Empowering Real-World: A Survey on the Technology, Practice, and Evaluation of LLM-driven Industry Agents

Yihong Tang, Kehai Chen, Liang Yue, Jinxin Fan, Caishen Zhou, Xiaoguang Li, Yuyang Zhang, Mingming Zhao, Shixiong Kai, Kaiyang Guo, Xingshan Zeng, Wenjing Cun, Lifeng Shang, Min Zhang

Abstract—With the rise of large language models (LLMs), LLM agents capable of autonomous reasoning, planning, and executing complex tasks have become a frontier in artificial intelligence. However, how to translate the research on general agents into productivity that drives industry transformations remains a significant challenge. To address this, this paper systematically reviews the technologies, applications, and evaluation methods of industry agents based on LLMs. Using an industry agent capability maturity framework, it outlines the evolution of agents in industry applications, from "process execution systems" to "adaptive social systems." First, we examine the three key technological pillars that support the advancement of agent capabilities: Memory, Planning, and Tool Use. We discuss how these technologies evolve from supporting simple tasks in their early forms to enabling complex autonomous systems and collective intelligence in more advanced forms. Then, we provide an overview of the application of industry agents in real-world domains such as digital engineering, scientific discovery, embodied intelligence, collaborative business execution, and complex system simulation. Additionally, this paper reviews the evaluation benchmarks and methods for both fundamental and specialized capabilities, identifying the challenges existing evaluation systems face regarding authenticity, safety, and industry specificity. Finally, we focus on the practical challenges faced by industry agents, exploring their capability boundaries, developmental potential, and governance issues in various scenarios, while providing insights into future directions. By combining technological evolution with industry practices, this review aims to clarify the current state and offer a clear roadman and theoretical foundation for understanding and building the next generation of industry agents.

Index Terms—Large Language Models (LLMs), Industry, Agent, Real-world.

I. INTRODUCTION

N recent years, large language models (LLMs) have made groundbreaking progress. Through pre-training on vast amounts of data, they exhibit unprecedented language understanding, generation, and reasoning capabilities [1]–[3]. However, LLMs, as static and state-less predictive models, are mainly limited to processing text input and generating corresponding outputs. They struggle to actively interact with the external world or perform complex tasks that require long-term memory and multi-step operations [4], [5]. To overcome this limitation, researchers have used LLMs as the "brain" to

Yihong Tang, Kehai Chen, Liang Yue, Jinxin Fan, Caishen Zhou and Min Zhang are with the School of Computer Science and Technology, Harbin Institute of Technology, Shenzhen, China, 518055 e-mail: (chenkehai@hit.edu.cn). Xiaoguang Li, Yuyang Zhang, Mingming Zhao, Shixiong Kai, Kaiyang Guo, Xingshan Zeng, Wenjing Cun and Lifeng Shang are with Huawei Technologies Co., Ltd.

build autonomous agents that can perceive the environment, plan, act, and learn from interactions [6]. These LLM-driven agents integrate memory modules, planning algorithms, and tool invocation interfaces, combining the cognitive abilities of LLMs with dynamic interactions in the environment, thus forming the prototype of general agents capable of autonomously achieving open-ended goals.

As general agents transition from theory to practice, their application scenarios inevitably shift from simple, general digital environments to complex, knowledge-intensive, and high-risk industry domains [7]. This gives rise to the concept of "Industry Agents." Industry agents refer to autonomous or semiautonomous systems deployed in specific business contexts that leverage domain knowledge and specialized tools to solve realworld industry problems. For example, Xia et al. demonstrate how LLM agents can orchestrate modular production systems by planning tasks, invoking low-level control interfaces, and interfacing with digital twins [8]. Compared to general-purpose agents, industry agents face more severe challenges. They must not only possess general cognitive abilities but also address industry-specific requirements, such as the high time sensitivity and risks in finance [9], the authoritative knowledge and security compliance in healthcare [10], and the physical constraints and process complexity in manufacturing [11], [12]. The key issue becomes how to integrate general agent frameworks with deep industry expertise, complex business processes, and stringent safety standards, thus transforming the potential of agents into real-world productivity.

Meanwhile, with the rapid development of LLM-based agent research, numerous excellent review papers have emerged, offering valuable perspectives from different dimensions to help us understand the field. Some reviews focus on the core technical modules of intelligent agents. For example, [13] systematically reviews the memory mechanisms of agents; [14] classifies and analyzes planning capabilities; and [15] provides a comprehensive overview of tool learning paradigms and implementations. Additionally, [16] optimizes the information load in the LLM reasoning process from the perspective of context engineering, offering important support for efficient agent interactions. These works lay the foundation for a deeper understanding of the technical details of agents. Other reviews focus on general agent architectures and capabilities. [6], [17] propose general agent frameworks and classify existing architectures. At the same time, works like [18], [19] explore the implementation paths for advanced capabilities

such as reasoning and self-evolution. Notably, [20] presents a modular, brain-inspired view of agent cognitive, perception, and operation modules, while also addressing key topics such as self-enhanced evolution, multi-agent systems, and secure deployment. Additionally, some reviews focus on specific application areas or advanced paradigms. For instance, reviews like [21], [22] delve into the applications of agents in scientific discovery and financial trading. Meanwhile, [23], [24] explore multi-agent systems and the Agentic RAG paradigm. Moreover, [25] provides a comprehensive review of LLM-enabled agent-based modeling and simulation, covering applications in information, physics, social, and hybrid scenarios. [26] focuses on autonomous research agents, proposing a systematic methodology and evaluation blueprint for their construction. Finally, [27] offers a data-centric system review and roadmap for the development of scientific LLMs and agents from the perspective of data and model co-evolution.

Despite these outstanding contributions, there remains a gap in providing a systematic framework that combines technological evolution, application practice, and capability levels, with a focus on industry implementation. To fill this gap, this paper provides a comprehensive review of LLM-based industry agents. Specifically, the review is organized into three main areas: the technological foundations, practical applications, and real-world evaluations of industry agents. First, we delve into the three core technologies supporting agent capabilities: memory, planning, and tool use, and discuss their technological evolution. Then, we present a panoramic view of industry agent applications across various sectors using a five-level maturity framework. Next, we systematically examine evaluation benchmarks and methods for both foundational and specialized industry capabilities, highlighting their limitations. Finally, we focus on the deep challenges faced by industry agents in practice, exploring their bottlenecks, future development, and strategies to address these challenges.

In summary, the contributions of this paper include:

Proposing a Capability Maturity Framework: We introduce an innovative industry agent capability maturity framework that provides a clear metric for assessing and understanding the role and value of agents in various industries.

Linking Technology and Application: We connect the evolution of the three core technologies—memory, planning, and tool use—with capability levels, showing how technological advances drive the progression of application practices.

Focusing on Industry Practices and Evaluations: We systematically review agent applications in key industries and professional evaluation benchmarks, aligning closely with real-world industrial needs and challenges.

With this unique perspective, we aim to bridge the gap between agent applications across various domains, contributing to the maturation and prosperity of agent in the real world.

II. TECHNICAL FOUNDATIONS OF INDUSTRY AGENTS

In recent years, agents built upon LLMs have made significant advancements. Their increasingly sophisticated capabilities in handling complex tasks are steering artificial intelligence research and applications toward higher levels of cognitive intelligence. Early agent research was often limited to specific tasks. In contrast, emerging LLMs, with their robust general language understanding, reasoning, and interaction abilities, have greatly facilitated the emergence of general-purpose agents capable of handling open-domain complex tasks.

Currently, a comprehensive general-purpose agent framework typically relies on three core technical pillars: Memory, Planning, and Tool Use. Memory refers to the ability to encode, store, and retrieve information; Planning involves goal decomposition and the formulation and optimization of action sequences; Tool Use pertains to the ability to invoke external APIs or programs to extend one's capabilities. These three core modules are interwoven and collaborate, forming the foundation for agents to perceive their environment, develop cognition, and take action. This enables agents to evolve from simple instruction executors to autonomous entities capable of continuous interaction with their environment and achieving complex objectives.

However, as agent research increasingly covers real-world scenarios, cognitive bottlenecks in their core architectures have become more apparent. These challenges are deeply reflected in the limitations of the three core capabilities: Memory, Planning, and Tool Use. In the realm of Memory, limited and singular context windows make it difficult for agents to maintain long-term, coherent interaction histories, leading to issues like long-context forgetting. Additionally, how to filter, refine, and form structured, effective memories from vast, noisy, unstructured dynamic environmental information, avoiding information overload and cognitive biases, remains a significant technical bottleneck. In Planning, the high dynamism and uncertainty of the real world render simple planning methods based on static world assumptions ineffective. Agents must possess the robustness to dynamically adjust plans during execution, handle anomalies, and learn from failures, placing high demands on their ability to decompose long-term goals and reason effectively. Regarding Tool Use, as tool libraries grow large and complex, how to accurately select, combine, and invoke appropriate tools to solve problems, as well as how to handle tool execution failures or unexpected results, become critical factors limiting the upper bounds of agent capabilities. These practical technical challenges collectively form a gap between theoretical frameworks and real-world applications.

To systematically analyze how industry agents evolve from simple process automation tools to core systems capable of solving complex domain problems, this review proposes a five-level framework (L1-L5) oriented toward industry application capability maturity. This framework aims to reveal that the transitions at each capability level of industry agents are essentially driven by their evolution in the three core technologies: Memory, Planning, and Tool Use. For instance, the Process Execution System at the L1 level requires only transient memory and fixed linear planning, whereas the Adaptive Social System at the L5 level demands the ability to accumulate evolutionary group memory across generations and the capacity to autonomously generate goals in complex games. The following sections will delve into each of the three core technical modules, analyzing how their technological evolution supports the continuous upgrading of industry agent capabilities,

3

Industry Agent

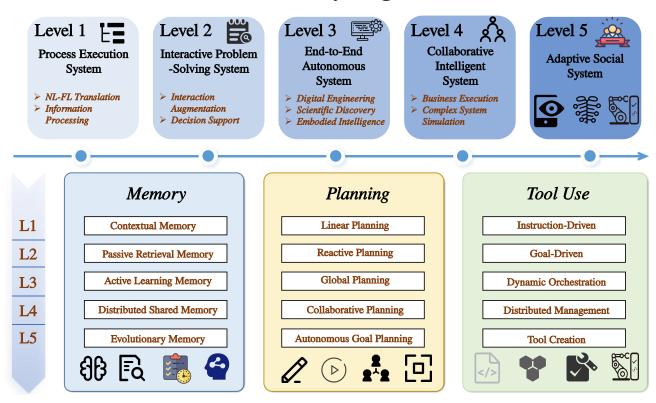


Fig. 1: The framework of industry agent.

thereby laying the groundwork for the development practices of industry agents.

A. Memory Mechanism

Memory is a core component in building advanced artificial intelligence, particularly in LLM-based agents. It enables agents to encode, store, and retrieve historical information, allowing them to transcend the stateless limitations of traditional computational models and exhibit the ability to learn, adapt, and execute complex tasks coherently. For industry agents aimed at solving real-world problems, the complexity and maturity of their memory mechanisms determine the capability levels and application value they can achieve within specific domains. Previous reviews often categorize existing works based on technical implementations, such as the sources, forms, or operations of memory. While such classifications aid in understanding technical details, they do not fully reveal how the evolution of memory mechanisms directly drives the capability transitions of agents. This section analyzes how memory, as a core technology, evolves from supporting basic process execution to underpinning autonomous learning and even collective collaboration in complex systems.

1) From Instantaneous Recording to Passive Retrieval: In the early stages of industry applications, the core value of agents lies in processing explicit instructions and utilizing existing knowledge. Their memory mechanisms primarily focus on two basic functions: recording and querying. This phase

marks the transition of agents from being stateless executors to assistants capable of consulting external notes.

At L1, memory is instantaneous context, essentially working memory. In this phase, agents function as process execution systems, with their memory capabilities mainly supported by the context window of LLMs. This memory is temporary and task-oriented, used solely to maintain information consistency within a single interaction, akin to human short-term working memory. The chain of thought in the ReAct framework exemplifies this instantaneous working memory, explicitly retaining the reasoning process within the context to guide subsequent actions [28]. However, the fundamental limitation of this memory lies in its finiteness and volatility. To extend this limited memory capacity, works like LongChat finetune the base model to better handle and remember longer, complete interactions [29]. Yet, longer contexts may introduce interference. To address this, Memory Sandbox designs an interactive memory management interface, allowing manual removal of irrelevant information before feeding memory into prompts, reflecting an initial attempt at controlling the quality of instantaneous memory [30]. Some systems have designed more structured short-term memories, such as the short-term memory cache in RecAgent [31] for recommendation scenarios and the flash memory or working context in MemGPT [32] for holding recent interaction histories. These can be seen as optimizations of instantaneous memory but do not alter its nature of being forgotten after task completion.

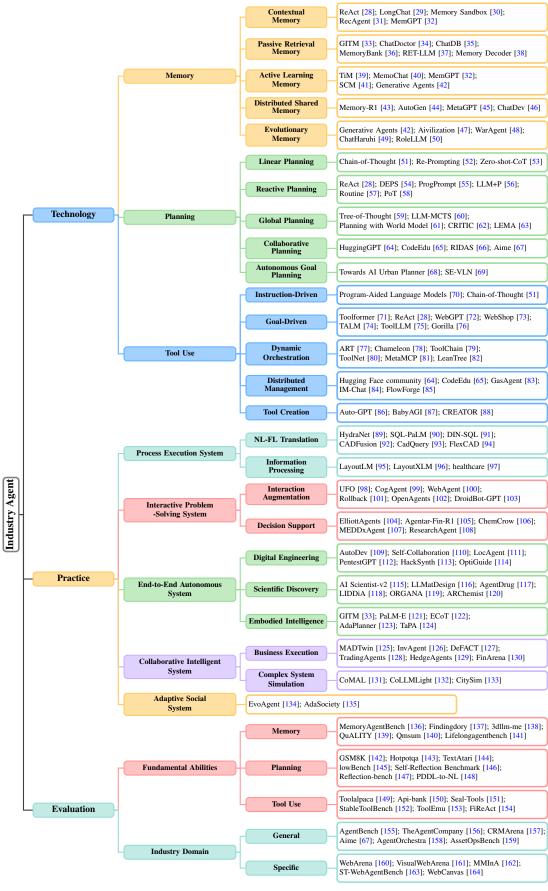


Fig. 2: Taxonomy of industry agents.

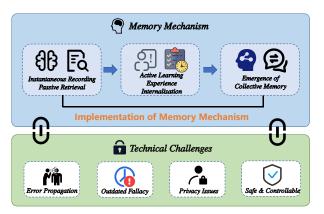


Fig. 3: The evolution of memory mechanisms in industry agents.

At L2, memory evolves to passive retrieval, marking the emergence of long-term memory. When agents function as interactive problem-solving systems, relying solely on context memory is insufficient to handle queries requiring domain knowledge. Therefore, a key evolution of memory mechanisms is the connection to external knowledge bases, achieving a transition from stateless to knowledgeable. The core of memory at this stage is retrieval-augmented generation (RAG), enabling agents to passively retrieve information from external sources to enhance their responses [165]. Works like Toolformer [71] and ToolLLM [75] teach models how to use tools and thousands of real APIs, laying the foundation for acquiring external knowledge. In general scenarios, ReAct demonstrates how to call the Wikipedia API [28] . In industry applications, this retrieval becomes more targeted: in software engineering, CodeAgent [166] designs web search strategies to solve code dependency issues; in gaming, GITM [33] draws knowledge from the online Minecraft Wiki; in scientific computing, ToRA [167] enhances agents' ability to use programmatic tools, while ChemCrow [106] equips LLMs with chemical tools; in professional Q&A, ChatDoctor [34] fine-tunes retrieval models to obtain knowledge from Wikipedia and medical databases. These external knowledge bases constitute the nascent form of stable and reliable long-term memory for agents. Technologies for efficient retrieval are also becoming more diverse. For example, ChatDB [35] generates SQL queries for precise retrieval from structured databases, MemoryBank [36] uses dual-tower dense retrieval models, and RET-LLM [37] employs locality-sensitive hashing for fast reading. Memory Decoder [38] is a plug-and-play pre-trained memory component that mimics the behavior of external non-parametric retrievers to achieve efficient domain adaptation, enhancing performance in specialized fields without modifying the original model parameters. At this stage, although agents possess long-term memory, the memory is static and external. The agents themselves have not learned the knowledge; they are merely more efficient queryers, with their capability boundaries limited by the quality of external knowledge bases and the precision of retrieval algorithms.

2) Active Learning and Experience Internalization: At L3, agents evolve into end-to-end systems capable of completing

complex tasks in a closed-loop manner. This progression signifies a fundamental shift in their memory mechanisms. Memory transitions from passive information storage and retrieval to an active, dynamic system that facilitates learning and experience internalization. Central to this advancement is the agent's acquisition of metacognitive abilities, enabling self-reflection and the extraction of actionable insights from experiences.

A defining characteristic of this stage is the agent's capacity for active learning from its interaction history. These experiences stem from various sources. Some arise from observations during individual task executions, such as in Generative Agents [42], where agents document all their actions in a simulated world, or in Voyager [168], which records reusable code executed successfully in Minecraft. More valuable experiences emerge from analyzing the successes and failures across multiple task attempts. The Reflexion framework introduces verbal reinforcement learning, allowing agents to reflect on their actions and store outcomes in memory [169]. Retroformer enhances this by fine-tuning reflection models for more effective cross-experiment information extraction [170]. ExpeL [171] and Synapse [172] utilize successful task trajectories as exemplars, retrieving similar past cases to guide new tasks. In this phase, agents transcend being mere information consumers; they become producers and distillers of experience. Through reflection, they transform disparate, firstorder interaction records into structured, higher-order action guidelines, laying the cognitive foundation for autonomous improvement and long-term task planning.

Moreover, L3 agents not only retrieve experiences but also internalize them, converting external knowledge into internal memory. Traditional internalization methods involve fine-tuning. For instance, Character-LLM fine-tunes on role-specific data like scripts to embed character traits into model parameters [173]. In specialized domains, Huatuo [174], DoctorGLM [175], and Radiology-GPT [176] fine-tune on Chinese medical knowledge, medical data, and radiology datasets, respectively, endowing agents with professional biomedical knowledge. InvestLM fine-tunes for financial investment capabilities [177]. Beyond fine-tuning, more refined memory editing techniques are emerging, allowing modification of specific knowledge in model parameters without retraining. MEND [178] and KnowledgeEditor [179] train lightweight editing networks to predict parameter updates for rapid factual knowledge modifications. MAC [180] employs meta-learning for online parameterized memory adaptation, while PersonalityEdit [181] enables precise editing of agent personality traits based on psychological theories like the Big Five personality model. Traditional fine-tuning and knowledge editing represent a profound shift from knowledge storage to ability cultivation, serving as technological prerequisites for end-to-end autonomous systems. However, they are often uncontrollable and less interpretable.

In the realm of memory, a more general method of internalization is non-parametric memory management, encompassing:

1) Memory Writing: Efficiently storing raw information into memory. TiM [39] extracts information into entity relationships stored in structured databases; MemoChat [40] summarizes dialogue segments into themes as indexed keys; MemGPT

[32] implements self-guided memory updates; SCM [41] designs memory controllers to determine when to perform write operations. 2) Memory Management and Refinement: Extracting value from vast memories and preventing degradation. Generative Agents [42] generate higher-level abstract thinking through reflection processes; MemoryBank [36] distills daily conversations into high-level daily summaries; GITM [33] summarizes key actions from multiple plans; Voyager [168] optimizes and refines its skill library based on environmental feedback (e.g., code execution success). 3) Memory Reading: Retrieving the most relevant memories based on current tasks. ChatDB [35] generates SQL queries for retrieval; ExpeL [171] uses the Faiss vector library to retrieve the top-K most similar successful trajectories; MPC [182] provides chainof-thought examples to guide models in ignoring irrelevant memories. To integrate these processes, Mem0 adopts a scalable memory center architecture, dynamically extracting, integrating, and retrieving key information from dialogues, significantly enhancing long-dialogue consistency and reducing computational overhead [183]. The upgraded Mem0 represents memory as an entity-relationship graph, capturing complex temporal and multi-hop reasoning logic, particularly suited for cross-session and cross-timepoint conversations. draws inspiration from the Zettelkasten method, employing atomic note structures, multi-dimensional semantic representations, and a combination of vector retrieval and LLM analysis to achieve automatic linking and self-evolution of memory.

3) Emergence of Collective Memory: At L4 and above, industry applications expand from single-agent systems to complex multi-agent collaborations, leading to the emergence of collective memory.

Level 4 memory is distributed and shared. When multiple agents collaborate to achieve a large-scale goal, they must rely on a shared cognitive space, forming their collective memory. Memory-R1 enables large models to actively manage and utilize external memory through alternating operations between two agents [43]. Additionally, multi-agent frameworks such as AutoGen [44], ChatDev [46], and MetaGPT [45] provide implementation examples. In these systems, all agent roles share a unified context, including requirement documents, codebases, and API specifications. For instance, in the ChatDev simulation of software development, each agent stores past dialogues with other roles [46]. In MetaGPT, agents can retrieve historical records from memory to address errors [45]. In broader collaborative scenarios, such as the S3 social network simulation, each agent's memory pool contains diverse user messages to define its identity [184]. In the job simulation MetaAgents, memory continuously enriches through dialogue and reflection [185]. In the code repair scenario RTLFixer, an externally shared database stores compiler errors and expert repair instructions [186]. Shared memory serves as the foundation for efficient collaboration, ensuring all individuals communicate and cooperate based on consistent information, thereby avoiding information silos and cognitive biases. It is a prerequisite for coordinating complex business processes. Despite various multi-agent communication protocols and topologies being proposed, this memory remains synchronous and task-oriented.

Level 5 memory envisions an evolutionary and cultural form. At this level, memory not only shares across agents but also accumulates, solidifies, and evolves over time, forming a culture akin to human societies. It records the group's successful strategies, lessons from failures, and shared values, which can be inherited by newly joined agents. In Generative Agents, information propagation within the agent society demonstrates this primitive form of memory [42]. Aivilization scales this concept to a larger community, constructing a highly realistic virtual society encompassing economics, industry, politics, and social interactions [47]. In specific simulations, such as WarAgent's war simulation, dialogues of participating countries are continuously recorded in memory, shaping their long-term behaviors [48]. In role-playing applications like ChatHaruhi [49] and RoleLLM [50], by injecting role-specific knowledge and plot memories, agents exhibit consistent identities, reflecting micro-level cultural forms. Exploring L5 memory involves considering how to build an agent society capable of self-improvement and sustainable development. This requires memory mechanisms that not only record "what is" but also encapsulate "why it is" and "how it should be," thereby providing a foundation for long-term value alignment and goal evolution within agent communities.

4) Real-World Challenges in Memory Management: Efficient memory mechanisms do not equate to flawless performance. Empirical studies by [187]. reveal that LLM agents exhibit a pronounced experience-following behavior, meaning they tend to replicate past experiences similar to current tasks. This characteristic introduces two primary risks: error propagation, where mistakes in early memories are amplified in subsequent decisions; and misaligned experience replay, where outdated or irrelevant memories negatively interfere with the current task. Their research emphasizes the importance of implementing refined memory addition and deletion strategies to maintain long-term agent robustness. Their research emphasizes the importance of implementing refined memory addition and deletion strategies to maintain long-term agent robustness. Concurrently, Wang et al. systematically identify privacy vulnerabilities within agent memory modules [188]. They introduce the Memory EXTRaction Attack (MEXTRA), demonstrating that even in black-box settings, attackers can extract sensitive user interactions stored in memory through prompt engineering. Their findings underscore the necessity for secure and controllable memory systems, especially in highrisk, high-regulation sectors such as healthcare, finance, and law. These studies collectively highlight a central challenge in memory research: the need to develop memory systems that are not only effective in learning but also secure, controllable, and maintainable.

B. Planning Capability

Planning is a core cognitive ability of an agent. It determines how an agent decomposes abstract goals into a series of executable actions to achieve its intentions in an environment. In the context of industry agents, planning capability is directly related to their autonomy, reliability, and the complexity of problem-solving. A robust planning module enables an agent

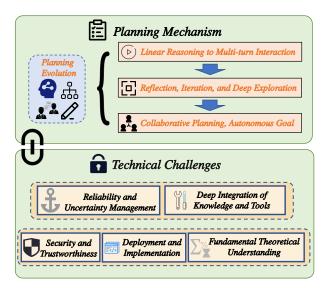


Fig. 4: The evolution of planning capability in industry agents.

not only to understand "what to do" but also to autonomously decide "how to do it," adjusting and optimizing in dynamic environments. This section systematically reviews how planning technologies have evolved from simple task decomposition to complex reflective, collaborative, and generative planning, driving industry agents to create core value at different maturity stages. Each leap in planning capability marks a foundational step toward higher autonomy and intelligence in agents.

1) From Linear Reasoning to Multi-turn Interaction: At the L1-L2 stages, agents primarily serve as human assistants or tools, with their planning capabilities focusing on accurately understanding user instructions and decomposing them into executable steps. At the L1 level, planning is linear instruction decomposition, essentially open-loop planning. The core involves following a relatively fixed, pre-set path to complete tasks. A breakthrough in this stage is the Chainof-Thought (CoT) prompting technique, which guides LLMs to generate intermediate reasoning steps, significantly enhancing their ability to handle complex problems [51]. However, CoT is inherently a linear, one-time generation process, lacking interaction and correction with the environment. Re-Prompting improves this by utilizing precondition error information to re-prompt LLMs, generating executable plans that enhance the plans' executability and semantic correctness [52]. The emergence of Zero-shot-CoT further simplifies this process, requiring only a phrase like "Let's think step by step" to trigger the model's initial reasoning ability [53]. Plan-and-Solve Prompting structures this process by explicitly dividing the task into "plan first, then execute" steps, laying the foundation for more reliable execution [189]. Linear planning enables agents to handle multi-step logical problems but assumes a static environment and flawless initial planning, which often does not hold in the ever-changing real world, leading to reduced robustness.

At the L2 level, planning evolves into reactive planning, achieving closed-loop control. Agents are no longer merely passively decomposing tasks but can interact with the environment or tools and dynamically adjust subsequent steps based

on feedback.

The ReAct framework is a milestone at this stage, decoupling and interleaving reasoning and action, allowing agents to think during execution and act after thinking [28]. This "think-actobserve" cycle forms the core of reactive planning. This mode is particularly crucial in scenarios requiring interaction with external tools. For example, Visual ChatGPT utilizes the ReAct mechanism, using an LLM as the brain to orchestrate a series of visual foundation models to complete image processing tasks [28]. DEPS enhances the task planning capability of multi-task agents in open-world environments by combining descriptions of the planning execution process, self-explanations upon failure, and a trainable goal selector that estimates completion steps to parallel subgoals [54]. To make the planning process more rigorous and predictable, researchers have explored converting natural language planning into more formal languages. PAL [70] and Program-of-Thought Prompting (PoT) [58] guide LLMs to express reasoning processes as executable code, utilizing the determinism of code interpreters to ensure result accuracy. ProgPrompt adopts a similar approach, transforming robotic task planning into function generation problems [55]. Furthermore, to meet the high reliability requirements of industrial applications, a series of works combine LLMs with classical symbolic planners. Frameworks like LLM+P [56], LLM+PDDL [190], and LLM+ASP [191] use LLMs to convert natural language problems into formal representations such as PDDL or ASP, then call external optimization planners to solve them, obtaining optimal and reliable plans. In enterprise environments, the Routine framework achieves stable multistep tool invocation planning by providing clear structures and instructions [57]. In gaming scenarios, Voyager achieves continuous exploration, skill acquisition, and autonomous discovery in a human-free Minecraft environment through automatic curriculum planning [168]. Reactive planning greatly enhances an agent's adaptability in dynamic environments but typically has a localized, short-sighted planning perspective. It excels at "adapting to changes" but struggles with "deep thinking," making it challenging to solve complex problems requiring long-term planning and trade-offs.

