

LLM×MapReduce-V3: Enabling Interactive In-Depth Survey Generation through a MCP-Driven Hierarchically Modular Agent System

Yu Chao^{1*} Siyu Lin^{2*} Xiaorong Wang³ Zhu Zhang¹ Zihan Zhou³

Haoyu Wang⁴ Shuo Wang^{1†} Jie Zhou³ Zhiyuan Liu^{1†} Maosong Sun^{1†}

¹Dept. of Comp. Sci. & Tech., Institute for AI, BNRist Center, Tsinghua University

²Peking University ³Modelbest Inc. ⁴Nanyang Technological University

Abstract

We introduce **LLM×MapReduce-V3**, a hierarchically modular agent system designed for long-form survey generation. Building on the prior work, **LLM×MapReduce-V2**, this version incorporates a multi-agent architecture where individual functional components, such as skeleton initialization, digest construction, and skeleton refinement, are implemented as independent model-context-protocol (MCP) servers. These atomic servers can be aggregated into higher-level servers, creating a hierarchically structured system. A high-level planner agent dynamically orchestrates the workflow by selecting appropriate modules based on their MCP tool descriptions and the execution history. This modular decomposition facilitates human-in-the-loop intervention, affording users greater control and customization over the research process. Through a multi-turn interaction, the system precisely captures the intended research perspectives to generate a comprehensive skeleton, which is then developed into an in-depth survey. Human evaluations demonstrate that our system demonstrates strong performance in both content depth and length, highlighting the strength of MCP-based modular planning.¹

1 Introduction

Generating high-quality long-form survey articles poses significant challenges to AI Agent systems (Que et al., 2024; Wu et al., 2025; Wang et al., 2024b). First, the system must effectively aggregate and synthesize information from a large collection of reference materials (Nakano et al., 2022; Asai et al., 2024; Baek et al., 2025), which span various subdomains and perspectives. Second, it requires the ability to construct a coherent and comprehensive skeleton that guides the organization

of content at a global level (Wang et al., 2024a; Wen et al., 2025). Third, as the upstream component, reference exploitation exerts a significant influence (Elovic, 2023), thus many AI-based survey generation tools have begun to focus on improving the quality of search results before the writing stage begins. Addressing these challenges demands controllable mechanisms for information acquisition, structural planning, and content generation.

To address these challenges, we propose **LLM×MapReduce-V3**, an interactive and self-organized modular agents system for long-form survey generation. Our system builds upon the design of **LLM×MapReduce-V2** (Wang et al., 2025), but extends it in several important ways to enable modularity, adaptability, and dynamic planning (Qiu et al., 2025b,a). At the core of our approach is the use of the model context protocol (MCP), a standardized function-calling mechanism that allows tools and modules to be composed as independent MCP servers (Basheer, 2025; Pesenzen, 2025). We propose a multi-stage workflow of document digestion, skeleton construction and refinement, and survey writing. Building upon this foundation, we re-architect the system into a multi-agent paradigm, wherein core functionalities are decomposed across specialized agents, and the algorithmic workflow is encapsulated as a suite of tools in MCP servers (Anthropic, 2024) to facilitate agent-level invocation and coordination.

A key improvement in our system lies in pipeline flexibility with an agent planner (CrewAIInc, 2024; Chen et al., 2024). Rather than following a fixed control flow, the agent receives a set of available MCP tools along with previous outputs, and dynamically selects the next module to invoke. This enables non-linear, adaptive workflows tailored to the specific needs of each writing task. To support user intent alignment, we also introduce **human-in-the-loop** interaction. Users begin by providing a target topic and basic writing instructions. The

*Equal contribution.

†Corresponding authors.

¹The code is publicly available at <https://github.com/thunlp/LLMxMapReduce>.

system then engages in a multi-turn dialogue to identify the user’s preferred and fine-grained research perspectives.

