these systems not only augment human inquiry but also foster equitable, innovative, and trustworthy ecosystems.

7.1. Creativity

Despite the considerable attention and rapid development in both academia and industry, Previous studies highlight fundamental limitations in LLM creativity based on next-token prediction [198]. While they excel at recombination [411, 417], emotion [422, 421], imitation [169, 363], and logical reasoning [348, 302], the question remains whether AI can evolve from these capabilities to achieve genuine innovation and novel concept generation. This transition may require mechanisms beyond statistical learning, drawing on psychological theories of human creativity, such as *insight* or *eureka moments,* which involve sudden restructuring of mental representations and are not easily explained by probabilistic models [389]. Some argue that hallucinations in AI could be interpreted as a form of creativity [198], potentially bridging this gap, but this perspective needs careful examination to distinguish between productive divergence and erroneous output.

7.2. Fairness

As noted in prior work [77], DR powered by autonomous agents may inadvertently inherit and amplify existing biases in academia. For example, they could favor mainstream fields, methodologies, or prominent researchers, thereby overlooking emerging interdisciplinary work or contributions from non-mainstream regions.r To mitigate this, such systems should incorporate built-in fairness frameworks that ensure comprehensive and impartial evaluation of all data, prevent the reinforcement of academic hierarchies, and provide equitable support to researchers from diverse backgrounds. A critical consideration is the impact of each agent's decision step on the overall fairness of outcomes: how much does bias in early steps shape subsequent decision spaces during interactions with the environment? Recent work [158] indicates that this cascading effect could limit exploration and perpetuate inequities if not addressed through debiasing techniques at every stage.

7.3. Safety and Reliability

Although some studies suggest that AI hallucinations can spark diversity [198, 330], they also pose risks of disseminating serious academic errors. To enhance safety and reliability, Deep research system should ensure conclusions are supported by clear, traceable evidence chains; offer highly transparent reasoning processes to avoid "black-box" decisions; and implement robust validation mechanisms to curb the spread of hallucinated science [33, 296]. These measures are essential for maintaining trust in AI-assisted research and preventing misinformation in scholarly pursuits.

8. Conclusion and Future Outlook

Deep research (DR) stands at the frontier of transforming large language models from passive responders into autonomous investigators capable of iterative reasoning, evidence synthesis, and verifiable knowledge creation. This survey consolidates recent advances in architectures, optimization methods, and evaluation frameworks, providing a unified roadmap for understanding and building future DR systems. By investigating relevant works, this survey facilitates future research and accelerates the advancement of DR systems toward more general, reliable, and interpretable intelligence. Given the rapid evolution of this field, we will continuously update this survey to encompass emerging paradigms such as multimodal reasoning, self-evolving memory, and agentic reinforcement learning. This effort aims to provide a comprehensive and up-to-date understanding of deep research systems.

References

[1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

[2] Kortix AI. Suna: Open-source generalist ai agent, 2025. URL https://github.com/kortix-ai/suna.

[3] Perplexity AI. Perplexity deep research, 2025. URL https://perplexity.ai/hub/blog/introducing-perplexity-deep-research.

[4] Skywork AI. Skywork-deepresearch. https://github.com/SkyworkAI/Skywork-DeepResearch, 2025.

[5] Sina Alemohammad, Josue Casco-Rodriguez, Lorenzo Luzi, Ahmed Imtiaz Humayun, Hossein Babaei, Daniel LeJeune, Ali Siahkoohi, and Richard G Baraniuk. Self-consuming generative models go mad. In *International Conference on Learning Representations*, 2024.

[6] Petr Anokhin, Nikita Semenov, Artyom Sorokin, Dmitry Evseev, Andrey Kravchenko, Mikhail Burtsev, and Evgeny Burnaev. Arigraph: Learning knowledge graph world models with episodic memory for llm agents. *arXiv preprint arXiv:2407.04363*, 2024.

[7] 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. *ArXiv*, abs/2505.08775, 2025.

[8] Akari Asai, Jacqueline He, Rulin Shao, Weijia Shi, Amanpreet Singh, Joseph Chee Chang, Kyle Lo, Luca Soldaini, Sergey Feldman, Mike D'arcy, et al. Openscholar: Synthesizing scientific literature with retrieval-augmented lms. *arXiv preprint arXiv:2411.14199*, 2024.

[9] 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*, 2024.

[10] Amos Azaria and Tom Mitchell. The internal state of an llm knows when it's lying. *arXiv preprint arXiv:2304.13734*, 2023.

[11] Zhangir Azerbayev, Bartosz Piotrowski, Hailey Schoelkopf, Edward W Ayers, Dragomir Radev, and Jeremy Avigad. Proofnet: Autoformalizing and formally proving undergraduate-level mathematics. *arXiv preprint arXiv:2302.12433*, 2023.

[12] 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*, 2025. URL https://openreview.net/forum?id=kQ5s9Yh0WI.

[13] Yejin Bang, Ziwei Ji, Alan Schelten, Anthony Hartshorn, Tara Fowler, Cheng Zhang, Nicola Cancedda, and Pascale Fung. HalluLens: LLM hallucination benchmark. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2025.

[14] Tong Bao, Mir Tafseer Nayeem, Davood Rafiei, and Chengzhi Zhang. Surveygen: Quality-aware scientific survey generation with large language models. *arXiv preprint arXiv:2508.17647*, 2025.

[15] Quentin Bertrand, Avishek Joey Bose, Alexandre Duplessis, Marco Jiralerspong, and Gauthier Gidel. On the stability of iterative retraining of generative models on their own data. *arXiv preprint arXiv:2310.00429*, 2023.

[16] 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, 2022.

[17] Nikos I Bosse, Jon Evans, Robert G Gambee, Daniel Hnyk, Peter Mühlbacher, Lawrence Phillips, Dan Schwarz, Jack Wildman, et al. Deep research bench: Evaluating ai web research agents. *arXiv preprint arXiv:2506.06287*, 2025.

[18] Martin Briesch, Dominik Sobania, and Franz Rothlauf. Large language models suffer from their own output: An analysis of the self-consuming training loop. *arXiv preprint arXiv:2311.16822*, 2023.

[19] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33, 2020.

[20] Cameron B Browne, Edward Powley, Daniel Whitehouse, Simon M Lucas, Peter I Cowling, Philipp Rohlfshagen, Stephen Tavener, Diego Perez, Spyridon Samothrakis, and Simon Colton. A survey of monte carlo tree search methods. *IEEE Transactions on Computational Intelligence and AI in games*, 2012.

[21] Boxi Cao, Mengjie Ren, Hongyu Lin, Xianpei Han, Feng Zhang, Junfeng Zhan, and Le Sun. Structeval: Deepen and broaden large language model assessment via structured evaluation. *arXiv preprint arXiv:2408.03281*, 2024.

[22] Jiaqi Cao, Jiarui Wang, Rubin Wei, Qipeng Guo, Kai Chen, Bowen Zhou, and Zhouhan Lin. Memory decoder: A pretrained, plug-and-play memory for large language models. *arXiv preprint arXiv:2508.09874*, 2025.

[23] Ruisheng Cao, Fangyu Lei, Haoyuan Wu, Jixuan Chen, Yeqiao Fu, Hongcheng Gao, Xinzhuang Xiong, Hanchong Zhang, Wenjing Hu, Yuchen Mao, et al. Spider2-v: How far are multimodal agents from automating data science and engineering workflows? *Advances in Neural Information Processing Systems*, 2024.

[24] Zhiwei Cao, Qian Cao, Yu Lu, Ningxin Peng, Luyang Huang, Shanbo Cheng, and Jinsong Su. Retaining key information under high compression ratios: Query-guided compressor for llms. *arXiv preprint arXiv:2406.02376*, 2024.

[25] Victor Carbune, Hassan Mansoor, Fangyu Liu, Rahul Aralikatte, Gilles Baechler, Jindong Chen, and Abhanshu Sharma. Chart-based reasoning: Transferring capabilities from llms to vlms (chartpali-5b). *arXiv preprint arXiv:2403.12596*, 2024.

[26] Jun Shern Chan, Neil Chowdhury, Oliver Jaffe, James Aung, Dane Sherburn, Evan Mays, Giulio Starace, Kevin Liu, Leon Maksin, Tejal Patwardhan, et al. Mle-bench: Evaluating machine learning agents on machine learning engineering. *arXiv preprint arXiv:2410.07095*, 2024.

[27] Prahaladh Chandrahasan, Jiahe Jin, Zhihan Zhang, Tevin Wang, Andy Tang, Lucy Mo, Morteza Ziyadi, Leonardo FR Ribeiro, Zimeng Qiu, Markus Dreyer, et al. Deep research comparator: A platform for fine-grained human annotations of deep research agents. *arXiv preprint arXiv:2507.05495*, 2025.

[28] Chao Chen, Kai Liu, Ze Chen, Yi Gu, Yue Wu, Mingyuan Tao, Zhihang Fu, and Jieping Ye. Inside: Llms' internal states retain the power of hallucination detection. *arXiv preprint arXiv:2402.03744*, 2024.

[29] Justin Chih-Yao Chen, Swarnadeep Saha, Elias Stengel-Eskin, and Mohit Bansal. Magdi: Structured distillation of multi-agent interaction graphs improves reasoning in smaller language models. *arXiv preprint arXiv:2402.01620*, 2024.

[30] Liyi Chen, Panrong Tong, Zhongming Jin, Ying Sun, Jieping Ye, and Hui Xiong. Plan-on-graph: Self-correcting adaptive planning of large language model on knowledge graphs. *Advances in Neural Information Processing Systems*, 37:37665–37691, 2024.

[31] Qiguang Chen, Mingda Yang, Libo Qin, Jinhao Liu, Zheng Yan, Jiannan Guan, Dengyun Peng, Yiyan Ji, Hanjing Li, Mengkang Hu, et al. Ai4research: A survey of artificial intelligence for scientific research. *arXiv preprint arXiv:2507.01903*, 2025.

[32] Si-An Chen, Lesly Miculicich, Julian Eisenschlos, Zifeng Wang, Zilong Wang, Yanfei Chen, Yasuhisa Fujii, Hsuan-Tien Lin, Chen-Yu Lee, and Tomas Pfister. Tablerag: Million-token table understanding with language models. *Advances in Neural Information Processing Systems*, 2024.

[33] Xiaoyang Chen, Ben He, Hongyu Lin, Xianpei Han, Tianshu Wang, Boxi Cao, Le Sun, and Yingfei Sun. Spiral of silence: How is large language model killing information retrieval? – a case study on open domain question answering, 2024.

[34] 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. *arXiv preprint arXiv:2510.24695*, 2025.

[35] Yiqun Chen, Qi Liu, Yi Zhang, Weiwei Sun, Xinyu Ma, Wei Yang, Daiting Shi, Jiaxin Mao, and Dawei Yin. Tourrank: Utilizing large language models for documents ranking with a tournament-inspired strategy. In *Proceedings of the ACM on Web Conference 2025*, 2025.

[36] Yiqun Chen, Lingyong Yan, Weiwei Sun, Xinyu Ma, Yi Zhang, Shuaiqiang Wang, Dawei Yin, Yiming Yang, and Jiaxin Mao. Improving retrieval-augmented generation through multi-agent reinforcement learning. *arXiv preprint arXiv:2501.15228*, 2025.

[37] Yiqun Chen, Erhan Zhang, Lingyong Yan, Shuaiqiang Wang, Jizhou Huang, Dawei Yin, and Jiaxin Mao. Mao-arag: Multi-agent orchestration for adaptive retrieval-augmented generation. *arXiv preprint arXiv:2508.01005*, 2025.

[38] Zijian Chen, Xueguang Ma, Shengyao Zhuang, Ping Nie, Kai Zou, Andrew Liu, Joshua Green, Kshama Patel, Ruoxi Meng, Mingyi Su, et al. Browsecomp-plus: A more fair and transparent evaluation benchmark of deep-research agent. *arXiv preprint arXiv:2508.06600*, 2025.

[39] Zixiang Chen, Yihe Deng, Huizhuo Yuan, Kaixuan Ji, and Quanquan Gu. Self-play fine-tuning converts weak language models to strong language models. *arXiv preprint arXiv:2401.01335*, 2024.

[40] Qinyuan Cheng, Xiaonan Li, Shimin Li, Qin Zhu, Zhangyue Yin, Yunfan Shao, Linyang Li, Tianxiang Sun, Hang Yan, and Xipeng Qiu. Unified active retrieval for retrieval augmented generation. *arXiv preprint arXiv:2406.12534*, 2024.

[41] Xin Cheng, Xun Wang, Xingxing Zhang, Tao Ge, Si-Qing Chen, Furu Wei, Huishuai Zhang, and Dongyan Zhao. xrag: Extreme context compression for retrieval-augmented generation with one token. *Advances in Neural Information Processing Systems*, 2024.

[42] Zhi-Qi Cheng, Qi Dai, Siyao Li, Jingdong Sun, Teruko Mitamura, and Alexander G Hauptmann. Chartreader: A unified framework for chart derendering and comprehension without heuristic rules. *arXiv preprint arXiv:2304.02173*, 2023.

[43] I Chern, Steffi Chern, Shiqi Chen, Weizhe Yuan, Kehua Feng, Chunting Zhou, Junxian He, Graham Neubig, Pengfei Liu, et al. Factool: Factuality detection in generative ai–a tool augmented framework for multi-task and multi-domain scenarios. *arXiv preprint arXiv:2307.13528*, 2023.

[44] Maryna Chernyshevich. Core intelligence at semeval-2025 task 8: Multi-hop llm agent for tabular question answering. In *Proceedings of the 19th International Workshop on Semantic Evaluation (SemEval-2025)*, 2025.

[45] Alexis Chevalier, Alexander Wettig, Anirudh Ajith, and Danqi Chen. Adapting language models to compress contexts. *arXiv preprint arXiv:2305.14788*, 2023.

[46] Prateek Chhikara, Dev Khant, Saket Aryan, Taranjeet Singh, and Deshraj Yadav. Mem0: Building production-ready ai agents with scalable long-term memory. *arXiv preprint arXiv:2504.19413*, 2025.

[47] Philipp Christmann and Gerhard Weikum. Recursive question understanding for complex question answering over heterogeneous personal data. *arXiv preprint arXiv:2505.11900*, 2025.

[48] Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. Scaling instruction-finetuned language models. *Journal of Machine Learning Research*, 25(70):1–53, 2024.

[49] João Coelho, Jingjie Ning, Jingyuan He, Kangrui Mao, Abhijay Paladugu, Pranav Setlur, Jiahe Jin, Jamie Callan, João Magalhães, Bruno Martins, et al. Deepresearchgym: A free, transparent, and reproducible evaluation sandbox for deep research. *arXiv preprint arXiv:2505.19253*, 2025.

[50] Gordon V. Cormack, Charles L. A. Clarke, and Stefan Buettcher. Reciprocal rank fusion outperforms condorcet and individual rank learning methods. In *SIGIR*, 2009.

[51] Thomas H Cormen, Charles E Leiserson, Ronald L Rivest, and Clifford Stein. *Introduction to algorithms*. MIT press, 2022.

[52] Manuel Cossio. A comprehensive taxonomy of hallucinations in large language models. *arXiv preprint arXiv:2508.01781*, 2025.

[53] Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, 2023.

[54] Mayur Datar, Nicole Immorlica, Piotr Indyk, and Vahab S Mirrokni. Locality-sensitive hashing scheme based on p-stable distributions. In *Proceedings of the twentieth annual symposium on Computational geometry*, 2004.

[55] Nicola De Cao, Wilker Aziz, and Ivan Titov. Editing factual knowledge in language models. *arXiv preprint arXiv:2104.08164*, 2021.