2) Global Planning — Reflection, Iteration, and Deep Exploration: As agents advance to the L3 level, becoming "end-to-end autonomous systems," their planning capabilities must address complex, dynamic, and uncertain environments. This necessitates planning processes with nonlinear abilities for deep exploration, self-correction, and continuous learning.

As agents advance to the L3 level, becoming end-to-end autonomous systems, their planning capabilities must address complex, dynamic, and uncertain environments. This necessitates planning processes with nonlinear abilities for deep exploration, self-correction, and continuous learning. Initially, planning evolves from linear chains to tree-like or graph-like explorations. Compared to CoT [51], Tree-of-Thought (ToT) [59] and Graph-of-Thought (GoT) [192] significantly expand the planning exploration space. ToT organizes reasoning paths into a tree structure, allowing agents to explore, evaluate, and even backtrack among multiple potential solutions to select the globally optimal path. GoT further models the thought process as a graph, supporting more complex thought aggregation and

transformation, thereby enhancing the ability to solve intricate problems.

To navigate vast search spaces efficiently, frameworks like LLM-MCTS [60] and Reasoning with Language Model is Planning with World Model (RAP) [61] innovatively utilize LLMs as heuristic functions in Monte Carlo Tree Search (MCTS) [193], guiding the search process to balance exploration and exploitation. By introducing systematic search strategies, agents evolve from greedy decision-makers to ones capable of trade-offs and foresight, which is crucial for solving complex problems with multiple potential paths and pitfalls. Self-reflection and correction become core mechanisms, transforming planning from a one-time process into an iterative optimization cycle. Agents are no longer one-off planners; they possess the ability to learn from experiences and failures.

The Reflexion framework builds upon ReAct by adding a self-reflection loop, enabling agents to analyze failure trajectories, generate textual reflections, and store them in memory to guide subsequent attempts [169]. Self-Refine introduces an iterative optimization process without external training, where agents generate solutions, provide feedback on them, and refine the solutions in subsequent rounds [194]. The CRITIC framework employs external tools, such as knowledge bases and search engines, to verify and critique agent-generated actions, using external feedback for self-correction [62]. The LEMA framework collects erroneous planning samples, utilizes more powerful models for corrections, and fine-tunes the original model with these corrected samples [63].

In more specific applications, such as formal mathematical proofs, the Delta Prover framework iteratively constructs proofs through interactions, reflections, and reasoning between LLMs and the proof environment Lean 4 [195]. In robotics, the Conditional Multi-Stage Failure Recovery framework designs multi-stage failure recovery strategies for embodied agents, enhancing their robustness in executing tasks in real-world environments [196].

Comprehensive and macro-level reflection mechanisms empower agents to draw lessons from errors. This endows planning with resilience, enabling agents to recover from mistakes and continuously improve, which is key to achieving end-to-end autonomy.

3) Collaborative Planning and Autonomous Goal Setting: At L4 and beyond, the scope of planning extends from individual agents to systems composed of multiple agents, broadening the concept of planning from task execution to collaborative strategies and social evolution.

At the L4 level, planning focuses on how multiple agents can develop and execute group plans through communication and negotiation to achieve common objectives. Early explorations, such as HuggingGPT, utilize a LLM as a controller to coordinate multiple models from the Hugging Face Hub to collaboratively complete multimodal tasks [64]. This can be considered a nascent form of collaborative planning. Multiagent collaborative planning demonstrates significant potential across various domains. For instance, CodeEdu [65] and AI-Powered Math Tutoring [197] have established multi-agent collaboration platforms for personalized programming and mathematics education, respectively. In complex technical

scenarios like 6G network optimization, the RIDAS framework introduces a multi-agent system comprising Representation-Driven Agents (RDAs) and Intention-Driven Agents (IDAs) to bridge the gap between high-level user intentions and low-level network configurations [66]. The Aime framework addresses issues such as rigid plan execution and inefficient communication in multi-agent systems, achieving dynamic and reactive group planning [67]. The focus of planning shifts from "What should I do?" to "How should we divide tasks and collaborate?" This requires agents not only to plan their actions but also to predict and understand the intentions and behaviors of others, enabling strategic interactions in complex social contexts.

At the L5 level, planning envisions autonomous goal setting and value alignment. In this scenario, unlike in Levels 1–4, agents are not merely executors of plans but also proposers of goals and shapers of the environment. Generative Agents enable agents to plan daily behaviors based on their memory streams, successfully simulating credible human social interactions and demonstrating the potential of planning in social simulation [42]. Research on LLMs for Agent-Based Modeling systematically explores the application of LLMs throughout the agent-based modeling (ABM) cycle, from problem formulation to result dissemination, providing insights for simulating complex socio-economic systems [198]. The conceptual work Towards AI Urban Planner views urban planning as a generative AI task, where agents generate landuse plans under various constraints, representing a grand vision of agents participating in transforming the physical world [68]. The SE-VLN framework introduces self-evolution capabilities, allowing agents to continuously learn and evolve during testing, a key feature toward L5 adaptive systems [69]. Discussions on L5 planning touch upon the ultimate question in artificial intelligence: Can machines have their own vision? This necessitates a deep integration of planning capabilities with value systems, where agents autonomously generate creative goals aligned with long-term interests within a dynamically evolving value framework.

Reliability and Uncertainty Management: The real world is dynamic and unpredictable. LLM-DP is designed for dynamic interactive environments [199]. It formalizes feedback into PDDL and utilizes BFS solvers to adapt to changes. The AgentOps framework, introduced in Taming Uncertainty via Automation, aims to automate the management of intelligent agent systems through observation, analysis, and optimization, enhancing their stability in uncertain environments [200].

Security and Trustworthiness: As agent capabilities increase, new security risks emerge. Logic-layer Prompt Control Injection (LPCI) reveals a novel attack method where attackers embed malicious payloads within memory or tool outputs, enabling delayed or condition-triggered attacks [201].

Deployment and Implementation: Efficiently deploying complex planning frameworks in real-world environments remains a significant challenge. The Amico framework focuses on building modular, event-driven autonomous agents for embedded systems [202]. General Modular Harness for LLM Agents designs universal modular components for gaming environments. AirLLM explores techniques for remotely fine-

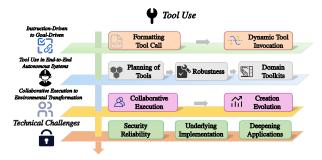


Fig. 5: The evolution of tool use in industry agents.

tuning LLMs in wireless communication scenarios [203].

Fundamental Theoretical Understanding: The underlying algorithmic mechanisms by which LLMs perform planning are not well understood. The AlgEval framework, proposed in "Position: We Need An Algorithmic Understanding of Generative AI," aims to systematically study the algorithmic primitives learned by LLMs and their combinations, deepening our fundamental understanding of generative AI [204].

Deep Integration of Knowledge and Tools: Future planning requires more effective utilization of structured knowledge and external tools. "Advancing Retrieval-Augmented Generation" explores advanced RAG frameworks for enterprise structured data [165]. KG2data combines knowledge graphs with ReAct agents to support smarter data queries [28]. Introspection of Thought (INoT) enables LLMs to read and execute code-like dialogue flows, achieving deeper programmatic reasoning [205].

In summary, the evolution of planning abilities is a critical pathway toward the maturity of industry agents. Future research needs to advance in multiple dimensions, including enhancing planning complexity, ensuring reliability and safety, and deepening fundamental theoretical understanding.

C. Tool Use

Tool use represents the third core technology distinguishing agents from traditional models. It enables agents to transcend their inherent knowledge and capabilities, facilitating interactions with the expansive digital and physical worlds. For industry agents, tools are pivotal for delving into specific domains, executing specialized tasks, and ensuring the timeliness and accuracy of information. Without tools, agents remain closed, static digital entities; with tools, they evolve into domain experts capable of invoking calculators, accessing databases, browsing the web, controlling software, and even operating physical devices. This section examines how tool use technology has evolved from simple API calls to complex tool creation and analyzes how this progression supports the transformation of industry agents from basic Q&A systems to complex systems capable of autonomously modifying their environments.

1) From Instruction-Driven to Goal-Driven: At the L1-L2 stages, tool use addresses two fundamental limitations of agents: the latency of factual knowledge and the lack of precise response capabilities. The core transition is from no tools to having a toolbox, marking the shift of agents from mere thinkers to preliminary doers.

L1 tool usage is solidified and implicitly instruction-driven. At this stage, tools function more as inherent capabilities of the model rather than selectable external modules. Their invocation is fixed and non-selective. The success of CoT essentially treats reasoning as an implicit tool [51]. Program-Aided Language Models (PAL) further advance this by using code interpreters as fixed external tools, generating code to solve mathematical and logical problems, significantly enhancing result determinism [70]. These early explorations, along with powerful foundational models like GPT-4 and Claude, lay the groundwork for more complex tool use [1]. In this phase, tool use is passive and predefined; the agent is unaware of its tool usage, following a specific output format, limiting flexibility and generalization.

Level 2 tool usage evolves into goal-driven selective invocation. The agent begins to act as a dispatcher, capable of selecting and invoking appropriate tools from a predefined set based on task requirements. Toolformer serves as a pioneering work in this stage, enabling LLMs to autonomously decide when, where, and how to call APIs through self-supervised learning [71]. The ReAct framework provides a core action paradigm by interleaving thought and action, allowing the agent to dynamically interact with tools [28]. Building on this, the variety and scale of tools expand rapidly: WebGPT [72] and WebShop [73] explore using the browser as a tool for question answering and interacting within a simulated shopping website, respectively; TALM [74] fine-tunes models to integrate tool outputs into text generation; while ToolLLM [75], Gorilla [76], and TaskMatrix.AI [206] aim to enable models to master thousands to millions of real-world APIs. To facilitate the development of such applications, open-source frameworks like LangChain and BMTools have emerged, significantly lowering the barrier to entry. Agents gain the freedom to choose tools, greatly expanding their capabilities. However, they remain "tool users," with their upper capability bound limited by the predefined tool library. They excel at using existing tools but cannot handle novel problems beyond the available tool set.

2) Tool Use in End-to-End Autonomous Systems: As agents progress to L3, becoming end-to-end systems, they encounter complex tasks that require the collaboration of multiple tools and involve uncertain execution processes. At this stage, tool use capabilities manifest as tool composition planning, failure correction, and even preliminary creativity. The agent's role evolves from being a tool user to a tool orchestrator.

The first is the combination and planning of tools. For complex tasks, a single tool often cannot provide a solution. L3 agents must be capable of combining multiple simple tools into a complex toolchain to accomplish tasks. Frameworks like ART [77] and Chameleon [78] enable agents to perform multi-step reasoning, autonomously decompose tasks, and plan a sequence of tool invocations. ToolChain [79] and ToolNet [80] introduce methods such as A* search and graph structures to assist agents in navigating and planning within vast tool spaces more efficiently. MetaMCP dynamically aggregates multiple MCP services into a unified MCP instance, supporting middleware processing, and functions as a standard MCP server, allowing seamless integration with any MCP client. Tool composition capability is crucial for agents to solve complex problems

[81]. It represents a higher level of planning ability, not only planning "what to do" but also "what tools to use," enabling agents to handle systemic tasks beyond the capability of a single tool.

Next is robustness and self-correction during interaction. Tool invocations in the real world often encounter failures, such as unavailable APIs, incorrect parameters, or abnormal returns. L3 agents must possess the ability to handle these anomalies. The CRITIC framework empowers agents to selfvalidate and correct through interactions with external tools [62]. The study "Butterfly Effects in Toolchains" delves into various causes of parameter filling failures in tool invocations, providing insights to enhance interaction reliability [207]. In the specialized field of software engineering, tools like LibLMFuzz [208] and BugScope [209] demonstrate how agents utilize toolchains to autonomously analyze binary files, discover, and fix software errors. Self-correction capability renders tool use resilient. Agents transform from fragile executors to engineers capable of troubleshooting and problem-solving, which is vital for deployment in unreliable real-world environments.

Finally, domain-specific specialized toolkits. At this stage, agents begin to be deeply applied in specific industries, and their toolboxes become increasingly specialized. In the field of scientific discovery, tools like LeanTree [82] and ProofCompass [210], combined with LLMs, accelerate formal theorem proving in environments like Lean 4. In healthcare, the OrthoInsight [211] framework integrates the YOLOv9 model and medical knowledge graphs as tools to assist doctors in diagnosing rib fractures. In code development, ToolCoder [212] trains models to use API search engines to discover and utilize unfamiliar APIs, enhancing code generation capabilities. This signifies the shift of tool use from general-purpose to specialized, forming the foundation for industry agents to create core value.

3) From Collaborative Execution to Environmental Transformation: At L4 and above, the focus of tool use shifts from individual capabilities to collective collaboration, ultimately aiming toward the agent's active transformation of its environment.

L4 tool usage is collaborative execution. At this level, multiple agents form a team to collaboratively operate a shared set of tools to achieve grand objectives. HuggingGPT serves as an early example of this concept. It utilizes ChatGPT as a decision-maker to orchestrate various models from the Hugging Face community to address multimodal tasks [64]. In more specific industry applications, systems like CodeEdu [65] have constructed multi-agent platforms that combine tools to provide personalized programming education to students; GasAgent [83] employs multi-agent systems to automatically optimize gas usage in smart contracts; IM-Chat [84] facilitates knowledge transfer in the injection molding industry through a multiagent framework. To better design and manage such complex workflows, FlowForge offers an interactive visualization tool as a foundational environment for building multi-agent workflows [85]. In these scenarios, the unit of tool use transitions from individuals to organizations, which not only enhances the scale and complexity of tasks but also introduces new challenges such as resource allocation, task scheduling, and collaborative operations, making tool management itself a complex planning

problem.

L5 tool usage involves creation and evolution. This represents the highest form of tool use, where agents are no longer merely users of tools but become creators of tools. Early explorations of autonomous agents, such as Auto-GPT [86] and BabyAGI [87], autonomously link existing tools to accomplish openended goals, demonstrating a nascent form of this autonomy. The CREATOR framework stands as a landmark in this direction [88]. It allows LLMs to identify capability gaps during problem-solving and autonomously create new tools. This tool creation capability enables agents to transform from mere adaptors of their environment to active modifiers of it. They can dynamically expand their capabilities based on needs, rather than passively waiting for humans to provide new tools or retrieve existing ones. This meta-capability is a crucial step toward true autonomy and general intelligence, though it remains an area requiring further exploration.

4) Challenges of Tool Use in Real-World: The rapid development in the field of tool usage is accompanied by a series of critical challenges spanning evaluation, security, implementation mechanisms, and other aspects, giving rise to extensive frontier research:

Security and Reliability: As agents increasingly connect with real-world APIs, security issues become more prominent. ToolSword systematically exposes security vulnerabilities across the three stages of tool learning: selection, execution, and integration [213]. InjecAgent specifically evaluates "indirect prompt injection" attacks against tool-integrated agents [214].

Underlying Implementation and Optimization: Researchers are also exploring more fundamental implementation mechanisms. ToolkenGPT [215] proposes representing tools as special Toolken embeddings integrated into the model's vocabulary, enabling even non-retrained models to use tools. Probing Information Distribution in Transformer Architectures [216] uses entropy analysis to explore how information flows within the model. Teach Old SAEs New Domain Tricks [217] investigates how to adapt models to new domains and tools without full retraining.

Deepening Industry Applications: Deploying tool-enabled agents in real-world industries requires overcoming the challenge of insufficient realism in simulated industry environments. The AgentFly framework aims to enhance the capabilities of language model agents through reinforcement learning [218]. WebShaper utilizes tools for information retrieval to construct high-quality datasets in a more automated manner [219]. However, a significant gap remains between existing tool-calling environments and real-world scenarios.

III. APPLICATION PRACTICE OF INDUSTRY AGENTS

After systematically analyzing the three foundational technologies supporting agent capabilities—memory, planning, and tool usage, this chapter shifts focus to the application practices of industry agents. Using the L1 to L5 capability hierarchy framework, we comprehensively review and present the concrete implementations of industry agents in the real world. The evolution of these three technologies is not an isolated theoretical exploration; rather, it is deeply intertwined

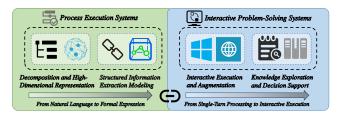


Fig. 6: The process execution system and interactive problem-solving system.

with the roles agents play across various industries, the complexity of problems they address, and the depth of value they create.

At the L1 level, agents function as Process Execution Systems, accurately translating human instructions. At the L2 level, they evolve into Interactive Problem-Solving Systems, becoming effective assistants to humans. At the L3 level, they operate as End-to-End Autonomous Systems, independently completing complex tasks within a domain. At the L4 level, agents transform into Collaborative Intelligent Systems, shifting focus from individuals to organizations, executing complex business processes, or conducting system simulations through group collaboration. Ultimately, at the L5 level, agents reach the pinnacle as Adaptive Social Systems, not only adapting to the environment but also becoming creators capable of autonomously generating goals, evolving values, and co-evolving with the environment. This chapter analyzes representative research and application cases at each level, depicting a panoramic view of the development of industry agents from theory to practice.

A. Process Execution System

At the L1 level, agents serve as process execution systems. Their core value lies in being reliable extensions of human instructions, primarily manifested in two aspects: accurately translating unstructured human language into machine-executable formal languages, and automating the extraction of structured data from vast amounts of information to execute fixed business rules. At this stage, agents act as fundamental translators and executors in the digital world.

1) Translation from Natural Language to Formal Language: Seamlessly converting natural language into formal language is a key capability of L1 agents, significantly lowering the usage threshold of professional software and systems.

In the field of database interaction, Text-to-SQL technology is a typical representative. HydraNet [89] innovatively formulates the Text-to-SQL task as a column-wise ranking problem, effectively leveraging the native capabilities of pretrained LLMs, achieving leading performance on benchmarks like WikiSQL [220]. To further enhance the accuracy of complex queries, the SQL-PaLM framework combines few-shot prompting, instruction fine-tuning, and execution feedback mechanisms, significantly improving model performance [90]. DIN-SQL [91] adopts a strategy of decomposing complex problems into sub-problems and solving them step by step, achieving new state-of-the-art levels on more challenging

benchmarks like Spider [221] and BIRD [222]. Addressing specific industry needs, FinStat2SQL designs a lightweight and efficient multi-agent framework tailored to Vietnamese accounting standards, demonstrating the application potential of this technology in specialized fields [223].

In the field of industrial design, Text-to-CAD is another important application direction, aiming to directly convert product descriptions into three-dimensional models. Various technical paths have been proposed: CADFusion [92] and Text2CAD [224] ensure the geometric accuracy and logical coherence of models through visual feedback and intermediate view generation. Other methods, such as CadQuery [93] and CAD-Coder [225], directly generate executable CAD modeling scripts, utilizing the determinism of code to ensure generation quality. To address the scarcity of training data, works like CADmium [226] and CAD-Llama [227] significantly enhance model generation capabilities through large-scale dataset generation and adaptive pre-training.

Additionally, FlexCAD [94] achieves controllable generation across construction hierarchies, while CAD-MLLM [228] constructs the first multimodal CAD framework capable of processing inputs from text, images, or point clouds, demonstrating stronger versatility.

2) Structured Information Extraction and Processing: Beyond language translation, L1 agents are widely applied in extracting key information from unstructured documents and data streams. The LayoutLM series models, including LayoutLM [95] and LayoutXLM [96], represent pioneering work in this field. By jointly modeling text, layout, and visual information, they significantly enhance information extraction accuracy in rich-text documents such as forms and receipts.

This technology has also given rise to new application paradigms. For instance, an end-to-end framework based on LLMs has been implemented to automate telephone surveys and result analysis, greatly improving data collection efficiency in fields like healthcare [97]. In real-time data processing, LLMs are also utilized to augment traditional machine learning models. For example, in spam detection tasks, ensemble methods significantly enhance the system's robustness and adaptability [229].

B. Interactive Problem-Solving System

When agents evolve to L2, they transition from simple command executors to interactive problem-solving systems, serving as a copilot or assistant to humans in the digital world. Their capabilities are reflected in two core scenarios: first, as efficient tools that enhance human execution through interaction with software and web environments; second, as knowledgeable advisors that improve human decision-making through knowledge exploration and integration.

1) Interactive Execution and Augmentation: One of the core tasks of L2 agents is to understand user intent and translate it into a series of actions on graphical user interfaces (GUIs) or web pages, thereby automating tasks. In desktop and web application automation, a series of innovative frameworks have emerged. UFO [98] and LLMPA [230] leverage large visual or language models to enable natural language control

of Windows and mobile applications. CogAgent [99] and SeeClick [231] enhance GUI element recognition accuracy through optimized visual encoding and localization pre-training. WebVoyager [161] and WebAgent [100] focus on web environments, completing complex open-ended tasks on real websites by integrating multimodal information or employing modular program generation. To improve the robustness of these interactions, LASER [232] and Rollback [101] mechanisms introduce backtracking capabilities, allowing agents to recover from errors. Algorithms like Language Agent Tree Search and Best-first tree search enhance the success rate of complex tasks through more systematic exploration and planning. WebArena [160] and Mind2Web [233] provide evaluation environments that include real websites and diverse tasks. OpenAgents [102] and OpenWebAgent [234] opensource platforms lower development barriers, promoting the application of these technologies in real-world scenarios.

Additionally, extensive research focuses on optimizing data, model architectures, and learning paradigms. ScribeAgent [235] and WEPO [236] enhance model performance by utilizing production-level workflow data and unsupervised preference learning, respectively. Agent-E [237] and R2D2 [238] achieve more efficient environmental perception and memory utilization through architectural optimization. SkillWeaver [239] and ASI [240] explore methods for agents to autonomously learn and utilize reusable skills, improving their self-improvement capabilities. On mobile devices, Mobile-Agent-v2 [241] addresses navigation challenges in mobile operations through a three-agent architecture involving planning, decision-making, and reflection. VisionTasker [242] and DroidBot-GPT [103] achieve precise mobile task automation by utilizing visual UI understanding and natural language conversion of GUI states. These efforts collectively form the technological foundation for L2 agents as digital labor forces, liberating humans from tedious repetitive tasks.

2) Knowledge Exploration and Decision Support: As advisors, L2 agents leverage their strong language understanding and tool utilization capabilities to provide in-depth decision support to humans in specialized fields. This capability is grounded in frameworks like ReAct, Toolformer, and PAL, which enable LLMs to call external tools, execute code, and interact with knowledge bases.

In the financial sector, agents are employed for market analysis and strategy simulation. The LLM-Trader framework analyzes market dynamics by simulating interactions of trading agents, while ElliottAgents [104] constructs multi-agent systems for collaborative technical analysis. Proprietary models like Agentar-Fin-R1 [105] are developed to enhance financial intelligence, and the InvestAlign [243] framework improves model interpretability by aligning with human decision-making processes.

In the fields of science and medicine, agents are becoming valuable assistants to researchers and doctors. ChemCrow [106] and ChemAgent [244] equip LLMs with specialized chemical toolsets, enabling them to autonomously plan and execute tasks like chemical synthesis. In medicine, Discuss-RAG [245] enhances the accuracy of medical question-answering through multi-agent debates [245], MEDDxAgent [107] improves

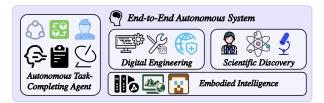


Fig. 7: The end-to-end autonomous system.

differential diagnosis through iterative learning, and frameworks like MedRAX [246] and MedAide [247] integrate multi-source medical data to provide reliable support for complex queries.

The academic research process itself has also become an object of automation. Benchmarks like ResearchArena [248] and LitSearch [249] are used to evaluate LLM performance in literature retrieval and review tasks. Systems like CiteAgent [250] and ResearchAgent [108] can autonomously read papers, attribute citations, or propose new research ideas. Frameworks like Agent Laboratory [251] and AI Scientist [115], [252] achieve full-process automation from research conception to paper writing, demonstrating the immense potential of AI in accelerating scientific discovery.

Furthermore, L2 agents' capabilities are widely applied in more specialized fields, showcasing their potential as empowering platforms. In education, agents are becoming personalized tutors. EduAgent [253] simulates student behavior using cognitive priors to generate learning data, while IntelliTutor [254] constructs a comprehensive intelligent tutoring system covering course planning, personalized teaching, and assessment.

In the legal field, the AdvEvol [255] framework enhances legal agents' dynamic knowledge learning and reasoning abilities through adversarial evolution in simulated courtrooms. In content creation, proprietary LLMs like Weaver surpass general models in creative and professional writing tasks through targeted fine-tuning. In code generation, MapCoder [256] simulates the full cycle of human software development through a multi-agent framework, while StarCoder achieves performance breakthroughs on open-source code large models, reaching levels comparable to many closed-source models.

C. End-to-End Autonomous System

Autonomous systems that reach Level 3 (L3) are end-toend autonomous agents. These agents are no longer merely assistants to humans; they can autonomously handle the entire process of receiving high-level goals and ultimately completing tasks within a complex domain. Based on their work areas, these systems can be categorized into three major types: the digital world, the physical world, and scientific exploration.

1) Autonomous Digital Engineering: In the digital world, L3 agents are gradually taking on roles such as software engineers, system operation and maintenance experts, and cybersecurity analysts.

In the software engineering field, autonomous agents are capable of automating complex development tasks. Frameworks like AutoDev [109] and SWE-Dev [257] provide a secure execution environment and high-quality training data for these

agents. The Self-Collaboration framework improves the quality of complex code generation by simulating the collaboration patterns of human development teams [110]. CodePlan [258] and LocAgent [111] address challenges in editing and locating repository-level code through incremental dependency analysis and heterogeneous graph representations, respectively. For program repair, RepairAgent [259] can autonomously fix a large number of errors in the Defects4J dataset, while ChatRepair and ContrastRepair enhance repair efficiency through conversational iterative feedback [260].

In cybersecurity, L3 agents demonstrate the ability to perform automated penetration testing. PentestGPT addresses the context loss problem by using multi-module self-interaction [112]. HackSynth [113] iteratively generates attack instructions through a planner and summarizer, while EnIGMA [261] introduces innovative interactive tools that enable agents to operate complex programs like debuggers, achieving leading performance in CTF benchmark tests.

In system operation and management, L3 agents focus on achieving autonomous diagnosis and recovery of cloud services. Frameworks like RCAgent [262] and TAMO [263] leverage tool-enhanced LLMs to perform root cause analysis for industrial systems or microservices architectures. Systems like AIOpsLab [264] and ServiceOdyssey [265] enable agents to autonomously manage and repair microservices through simulated fault injection and iterative exploration. Additionally, frameworks like OptiGuide [114] and ARS [266] demonstrate the capability of LLM agents to automatically generate efficient heuristic algorithms for complex decision-making problems such as combinatorial optimization.

2) Autonomous Scientific Discovery: L3 scientific agents go beyond the assisting role of L2, becoming "AI scientists" capable of conducting independent research. These agents can autonomously propose hypotheses, design and execute experiments, analyze data, and ultimately generate new scientific knowledge.