To thoroughly examine the performance of LLM \times MapReduce-V3, we conduct a human evaluation to assess system effectiveness. Domain experts compare the output of our system with those from other popular deep research systems across multiple topics. The results suggest that our system generates more informative outlines and higher-quality, more in-depth survey articles. Our main contributions are summarized as follows:

- **Methodological Innovation:** We propose the first MCP-based modular agent system for academic survey generation, enabling unprecedented customization and institutional integration while maintaining survey quality standards.
- **Architectural Advancement:** We propose a dynamic, LLM-driven planner that supports multi-stage module orchestration for adaptive, non-static workflows.
- **User-Centric Design:** We design a human-in-the-loop interaction framework that ensures generated surveys align with user expertise and research perspectives, bridging the gap between automation and scholarly rigor.

2 Related Work

2.1 AI-Powered Automated Research

Early automated research systems focused on web-based information retrieval, with WebGPT (Nakano et al., 2022) and Self-RAG (Asai et al., 2024) pioneering LLM-based web browsing through human feedback or self-reflection mechanisms for adaptive retrieval and generation. Recent advances include sophisticated autonomous agents like GPT-Researcher (Elovic, 2023) with dual-agent architectures, ResearchAgent (Baek et al., 2025) for iterative research idea generation through multi-agents collaboration, and products including Perplexity Deep Research (AI, 2024) and ChatGPT Deep Research (OpenAI, 2024). While some closed-source deep research systems can produce valuable results, we contend that flexible user involvement is critical for a truly practical system. We therefore propose an open-source, hierarchically modular system specifically designed to facilitate and support human intervention throughout the research process.

2.2 Survey Generation

LLM-driven survey generation has emerged with AutoSurvey (Wang et al., 2024a), which introduced a four-stage methodology that addresses context limitations and evaluation challenges, while InteractiveSurvey (Wen et al., 2025) advances through personalized, interactive generation with continuous user customization of reference categorization and content synthesis. SurveyX (Liang et al., 2025), on the other hand, focuses on extracting topics and content by pre-organizing the literature into the form of an attribute tree. Despite these advances, existing systems often lock users into a rigid, “all-or-nothing” paradigm. Stemming from inflexible integration approaches, they lack the necessary interfaces for iterative refinement and specialized customization. In response, we adopt MCP to modularize our algorithmic workflow, enabling the agents to iteratively select and invoke appropriate tools for survey generation.

2.3 Model Context Protocol and Modular Self-Organized Agent System

MCP (Anthropic, 2024) establishes open standards for connecting AI assistants to diverse tool sources through a unified client-server architecture. Recent applications demonstrate its potential for multi-agents intelligence (Basheer, 2025) and scalable systems (Pesenzon, 2025). Alita (Qiu et al., 2025b) leverages MCP to autonomously construct and reuse external capabilities through task-related protocols, achieving scalable agentic reasoning with minimal pre-definition. AgentDistill (Qiu et al., 2025a) enables training-free knowledge transfer via reusable MCP boxes, allowing smaller agents to achieve performance comparable to large LLM systems. These MCP-based self-evolution approaches demonstrate the its potential for adaptive agent systems. Our work first introduces the MCP-based hierarchically modular survey generation system, combining protocol standardization with specialized academic optimizations to enable unprecedented user customization and institutional integration while maintaining the quality of survey paper generation.

3 Hierarchically Modular Agent System

Our system employs a multi-agent paradigm where specialized agents handle distinct phases of survey generation, each independently connecting to both algorithm-internal and user-provided MCP servers.