[56] Boyi Deng, Wenjie Wang, Fengbin Zhu, Qifan Wang, and Fuli Feng. Cram: Credibility-aware attention modification in llms for combating misinformation in rag. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 23760–23768, 2025.

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

[58] Shrey Desai and Greg Durrett. Calibration of pre-trained transformers. *arXiv preprint arXiv:2003.07892*, 2020.

[59] Shehzaad Dhuliawala, Mojtaba Komeili, Jing Xu, Roberta Raileanu, Xian Li, Asli Celikyilmaz, and Jason Weston. Chain-of-verification reduces hallucination in large language models. *arXiv preprint arXiv:2309.11495*, 2023.

[60] Hanxing Ding, Liang Pang, Zihao Wei, Huawei Shen, and Xueqi Cheng. Retrieve only when it needs: Adaptive retrieval augmentation for hallucination mitigation in large language models. *arXiv preprint arXiv:2402.10612*, 2024.

[61] Elvis Dohmatob, Yunzhen Feng, Pu Yang, Francois Charton, and Julia Kempe. A tale of tails: Model collapse as a change of scaling laws. *arXiv preprint arXiv:2402.07043*, 2024.

[62] Guanting Dong, Licheng Bao, Zhongyuan Wang, Kangzhi Zhao, Xiaoxi Li, Jiajie Jin, Jinghan Yang, Hangyu Mao, Fuzheng Zhang, Kun Gai, et al. Agentic entropy-balanced policy optimization. *arXiv preprint arXiv:2510.14545*, 2025.

[63] 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. *arXiv preprint arXiv:2505.16410*, 2025.

[64] Guanting Dong, Hangyu Mao, Kai Ma, Licheng Bao, Yifei Chen, Zhongyuan Wang, Zhongxia Chen, Jiazhen Du, Huiyang Wang, Fuzheng Zhang, et al. Agentic reinforced policy optimization. *arXiv preprint arXiv:2507.19849*, 2025.

[65] Guanting Dong, Yutao Zhu, Chenghao Zhang, Zechen Wang, Ji-Rong Wen, and Zhicheng Dou. Understand what llm needs: Dual preference alignment for retrieval-augmented generation. In *Proceedings of the ACM on Web Conference 2025*, pages 4206–4225, 2025.

[66] Mingxuan Du, Benfeng Xu, Chiwei Zhu, Xiaorui Wang, and Zhendong Mao. Deepresearch bench: A comprehensive benchmark for deep research agents. *arXiv preprint arXiv:2506.11763*, 2025.

[67] Yiming Du, Wenyu Huang, Danna Zheng, Zhaowei Wang, Sebastien Montella, Mirella Lapata, Kam-Fai Wong, and Jeff Z Pan. Rethinking memory in ai: Taxonomy, operations, topics, and future directions. *arXiv preprint arXiv:2505.00675*, 2025.

[68] Yiming Du, Bingbing Wang, Yang He, Bin Liang, Baojun Wang, Zhongyang Li, Lin Gui, Jeff Z Pan, Ruifeng Xu, and Kam-Fai Wong. Bridging the long-term gap: A memory-active policy for multi-session task-oriented dialogue. *arXiv preprint arXiv:2505.20231*, 2025.

[69] Jinhao Duan, Hao Cheng, Shiqi Wang, Alex Zavalny, Chenan Wang, Renjing Xu, Bhavya Kailkhura, and Kaidi Xu. Shifting attention to relevance: Towards the predictive uncertainty quantification of free-form large language models. *arXiv preprint arXiv:2307.01379*, 2023.

[70] Angela Fan, Yacine Jernite, Ethan Perez, David Grangier, Jason Weston, and Michael Auli. Eli5: Long form question answering. *arXiv preprint arXiv:1907.09190*, 2019.

[71] Run-Ze Fan, Zengzhi Wang, and Pengfei Liu. Megascience: Pushing the frontiers of post-training datasets for science reasoning. *arXiv preprint arXiv:2507.16812*, 2025.

[72] Wenqi Fan, Yujuan Ding, Liang bo Ning, Shijie Wang, Hengyun Li, Dawei Yin, Tat-Seng Chua, and Qing Li. A survey on rag meeting llms: Towards retrieval-augmented large language models. *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2024.

[73] Feiteng Fang, Yuelin Bai, Shiwen Ni, Min Yang, Xiaojun Chen, and Ruifeng Xu. Enhancing noise robustness of retrieval-augmented language models with adaptive adversarial training. *arXiv preprint arXiv:2405.20978*, 2024.

[74] Jinyuan Fang, Yanwen Peng, Xi Zhang, Yingxu Wang, Xinhao Yi, Guibin Zhang, Yi Xu, Bin Wu, Siwei Liu, Zihao Li, et al. A comprehensive survey of self-evolving ai agents: A new paradigm bridging foundation models and lifelong agentic systems. *arXiv preprint arXiv:2508.07407*, 2025.

[75] Runnan Fang, Shihao Cai, Baixuan Li, Jialong Wu, Guangyu Li, Wenbiao Yin, Xinyu Wang, Xiaobin Wang, Liangcai Su, Zhen Zhang, et al. Towards general agentic intelligence via environment scaling. *arXiv preprint arXiv:2509.13311*, 2025.

[76] Alhussein Fawzi, Matej Balog, Aja Huang, Thomas Hubert, Bernardino Romera-Paredes, Mohammadamin Barekatain, Alexander Novikov, Francisco J R. Ruiz, Julian Schrittwieser, Grzegorz Swirszcz, et al. Discovering faster matrix multiplication algorithms with reinforcement learning. *Nature*, 610(7930):47–53, 2022.

[77] Emilio Ferrara. Fairness and bias in artificial intelligence: A brief survey of sources, impacts, and mitigation strategies. *Sci*, 2024.

[78] Marina Fomicheva, Shuo Sun, Lisa Yankovskaya, Frédéric Blain, Francisco Guzmán, Mark Fishel, Nikolaos Aletras, Vishrav Chaudhary, and Lucia Specia. Unsupervised quality estimation for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8: 539–555, 2020.

[79] Thibault Formal, Carlos Lassance, Benjamin Piwowarski, and Stéphane Clinchant. Splade v2: Sparse lexical and expansion model for information retrieval. *arXiv preprint arXiv:2109.10086*, 2021.

[80] Thibault Formal, Benjamin Piwowarski, and Stéphane Clinchant. Splade: Sparse lexical and expansion model for first stage ranking. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2288–2292, 2021.

[81] Giorgio Franceschelli and Mirco Musolesi. On the creativity of large language models. *AI & society*, 40(5), 2025.

[82] Tsu-Jui Fu, William Yang Wang, Daniel McDuff, and Yale Song. Doc2ppt: Automatic presentation slides generation from scientific documents. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 634–642, 2022.

[83] Chen Gao, Xiaochong Lan, Nian Li, Yuan Yuan, Jingtao Ding, Zhilun Zhou, Fengli Xu, and Yong Li. Large language models empowered agent-based modeling and simulation: A survey and perspectives. *Humanities and Social Sciences Communications*, 2024.

[84] Huan-ang Gao, Jiayi Geng, Wenyue Hua, Mengkang Hu, Xinzhe Juan, Hongzhang Liu, Shilong Liu, Jiahao Qiu, Xuan Qi, Yiran Wu, et al. A survey of self-evolving agents: On path to artificial super intelligence. *arXiv preprint arXiv:2507.21046*, 2025.

[85] 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. *arXiv preprint arXiv:2508.07976*, 2025.

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

[87] Xian Gao, Zongyun Zhang, Mingye Xie, Ting Liu, and Yuzhuo Fu. Graph of ai ideas: Leveraging knowledge graphs and llms for ai research idea generation. *arXiv preprint arXiv:2503.08549*, 2025.

[88] Yifei Gao, Junhong Ye, Jiaqi Wang, and Jitao Sang. Websynthesis: World-model-guided mcts for efficient webui-trajectory synthesis. *arXiv preprint arXiv:2507.04370*, 2025.

[89] Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Qianyu Guo, Meng Wang, and Haofen Wang. Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997*, 2023.

[90] Zhaolin Gao, Kianté Brantley, and Thorsten Joachims. Reviewer2: Optimizing review generation through prompt generation. *arXiv preprint arXiv:2402.10886*, 2024.

[91] Jiaxin Ge, Zora Zhiruo Wang, Xuhui Zhou, Yi-Hao Peng, Sanjay Subramanian, Qinyue Tan, Maarten Sap, Alane Suhr, Daniel Fried, Graham Neubig, et al. Autopresent: Designing structured visuals from scratch. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025.

[92] Tao Ge, Jing Hu, Lei Wang, Xun Wang, Si-Qing Chen, and Furu Wei. In-context autoencoder for context compression in a large language model. *arXiv preprint arXiv:2307.06945*, 2023.

[93] Tao Ge, Xin Chan, Xiaoyang Wang, Dian Yu, Haitao Mi, and Dong Yu. Scaling synthetic data creation with 1,000,000,000 personas. *arXiv preprint arXiv:2406.20094*, 2024.

[94] Ziyu Ge, Yuhao Wu, Daniel Wai Kit Chin, Roy Ka-Wei Lee, and Rui Cao. Resolving conflicting evidence in automated fact-checking: A study on retrieval-augmented llms, 2025.

[95] Carlos G'omez-Rodr'iguez and Paul Williams. A confederacy of models: a comprehensive evaluation of llms on creative writing. *arXiv preprint arXiv:2310.08433*, 2023.

[96] Peiyuan Gong, Feiran Zhu, Yaqi Yin, Chenglei Dai, Chao Zhang, Kai Zheng, Wentian Bao, Jiaxin Mao, and Yi Zhang. Cardrewriter: Leveraging knowledge cards for long-tail query rewriting on short-video platforms. *arXiv preprint arXiv:2510.10095*, 2025.

[97] Google. Deep research is now available on gemini 2.5 pro experimental. https://blog.google/products/gemini/deep-research-gemini-2-5-pro-experimental/, February 2025.

[98] Boyu Gou, Zanming Huang, Yuting Ning, Yu Gu, Michael Lin, Weijian Qi, Andrei Kopanev, Botao Yu, Bernal Jiménez Gutiérrez, Yiheng Shu, et al. Mind2web 2: Evaluating agentic search with agent-as-a-judge. *arXiv preprint arXiv:2506.21506*, 2025.

[99] Hongchao Gu, Dexun Li, Kuicai Dong, Hao Zhang, Hang Lv, Hao Wang, Defu Lian, Yong Liu, and Enhong Chen. RAPID: Efficient retrieval-augmented long text generation with writing planning and information discovery. In *Findings of the Association for Computational Linguistics: ACL 2025*, 2025.

[100] Jiawei Gu, Xuhui Jiang, Zhichao Shi, Hexiang Tan, Xuehao Zhai, Chengjin Xu, Wei Li, Yinghan Shen, Shengjie Ma, Honghao Liu, et al. A survey on llm-as-a-judge. *arXiv preprint arXiv:2411.15594*, 2024.

[101] Tianle Gu, Kexin Huang, Ruilin Luo, Yuanqi Yao, Yujiu Yang, Yan Teng, and Yingchun Wang. Meow: Memory supervised llm unlearning via inverted facts. *arXiv preprint arXiv:2409.11844*, 2024.

[102] Jian Guan, Xiaoxi Mao, Changjie Fan, Zitao Liu, Wenbiao Ding, and Minlie Huang. Long text generation by modeling sentence-level and discourse-level coherence. *arXiv preprint arXiv:2105.08963*, 2021.

[103] Xinyan Guan, Jiali Zeng, Fandong Meng, Chunlei Xin, Yaojie Lu, Hongyu Lin, Xianpei Han, Le Sun, and Jie Zhou. Deeprag: Thinking to retrieve step by step for large language models. *arXiv preprint arXiv:2502.01142*, 2025.

[104] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. In *International conference on machine learning*, pages 1321–1330. PMLR, 2017.

[105] Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

[106] Minghao Guo, Qingcheng Zeng, Xujiang Zhao, Yanchi Liu, Wenchao Yu, Mengnan Du, Haifeng Chen, and Wei Cheng. Deepsieve: Information sieving via llm-as-a-knowledge-router. *arXiv preprint arXiv:2507.22050*, 2025.

[107] Shuyu Guo and Zhaochun Ren. Dynamic context compression for efficient rag. *arXiv preprint arXiv:2507.22931*, 2025.

[108] H2O.ai. H2o.ai deep research, 2025. URL https://h2o.ai/.

[109] Muhammad Usman Hadi, Rizwan Qureshi, Abbas Shah, Muhammad Irfan, Anas Zafar, Muhammad Bilal Shaikh, Naveed Akhtar, Jia Wu, Seyedali Mirjalili, et al. A survey on large language models: Applications, challenges, limitations, and practical usage. *Authorea Preprints*, 2023.

[110] Chuzhan Hao, Wenfeng Feng, Yuewei Zhang, and Hao Wang. Dynasearcher: Dynamic knowledge graph augmented search agent via multi-reward reinforcement learning. *arXiv preprint arXiv:2507.17365*, 2025.

[111] Jiashu He, Jinxuan Fan, Bowen Jiang, Ignacio Houine, Dan Roth, and Alejandro Ribeiro. Self-give: Associative thinking from limited structured knowledge for enhanced large language model reasoning. *arXiv preprint arXiv:2505.15062*, 2025.

[112] Zhiwei He, Tian Liang, Jiahao Xu, Qiuzhi Liu, Xingyu Chen, Yue Wang, Linfeng Song, Dian Yu, Zhenwen Liang, Wenxuan Wang, Zhuosheng Zhang, Rui Wang, Zhaopeng Tu, Haitao Mi, and Dong Yu. Deepmath-103k: A large-scale, challenging, decontaminated, and verifiable mathematical dataset for advancing reasoning. *arXiv preprint arXiv:2504.11456*, 2025.

[113] Zihong He, Weizhe Lin, Hao Zheng, Fan Zhang, Matt W Jones, Laurence Aitchison, Xuhai Xu, Miao Liu, Per Ola Kristensson, and Junxiao Shen. Human-inspired perspectives: A survey on ai long-term memory. *arXiv preprint arXiv:2411.00489*, 2024.

[114] David Herel and Tomas Mikolov. Collapse of self-trained language models. *arXiv preprint arXiv:2404.02305*, 2024.

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

[116] Sirui Hong, Mingchen Zhuge, Jonathan Chen, Xiawu Zheng, Yuheng Cheng, Jinlin Wang, Ceyao Zhang, Zili Wang, Steven Ka Shing Yau, Zijuan Lin, et al. Metagpt: Meta programming for a multi-agent collaborative framework. In *The Twelfth International Conference on Learning Representations*, 2023.

[117] Niklas Höpner, Leon Eshuijs, Dimitrios Alivanistos, Giacomo Zamprogno, and Ilaria Tiddi. Automatic evaluation metrics for artificially generated scientific research. *arXiv preprint arXiv:2503.05712*, 2025.

[118] Xinying Hou, Yanjie Zhao, Yue Liu, Zhou Yang, Kailong Wang, Li Li, Xiapu Luo, David Lo, John C. Grundy, and Haoyu Wang. Large language models for software engineering: A systematic literature review. *ACM Transactions on Software Engineering and Methodology*, 2023.

[119] Chenxu Hu, Jie Fu, Chenzhuang Du, Simian Luo, Junbo Zhao, and Hang Zhao. Chatdb: Augmenting llms with databases as their symbolic memory. *arXiv preprint arXiv:2306.03901*, 2023.

[120] Mengkang Hu, Yuhang Zhou, Wendong Fan, Yuzhou Nie, Bowei Xia, Tao Sun, Ziyu Ye, Zhaoxuan Jin, Yingru Li, Qiguang Chen, Zeyu Zhang, Yifeng Wang, Qianshuo Ye, Bernard Ghanem, Ping Luo, and Guohao Li. Owl: Optimized workforce learning for general multi-agent assistance in real-world task automation, 2025. Accessed: 2025-11-13.