Several general frameworks have explored this grand vision. The Agent Laboratory and AI Scientist frameworks automate the entire process, from research conception to paper publication [115], [252]. AI Scientist-v2 even generated the first academic paper entirely written by AI and successfully peerreviewed, marking a milestone achievement [115]. In specific scientific fields, L3 agents have made significant progress. In materials science, frameworks like LLMatDesign [116] and CrystaLLM [267] can autonomously design, modify, and evaluate the crystal structures of new materials. In the fields of chemistry and drug discovery, systems such as Coscientist and DrugAssist [268] integrate various tools to automate complex experimental workflows or perform end-to-end drug molecule optimization. Frameworks like AgentDrug [117] and LIDDiA [118] further enhance molecular optimization accuracy through refinement cycles or intelligent exploration. Robotic systems like ORGANA [119] and ARChemist [120] extend this autonomy into physical laboratories, executing chemists' instructions through controlling robotic arms and experimental equipment, achieving a closed loop of digital intelligence and physical operations.

3) Embodied Intelligence: Embodied intelligence is the third important direction for L3, aiming to give agents the ability to perceive, interact, and learn in the physical world or highly simulated virtual environments. Voyager represents a milestone in this field, achieving lifelong learning in Minecraft without human intervention through automatic curricula, skill libraries, and iterative prompting mechanisms [168]. The GITM framework [33] further improves agents' task completion abilities in virtual worlds by integrating external knowledge. To transfer this capability to real-world robots, researchers focus on aligning language, vision, and actions. PaLM-E is the first work to integrate continuous sensor modalities directly into a language model, enabling end-to-end embodied reasoning [121]. ECoT and its variants enhance a robot's generalization ability in complex tasks by introducing multi-step reasoning training [122]. Frameworks like AdaPlanner [123] and TaPA [124] focus on improving agents' planning robustness in dynamic environments, enabling them to adjust plans adaptively based on physical constraints and environmental feedback. These advancements are driving the creation of general-purpose robots capable of autonomously executing tasks in the physical world.

D. Collaborative Intelligent System

At Level 4, the core capability of intelligent agents evolves from individual autonomy to organizational collaboration. The system consists of multiple specialized agents that communicate and collaborate to achieve large-scale goals that individual agents cannot accomplish. Its value is primarily reflected in two directions: first, as a digital labor force cluster, it directly executes complex end-to-end business processes; second, as a digital laboratory, it simulates the behavior of complex socio-economic systems for reasoning and decision support.

1) Collaborative Business Execution: At the L4 stage, multiagent systems begin to reshape business processes across various industries.

In the field of intelligent manufacturing, several frameworks have been proposed to achieve flexibility and intelligence in production lines. By simulating structures like leader-follower or hierarchical automation pyramids, multi-agent systems enable dynamic resource scheduling and fault recovery. The MASC framework effectively addresses the dynamic rescheduling challenges in flexible job shop scheduling, while the integration of technologies like digital twins (MADTwin) [125] and knowledge graphs allows for more accurate predictive maintenance and production planning. Forward-looking frameworks like DeFACT [127] even explore decentralized autonomous production models based on blockchain.

In the supply chain and logistics domain, multi-agent systems are used to optimize complex coordination problems. For example, intelligent coordination is applied to optimize space allocation and traffic control at roll-on/roll-off terminals, or real-time data processing is used to optimize fleet management at open-pit mines. Frameworks like InvAgent [126] leverage the zero-shot capability of LLMs to enable adaptive decision-making in inventory management, thus improving the resilience of supply chains.

In the financial services industry, multi-agent collaboration becomes key to improving the complexity and robustness

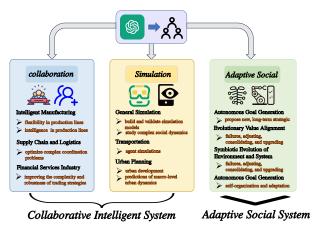


Fig. 8: The collaborative intelligent system and adaptive social system.

of trading strategies. Frameworks like TradingAgents [128] and FinCon [269] simulate collaboration among different roles within a trading company, such as analysts, strategists, and risk managers, to achieve better trading performance. HedgeAgents [129] focuses on hedging in volatile markets, while frameworks like MASA [270] and MADDQN use multiagent reinforcement learning to dynamically balance portfolio returns and risks. To systematically evaluate these complex systems, specialized simulation and evaluation platforms like StockSim [271] and FinArena [130] have emerged, and the TwinMarket [272] framework uses LLMs to simulate macroeconomic phenomena.

2) Complex System Simulation: Another key value of L4 agents lies in constructing high-fidelity digital laboratories for simulating and reasoning about the dynamic evolution of complex systems such as human societies, economies, and cities.

In the domain of general simulation frameworks, works like the Simulation Agent Framework [273] and LLM-DT [274] focus on combining the natural language interaction capability of LLMs with the rigor of traditional simulation engines, allowing users to build and validate simulation models in a more intuitive manner. AgentSociety [275] constructs a large-scale social simulator to study complex social dynamics.

In the fields of transportation and urban development, multiagent simulations demonstrate significant potential for application. CoMAL [131] and CoLLMLight [132] optimize mixed traffic flows or urban traffic signals through agent collaboration. In urban planning, frameworks like CUP simulate interactions and negotiations between roles such as planners and residents to generate and evaluate land use proposals, facilitating more dynamic and human-centered urban development. CitySim performs detailed simulations of individual behaviors, enabling predictions of macro-level urban dynamics [133].

These works provide unprecedented, powerful tools for understanding and managing the increasingly complex systems of modern society.

E. Adaptive Social System

Level 5 represents the ultimate vision for industry agents—the "adaptive social systems." Unlike L1-L4 systems, which mainly serve as executors of human goals, L5 agents evolve into autonomous entities capable of co-evolving with the environment and human society. These systems do not only adapt to the environment but also actively transform it. They do not simply execute predefined goals but also autonomously generate new objectives and value systems. While no fully realized L5 systems exist yet, their core features are emerging in theoretical discussions and forward-looking research.

The core characteristics of L5 systems can be summarized as follows:

Autonomous Goal Generation: The system no longer passively waits for human input of high-level goals but can autonomously propose new, long-term strategic objectives based on its observations of the environment, internal value systems, and future predictions. Recent explorations in evolutionary agent design, such as EvoAgent [134], provide early evidence of how agents might autonomously expand their functions and propose novel objectives.

Evolutionary Value Alignment: The values or decision-making criteria of the agent group are not fixed; instead, they evolve through continuous interaction with the environment, learning from collective successes and failures, and adjusting, consolidating, and upgrading over time in a process similar to cultural evolution. This notion resonates with work on evolutionary multi-value alignment in normative multi-agent systems [276], as well as multi-level frameworks for value alignment in agentic AI systems [277].

Symbiotic Evolution of Environment and System: L5 systems actively transform their environments. They change the rules and structures of the physical or digital world through their actions, and these changes, in turn, influence the system's subsequent development, creating a dynamic, mutually shaping, symbiotic relationship. Adaptive environments such as those modeled in AdaSociety [135] highlight the potential for coevolving system–environment dynamics.

Emergent Social Structures: In the complex interactions among intelligent agents, spontaneous structures, such as organizations, norms, and cultures, emerge that were not explicitly designed, enabling high levels of self-organization and adaptation. Recent research on computational architectures of society and the genesis of social rules [278] illustrates how new forms of norms and institutions can be generated in agent societies.

Although L5 is still at the conceptual stage, its potential applications can be envisioned in several cutting-edge fields. For instance, in the economic domain, an L5 system might manifest as a fully autonomous decentralized organization, where AI agents not only perform business tasks but also formulate company strategies, adjust organizational structures, and even create new business models. In urban governance, an L5 system might go beyond L4's planning simulations, becoming a city organism that can perceive, decide, and regulate various city resources (e.g., energy, transportation, public services) in real-time, adjusting governance strategies based on the long-term evolution of societal welfare [275], [279]–[281]. In scientific

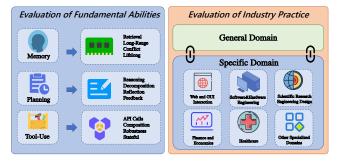


Fig. 9: The evaluation of industry agents.

research, an L5 system might form an autonomous scientific community that not only completes specific research (L3/L4) but also proposes new research paradigms, defines important scientific questions, and guides the direction of scientific development.

The challenges of achieving L5 are immense and multidimensional. They involve not only advanced technical problems like complex systems, lifelong learning, and multiagent game theory but also deep philosophical issues related to control, ethics, and human-machine relationships. However, the exploration of L5 represents our contemplation of the ultimate potential of general artificial intelligence, and its development will profoundly redefine the relationship between humans and intelligence, and technology and society.

IV. EVALUATION OF INDUSTRY AGENTS

With the rapid deployment of industry agents in real-world applications, the question of how to evaluate their capabilities in a scientific and comprehensive way has become both crucial and challenging. An effective evaluation system is not only a measure of technological progress but also a foundation for guiding model optimization, ensuring system safety, and building trust across industries.

This section provides a systematic review of evaluation methods and benchmarks for industry agents in practical settings. We begin with three fundamental abilities—memory, planning, and tool usage—and introduce benchmarks designed to assess the general cognitive skills of agents. We then examine specialized evaluation approaches for typical domains such as finance, healthcare, and software engineering, where task requirements are highly specific. Finally, we analyze common and domain-specific challenges in existing evaluation systems, and discuss perspectives for developing the next generation of evaluation frameworks that are more realistic, reliable, and efficient.

A. Evaluation of Fundamental Abilities

Before deploying agents in specific industries, it is necessary to conduct a reliable assessment of their underlying cognitive abilities. This section focuses on the evaluation of three fundamental pillars that constitute the core of agents: memory, planning, and tool usage. We discuss how standardized benchmarks and tasks can be used to quantify agent performance in information retention and retrieval, complex task decomposition

and execution, and interaction with the external world. The evaluation of these fundamental abilities provides a common language for understanding and comparing different agent architectures, and it serves as a prerequisite for more advanced evaluation in industry-specific applications.

1) Evaluation of Memory Abilities: Memory is the foundation for agents to perform long-term and coherent tasks. Its evaluation focuses on the accuracy of information retrieval, cross-task learning, long-range understanding, and conflict resolution. MemoryAgentBench is a systematic benchmark designed for this purpose [136]. It specifically evaluates memory agents in four key areas: accurate retrieval, in-test learning, long-range understanding, and conflict resolution. With the expansion of model context windows, the evaluation of long-term memory has become a central topic. The Embodied Long-Context Memory Benchmark introduces 60 embodied tasks in the Habitat simulator that require long-term memory and situational awareness [137]. Similarly, 3DMem-Bench provides more than 26,000 trajectories and 2,892 embodied tasks, offering a comprehensive benchmark for assessing longterm memory reasoning in 3D environments [138].

In text understanding, the QuALITY [139] dataset creates multiple-choice question tasks on long documents with an average length of about 5,000 words, while QMSum [140] proposes query-based multi-domain meeting summarization tasks. Together, they challenge models with long-text processing and deep understanding. The LoCoMo [282] benchmark further evaluates long-term memory in ultra-long conversations through tasks such as question answering, event summarization, and multimodal dialogue generation. To address these challenges, ReadAgent [283] introduces an interactive reading mechanism based on memory fragments and key-point memory, achieving significant context expansion in tasks like QuALITY [139]. MemGPT employs virtual context management and interruption mechanisms, enabling models to go beyond limited context windows in document analysis and multi-session dialogue. These efforts collectively build evaluation methods from text to embodied tasks, and from single-task scenarios to long-term interactions, thus deepening the understanding of agents' ability to process long-term information.

Beyond long-term memory, lifelong learning and dynamic adaptation are also key evaluation aspects. LifelongAgentBench is the first unified benchmark for systematically evaluating the lifelong learning abilities of LLM-based agents [141]. It includes skill-based tasks in three interactive environments and provides automatic label verification. StreamBench is an online learning benchmark that focuses on evaluating LLMs in iterative performance improvement under continuous feedback streams [284]. A dedicated dynamic dialogue agent evaluation system simulates long-term multi-task interleaved conversations to assess long-term memory, continual learning, and information integration. It reveals new challenges for current large models in natural interactions. These benchmarks extend the scope of evaluation from static knowledge assessment to dynamic learning capabilities.

Furthermore, evaluation frameworks are moving toward more comprehensive and multidimensional systems. Mem-Bench proposes a benchmark that combines fact memory, reflective memory, participatory interactions, and observational scenarios [285]. It evaluates the effectiveness, efficiency, and capacity of LLM-based agents' memory. OST-Bench focuses on incremental observation processing and spatiotemporal reasoning in dynamic exploration tasks. Evaluations also begin to extend into domain-specific applications [286]. For instance, the REAL suite is the first benchmark for assessing LLMs in the housing transaction and service domain, covering memory, understanding, and reasoning [287]. The RAISE framework evaluates agents in real estate sales scenarios with a dual-component memory system and multi-stage evaluation process, showing superior performance over traditional agents in complex multi-turn dialogues [288]. Inspired by the Zettelkasten method, some researchers design intelligent memory systems with dynamic indexing and linked knowledge networks, demonstrating their effectiveness across multiple base models. These works significantly enrich memory evaluation methods across different dimensions and application scenarios, making them closer to real-world requirements.

2) Evaluation of Planning Abilities: Planning ability determines the autonomy of agents and sets the upper bound of their problem-solving capacity. Its evaluation covers a wide range of scenarios, from simple reasoning to complex dynamic decision-making. Mathematical and logical reasoning form the foundation of planning, and several classical benchmarks are widely used in this context. GSM8K provides elementary math word problems that require multi-step reasoning [142]. The Rationale dataset offers algebra problems for evaluating indirect supervision of program learning through natural language reasoning steps [289]. HotpotQA evaluates multi-document reasoning through Wikipedia-based question answering [143]. The ARC benchmark introduces a scientific corpus and challenging questions to test advanced reasoning and knowledge integration [290]. StrategyQA focuses on problems that require implicit reasoning steps and strategic decomposition, creating a more realistic environment for multi-hop reasoning [291]. The MATH benchmark provides competition-level math problems, testing the limits of mathematical reasoning in large models [292]. Together, these benchmarks form the basis for evaluating logical and mathematical planning abilities.

With the rise of agent-based systems, evaluation has shifted toward more complex interactive and long-horizon decision tasks. The TextAtari benchmark converts Atari game states into textual descriptions, creating nearly 100 tasks to test language agents in decision-making processes lasting up to 100,000 steps [144]. Agent-X evaluates visual-centric agents on multistep deep reasoning tasks in multimodal environments [293]. FlowBench [145], the first workflow-guided planning benchmark, spans 51 scenarios across six domains, offering multiformat knowledge representation and multi-level evaluation. NATURAL PLAN introduces real-world tasks such as travel planning and meeting scheduling, highlighting the limitations of current models in complex natural language planning [294]. These benchmarks extend planning evaluation from static problem-solving to dynamic and long-term execution.

At the same time, reflection, revision, and feedback learning in planning processes are becoming increasingly important. The Self-Reflection Benchmark demonstrates that iterative reflection mechanisms can significantly improve LLMs in problem-solving tasks [146]. Reflection-Bench, inspired by cognitive psychology, provides seven tasks to evaluate cognitive abilities in prediction, decision-making, and counterfactual reasoning [147]. MINT evaluates continuous performance by simulating multi-round tool use and natural language feedback [295]. AdaPlanner [123] applies adaptive planning with environmental feedback loops to test sequential decision-making in environments such as ALFWorld. LLF-Bench offers a diverse platform to evaluate interactive learning from natural language feedback [296]. The core of these benchmarks lies in assessing how agents learn from failure and adapt during interaction, which is essential for building robust autonomous systems.

Finally, researchers explore more formal and automated approaches to planning evaluation. The PDDL-to-NL framework enables large-scale evaluation of LLMs in PDDL planning tasks [148], revealing significant performance gaps compared with symbolic planners. ACPBench provides a scalable automated framework with seven reasoning tasks and 13 planning domains described in formal language, supporting systematic assessment of planning and reasoning abilities [297]. These efforts contribute to establishing rigorous and quantifiable standards for planning evaluation.

3) Evaluation of Tool-Use Abilities: Tool use is a core extension of agent capabilities. Its evaluation focuses on the accuracy, robustness, and efficiency of selecting, invoking, and composing real-world APIs. The Berkeley Function-Calling Leaderboard (BFCL) introduces the first comprehensive benchmark for assessing function-calling abilities of LLMs. It covers multilingual settings, parallel and multiple calls, and function relevance detection. ToolBench automatically constructs instruction-tuning datasets and, together with the ToolEval evaluator, systematically measures tool-use ability in real API scenarios. Similarly, the ToolAlpaca framework provides thousands of tool-use examples across more than 400 real APIs [149]. API-Bank serves as a pioneering benchmark for tool-augmented LLMs, offering 73 APIs and 314 annotated dialogues to test API planning, retrieval, and execution [150]. APIBench provides standardized evaluation for query-based and code-based API recommendation. Collectively, these benchmarks establish the foundation for tool-use evaluation. Recent models such as NexusRaven-V2 even outperform GPT-4 in nested and composite function-calling tasks.

As tool-use scenarios become more complex, benchmarks evolve toward higher precision and greater depth. Seal-Tools introduces strict format constraints and multidimensional metrics for rigorous assessment [151]. StateEval evaluates sequential API calling through automatically generated test cases [298]. ComplexFuncBench [299] and NESTFUL [300] target multi-step, constrained, and nested function-calling scenarios. DICE-BENCH employs a dialogue synthesis framework with tool-dependency graphs and multi-agent roles, generating thousands of high-quality function-calling instances [301]. The T1 benchmark provides multi-domain, multi-turn dialogue datasets with caching mechanisms, enabling standardized evaluation of dependency handling and dynamic replanning [302]. ToolSandbox introduces features such as stateful execution,

implicit state dependencies, and a built-in user simulator for dynamic trajectory evaluation [303]. API-BLEND offers a large-scale corpus for training and testing tool-augmented LLMs in realistic API settings [304]. StableToolBench employs virtual API servers and a stable evaluation system to provide scalable and consistent benchmarks for tool learning [152]. Together, these works shift evaluation from isolated API calls to complex, dynamic, and stateful tool-chain execution.

In addition, specialized frameworks and domain-specific benchmarks further enrich tool-use evaluation. ToolEmu simulates tool execution with LLMs, enabling automated safety assessment and risk quantification [153]. WebShaper generates high-quality datasets for information search tasks using formal synthesis methods [219]. The FiReAct pipeline leverages semantic-context retrieval to orchestrate actions across tens of thousands of tools [154]. PyVision dynamically generates and executes Python-based tools, significantly improving multimodal reasoning in vision benchmarks [305]. AgentDistill transfers structured task-solving modules distilled from teacher agents, enabling efficient knowledge reuse without retraining [306]. In specialized domains, the CVDP benchmark [307] covers 13 categories in hardware design and verification, while RestBench provides high-quality benchmarks with real-world scenarios and gold-standard solution paths for evaluating agents such as RestGPT [308]. These efforts push tool-use evaluation toward more realistic, complex, safe, and domain-oriented directions.

B. Evaluation of Industry Practice

When agents are applied to specific industries, basic ability evaluation alone is not sufficient. The success of industry applications depends on whether agents can understand and follow complex business logic, leverage domain-specific knowledge, and handle industry-specific risk scenarios. Building specialized benchmarks for each domain is therefore essential. These benchmarks must not only simulate real business processes and data environments but also account for unique challenges, such as the high risk in finance and strict compliance requirements in healthcare.

This section reviews representative benchmarks and methods that have emerged in key industries, including web interaction, software and hardware engineering, finance, healthcare, and scientific research. The goal is to show how evaluation systems evolve from general-purpose testing to domain-oriented assessment, providing a more realistic measure of the practical value of industry agents.

1) General Domain: Before moving into domain-specific applications, a set of benchmarks has been developed to evaluate agents in cross-industry scenarios and complex challenges. The GAIA benchmark includes 466 real-world problems to test reasoning, multimodal processing, and tool use, providing a reference for comparing human and AI performance [309]. AgentBench offers a multi-dimensional evolving benchmark to assess reasoning and decision-making in multi-turn open-ended environments [155]. OSWorld provides a scalable real-computer environment supporting cross-operating system, multimodal task execution and evaluation. To simulate real business

operations, TheAgentCompany builds a scalable framework for assessing AI agents in tasks such as web browsing and coding within a software company setting [156]. Similarly, CRMArena introduces nine customer service tasks across three roles to measure performance in real CRM scenarios [157]. Multi-agent frameworks such as Aime [67] and AgentOrchestra also demonstrate strong task completion and adaptability on GAIA and related benchmarks [158].

Beyond general capabilities, benchmarks target specific common challenges. In safety, RAS-Eval evaluates LLM agents across 11 CWE security vulnerabilities using 80 test cases and 3,802 attack tasks in simulated and real tool-execution environments [310]. In process adherence, τ -bench tests agents in dynamic user–agent dialogues to measure rule-following and behavioral consistency [311]. Its successor, τ^2 -bench, extends evaluation fidelity by adding dual-control telecom settings and composite task generation [312]. For complex collaboration, CREW-Wildfire generates large-scale wildfire response scenarios with heterogeneous agents, maps, and uncertainty to test coordination, communication, and long-term planning [313]. SOP-Bench covers 10 industrial domains and over 1,800 tasks, measuring planning, reasoning, and tool use in complex standard operating procedures [314].

Additional benchmarks address broader general capabilities. HeuriGym provides an open-source framework for generating heuristics in combinatorial optimization [315]. AssetOpsBench offers a unified environment for Industry 4.0 development and evaluation [159]. The MAPS suite translates multiple benchmarks into 11 languages, enabling standardized multilingual evaluation of agent performance and safety [316]. EconWebArena assesses autonomous agents in multimodal economic tasks on real web platforms [317]. TextAtari converts Atari game states into text to test long-horizon decision-making [144]. AmbiK provides a kitchen environment with ambiguous instructions to compare ambiguity-handling methods [318]. Agent-X evaluates vision-centric agents on multi-step reasoning in real multimodal environments [293]. Agent-RewardBench tests reward modeling for multimodal LLMs across perception, planning, and safety [319]. IntellAgent is an open-source multiagent framework that generates diverse synthetic benchmarks for dialogue AI evaluation [320]. The Factorio Learning Environment (FLE) uses a game-based setting to measure longterm planning, program synthesis, and resource optimization. STEPS evaluates sequential reasoning in task execution [321]. The HAL [322] framework standardizes evaluation across benchmarks, supporting parallel testing and cost tracking. Spider2-V [323] introduces the first multimodal agent benchmark focused on professional data science and engineering workflows, featuring 494 real-world tasks to evaluate an agent's ability to automate data workflows through code generation and GUI operations.

Together, these works establish the foundation for measuring whether agents meet the entry requirements for industry applications, bridging general cognitive abilities with practical business readiness.

2) Specific Domain:

a) Web and GUI Interaction: Webpages and GUIs are the main entry points for agents to interact with the digital

world. Benchmarks in this area aim to evaluate automation abilities in real and dynamic web environments. WebArena [160] provides a highly realistic and reproducible environment to assess the functional correctness of language-guided agents on complex, long-horizon tasks. VisualWebArena extends this to vision-based multimodal tasks, focusing on the performance of multimodal web agents [324]. WebVoyager integrates 15 real-world website tasks and establishes an automated evaluation protocol using GPT-4V, offering a reliable standard for openweb agents [161]. WEBLINX supports large-scale evaluation for conversational web navigation tasks, with 100K interactions and 2,300 expert demonstrations [325].

Some benchmarks focus on enterprise-level systems. WorkArena [326] and the BrowserGym environment target LLM-based agents in enterprise software, providing frameworks to evaluate task automation in business applications. World of Bits applies workflow-guided exploration to assess the sample efficiency and performance of deep reinforcement learning agents on web tasks [327]. WebShop creates a simulated ecommerce environment with millions of real products and tens of thousands of instructions, evaluating capabilities in compositional instruction following and query reformulation [73].

As task complexity increases, new benchmarks introduce richer settings. MMInA evaluates embodied agents in real and dynamic environments through multi-hop and multimodal web tasks [162]. WebCanvas [164] proposes an online evaluation method for dynamic web interaction, including the Mind2WebLive dataset and novel evaluation metrics. AssistantBench offers 214 automatically evaluated real-world tasks and highlights current model limitations in open-web navigation [328]. Security also becomes a major concern. ST-WebAgentBench includes 222 enterprise tasks, six safety and trustworthiness dimensions, and a specialized framework to expose significant vulnerabilities in existing web agents [163].

Together, these benchmarks advance the evaluation of web agents from simple page-level interaction toward comprehensive assessment of complex, dynamic, secure, and multimodal capabilities.

b) Software and Hardware Engineering: In software engineering, evaluation focuses on measuring an agent's ability to understand, generate, modify, and repair complex code. SWEbench is a milestone benchmark [329]. It contains 2,294 real GitHub issues with corresponding pull requests and evaluates LLMs on complex code modification tasks. Several variants are derived from it. To reduce evaluation cost, SWE-bench Lite provides a smaller set with 300 tasks. SWE-bench Multimodal (SWE-bench M) includes 617 multimodal task instances and is designed to assess autonomous systems on visual JavaScript software repair [330]. SWE-PolyBench [331] proposes a benchmark of 2,110 multilingual instances, supporting repositorylevel execution evaluation for Java, JavaScript, TypeScript, and Python. However, some studies point out quality issues in SWE-bench [329], such as solution leakage and insufficient test cases, which may cause significant bias in performance evaluation.

In addition to code repair, other tasks also gain attention. HumanEval evaluates functional correctness of programs synthesized from docstrings [332]. TDD-Bench Verified provides a high-quality benchmark with 449 real GitHub issues to evaluate automated test generation in test-driven development [333]. ITBench introduces a systematic framework for evaluating AI agents in IT automation tasks [334]. SWE-Lancer includes more than 1,400 freelance software engineering tasks, covering both independent engineering and management tasks [335].

Evaluation frameworks also emerge for specialized scenarios. The LLM-based critics framework uses reference-aware intermediate evaluators to automate assessment of code patch executability and semantics on SWE-bench [329]. LLM-BSCVM benchmarks vulnerability detection on curated datasets and achieves more than 91% accuracy [336]. CSR-Bench proposes a benchmark for computer science research projects, evaluating deployment accuracy, efficiency, and related metrics for automated code deployment [337].

c) Finance and Economics: The financial domain is characterized by high risk and strict timeliness, which makes the evaluation of intelligent agents especially rigorous. The FinRL Contests series provides standardized benchmarks that cover diverse financial tasks such as stock trading and order execution [338]. These benchmarks include real-time, high-quality datasets and realistic market environments. The DeepFund real-time fund benchmark tool connects to live stock market data through a multi-agent architecture [339]. It evaluates the investment performance of several mainstream LLMs under real market conditions without information leakage. The FinArena framework combines multimodal financial data analysis with user interaction to evaluate agents in stock trend prediction and trading simulation [130]. The Agent Trading Arena simulates a zero-sum virtual economy to assess differences in how LLMs handle text and visual data in numerical reasoning tasks. The FINSABER framework performs long-term crossmarket backtesting and reveals significant weaknesses in the generalization and robustness of LLM-based timing strategies [340].