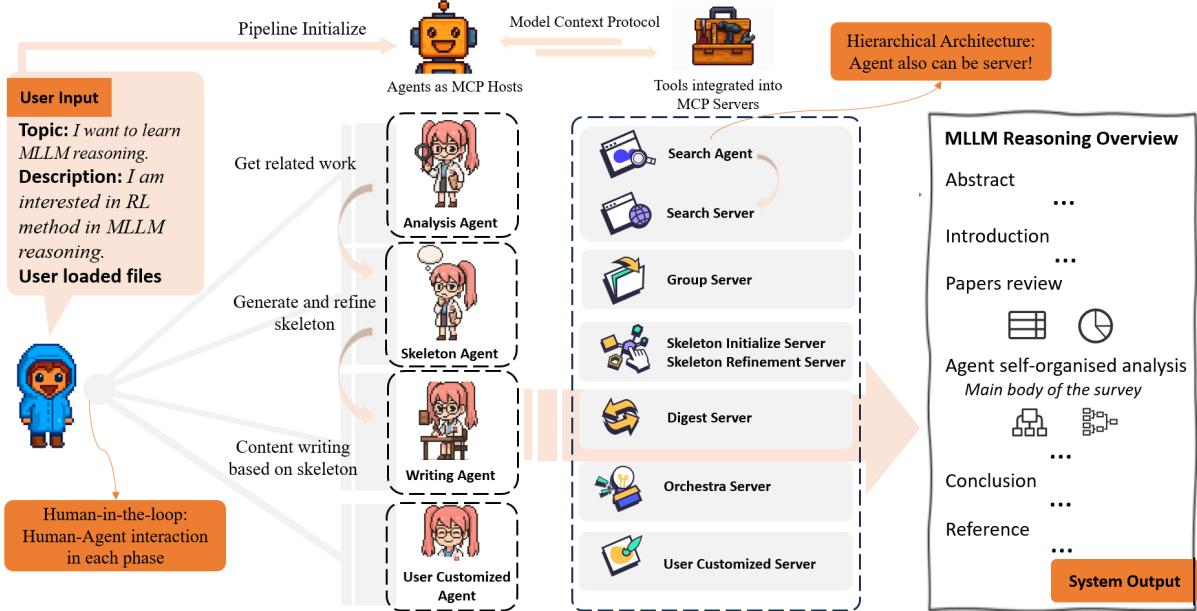


Figure 1: Our agent-server ecosystem pipeline. Users begin by specifying a topic, optionally adding detailed descriptions or uploading documents. The Analysis Agent interprets user intent and coordinates with the Search Agent to retrieve and organize relevant literature. The Skeleton Agent then generates and refines an outline, which is used by the Writing Agent to complete the paper.

3.1 System Design

Let $\mathcal{A} = \{A_1, A_2, A_3\}$ denote the set of specialized agents, where A_1, A_2, A_3 denote the Analysis Agent, Skeleton Agent, and the Writing Agent, respectively. The MCP server ecosystem can be represented as $\mathcal{S} = \{S_1, S_2, S_3, S_4, S_5, S_6, S^*\}$, including the Search Server, Group Server, Skeleton Initialization Server, Digest Server, Skeleton Refinement Server, Orchestra Server, and user customized servers (i.e., S^*).

At each tool-calling round, the connection between the Agent and the Server is defined as \mathcal{E} , which is determined based on the previous output and the Agent's current plan.

$$\mathcal{E} = \text{MCP}(A_i(\text{output}_{i-1}, \text{plan}), \phi(A_i))$$

The agent-server mapping is defined as:

$$\phi : \mathcal{A} \rightarrow 2^{\mathcal{S}}$$

where $\phi(A_i)$ specifies the server subset accessible to agent A_i .

The system architecture forms a directed graph $G = (\mathcal{A}, \mathcal{E}, \mathcal{S})$ where $\text{MCP}(\mathcal{A}_i, \mathcal{S}_i)$ represents agents communicate through standardized MCP interfaces.

Each server $S_i \in \mathcal{S}$ exposes a tool collection $\mathcal{T}(S_i) = \{t_{i,1}, t_{i,2}, \dots, t_{i,k_i}\}$ through standardized

MCP protocols. Tool invocation is formalized as:

$$\text{invoke} : \mathcal{A} \times \mathcal{T} \times \mathcal{I} \rightarrow \mathcal{O}$$

where \mathcal{I} and \mathcal{O} represent input and output spaces respectively.