[121] Minda Hu, Tianqing Fang, Jianshu Zhang, Junyu Ma, Zhisong Zhang, Jingyan Zhou, Hongming Zhang, Haitao Mi, Dong Yu, and Irwin King. Webcot: Enhancing web agent reasoning by reconstructing chain-of-thought in reflection, branching, and rollback. *arXiv preprint arXiv:2505.20013*, 2025.

[122] Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and Ting Liu. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Transactions on Information Systems*, 2023.

[123] Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, et al. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Transactions on Information Systems*, 43(2):1–55, 2025.

[124] Qian Huang, Jian Vora, Percy Liang, and Jure Leskovec. Mlagentbench: Evaluating language agents on machine learning experimentation. *arXiv preprint arXiv:2310.03302*, 2023.

[125] Yuxuan Huang, Yihang Chen, Haozheng Zhang, Kang Li, Huichi Zhou, Meng Fang, Linyi Yang, Xiaoguang Li, Lifeng Shang, Songcen Xu, et al. Deep research agents: A systematic examination and roadmap. *arXiv preprint arXiv:2506.18096*, 2025.

[126] Liu Huanshuo, Hao Zhang, Zhijiang Guo, Jing Wang, Kuicai Dong, Xiangyang Li, Yi Quan Lee, Cong Zhang, and Yong Liu. CtrlA: Adaptive retrieval-augmented generation via inherent control. In *Findings of the Association for Computational Linguistics: ACL 2025*, 2025.

[127] Gautier Izacard and Edouard Grave. Leveraging passage retrieval with generative models for open domain question answering. *arXiv preprint arXiv:2007.01282*, 2020.

[128] Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. Unsupervised dense information retrieval with contrastive learning. *arXiv preprint arXiv:2112.09118*, 2021.

[129] Peter Jansen, Marc-Alexandre Côté, Tushar Khot, Erin Bransom, Bhavana Dalvi Mishra, Bodhisattwa Prasad Majumder, Oyvind Tafjord, and Peter Clark. Discoveryworld: A virtual environment for developing and evaluating automated scientific discovery agents. *Advances in Neural Information Processing Systems*, 2024.

[130] Abhinav Java, Ashmit Khandelwal, Sukruta Midigeshi, Aaron Halfaker, Amit Deshpande, Navin Goyal, Ankur Gupta, Nagarajan Natarajan, and Amit Sharma. Characterizing deep research: A benchmark and formal definition. *arXiv preprint arXiv:2508.04183*, 2025.

[131] Soyeong Jeong, Jinheon Baek, Sukmin Cho, Sung Ju Hwang, and Jong C Park. Adaptive-rag: Learning to adapt retrieval-augmented large language models through question complexity. *arXiv preprint arXiv:2403.14403*, 2024.

[132] Jinhao Jiang, Jiayi Chen, Junyi Li, Ruiyang Ren, Shijie Wang, Wayne Xin Zhao, Yang Song, and Tao Zhang. Rag-star: Enhancing deliberative reasoning with retrieval augmented verification and refinement. *arXiv preprint arXiv:2412.12881*, 2024.

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

[134] Pengcheng Jiang, Xueqiang Xu, Jiacheng Lin, Jinfeng Xiao, Zifeng Wang, Jimeng Sun, and Jiawei Han. s3: You don't need that much data to train a search agent via rl. *arXiv preprint arXiv:2505.14146*, 2025.

[135] Xuhui Jiang, Yuxing Tian, Fengrui Hua, Chengjin Xu, Yuanzhuo Wang, and Jian Guo. A survey on large language model hallucination via a creativity perspective. *arXiv preprint arXiv:2402.06647*, 2024.

[136] Xun Jiang, Feng Li, Han Zhao, Jiahao Qiu, Jiaying Wang, Jun Shao, Shihao Xu, Shu Zhang, Weiling Chen, Xavier Tang, et al. Long term memory: The foundation of ai self-evolution. *arXiv preprint arXiv:2410.15665*, 2024.

[137] Zhengbao Jiang, Jun Araki, Haibo Ding, and Graham Neubig. How can we know when language models know? on the calibration of language models for question answering. *Transactions of the Association for Computational Linguistics*, 2021.

[138] Zhengbao Jiang, Frank F Xu, Luyu Gao, Zhiqing Sun, Qian Liu, Jane Dwivedi-Yu, Yiming Yang, Jamie Callan, and Graham Neubig. Active retrieval augmented generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 7969–7992, 2023.

[139] Zhouyu Jiang, Mengshu Sun, Lei Liang, and Zhiqiang Zhang. Retrieve, summarize, plan: Advancing multi-hop question answering with an iterative approach. In *Companion Proceedings of the ACM on Web Conference 2025*, 2025.

[140] Carlos E. Jimenez, John Yang, Alexander Wettig, Shunyu Yao, Kexin Pei, Ofir Press, and Karthik Narasimhan. Swe-bench: Can language models resolve real-world github issues? *arXiv preprint arXiv:2310.06770*, 2023.

[141] Carlos E Jimenez, John Yang, Alexander Wettig, Shunyu Yao, Kexin Pei, Ofir Press, and Karthik Narasimhan. Swe-bench: Can language models resolve real-world github issues? *arXiv preprint arXiv:2310.06770*, 2023.

[142] Bernal Jimenez Gutierrez, Yiheng Shu, Yu Gu, Michihiro Yasunaga, and Yu Su. Hipporag: Neurobiologically inspired long-term memory for large language models. *Advances in Neural Information Processing Systems*, 2024.

[143] Bowen Jin, Jinsung Yoon, Jiawei Han, and Sercan O Arik. Long-context llms meet rag: Overcoming challenges for long inputs in rag. In *The Thirteenth International Conference on Learning Representations*, 2024.

[144] Bowen Jin, Jinsung Yoon, Priyanka Kargupta, Sercan O Arik, and Jiawei Han. An empirical study on reinforcement learning for reasoning-search interleaved llm agents. *arXiv preprint arXiv:2505.15117*, 2025.

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

[146] Jiajie Jin, Yutao Zhu, Yujia Zhou, and Zhicheng Dou. Bider: Bridging knowledge inconsistency for efficient retrieval-augmented llms via key supporting evidence. *arXiv preprint arXiv:2402.12174*, 2024.

[147] Xu Jin, Guo Zhifang, He Jinzheng, Hu Hangrui, He Ting, Bai Shuai, Chen Keqin, Wang Jialin, Fan Yang, Dang Kai, Zhang Bin, Wang Xiong, Chu Yunfei, and Lin Junyang. Qwen2.5-omni technical report. *arXiv preprint arXiv:2503.20215v1*, 2025.

[148] Liqiang Jing, Zhehui Huang, Xiaoyang Wang, Wenlin Yao, Wenhao Yu, Kaixin Ma, Hongming Zhang, Xinya Du, and Dong Yu. Dsbench: How far are data science agents from becoming data science experts? *arXiv preprint arXiv:2409.07703*, 2024.

[149] Shi Jingwei, Zhang Zeyu, Wu Biao, Liang Yanjie, Fang Meng, Chen Ling, and Zhao Yang. Presentagent: Multimodal agent for presentation video generation. *arXiv preprint arXiv:2507.04036*, 2025.

[150] Jeff Johnson, Matthijs Douze, and Hervé Jégou. Billion-scale similarity search with gpus. *IEEE Transactions on Big Data*, 7(3), 2019.

[151] Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, 2017.

[152] Kyudan Jung, Hojun Cho, Jooyeol Yun, Soyoung Yang, Jaehyeok Jang, and Jaegul Choo. Talk to your slides: Language-driven agents for efficient slide editing. *arXiv preprint arXiv:2505.11604*, 2025.

[153] Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, et al. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*, 2022.

[154] Chia Hsiang Kao, Wenting Zhao, Shreelekha Revankar, Samuel Speas, Snehal Bhagat, Rajeev Datta, Cheng Perng Phoo, Utkarsh Mall, Carl Vondrick, Kavita Bala, et al. Towards llm agents for earth observation. *arXiv preprint arXiv:2504.12110*, 2025.

[155] Ezra Karger, Houtan Bastani, Chen Yueh-Han, Zachary Jacobs, Danny Halawi, Fred Zhang, and Philip E Tetlock. Forecastbench: A dynamic benchmark of ai forecasting capabilities. *arXiv preprint arXiv:2409.19839*, 2024.

[156] 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 *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, 2020.

[157] Omar Khattab and Matei Zaharia. Colbert: Efficient and effective passage search via late interaction over BERT. In *SIGIR*, 2020.

[158] Tahsin Alamgir Kheya. The pursuit of fairness in artificial intelligence models. *arXiv preprint arXiv:2403.17333*, 2024.

[159] Geewook Kim and et al. Donut: Document understanding transformer without ocr. In *ECCV*, 2022.

[160] Yanghoon Kim, Hwanhee Lee, Joongbo Shin, and Kyomin Jung. Improving neural question generation using answer separation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 6602–6609, 2019.

[161] Levente Kocsis and Csaba Szepesvári. Bandit based monte-carlo planning. In *European conference on machine learning*. Springer, 2006.

[162] Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 2022.

[163] Satyapriya Krishna, Kalpesh Krishna, Anhad Mohananey, Steven Schwarcz, Adam Stambler, Shyam Upadhyay, and Manaal Faruqui. Fact, fetch, and reason: A unified evaluation of retrieval-augmented generation. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2025.

[164] Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. *arXiv preprint arXiv:2302.09664*, 2023.

[165] Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, et al. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 2019.

[166] Cheryl Lee, Chunqiu Steven Xia, Longji Yang, Jen-tse Huang, Zhouruixin Zhu, Lingming Zhang, and Michael R Lyu. Unidebugger: Hierarchical multi-agent framework for unified software debugging. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, 2025.

[167] 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 *Proceedings of the 34th International Conference on Neural Information Processing Systems*, 2020.

[168] Anguo Li and Lei Yu. Summary factual inconsistency detection based on llms enhanced by universal information extraction. In *Findings of the Association for Computational Linguistics: ACL 2025*, 2025.

[169] Haitao Li, Qian Dong, Junjie Chen, Huixue Su, Yujia Zhou, Qingyao Ai, Ziyi Ye, and Yiqun Liu. Llms-as-judges: a comprehensive survey on llm-based evaluation methods. *arXiv preprint arXiv:2412.05579*, 2024.

[170] Junlong Li, Shichao Sun, Weizhe Yuan, Run-Ze Fan, Hai Zhao, and Pengfei Liu. Generative judge for evaluating alignment. *arXiv preprint arXiv:2310.05470*, 2023.

[171] Junnan Li and et al. Blip: Bootstrapping language-image pre-training. In *International Conference on Machine Learning*, 2022.

[172] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: bootstrapping language-image pre-training with frozen image encoders and large language models. In *International Conference on Machine Learning*, 2023.

[173] Kuan Li, Zhongwang Zhang, Huifeng Yin, Rui Ye, Yida Zhao, Liwen Zhang, Litu Ou, Dingchu Zhang, Xixi Wu, Jialong Wu, et al. Websailor-v2: Bridging the chasm to proprietary agents via synthetic data and scalable reinforcement learning. *arXiv preprint arXiv:2509.13305*, 2025.

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

[175] Minghao Li, Ying Zeng, Zhihao Cheng, Cong Ma, and Kai Jia. Reportbench: Evaluating deep research agents via academic survey tasks, 2025.

[176] Moxin Li, Yong Zhao, Wenxuan Zhang, Shuaiyi Li, Wenya Xie, See Kiong Ng, Tat-Seng Chua, and Yang Deng. Knowledge boundary of large language models: A survey. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5131–5157, 2025.

[177] Ruochen Li, Liqiang Jing, Chi Han, Jiawei Zhou, and Xinya Du. Learning to generate research idea with dynamic control. *arXiv preprint arXiv:2412.14626*, 2024.

[178] Shutao Li, Bin Li, Bin Sun, and Yixuan Weng. Towards visual-prompt temporal answer grounding in instructional video. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2024.

[179] Sijie Li, Weiwei Sun, Shanda Li, Ameet Talwalkar, and Yiming Yang. Towards community-driven agents for machine learning engineering. *arXiv preprint arXiv: 2506.20640*, 2025.

[180] Weizhen Li, Jianbo Lin, Zhuosong Jiang, Jingyi Cao, Xinpeng Liu, Jiayu Zhang, Zhenqiang Huang, Qianben Chen, Weichen Sun, Qiexiang Wang, et al. Chain-of-agents: End-to-end agent foundation models via multi-agent distillation and agentic rl. *arXiv preprint arXiv:2508.13167*, 2025.

[181] Xiaonan Li, Changtai Zhu, Linyang Li, Zhangyue Yin, Tianxiang Sun, and Xipeng Qiu. Llatrieval: Llm-verified retrieval for verifiable generation. *arXiv preprint arXiv:2311.07838*, 2023.

[182] 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. *arXiv preprint arXiv:2501.05366*, 2025.

[183] Xiaoxi Li, Jiajie Jin, Guanting Dong, Hongjin Qian, Yutao Zhu, Yongkang Wu, Ji-Rong Wen, and Zhicheng Dou. Webthinker: Empowering large reasoning models with deep research capability. *arXiv preprint arXiv:2504.21776*, 2025.

[184] Ethan Lin, Zhiyuan Peng, and Yi Fang. Evaluating and enhancing large language models for novelty assessment in scholarly publications. In *Proceedings of the 1st Workshop on AI and Scientific Discovery: Directions and Opportunities*, 2025.

[185] Stephanie Lin, Jacob Hilton, and Owain Evans. Teaching models to express their uncertainty in words. *Transactions on Machine Learning Research*, 2022.

[186] Junwei Liu, Kaixin Wang, Yixuan Chen, Xin Peng, Zhenpeng Chen, Lingming Zhang, and Yiling Lou. Large language model-based agents for software engineering: A survey. *arXiv preprint arXiv:2409.02977*, 2024.

[187] Lei Liu, Xiaoyan Yang, Yue Shen, Binbin Hu, Zhiqiang Zhang, Jinjie Gu, and Guannan Zhang. Think-in-memory: Recalling and post-thinking enable llms with long-term memory. *arXiv preprint arXiv:2311.08719*, 2023.

[188] Wenhan Liu, Xinyu Ma, Weiwei Sun, Yutao Zhu, Yuchen Li, Dawei Yin, and Zhicheng Dou. Reasonrank: Empowering passage ranking with strong reasoning ability. *arXiv preprint arXiv:2508.07050*, 2025.

[189] Xiang Liu, Peijie Dong, Xuming Hu, and Xiaowen Chu. Longgenbench: Long-context generation benchmark. *arXiv preprint arXiv:2410.04199*, 2024.

[190] Xiao Liu, Bo Qin, Dongzhu Liang, Guang Dong, Hanyu Lai, Hanchen Zhang, Hanlin Zhao, Iat Long Iong, Jiadai Sun, Jiaqi Wang, et al. Autoglm: Autonomous foundation agents for guis. *arXiv preprint arXiv:2411.00820*, 2024.

[191] Yachuan Liu, Xiaochun Wei, Lin Shi, Xinnuo Li, Bohan Zhang, Paramveer Dhillon, and Qiaozhu Mei. Exante: A benchmark for ex-ante inference in large language models. *arXiv preprint arXiv:2505.19533*, 2025.

[192] Yu Liu, Yanbing Liu, Fangfang Yuan, Cong Cao, Youbang Sun, Kun Peng, WeiZhuo Chen, Jianjun Li, and Zhiyuan Ma. Opera: A reinforcement learning–enhanced orchestrated planner-executor architecture for reasoning-oriented multi-hop retrieval. *arXiv preprint arXiv:2508.16438*, 2025.