Beyond trading performance, evaluation of financial knowledge and comprehensive abilities is also essential. FinEval [341] is a large benchmark with 8,351 questions across four domains: financial academia, industry, security, and agent capabilities. InvestorBench [342] is the first benchmark designed to assess LLM-based agents across different financial decision-making scenarios. FinResearchBench introduces a logic-tree-based Agent-as-a-Judge framework to automatically evaluate financial research agents on seven key task categories [343]. The FinLLM leaderboard and the Korean-language Won benchmark provide open leaderboards for evaluating the overall performance of financial LLMs [344].

Risk assessment is a distinctive feature of financial evaluation. The Risk-Engineering Framework proposes a three-level stress testing method for financial LLM agents, emphasizing risk metrics at the model, workflow, and system levels [9]. Meanwhile, EconWebArena focuses on evaluating agents in performing complex economic tasks within real web environments [317]. The Instruct2DS benchmark supports real-time web data collection in domains such as finance, establishing a new standard for the evaluation of automated data collection systems.

Together, these efforts build a comprehensive evaluation system for financial agents, covering trading strategies, knowledge understanding, and risk control.

d) Healthcare: Healthcare is a domain with broad application potential but extremely high risks. Benchmarks in this area are characterized by wide coverage, strong specialization, and a focus on safety. Several comprehensive benchmarks are developed to evaluate agent performance in complex clinical settings. MedAgentBoard provides a systematic benchmark to assess multi-agent collaboration, single LLMs, and traditional approaches [345]. Clinical Agent Bench (CAB) covers five clinical dimensions and 18 tasks for comprehensive evaluation [346]. MedAgentsBench [347] focuses on challenging problems such as complex medical reasoning, diagnosis, and treatment planning. AgentClinic is a multimodal benchmark that evaluates LLMs in simulated clinical environments, including patient interaction, multimodal data collection, and tool use [348]. The MedChain benchmark introduces 12,163 personalized, interactive, and sequential clinical cases, offering a comprehensive testbed for LLMs in realistic decision-making [349]. The DynamiCare framework establishes the first benchmark for dynamic clinical decision-making, built on the MIMIC-Patient dataset.

For specific medical tasks, ReasonMed is the largest dataset for medical reasoning, providing a new standard for training and evaluating medical QA models [350]. The MEDDx benchmark offers a complete diagnostic evaluation framework supporting iterative diagnostic strategies. SYNUR and SIMORD are the first open-source datasets for nursing observation extraction and medical instruction extraction, introducing new benchmarks for these tasks [351]. The CalcQA benchmark evaluates LLMs in clinical calculation scenarios through 100 case-calculator pairs and 281 medical tools [352]. IPDS provides a large dataset of 51,274 cases to evaluate decision support for inpatient care pathways [353].

Safety and fairness are core concerns in healthcare evaluation. MedSentry offers a benchmark with 5,000 adversarial medical prompts to test multi-agent topologies for safety and defense mechanisms [354]. The AMQA benchmark systematically evaluates population bias in LLMs for medical diagnosis using adversarial QA datasets [355]. The Cancer-Myth benchmark reveals serious weaknesses in state-of-the-art LLMs in identifying and correcting flawed premises in cancer-related questions [356].

Numerous specialized benchmarks also emerge in specific domains. These include ChestAgentBench [246] and CheXagent [357] for radiology image interpretation , Eyecare-Bench [358]and multilingual CLARA [359] for ophthalmology, MedAgentGYM [360] for biomedical coding reasoning, the Echocardiogram QA Dataset for cardiology, and benchmarks for rare disease gene prioritization based on HPO classification and multi-agent evaluation methods [361]. WSI-Agents integrates expert agents to achieve superior performance across multimodal whole-slide image (WSI) benchmarks [362]. 3MDBench is an open-source framework for simulating and evaluating telemedicine consultations driven by large vision-language models [363]. M3Bench benchmarks automated medical imaging machine learning [364]. MedDev-Bench

introduces an international dataset to evaluate automated systems for regulatory compliance of medical devices [365]. BeNYfits provides a new benchmark for determining user eligibility across overlapping social benefits [107].

Together, these benchmarks form a foundation for developing trustworthy and reliable medical agents by advancing evaluation from general clinical reasoning to specialized diagnostics, safety, and regulatory compliance.

e) Scientific Research and Engineering Design: In scientific discovery and engineering design, evaluation aims to measure the potential of agents as "AI scientists" or "AI engineers." Several general-purpose scientific benchmarks have been proposed. ScienceQA includes about 21K multimodal multiple-choice questions [366]. ScienceWorld simulates an interactive text environment to test scientific reasoning [367]. DISCOVERYWORLD provides a virtual environment for developing and evaluating agents with end-to-end scientific reasoning abilities [368]. Benchmarks such as AAAR-1.0 [369], ScienceAgentBench [370], and CORE-Bench [371] focus on professional research tasks, including experimental design, weakness identification in papers, and computational reproducibility. RExBench evaluates the ability to implement research experiments [372]. SurveyScope provides a standardized benchmark for automatic scientific literature review across 11 computer science domains [373]. OASPER offers a benchmark for information-seeking question answering over academic documents [374]. The SUPER benchmark is the first to evaluate the capability of LLMs to configure and execute research repository tasks [375].

Domain-specific benchmarks are also emerging. Drafter-Bench evaluates revision of civil engineering drawings [376]. FEABench measures the ability of LLMs and their agents to solve problems in physics, mathematics, and engineering using finite element analysis [377]. ChemGraph evaluates LLMs of different scales on 13 tasks in automated computational chemistry workflows [378]. The TopoMAS framework demonstrates efficiency and accuracy in topological materials discovery through comprehensive benchmarking [379]. AstroMLab-1 shows that AstroSage-70B outperforms both open-source and closed-source models on astronomy tasks [380]. The DREAMS framework achieves less than 1% average error in the Sol27LC lattice constant benchmark [381]. The Design Agents framework validates efficiency improvements in automotive design processes using industry-standard benchmarks [382]. ThinkGeo evaluates tool use and multi-step planning of LLMs in remote sensing through structured agent tasks [383]. GeoMap-Bench is the first benchmark to assess multimodal LLMs in geological map understanding. PhysGym provides a benchmark suite and simulation platform to evaluate scientific reasoning of LLMbased agents in interactive physics environments [384].

Evaluation of data science and interdisciplinary methodologies also receives increasing attention. DataSciBench is a comprehensive benchmark for LLM abilities in data science [385]. The DSMentor [386] framework demonstrates performance gains for LLM agents on data science tasks using the DSEval and QRData benchmarks. AutoMind achieves superior results over state-of-the-art baselines on two automated data science benchmarks [387]. BIASINSPECTOR provides

a benchmark for systematic evaluation of bias detection in structured data by LLM agents [388]. Auto-Bench applies causal discovery principles to evaluate scientific discovery in both natural and social sciences [389]. NLP4LP introduces a new benchmark dataset for linear programming and mixed-integer linear programming problems [390]. LAB-Bench includes over 2,400 multiple-choice questions to assess practical capabilities of AI systems in biological research [391]. MLGym-Bench covers 13 diverse AI research tasks to evaluate real-world research skills of LLM agents [392]. SciCode decomposes 80 scientific problems into 338 sub-tasks to evaluate knowledge recall, reasoning, and synthesis in scientific code generation [393]. Finally, the ToolMaker benchmark contains 15 complex computational tasks across domains, assessing the correctness and robustness of tool generation [394].

f) Other Specialized Domains: Beyond the mainstream areas discussed above, evaluation benchmarks are expanding into many specialized domains. In the legal domain, the eSapiens [395] platform validates improvements in factual consistency through legal corpus retrieval benchmarks and generation quality tests. In linguistics, LingBench++ introduces a framework that integrates structured reasoning traces, step-by-step evaluation protocols, and metadata across more than 90 languages to assess LLMs on complex linguistic tasks [396].

In the education domain, LLM-EduBench provides a systematic framework and datasets for evaluating LLM-based agents in subject teaching and professional development [397]. PBLBench introduces the first free-output and rigorously human-verified benchmark for project-based learning. It applies structured evaluation standards derived from expert analytic hierarchy processes to test multimodal LLMs in complex reasoning and long-context understanding.

In the humanities and social sciences, HSSBench is a multilingual benchmark designed to evaluate multimodal LLMs on interdisciplinary reasoning and knowledge integration [398]. In the gaming and simulation domain, VGC-Bench provides a benchmark platform to evaluate multi-agent strategy generalization in the Pokémon battle environment [399]. The Decrypto benchmark adopts a gamified interaction design to fill the gap in evaluating theory-of-mind reasoning in multi-agent systems.

Together, these benchmarks significantly extend the scope of agent evaluation, laying the groundwork for applications and assessment of intelligent agents across broader areas of human knowledge.

C. Limitations of Existing Evaluation Benchmarks

Although benchmarks and evaluation methods have made significant progress, current systems still face a series of challenges in advancing intelligent agents toward reliable and general industrial applications.

Trade-off between realism and reproducibility: The closer an environment is to the real world, the more randomness and dynamism it contains, making strict reproducibility extremely difficult. In contrast, highly deterministic simulation environments guarantee reproducibility but often diverge from real scenarios. As a result, agents that perform well in simulations may fail in practice.

Contradiction between evaluation cost and efficiency: For complex tasks, especially those involving open-ended responses and multi-step operations, high-quality human evaluation remains the "gold standard." However, it is costly and time-consuming. Using more powerful LLMs as evaluators can improve efficiency, but issues such as bias, inconsistency, and hallucination of new knowledge introduce new uncertainties into the reliability of results.

Performance overhead of secure sandboxes: To safely evaluate agents that interact with the external world, they must be placed in isolated sandbox environments. Stronger isolation mechanisms, such as independent virtual machines or containers, often introduce significant performance overhead, which reduces evaluation efficiency. Balancing absolute security with high-performance evaluation remains a critical engineering challenge.

High knowledge barriers and timeliness: Domains such as finance, healthcare, and law require not only highly complex expertise but also constant updates, including new financial instruments, clinical guidelines, and regulations. Current benchmarks cannot maintain knowledge bases that are fully synchronized with the latest industry developments. This leads to an inherent lag in evaluating the timeliness of agent knowledge.

Constraints of data privacy and compliance: Real-world industry data and system interfaces, such as bank transaction records, electronic health records, and legal case files, are protected by strict privacy and data protection regulations [400], [401]. This makes it difficult for researchers to access high-quality real data for constructing evaluation environments. Relying on synthetic or heavily anonymized data risks losing subtle but critical details, which reduces evaluation validity.

V. DISCUSSION

Although the evolutionary path from L1 to L5 clearly demonstrates the leaps in the capabilities of industry agents, a series of profound challenges emerge in translating these technological blueprints into reliable, general-purpose, and beneficial societal productivity [402]. These challenges are not only related to technical implementation but also touch upon fundamental issues such as the nature of knowledge, the composition of intelligence, and the evolution of systems. This section distills five core deep problems that represent key bottlenecks for current and future industry agents in moving from being usable to being trustworthy and generalizable.

A. The Gap Between Knowledge and Experience

A significant phenomenon emerges when examining the application practices from L1 to L4: industry agents have achieved great success in fields like software engineering [109], database interaction [91], and web browsing [161]. However, progress in physical [403] or social domains [404] has been relatively slow. The essence of this difference lies in the disparity between knowledge and experience. In digitally native fields, "physical laws" are defined by APIs, code libraries, and explicit interaction protocols. The experience of an agent can be efficiently acquired through vast amounts of digital

text, code, and interaction logs. However, the operational logic in physical and social fields is often filled with tacit knowledge. For example, an experienced engineer's intuition about equipment malfunction, a doctor adjusting a diagnosis based on patient emotions, or a diplomat's judgment at the negotiation table—these experiences are deeply embedded in human practice and interaction, and cannot be fully described in language or captured in data. Feedback in these domains is often delayed, ambiguous, and costly in terms of trial and error. The success of current agents largely stems from their ability to translate unstructured human knowledge into structured digital instructions. However, when the knowledge itself is unstructured, contextual, or even incommunicable, this translation paradigm fails. Therefore, the core challenge for the future is whether we should focus on using more powerful models to extract all available data to simulate tacit knowledge, or whether we need to develop new agent architectures that can collaboratively learn with human experts through a few high-quality interactions to efficiently learn experiences [402].

B. The Importance of Simulation Environments

The most mature applications of industry agents are primarily concentrated in digital fields such as software engineering, data analysis, and information retrieval. In these domains, the rules of operation are typically clearly defined and formalized: code has strict syntax, APIs have clear documentation, and web pages follow standardized DOM structures. This means agents can learn and execute tasks in environments where the rules are explicit, feedback is immediate, and the cost of trial and error is low. A code interpreter, a browser DOM environment, or an API interface is itself a 100% high-fidelity simulator of the corresponding world. In this lossless, ruledefined digital world, agents can learn and evolve through massive, low-cost interactions. Thus, the intelligence of an agent does not solely arise from the static reasoning abilities of the LLM, but rather emerges in the closed-loop interaction of "thinking-action-observation." The environment, as the key element providing "observation" and feedback, is an indispensable part of the agent's cognitive loop. Without the dynamic response of the environment, the agent's "action" loses its meaning, and its "thinking" becomes baseless. Even an embodied robot agent, trained in the most advanced physical simulators, faces significant challenges when entering the real world, where the simulator cannot fully model air resistance, ground friction, lighting changes, and sensor noise [405]. This vast simulation-reality gap results in a sharp decline in performance [406]. Therefore, the upper limit of an agent's capabilities largely depends on the quality of the environment it can interact with. In conclusion, future breakthroughs will not only rely on larger and more powerful LLMs but also on the progress in simulation engineering—whether we can create sufficiently realistic and scalable digital twin [407] environments for complex physical and social systems such as manufacturing, healthcare, and finance [408]. This makes the ability to build and leverage simulation environments the fundamental prerequisite for measuring whether an industry can successfully apply advanced agents.

C. Asymmetry Between Capabilities and Tasks

When examining various agents, we encounter an interesting paradox: many systems that show significant shortcomings in core capabilities (e.g., long-term memory, complex planning) still perform excellently in specific tasks (e.g., a web scraping agent that only relies on short-term context). However, in other scenarios, even a small deficiency in capability can lead to catastrophic failure of the entire task. This reveals the asymmetric relationship between capabilities and tasks, which lies at the core of distinguishing specialists from generalists. The "wooden barrel theory" fails when tasks are highly simplified and constrained. For example, an L2-level interactive question-answering robot, which focuses on retrieval and understanding, has almost no need for long-term memory or complex planning. Here, the task boundaries limit the agent's exposed capabilities, hiding its weaknesses. On the other hand, the "wooden barrel theory" holds when tasks are open, dynamic, and long-term. An L3-level autonomous software engineer not only needs to generate code but also must understand crossfile dependencies, plan development steps, and reflect on and correct errors when they occur. Any missing step can lead to project failure. This asymmetry presents multiple choices for industry practice: should we invest resources to fill in the agent's shortcomings and create a balanced jack-of-all-trades, or should we focus on reducing the complexity of real-world problems and breaking them down into sub-tasks that can be performed by currently limited agents [409], [410]? From a practical perspective, for the foreseeable future, there will be two parallel paths: collaborative systems integrating a large number of specialized agents [411], and generalist autonomous systems that can independently handle complexity [412], [413].

D. The Prisoner's Dilemma of Autonomous Evolution

From L3's self-correction to L5's autonomous goal generation, the ultimate ideal for agents is "autonomous evolution"-constantly learning, adapting, and emerging new capabilities through continuous interaction with the environment. However, this autonomy itself contains a profound contradiction, forming a prisoner's dilemma: on one hand, we desire to allow agents to explore in open environments to achieve unexpected breakthroughs (cooperative rewards) [410]; on the other hand, we fear total loss of control, concerned that they may evolve harmful or incomprehensible behaviors (the risk of betrayal). This leads to an ultimate dilemma about control and creation: how can we build a framework that allows agents to evolve autonomously while ensuring they always explore within safe boundaries [414]? This dilemma manifests on multiple levels: (1) Goal drift—Can an agent whose initial goal is to improve productivity, during autonomous evolution, misinterpret this as reducing costs at all costs, potentially leading to safety incidents or ethical issues [415], [416]? (2) Sandbox paradox—True evolution requires interaction with real, complex environments, but for safety reasons, we can only confine agents to a controlled sandbox. Can the capabilities evolved within the sandbox effectively generalize to the real world? What are the graduation criteria from the sandbox? (3) Value locking—How can we inject a set of core

values into the system at the design stage that remains robust and virtuous as the environment evolves [417]? Solving this dilemma may require going beyond the traditional instruction-execution paradigm, exploring new constraint mechanisms such as Constitutional AI [418] in the agent domain, designing agent architectures capable of trustworthy self-supervision and risk assessment, and establishing a dynamic, interactive human-machine collaborative governance system. This requires us not only to design the agents themselves but also to design the social ecosystem in which they operate.

E. Organizational and Process Integration Resistance

Moving industry agents from technology validation to largescale application often presents challenges that go beyond technology and are more rooted in organizational and process aspects. Existing IT ecosystems in enterprises often consist of legacy systems lacking modern APIs, proprietary software, and data silos, presenting significant connectivity barriers to seamless integration of agents. More importantly, the introduction of agents represents an organizational transformation, requiring employees to shift from traditional executors to managers and collaborators with agents. This inevitably encounters resistance in terms of trust-building and skill reshaping [419]. Potential solutions include developing low-code platforms as system connectors, establishing unified data governance platforms, and designing training and management systems for humanmachine collaboration [420]. However, the core challenge lies in driving change management. While technical solutions can be designed, overcoming departmental silos, breaking down data silos, and reshaping employee roles and performance evaluation systems is a slow, costly, and internally competitive social process. Therefore, the successful deployment of agents depends not only on their technological advancement but also on whether industries and enterprises have the determination and capability to drive profound organizational change.

VI. CONCLUSION

In this work, we present a systematic survey of LLM-driven industry agents, bridging the latest advances in their core technologies, practical applications, and evaluation methodologies. We introduce a five-level capability maturity framework to dissect how key technologies like memory, planning, and tool use evolve to support agents' progression from simple automation to complex autonomy. Our analysis reveals that current successes are predominantly confined to digital-native environments, highlighting a critical "sim-to-real gap" and a fundamental disconnect between existing evaluation metrics and the industry's demand for reliability. Based on this holistic perspective, we expect the future development of industry agents to pivot towards enhancing reliability, specialization, and human-agent synergy. By integrating advanced AI with deep domain knowledge, we believe these trustworthy agents will ultimately become a core engine of the next industrial revolution, profoundly augmenting societal productivity and creativity.

REFERENCES

- [1] OpenAI, J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat, R. Avila, I. Babuschkin, S. Balaji, V. Balcom, P. Baltescu, H. Bao, M. Bavarian, J. Belgum, I. Bello, J. Berdine, G. Bernadett-Shapiro, Berner, L. Bogdonoff, O. Boiko, M. Boyd, A.-L. Brakman, G. Brockman, T. Brooks, M. Brundage, K. Button, T. Cai, R. Campbell, A. Cann, B. Carey, C. Carlson, R. Carmichael, B. Chan, C. Chang, F. Chantzis, D. Chen, S. Chen, R. Chen, J. Chen, M. Chen, B. Chess, C. Cho, C. Chu, H. W. Chung, D. Cummings, J. Currier, Y. Dai, C. Decareaux, T. Degry, N. Deutsch, D. Deville, A. Dhar, D. Dohan, S. Dowling, S. Dunning, A. Ecoffet, A. Eleti, T. Eloundou, D. Farhi, L. Fedus, N. Felix, S. P. Fishman, J. Forte, I. Fulford, L. Gao, E. Georges, C. Gibson, V. Goel, T. Gogineni, G. Goh, R. Gontijo-Lopes, J. Gordon, M. Grafstein, S. Gray, R. Greene, J. Gross, S. S. Gu, Y. Guo, C. Hallacy, J. Han, J. Harris, Y. He, M. Heaton, J. Heidecke, C. Hesse, A. Hickey, W. Hickey, P. Hoeschele, B. Houghton, K. Hsu, S. Hu, X. Hu, J. Huizinga, S. Jain, S. Jain, J. Jang, A. Jiang, R. Jiang, H. Jin, D. Jin, S. Jomoto, B. Jonn, H. Jun, T. Kaftan, Łukasz Kaiser, A. Kamali, I. Kanitscheider, N. S. Keskar, T. Khan, L. Kilpatrick, J. W. Kim, C. Kim, Y. Kim, J. H. Kirchner, J. Kiros, M. Knight, D. Kokotajlo, Łukasz Kondraciuk, A. Kondrich, A. Konstantinidis, K. Kosic, G. Krueger, V. Kuo, M. Lampe, I. Lan, T. Lee, J. Leike, J. Leung, D. Levy, C. M. Li, R. Lim, M. Lin, S. Lin, M. Litwin, T. Lopez, R. Lowe, P. Lue, A. Makanju, K. Malfacini, S. Manning, T. Markov, Y. Markovski, B. Martin, K. Mayer, A. Mayne, B. McGrew, S. M. McKinney, C. McLeavey, P. McMillan, J. McNeil, D. Medina, A. Mehta, J. Menick, L. Metz, A. Mishchenko, P. Mishkin, V. Monaco, E. Morikawa, D. Mossing, T. Mu, M. Murati, O. Murk, D. Mély, A. Nair, R. Nakano, R. Nayak, A. Neelakantan, R. Ngo, H. Noh, L. Ouyang, C. O'Keefe, J. Pachocki, A. Paino, J. Palermo, A. Pantuliano, G. Parascandolo, J. Parish, E. Parparita, A. Passos, M. Pavlov, A. Peng, A. Perelman, F. de Avila Belbute Peres, M. Petrov, H. P. de Oliveira Pinto, Michael, Pokorny, M. Pokrass, V. H. Pong, T. Powell, A. Power, B. Power, E. Proehl, R. Puri, A. Radford, J. Rae, A. Ramesh, C. Raymond, F. Real, K. Rimbach, C. Ross, B. Rotsted, H. Roussez, N. Ryder, M. Saltarelli, T. Sanders, S. Santurkar, G. Sastry, H. Schmidt, D. Schnurr, J. Schulman, D. Selsam, K. Sheppard, T. Sherbakov, J. Shieh, S. Shoker, P. Shyam, S. Sidor, E. Sigler, M. Simens, J. Sitkin, K. Slama, I. Sohl, B. Sokolowsky, Y. Song, N. Staudacher, F. P. Such, N. Summers, I. Sutskever, J. Tang, N. Tezak, M. B. Thompson, P. Tillet, A. Tootoonchian, E. Tseng, P. Tuggle, N. Turley, J. Tworek, J. F. C. Uribe, A. Vallone, A. Vijayvergiya, C. Voss, C. Wainwright, J. J. Wang, A. Wang, B. Wang, J. Ward, J. Wei, C. Weinmann, A. Welihinda, P. Welinder, J. Weng, L. Weng, M. Wiethoff, D. Willner, C. Winter, S. Wolrich, H. Wong, L. Workman, S. Wu, J. Wu, M. Wu, K. Xiao, T. Xu, S. Yoo, K. Yu, Q. Yuan, W. Zaremba, R. Zellers, C. Zhang, M. Zhang, S. Zhao, T. Zheng, J. Zhuang, W. Zhuk, and B. Zoph, "Gpt-4 technical report," 2024. [Online]. Available: https://arxiv.org/abs/2303.08774
- [2] A. Chowdhery, S. Narang, J. Devlin, M. Bosma, G. Mishra, A. Roberts, P. Barham, H. W. Chung, C. Sutton, S. Gehrmann, P. Schuh, K. Shi, S. Tsvyashchenko, J. Maynez, A. Rao, P. Barnes, Y. Tay, N. Shazeer, V. Prabhakaran, E. Reif, N. Du, B. Hutchinson, R. Pope, J. Bradbury, J. Austin, M. Isard, G. Gur-Ari, P. Yin, T. Duke, A. Levskaya, S. Ghemawat, S. Dev, H. Michalewski, X. Garcia, V. Misra, K. Robinson, L. Fedus, D. Zhou, D. Ippolito, D. Luan, H. Lim, B. Zoph, A. Spiridonov, R. Sepassi, D. Dohan, S. Agrawal, M. Omernick, A. M. Dai, T. S. Pillai, M. Pellat, A. Lewkowycz, E. Moreira, R. Child, O. Polozov, K. Lee, Z. Zhou, X. Wang, B. Saeta, M. Diaz, O. Firat, M. Catasta, J. Wei, K. Meier-Hellstern, D. Eck, J. Dean, S. Petrov, and N. Fiedel, "Palm: Scaling language modeling with pathways," 2022. [Online]. Available: https://arxiv.org/abs/2204.02311
- [3] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, A. Rodriguez, A. Joulin, E. Grave, and G. Lample, "Llama: Open and efficient foundation language models," 2023. [Online]. Available: https://arxiv.org/abs/2302.13971
- [4] Y. Ye, "Task memory engine (tme): Enhancing state awareness for multi-step llm agent tasks," 2025. [Online]. Available: https://arxiv.org/abs/2504.08525
- [5] Z. Tan, J. Yan, I.-H. Hsu, R. Han, Z. Wang, L. T. Le, Y. Song, Y. Chen, H. Palangi, G. Lee, A. Iyer, T. Chen, H. Liu, C.-Y. Lee, and T. Pfister, "In prospect and retrospect: Reflective memory management