3.1.1 Analysis Agent

The Analysis Agent orchestrates the initial literature processing and manages interaction with the Search Agent. Given a user-defined research topic and initial description, it first explores users' un-written needs by multi-turn dialogue with users, thus enhance both retrieval width and depth.

It then invokes the search agent for references acquisition and integrates materials uploaded by the user. Subsequently, the agent applies clustering algorithms from the group server to group the references by thematic relevance and methodological similarity, constructing a structured representation of the retrieve result. Finally, the agent initializes a tree graph to store the grouped references, and all downstream data structures are organized by this tree.

3.1.2 Skeleton Agent

The Skeleton Agent is responsible for constructing a globally organized and content-aware skeleton, serving as the structural backbone of the generated survey article. It operates under the coordination

of an orchestration module and proceeds through three main stages: *Skeleton Initialization*, *Digest Construction*, and *Skeleton Refinement*.

3.1.3 Writing Agent

The Writing Agent executes the final content synthesis, transforming the refined skeleton into coherent survey sections. This agent integrates literature digests, maintains citation consistency, and ensures academic writing standards while preserving the logical flow established by the skeleton structure.

To enhance generation quality, users may customize the writing agent to adapt to varying formatting specifications, support figure generation, and address other individualized requirements. Here we implement a dedicated figure server to support the generation of Mermaid-style diagrams.

3.2 External and Replaceable Agents

3.2.1 Search Agent

The Search Agent provides literature retrieval capabilities through the interface. This agent can be seamlessly replaced with alternative search implementations, including domain-specific databases, institutional repositories, or specialized academic search engines.

In parallel, the agent is also capable of acting as an MCP server. In our implementation, the Search Agent is supported by a dedicated search server, which integrates four core tools: query generation, web retrieval, crawling, and similarity analysis. This high-level server operates under the coordination of the Analysis Agent.

3.2.2 Other Agents

The system’s extensibility is reflected in its support for user-defined agents—such as academic critics, formatting agent, and domain-specific analyzers—enabling institutional customization and specialized workflows while preserving overall coherence through standardized interfaces.

4 MCP Implementation Framework

Our implementation leverages MCP’s client-server architecture to encapsulate core functionalities as reusable, composable modules. Each functional component operates as an independent MCP server, exposing standardized tool interfaces for agent invocation.

4.1 Native Server Construction

We re-organize the procedure proposed in LLM \times MapReduce-V2 (Wang et al., 2025) into several MCP servers.

4.1.1 Group Server

The Group Server serves as a content-based organizer that preprocesses the retrieved reference corpus. Before structural planning begins, this module clusters the documents into coherent topical groups. This pre-organization significantly reduces topic fragmentation and provides a more stable input for downstream outline construction. The grouping strategy is guided by both thematic relevance and methodological similarity.

4.1.2 Orchestra Server

The orchestration of the skeleton construction process is managed by the Orchestra Server, which acts as a lightweight planner based on an LLM backbone. At each stage, it takes the current intermediate outputs as input, along with formal descriptions of the available servers. Based on these inputs, it generates next-step instructions. This centralized coordination allows the system to adaptively guide the overall skeleton-building process, ensuring alignment between user intent, module capability, and evolving outline structure. The Orchestra Server thus serves as a dynamic planner and command generator that orchestrates multi-module collaboration across the pipeline.

4.1.3 Skeleton Initialization Server

Given the refined research angles and grouped references provided by the Analysis Agent, the Skeleton Initialization Server constructs a high-level section-wise outline. This initial structure serves as a coarse-grained scaffold, segmenting the article into major thematic areas that reflect distinct facets of the research topic. Each section is derived by mapping the user-specified focus onto prominent subfields within the references, ensuring both topical coverage and logical segmentation.