[193] Google LLC. Gemini deep research, 2024. URL https://gemini.google/overview/deep-research/.

[194] Lin Long, Yichen He, Wentao Ye, Yiyuan Pan, Yuan Lin, Hang Li, Junbo Zhao, and Wei Li. Seeing, listening, remembering, and reasoning: A multimodal agent with long-term memory. *arXiv preprint arXiv:2508.09736*, 2025.

[195] Manus AI (Butterfly Effect Pte. Ltd.). Manus ai, 2025. URL https://manus.im/.

[196] Chris Lu, Cong Lu, Robert Tjarko Lange, Jakob Foerster, Jeff Clune, and David Ha. The ai scientist: Towards fully automated open-ended scientific discovery. *arXiv preprint arXiv:2408.06292*, 2024.

[197] Junru Lu, Siyu An, Mingbao Lin, Gabriele Pergola, Yulan He, Di Yin, Xing Sun, and Yunsheng Wu. Memochat: Tuning llms to use memos for consistent long-range open-domain conversation. *arXiv preprint arXiv:2308.08239*, 2023.

[198] Li-Chun Lu. A critical analysis of existing creativity evaluations, 2025.

[199] Michal Lukasik, Srinadh Bhojanapalli, Aditya Krishna Menon, and Sanjiv Kumar. Teacher's pet: understanding and mitigating biases in distillation. *arXiv preprint arXiv:2106.10494*, 2021.

[200] Haoran Luo, Haihong E, Guanting Chen, Qika Lin, Yikai Guo, Fangzhi Xu, Zemin Kuang, Meina Song, Xiaobao Wu, Yifan Zhu, and Luu Anh Tuan. Graph-r1: Towards agentic graphrag framework via end-to-end reinforcement learning. *arXiv preprint arXiv:2507.21892*, 2025.

[201] Yi Luo, Linghang Shi, Yihao Li, Aobo Zhuang, Yeyun Gong, Ling Liu, and Chen Lin. From intention to implementation: automating biomedical research via llms. *Science China Information Sciences*, 2025.

[202] Yougang Lyu, Xiaoyu Zhang, Lingyong Yan, Maarten de Rijke, Zhaochun Ren, and Xiuying Chen. Deepshop: A benchmark for deep research shopping agents. *arXiv preprint arXiv:2506.02839*, 2025.

[203] Xinbei Ma, Yeyun Gong, Pengcheng He, Hai Zhao, and Nan Duan. Query rewriting in retrieval-augmented large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 2023.

[204] Aru Maekawa, Hidetaka Kamigaito, Kotaro Funakoshi, and Manabu Okumura. Generative replay inspired by hippocampal memory indexing for continual language learning. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, 2023.

[205] Yu A Malkov and Dmitry A Yashunin. Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs. *IEEE transactions on pattern analysis and machine intelligence*, 42(4), 2018.

[206] Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Toronto, Canada, July 2023.

[207] Potsawee Manakul, Adian Liusie, and Mark JF Gales. Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models. *arXiv preprint arXiv:2303.08896*, 2023.

[208] Ahmed Masry, Do Xuan Long, Jia Qing Tan, Shafiq Joty, and Enamul Hoque. ChartQA: A benchmark for question answering about charts with visual and logical reasoning. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio, editors, *Findings of the Association for Computational Linguistics: ACL 2022*, May 2022.

[209] Ahmed Masry, Parsa Kavehzadeh, Xuan Long Do, Enamul Hoque, and Shafiq Joty. Unichart: A universal vision-language pretrained model for chart comprehension and reasoning. *arXiv preprint arXiv:2305.14761*, 2023.

[210] Minesh Mathew, Umapada Pal, and CV Jawahar. Docvqa: A dataset for document visual question answering. In *WACV Workshops*, 2021.

[211] Sanket Vaibhav Mehta, Jai Gupta, Yi Tay, Mostafa Dehghani, Vinh Q Tran, Jinfeng Rao, Marc Najork, Emma Strubell, and Donald Metzler. Dsi++: Updating transformer memory with new documents. *arXiv preprint arXiv:2212.09744*, 2022.

[212] Jianbiao Mei, Tao Hu, Daocheng Fu, Licheng Wen, Xuemeng Yang, Rong Wu, Pinlong Cai, Xinyu Cai, Xing Gao, Yu Yang, et al. O2-searcher: A searching-based agent model for open-domain open-ended question answering. *arXiv preprint arXiv:2505.16582*, 2025.

[213] Lang Mei, Zhihan Yang, and Chong Chen. Ai-searchplanner: Modular agentic search via pareto-optimal multi-objective reinforcement learning. *arXiv preprint arXiv:2508.20368*, 2025.

[214] Fanqing Meng, Wenqi Shao, Quanfeng Lu, Peng Gao, Kaipeng Zhang, Yu Qiao, and Ping Luo. Chartassistant: A universal chart multimodal language model via chart-to-table pre-training and multitask instruction tuning. *arXiv preprint arXiv:2401.02384*, 2024.

[215] 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*, 2023.

[216] Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. FActScore: Fine-grained atomic evaluation of factual precision in long form text generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 2023.

[217] Yifei Ming, Senthil Purushwalkam, Shrey Pandit, Zixuan Ke, Xuan-Phi Nguyen, Caiming Xiong, and Shafiq Joty. Faitheval: Can your language model stay faithful to context, even if" the moon is made of marshmallows". *arXiv preprint arXiv:2410.03727*, 2024.

[218] Eric Mitchell, Charles Lin, Antoine Bosselut, Chelsea Finn, and Christopher D Manning. Fast model editing at scale. *arXiv preprint arXiv:2110.11309*, 2021.

[219] Jesse Mu, Xiang Li, and Noah Goodman. Learning to compress prompts with gist tokens. *Advances in Neural Information Processing Systems*, 2023.

[220] Vaishnavh Nagarajan, Aditya K Menon, Srinadh Bhojanapalli, Hossein Mobahi, and Sanjiv Kumar. On student-teacher deviations in distillation: does it pay to disobey? *Advances in Neural Information Processing Systems*, 36:5961–6000, 2023.

[221] Reiichiro Nakano, Jacob Hilton, Suchir Balaji, Jeff Wu, Long Ouyang, Christina Kim, Christopher Hesse, Shantanu Jain, Vineet Kosaraju, William Saunders, et al. Webgpt: Browser-assisted question-answering with human feedback. *arXiv preprint arXiv:2112.09332*, 2021.

[222] Jiayan Nan, Wenquan Ma, Wenlong Wu, and Yize Chen. Nemori: Self-organizing agent memory inspired by cognitive science. *arXiv preprint arXiv:2508.03341*, 2025.

[223] Deepak Nathani, Lovish Madaan, Nicholas Roberts, Nikolay Bashlykov, Ajay Menon, Vincent Moens, Amar Budhiraja, Despoina Magka, Vladislav Vorotilov, Gaurav Chaurasia, et al. Mlgym: A new framework and benchmark for advancing ai research agents. *arXiv preprint arXiv:2502.14499*, 2025.

[224] Humza Naveed, Asad Ullah Khan, Shi Qiu, Muhammad Saqib, Saeed Anwar, Muhammad Usman, Naveed Akhtar, Nick Barnes, and Ajmal Mian. A comprehensive overview of large language models. *ACM Transactions on Intelligent Systems and Technology*, 2025.

[225] Fnu Neha and Deepshikha Bhati. Traditional rag vs. agentic rag: A comparative study of retrieval-augmented systems. *Authorea Preprints*, 2025.

[226] Giang Nguyen, Ivan Brugere, Shubham Sharma, Sanjay Kariyappa, Anh Totti Nguyen, and Freddy Lecue. Interpretable llm-based table question answering. *arXiv preprint arXiv:2412.12386*, 2024.

[227] Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. Ms marco: A human-generated machine reading comprehension dataset. In *International Conference on Learning Representations*, 2016.

[228] Shiyu Ni, Keping Bi, Jiafeng Guo, and Xueqi Cheng. When do llms need retrieval augmentation? mitigating llms' overconfidence helps retrieval augmentation. *arXiv preprint arXiv:2402.11457*, 2024.

[229] Shiyu Ni, Keping Bi, Jiafeng Guo, Minghao Tang, Jingtong Wu, Zengxin Han, and Xueqi Cheng. Annotation-efficient universal honesty alignment. *arXiv preprint arXiv:2510.17509*, 2025.

[230] Shiyu Ni, Keping Bi, Jiafeng Guo, Lulu Yu, Baolong Bi, and Xueqi Cheng. Towards fully exploiting llm internal states to enhance knowledge boundary perception. *arXiv preprint arXiv:2502.11677*, 2025.

[231] Zihan Niu, Zheyong Xie, Shaosheng Cao, Chonggang Lu, Zheyu Ye, Tong Xu, Zuozhu Liu, Yan Gao, Jia Chen, Zhe Xu, et al. Part: Enhancing proactive social chatbots with personalized real-time retrieval. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2025.

[232] Agada Joseph Oche, Ademola Glory Folashade, Tirthankar Ghosal, and Arpan Biswas. A systematic review of key retrieval-augmented generation (rag) systems: Progress, gaps, and future directions. *arXiv preprint arXiv:2507.18910*, 2025.

[233] Minhae Oh, Jeonghye Kim, Nakyung Lee, Donggeon Seo, Taeuk Kim, and Jungwoo Lee. Raise: Enhancing scientific reasoning in llms via step-by-step retrieval. *arXiv preprint arXiv:2506.08625*, 2025.

[234] Utkarsh Ojha, Yuheng Li, Anirudh Sundara Rajan, Yingyu Liang, and Yong Jae Lee. What knowledge gets distilled in knowledge distillation? *Advances in Neural Information Processing Systems*, 36:11037–11048, 2023.

[235] Kai Tzu-iunn Ong, Namyoung Kim, Minju Gwak, Hyungjoo Chae, Taeyoon Kwon, Yohan Jo, Seung-won Hwang, Dongha Lee, and Jinyoung Yeo. Towards lifelong dialogue agents via timeline-based memory management. *arXiv preprint arXiv:2406.10996*, 2024.

[236] OpenAI. Chatgpt: Language model. https://openai.com/, 2023.

[237] OpenAI. Deep research system card. Technical report, OpenAI, February 2025.

[238] OpenAI. Deep research, 2025. URL https://openai.com/index/introducing-deep-research/.

[239] OpenManus. Openmanus: An open multi-agent research framework. https://openmanus.github.io/, 2025. Accessed: 2025-11-13.

[240] Litu Ou, Kuan Li, Huifeng Yin, Liwen Zhang, Zhongwang Zhang, Xixi Wu, Rui Ye, Zile Qiao, Pengjun Xie, Jingren Zhou, et al. Browseconf: Confidence-guided test-time scaling for web agents. *arXiv preprint arXiv:2510.23458*, 2025.

[241] Anne Ouyang, Simon Guo, Simran Arora, Alex L. Zhang, William Hu, Christopher R'e, and Azalia Mirhoseini. Kernelbench: Can llms write efficient gpu kernels? *arXiv preprint arXiv:2502.10517*, 2025.

[242] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 2022.

[243] Charles Packer, Vivian Fang, Shishir_G Patil, Kevin Lin, Sarah Wooders, and Joseph_E Gonzalez. Memgpt: Towards llms as operating systems. *arXiv preprint arXiv:2310.08560*, 2023.

[244] Wei Pang, Kevin Qinghong Lin, Xiangru Jian, Xi He, and Philip Torr. Paper2poster: Towards multimodal poster automation from scientific papers. *arXiv preprint arXiv:2505.21497*, 2025.

[245] Joon Sung Park, Joseph O'Brien, Carrie Jun Cai, Meredith Ringel Morris, Percy Liang, and Michael S Bernstein. Generative agents: Interactive simulacra of human behavior. In *Proceedings of the 36th annual acm symposium on user interface software and technology*, 2023.

[246] Mihir Parmar, Nisarg Patel, Neeraj Varshney, Mutsumi Nakamura, Man Luo, Santosh Mashetty, Arindam Mitra, and Chitta Baral. Towards systematic evaluation of logical reasoning ability of large language models. *CoRR*, 2024.

[247] Boci Peng, Yun Zhu, Yongchao Liu, Xiaohe Bo, Haizhou Shi, Chuntao Hong, Yan Zhang, and Siliang Tang. Graph retrieval-augmented generation: A survey. *arXiv preprint arXiv:2408.08921*, 2024.

[248] Qiyao Peng, Hongtao Liu, Hongyan Xu, Qing Yang, Minglai Shao, and Wenjun Wang. Review-llm: Harnessing large language models for personalized review generation. *arXiv preprint arXiv:2407.07487*, 2024.

[249] Long Phan, Alice Gatti, Ziwen Han, Nathaniel Li, Josephina Hu, Hugh Zhang, Chen Bo Calvin Zhang, Mohamed Shaaban, John Ling, Sean Shi, et al. Humanity's last exam. *arXiv preprint arXiv:2501.14249*, 2025.

[250] Ofir Press, Muru Zhang, Sewon Min, Ludwig Schmidt, Noah A Smith, and Mike Lewis. Measuring and narrowing the compositionality gap in language models. *arXiv preprint arXiv:2210.03350*, 2022.

[251] Ofir Press, Muru Zhang, Sewon Min, Ludwig Schmidt, Noah A Smith, and Mike Lewis. Measuring and narrowing the compositionality gap in language models. In *The 2023 Conference on Empirical Methods in Natural Language Processing*, 2023.

[252] Zehan Qi, Xiao Liu, Iat Long Iong, Hanyu Lai, Xueqiao Sun, Wenyi Zhao, Yu Yang, Xinyue Yang, Jiadai Sun, Shuntian Yao, et al. Webrl: Training llm web agents via self-evolving online curriculum reinforcement learning. *arXiv preprint arXiv:2411.02337*, 2024.

[253] Zile Qiao, Guoxin Chen, Xuanzhong Chen, Donglei Yu, Wenbiao Yin, Xinyu Wang, Zhen Zhang, Baixuan Li, Huifeng Yin, Kuan Li, et al. Webresearcher: Unleashing unbounded reasoning capability in long-horizon agents. *arXiv preprint arXiv:2509.13309*, 2025.

[254] Yujia Qin, Shihao Liang, Yining Ye, Kunlun Zhu, Lan Yan, Yaxi Lu, Yankai Lin, Xin Cong, Xiangru Tang, Bill Qian, et al. Toolllm: Facilitating large language models to master 16000+ real-world apis. In *International Conference on Learning Representations*, 2023.

[255] Zhen Qin, Rolf Jagerman, Kai Hui, Honglei Zhuang, Junru Wu, Le Yan, Jiaming Shen, Tianqi Liu, Jialu Liu, Donald Metzler, et al. Large language models are effective text rankers with pairwise ranking prompting. *arXiv preprint arXiv:2306.17563*, 2023.

[256] Jiahao Qiu, Xinzhe Juan, Yimin Wang, Ling Yang, Xuan Qi, Tongcheng Zhang, Jiacheng Guo, Yifu Lu, Zixin Yao, Hongru Wang, et al. Agentdistill: Training-free agent distillation with generalizable mcp boxes. *arXiv preprint arXiv:2506.14728*, 2025.

[257] Jiahao Qiu, Xuan Qi, Tongcheng Zhang, Xinzhe Juan, Jiacheng Guo, Yifu Lu, Yimin Wang, Zixin Yao, Qihan Ren, Xun Jiang, et al. Alita: Generalist agent enabling scalable agentic reasoning with minimal predefinition and maximal self-evolution. *arXiv preprint arXiv:2505.20286*, 2025.