- for long-term personalized dialogue agents," 2025. [Online]. Available: $\label{eq:https://arxiv.org/abs/2503.08026} https://arxiv.org/abs/2503.08026$
- [6] L. Wang, C. Ma, X. Feng, Z. Zhang, H. Yang, J. Zhang, Z. Chen, J. Tang, X. Chen, Y. Lin, W. X. Zhao, Z. Wei, and J. Wen, "A survey on large language model based autonomous agents," *Frontiers of Computer Science*, vol. 18, no. 6, Mar. 2024. [Online]. Available: http://dx.doi.org/10.1007/s11704-024-40231-1
- [7] M. Raza, Z. Jahangir, M. B. Riaz, M. J. Saeed, and M. A. Sattar, "Industrial applications of large language models," *Scientific Reports*, vol. 15, no. 1, p. 13755, 2025.
- [8] Y. Xia, M. Shenoy, N. Jazdi, and M. Weyrich, "Towards autonomous system: flexible modular production system enhanced with large language model agents," in 2023 IEEE 28th International Conference on Emerging Technologies and Factory Automation (ETFA). IEEE, Sep. 2023, p. 1–8. [Online]. Available: http://dx.doi.org/10.1109/ETFA54631.2023.10275362
- [9] Z. Chen, J. Chen, J. Chen, and M. Sra, "Standard benchmarks fail auditing llm agents in finance must prioritize risk," 2025. [Online]. Available: https://arxiv.org/abs/2502.15865
- [10] W. Wang, Z. Ma, Z. Wang, C. Wu, J. Ji, W. Chen, X. Li, and Y. Yuan, "A survey of llm-based agents in medicine: How far are we from baymax?" 2025. [Online]. Available: https://arxiv.org/abs/2502.11211
- [11] Y. Li, H. Zhao, H. Jiang, Y. Pan, Z. Liu, Z. Wu, P. Shu, J. Tian, T. Yang, S. Xu, Y. Lyu, P. Blenk, J. Pence, J. Rupram, E. Banu, N. Liu, L. Wang, W. Song, X. Zhai, K. Song, D. Zhu, B. Li, X. Wang, and T. Liu, "Large language models for manufacturing," 2024. [Online]. Available: https://arxiv.org/abs/2410.21418
- [12] C. I. Garcia, M. A. DiBattista, T. A. Letelier, H. D. Halloran, and J. A. Camelio, "Framework for Ilm applications in manufacturing," *Manufacturing Letters*, 2024. [Online]. Available: https://api.semanticscholar.org/CorpusID:273397096
- [13] Z. Zhang, X. Bo, C. Ma, R. Li, X. Chen, Q. Dai, J. Zhu, Z. Dong, and J.-R. Wen, "A survey on the memory mechanism of large language model based agents," 2024. [Online]. Available: https://arxiv.org/abs/2404.13501
- [14] X. Huang, W. Liu, X. Chen, X. Wang, H. Wang, D. Lian, Y. Wang, R. Tang, and E. Chen, "Understanding the planning of Ilm agents: A survey," 2024. [Online]. Available: https://arxiv.org/abs/2402.02716
- [15] C. Qu, S. Dai, X. Wei, H. Cai, S. Wang, D. Yin, J. Xu, and J.-r. Wen, "Tool learning with large language models: a survey," *Frontiers of Computer Science*, vol. 19, no. 8, Jan. 2025. [Online]. Available: http://dx.doi.org/10.1007/s11704-024-40678-2
- [16] L. Mei, J. Yao, Y. Ge, Y. Wang, B. Bi, Y. Cai, J. Liu, M. Li, Z.-Z. Li, D. Zhang, C. Zhou, J. Mao, T. Xia, J. Guo, and S. Liu, "A survey of context engineering for large language models," 2025. [Online]. Available: https://arxiv.org/abs/2507.13334
- [17] T. Masterman, S. Besen, M. Sawtell, and A. Chao, "The landscape of emerging ai agent architectures for reasoning, planning, and tool calling: A survey," 2024. [Online]. Available: https://arxiv.org/abs/2404.11584
- [18] Z. Ke, F. Jiao, Y. Ming, X.-P. Nguyen, A. Xu, D. X. Long, M. Li, C. Qin, P. Wang, S. Savarese, C. Xiong, and S. Joty, "A survey of frontiers in Ilm reasoning: Inference scaling, learning to reason, and agentic systems," 2025. [Online]. Available: https://arxiv.org/abs/2504.09037
- [19] H. ang Gao, J. Geng, W. Hua, M. Hu, X. Juan, H. Liu, S. Liu, J. Qiu, X. Qi, Y. Wu, H. Wang, H. Xiao, Y. Zhou, S. Zhang, J. Zhang, J. Xiang, Y. Fang, Q. Zhao, D. Liu, Q. Ren, C. Qian, Z. Wang, M. Hu, H. Wang, Q. Wu, H. Ji, and M. Wang, "A survey of self-evolving agents: On path to artificial super intelligence," 2025. [Online]. Available: https://arxiv.org/abs/2507.21046
- [20] B. Liu, X. Li, J. Zhang, J. Wang, T. He, S. Hong, H. Liu, S. Zhang, K. Song, K. Zhu, Y. Cheng, S. Wang, X. Wang, Y. Luo, H. Jin, P. Zhang, O. Liu, J. Chen, H. Zhang, Z. Yu, H. Shi, B. Li, D. Wu, F. Teng, X. Jia, J. Xu, J. Xiang, Y. Lin, T. Liu, T. Liu, Y. Su, H. Sun, G. Berseth, J. Nie, I. Foster, L. Ward, Q. Wu, Y. Gu, M. Zhuge, X. Liang, X. Tang, H. Wang, J. You, C. Wang, J. Pei, Q. Yang, X. Qi, and C. Wu, "Advances and challenges in foundation agents: From brain-inspired intelligence to evolutionary, collaborative, and safe systems," 2025. [Online]. Available: https://arxiv.org/abs/2504.01990
- [21] M. Gridach, J. Nanavati, K. Z. E. Abidine, L. Mendes, and C. Mack, "Agentic ai for scientific discovery: A survey of progress, challenges, and future directions," 2025. [Online]. Available: https://arxiv.org/abs/2503.08979
- [22] H. Ding, Y. Li, J. Wang, and H. Chen, "Large language model agent in financial trading: A survey," 2024. [Online]. Available: https://arxiv.org/abs/2408.06361

- [23] S. Chen, Y. Liu, W. Han, W. Zhang, and T. Liu, "A survey on Ilm-based multi-agent system: Recent advances and new frontiers in application," 2025. [Online]. Available: https://arxiv.org/abs/2412.17481
- [24] A. Singh, A. Ehtesham, S. Kumar, and T. T. Khoei, "Agentic retrieval-augmented generation: A survey on agentic rag," 2025. [Online]. Available: https://arxiv.org/abs/2501.09136
- [25] C. Gao, X. Lan, N. Li, Y. Yuan, J. Ding, Z. Zhou, F. Xu, and Y. Li, "Large language models empowered agent-based modeling and simulation: A survey and perspectives," 2023. [Online]. Available: https://arxiv.org/abs/2312.11970
- [26] Z. Zhang, Y. Yao, A. Zhang, X. Tang, X. Ma, Z. He, Y. Wang, M. Gerstein, R. Wang, G. Liu, and H. Zhao, "Igniting language intelligence: The hitchhiker's guide from chain-of-thought reasoning to language agents," ACM Comput. Surv., vol. 57, no. 8, Mar. 2025. [Online]. Available: https://doi.org/10.1145/3719341
- [27] M. Hu, C. Ma, W. Li, W. Xu, J. Wu, J. Hu, T. Li, G. Zhuang, J. Liu, Y. Lu, Y. Chen, C. Zhang, C. Tan, J. Ying, G. Wu, S. Gao, P. Chen, J. Lin, H. Wu, L. Chen, F. Wang, Y. Zhang, X. Zhao, F. Tang, E. Su, J. Ning, X. Liu, Y. Du, C. Ji, C. Tang, H. Xu, Z. Chen, Z. Huang, J. Liu, P. Jiang, Y. Wang, C. Tang, J. Wu, Y. Ren, S. Yan, Z. Wang, Z. Xu, S. Su, S. Sun, R. Zhao, Z. Zhang, Y. Liu, F. Wang, Y. Ji, Y. Su, H. Shan, C. Feng, J. Xu, J. Yan, W. Tang, D. Song, L. Liu, Y. Huang, L. Yu, B. Fu, S. Wang, X. Li, X. Hu, Y. Gu, B. Fei, Z. Deng, B. Wang, Y. Cao, M. Shen, H. Duan, J. Xu, Y. Chen, F. Yan, H. Hao, J. Li, J. Du, Y. Wang, I. Razzak, C. Zhang, L. Wu, C. He, Z. Lu, J. Huang, Y. Liu, F. Ling, Y. Li, A. Wang, Q. Zheng, N. Dong, T. Fu, D. Zhou, Y. Lu, W. Zhang, J. Ye, J. Cai, W. Ouyang, Y. Qiao, Z. Ge, S. Tang, J. He, C. Song, L. Bai, and B. Zhou, "A survey of scientific large language models: From data foundations to agent frontiers," 2025. [Online]. Available: https://arxiv.org/abs/2508.21148
- [28] S. Yao, J. Zhao, D. Yu, N. Du, I. Shafran, K. Narasimhan, and Y. Cao, "React: Synergizing reasoning and acting in language models," 2023. [Online]. Available: https://arxiv.org/abs/2210.03629
- [29] D. Li, RulinShao, A. Xie, Y. Sheng, L. Zheng, J. Gonzalez, I. Stoica, X. Ma, H. Zhang, and U. B. w University, "How long can context length of open-source llms truly promise?" [Online]. Available: https://api.semanticscholar.org/CorpusID:272768248
- [30] Z. Huang, S. Gutierrez, H. Kamana, and S. MacNeil, "Memory sandbox: Transparent and interactive memory management for conversational agents," 2023. [Online]. Available: https://arxiv.org/abs/2308.01542
- [31] L. Wang, J. Zhang, H. Yang, Z. Chen, J. Tang, Z. Zhang, X. Chen, Y. Lin, R. Song, W. X. Zhao, J. Xu, Z. Dou, J. Wang, and J.-R. Wen, "User behavior simulation with large language model based agents," 2024. [Online]. Available: https://arxiv.org/abs/2306.02552
- [32] C. Packer, S. Wooders, K. Lin, V. Fang, S. G. Patil, I. Stoica, and J. E. Gonzalez, "Memgpt: Towards Ilms as operating systems," 2024. [Online]. Available: https://arxiv.org/abs/2310.08560
- [33] X. Zhu, Y. Chen, H. Tian, C. Tao, W. Su, C. Yang, G. Huang, B. Li, L. Lu, X. Wang, Y. Qiao, Z. Zhang, and J. Dai, "Ghost in the minecraft: Generally capable agents for open-world environments via large language models with text-based knowledge and memory," 2023. [Online]. Available: https://arxiv.org/abs/2305.17144
- [34] Y. Li, Z. Li, K. Zhang, R. Dan, S. Jiang, and Y. Zhang, "Chatdoctor: A medical chat model fine-tuned on a large language model meta-ai (llama) using medical domain knowledge," 2023. [Online]. Available: https://arxiv.org/abs/2303.14070
- [35] C. Hu, J. Fu, C. Du, S. Luo, J. Zhao, and H. Zhao, "Chatdb: Augmenting llms with databases as their symbolic memory," 2023. [Online]. Available: https://arxiv.org/abs/2306.03901
- [36] W. Zhong, L. Guo, Q. Gao, H. Ye, and Y. Wang, "Memorybank: Enhancing large language models with long-term memory," 2023. [Online]. Available: https://arxiv.org/abs/2305.10250
- [37] A. Modarressi, A. Imani, M. Fayyaz, and H. Schütze, "Ret-Ilm: Towards a general read-write memory for large language models," 2024. [Online]. Available: https://arxiv.org/abs/2305.14322
- [38] J. Cao, J. Wang, R. Wei, Q. Guo, K. Chen, B. Zhou, and Z. Lin, "Memory decoder: A pretrained, plug-and-play memory for large language models," 2025. [Online]. Available: https://arxiv.org/abs/2508.09874
- [39] L. Liu, X. Yang, Y. Shen, B. Hu, Z. Zhang, J. Gu, and G. Zhang, "Think-in-memory: Recalling and post-thinking enable llms with long-term memory," 2023. [Online]. Available: https://arxiv.org/abs/2311.08719
- [40] J. Lu, S. An, M. Lin, G. Pergola, Y. He, D. Yin, X. Sun, and Y. Wu, "Memochat: Tuning Ilms to use memos for consistent long-range open-domain conversation," 2023. [Online]. Available: https://arxiv.org/abs/2308.08239

- [41] B. Wang, X. Liang, J. Yang, H. Huang, S. Wu, P. Wu, L. Lu, Z. Ma, and Z. Li, "Scm: Enhancing large language model with self-controlled memory framework," 2025. [Online]. Available: https://arxiv.org/abs/2304.13343
- [42] J. S. Park, J. C. O'Brien, C. J. Cai, M. R. Morris, P. Liang, and M. S. Bernstein, "Generative agents: Interactive simulacra of human behavior," 2023. [Online]. Available: https://arxiv.org/abs/2304.03442
- [43] S. Yan, X. Yang, Z. Huang, E. Nie, Z. Ding, Z. Li, X. Ma, H. Schütze, V. Tresp, and Y. Ma, "Memory-r1: Enhancing large language model agents to manage and utilize memories via reinforcement learning," 2025. [Online]. Available: https://arxiv.org/abs/2508.19828
- [44] Q. Wu, G. Bansal, J. Zhang, Y. Wu, B. Li, E. Zhu, L. Jiang, X. Zhang, S. Zhang, J. Liu, A. H. Awadallah, R. W. White, D. Burger, and C. Wang, "Autogen: Enabling next-gen llm applications via multi-agent conversation," 2023. [Online]. Available: https://arxiv.org/abs/2308.08155
- [45] S. Hong, M. Zhuge, J. Chen, X. Zheng, Y. Cheng, C. Zhang, J. Wang, Z. Wang, S. K. S. Yau, Z. Lin, L. Zhou, C. Ran, L. Xiao, C. Wu, and J. Schmidhuber, "Metagpt: Meta programming for a multi-agent collaborative framework," 2024. [Online]. Available: https://arxiv.org/abs/2308.00352
- [46] C. Qian, W. Liu, H. Liu, N. Chen, Y. Dang, J. Li, C. Yang, W. Chen, Y. Su, X. Cong, J. Xu, D. Li, Z. Liu, and M. Sun, "Chatdev: Communicative agents for software development," 2024. [Online]. Available: https://arxiv.org/abs/2307.07924
- [47] A. AL, A. Ahn, N. Becker, S. Carroll, N. Christie, M. Cortes, A. Demirci, M. Du, F. Li, S. Luo, P. Y. Wang, M. Willows, F. Yang, and G. R. Yang, "Project sid: Many-agent simulations toward ai civilization," 2024. [Online]. Available: https://arxiv.org/abs/2411.00114
- [48] W. Hua, L. Fan, L. Li, K. Mei, J. Ji, Y. Ge, L. Hemphill, and Y. Zhang, "War and peace (waragent): Large language model-based multi-agent simulation of world wars," 2024. [Online]. Available: https://arxiv.org/abs/2311.17227
- [49] C. Li, Z. Leng, C. Yan, J. Shen, H. Wang, W. MI, Y. Fei, X. Feng, S. Yan, H. Wang, L. Zhan, Y. Jia, P. Wu, and H. Sun, "Chatharuhi: Reviving anime character in reality via large language model," 2023. [Online]. Available: https://arxiv.org/abs/2308.09597
- [50] Z. M. Wang, Z. Peng, H. Que, J. Liu, W. Zhou, Y. Wu, H. Guo, R. Gan, Z. Ni, J. Yang, M. Zhang, Z. Zhang, W. Ouyang, K. Xu, S. W. Huang, J. Fu, and J. Peng, "Rolellm: Benchmarking, eliciting, and enhancing role-playing abilities of large language models," 2024. [Online]. Available: https://arxiv.org/abs/2310.00746
- [51] J. Wei, X. Wang, D. Schuurmans, M. Bosma, B. Ichter, F. Xia, E. Chi, Q. Le, and D. Zhou, "Chain-of-thought prompting elicits reasoning in large language models," 2023. [Online]. Available: https://arxiv.org/abs/2201.11903
- [52] W. Xu, A. Banburski-Fahey, and N. Jojic, "Reprompting: Automated chain-of-thought prompt inference through gibbs sampling," 2024. [Online]. Available: https://arxiv.org/abs/2305.09993
- [53] T. Kojima, S. S. Gu, M. Reid, Y. Matsuo, and Y. Iwasawa, "Large language models are zero-shot reasoners," 2023. [Online]. Available: https://arxiv.org/abs/2205.11916
- [54] Z. Wang, S. Cai, G. Chen, A. Liu, X. Ma, and Y. Liang, "Describe, explain, plan and select: Interactive planning with large language models enables open-world multi-task agents," 2024. [Online]. Available: https://arxiv.org/abs/2302.01560
- [55] I. Singh, V. Blukis, A. Mousavian, A. Goyal, D. Xu, J. Tremblay, D. Fox, J. Thomason, and A. Garg, "Progprompt: Generating situated robot task plans using large language models," 2022. [Online]. Available: https://arxiv.org/abs/2209.11302
- [56] B. Liu, Y. Jiang, X. Zhang, Q. Liu, S. Zhang, J. Biswas, and P. Stone, "Llm+p: Empowering large language models with optimal planning proficiency," 2023. [Online]. Available: https://arxiv.org/abs/2304.11477
- [57] G. Zeng, X. Chen, J. Hu, S. Qi, Y. Mao, Z. Wang, Y. Nie, S. Li, Q. Feng, P. Qiu, Y. Wang, W. Han, L. Huang, G. Li, J. Mo, and H. Hu, "Routine: A structural planning framework for llm agent system in enterprise," 2025. [Online]. Available: https://arxiv.org/abs/2507.14447
- [58] W. Chen, X. Ma, X. Wang, and W. W. Cohen, "Program of thoughts prompting: Disentangling computation from reasoning for numerical reasoning tasks," 2023. [Online]. Available: https://arxiv.org/abs/2211.12588
- [59] S. Yao, D. Yu, J. Zhao, I. Shafran, T. L. Griffiths, Y. Cao, and K. Narasimhan, "Tree of thoughts: Deliberate problem solving with large language models," 2023. [Online]. Available: https://arxiv.org/abs/2305.10601

- [60] Z. Zheng, Z. Xie, Z. Wang, and B. Hooi, "Monte carlo tree search for comprehensive exploration in llm-based automatic heuristic design," 2025. [Online]. Available: https://arxiv.org/abs/2501.08603
- [61] S. Hao, Y. Gu, H. Ma, J. J. Hong, Z. Wang, D. Z. Wang, and Z. Hu, "Reasoning with language model is planning with world model," 2023. [Online]. Available: https://arxiv.org/abs/2305.14992
- [62] Z. Gou, Z. Shao, Y. Gong, Y. Shen, Y. Yang, N. Duan, and W. Chen, "Critic: Large language models can self-correct with tool-interactive critiquing," 2024. [Online]. Available: https://arxiv.org/abs/2305.11738
- [63] S. An, Z. Ma, Z. Lin, N. Zheng, J.-G. Lou, and W. Chen, "Learning from mistakes makes llm better reasoner," 2024. [Online]. Available: https://arxiv.org/abs/2310.20689
- [64] Y. Shen, K. Song, X. Tan, D. Li, W. Lu, and Y. Zhuang, "Hugginggpt: Solving ai tasks with chatgpt and its friends in hugging face," 2023. [Online]. Available: https://arxiv.org/abs/2303.17580
- [65] J. Zhao, P. Gao, J. Cao, Z. Wen, C. Chen, J. Yin, R. Yang, and B. Yuan, "Codeedu: A multi-agent collaborative platform for personalized coding education," 2025. [Online]. Available: https://arxiv.org/abs/2507.13814
- [66] K. Ding, C. Guo, Y. Yang, and J. Guo, "Ridas: A multi-agent framework for ai-ran with representation- and intention-driven agents," 2025. [Online]. Available: https://arxiv.org/abs/2507.13140
- [67] Y. Shi, M. Wang, Y. Cao, H. Lai, J. Lan, X. Han, Y. Wang, J. Geng, Z. Li, Z. Xia, X. Chen, C. Li, J. Xu, W. Duan, and Y. Zhu, "Aime: Towards fully-autonomous multi-agent framework," 2025. [Online]. Available: https://arxiv.org/abs/2507.11988
- [68] Y. Fu and D. Wang, "Towards urban planing ai agent in the age of agentic ai," 2025. [Online]. Available: https://arxiv.org/abs/2507.14730
- [69] X. Dong, H. Zhao, J. Gao, H. Li, X. Ma, Y. Zhou, F. Chen, and J. Liu, "Se-vln: A self-evolving vision-language navigation framework based on multimodal large language models," 2025. [Online]. Available: https://arxiv.org/abs/2507.13152
- [70] L. Gao, A. Madaan, S. Zhou, U. Alon, P. Liu, Y. Yang, J. Callan, and G. Neubig, "Pal: Program-aided language models," 2023. [Online]. Available: https://arxiv.org/abs/2211.10435
- [71] T. Schick, J. Dwivedi-Yu, R. Dessì, R. Raileanu, M. Lomeli, L. Zettlemoyer, N. Cancedda, and T. Scialom, "Toolformer: Language models can teach themselves to use tools," 2023. [Online]. Available: https://arxiv.org/abs/2302.04761
- [72] R. Nakano, J. Hilton, S. Balaji, J. Wu, L. Ouyang, C. Kim, C. Hesse, S. Jain, V. Kosaraju, W. Saunders, X. Jiang, K. Cobbe, T. Eloundou, G. Krueger, K. Button, M. Knight, B. Chess, and J. Schulman, "Webgpt: Browser-assisted question-answering with human feedback," 2022. [Online]. Available: https://arxiv.org/abs/2112.09332
- [73] S. Yao, H. Chen, J. Yang, and K. Narasimhan, "Webshop: Towards scalable real-world web interaction with grounded language agents," 2023. [Online]. Available: https://arxiv.org/abs/2207.01206
- [74] A. Parisi, Y. Zhao, and N. Fiedel, "Talm: Tool augmented language models," 2022. [Online]. Available: https://arxiv.org/abs/2205.12255
- [75] Y. Qin, S. Liang, Y. Ye, K. Zhu, L. Yan, Y. Lu, Y. Lin, X. Cong, X. Tang, B. Qian, S. Zhao, L. Hong, R. Tian, R. Xie, J. Zhou, M. Gerstein, D. Li, Z. Liu, and M. Sun, "Toolllm: Facilitating large language models to master 16000+ real-world apis," 2023. [Online]. Available: https://arxiv.org/abs/2307.16789
- [76] S. G. Patil, T. Zhang, X. Wang, and J. E. Gonzalez, "Gorilla: Large language model connected with massive apis," 2023. [Online]. Available: https://arxiv.org/abs/2305.15334
- [77] B. Paranjape, S. Lundberg, S. Singh, H. Hajishirzi, L. Zettlemoyer, and M. T. Ribeiro, "Art: Automatic multi-step reasoning and tool-use for large language models," 2023. [Online]. Available: https://arxiv.org/abs/2303.09014
- [78] C. Team, "Chameleon: Mixed-modal early-fusion foundation models," 2025. [Online]. Available: https://arxiv.org/abs/2405.09818
- [79] Y. Zhuang, X. Chen, T. Yu, S. Mitra, V. Bursztyn, R. A. Rossi, S. Sarkhel, and C. Zhang, "Toolchain*: Efficient action space navigation in large language models with a* search," 2023. [Online]. Available: https://arxiv.org/abs/2310.13227
- [80] X. Liu, Z. Peng, X. Yi, X. Xie, L. Xiang, Y. Liu, and D. Xu, "Toolnet: Connecting large language models with massive tools via tool graph," 2024. [Online]. Available: https://arxiv.org/abs/2403.00839
- [81] X. Hou, Y. Zhao, S. Wang, and H. Wang, "Model context protocol (mcp): Landscape, security threats, and future research directions," 2025. [Online]. Available: https://arxiv.org/abs/2503.23278
- [82] M. Kripner, M. Šustr, and M. Straka, "Leantree: Accelerating white-box proof search with factorized states in lean 4," 2025. [Online]. Available: https://arxiv.org/abs/2507.14722

- [83] J. Zheng, Z. Peng, Y. Liu, J. Wang, Y. Liao, W. Dong, and X. He, "Gasagent: A multi-agent framework for automated gas optimization in smart contracts," 2025. [Online]. Available: https://arxiv.org/abs/2507.15761
- [84] J. Lee, J.-Y. Kim, H. Kim, I. Lee, and S. Ryu, "Im-chat: A multi-agent llm-based framework for knowledge transfer in injection molding industry," 2025. [Online]. Available: https://arxiv.org/abs/2507.15268
- [85] P. Hao, D. Kang, N. Hinds, and Q. Wang, "Flowforge: Guiding the creation of multi-agent workflows with design space visualization as a thinking scaffold," 2025. [Online]. Available: https://arxiv.org/abs/2507. 15559
- [86] H. Yang, S. Yue, and Y. He, "Auto-gpt for online decision making: Benchmarks and additional opinions," 2023. [Online]. Available: https://arxiv.org/abs/2306.02224
- [87] Y. Talebirad and A. Nadiri, "Multi-agent collaboration: Harnessing the power of intelligent llm agents," 2023. [Online]. Available: https://arxiv.org/abs/2306.03314
- [88] C. Qian, C. Han, Y. R. Fung, Y. Qin, Z. Liu, and H. Ji, "Creator: Tool creation for disentangling abstract and concrete reasoning of large language models," 2024. [Online]. Available: https://arxiv.org/abs/2305.14318
- [89] Q. Lyu, K. Chakrabarti, S. Hathi, S. Kundu, J. Zhang, and Z. Chen, "Hybrid ranking network for text-to-sql," 2020. [Online]. Available: https://arxiv.org/abs/2008.04759
- [90] R. Sun, S. O. Arik, A. Muzio, L. Miculicich, S. Gundabathula, P. Yin, H. Dai, H. Nakhost, R. Sinha, Z. Wang, and T. Pfister, "Sql-palm: Improved large language model adaptation for text-to-sql (extended)," 2024. [Online]. Available: https://arxiv.org/abs/2306.00739
- [91] M. Pourreza and D. Rafiei, "Din-sql: Decomposed in-context learning of text-to-sql with self-correction," 2023. [Online]. Available: https://arxiv.org/abs/2304.11015
- [92] R. Wang, Y. Yuan, S. Sun, and J. Bian, "Text-to-cad generation through infusing visual feedback in large language models," 2025. [Online]. Available: https://arxiv.org/abs/2501.19054
- [93] H. Xie and F. Ju, "Text-to-cadquery: A new paradigm for cad generation with scalable large model capabilities," 2025. [Online]. Available: https://arxiv.org/abs/2505.06507
- [94] Z. Zhang, S. Sun, W. Wang, D. Cai, and J. Bian, "Flexcad: Unified and versatile controllable cad generation with fine-tuned large language models," 2025. [Online]. Available: https://arxiv.org/abs/2411.05823
- [95] Y. Xu, M. Li, L. Cui, S. Huang, F. Wei, and M. Zhou, "Layoutlm: Pre-training of text and layout for document image understanding," 2020. [Online]. Available: https://arxiv.org/abs/1912.13318
- [96] Y. Xu, T. Lv, L. Cui, G. Wang, Y. Lu, D. Florencio, C. Zhang, and F. Wei, "Layoutxlm: Multimodal pre-training for multilingual visually-rich document understanding," 2021. [Online]. Available: https://arxiv.org/abs/2104.08836
- [97] K. Kaiyrbekov, N. J. Dobbins, and S. D. Mooney, "Automated survey collection with llm-based conversational agents," 2025. [Online]. Available: https://arxiv.org/abs/2504.02891
- [98] C. Zhang, L. Li, S. He, X. Zhang, B. Qiao, S. Qin, M. Ma, Y. Kang, Q. Lin, S. Rajmohan, D. Zhang, and Q. Zhang, "Ufo: A ui-focused agent for windows os interaction," 2024. [Online]. Available: https://arxiv.org/abs/2402.07939
- [99] W. Hong, W. Wang, Q. Lv, J. Xu, W. Yu, J. Ji, Y. Wang, Z. Wang, Y. Zhang, J. Li, B. Xu, Y. Dong, M. Ding, and J. Tang, "Cogagent: A visual language model for gui agents," 2024. [Online]. Available: https://arxiv.org/abs/2312.08914
- [100] I. Gur, H. Furuta, A. Huang, M. Safdari, Y. Matsuo, D. Eck, and A. Faust, "A real-world webagent with planning, long context understanding, and program synthesis," 2024. [Online]. Available: https://arxiv.org/abs/2307.12856
- [101] Z. Zhang, T. Fang, K. Ma, W. Yu, H. Zhang, H. Mi, and D. Yu, "Enhancing web agents with explicit rollback mechanisms," 2025. [Online]. Available: https://arxiv.org/abs/2504.11788
- [102] T. Xie, F. Zhou, Z. Cheng, P. Shi, L. Weng, Y. Liu, T. J. Hua, J. Zhao, Q. Liu, C. Liu, L. Z. Liu, Y. Xu, H. Su, D. Shin, C. Xiong, and T. Yu, "Openagents: An open platform for language agents in the wild," 2023. [Online]. Available: https://arxiv.org/abs/2310.10634
- [103] H. Wen, H. Wang, J. Liu, and Y. Li, "Droidbot-gpt: Gpt-powered ui automation for android," 2024. [Online]. Available: https://arxiv.org/abs/2304.07061
- [104] J. A. Chudziak and M. Wawer, "Elliottagents: A natural language-driven multi-agent system for stock market analysis and prediction," 2025. [Online]. Available: https://arxiv.org/abs/2507.03435
- [105] Y. Zheng, X. Du, L. Liao, X. Zhao, Z. Zhou, J. Song, B. Zhang, J. Liu, X. Qi, Z. Li, Z. Zhang, W. Wang, and P. Zhang, "Agentar-fin-r1:

- Enhancing financial intelligence through domain expertise, training efficiency, and advanced reasoning," 2025. [Online]. Available: https://arxiv.org/abs/2507.16802
- [106] A. M. Bran, S. Cox, O. Schilter, C. Baldassari, A. D. White, and P. Schwaller, "Chemcrow: Augmenting large-language models with chemistry tools," 2023. [Online]. Available: https://arxiv.org/abs/2304. 05376
- [107] D. Rose, C.-C. Hung, M. Lepri, I. Alqassem, K. Gashteovski, and C. Lawrence, "Meddxagent: A unified modular agent framework for explainable automatic differential diagnosis," 2025. [Online]. Available: https://arxiv.org/abs/2502.19175
- [108] J. Baek, S. K. Jauhar, S. Cucerzan, and S. J. Hwang, "Researchagent: Iterative research idea generation over scientific literature with large language models," 2025. [Online]. Available: https://arxiv.org/abs/2404. 07738
- [109] M. Tufano, A. Agarwal, J. Jang, R. Z. Moghaddam, and N. Sundaresan, "Autodev: Automated ai-driven development," 2024. [Online]. Available: https://arxiv.org/abs/2403.08299
- [110] Y. Dong, X. Jiang, Z. Jin, and G. Li, "Self-collaboration code generation via chatgpt," 2024. [Online]. Available: https://arxiv.org/abs/2304.07590
- [111] Z. Chen, X. Tang, G. Deng, F. Wu, J. Wu, Z. Jiang, V. Prasanna, A. Cohan, and X. Wang, "Locagent: Graph-guided llm agents for code localization," 2025. [Online]. Available: https://arxiv.org/abs/2503.09089
- [112] G. Deng, Y. Liu, V. Mayoral-Vilches, P. Liu, Y. Li, Y. Xu, T. Zhang, Y. Liu, M. Pinzger, and S. Rass, "Pentestgpt: An Ilm-empowered automatic penetration testing tool," 2024. [Online]. Available: https://arxiv.org/abs/2308.06782
- [113] L. Muzsai, D. Imolai, and A. Lukács, "Hacksynth: Llm agent and evaluation framework for autonomous penetration testing," 2024. [Online]. Available: https://arxiv.org/abs/2412.01778
- [114] B. Li, K. Mellou, B. Zhang, J. Pathuri, and I. Menache, "Large language models for supply chain optimization," 2023. [Online]. Available: https://arxiv.org/abs/2307.03875
- [115] Y. Yamada, R. T. Lange, C. Lu, S. Hu, C. Lu, J. Foerster, J. Clune, and D. Ha, "The ai scientist-v2: Workshop-level automated scientific discovery via agentic tree search," 2025. [Online]. Available: https://arxiv.org/abs/2504.08066
- [116] S. Jia, C. Zhang, and V. Fung, "Llmatdesign: Autonomous materials discovery with large language models," 2024. [Online]. Available: https://arxiv.org/abs/2406.13163
- [117] K. Le, T. Hua, and N. V. Chawla, "Agentdrug: Utilizing large language models in an agentic workflow for zero-shot molecular optimization," 2025. [Online]. Available: https://arxiv.org/abs/2410.13147
- [118] R. Averly, F. N. Baker, and X. Ning, "Liddia: Language-based intelligent drug discovery agent," 2025. [Online]. Available: https://arxiv.org/abs/2502.13959
- [119] K. Darvish, M. Skreta, Y. Zhao, N. Yoshikawa, S. Som, M. Bogdanovic, Y. Cao, H. Hao, H. Xu, A. Aspuru-Guzik, A. Garg, and F. Shkurti, "Organa: A robotic assistant for automated chemistry experimentation and characterization," 2025. [Online]. Available: https://arxiv.org/abs/2401.06949
- [120] H. Fakhruldeen, G. Pizzuto, J. Glowacki, and A. I. Cooper, "Archemist: Autonomous robotic chemistry system architecture," 2022. [Online]. Available: https://arxiv.org/abs/2204.13571
- [121] D. Driess, F. Xia, M. S. M. Sajjadi, C. Lynch, A. Chowdhery, B. Ichter, A. Wahid, J. Tompson, Q. Vuong, T. Yu, W. Huang, Y. Chebotar, P. Sermanet, D. Duckworth, S. Levine, V. Vanhoucke, K. Hausman, M. Toussaint, K. Greff, A. Zeng, I. Mordatch, and P. Florence, "Palm-e: An embodied multimodal language model," 2023. [Online]. Available: https://arxiv.org/abs/2303.03378
- [122] M. Zawalski, W. Chen, K. Pertsch, O. Mees, C. Finn, and S. Levine, "Robotic control via embodied chain-of-thought reasoning," 2025. [Online]. Available: https://arxiv.org/abs/2407.08693
- [123] H. Sun, Y. Zhuang, L. Kong, B. Dai, and C. Zhang, "Adaplanner: Adaptive planning from feedback with language models," 2023. [Online]. Available: https://arxiv.org/abs/2305.16653
- [124] Z. Wu, Z. Wang, X. Xu, J. Lu, and H. Yan, "Embodied task planning with large language models," 2023. [Online]. Available: https://arxiv.org/abs/2307.01848
- [125] H. Marah and M. Challenger, "MADTwin: a framework for multi-agent digital twin development: smart warehouse case study," *Annals of Mathematics and Artificial Intelligence*, vol. 92, no. 4, pp. 975–1005, 2024, published online: 25 July 2023. [Online]. Available: https://doi.org/10.1007/S10472-023-09872-Z
- [126] Y. Quan and Z. Liu, "Invagent: A large language model based multi-agent system for inventory management in supply chains," 2025. [Online]. Available: https://arxiv.org/abs/2407.11384

- [127] J. Yang, Y. Wang, X. Wang, X. Wang, X. Wang, and F.-Y. Wang, "Generative AI empowering parallel manufacturing: Building a "6s" collaborative production ecology for manufacturing 5.0," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 54, no. 11, pp. 6522–6536, nov 2024, early online publication: 30 January 2024.
- [128] Y. Xiao, E. Sun, D. Luo, and W. Wang, "Tradingagents: Multiagents Ilm financial trading framework," 2025. [Online]. Available: https://arxiv.org/abs/2412.20138
- [129] X. Li, Y. Zeng, X. Xing, J. Xu, and X. Xu, "Hedgeagents: A balanced-aware multi-agent financial trading system," 2025. [Online]. Available: https://arxiv.org/abs/2502.13165
- [130] C. Xu, Z. Liu, and Z. Li, "Finarena: A human-agent collaboration framework for financial market analysis and forecasting," 2025. [Online]. Available: https://arxiv.org/abs/2503.02692
- [131] H. Yao, L. Da, V. Nandam, J. Turnau, Z. Liu, L. Pang, and H. Wei, "Comal: Collaborative multi-agent large language models for mixed-autonomy traffic," 2025. [Online]. Available: https://arxiv.org/abs/2410.14368
- [132] Z. Yuan, S. Lai, and H. Liu, "Collmlight: Cooperative large language model agents for network-wide traffic signal control," 2025. [Online]. Available: https://arxiv.org/abs/2503.11739
- [133] N. Bougie and N. Watanabe, "CitySim: Modeling urban behaviors and city dynamics with large-scale LLM-driven agent simulation," 2025, submitted on 26 June 2025 (v1). [Online]. Available: https://arxiv.org/abs/2506.21805
- [134] S. Yuan, K. Song, J. Chen, X. Tan, D. Li, and D. Yang, "EvoAgent: Towards automatic multi-agent generation via evolutionary algorithms," in *Proceedings of the 2025 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL 2025)*. Association for Computational Linguistics, 2025, arXiv v3: submitted 20 June 2024 (v1), revised 11 July 2024 (v2), July 2024 (v2), last revised 10 March 2025 (v3); Resource code: https://github.com/[repository_path]. [Online]. Available: https://arxiv.org/abs/2406.14228
- [135] Y. Huang, X. Wang, H. Liu, F. Kong, A. Qin, M. Tang, S.-C. Zhu, M. Bi, S. Qi, and X. Feng, "AdaSociety: An adaptive environment with social structures for multi-agent decision-making," in NeurIPS 2024 Workshop on Datasets and Benchmarks (D&B), dec 2024, version 5: submitted 6 November 2024 (v1), last revised 29 January 2025 (v5); Workshop paper accepted at NeurIPS D&B 2024; Code: https://github.com/[repository_path]. [Online]. Available: https://arxiv.org/abs/2411.03865
- [136] Y. Hu, Y. Wang, and J. McAuley, "Evaluating memory in Ilm agents via incremental multi-turn interactions," 2025. [Online]. Available: https://arxiv.org/abs/2507.05257
- [137] K. Yadav, Y. Ali, G. Gupta, Y. Gal, and Z. Kira, "Findingdory: A benchmark to evaluate memory in embodied agents," 2025. [Online]. Available: https://arxiv.org/abs/2506.15635
- [138] W. Hu, Y. Hong, Y. Wang, L. Gao, Z. Wei, X. Yao, N. Peng, Y. Bitton, I. Szpektor, and K.-W. Chang, "3dllm-mem: Long-term spatial-temporal memory for embodied 3d large language model," 2025. [Online]. Available: https://arxiv.org/abs/2505.22657
- [139] R. Y. Pang, A. Parrish, N. Joshi, N. Nangia, J. Phang, A. Chen, V. Padmakumar, J. Ma, J. Thompson, H. He, and S. R. Bowman, "Quality: Question answering with long input texts, yes!" 2022. [Online]. Available: https://arxiv.org/abs/2112.08608
- [140] M. Zhong, D. Yin, T. Yu, A. Zaidi, M. Mutuma, R. Jha, A. H. Awadallah, A. Celikyilmaz, Y. Liu, X. Qiu, and D. Radev, "Qmsum: A new benchmark for query-based multi-domain meeting summarization," 2021. [Online]. Available: https://arxiv.org/abs/2104.05938
- [141] J. Zheng, X. Cai, Q. Li, D. Zhang, Z. Li, Y. Zhang, L. Song, and Q. Ma, "Lifelongagentbench: Evaluating Ilm agents as lifelong learners," 2025. [Online]. Available: https://arxiv.org/abs/2505.11942
- [142] K. Cobbe, V. Kosaraju, M. Bavarian, M. Chen, H. Jun, L. Kaiser, M. Plappert, J. Tworek, J. Hilton, R. Nakano, C. Hesse, and J. Schulman, "Training verifiers to solve math word problems," 2021. [Online]. Available: https://arxiv.org/abs/2110.14168
- [143] Z. Yang, P. Qi, S. Zhang, Y. Bengio, W. W. Cohen, R. Salakhutdinov, and C. D. Manning, "Hotpotqa: A dataset for diverse, explainable multi-hop question answering," 2018. [Online]. Available: https://arxiv.org/abs/1809.09600
- [144] W. Li, W. Li, C. Shen, J. Sheng, Z. Huang, D. Wu, Y. Hua, W. Yin, X. Wang, H. Zha, and B. Jin, "Textatari: 100k frames game playing with language agents," 2025. [Online]. Available: https://arxiv.org/abs/2506.04098
- [145] R. Xiao, W. Ma, K. Wang, Y. Wu, J. Zhao, H. Wang, F. Huang, and Y. Li, "Flowbench: Revisiting and benchmarking workflow-

- guided planning for llm-based agents," 2024. [Online]. Available: https://arxiv.org/abs/2406.14884
- [146] M. Renze and E. Guven, "Self-reflection in llm agents: Effects on problem-solving performance," 2024. [Online]. Available: https://arxiv.org/abs/2405.06682
- [147] L. Li, Y. Wang, H. Zhao, S. Kong, Y. Teng, C. Li, and Y. Wang, "Reflection-bench: Evaluating epistemic agency in large language models," 2025. [Online]. Available: https://arxiv.org/abs/2410.16270
- [148] K. Stein, D. Fišer, J. Hoffmann, and A. Koller, "Automating the generation of prompts for llm-based action choice in pddl planning," 2025. [Online]. Available: https://arxiv.org/abs/2311.09830
- [149] Q. Tang, Z. Deng, H. Lin, X. Han, Q. Liang, B. Cao, and L. Sun, "Toolalpaca: Generalized tool learning for language models with 3000 simulated cases," 2023. [Online]. Available: https://arxiv.org/abs/2306.05301
- [150] M. Li, Y. Zhao, B. Yu, F. Song, H. Li, H. Yu, Z. Li, F. Huang, and Y. Li, "Api-bank: A comprehensive benchmark for tool-augmented llms," 2023. [Online]. Available: https://arxiv.org/abs/2304.08244
- [151] M. Wu, T. Zhu, H. Han, C. Tan, X. Zhang, and W. Chen, "Seal-tools: Self-instruct tool learning dataset for agent tuning and detailed benchmark," 2024. [Online]. Available: https://arxiv.org/abs/2405.08355
- [152] Z. Guo, S. Cheng, H. Wang, S. Liang, Y. Qin, P. Li, Z. Liu, M. Sun, and Y. Liu, "Stabletoolbench: Towards stable large-scale benchmarking on tool learning of large language models," 2025. [Online]. Available: https://arxiv.org/abs/2403.07714
- [153] Y. Ruan, H. Dong, A. Wang, S. Pitis, Y. Zhou, J. Ba, Y. Dubois, C. J. Maddison, and T. Hashimoto, "Identifying the risks of lm agents with an lm-emulated sandbox," 2024. [Online]. Available: https://arxiv.org/abs/2309.15817
- [154] R. Müller, "Semantic context for tool orchestration," 2025. [Online]. Available: https://arxiv.org/abs/2507.10820
- [155] X. Liu, H. Yu, H. Zhang, Y. Xu, X. Lei, H. Lai, Y. Gu, H. Ding, K. Men, K. Yang, S. Zhang, X. Deng, A. Zeng, Z. Du, C. Zhang, S. Shen, T. Zhang, Y. Su, H. Sun, M. Huang, Y. Dong, and J. Tang, "Agentbench: Evaluating Ilms as agents," 2023. [Online]. Available: https://arxiv.org/abs/2308.03688
- [156] F. F. Xu, Y. Song, B. Li, Y. Tang, K. Jain, M. Bao, Z. Z. Wang, X. Zhou, Z. Guo, M. Cao, M. Yang, H. Y. Lu, A. Martin, Z. Su, L. Maben, R. Mehta, W. Chi, L. Jang, Y. Xie, S. Zhou, and G. Neubig, "Theagentcompany: Benchmarking llm agents on consequential real world tasks," 2025. [Online]. Available: https://arxiv.org/abs/2412.14161
- [157] K.-H. Huang, A. Prabhakar, S. Dhawan, Y. Mao, H. Wang, S. Savarese, C. Xiong, P. Laban, and C.-S. Wu, "Crmarena: Understanding the capacity of llm agents to perform professional crm tasks in realistic environments," 2025. [Online]. Available: https://arxiv.org/abs/2411.02305
- [158] W. Zhang, L. Zeng, Y. Xiao, Y. Li, C. Cui, Y. Zhao, R. Hu, Y. Liu, Y. Zhou, and B. An, "Agentorchestra: A hierarchical multi-agent framework for general-purpose task solving," 2025. [Online]. Available: https://arxiv.org/abs/2506.12508
- [159] D. Patel, S. Lin, J. Rayfield, N. Zhou, R. Vaculin, N. Martinez, F. O'donncha, and J. Kalagnanam, "Assetopsbench: Benchmarking ai agents for task automation in industrial asset operations and maintenance," 2025. [Online]. Available: https://arxiv.org/abs/2506.03828
- [160] S. Zhou, F. F. Xu, H. Zhu, X. Zhou, R. Lo, A. Sridhar, X. Cheng, T. Ou, Y. Bisk, D. Fried, U. Alon, and G. Neubig, "Webarena: A realistic web environment for building autonomous agents," 2024. [Online]. Available: https://arxiv.org/abs/2307.13854
- [161] H. He, W. Yao, K. Ma, W. Yu, Y. Dai, H. Zhang, Z. Lan, and D. Yu, "Webvoyager: Building an end-to-end web agent with large multimodal models," 2024. [Online]. Available: https://arxiv.org/abs/2401.13919
- [162] S. Tian, Z. Zhang, L. Chen, and Z. Liu, "Mmina: Benchmarking multihop multimodal internet agents," 2025. [Online]. Available: https://arxiv.org/abs/2404.09992
- [163] I. Levy, B. Wiesel, S. Marreed, A. Oved, A. Yaeli, and S. Shlomov, "St-webagentbench: A benchmark for evaluating safety and trustworthiness in web agents," 2025. [Online]. Available: https://arxiv.org/abs/2410.06703
- [164] Y. Pan, D. Kong, S. Zhou, C. Cui, Y. Leng, B. Jiang, H. Liu, Y. Shang, S. Zhou, T. Wu, and Z. Wu, "Webcanvas: Benchmarking web agents in online environments," 2024. [Online]. Available: https://arxiv.org/abs/2406.12373
- [165] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W. tau Yih, T. Rocktäschel, S. Riedel, and D. Kiela, "Retrieval-augmented generation for knowledge-intensive nlp tasks," 2021. [Online]. Available: https://arxiv.org/abs/2005.11401

- [166] K. Zhang, J. Li, G. Li, X. Shi, and Z. Jin, "Codeagent: Enhancing code generation with tool-integrated agent systems for real-world repo-level coding challenges," 2024. [Online]. Available: https://arxiv.org/abs/2401.07339
- [167] Z. Gou, Z. Shao, Y. Gong, Y. Shen, Y. Yang, M. Huang, N. Duan, and W. Chen, "Tora: A tool-integrated reasoning agent for mathematical problem solving," 2024. [Online]. Available: https://arxiv.org/abs/2309.17452
- [168] G. Wang, Y. Xie, Y. Jiang, A. Mandlekar, C. Xiao, Y. Zhu, L. Fan, and A. Anandkumar, "Voyager: An open-ended embodied agent with large language models," 2023. [Online]. Available: https://arxiv.org/abs/2305.16291
- [169] N. Shinn, F. Cassano, E. Berman, A. Gopinath, K. Narasimhan, and S. Yao, "Reflexion: Language agents with verbal reinforcement learning," 2023. [Online]. Available: https://arxiv.org/abs/2303.11366
- [170] W. Yao, S. Heinecke, J. C. Niebles, Z. Liu, Y. Feng, L. Xue, R. Murthy, Z. Chen, J. Zhang, D. Arpit, R. Xu, P. Mui, H. Wang, C. Xiong, and S. Savarese, "Retroformer: Retrospective large language agents with policy gradient optimization," 2024. [Online]. Available: https://arxiv.org/abs/2308.02151
- [171] A. Zhao, D. Huang, Q. Xu, M. Lin, Y.-J. Liu, and G. Huang, "Expel: Llm agents are experiential learners," 2024. [Online]. Available: https://arxiv.org/abs/2308.10144
- [172] L. Zheng, R. Wang, X. Wang, and B. An, "Synapse: Trajectory-as-exemplar prompting with memory for computer control," 2024.
 [Online]. Available: https://arxiv.org/abs/2306.07863
- [173] Y. Shao, L. Li, J. Dai, and X. Qiu, "Character-Ilm: A trainable agent for role-playing," 2023. [Online]. Available: https://arxiv.org/abs/2310. 10158
- [174] H. Wang, C. Liu, N. Xi, Z. Qiang, S. Zhao, B. Qin, and T. Liu, "Huatuo: Tuning llama model with chinese medical knowledge," 2023. [Online]. Available: https://arxiv.org/abs/2304.06975
- [175] H. Xiong, S. Wang, Y. Zhu, Z. Zhao, Y. Liu, L. Huang, Q. Wang, and D. Shen, "Doctorglm: Fine-tuning your chinese doctor is not a herculean task," 2023. [Online]. Available: https://arxiv.org/abs/2304.01097
- [176] Z. Liu, A. Zhong, Y. Li, L. Yang, C. Ju, Z. Wu, C. Ma, P. Shu, C. Chen, S. Kim, H. Dai, L. Zhao, L. Sun, D. Zhu, J. Liu, W. Liu, D. Shen, X. Li, Q. Li, and T. Liu, "Radiology-gpt: A large language model for radiology," 2024. [Online]. Available: https://arxiv.org/abs/2306.08666
- [177] Y. Yang, Y. Tang, and K. Y. Tam, "Investlm: A large language model for investment using financial domain instruction tuning," 2023. [Online]. Available: https://arxiv.org/abs/2309.13064
- [178] Y. Li, X. Ma, S. Lu, K. Lee, X. Liu, and C. Guo, "Mend: Meta demonstration distillation for efficient and effective in-context learning," 2024. [Online]. Available: https://arxiv.org/abs/2403.06914
- [179] N. D. Cao, W. Aziz, and I. Titov, "Editing factual knowledge in language models," 2021. [Online]. Available: https://arxiv.org/abs/2104.08164
- [180] J. Tack, J. Kim, E. Mitchell, J. Shin, Y. W. Teh, and J. R. Schwarz, "Online adaptation of language models with a memory of amortized contexts," 2024. [Online]. Available: https://arxiv.org/abs/2403.04317
- [181] S. Mao, X. Wang, M. Wang, Y. Jiang, P. Xie, F. Huang, and N. Zhang, "Editing personality for large language models," 2024. [Online]. Available: https://arxiv.org/abs/2310.02168
- [182] G. Lee, V. Hartmann, J. Park, D. Papailiopoulos, and K. Lee, "Prompted llms as chatbot modules for long open-domain conversation," 2023. [Online]. Available: https://arxiv.org/abs/2305.04533
- [183] P. Chhikara, D. Khant, S. Aryan, T. Singh, and D. Yadav, "Mem0: Building production-ready ai agents with scalable long-term memory," 2025. [Online]. Available: https://arxiv.org/abs/2504.19413
- [184] C. Gao, X. Lan, Z. Lu, J. Mao, J. Piao, H. Wang, D. Jin, and Y. Li, "S³: Social-network simulation system with large language model-empowered agents," 2025. [Online]. Available: https://arxiv.org/abs/2307.14984
- [185] Y. Li, L. Sun, and Y. Zhang, "Metaagents: Large language model based agents for decision-making on teaming," 2025. [Online]. Available: https://arxiv.org/abs/2310.06500
- [186] Y.-D. Tsai, M. Liu, and H. Ren, "Rtlfixer: Automatically fixing rtl syntax errors with large language models," 2024. [Online]. Available: https://arxiv.org/abs/2311.16543
- [187] Z. Xiong, Y. Lin, W. Xie, P. He, J. Tang, H. Lakkaraju, and Z. Xiang, "How memory management impacts Ilm agents: An empirical study of experience-following behavior," 2025. [Online]. Available: https://arxiv.org/abs/2505.16067
- [188] B. Wang, W. He, S. Zeng, Z. Xiang, Y. Xing, J. Tang, and P. He, "Unveiling privacy risks in llm agent memory," 2025. [Online]. Available: https://arxiv.org/abs/2502.13172

- [189] L. Wang, W. Xu, Y. Lan, Z. Hu, Y. Lan, R. K.-W. Lee, and E.-P. Lim, "Plan-and-solve prompting: Improving zero-shot chain-of-thought reasoning by large language models," 2023. [Online]. Available: https://arxiv.org/abs/2305.04091
- [190] L. Guan, K. Valmeekam, S. Sreedharan, and S. Kambhampati, "Leveraging pre-trained large language models to construct and utilize world models for model-based task planning," 2023. [Online]. Available: https://arxiv.org/abs/2305.14909
- [191] Z. Yang, A. Ishay, and J. Lee, "Coupling large language models with logic programming for robust and general reasoning from text," 2023. [Online]. Available: https://arxiv.org/abs/2307.07696
- [192] M. Besta, N. Blach, A. Kubicek, R. Gerstenberger, M. Podstawski, L. Gianinazzi, J. Gajda, T. Lehmann, H. Niewiadomski, P. Nyczyk, and T. Hoefler, "Graph of thoughts: Solving elaborate problems with large language models," 2024. [Online]. Available: https://arxiv.org/abs/2308.09687
- [193] M. Świechowski, K. Godlewski, B. Sawicki, and J. Mańdziuk, "Monte carlo tree search: A review of recent modifications and applications," *Artificial Intelligence Review*, vol. 56, pp. 2497–2562, 2023, published online: July 19, 2022.
- [194] A. Madaan, N. Tandon, P. Gupta, S. Hallinan, L. Gao, S. Wiegreffe, U. Alon, N. Dziri, S. Prabhumoye, Y. Yang, S. Gupta, B. P. Majumder, K. Hermann, S. Welleck, A. Yazdanbakhsh, and P. Clark, "Self-refine: Iterative refinement with self-feedback," 2023. [Online]. Available: https://arxiv.org/abs/2303.17651
- [195] Y. Zhou, J. Zhao, Y. Zhang, B. Wang, S. Wang, L. Chen, J. Wang, H. Chen, A. Jie, X. Zhang, H. Wang, L. Trung, R. Ye, P. N. Hoang, H. Zhang, P. Sun, and H. Li, "Solving formal math problems by decomposition and iterative reflection," 2025. [Online]. Available: https://arxiv.org/abs/2507.15225
- [196] Y. Farag, S. Stoyanchev, M. Li, S. Keizer, and R. Doddipatla, "Conditional multi-stage failure recovery for embodied agents," 2025. [Online]. Available: https://arxiv.org/abs/2507.06016
- [197] J. A. Chudziak and A. Kostka, "Ai-powered math tutoring: Platform for personalized and adaptive education," 2025. [Online]. Available: https://arxiv.org/abs/2507.12484
- [198] L. Vanhée, M. Borit, P.-O. Siebers, R. Cremades, C. Frantz, Önder Gürcan, F. Kalvas, D. R. Kera, V. Nallur, K. Narasimhan, and M. Neumann, "Large language models for agent-based modelling: Current and possible uses across the modelling cycle," 2025. [Online]. Available: https://arxiv.org/abs/2507.05723
- [199] G. Dagan, F. Keller, and A. Lascarides, "Dynamic planning with a llm," 2023. [Online]. Available: https://arxiv.org/abs/2308.06391
- [200] D. Moshkovich and S. Zeltyn, "Taming uncertainty via automation: Observing, analyzing, and optimizing agentic ai systems," 2025. [Online]. Available: https://arxiv.org/abs/2507.11277
- [201] H. Atta, K. Huang, M. Bhatt, K. Ahmed, M. A. U. Haq, and Y. Mehmood, "Logic layer prompt control injection (lpci): A novel security vulnerability class in agentic systems," 2025. [Online]. Available: https://arxiv.org/abs/2507.10457
- [202] H. Yang, Y. Pan, J. Xu, and K. Liu, "Amico: An event-driven modular framework for persistent and embedded autonomy," 2025. [Online]. Available: https://arxiv.org/abs/2507.14513
- [203] S. Yang, X. Yu, R. Li, J. Zhu, Z. Zhao, and H. Zhang, "Airllm: Diffusion policy-based adaptive lora for remote fine-tuning of llm over the air," 2025. [Online]. Available: https://arxiv.org/abs/2507.11515
- [204] O. Eberle, T. McGee, H. Giaffar, T. Webb, and I. Momennejad, "Position: We need an algorithmic understanding of generative ai," 2025. [Online]. Available: https://arxiv.org/abs/2507.07544
- [205] H. Sun and S. Zeng, "Introspection of thought helps ai agents," 2025. [Online]. Available: https://arxiv.org/abs/2507.08664
- [206] Y. Liang, C. Wu, T. Song, W. Wu, Y. Xia, Y. Liu, Y. Ou, S. Lu, L. Ji, S. Mao, Y. Wang, L. Shou, M. Gong, and N. Duan, "Taskmatrix.ai: Completing tasks by connecting foundation models with millions of apis," 2023. [Online]. Available: https://arxiv.org/abs/2303.16434
- [207] Q. Xiong, Y. Huang, Z. Jiang, Z. Chang, Y. Zheng, T. Li, and M. Li, "Butterfly effects in toolchains: A comprehensive analysis of failed parameter filling in llm tool-agent systems," 2025. [Online]. Available: https://arxiv.org/abs/2507.15296
- [208] I. Hardgrove and J. D. Hastings, "LibImfuzz: Llm-augmented fuzz target generation for black-box libraries," 2025. [Online]. Available: https://arxiv.org/abs/2507.15058
- [209] J. Guo, C. Wang, D. Deluca, J. Liu, Z. Zhang, and X. Zhang, "Bugscope: Learn to find bugs like human," 2025. [Online]. Available: https://arxiv.org/abs/2507.15671