4.1.4 Digest Server

The Digest Construction Server enhances the initial skeleton by generating content aware revision signals derived from the reference documents. It includes two steps: firstly, for each reference document, the system prompts an LLM to generate a brief summary along with suggestions for improving the current outline. These digests reflect how

individual sources align or conflict with the existing skeleton. Secondly, the system aggregates all digests and suggestions, merges redundant feedback, and synthesizes a consolidated revision plan. This output serves as a high-level guide for improving the outline's coverage, organization, and alignment with the source materials.

4.1.5 Skeleton Refine Server

Skeleton Refinement Server applies an iterative multi-layer convolution-inspired process to optimize the structure for coherence, consistency, and informativeness. This refinement mechanism operates both within sections (to enhance semantic alignment among digests) and across sections (to improve global representation and eliminate redundancy). This process simulates the multi-layer grouped convolution in Convolutional Neural Networks (CNN), which effectively integrates intra-group information and expand the contextual "receptive field" during information aggregation through a multi-layer iterative method. When the references are effectively aggregated, Skeleton Refine Server produces an in-depth and fine-grained survey skeleton, which also determines the structural integrity of the subsequent survey writing.

4.1.6 Iterative Refinement Through Multi-turn Tool-use

The refinement process is conceptualized as a multi-turn, tool-based self-evolution framework, wherein the agent acts as a dynamic coordinator. This coordination is guided by intermediate outputs and, when available, user feedback.

The Orchestra Server implements a planning function:

$$\pi : \mathcal{H} \times \mathcal{C} \rightarrow \mathcal{T}^*$$

where \mathcal{H} denotes execution history, \mathcal{C} represents current context, and \mathcal{T}^* is the space of tool sequences.

At each decision point t , given skeleton state $x^{(t)} \in \mathcal{X}$ and history $h^{(t)} \in \mathcal{H}$, the agent selects action:

$$u^{(t)} = \pi(x^{(t)}, h^{(t)})$$

The refinement process operates through iterative state transitions:

$$x^{(t+1)} = f(x^{(t)}, u^{(t)})$$

Through this mechanism, the system progressively refines the survey skeleton and content analysis, enabling adaptive improvement across multiple iterations.

At each refinement iteration, skeleton agent consults the orchestra server to determine the optimal sequence of actions based on current skeleton state and available feedback. The Orchestra Server analyzes the current context and returns structured action plans which includes subsequent MCP server invocations, enabling adaptive workflow management based on real-time assessment of skeleton quality and user requirements.

Skeleton agent employs the Digest Server to generate targeted literature summaries for specific skeleton sections.

These digests are then integrated into the skeleton structure through coordinated calls to the Skeleton Server, which maintains structural consistency while incorporating new content elements.

4.2 Agent Integration and Replaceable Invocation

Our orchestration mechanism treats each MCP server as a callable tool, associated with relevant metadata, which allows the central planner to make informed decisions regarding the sequencing of modules.

Additionally, the system supports user-defined extensions, enabling the integration of custom MCP servers or external servers tailored to specific agents or tasks.

5 Human-Agent Interaction

Our system incorporates human feedback at key decision points to enhance survey quality and ensure alignment with user objectives. The interaction mechanism unfolds in structured phases that iteratively capture user input and refine system outputs.

5.1 Consensus Achievement

Topic Consensus Phase begins with user-defined topics and goals, followed by LLM-generated topic analysis and expansion. Through multi-turn dialogue, the system clarifies the research scope, identifies critical perspectives, and develops search strategies. This phase continues until a consensus is reached between the user and the AI on the survey's focus and coverage.