[258] Yansheng Qiu, Haoquan Zhang, Zhaopan Xu, Ming Li, Diping Song, Zheng Wang, and Kaipeng Zhang. Ai idea bench 2025: Ai research idea generation benchmark. *arXiv preprint arXiv:2504.14191*, 2025.

[259] Yingqi Qu, Yuchen Ding, Jing Liu, Kai Liu, Ruiyang Ren, Wayne Xin Zhao, Daxiang Dong, Hua Wu, and Haifeng Wang. Rocketqa: An optimized training approach to dense passage retrieval for open-domain question answering. *arXiv preprint arXiv:2010.08191*, 2020.

[260] Haoran Que, Feiyu Duan, Liqun He, Yutao Mou, Wangchunshu Zhou, Jiaheng Liu, Wenge Rong, Zekun Moore Wang, Jian Yang, Ge Zhang, et al. Hellobench: Evaluating long text generation capabilities of large language models. *arXiv preprint arXiv:2409.16191*, 2024.

[261] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, 2021.

[262] Preston Rasmussen, Pavlo Paliychuk, Travis Beauvais, Jack Ryan, and Daniel Chalef. Zep: a temporal knowledge graph architecture for agent memory. *arXiv preprint arXiv:2501.13956*, 2025.

[263] David Rau, Shuai Wang, Hervé Déjean, and Stéphane Clinchant. Context embeddings for efficient answer generation in rag. *arXiv preprint arXiv:2407.09252*, 2024.

[264] Alistair Reid, Simon O'Callaghan, Liam Carroll, and Tiberio Caetano. Risk analysis techniques for governed llm-based multi-agent systems. *arXiv preprint arXiv:2508.05687*, 2025.

[265] David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R Bowman. Gpqa: A graduate-level google-proof q&a benchmark. In *First Conference on Language Modeling*, 2024.

[266] Ruiyang Ren, Yuhao Wang, Yingqi Qu, Wayne Xin Zhao, Jing Liu, Hua Wu, Ji-Rong Wen, and Haifeng Wang. Investigating the factual knowledge boundary of large language models with retrieval augmentation. In *Proceedings of the 31st International Conference on Computational Linguistics*, 2025.

[267] ZZ Ren, Zhihong Shao, Junxiao Song, Huajian Xin, Haocheng Wang, Wanjia Zhao, Liyue Zhang, Zhe Fu, Qihao Zhu, Dejian Yang, et al. Deepseek-prover-v2: Advancing formal mathematical reasoning via reinforcement learning for subgoal decomposition. *arXiv preprint arXiv:2504.21801*, 2025.

[268] Alireza Rezazadeh, Zichao Li, Wei Wei, and Yujia Bao. From isolated conversations to hierarchical schemas: Dynamic tree memory representation for llms. *arXiv preprint arXiv:2410.14052*, 2024.

[269] Stephen Robertson and Hugo Zaragoza. *The Probabilistic Relevance Framework: BM25 and Beyond*. Now Publishers Inc., 2009.

[270] Bernardino Romera-Paredes, Mohammadamin Barekatain, Alexander Novikov, Matej Balog, M. Pawan Kumar, Emilien Dupont, Francisco J. R. Ruiz, Jordan S. Ellenberg, Pengming Wang, Omar Fawzi, Pushmeet Kohli, Alhussein Fawzi, Josh Grochow, Andrea Lodi, Jean-Baptiste Mouret, Talia Ringer, and Tao Yu. Mathematical discoveries from program search with large language models. *Nature*, 2023.

[271] Nirmal Roy, Leonardo F. R. Ribeiro, Rexhina Blloshmi, and Kevin Small. Learning when to retrieve, what to rewrite, and how to respond in conversational QA. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, 2024.

[272] Aishwarya Sahoo, Jeevana Kruthi Karnuthala, Tushar Parmanand Budhwani, Pranchal Agarwal, Sankaran Vaidyanathan, Alexa Siu, Franck Dernoncourt, Jennifer Healey, Nedim Lipka, Ryan Rossi, et al. Quantitative llm judges. *arXiv preprint arXiv:2506.02945*, 2025.

[273] Keshav Santhanam, Omar Khattab, Jon Saad-Falcon, Christopher Potts, and Matei Zaharia. Colbertv2: Effective and efficient retrieval via lightweight late interaction. *arXiv preprint arXiv:2112.01488*, 2021.

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

[275] John Schulman, Philipp Moritz, Sergey Levine, Michael I. Jordan, and P. Abbeel. High-dimensional continuous control using generalized advantage estimation. *CoRR*, 2015.

[276] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

[277] Hinrich Schütze, Christopher D Manning, and Prabhakar Raghavan. *Introduction to information retrieval*, volume 39. Cambridge University Press Cambridge, 2008.

[278] Yunfan Shao, Linyang Li, Junqi Dai, and Xipeng Qiu. Character-llm: A trainable agent for role-playing. *arXiv preprint arXiv:2310.10158*, 2023.

[279] Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

[280] Xu Shen, Yixin Liu, Yiwei Dai, Yili Wang, Rui Miao, Yue Tan, Shirui Pan, and Xin Wang. Understanding the information propagation effects of communication topologies in llm-based multi-agent systems. *arXiv preprint arXiv:2505.23352*, 2025.

[281] Zhang Shengyu, Dong Linfeng, Li Xiaoya, Zhang Sen, Sun Xiaofei, Wang Shuhe, Li Jiwei, Runyi Hu, Zhang Tianwei, Fei Wu, et al. Instruction tuning for large language models: A survey. *arXiv preprint arXiv:2308.10792*, 2023.

[282] Wenxuan Shi, Haochen Tan, Chuqiao Kuang, Xiaoguang Li, Xiaozhe Ren, Chen Zhang, Hanting Chen, Yasheng Wang, Lifeng Shang, Fisher Yu, et al. Pangu deepdiver: Adaptive search intensity scaling via open-web reinforcement learning. *arXiv preprint arXiv:2505.24332*, 2025.

[283] Yaorui Shi, Shihan Li, Chang Wu, Zhiyuan Liu, Junfeng Fang, Hengxing Cai, An Zhang, and Xiang Wang. Search and refine during think: Autonomous retrieval-augmented reasoning of llms. *arXiv e-prints*, pages arXiv–2505, 2025.

[284] Zhengliang Shi, Shen Gao, Zhen Zhang, Xiuying Chen, Zhumin Chen, Pengjie Ren, and Zhaochun Ren. Towards a unified framework for reference retrieval and related work generation. In *Findings of the Association for Computational Linguistics*, 2023.

[285] Zhengliang Shi, Shen Gao, Xiuyi Chen, Yue Feng, Lingyong Yan, Haibo Shi, Dawei Yin, Pengjie Ren, Suzan Verberne, and Zhaochun Ren. Learning to use tools via cooperative and interactive agents. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, 2024.

[286] Zhengliang Shi, Shuo Zhang, Weiwei Sun, Shen Gao, Pengjie Ren, Zhumin Chen, and Zhaochun Ren. Generate-then-ground in retrieval-augmented generation for multi-hop question answering. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2024.

[287] Zhengliang Shi, Shen Gao, Lingyong Yan, Yue Feng, Xiuyi Chen, Zhumin Chen, Dawei Yin, Suzan Verberne, and Zhaochun Ren. Tool learning in the wild: Empowering language models as automatic tool agents. In *Proceedings of the ACM on Web Conference 2025*, pages 2222–2237, 2025.

[288] Zhengliang Shi, Ruotian Ma, Jen-tse Huang, Xinbei Ma, Xingyu Chen, Mengru Wang, Qu Yang, Yue Wang, Fanghua Ye, Ziyang Chen, et al. Social welfare function leaderboard: When llm agents allocate social welfare. *arXiv preprint arXiv:2510.01164*, 2025.

[289] Zhengliang Shi, Lingyong Yan, Dawei Yin, Suzan Verberne, Maarten de Rijke, and Zhaochun Ren. Iterative self-incentivization empowers large language models as agentic searchers. *arXiv preprint arXiv:2505.20128*, 2025.

[290] Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning. *Advances in Neural Information Processing Systems*, 36, 2023.

[291] Parshin Shojaee, Kazem Meidani, Shashank Gupta, Amir Barati Farimani, and Chandan K Reddy. Llm-sr: Scientific equation discovery via programming with large language models. *arXiv preprint arXiv:2404.18400*, 2024.

[292] Parshin Shojaee, Ngoc-Hieu Nguyen, Kazem Meidani, Amir Barati Farimani, Khoa D Doan, and Chandan K Reddy. Llm-srbench: A new benchmark for scientific equation discovery with large language models. *arXiv preprint arXiv:2504.10415*, 2025.

[293] Ilia Shumailov, Zakhar Shumaylov, Yiren Zhao, Yarin Gal, Nicolas Papernot, and Ross Anderson. The curse of recursion: Training on generated data makes models forget. *arXiv preprint arXiv:2305.17493*, 2023.

[294] Ilia Shumailov, Zakhar Shumaylov, Yiren Zhao, Nicolas Papernot, Ross Anderson, and Yarin Gal. Ai models collapse when trained on recursively generated data. *Nature*, 2024.

[295] Chenglei Si, Zhe Gan, Zhengyuan Yang, Shuohang Wang, Jianfeng Wang, Jordan Boyd-Graber, and Lijuan Wang. Prompting gpt-3 to be reliable. *arXiv preprint arXiv:2210.09150*, 2022.

[296] Chenglei Si, Diyi Yang, and Tatsunori Hashimoto. Can llms generate novel research ideas? a large-scale human study with 100+ nlp researchers. *arXiv preprint arXiv:2409.04109*, 2024.

[297] Zachary S Siegel, Sayash Kapoor, Nitya Nagdir, Benedikt Stroebl, and Arvind Narayanan. Corebench: Fostering the credibility of published research through a computational reproducibility agent benchmark. *arXiv preprint arXiv:2409.11363*, 2024.

[298] David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, et al. Mastering the game of go without human knowledge. *nature*, 2017.

[299] Amit Singhal et al. Modern information retrieval: A brief overview. *IEEE Data Eng. Bull.*, 2001.

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

[301] Huatong Song, Jinhao Jiang, Wenqing Tian, Zhipeng Chen, Yuhuan Wu, Jiahao Zhao, Yingqian Min, Wayne Xin Zhao, Lei Fang, and Ji-Rong Wen. R1-searcher++: Incentivizing the dynamic knowledge acquisition of llms via reinforcement learning. *arXiv preprint arXiv:2505.17005*, 2025.

[302] Zayne Sprague, Fangcong Yin, Juan Diego Rodriguez, Dongwei Jiang, Manya Wadhwa, Prasann Singhal, Xinyu Zhao, Xi Ye, Kyle Mahowald, and Greg Durrett. To cot or not to cot? chain-of-thought helps mainly on math and symbolic reasoning. *arXiv preprint arXiv:2409.12183*, 2024.

[303] Larry R Squire, Lisa Genzel, John T Wixted, and Richard G Morris. Memory consolidation. *Cold Spring Harbor perspectives in biology*, 7(8):a021766, 2015.

[304] Giulio Starace, Oliver Jaffe, Dane Sherburn, James Aung, Jun Shern Chan, Leon Maksin, Rachel Dias, Evan Mays, Benjamin Kinsella, Wyatt Thompson, et al. Paperbench: Evaluating ai's ability to replicate ai research. *arXiv preprint arXiv:2504.01848*, 2025.

[305] Buxin Su, Jiayao Zhang, Natalie Collina, Yuling Yan, Didong Li, Kyunghyun Cho, Jianqing Fan, Aaron Roth, and Weijie Su. The icml 2023 ranking experiment: Examining author self-assessment in ml/ai peer review. *Journal of the American Statistical Association*, pages 1–16, 2025.

[306] Haoyang Su, Renqi Chen, Shixiang Tang, Xinzhe Zheng, Jingzhe Li, Zhenfei Yin, Wanli Ouyang, and Nanqing Dong. Two heads are better than one: A multi-agent system has the potential to improve scientific idea generation. *CoRR*, 2024.

[307] Liangcai Su, Zhen Zhang, Guangyu Li, Zhuo Chen, Chenxi Wang, Maojia Song, Xinyu Wang, Kuan Li, Jialong Wu, Xuanzhong Chen, et al. Scaling agents via continual pre-training. *arXiv preprint arXiv:2509.13310*, 2025.

[308] Weihang Su, Yichen Tang, Qingyao Ai, Zhijing Wu, and Yiqun Liu. Dragin: dynamic retrieval augmented generation based on the information needs of large language models. *arXiv preprint arXiv:2403.10081*, 2024.

[309] Weihang Su, Changyue Wang, Qingyao Ai, Yiran Hu, Zhijing Wu, Yujia Zhou, and Yiqun Liu. Unsupervised real-time hallucination detection based on the internal states of large language models. *arXiv preprint arXiv:2403.06448*, 2024.

[310] Hao Sun, Hengyi Cai, Bo Wang, Yingyan Hou, Xiaochi Wei, Shuaiqiang Wang, Yan Zhang, and Dawei Yin. Towards verifiable text generation with evolving memory and self-reflection. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, 2024.

[311] Hao Sun, Zile Qiao, Jiayan Guo, Xuanbo Fan, Yingyan Hou, Yong Jiang, Pengjun Xie, Yan Zhang, Fei Huang, and Jingren Zhou. Zerosearch: Incentivize the search capability of llms without searching. *arXiv preprint arXiv:2505.04588*, 2025.

[312] Haoran Sun and Shaoning Zeng. Hierarchical memory for high-efficiency long-term reasoning in llm agents. *arXiv preprint arXiv:2507.22925*, 2025.

[313] Jiashuo Sun, Xianrui Zhong, Sizhe Zhou, and Jiawei Han. Dynamicrag: Leveraging outputs of large language model as feedback for dynamic reranking in retrieval-augmented generation, 2025.

[314] Shuang Sun, Huatong Song, Yuhao Wang, Ruiyang Ren, Jinhao Jiang, Junjie Zhang, Fei Bai, Jia Deng, Wayne Xin Zhao, Zheng Liu, et al. Simpledeepsearcher: Deep information seeking via web-powered reasoning trajectory synthesis. *arXiv preprint arXiv:2505.16834*, 2025.

[315] Tao Sun, Enhao Pan, Zhengkai Yang, Kaixin Sui, Jiajun Shi, Xianfu Cheng, Tongliang Li, Wenhao Huang, Ge Zhang, Jian Yang, et al. P2p: Automated paper-to-poster generation and fine-grained benchmark. *arXiv preprint arXiv:2505.17104*, 2025.

[316] Weiwei Sun, Zhengliang Shi, Shen Gao, Pengjie Ren, Maarten de Rijke, and Zhaochun Ren. Contrastive learning reduces hallucination in conversations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2023.

[317] Weiwei Sun, Lingyong Yan, Xinyu Ma, Shuaiqiang Wang, Pengjie Ren, Zhumin Chen, Dawei Yin, and Zhaochun Ren. Is chatgpt good at search? investigating large language models as re-ranking agents. *arXiv preprint arXiv:2304.09542*, 2023.

[318] Weiwei Sun, Shengyu Feng, Shanda Li, and Yiming Yang. Co-bench: Benchmarking language model agents in algorithm search for combinatorial optimization. *arXiv preprint arXiv:2504.04310*, 2025.

[319] Weiwei Sun, Miao Lu, Zhan Ling, Kang Liu, Xuesong Yao, Yiming Yang, and Jiecao Chen. Scaling long-horizon llm agent via context-folding. *arXiv preprint arXiv:2510.11967*, 2025.

[320] supermancmk. when running the official gsm8k with tool, multi turn async rollout sglang example without any modifications, the model crashes and appears nan. https://github.com/volcengine/verl/issues/1581, 2025.