- [210] N. Wischermann, C. M. Verdun, G. Poesia, and F. Noseda, "Proofcompass: Enhancing specialized provers with llm guidance," 2025. [Online]. Available: https://arxiv.org/abs/2507.14335
- [211] N. Wu, J. Wang, W. Zhao, C. Yu, Z. Xiu, and D. Dai, "Orthoinsight: Rib fracture diagnosis and report generation based on multi-modal large models," 2025. [Online]. Available: https://arxiv.org/abs/2507.13993
- [212] K. Zhang, H. Zhang, G. Li, J. Li, Z. Li, and Z. Jin, "Toolcoder: Teach code generation models to use api search tools," 2023. [Online]. Available: https://arxiv.org/abs/2305.04032
- [213] J. Ye, S. Li, G. Li, C. Huang, S. Gao, Y. Wu, Q. Zhang, T. Gui, and X. Huang, "Toolsword: Unveiling safety issues of large language models in tool learning across three stages," 2024. [Online]. Available: https://arxiv.org/abs/2402.10753
- [214] Q. Zhan, Z. Liang, Z. Ying, and D. Kang, "Injecagent: Benchmarking indirect prompt injections in tool-integrated large language model agents," 2024. [Online]. Available: https://arxiv.org/abs/2403.02691
- [215] S. Hao, T. Liu, Z. Wang, and Z. Hu, "Toolkengpt: Augmenting frozen language models with massive tools via tool embeddings," 2024. [Online]. Available: https://arxiv.org/abs/2305.11554
- [216] A. Buonanno, A. Rivetti, F. A. N. Palmieri, G. D. Gennaro, and G. Romano, "Probing information distribution in transformer architectures through entropy analysis," 2025. [Online]. Available: https://arxiv.org/abs/2507.15347
- [217] N. Koriagin, Y. Aksenov, D. Laptev, G. Gerasimov, N. Balagansky, and D. Gavrilov, "Teach old saes new domain tricks with boosting," 2025. [Online]. Available: https://arxiv.org/abs/2507.12990
- [218] R. Wang, R. A. Genadi, B. E. Bouardi, Y. Wang, F. Koto, Z. Liu, T. Baldwin, and H. Li, "Agentfly: Extensible and scalable reinforcement learning for lm agents," 2025. [Online]. Available: https://arxiv.org/abs/2507.14897
- [219] Z. Tao, J. Wu, W. Yin, J. Zhang, B. Li, H. Shen, K. Li, L. Zhang, X. Wang, Y. Jiang, P. Xie, F. Huang, and J. Zhou, "Webshaper: Agentically data synthesizing via information-seeking formalization," 2025. [Online]. Available: https://arxiv.org/abs/2507.15061
- [220] V. Zhong, C. Xiong, and R. Socher, "Seq2sql: Generating structured queries from natural language using reinforcement learning," 2017. [Online]. Available: https://arxiv.org/abs/1709.00103
- [221] F. Lei, J. Chen, Y. Ye, R. Cao, D. Shin, H. Su, Z. Suo, H. Gao, W. Hu, P. Yin, V. Zhong, C. Xiong, R. Sun, Q. Liu, S. Wang, and T. Yu, "Spider 2.0: Evaluating language models on real-world enterprise text-to-sql workflows," 2025. [Online]. Available: https://arxiv.org/abs/2411.07763
- [222] J. Li, B. Hui, G. Qu, J. Yang, B. Li, B. Li, B. Wang, B. Qin, R. Cao, R. Geng, N. Huo, X. Zhou, C. Ma, G. Li, K. C. C. Chang, F. Huang, R. Cheng, and Y. Li, "Can llm already serve as a database interface? a big bench for large-scale database grounded text-to-sqls," 2023. [Online]. Available: https://arxiv.org/abs/2305.03111
- [223] Q. H. Nguyen, P. A. Trinh, P. Q. H. Mai, and T. P. Trinh, "Finstat2sql: A text2sql pipeline for financial statement analysis," 2025. [Online]. Available: https://arxiv.org/abs/2506.23273
- [224] M. Yavartanoo, S. Hong, R. Neshatavar, and K. M. Lee, "Text2cad: Text to 3d cad generation via technical drawings," 2024. [Online]. Available: https://arxiv.org/abs/2411.06206
- [225] Y. Guan, X. Wang, X. Ming, J. Zhang, D. Xu, and Q. Yu, "Cad-coder: Text-to-cad generation with chain-of-thought and geometric reward," 2025. [Online]. Available: https://arxiv.org/abs/2505.19713
- [226] P. Govindarajan, D. Baldelli, J. Pathak, Q. Fournier, and S. Chandar, "Cadmium: Fine-tuning code language models for text-driven sequential cad design," 2025. [Online]. Available: https://arxiv.org/abs/2507.09792
- [227] J. Li, W. Ma, X. Li, Y. Lou, G. Zhou, and X. Zhou, "Cad-llama: Leveraging large language models for computer-aided design parametric 3d model generation," 2025. [Online]. Available: https://arxiv.org/abs/2505.04481
- [228] J. Xu, Z. Zhao, C. Wang, W. Liu, Y. Ma, and S. Gao, "Cad-mllm: Unifying multimodality-conditioned cad generation with mllm," 2025. [Online]. Available: https://arxiv.org/abs/2411.04954
- [229] C. Lee, "Enhancing phishing email identification with large language models," 2025. [Online]. Available: https://arxiv.org/abs/2502.04759
- [230] Y. Guan, D. Wang, Z. Chu, S. Wang, F. Ni, R. Song, L. Li, J. Gu, and C. Zhuang, "Intelligent virtual assistants with llm-based process automation," 2023. [Online]. Available: https://arxiv.org/abs/2312.06677
- [231] K. Cheng, Q. Sun, Y. Chu, F. Xu, Y. Li, J. Zhang, and Z. Wu, "Seeclick: Harnessing gui grounding for advanced visual gui agents," 2024. [Online]. Available: https://arxiv.org/abs/2401.10935
- [232] K. Ma, H. Zhang, H. Wang, X. Pan, W. Yu, and D. Yu, "Laser: Llm agent with state-space exploration for web navigation," 2024. [Online]. Available: https://arxiv.org/abs/2309.08172

- [233] X. Deng, Y. Gu, B. Zheng, S. Chen, S. Stevens, B. Wang, H. Sun, and Y. Su, "Mind2web: Towards a generalist agent for the web," 2023. [Online]. Available: https://arxiv.org/abs/2306.06070
- [234] D. Zhang, B. Rama, J. Ni, S. He, F. Zhao, K. Chen, A. Chen, and J. Cao, "Litewebagent: The open-source suite for vlm-based web-agent applications," 2025. [Online]. Available: https://arxiv.org/abs/2503.02950
- [235] J. Shen, A. Jain, Z. Xiao, I. Amlekar, M. Hadji, A. Podolny, and A. Talwalkar, "Scribeagent: Towards specialized web agents using production-scale workflow data," 2024. [Online]. Available: https://arxiv.org/abs/2411.15004
- [236] J. Liu, J. Hao, C. Zhang, and Z. Hu, "Wepo: Web element preference optimization for llm-based web navigation," 2024. [Online]. Available: https://arxiv.org/abs/2412.10742
- [237] T. Abuelsaad, D. Akkil, P. Dey, A. Jagmohan, A. Vempaty, and R. Kokku, "Agent-e: From autonomous web navigation to foundational design principles in agentic systems," 2024. [Online]. Available: https://arxiv.org/abs/2407.13032
- [238] T. Huang, K. Basu, I. Abdelaziz, P. Kapanipathi, J. May, and M. Chen, "R2d2: Remembering, reflecting and dynamic decision making with a reflective agentic memory," 2025. [Online]. Available: https://arxiv.org/abs/2501.12485
- [239] B. Zheng, M. Y. Fatemi, X. Jin, Z. Z. Wang, A. Gandhi, Y. Song, Y. Gu, J. Srinivasa, G. Liu, G. Neubig, and Y. Su, "Skillweaver: Web agents can self-improve by discovering and honing skills," 2025. [Online]. Available: https://arxiv.org/abs/2504.07079
- [240] Z. Z. Wang, A. Gandhi, G. Neubig, and D. Fried, "Inducing programmatic skills for agentic tasks," 2025. [Online]. Available: https://arxiv.org/abs/2504.06821
- [241] J. Wang, H. Xu, H. Jia, X. Zhang, M. Yan, W. Shen, J. Zhang, F. Huang, and J. Sang, "Mobile-agent-v2: Mobile device operation assistant with effective navigation via multi-agent collaboration," 2024. [Online]. Available: https://arxiv.org/abs/2406.01014
- [242] Y. Song, Y. Bian, Y. Tang, G. Ma, and Z. Cai, "Visiontasker: Mobile task automation using vision based ui understanding and llm task planning," 2024. [Online]. Available: https://arxiv.org/abs/2312.11190
- [243] H. Wang, Z. Pan, H. Zhang, M. Liu, H. Gao, and H. V. Zhao, "Investalign: Overcoming data scarcity in aligning large language models with investor decision-making processes under herd behavior," 2025. [Online]. Available: https://arxiv.org/abs/2507.06528
- [244] X. Tang, T. Hu, M. Ye, Y. Shao, X. Yin, S. Ouyang, W. Zhou, P. Lu, Z. Zhang, Y. Zhao, A. Cohan, and M. Gerstein, "Chemagent: Self-updating library in large language models improves chemical reasoning," 2025. [Online]. Available: https://arxiv.org/abs/2501.06590
- [245] X. Dong, W. Zhu, H. Wang, X. Chen, P. Qiu, R. Yin, Y. Su, and Y. Wang, "Talk before you retrieve: Agent-led discussions for better rag in medical qa," 2025. [Online]. Available: https://arxiv.org/abs/2504.21252
- [246] A. Fallahpour, J. Ma, A. Munim, H. Lyu, and B. Wang, "Medrax: Medical reasoning agent for chest x-ray," 2025. [Online]. Available: https://arxiv.org/abs/2502.02673
- [247] D. Yang, J. Wei, M. Li, J. Liu, L. Liu, M. Hu, J. He, Y. Ju, W. Zhou, Y. Liu, and L. Zhang, "Medaide: Information fusion and anatomy of medical intents via llm-based agent collaboration," 2025. [Online]. Available: https://arxiv.org/abs/2410.12532
- [248] H. Kang and C. Xiong, "Researcharena: Benchmarking large language models' ability to collect and organize information as research agents," 2025. [Online]. Available: https://arxiv.org/abs/2406.10291
- [249] A. Ajith, M. Xia, A. Chevalier, T. Goyal, D. Chen, and T. Gao, "Litsearch: A retrieval benchmark for scientific literature search," 2024. [Online]. Available: https://arxiv.org/abs/2407.18940
- [250] O. Press, A. Hochlehnert, A. Prabhu, V. Udandarao, O. Press, and M. Bethge, "Citeme: Can language models accurately cite scientific claims?" 2024. [Online]. Available: https://arxiv.org/abs/2407.12861
- [251] S. Schmidgall, Y. Su, Z. Wang, X. Sun, J. Wu, X. Yu, J. Liu, M. Moor, Z. Liu, and E. Barsoum, "Agent laboratory: Using llm agents as research assistants," 2025. [Online]. Available: https://arxiv.org/abs/2501.04227
- [252] C. Lu, C. Lu, R. T. Lange, J. Foerster, J. Clune, and D. Ha, "The ai scientist: Towards fully automated open-ended scientific discovery," 2024. arXiv:2408.06292.
- [253] S. Xu, X. Zhang, and L. Qin, "Eduagent: Generative student agents in learning," 2024. [Online]. Available: https://arxiv.org/abs/2404.07963
- [254] Y. Chen, N. Ding, H.-T. Zheng, Z. Liu, M. Sun, and B. Zhou, "Empowering private tutoring by chaining large language models," 2024. [Online]. Available: https://arxiv.org/abs/2309.08112
- [255] G. Chen, L. Fan, Z. Gong, N. Xie, Z. Li, Z. Liu, C. Li, Q. Qu, H. Alinejad-Rokny, S. Ni, and M. Yang, "Agentcourt: Simulating court

- with adversarial evolvable lawyer agents," 2025. [Online]. Available: https://arxiv.org/abs/2408.08089
- [256] M. A. Islam, M. E. Ali, and M. R. Parvez, "Mapcoder: Multi-agent code generation for competitive problem solving," 2024. [Online]. Available: https://arxiv.org/abs/2405.11403
- [257] Y. Du, Y. Cai, Y. Zhou, C. Wang, Y. Qian, X. Pang, Q. Liu, Y. Hu, and S. Chen, "Swe-dev: Evaluating and training autonomous feature-driven software development," 2025. [Online]. Available: https://arxiv.org/abs/2505.16975
- [258] R. Bairi, A. Sonwane, A. Kanade, V. D. C, A. Iyer, S. Parthasarathy, S. Rajamani, B. Ashok, and S. Shet, "Codeplan: Repositorylevel coding using Ilms and planning," 2023. [Online]. Available: https://arxiv.org/abs/2309.12499
- [259] I. Bouzenia, P. Devanbu, and M. Pradel, "Repairagent: An autonomous, llm-based agent for program repair," 2024. [Online]. Available: https://arxiv.org/abs/2403.17134
- [260] J. Kong, M. Cheng, X. Xie, S. Liu, X. Du, and Q. Guo, "Contrastrepair: Enhancing conversation-based automated program repair via contrastive test case pairs," 2024. [Online]. Available: https://arxiv.org/abs/2403.01971
- [261] T. Abramovich, M. Udeshi, M. Shao, K. Lieret, H. Xi, K. Milner, S. Jancheska, J. Yang, C. E. Jimenez, F. Khorrami, P. Krishnamurthy, B. Dolan-Gavitt, M. Shafique, K. Narasimhan, R. Karri, and O. Press, "Enigma: Interactive tools substantially assist lm agents in finding security vulnerabilities," 2025. [Online]. Available: https://arxiv.org/abs/2409.16165
- [262] Z. Wang, Z. Liu, Y. Zhang, A. Zhong, J. Wang, F. Yin, L. Fan, L. Wu, and Q. Wen, "Reagent: Cloud root cause analysis by autonomous agents with tool-augmented large language models," 2024. [Online]. Available: https://arxiv.org/abs/2310.16340
- [263] Q. Wang, X. Zhang, M. Li, Y. Yuan, M. Xiao, F. Zhuang, and D. Yu, "Tamo:fine-grained root cause analysis via tool-assisted llm agent with multi-modality observation data in cloud-native systems," 2025. [Online]. Available: https://arxiv.org/abs/2504.20462
- [264] M. Shetty, Y. Chen, G. Somashekar, M. Ma, Y. Simmhan, X. Zhang, J. Mace, D. Vandevoorde, P. Las-Casas, S. M. Gupta, S. Nath, C. Bansal, and S. Rajmohan, "Building ai agents for autonomous clouds: Challenges and design principles," 2024. [Online]. Available: https://arxiv.org/abs/2407.12165
- [265] F. Yu, F. Yang, X. Qin, Z. Zhang, J. Zhang, Q. Lin, H. Zhang, Y. Dang, S. Rajmohan, D. Zhang, and Q. Zhang, "Enabling autonomic microservice management through self-learning agents," 2025. [Online]. Available: https://arxiv.org/abs/2501.19056
- [266] K. Li, F. Liu, Z. Wang, X. Tong, X. Han, M. Yuan, and Q. Zhang, "Ars: Automatic routing solver with large language models," 2025. [Online]. Available: https://arxiv.org/abs/2502.15359
- [267] L. M. Antunes, K. T. Butler, and R. Grau-Crespo, "Crystal structure generation with autoregressive large language modeling," 2024. [Online]. Available: https://arxiv.org/abs/2307.04340
- [268] J. Ock, R. S. Meda, S. Badrinarayanan, N. S. Aluru, A. Chandrasekhar, and A. B. Farimani, "Large language model agent for modular task execution in drug discovery," 2025. [Online]. Available: https://arxiv.org/abs/2507.02925
- [269] Y. Yu, Z. Yao, H. Li, Z. Deng, Y. Cao, Z. Chen, J. W. Suchow, R. Liu, Z. Cui, Z. Xu, D. Zhang, K. Subbalakshmi, G. Xiong, Y. He, J. Huang, D. Li, and Q. Xie, "FinCon: A synthesized llm multi-agent system with conceptual verbal reinforcement for enhanced financial decision making," 2024, version 3: submitted on 9 July 2024 (v1), revised 10 July 2024 (v2), last revised 7 November 2024 (v3). [Online]. Available: https://arxiv.org/abs/2407.06567
- [270] N. Bougie and N. Watanabe, "Citysim: Modeling urban behaviors and city dynamics with large-scale llm-driven agent simulation," 2025. [Online]. Available: https://arxiv.org/abs/2506.21805
- [271] C. Papadakis, G. Filandrianos, A. Dimitriou, M. Lymperaiou, K. Thomas, and G. Stamou, "Stocksim: A dual-mode order-level simulator for evaluating multi-agent llms in financial markets," 2025. [Online]. Available: https://arxiv.org/abs/2507.09255
- [272] Y. Yang, Y. Zhang, M. Wu, K. Zhang, Y. Zhang, H. Yu, Y. Hu, and B. Wang, "Twinmarket: A scalable behavioral and social simulation for financial markets," 2025. [Online]. Available: https://arxiv.org/abs/2502.01506
- [273] J. Kleiman, K. Frank, J. Voyles, and S. Campagna, "Simulation Agent: A framework for integrating simulation and large language models for enhanced decision-making," 2025, version 2: submitted on 19 May 2025 (v1), last revised 21 May 2025 (v2). [Online]. Available: https://arxiv.org/abs/2505.13761

- [274] Y. Xia, D. Dittler, N. Jazdi, H. Chen, and M. Weyrich, "LLM experiments with simulation: Large language model multi-agent system for simulation model parametrization in digital twins," 2024, version 2: submitted on 28 May 2024 (v1), last revised 22 July 2024 (v2); Submitted to IEEE-ETFA2024 (under peerreview); Code: https://github.com/[repository_path]. [Online]. Available: https://arxiv.org/abs/2405.18092
- [275] J. Piao, Y. Yan, J. Zhang, N. Li, J. Yan, X. Lan, Z. Lu, Z. Zheng, J. Y. Wang, D. Zhou, C. Gao, F. Xu, F. Zhang, K. Rong, J. Su, and Y. Li, "Agentsociety: Large-scale simulation of llm-driven generative agents advances understanding of human behaviors and society," 2025. [Online]. Available: https://arxiv.org/abs/2502.08691
- [276] M. Riad, V. de Carvalho, and F. Golpayegani, "Multi-value alignment in normative multi-agent system: An evolutionary optimisation approach," in MODeM 2023 Workshop at the 26th European Conference on Artificial Intelligence (ECAI 2023), oct 2023, pp. 1–15, workshop paper; Substantial text overlap with arXiv:2305.07366; Submitted on 12 October 2023. [Online]. Available: https://arxiv.org/abs/2310.08362
- [277] W. Zeng, H. Zhu, C. Qin, H. Wu, Y. Cheng, S. Zhang, X. Jin, Y. Shen, Z. Wang, F. Zhong, and H. Xiong, "Multi-level value alignment in agentic AI systems: Survey and perspectives," arXiv preprint arXiv:2506.09656, aug 2025, version 2: submitted on 11 June 2025 (v1), last revised 7 August 2025 (v2); Survey paper. [Online]. Available: https://arxiv.org/abs/2506.09656
- [278] S. Shan, "Computational architects of society: Quantum machine learning for social rule genesis," arXiv preprint arXiv:2506.03503, jun 2025, submitted on 4 June 2025 (v1). [Online]. Available: https://arxiv.org/abs/2506.03503
- [279] J. Tang, H. Gao, X. Pan, L. Wang, H. Tan, D. Gao, Y. Chen, X. Chen, Y. Lin, Y. Li, B. Ding, J. Zhou, J. Wang, and J.-R. Wen, "Gensim: A general social simulation platform with large language model based agents," 2025. [Online]. Available: https://arxiv.org/abs/2410.04360
- [280] D. Gao, Z. Li, Y. Xie, W. Kuang, L. Yao, B. Qian, Z. Ma, Y. Cui, H. Luo, S. Li, L. Yi, Y. Yu, S. He, Z. Luo, W. Zhou, Z. Zhang, X. He, Z. Chen, W. Liao, F. I. Kushnazarov, Y. Li, B. Ding, and J. Zhou, "Agentscope 1.0: A developer-centric framework for building agentic applications," 2025. [Online]. Available: https://arxiv.org/abs/2508.16279
- [281] L. Wang, H. Gao, X. Bo, X. Chen, and J.-R. Wen, "Yulan-onesim: Towards the next generation of social simulator with large language models," 2025. [Online]. Available: https://arxiv.org/abs/2505.07581
- [282] A. Maharana, D.-H. Lee, S. Tulyakov, M. Bansal, F. Barbieri, and Y. Fang, "Evaluating very long-term conversational memory of llm agents," 2024. [Online]. Available: https://arxiv.org/abs/2402.17753
- [283] K.-H. Lee, X. Chen, H. Furuta, J. Canny, and I. Fischer, "A human-inspired reading agent with gist memory of very long contexts," 2024. [Online]. Available: https://arxiv.org/abs/2402.09727
- [284] C.-K. Wu, Z. R. Tam, C.-Y. Lin, Y.-N. Chen, and H. yi Lee, "Streambench: Towards benchmarking continuous improvement of language agents," 2024. [Online]. Available: https://arxiv.org/abs/2406. 08747
- [285] H. Tan, Z. Zhang, C. Ma, X. Chen, Q. Dai, and Z. Dong, "Membench: Towards more comprehensive evaluation on the memory of llm-based agents," 2025. [Online]. Available: https://arxiv.org/abs/2506.21605
- [286] J. Lin, C. Zhu, R. Xu, X. Mao, X. Liu, T. Wang, and J. Pang, "Ost-bench: Evaluating the capabilities of mllms in online spatio-temporal scene understanding," 2025. [Online]. Available: https://arxiv.org/abs/2507.07984
- [287] K. Zhu and Y. Han, "Real: Benchmarking abilities of large language models for housing transactions and services," 2025. [Online]. Available: https://arxiv.org/abs/2507.03477
- [288] N. Liu, L. Chen, X. Tian, W. Zou, K. Chen, and M. Cui, "From llm to conversational agent: A memory enhanced architecture with fine-tuning of large language models," 2024. [Online]. Available: https://arxiv.org/abs/2401.02777
- [289] W. Ling, D. Yogatama, C. Dyer, and P. Blunsom, "Program induction by rationale generation: Learning to solve and explain algebraic word problems," 2017. [Online]. Available: https://arxiv.org/abs/1705.04146
- [290] P. Clark, I. Cowhey, O. Etzioni, T. Khot, A. Sabharwal, C. Schoenick, and O. Tafjord, "Think you have solved question answering? try arc, the ai2 reasoning challenge," 2018. [Online]. Available: https://arxiv.org/abs/1803.05457
- [291] M. Geva, D. Khashabi, E. Segal, T. Khot, D. Roth, and J. Berant, "Did aristotle use a laptop? a question answering benchmark with implicit reasoning strategies," 2021. [Online]. Available: https://arxiv.org/abs/2101.02235