5.2 Feedback Integration

Outline Refinement Phase presents the generated survey skeleton for user review and modification. Users may request structural adjustments, section reordering, or changes in content emphasis. The system processes this feedback via the skeleton

Legend: ✓ = Full Support; ~ = Limited Support; ✗ = No Support

System	User Interaction	Modular Design	MCP Support	Custom Tools	Survey Focus	Open Source
Commercial Platforms						
Perplexity DR (AI, 2024)	✓	✗	✗	✗	✗	✗
OpenAI DR (OpenAI, 2024)	✓	✗	✗	✗	✗	✗
Gemini DR (Citron, 2024)	✓	✗	✗	✗	✗	✗
Manus AI (Monica (Butterfly Effect AI), 2025)	✓	✗	✗	✗	✗	✗
Deep Research Systems						
WebGPT (Nakano et al., 2022)	✓	~	✗	✗	✗	✗
GPT-Researcher (Elovic, 2023)	~	✓	✓	✓	✗	✓
ResearchAgent (Baek et al., 2025)	✓	✓	✗	~	✗	~
Self-RAG (Asai et al., 2024)	✗	✓	✗	✗	✗	✓
CoSearchAgent (Gong et al., 2024)	~	~	✗	✗	✗	✗
OpenResearcher (Zheng et al., 2024)	✗	~	✗	✗	✗	~
Search-o1 (Li et al., 2025)	✗	~	✗	✗	✗	✓
Agent-R1 (Cheng et al., 2025)	✗	~	✗	✗	✗	✓
Multi-Agent Frameworks						
CrewAI (CrewAIInc, 2024)	~	✓	✓	✓	✗	✓
AutoAgent (Chen et al., 2024)	✗	✓	✗	~	✗	✓
Alita (Qiu et al., 2025b)	✗	✓	✓	✓	✗	✓
Survey Generation Systems						
AutoSurvey (Wang et al., 2024a)	✗	✗	✗	✗	✓	✓
InteractiveSurvey (Wen et al., 2025)	✓	~	✗	✗	✓	✓
SurveyX (Liang et al., 2025)	~	~	✗	✗	✓	✓
LLM×MapReduce-V3 (Ours)	✓	✓	✓	✓	✓	✓

Table 1: Comparison of Survey Generation and Deep Research Systems

server, ensuring coherence while accommodating user preferences.

Quality Assurance Integration enables users to assess intermediate outputs at each stage, providing feedback that influences subsequent module executions. This ensures the final survey reflects both user expertise and the system’s ability to synthesize comprehensive literature.

6 Comparison and Evaluation

As shown in Table 1, our LLM×MapReduce-V3 system is uniquely equipped to meet the comprehensive demands of academic survey generation. While existing solutions excel in specific areas, commercial platforms in user interaction, research systems in modularity, and frameworks in MCP integration, none provide an all-encompassing solution. Our system is the first to holistically integrate comprehensive user interaction, a modular design, MCP standardization, custom tool integration, and survey-specific optimizations.

The evaluation results in Table 2 indicate that Manus AI just generates preliminary outlines and contents lacks depth. Gemini DeepResearch provides superior depth, structural coherence, and fluency. Compared to other systems, ours provides broader coverage, particularly in literature reviews, and produces significantly longer contexts.

System	Skeleton	Length	Quality
Gemini DR	54.54%	9.09%	42.86%
Manus AI	27.27%	9.09%	0.00%
Our work	18.18%	81.81%	57.14%

Table 2: Human evaluation results. We recruited five human expert reviewers to evaluate articals generated by Gemini DeepResearch, Manus AI, and our system across eleven topics. Reviewers cast their votes based in three criteria: skeleton, length, and quality.

7 Conclusion

LLM×MapReduce-V3 introduces a modular, MCP-based architecture that enables automated research assistance. By supporting open integration of customizable agents and servers, it overcomes the rigidity of traditional closed-agent systems. Through standardized interfaces, our system enables flexible composition of community-developed components to meet diverse research needs. The proposed system also shows strong potential for broader application in knowledge-intensive tasks. With human-in-the-loop design, it achieves better alignment with human objectives. Despite existing challenges, such as fixed workflows and complex data exchange, we advocate for open, adaptable agent ecosystems that evolve with advancing tools and demands.