[321] Jihoon Tack, Jaehyung Kim, Eric Mitchell, Jinwoo Shin, Yee Whye Teh, and Jonathan Richard Schwarz. Online adaptation of language models with a memory of amortized contexts. *Advances in Neural Information Processing Systems*, 2024.

[322] Haochen Tan, Zhijiang Guo, Zhan Shi, Lu Xu, Zhili Liu, Yunlong Feng, Xiaoguang Li, Yasheng Wang, Lifeng Shang, Qun Liu, et al. Proxyqa: An alternative framework for evaluating long-form text generation with large language models. *arXiv preprint arXiv:2401.15042*, 2024.

[323] Jiejun Tan, Zhicheng Dou, Wen Wang, Mang Wang, Weipeng Chen, and Ji-Rong Wen. Htmlrag: Html is better than plain text for modeling retrieved knowledge in rag systems. In *Proceedings of the ACM on Web Conference 2025*, 2025.

[324] Jiejun Tan, Zhicheng Dou, Yan Yu, Jiehan Cheng, Qiang Ju, Jian Xie, and Ji-Rong Wen. Hiersearch: A hierarchical enterprise deep search framework integrating local and web searches. *arXiv preprint arXiv:2508.08088*, 2025.

[325] Zhen Tan, Jun Yan, I Hsu, Rujun Han, Zifeng Wang, Long T Le, Yiwen Song, Yanfei Chen, Hamid Palangi, George Lee, et al. In prospect and retrospect: Reflective memory management for long-term personalized dialogue agents. *arXiv preprint arXiv:2503.08026*, 2025.

[326] Jiabin Tang, Lianghao Xia, Zhonghang Li, and Chao Huang. Ai-researcher: Autonomous scientific innovation. *arXiv preprint arXiv:2505.18705*, 2025.

[327] Liyan Tang, Grace Kim, Xinyu Zhao, Thom Lake, Wenxuan Ding, Fangcong Yin, Prasann Singhal, Manya Wadhwa, Zeyu Leo Liu, Zayne Sprague, Ramya Namuduri, Bodun Hu, Juan Diego Rodriguez, Puyuan Peng, and Greg Durrett. Chartmuseum: Testing visual reasoning capabilities of large vision-language models. *arXiv preprint arXiv:2505.13444*, 2025.

[328] Shuo Tang, Xianghe Pang, Zexi Liu, Bohan Tang, Rui Ye, Tian Jin, Xiaowen Dong, Yanfeng Wang, and Siheng Chen. Synthesizing post-training data for llms through multi-agent simulation. *arXiv preprint arXiv:2410.14251*, 2024.

[329] Xiangru Tang, Tianrui Qin, Tianhao Peng, Ziyang Zhou, Daniel Shao, Tingting Du, Xinming Wei, Peng Xia, Fang Wu, He Zhu, et al. Agent kb: Leveraging cross-domain experience for agentic problem solving. *arXiv preprint arXiv:2507.06229*, 2025.

[330] Xiaqiang Tang, Jian Li, Keyu Hu, Du Nan, Xiaolong Li, Xi Zhang, Weigao Sun, and Sihong Xie. Cognibench: A legal-inspired framework and dataset for assessing cognitive faithfulness of large language models. *arXiv preprint arXiv:2505.20767*, 2025.

[331] Yihong Tang, Zhaokai Wang, Ao Qu, Yihao Yan, Zhaofeng Wu, Dingyi Zhuang, Jushi Kai, Kebing Hou, Xiaotong Guo, Han Zheng, et al. Itinera: Integrating spatial optimization with large language models for open-domain urban itinerary planning. *arXiv preprint arXiv:2402.07204*, 2024.

[332] Yixuan Tang and Yi Yang. Multihop-rag: Benchmarking retrieval-augmented generation for multi-hop queries. *arXiv preprint arXiv:2401.15391*, 2024.

[333] Md Mehrab Tanjim, Yeonjun In, Xiang Chen, Victor S Bursztyn, Ryan A Rossi, Sungchul Kim, Guang-Jie Ren, Vaishnavi Muppala, Shun Jiang, Yongsung Kim, et al. Disambiguation in conversational question answering in the era of llm: A survey. *arXiv preprint arXiv:2505.12543*, 2025.

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

[335] Zhengwei Tao, Jialong Wu, Wenbiao Yin, Junkai Zhang, Baixuan Li, Haiyang Shen, Kuan Li, Liwen Zhang, Xinyu Wang, Yong Jiang, et al. Webshaper: Agentically data synthesizing via information-seeking formalization. *arXiv preprint arXiv:2507.15061*, 2025.

[336] Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B Hashimoto. Stanford alpaca: An instruction-following llama model, 2023.

[337] Anthropic Engineering Team. How we built our multi-agent research system, June 2025. URL https://www.anthropic.com/engineering/built-multi-agent-research-system. Accessed 2025-08-27.

[338] Kimi Team, Yifan Bai, Yiping Bao, Guanduo Chen, Jiahao Chen, Ningxin Chen, Ruijue Chen, Yanru Chen, Yuankun Chen, Yutian Chen, et al. Kimi k2: Open agentic intelligence. *arXiv preprint arXiv:2507.20534*, 2025.

[339] James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. FEVER: a large-scale dataset for fact extraction and VERification. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2018.

[340] Katherine Tian, Eric Mitchell, Allan Zhou, Archit Sharma, Rafael Rafailov, Huaxiu Yao, Chelsea Finn, and Christopher D Manning. Just ask for calibration: Strategies for eliciting calibrated confidence scores from language models fine-tuned with human feedback. *arXiv preprint arXiv:2305.14975*, 2023.

[341] Shulin Tian, Ziniu Zhang, Liangyu Chen, and Ziwei Liu. Mmina: Benchmarking multihop multimodal internet agents. *arXiv preprint arXiv:2404.09992*, 2024.

[342] Yong-En Tian, Yu-Chien Tang, Kuang-Da Wang, An-Zi Yen, and Wen-Chih Peng. Template-based financial report generation in agentic and decomposed information retrieval. *arXiv preprint arXiv:2504.14233*, 2025.

[343] Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. Interleaving retrieval with chain-of-thought reasoning for knowledge-intensive multi-step questions. *arXiv preprint arXiv:2212.10509*, 2022.

[344] Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. Musique: Multihop questions via single-hop question composition. *Transactions of the Association for Computational Linguistics*, 10, 2022.

[345] 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. *Advances in Neural Information Processing Systems*, 2024.

[346] Dibia Victor. Lida: A tool for automatic generation of grammar-agnostic visualizations and infographics using large language models. *arXiv preprint arXiv:2303.02927*, 2023.

[347] Alisa Vinogradova, Vlad Vinogradov, Dmitrii Radkevich, Ilya Yasny, Dmitry Kobyzev, Ivan Izmailov, Katsiaryna Yanchanka, Roman Doronin, and Andrey Doronichev. Llm-based agents for competitive landscape mapping in drug asset due diligence. *arXiv preprint arXiv:2508.16571*, 2025.

[348] Yuxuan Wan, Wenxuan Wang, Yiliu Yang, Youliang Yuan, Jen-tse Huang, Pinjia He, Wenxiang Jiao, and Michael Lyu. Logicasker: Evaluating and improving the logical reasoning ability of large language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, 2024.

[349] 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. *arXiv preprint arXiv:2305.16291*, 2023.

[350] Han Wang, Archiki Prasad, Elias Stengel-Eskin, and Mohit Bansal. Retrieval-augmented generation with conflicting evidence. In *Second Conference on Language Modeling*, 2025.

[351] Han Wang, Archiki Prasad, Elias Stengel-Eskin, and Mohit Bansal. Retrieval-augmented generation with conflicting evidence. *arXiv preprint arXiv:2504.13079*, 2025.

[352] Jiayu Wang, Yifei Ming, Riya Dulepet, Qinglin Chen, Austin Xu, Zixuan Ke, Frederic Sala, Aws Albarghouthi, Caiming Xiong, and Shafiq Joty. Liveresearchbench: A live benchmark for user-centric deep research in the wild. *arXiv preprint arXiv:2510.14240*, 2025.

[353] Juyuan Wang, Rongchen Zhao, Wei Wei, Yufeng Wang, Mo Yu, Jie Zhou, Jin Xu, and Liyan Xu. Comorag: A cognitive-inspired memory-organized rag for stateful long narrative reasoning. *arXiv preprint arXiv:2508.10419*, 2025.

[354] Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. Text embeddings by weakly-supervised contrastive pre-training. *arXiv preprint arXiv:2212.03533*, 2022.

[355] Liang Wang, Nan Yang, and Furu Wei. Learning to retrieve in-context examples for large language models. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics*, March 2024.

[356] Maggie Wang, Ella Colby, Jennifer Okwara, Varun Nagaraj Rao, Yuhan Liu, and Andrés Monroy-Hernández. Policypulse: Llm-synthesis tool for policy researchers. In *Proceedings of the Extended Abstracts of the CHI Conference on Human Factors in Computing Systems*, 2025.

[357] Peng Wang, Zexi Li, Ningyu Zhang, Ziwen Xu, Yunzhi Yao, Yong Jiang, Pengjun Xie, Fei Huang, and Huajun Chen. Wise: Rethinking the knowledge memory for lifelong model editing of large language models. *Advances in Neural Information Processing Systems*, 2024.

[358] Qingyue Wang, Yanhe Fu, Yanan Cao, Shuai Wang, Zhiliang Tian, and Liang Ding. Recursively summarizing enables long-term dialogue memory in large language models. *Neurocomputing*, 639:130193, 2025.

[359] Ruoyao Wang, Peter Jansen, Marc-Alexandre Côté, and Prithviraj Ammanabrolu. Scienceworld: Is your agent smarter than a 5th grader? *arXiv preprint arXiv:2203.07540*, 2022.

[360] Ruoyao Wang, Graham Todd, Ziang Xiao, Xingdi Yuan, Marc-Alexandre Côté, Peter Clark, and Peter Jansen. Can language models serve as text-based world simulators? *arXiv preprint arXiv:2406.06485*, 2024.

[361] Song Wang, Yaochen Zhu, Haochen Liu, Zaiyi Zheng, Chen Chen, and Jundong Li. Knowledge editing for large language models: A survey. *ACM Computing Surveys*, 2024.

[362] Xingyao Wang, Boxuan Li, Yufan Song, Frank F Xu, Xiangru Tang, Mingchen Zhuge, Jiayi Pan, Yueqi Song, Bowen Li, Jaskirat Singh, et al. Openhands: An open platform for ai software developers as generalist agents. *arXiv preprint arXiv:2407.16741*, 2024.

[363] Xintao Wang, Heng Wang, Yifei Zhang, Xinfeng Yuan, Rui Xu, Jen-tse Huang, Siyu Yuan, Haoran Guo, Jiangjie Chen, Shuchang Zhou, et al. Coser: Coordinating llm-based persona simulation of established roles. In *The Forty-second International Conference on Machine Learning*, 2025.

[364] Yanling Wang, Haoyang Li, Hao Zou, Jing Zhang, Xinlei He, Qi Li, and Ke Xu. Hidden question representations tell non-factuality within and across large language models. *arXiv preprint arXiv:2406.05328*, 2024.

[365] Yao Wang, Mingxuan Cui, and Arthur Jiang. Enabling ai scientists to recognize innovation: A domain-agnostic algorithm for assessing novelty. *arXiv preprint arXiv:2503.01508*, 2025.

[366] Yidong Wang, Qi Guo, Wenjin Yao, Hongbo Zhang, Xin Zhang, Zhen Wu, Meishan Zhang, Xinyu Dai, Qingsong Wen, Wei Ye, et al. Autosurvey: Large language models can automatically write surveys. *Advances in neural information processing systems*, 2024.

[367] Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A Smith, Daniel Khashabi, and Hannaneh Hajishirzi. Self-instruct: Aligning language models with self-generated instructions. *arXiv preprint arXiv:2212.10560*, 2022.

[368] Yu Wang and Xi Chen. Mirix: Multi-agent memory system for llm-based agents. *arXiv preprint arXiv:2507.07957*, 2025.

[369] Yu Wang, Yifan Gao, Xiusi Chen, Haoming Jiang, Shiyang Li, Jingfeng Yang, Qingyu Yin, Zheng Li, Xian Li, Bing Yin, et al. Memoryllm: Towards self-updatable large language models. *arXiv preprint arXiv:2402.04624*, 2024.

[370] Yu Wang, Chi Han, Tongtong Wu, Xiaoxin He, Wangchunshu Zhou, Nafis Sadeq, Xiusi Chen, Zexue He, Wei Wang, Gholamreza Haffari, et al. Towards lifespan cognitive systems. *arXiv preprint arXiv:2409.13265*, 2024.

[371] Yuhao Wang, Ruiyang Ren, Junyi Li, Wayne Xin Zhao, Jing Liu, and Ji-Rong Wen. Rear: A relevance-aware retrieval-augmented framework for open-domain question answering. *arXiv preprint arXiv:2402.17497*, 2024.

[372] Yuhao Wang, Ruiyang Ren, Yucheng Wang, Wayne Xin Zhao, Jing Liu, Hua Wu, and Haifeng Wang. Reinforced informativeness optimization for long-form retrieval-augmented generation, 2025.

[373] Zihan Wang, Kangrui Wang, Qineng Wang, Pingyue Zhang, Linjie Li, Zhengyuan Yang, Xing Jin, Kefan Yu, Minh Nhat Nguyen, Licheng Liu, et al. Ragen: Understanding self-evolution in llm agents via multi-turn reinforcement learning. *arXiv preprint arXiv:2504.20073*, 2025.

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

[375] Anjiang Wei, Yuheng Wu, Yingjia Wan, Tarun Suresh, Huanmi Tan, Zhanke Zhou, Sanmi Koyejo, Ke Wang, and Alex Aiken. Satbench: Benchmarking llms' logical reasoning via automated puzzle generation from sat formulas. *arXiv preprint arXiv:2505.14615*, 2025.

[376] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 2022.

[377] Jason Wei, Nguyen Karina, Hyung Won Chung, Yunxin Joy Jiao, Spencer Papay, Amelia Glaese, John Schulman, and William Fedus. Measuring short-form factuality in large language models. *arXiv preprint arXiv:2411.04368*, 2024.

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

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

[380] Rubin Wei, Jiaqi Cao, Jiarui Wang, Jushi Kai, Qipeng Guo, Bowen Zhou, and Zhouhan Lin. Mlp memory: Language modeling with retriever-pretrained external memory. *arXiv preprint arXiv:2508.01832*, 2025.

[381] Zhepei Wei, Wei-Lin Chen, and Yu Meng. Instructrag: Instructing retrieval-augmented generation via self-synthesized rationales. *arXiv preprint arXiv:2406.13629*, 2024.

[382] Zhepei Wei, Wenlin Yao, Yao Liu, Weizhi Zhang, Qin Lu, Liang Qiu, Changlong Yu, Puyang Xu, Chao Zhang, Bing Yin, et al. Webagent-r1: Training web agents via end-to-end multi-turn reinforcement learning. *arXiv preprint arXiv:2505.16421*, 2025.

[383] Yixuan Weng and Bin Li. Visual answer localization with cross-modal mutual knowledge transfer. In *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023.

[384] Yixuan Weng, Minjun Zhu, Guangsheng Bao, Hongbo Zhang, Jindong Wang, Yue Zhang, and Linyi Yang. Cycleresearcher: Improving automated research via automated review. *arXiv preprint arXiv:2411.00816*, 2024.

[385] Yixuan Weng, Minjun Zhu, Guangsheng Bao, Hongbo Zhang, Jindong Wang, Yue Zhang, and Linyi Yang. Cycleresearcher: Improving automated research via automated review. In *The Thirteenth International Conference on Learning Representations*, 2025.

[386] Yixuan Weng, Minjun Zhu, Qiujie Xie, Qiyao Sun, Zhen Lin, Sifan Liu, and Yue Zhang. Deepscientist: Advancing frontier-pushing scientific findings progressively. *arXiv preprint arXiv:2509.26603*, 2025.

[387] Hjalmar Wijk, Tao Lin, Joel Becker, Sami Jawhar, Neev Parikh, Thomas Broadley, Lawrence Chan, Michael Chen, Josh Clymer, Jai Dhyani, et al. Re-bench: Evaluating frontier ai r&d capabilities of language model agents against human experts. *arXiv preprint arXiv:2411.15114*, 2024.

[388] Ryan Wong, Jiawei Wang, Junjie Zhao, Li Chen, Yan Gao, Long Zhang, Xuan Zhou, Zuo Wang, Kai Xiang, Ge Zhang, et al. Widesearch: Benchmarking agentic broad info-seeking. *arXiv preprint arXiv:2508.07999*, 2025.

[389] Chen Henry Wu. Roll the dice & look before you leap: Going beyond the creative limits of next-token prediction. In *International Conference on Machine Learning*, 2025.

[390] Di Wu, Hongwei Wang, Wenhao Yu, Yuwei Zhang, Kai-Wei Chang, and Dong Yu. Longmemeval: Benchmarking chat assistants on long-term interactive memory. *arXiv preprint arXiv:2410.10813*, 2024.

[391] Jialong Wu, Baixuan Li, Runnan Fang, Wenbiao Yin, Liwen Zhang, Zhengwei Tao, Dingchu Zhang, Zekun Xi, Gang Fu, Yong Jiang, et al. Webdancer: Towards autonomous information seeking agency. *arXiv preprint arXiv:2505.22648*, 2025.

[392] Jialong Wu, Wenbiao Yin, Yong Jiang, Zhenglin Wang, Zekun Xi, Runnan Fang, Linhai Zhang, Yulan He, Deyu Zhou, Pengjun Xie, et al. Webwalker: Benchmarking llms in web traversal. *arXiv preprint arXiv:2501.07572*, 2025.

[393] Jiayi Wu, Hengyi Cai, Lingyong Yan, Hao Sun, Xiang Li, Shuaiqiang Wang, Dawei Yin, and Ming Gao. Pa-rag: Rag alignment via multi-perspective preference optimization. *arXiv preprint arXiv:2412.14510*, 2024.

[394] Jinming Wu, Zihao Deng, Wei Li, Yiding Liu, Bo You, Bo Li, Zejun Ma, and Ziwei Liu. Mmsearch-r1: Incentivizing lmms to search. *arXiv preprint arXiv:2506.20670*, 2025.

[395] Mingyan Wu, Zhenghao Liu, Yukun Yan, Xinze Li, Shi Yu, Zheni Zeng, Yu Gu, and Ge Yu. Rankcot: Refining knowledge for retrieval-augmented generation through ranking chain-of-thoughts. *arXiv preprint arXiv:2502.17888*, 2025.

[396] Peilin Wu, Mian Zhang, Xinlu Zhang, Xinya Du, and Zhiyu Chen. Search wisely: Mitigating sub-optimal agentic searches by reducing uncertainty. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, 2025.

[397] Weiqi Wu, Xin Guan, Shen Huang, Yong Jiang, Pengjun Xie, Fei Huang, Jiuxin Cao, Hai Zhao, and Jingren Zhou. Masksearch: A universal pre-training framework to enhance agentic search capability. *arXiv preprint arXiv:2505.20285*, 2025.

[398] 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. *arXiv preprint arXiv:2504.15965*, 2025.

[399] Yuhao Wu, Yushi Bai, Zhiqiang Hu, Juanzi Li, and Roy Ka-Wei Lee. Superwriter: Reflection-driven long-form generation with large language models, 2025.

[400] xAI. Grok deepsearch, 2025. URL https://grok.com/.

[401] Yunjia Xi, Jianghao Lin, Yongzhao Xiao, Zheli Zhou, Rong Shan, Te Gao, Jiachen Zhu, Weiwen Liu, Yong Yu, and Weinan Zhang. A survey of llm-based deep search agents: Paradigm, optimization, evaluation, and challenges. *arXiv preprint arXiv:2508.05668*, 2025.

[402] Yunjia Xi, Jianghao Lin, Menghui Zhu, Yongzhao Xiao, Zhuoying Ou, Jiaqi Liu, Tong Wan, Bo Chen, Weiwen Liu, Yasheng Wang, et al. Infodeepseek: Benchmarking agentic information seeking for retrieval-augmented generation. *arXiv preprint arXiv:2505.15872*, 2025.

[403] Fei Xia, Bin Li, Yixuan Weng, Shizhu He, Kang Liu, Bin Sun, Shutao Li, and Jun Zhao. MedConQA: Medical conversational question answering system based on knowledge graphs. In Wanxiang Che and Ekaterina Shutova, editors, *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 2022.

[404] Renqiu Xia, Bo Zhang, Hancheng Ye, Xiangchao Yan, Qi Liu, Hongbin Zhou, Zijun Chen, Peng Ye, Min Dou, Botian Shi, Junchi Yan, and Yu Qiao. Chartx & chartvlm: A versatile benchmark and foundation model for complicated chart reasoning. *arXiv preprint arXiv:2402.12185*, 2024.

[405] Chenghao Xiao, Hou Pong Chan, Hao Zhang, Mahani Aljunied, Lidong Bing, Noura Al Moubayed, and Yu Rong. Analyzing llms' knowledge boundary cognition across languages through the lens of internal representations. *arXiv preprint arXiv:2504.13816*, 2025.

[406] Shitao Xiao, Zheng Liu, Peitian Zhang, Niklas Muennighoff, Defu Lian, and Jian-Yun Nie. C-pack: Packed resources for general chinese embeddings. In *Proceedings of the 47th international ACM SIGIR conference on research and development in information retrieval*, pages 641–649, 2024.

[407] Qiujie Xie, Qiming Feng, Yuejie Zhang, Rui Feng, Tao Zhang, and Shang Gao. Controlcap: Controllable captioning via no-fuss lexicon. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8326–8330. IEEE, 2024.

[408] Qiujie Xie, Yixuan Weng, Minjun Zhu, Fuchen Shen, Shulin Huang, Zhen Lin, Jiahui Zhou, Zilan Mao, Zijie Yang, Linyi Yang, et al. How far are ai scientists from changing the world? *arXiv preprint arXiv:2507.23276*, 2025.

[409] Miao Xiong, Zhiyuan Hu, Xinyang Lu, Yifei Li, Jie Fu, Junxian He, and Bryan Hooi. Can llms express their uncertainty? an empirical evaluation of confidence elicitation in llms. *arXiv preprint arXiv:2306.13063*, 2023.

[410] Fangyuan Xu, Weijia Shi, and Eunsol Choi. Recomp: Improving retrieval-augmented lms with compression and selective augmentation. *arXiv preprint arXiv:2310.04408*, 2023.

[411] Mufan Xu, Gewen Liang, Kehai Chen, Wei Wang, Xun Zhou, Muyun Yang, Tiejun Zhao, and Min Zhang. Memory-augmented query reconstruction for llm-based knowledge graph reasoning. *arXiv preprint arXiv:2503.05193*, 2025.

[412] Ruiyun Xu, Yue Feng, and Hailiang Chen. Chatgpt vs. google: A comparative study of search performance and user experience. *arXiv preprint arXiv:2307.01135*, 2023.

[413] Tianze Xu, Pengrui Lu, Lyumanshan Ye, Xiangkun Hu, and Pengfei Liu. Researcherbench: Evaluating deep ai research systems on the frontiers of scientific inquiry. *arXiv preprint arXiv:2507.16280*, 2025.

[414] Wujiang Xu, Kai Mei, Hang Gao, Juntao Tan, Zujie Liang, and Yongfeng Zhang. A-mem: Agentic memory for llm agents. *arXiv preprint arXiv:2502.12110*, 2025.

[415] Yiheng Xu and et al. Layoutlm: Pre-training of text and layout for document image understanding. In *KDD*, 2020.

[416] Zhenghai Xue, Longtao Zheng, Qian Liu, Yingru Li, Zejun Ma, and Bo An. Simpletir: End-to-end reinforcement learning for multi-turn tool-integrated reasoning, 2025. Notion Blog.

[417] Sikuan Yan, Xiufeng Yang, Zuchao Huang, Ercong Nie, Zifeng Ding, Zonggen Li, Xiaowen Ma, Hinrich Schütze, Volker Tresp, and Yunpu Ma. Memory-r1: Enhancing large language model agents to manage and utilize memories via reinforcement learning. *arXiv preprint arXiv:2508.19828*, 2025.

[418] Linyi Yang and Yixuan Weng. Researstudio: A human-intervenable framework for building controllable deep-research agents. *arXiv preprint arXiv:2510.12194*, 2025.

[419] Qu Yang, Mang Ye, Zhaohui Cai, Kehua Su, and Bo Du. Composed image retrieval via cross relation network with hierarchical aggregation transformer. *IEEE Transactions on Image Processing*, 32:4543–4554, 2023.

[420] Qu Yang, Mang Ye, Zhaohui Cai, Kehua Su, and Bo Du. Composed image retrieval via cross relation network with hierarchical aggregation transformer. *IEEE Transactions on Image Processing*, 2023.

[421] Qu Yang, Mang Ye, and Bo Du. Emollm: Multimodal emotional understanding meets large language models. *arXiv preprint arXiv:2406.16442*, 2024.

[422] Qu Yang, Qinghongya Shi, Tongxin Wang, and Mang Ye. Uncertain multimodal intention and emotion understanding in the wild. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025.

[423] Wei Yang, Jinwei Xiao, Hongming Zhang, Qingyang Zhang, Yanna Wang, and Bo Xu. Coarse-to-fine grounded memory for llm agent planning. *arXiv preprint arXiv:2508.15305*, 2025.

[424] Yuqing Yang, Ethan Chern, Xipeng Qiu, Graham Neubig, and Pengfei Liu. Alignment for honesty. *arXiv preprint arXiv:2312.07000*, 2023.

[425] Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W Cohen, Ruslan Salakhutdinov, and Christopher D Manning. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. *arXiv preprint arXiv:1809.09600*, 2018.

[426] Shunyu Yao and et al. React: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations*, 2023.

[427] 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. *Advances in neural information processing systems*, 2023.

[428] Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. ReAct: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations*, 2023.

[429] Zijun Yao, Weijian Qi, Liangming Pan, Shulin Cao, Linmei Hu, Weichuan Liu, Lei Hou, and Juanzi Li. Seakr: Self-aware knowledge retrieval for adaptive retrieval augmented generation. *arXiv preprint arXiv:2406.19215*, 2024.

[430] Guo Yi, Cao Nan, Qi Xiaoyu, Li Haoyang, Shi Danqing, Zhang Jing, Chen Qing, and Weiskopf and, Daniel. Urania: Visualizing data analysis pipelines for natural language-based data exploration. *arXiv preprint arXiv:2306.07760*, 2023.

[431] Zhangyue Yin, Qiushi Sun, Qipeng Guo, Jiawen Wu, Xipeng Qiu, and Xuanjing Huang. Do large language models know what they don't know? *arXiv preprint arXiv:2305.18153*, 2023.

[432] Soyoung Yoon, Eunbi Choi, Jiyeon Kim, Hyeongu Yun, Yireun Kim, and Seung-won Hwang. Listt5: Listwise reranking with fusion-in-decoder improves zero-shot retrieval. *arXiv preprint arXiv:2402.15838*, 2024.

[433] Ori Yoran, Tomer Wolfson, Ori Ram, and Jonathan Berant. Making retrieval-augmented language models robust to irrelevant context. *arXiv preprint arXiv:2310.01558*, 2023.

[434] Ori Yoran, Samuel Joseph Amouyal, Chaitanya Malaviya, Ben Bogin, Ofir Press, and Jonathan Berant. Assistantbench: Can web agents solve realistic and time-consuming tasks? *arXiv preprint arXiv:2407.15711*, 2024.

[435] Fei Xu Yu, Gina Adam, Nathaniel D Bastian, and Tian Lan. Optimizing prompt sequences using monte carlo tree search for llm-based optimization. *arXiv preprint arXiv:2508.05995*, 2025.

[436] Hongli Yu, Tinghong Chen, Jiangtao Feng, Jiangjie Chen, Weinan Dai, Qiying Yu, Ya-Qin Zhang, Wei-Ying Ma, Jingjing Liu, Mingxuan Wang, et al. Memagent: Reshaping long-context llm with multi-conv rl-based memory agent. *arXiv preprint arXiv:2507.02259*, 2025.

[437] Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Tiantian Fan, Gaohong Liu, Lingjun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, Wang Zhang, Hang Zhu, Jinhua Zhu, Jiaze Chen, Jiangjie Chen, Chengyi Wang, Honglin Yu, Weinan Dai, Yuxuan Song, Xiang Wei, Haodong Zhou, Jingjing Liu, Wei Ma, Ya-Qin Zhang, Lin Yan, Mu Qiao, Yong-Xu Wu, and Mingxuan Wang. Dapo: An open-source llm reinforcement learning system at scale. *arXiv preprint arXiv:2503.14476*, 2025.

[438] Wenhao Yu, Hongming Zhang, Xiaoman Pan, Kaixin Ma, Hongwei Wang, and Dong Yu. Chain-of-note: Enhancing robustness in retrieval-augmented language models. *arXiv preprint arXiv:2311.09210*, 2023.

[439] Yue Yu, Wei Ping, Zihan Liu, Boxin Wang, Jiaxuan You, Chao Zhang, Mohammad Shoeybi, and Bryan Catanzaro. Rankrag: Unifying context ranking with retrieval-augmented generation in llms. *Advances in Neural Information Processing Systems*, 2024.

[440] Zhouliang Yu, Ruotian Peng, Keyi Ding, Yizhe Li, Zhongyuan Peng, Minghao Liu, Yifan Zhang, Zheng Yuan, Huajian Xin, Wenhao Huang, et al. Formalmath: Benchmarking formal mathematical reasoning of large language models. *arXiv preprint arXiv:2505.02735*, 2025.

[441] Tian Yuan, Cui Weiwei, Deng Dazhen, Yi Xinjing, Yang Yurun, Zhang Haidong, and Wu Yingcai. Chartgpt: Leveraging llms to generate charts from abstract natural language. *arXiv preprint arXiv:2311.01920*, 2023.

[442] Weizhe Yuan, Pengfei Liu, and Graham Neubig. Can we automate scientific reviewing? *Journal of Artificial Intelligence Research*, 2022.

[443] Weizhe Yuan, Richard Yuanzhe Pang, Kyunghyun Cho, Sainbayar Sukhbaatar, Jing Xu, and Jason Weston. Self-rewarding language models. *arXiv preprint arXiv:2401.10020*, 3, 2024.

[444] Munazza Zaib, Wei Emma Zhang, Quan Z Sheng, Adnan Mahmood, and Yang Zhang. Conversational question answering: A survey. *Knowledge and Information Systems*, 64(12):3151–3195, 2022.

[445] Weihao Zeng, Can Xu, Yingxiu Zhao, Jian-Guang Lou, and Weizhu Chen. Automatic instruction evolving for large language models. *arXiv preprint arXiv:2406.00770*, 2024.

[446] Xinyang Zhai and et al. Siglip: Scaling up visual pre-training with semantics-aware contrastive learning. *arXiv preprint arXiv:2303.15343*, 2023.

[447] Dingchu Zhang, Yida Zhao, Jialong Wu, Baixuan Li, Wenbiao Yin, Liwen Zhang, Yong Jiang, Yufeng Li, Kewei Tu, Pengjun Xie, et al. Evolvesearch: An iterative self-evolving search agent. *arXiv preprint arXiv:2505.22501*, 2025.