Ethics Statement

This work focuses on the development of an open-source, modular agent system for academic survey generation. Our system is intended to support and augment human researchers, not to replace them. We emphasize human-in-the-loop design to ensure transparency, user control, and alignment with scholarly standards. No sensitive personal data or copyrighted materials were used in model training or evaluation. All evaluation was conducted with the consent of expert annotators. We acknowledge the potential risks of misuse in automating academic writing and encourage responsible deployment with clear disclosure of AI assistance.

Acknowledgement

This work is supported by the AI9Stars community. We also thank the anonymous reviewers for their insightful suggestions.

References

Perplexity AI. 2024. [Perplexity deep research](#).

Anthropic. 2024. [Introducing the model context protocol](#).

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. 2024. [Self-RAG: Learning to retrieve, generate, and critique through self-reflection](#). In *The Twelfth International Conference on Learning Representations*.

Jinheon Baek, Sujay Kumar Jauhar, Silviu Cucerzan, and Sung Ju Hwang. 2025. [ResearchAgent: Iterative research idea generation over scientific literature with large language models](#). In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 6709–6738, Albuquerque, New Mexico. Association for Computational Linguistics.

Harun Raseed Basheer. 2025. [The model context protocol \(mcp\): A new standard for multi-agent intelligence in ai systems](#).

Guangyao Chen, Siwei Dong, Yu Shu, Ge Zhang, Jaward Sesay, Börje F. Karlsson, Jie Fu, and Yemin Shi. 2024. [AutoAgents: A framework for automatic agent generation](#). In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI-24*, pages 22–30. Main Track.

Mingyue Cheng, Jie Ouyang, Shuo Yu, Ruiran Yan, Yucong Luo, Zirui Liu, Daoyu Wang, Qi Liu, and Enhong Chen. 2025. [Agent-r1: Training powerful llm agents with end-to-end reinforcement learning](#). *Preprint*, arXiv:2511.14460.

Dave Citron. 2024. [Try deep research and gemini 2.0 flash experimental](#). Google Blog.

CrewAIInc. 2024. [Crewai: Framework for orchestrating role-playing, autonomous ai agents](#). <https://github.com/crewAIInc/crewAI>.

Assaf Elovic. 2023. [gpt-researcher](#).

Peiyuan Gong, Jiamian Li, and Jiaxin Mao. 2024. [Cosearchagent: a lightweight collaborative search agent with large language models](#). In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2729–2733.

Xiaoxi Li, Guanting Dong, Jiajie Jin, Yuyao Zhang, Yujia Zhou, Yutao Zhu, Peitian Zhang, and Zhicheng Dou. 2025. [Search-o1: Agentic search-enhanced large reasoning models](#). *Preprint*, arXiv:2501.05366.

Xun Liang, Jiawei Yang, Yezhaozui Wang, Chen Tang, Zifan Zheng, Shichao Song, Zehao Lin, Yebin Yang, Simin Niu, Hanyu Wang, Bo Tang, Feiyu Xiong, Keming Mao, and Zhiyu li. 2025. [Surveyx: Academic survey automation via large language models](#). *Preprint*, arXiv:2502.14776.

Monica (Butterfly Effect AI). 2025. [Manus \(ai agent\)](#). Autonomous artificial intelligence agent.

Reiichiro Nakano, Jacob Hilton, Suchir Balaji, Jeff Wu, Long Ouyang, Christina Kim, Christopher Hesse, Shantanu Jain, Vineet Kosaraju, William Saunders, Xu Jiang, Karl Cobbe, Tyna Eloundou, Gretchen Krueger, Kevin Button, Matthew Knight, Benjamin Chess, and John Schulman. 2022. [Webgpt: Browser-assisted question-answering with human feedback](#). *Preprint*, arXiv:2112.09332.

OpenAI. 2024. [Deep research with chatgpt](#).

Tomer Pesenzon. 2025. [Mcp for ai agents: Enabling modular, scalable agentic systems](#).

Jiahao Qiu, Xinzhe Juan, Yimin Wang, Ling Yang, Xuan Qi, Tongcheng Zhang, Jiacheng Guo, Yifu Lu, Zixin Yao, Hongru Wang, Shilong Liu, Xun Jiang, Liu Leqi, and Mengdi Wang. 2025a. [Agentdistill: Training-free agent distillation with generalizable mcp boxes](#). *Preprint*, arXiv:2506.14728.

Jiahao Qiu, Xuan Qi, Tongcheng Zhang, Xinzhe Juan, Jiacheng Guo, Yifu Lu, Yimin Wang, Zixin Yao, Qihan Ren, Xun Jiang, Xing Zhou, Dongrui Liu, Ling Yang, Yue Wu, Kaixuan Huang, Shilong Liu, Hongru Wang, and Mengdi Wang. 2025b. [Alita: Generalist agent enabling scalable agentic reasoning with minimal predefinition and maximal self-evolution](#). *Preprint*, arXiv:2505.20286.

Haoran Que, Feiyu Duan, Liqun He, Yutao Mou, Wangchunshu Zhou, Jiaheng Liu, Wenge Rong, Zekun Moore Wang, Jian Yang, Ge Zhang, Junran Peng, Zhaoxiang Zhang, Songyang Zhang, and Kai Chen. 2024. [Hellobench: Evaluating long text generation capabilities of large language models](#). *Preprint*, arXiv:2409.16191.

Haoyu Wang, Yujia Fu, Zhu Zhang, Shuo Wang, Zirui Ren, Xiaorong Wang, Zhili Li, Chaoqun He, Bo An, Zhiyuan Liu, and Maosong Sun. 2025. [Llm×mapreduce-v2: Entropy-driven convolutional test-time scaling for generating long-form articles from extremely long resources](#). *Preprint*, arXiv:2504.05732.

Yidong Wang, Qi Guo, Wenjin Yao, Hongbo Zhang, Xin Zhang, Zhen Wu, Meishan Zhang, Xinyu Dai, Min Zhang, Qingsong Wen, Wei Ye, Shikun Zhang, and Yue Zhang. 2024a. [Autosurvey: Large language models can automatically write surveys](#). *Preprint*, arXiv:2406.10252.

Yuxia Wang, Minghan Wang, Muhammad Arslan Manzoor, Fei Liu, Georgi Georgiev, Rocktim Jyoti Das, and Preslav Nakov. 2024b. [Factuality of large language models: A survey](#). *Preprint*, arXiv:2402.02420.

Zhiyuan Wen, Jiannong Cao, Zian Wang, Beichen Guo, Ruosong Yang, and Shuaiqi Liu. 2025. [Interactivesurvey: An llm-based personalized and interactive survey paper generation system](#). *arXiv preprint arXiv:2504.08762*.

Yuhao Wu, Yushi Bai, Zhiqiang Hu, Roy Ka-Wei Lee, and Juanzi Li. 2025. [Longwriter-zero: Mastering ultra-long text generation via reinforcement learning](#). *Preprint*, arXiv:2506.18841.

Yuxiang Zheng, Shichao Sun, Lin Qiu, Dongyu Ru, Cheng Jiayang, Xuefeng Li, Jifan Lin, Binjie Wang, Yun Luo, Renjie Pan, Yang Xu, Qingkai Min, Zizhao Zhang, Yiwen Wang, Wenjie Li, and Pengfei Liu. 2024. [Openresearcher: Unleashing ai for accelerated scientific research](#). *Preprint*, arXiv:2408.06941.