[448] Gaoke Zhang, Bo Wang, Yunlong Ma, Dongming Zhao, and Zifei Yu. Multiple memory systems for enhancing the long-term memory of agent. *arXiv preprint arXiv:2508.15294*, 2025.

[449] Guibin Zhang, Luyang Niu, Junfeng Fang, Kun Wang, Lei Bai, and Xiang Wang. Multi-agent architecture search via agentic supernet. *arXiv preprint arXiv:2502.04180*, 2025.

[450] Hanning Zhang, Shizhe Diao, Yong Lin, Yi Fung, Qing Lian, Xingyao Wang, Yangyi Chen, Heng Ji, and Tong Zhang. R-tuning: Instructing large language models to say 'i don't know'. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2024.

[451] Jiaxin Zhang, Zhuohang Li, Kamalika Das, Bradley A Malin, and Sricharan Kumar. Sac3: Reliable hallucination detection in black-box language models via semantic-aware cross-check consistency. *arXiv preprint arXiv:2311.01740*, 2023.

[452] Liang Zhang, Anwen Hu, Haiyang Xu, Ming Yan, Yichen Xu, Qin Jin, Ji Zhang, and Fei Huang. Tinychart: Efficient chart understanding with visual token merging and program-of-thoughts learning. *arXiv preprint arXiv:2404.16635*, 2024.

[453] Qingjie Zhang, Yujia Fu, Yang Wang, Liu Yan, Tao Wei, Ke Xu, Minlie Huang, and Han Qiu. On the self-awareness of large reasoning models' capability boundaries. *arXiv preprint arXiv:2509.24711*, 2025.

[454] Tianyi Zhang*, Varsha Kishore*, Felix Wu*, Kilian Q. Weinberger, and Yoav Artzi. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=SkeHuCVFDr.

[455] Weinan Zhang, Junwei Liao, Ning Li, Kounianhua Du, and Jianghao Lin. Agentic information retrieval. *arXiv preprint arXiv:2410.09713*, 2024.

[456] Weizhi Zhang, Yangning Li, Yuan-Qi Bei, Junyu Luo, Guancheng Wan, Liangwei Yang, Chenxuan Xie, Yuyao Yang, Wei-Chieh Huang, Chunyu Miao, Henry Peng Zou, Xiao Luo, Yusheng Zhao, Yankai Chen, Chunkit Chan, Peilin Zhou, Xinyang Zhang, Chenwei Zhang, Jingbo Shang, Ming Zhang, Yangqiu Song, Irwin King, and Philip S. Yu. From web search towards agentic deep research: Incentivizing search with reasoning agents. *arXiv preprint arXiv:2506.18959*, 2025.

[457] Weizhi Zhang, Xinyang Zhang, Chenwei Zhang, Liangwei Yang, Jingbo Shang, Zhepei Wei, Henry Peng Zou, Zijie Huang, Zhengyang Wang, Yifan Gao, et al. Personaagent: When large language model agents meet personalization at test time. *arXiv preprint arXiv:2506.06254*, 2025.

[458] 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. *arXiv preprint arXiv:2508.12752*, 2025.

[459] Yiming Zhang, Harshita Diddee, Susan Holm, Hanchen Liu, Xinyue Liu, Vinay Samuel, Barry Wang, and Daphne Ippolito. Noveltybench: Evaluating language models for humanlike diversity. *arXiv preprint arXiv:2504.05228*, 2025.

[460] Yu Zhang, Xiusi Chen, Bowen Jin, Sheng Wang, Shuiwang Ji, Wei Wang, and Jiawei Han. A comprehensive survey of scientific large language models and their applications in scientific discovery. *arXiv preprint arXiv:2406.10833*, 2024.

[461] Yuge Zhang, Qiyang Jiang, Xingyu Han, Nan Chen, Yuqing Yang, and Kan Ren. Benchmarking data science agents. *arXiv preprint arXiv:2402.17168*, 2024.

[462] 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 Transactions on Information Systems*, 2024.

[463] Zeyu Zhang, Quanyu Dai, Rui Li, Xiaohe Bo, Xu Chen, and Zhenhua Dong. Learn to memorize: Optimizing llm-based agents with adaptive memory framework. *arXiv preprint arXiv:2508.16629*, 2025.

[464] Zhilin Zhang, Xiang Zhang, Jiaqi Wei, Yiwei Xu, and Chenyu You. Postergen: Aesthetic-aware paper-to-poster generation via multi-agent llms. *arXiv preprint arXiv:2508.17188*, 2025.

[465] Zijing Zhang, Ziyang Chen, Mingxiao Li, Zhaopeng Tu, and Xiaolong Li. Rlvmr: Reinforcement learning with verifiable meta-reasoning rewards for robust long-horizon agents. *arXiv preprint arXiv:2507.22844*, 2025.

[466] Ziyin Zhang, Jiahao Xu, Zhiwei He, Tian Liang, Qiuzhi Liu, Yansi Li, Linfeng Song, Zhenwen Liang, Zhuosheng Zhang, Rui Wang, et al. Deeptheorem: Advancing llm reasoning for theorem proving through natural language and reinforcement learning. *arXiv preprint arXiv:2505.23754*, 2025.

[467] Andrew Zhao, Yiran Wu, Yang Yue, Tong Wu, Quentin Xu, Matthieu Lin, Shenzhi Wang, Qingyun Wu, Zilong Zheng, and Gao Huang. Absolute zero: Reinforced self-play reasoning with zero data. *arXiv preprint arXiv:2505.03335*, 2025.

[468] Qingfei Zhao, Ruobing Wang, Dingling Xu, Daren Zha, and Limin Liu. R-search: Empowering llm reasoning with search via multi-reward reinforcement learning. *arXiv preprint arXiv:2506.04185*, 2025.

[469] Qinlin Zhao, Jindong Wang, Yixuan Zhang, Yiqiao Jin, Kaijie Zhu, Hao Chen, and Xing Xie. Competeai: Understanding the competition dynamics in large language model-based agents. *arXiv preprint arXiv:2310.17512*, 2023.

[470] Shengming Zhao, Yuheng Huang, Jiayang Song, Zhijie Wang, Chengcheng Wan, and Lei Ma. Towards understanding retrieval accuracy and prompt quality in rag systems. *arXiv preprint arXiv:2411.19463*, 2024.

[471] Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. A survey of large language models. *arXiv preprint arXiv:2303.18223*, 2023.

[472] Wenting Zhao, Nan Jiang, Celine Lee, Justin T Chiu, Claire Cardie, Matthias Gallé, and Alexander M Rush. Commit0: Library generation from scratch. *arXiv preprint arXiv:2412.01769*, 2024.

[473] Zheng Zhao, Clara Vania, Subhradeep Kayal, Naila Khan, Shay B Cohen, and Emine Yilmaz. Personalens: A benchmark for personalization evaluation in conversational ai assistants. *arXiv preprint arXiv:2506.09902*, 2025.

[474] Zhengyi Zhao, Shubo Zhang, Yiming Du, Bin Liang, Baojun Wang, Zhongyang Li, Binyang Li, and Kam-Fai Wong. Eventweave: A dynamic framework for capturing core and supporting events in dialogue systems. *arXiv preprint arXiv:2503.23078*, 2025.

[475] Cheng Zhen, Ervine Zheng, Jilong Kuang, and Geoffrey Jay Tso. Enhancing llm-as-a-judge through active-sampling-based prompt optimization. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics*, 2025.

[476] Danna Zheng, Mirella Lapata, and Jeff Z Pan. Long-form information alignment evaluation beyond atomic facts. *arXiv preprint arXiv:2505.15792*, 2025.

[477] Hanwen Zheng, Sijia Wang, Chris Thomas, and Lifu Huang. Advancing chart question answering with robust chart component recognition. *arXiv preprint arXiv:2407.21038*, 2024.

[478] Hao Zheng, Xinyan Guan, Hao Kong, Jia Zheng, Weixiang Zhou, Hongyu Lin, Yaojie Lu, Ben He, Xianpei Han, and Le Sun. Pptagent: Generating and evaluating presentations beyond text-to-slides. *arXiv preprint arXiv:2501.03936*, 2025.

[479] Hao Zheng, Xinyan Guan, Hao Kong, Jia Zheng, Weixiang Zhou, Hongyu Lin, Yaojie Lu, Ben He, Xianpei Han, and Le Sun. Pptagent: Generating and evaluating presentations beyond text-to-slides, 2025.

[480] Kunhao Zheng, Jesse Michael Han, and Stanislas Polu. Minif2f: a cross-system benchmark for formal olympiad-level mathematics. *arXiv preprint arXiv:2109.00110*, 2021.

[481] 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. *arXiv preprint arXiv:2504.03160*, 2025.

[482] Wanjun Zhong, Lianghong Guo, Qiqi Gao, He Ye, and Yanlin Wang. Memorybank: Enhancing large language models with long-term memory. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2024.

[483] Chunting Zhou, Pengfei Liu, Puxin Xu, Srinivasan Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, et al. Lima: Less is more for alignment. *Advances in Neural Information Processing Systems*, 2023.

[484] Denny Zhou, Nathanael Schärli, Le Hou, Jason Wei, Nathan Scales, Xuezhi Wang, Dale Schuurmans, Claire Cui, Olivier Bousquet, Quoc Le, et al. Least-to-most prompting enables complex reasoning in large language models. *arXiv preprint arXiv:2205.10625*, 2022.

[485] Huichi Zhou, Yihang Chen, Siyuan Guo, Xue Yan, Kin Hei Lee, Zihan Wang, Ka Yiu Lee, Guchun Zhang, Kun Shao, Linyi Yang, et al. Agentfly: Fine-tuning llm agents without fine-tuning llms. *arXiv preprint arXiv:2508.16153*, 2025.

[486] Huichi Zhou, Yihang Chen, Siyuan Guo, Xue Yan, Kin Hei Lee, Zihan Wang, Ka Yiu Lee, Guchun Zhang, Kun Shao, Linyi Yang, et al. Memento: Fine-tuning llm agents without fine-tuning llms. *arXiv preprint arXiv:2508.16153*, 2025.

[487] Shuyan Zhou, Frank F Xu, Hao Zhu, Xuhui Zhou, Robert Lo, Abishek Sridhar, Xianyi Cheng, Tianyue Ou, Yonatan Bisk, Daniel Fried, et al. Webarena: A realistic web environment for building autonomous agents. *arXiv preprint arXiv:2307.13854*, 2023.

[488] Shuyan Zhou, Frank F Xu, Hao Zhu, Xuhui Zhou, Robert Lo, Abishek Sridhar, Xianyi Cheng, Tianyue Ou, Yonatan Bisk, Daniel Fried, et al. Webarena: A realistic web environment for building autonomous agents. In *The Twelfth International Conference on Learning Representations*, 2023.

[489] Xiaofeng Zhou, Heyan Huang, and Lizi Liao. Debate, reflect, and distill: Multi-agent feedback with tree-structured preference optimization for efficient language model enhancement. *arXiv preprint arXiv:2506.03541*, 2025.

[490] Yangqiaoyu Zhou, Haokun Liu, Tejes Srivastava, Hongyuan Mei, and Chenhao Tan. Hypothesis generation with large language models. In *Proceedings of the 1st Workshop on NLP for Science (NLP4Science)*, 2024.

[491] 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. *arXiv preprint arXiv:2506.15841*, 2025.

[492] Changtai Zhu, Siyin Wang, Ruijun Feng, Kai Song, and Xipeng Qiu. Convsearch-r1: Enhancing query reformulation for conversational search with reasoning via reinforcement learning. *arXiv preprint arXiv:2505.15776*, 2025.

[493] Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. MiniGPT-4: Enhancing vision-language understanding with advanced large language models. In *The Twelfth International Conference on Learning Representations*, 2024.

[494] Dongsheng Zhu, Weixian Shi, Zhengliang Shi, Zhaochun Ren, Shuaiqiang Wang, Lingyong Yan, and Dawei Yin. Divide-then-aggregate: An efficient tool learning method via parallel tool invocation. *arXiv preprint arXiv:2501.12432*, 2025.

[495] He Zhu, Tianrui Qin, King Zhu, Heyuan Huang, Yeyi Guan, Jinxiang Xia, Yi Yao, Hanhao Li, Ningning Wang, Pai Liu, et al. Oagents: An empirical study of building effective agents. *arXiv preprint arXiv:2506.15741*, 2025.

[496] Kun Zhu, Xiaocheng Feng, Xiyuan Du, Yuxuan Gu, Weijiang Yu, Haotian Wang, Qianglong Chen, Zheng Chu, Jingchang Chen, and Bing Qin. An information bottleneck perspective for effective noise filtering on retrieval-augmented generation. *arXiv preprint arXiv:2406.01549*, 2024.

[497] Lianghui Zhu, Xinggang Wang, and Xinlong Wang. Judgelm: Fine-tuned large language models are scalable judges. *arXiv preprint arXiv:2310.17631*, 2023.

[498] Minjun Zhu, Yixuan Weng, Linyi Yang, and Yue Zhang. DeepReview: Improving LLM-based paper review with human-like deep thinking process. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics*, 2025.

[499] Siyu Zhu, Yanbin Jiang, Hejian Sang, Shao Tang, Qingquan Song, Biao He, Rohit Jain, Zhipeng Wang, and Alborz Geramifard. Planner-r1: Reward shaping enables efficient agentic rl with smaller llms. *arXiv preprint arXiv:2509.25779*, 2025.

[500] Wenhong Zhu, Ruobing Xie, Rui Wang, Xingwu Sun, Di Wang, and Pengfei Liu. Proximal supervised fine-tuning. *arXiv preprint arXiv:2508.17784*, 2025.

[501] Xizhou Zhu, Yuntao Chen, Hao Tian, Chenxin Tao, Weijie Su, Chenyu Yang, Gao Huang, Bin Li, Lewei Lu, Xiaogang Wang, et al. Ghost in the minecraft: Generally capable agents for open-world environments via large language models with text-based knowledge and memory. *arXiv preprint arXiv:2305.17144*, 2023.

[502] Yichen Zhu, Ning Liu, Zhiyuan Xu, Xin Liu, Weibin Meng, Louis Wang, Zhicai Ou, and Jian Tang. Teach less, learn more: On the undistillable classes in knowledge distillation. *Advances in Neural Information Processing Systems*, 35:32011–32024, 2022.

[503] Yutao Zhu, Huaying Yuan, Shuting Wang, Jiongnan Liu, Wenhan Liu, Chenlong Deng, Haonan Chen, Zheng Liu, Zhicheng Dou, and Ji-Rong Wen. Large language models for information retrieval: A survey. *arXiv preprint arXiv:2308.07107*, 2023.

[504] Zeyu Zhu, Kevin Qinghong Lin, and Mike Zheng Shou. Paper2video: Automatic video generation from scientific papers. *arXiv preprint arXiv:2510.05096*, 2025.

[505] Shengyao Zhuang, Xueguang Ma, Bevan Koopman, Jimmy Lin, and Guido Zuccon. Rank-r1: Enhancing reasoning in llm-based document rerankers via reinforcement learning. *arXiv preprint arXiv:2503.06034*, 2025.

[506] Zhenzhen Zhuang, Jiandong Chen, Hongfeng Xu, Yuwen Jiang, and Jialiang Lin. Large language models for automated scholarly paper review: A survey. *Information Fusion*, page 103332, 2025.

[507] Tang Zineng, Yang Ziyi, Zhu Chenguang, Zeng Michael, and Bansal Mohit. Any-to-any generation via composable diffusion. *arXiv preprint arXiv:2305.11846*, 2023.