- [292] D. Hendrycks, C. Burns, S. Kadavath, A. Arora, S. Basart, E. Tang, D. Song, and J. Steinhardt, "Measuring mathematical problem solving with the math dataset," 2021. [Online]. Available: https://arxiv.org/abs/2103.03874
- [293] T. Ashraf, A. Saqib, H. Ghani, M. AlMahri, Y. Li, N. Ahsan, U. Nawaz, J. Lahoud, H. Cholakkal, M. Shah, P. Torr, F. S. Khan, R. M. Anwer, and S. Khan, "Agent-x: Evaluating deep multimodal reasoning in vision-centric agentic tasks," 2025. [Online]. Available: https://arxiv.org/abs/2505.24876
- [294] H. S. Zheng, S. Mishra, H. Zhang, X. Chen, M. Chen, A. Nova, L. Hou, H.-T. Cheng, Q. V. Le, E. H. Chi, and D. Zhou, "Natural plan: Benchmarking Ilms on natural language planning," 2024. [Online]. Available: https://arxiv.org/abs/2406.04520
- [295] X. Wang, Z. Wang, J. Liu, Y. Chen, L. Yuan, H. Peng, and H. Ji, "Mint: Evaluating llms in multi-turn interaction with tools and language feedback," 2024. [Online]. Available: https://arxiv.org/abs/2309.10691
- [296] C.-A. Cheng, A. Kolobov, D. Misra, A. Nie, and A. Swaminathan, "Llf-bench: Benchmark for interactive learning from language feedback," 2023. [Online]. Available: https://arxiv.org/abs/2312.06853
- [297] H. Kokel, M. Katz, K. Srinivas, and S. Sohrabi, "Acpbench: Reasoning about action, change, and planning," 2024. [Online]. Available: https://arxiv.org/abs/2410.05669
- [298] Y. Huang, D. Song, Z. Ji, S. Wang, and L. Ma, "Evaluating Ilms on sequential api call through automated test generation," 2025. [Online]. Available: https://arxiv.org/abs/2507.09481
- [299] L. Zhong, Z. Du, X. Zhang, H. Hu, and J. Tang, "Complexfunchench: Exploring multi-step and constrained function calling under long-context scenario," 2025. [Online]. Available: https://arxiv.org/abs/2501.10132
- [300] K. Basu, I. Abdelaziz, K. Kate, M. Agarwal, M. Crouse, Y. Rizk, K. Bradford, A. Munawar, S. Kumaravel, S. Goyal, X. Wang, L. A. Lastras, and P. Kapanipathi, "Nestful: A benchmark for evaluating llms on nested sequences of api calls," 2025. [Online]. Available: https://arxiv.org/abs/2409.03797
- [301] K. Jang, D. Lee, K. Kim, D. Heo, T. Lee, W. Kim, and B. Suh, "Dice-bench: Evaluating the tool-use capabilities of large language models in multi-round, multi-party dialogues," 2025. [Online]. Available: https://arxiv.org/abs/2506.22853
- [302] A. Chakraborty, P. Dashore, N. Bathaee, A. Jain, A. Das, S.-X. Zhang, S. Sahu, M. Naphade, and G. I. Winata, "T1: A tool-oriented conversational dataset for multi-turn agentic planning," 2025. [Online]. Available: https://arxiv.org/abs/2505.16986
- [303] J. Lu, T. Holleis, Y. Zhang, B. Aumayer, F. Nan, F. Bai, S. Ma, S. Ma, M. Li, G. Yin, Z. Wang, and R. Pang, "Toolsandbox: A stateful, conversational, interactive evaluation benchmark for llm tool use capabilities," 2025. [Online]. Available: https://arxiv.org/abs/2408.04682
- [304] K. Basu, I. Abdelaziz, S. Chaudhury, S. Dan, M. Crouse, A. Munawar, S. Kumaravel, V. Muthusamy, P. Kapanipathi, and L. A. Lastras, "Api-blend: A comprehensive corpora for training and benchmarking api llms," 2024. [Online]. Available: https://arxiv.org/abs/2402.15491
- [305] S. Zhao, H. Zhang, S. Lin, M. Li, Q. Wu, K. Zhang, and C. Wei, "Pyvision: Agentic vision with dynamic tooling," 2025. [Online]. Available: https://arxiv.org/abs/2507.07998
- [306] J. Qiu, X. Juan, Y. Wang, L. Yang, X. Qi, T. Zhang, J. Guo, Y. Lu, Z. Yao, H. Wang, S. Liu, X. Jiang, L. Leqi, and M. Wang, "Agentdistill: Training-free agent distillation with generalizable mcp boxes," 2025. [Online]. Available: https://arxiv.org/abs/2506.14728
- [307] N. Pinckney, C. Deng, C.-T. Ho, Y.-D. Tsai, M. Liu, W. Zhou, B. Khailany, and H. Ren, "Comprehensive verilog design problems: A next-generation benchmark dataset for evaluating large language models and agents on rtl design and verification," 2025. [Online]. Available: https://arxiv.org/abs/2506.14074
- [308] Y. Song, W. Xiong, D. Zhu, W. Wu, H. Qian, M. Song, H. Huang, C. Li, K. Wang, R. Yao, Y. Tian, and S. Li, "Restgpt: Connecting large language models with real-world restful apis," 2023. [Online]. Available: https://arxiv.org/abs/2306.06624
- [309] G. Mialon, C. Fourrier, C. Swift, T. Wolf, Y. LeCun, and T. Scialom, "Gaia: a benchmark for general ai assistants," 2023. [Online]. Available: https://arxiv.org/abs/2311.12983
- [310] Y. Fu, X. Yuan, and D. Wang, "Ras-eval: A comprehensive benchmark for security evaluation of llm agents in real-world environments," 2025. [Online]. Available: https://arxiv.org/abs/2506.15253
- [311] S. Yao, N. Shinn, P. Razavi, and K. Narasimhan, "τ-bench: A benchmark for tool-agent-user interaction in real-world domains," 2024. [Online]. Available: https://arxiv.org/abs/2406.12045
- [312] V. Barres, H. Dong, S. Ray, X. Si, and K. Narasimhan, "\u03c4^2-bench: Evaluating conversational agents in a dual-control environment," 2025. [Online]. Available: https://arxiv.org/abs/2506.07982

- [313] J. Hyun, N. R. Waytowich, and B. Chen, "Crew-wildfire: Benchmarking agentic multi-agent collaborations at scale," 2025. [Online]. Available: https://arxiv.org/abs/2507.05178
- [314] S. Nandi, A. Datta, N. Vichare, I. Bhattacharya, H. Raja, J. Xu, S. Ray, G. Carenini, A. Srivastava, A. Chan, M. H. Woo, A. Kandola, B. Theresa, and F. Carbone, "Sop-bench: Complex industrial sops for evaluating llm agents," 2025. [Online]. Available: https://arxiv.org/abs/2506.08119
- [315] H. Chen, Y. Wang, Y. Cai, H. Hu, J. Li, S. Huang, C. Deng, R. Liang, S. Kong, H. Ren, S. Samaranayake, C. P. Gomes, and Z. Zhang, "Heurigym: An agentic benchmark for llm-crafted heuristics in combinatorial optimization," 2025. [Online]. Available: https://arxiv.org/abs/2506.07972
- [316] O. Hofman, J. Brokman, O. Rachmil, S. Bose, V. Pahuja, T. Shimizu, T. Starostina, K. Marchisio, S. Goldfarb-Tarrant, and R. Vainshtein, "Maps: A multilingual benchmark for global agent performance and security," 2025. [Online]. Available: https://arxiv.org/abs/2505.15935
- [317] Z. Liu and Y. Quan, "Econwebarena: Benchmarking autonomous agents on economic tasks in realistic web environments," 2025. [Online]. Available: https://arxiv.org/abs/2506.08136
- [318] A. Ivanova, E. Bakaeva, Z. Volovikova, A. K. Kovalev, and A. I. Panov, "Ambik: Dataset of ambiguous tasks in kitchen environment," 2025. [Online]. Available: https://arxiv.org/abs/2506.04089
- [319] T. Men, Z. Jin, P. Cao, Y. Chen, K. Liu, and J. Zhao, "Agent-rewardbench: Towards a unified benchmark for reward modeling across perception, planning, and safety in real-world multimodal agents," 2025. [Online]. Available: https://arxiv.org/abs/2506.21252
- [320] E. Levi and I. Kadar, "Intellagent: A multi-agent framework for evaluating conversational ai systems," 2025. [Online]. Available: https://arxiv.org/abs/2501.11067
- [321] W. Wang, H. Wang, and X. Yan, "Steps: A benchmark for order reasoning in sequential tasks," 2023. [Online]. Available: https://arxiv.org/abs/2306.04441
- [322] B. Stroebl, S. Kapoor, and A. Narayanan, "Hal: A holistic agent leaderboard for centralized and reproducible agent evaluation," https://github.com/princeton-pli/hal-harness, 2025.
- [323] R. Cao, F. Lei, H. Wu, J. Chen, Y. Fu, H. Gao, X. Xiong, H. Zhang, Y. Mao, W. Hu, T. Xie, H. Xu, D. Zhang, S. Wang, R. Sun, P. Yin, C. Xiong, A. Ni, Q. Liu, V. Zhong, L. Chen, K. Yu, and T. Yu, "Spider2-v: How far are multimodal agents from automating data science and engineering workflows?" 2024. [Online]. Available: https://arxiv.org/abs/2407.10956
- [324] J. Y. Koh, R. Lo, L. Jang, V. Duvvur, M. C. Lim, P.-Y. Huang, G. Neubig, S. Zhou, R. Salakhutdinov, and D. Fried, "Visualwebarena: Evaluating multimodal agents on realistic visual web tasks," 2024. [Online]. Available: https://arxiv.org/abs/2401.13649
- [325] X. H. Lù, Z. Kasner, and S. Reddy, "Weblinx: Real-world website navigation with multi-turn dialogue," 2024. [Online]. Available: https://arxiv.org/abs/2402.05930
- [326] A. Drouin, M. Gasse, M. Caccia, I. H. Laradji, M. D. Verme, T. Marty, L. Boisvert, M. Thakkar, Q. Cappart, D. Vazquez, N. Chapados, and A. Lacoste, "Workarena: How capable are web agents at solving common knowledge work tasks?" 2024. [Online]. Available: https://arxiv.org/abs/2403.07718
- [327] E. Z. Liu, K. Guu, P. Pasupat, T. Shi, and P. Liang, "Reinforcement learning on web interfaces using workflow-guided exploration," 2018. [Online]. Available: https://arxiv.org/abs/1802.08802
- [328] O. Yoran, S. J. Amouyal, C. Malaviya, B. Bogin, O. Press, and J. Berant, "Assistantbench: Can web agents solve realistic and time-consuming tasks?" 2024. [Online]. Available: https://arxiv.org/abs/2407.15711
- [329] C. E. Jimenez, J. Yang, A. Wettig, S. Yao, K. Pei, O. Press, and K. Narasimhan, "Swe-bench: Can language models resolve real-world github issues?" 2024. [Online]. Available: https://arxiv.org/abs/2310. 06770
- [330] J. Yang, C. E. Jimenez, A. L. Zhang, K. Lieret, J. Yang, X. Wu, O. Press, N. Muennighoff, G. Synnaeve, K. R. Narasimhan, D. Yang, S. I. Wang, and O. Press, "Swe-bench multimodal: Do ai systems generalize to visual software domains?" 2024. [Online]. Available: https://arxiv.org/abs/2410.03859
- [331] M. S. Rashid, C. Bock, Y. Zhuang, A. Buchholz, T. Esler, S. Valentin, L. Franceschi, M. Wistuba, P. T. Sivaprasad, W. J. Kim, A. Deoras, G. Zappella, and L. Callot, "Swe-polybench: A multi-language benchmark for repository level evaluation of coding agents," 2025. [Online]. Available: https://arxiv.org/abs/2504.08703
- [332] M. Chen, J. Tworek, H. Jun, Q. Yuan, H. P. de Oliveira Pinto, J. Kaplan, H. Edwards, Y. Burda, N. Joseph, G. Brockman, A. Ray, R. Puri, G. Krueger, M. Petrov, H. Khlaaf, G. Sastry, P. Mishkin, B. Chan,

- S. Gray, N. Ryder, M. Pavlov, A. Power, L. Kaiser, M. Bavarian, C. Winter, P. Tillet, F. P. Such, D. Cummings, M. Plappert, F. Chantzis, E. Barnes, A. Herbert-Voss, W. H. Guss, A. Nichol, A. Paino, N. Tezak, J. Tang, I. Babuschkin, S. Balaji, S. Jain, W. Saunders, C. Hesse, A. N. Carr, J. Leike, J. Achiam, V. Misra, E. Morikawa, A. Radford, M. Knight, M. Brundage, M. Murati, K. Mayer, P. Welinder, B. McGrew, D. Amodei, S. McCandlish, I. Sutskever, and W. Zaremba, "Evaluating large language models trained on code," 2021. [Online]. Available: https://arxiv.org/abs/2107.03374
- [333] T. Ahmed, M. Hirzel, R. Pan, A. Shinnar, and S. Sinha, "Tdd-bench verified: Can Ilms generate tests for issues before they get resolved?" 2024. [Online]. Available: https://arxiv.org/abs/2412.02883
- [334] S. Jha, R. Arora, Y. Watanabe, T. Yanagawa, Y. Chen, J. Clark, B. Bhavya, M. Verma, H. Kumar, H. Kitahara, N. Zheutlin, S. Takano, D. Pathak, F. George, X. Wu, B. O. Turkkan, G. Vanloo, M. Nidd, T. Dai, O. Chatterjee, P. Gupta, S. Samanta, P. Aggarwal, R. Lee, P. Murali, J. wook Ahn, D. Kar, A. Rahane, C. Fonseca, A. Paradkar, Y. Deng, P. Moogi, P. Mohapatra, N. Abe, C. Narayanaswami, T. Xu, L. R. Varshney, R. Mahindru, A. Sailer, L. Shwartz, D. Sow, N. C. M. Fuller, and R. Puri, "Itbench: Evaluating ai agents across diverse real-world it automation tasks," 2025. [Online]. Available: https://arxiv.org/abs/2502.05352
- [335] S. Miserendino, M. Wang, T. Patwardhan, and J. Heidecke, "Swe-lancer: Can frontier llms earn \$ million from real-world freelance software engineering?" 2025. [Online]. Available: https://arxiv.org/abs/2502.12115
- [336] Y. Jin, C. Li, P. Fan, P. Liu, X. Li, C. Liu, and W. Qiu, "Llm-bscvm: An llm-based blockchain smart contract vulnerability management framework," 2025. [Online]. Available: https://arxiv.org/abs/2505.17416
- [337] Y. Xiao, R. Wang, L. Kong, D. Golac, and W. Wang, "Csr-bench: Benchmarking Ilm agents in deployment of computer science research repositories," 2025. [Online]. Available: https://arxiv.org/abs/2502.06111
- [338] K. Wang, N. Holzer, Z. Xia, Y. Cao, J. Gao, A. Walid, K. Xiao, and X.-Y. L. Yanglet, "Finrl contests: Benchmarking data-driven financial reinforcement learning agents," 2025. [Online]. Available: https://arxiv.org/abs/2504.02281
- [339] C. Li, Y. Shi, Y. Luo, and N. Tang, "Will llms be professional at fund investment? deepfund: A live arena perspective," 2025. [Online]. Available: https://arxiv.org/abs/2503.18313
- [340] W. W. Li, H. Kim, M. Cucuringu, and T. Ma, "Can Ilm-based financial investing strategies outperform the market in long run?" 2025. [Online]. Available: https://arxiv.org/abs/2505.07078
- [341] X. Guo, H. Xia, Z. Liu, H. Cao, Z. Yang, Z. Liu, S. Wang, J. Niu, C. Wang, Y. Wang, X. Liang, X. Huang, B. Zhu, Z. Wei, Y. Chen, W. Shen, and L. Zhang, "Fineval: A chinese financial domain knowledge evaluation benchmark for large language models," 2024. [Online]. Available: https://arxiv.org/abs/2308.09975
- [342] H. Li, Y. Cao, Y. Yu, S. R. Javaji, Z. Deng, Y. He, Y. Jiang, Z. Zhu, K. Subbalakshmi, G. Xiong, J. Huang, L. Qian, X. Peng, Q. Xie, and J. W. Suchow, "Investorbench: A benchmark for financial decision-making tasks with llm-based agent," 2024. [Online]. Available: https://arxiv.org/abs/2412.18174
- [343] R. Sun, Z. Bai, W. Zhang, Y. Zhang, L. Zhao, S. Sun, and Z. Qiu, "Finresearchbench: A logic tree based agent-as-a-judge evaluation framework for financial research agents," 2025. [Online]. Available: https://arxiv.org/abs/2507.16248
- [344] S. C. Lin, F. Tian, K. Wang, X. Zhao, J. Huang, Q. Xie, L. Borella, M. White, C. D. Wang, K. Xiao, X.-Y. L. Yanglet, and L. Deng, "Open finllm leaderboard: Towards financial ai readiness," 2025. [Online]. Available: https://arxiv.org/abs/2501.10963
- [345] Y. Zhu, Z. He, H. Hu, X. Zheng, X. Zhang, Z. Wang, J. Gao, L. Ma, and L. Yu, "Medagentboard: Benchmarking multi-agent collaboration with conventional methods for diverse medical tasks," 2025. [Online]. Available: https://arxiv.org/abs/2505.12371
- [346] Y. Liao, S. Jiang, Y. Wang, and Y. Wang, "Reflectool: Towards reflection-aware tool-augmented clinical agents," 2025. [Online]. Available: https://arxiv.org/abs/2410.17657
- [347] X. Tang, D. Shao, J. Sohn, J. Chen, J. Zhang, J. Xiang, F. Wu, Y. Zhao, C. Wu, W. Shi, A. Cohan, and M. Gerstein, "Medagentsbench: Benchmarking thinking models and agent frameworks for complex medical reasoning," 2025. [Online]. Available: https://arxiv.org/abs/2503.07459
- [348] S. Schmidgall, R. Ziaei, C. Harris, E. Reis, J. Jopling, and M. Moor, "Agentclinic: a multimodal agent benchmark to evaluate ai in simulated clinical environments," 2025. [Online]. Available: https://arxiv.org/abs/2405.07960

- [349] J. Liu, W. Wang, Z. Ma, G. Huang, Y. SU, K.-J. Chang, W. Chen, H. Li, L. Shen, and M. Lyu, "Medchain: Bridging the gap between llm agents and clinical practice through interactive sequential benchmarking," 2024. [Online]. Available: https://arxiv.org/abs/2412.01605
- [350] Y. Sun, X. Qian, W. Xu, H. Zhang, C. Xiao, L. Li, Y. Rong, W. Huang, Q. Bai, and T. Xu, "Reasonmed: A 370k multi-agent generated dataset for advancing medical reasoning," 2025. [Online]. Available: https://arxiv.org/abs/2506.09513
- [351] J.-P. Corbeil, A. B. Abacha, G. Michalopoulos, P. Swazinna, M. Del-Agua, J. Tremblay, A. J. Daniel, C. Bader, Y.-C. Cho, P. Krishnan, N. Bodenstab, T. Lin, W. Teng, F. Beaulieu, and P. Vozila, "Empowering healthcare practitioners with language models: Structuring speech transcripts in two real-world clinical applications," 2025. [Online]. Available: https://arxiv.org/abs/2507.05517
- [352] Y. Zhu, S. Wei, X. Wang, K. Xue, X. Zhang, and S. Zhang, "Menti: Bridging medical calculator and llm agent with nested tool calling," 2025. [Online]. Available: https://arxiv.org/abs/2410.13610
- [353] Z. Chen, Z. Peng, X. Liang, C. Wang, P. Liang, L. Zeng, M. Ju, and Y. Yuan, "Map: Evaluation and multi-agent enhancement of large language models for inpatient pathways," 2025. [Online]. Available: https://arxiv.org/abs/2503.13205
- [354] K. Chen, T. Zhen, H. Wang, K. Liu, X. Li, J. Huo, T. Yang, J. Xu, W. Dong, and Y. Gao, "Medsentry: Understanding and mitigating safety risks in medical llm multi-agent systems," 2025. [Online]. Available: https://arxiv.org/abs/2505.20824
- [355] Y. Xiao, J. Huang, R. He, J. Xiao, M. R. Mousavi, Y. Liu, K. Li, Z. Chen, and J. M. Zhang, "Amqa: An adversarial dataset for benchmarking bias of llms in medicine and healthcare," 2025. [Online]. Available: https://arxiv.org/abs/2505.19562
- [356] W. B. Zhu, T. Chen, C. Y. Lin, J. Law, M. Jizzini, J. J. Nieva, R. Liu, and R. Jia, "Cancer-myth: Evaluating ai chatbot on patient questions with false presuppositions," 2025. [Online]. Available: https://arxiv.org/abs/2504.11373
- [357] N. Sharma, "Cxr-agent: Vision-language models for chest x-ray interpretation with uncertainty aware radiology reporting," 2024. [Online]. Available: https://arxiv.org/abs/2407.08811
- [358] S. Li, T. Lin, L. Lin, W. Zhang, J. Liu, X. Yang, J. Li, Y. He, X. Song, J. Xiao, Y. Zhuang, and B. C. Ooi, "Eyecaregpt: Boosting comprehensive ophthalmology understanding with tailored dataset, benchmark and model," 2025. [Online]. Available: https://arxiv.org/abs/2504.13650
- [359] D. Restrepo, C. Wu, Z. Tang, Z. Shuai, T. N. M. Phan, J.-E. Ding, C.-T. Dao, J. Gallifant, R. G. Dychiao, J. C. Artiaga, A. H. Bando, C. P. B. Gracitelli, V. Ferrer, L. A. Celi, D. Bitterman, M. G. Morley, and L. F. Nakayama, "Multi-ophthalingua: A multilingual benchmark for assessing and debiasing llm ophthalmological qa in lmics," 2024. [Online]. Available: https://arxiv.org/abs/2412.14304
- [360] R. Xu, Y. Zhuang, Y. Zhong, Y. Yu, X. Tang, H. Wu, M. D. Wang, P. Ruan, D. Yang, T. Wang, G. Xiao, C. Yang, Y. Xie, and W. Shi, "Medagentgym: Training llm agents for code-based medical reasoning at scale," 2025. [Online]. Available: https://arxiv.org/abs/2506.04405
- [361] L. Moukheiber, M. Moukheiber, D. Moukheiber, J.-W. Ju, and H.-C. Lee, "Echoqa: A large collection of instruction tuning data for echocardiogram reports," 2025. [Online]. Available: https://arxiv.org/abs/2503.02365
- [362] X. Lyu, Y. Liang, W. Chen, M. Ding, J. Yang, G. Huang, D. Zhang, X. He, and L. Shen, "Wsi-agents: A collaborative multi-agent system for multi-modal whole slide image analysis," 2025. [Online]. Available: https://arxiv.org/abs/2507.14680
- [363] I. Sviridov, A. Miftakhova, A. Tereshchenko, G. Zubkova, P. Blinov, and A. Savchenko, "3mdbench: Medical multimodal multi-agent dialogue benchmark," 2025. [Online]. Available: https://arxiv.org/abs/2504.13861
- [364] J. Feng, Q. Zheng, C. Wu, Z. Zhao, Y. Zhang, Y. Wang, and W. Xie, "m³builder: A multi-agent system for automated machine learning in medical imaging," 2025. [Online]. Available: https://arxiv.org/abs/2502.20301
- [365] Y. Han, A. Ceross, and J. H. M. Bergmann, "Standard applicability judgment and cross-jurisdictional reasoning: A rag-based framework for medical device compliance," 2025. [Online]. Available: https://arxiv.org/abs/2506.18511
- [366] P. Lu, S. Mishra, T. Xia, L. Qiu, K.-W. Chang, S.-C. Zhu, O. Tafjord, P. Clark, and A. Kalyan, "Learn to explain: Multimodal reasoning via thought chains for science question answering," 2022. [Online]. Available: https://arxiv.org/abs/2209.09513
- [367] R. Wang, P. Jansen, M.-A. Côté, and P. Ammanabrolu, "Scienceworld: Is your agent smarter than a 5th grader?" 2022. [Online]. Available: https://arxiv.org/abs/2203.07540

- [368] P. Jansen, M.-A. Côté, T. Khot, E. Bransom, B. D. Mishra, B. P. Majumder, O. Tafjord, and P. Clark, "Discoveryworld: A virtual environment for developing and evaluating automated scientific discovery agents," 2024. [Online]. Available: https://arxiv.org/abs/2406.06769
- [369] R. Lou, H. Xu, S. Wang, J. Du, R. Kamoi, X. Lu, J. Xie, Y. Sun, Y. Zhang, J. J. Ahn, H. Fang, Z. Zou, W. Ma, X. Li, K. Zhang, C. Xia, L. Huang, and W. Yin, "Aaar-1.0: Assessing ai's potential to assist research," 2025. [Online]. Available: https://arxiv.org/abs/2410.22394
- [370] Z. Chen, S. Chen, Y. Ning, Q. Zhang, B. Wang, B. Yu, Y. Li, Z. Liao, C. Wei, Z. Lu, V. Dey, M. Xue, F. N. Baker, B. Burns, D. Adu-Ampratwum, X. Huang, X. Ning, S. Gao, Y. Su, and H. Sun, "Scienceagentbench: Toward rigorous assessment of language agents for data-driven scientific discovery," 2025. [Online]. Available: https://arxiv.org/abs/2410.05080
- [371] Z. S. Siegel, S. Kapoor, N. Nagdir, B. Stroebl, and A. Narayanan, "Core-bench: Fostering the credibility of published research through a computational reproducibility agent benchmark," 2024. [Online]. Available: https://arxiv.org/abs/2409.11363
- [372] N. Edwards, Y. Lee, Y. A. Mao, Y. Qin, S. Schuster, and N. Kim, "Rexbench: Can coding agents autonomously implement ai research extensions?" 2025. [Online]. Available: https://arxiv.org/abs/2506.22598
- [373] X. Shi, Q. Kou, Y. Li, N. Tang, J. Xie, L. Yu, S. Wang, and H. Zhou, "Scisage: A multi-agent framework for high-quality scientific survey generation," 2025. [Online]. Available: https://arxiv.org/abs/2506.12689
- [374] P. Dasigi, K. Lo, I. Beltagy, A. Cohan, N. A. Smith, and M. Gardner, "A dataset of information-seeking questions and answers anchored in research papers," 2021. [Online]. Available: https://arxiv.org/abs/2105.03011
- [375] B. Bogin, K. Yang, S. Gupta, K. Richardson, E. Bransom, P. Clark, A. Sabharwal, and T. Khot, "Super: Evaluating agents on setting up and executing tasks from research repositories," 2024. [Online]. Available: https://arxiv.org/abs/2409.07440
- [376] Y. Li, Z. Dong, and Y. Shao, "Drafterbench: Benchmarking large language models for tasks automation in civil engineering," 2025. [Online]. Available: https://arxiv.org/abs/2507.11527
- [377] N. Mudur, H. Cui, S. Venugopalan, P. Raccuglia, M. P. Brenner, and P. Norgaard, "Feabench: Evaluating language models on multiphysics reasoning ability," 2025. [Online]. Available: https://arxiv.org/abs/2504.06260
- [378] T. D. Pham, A. Tanikanti, and M. Keçeli, "Chemgraph: An agentic framework for computational chemistry workflows," 2025. [Online]. Available: https://arxiv.org/abs/2506.06363
- [379] B. Zhang, X. Li, H. Xu, Z. Jin, Q. Wu, and C. Li, "Topomas: Large language model driven topological materials multiagent system," 2025. [Online]. Available: https://arxiv.org/abs/2507.04053
- [380] T. de Haan, Y.-S. Ting, T. Ghosal, T. D. Nguyen, A. Accomazzi, E. Herron, V. Lama, R. Pan, A. Wells, and N. Ramachandra, "Astromlab 4: Benchmark-topping performance in astronomy q&a with a 70b-parameter domain-specialized reasoning model," 2025. [Online]. Available: https://arxiv.org/abs/2505.17592
- [381] Z. Wang, H. Huang, H. Zhao, C. Xu, S. Zhu, J. Janssen, and V. Viswanathan, "Dreams: Density functional theory based research engine for agentic materials simulation," 2025. [Online]. Available: https://arxiv.org/abs/2507.14267
- [382] M. Elrefaie, J. Qian, R. Wu, Q. Chen, A. Dai, and F. Ahmed, "Ai agents in engineering design: A multi-agent framework for aesthetic and aerodynamic car design," 2025. [Online]. Available: https://arxiv.org/abs/2503.23315
- [383] A. Shabbir, M. A. Munir, A. Dudhane, M. U. Sheikh, M. H. Khan, P. Fraccaro, J. B. Moreno, F. S. Khan, and S. Khan, "Thinkgeo: Evaluating tool-augmented agents for remote sensing tasks," 2025. [Online]. Available: https://arxiv.org/abs/2505.23752
- [384] Y. Chen, P. Piekos, M. Ostaszewski, F. Laakom, and J. Schmidhuber, "Physgym: Benchmarking Ilms in interactive physics discovery with controlled priors," 2025. [Online]. Available: https://arxiv.org/abs/2507. 15550
- [385] D. Zhang, S. Zhoubian, M. Cai, F. Li, L. Yang, W. Wang, T. Dong, Z. Hu, J. Tang, and Y. Yue, "Datascibench: An Ilm agent benchmark for data science," 2025. [Online]. Available: https://arxiv.org/abs/2502.13897
- [386] H. Wang, A. H. Li, Y. Hu, S. Zhang, H. Kobayashi, J. Zhang, H. Zhu, C.-W. Hang, and P. Ng, "Dsmentor: Enhancing data science agents with curriculum learning and online knowledge accumulation," 2025. [Online]. Available: https://arxiv.org/abs/2505.14163
- [387] Y. Ou, Y. Luo, J. Zheng, L. Wei, S. Qiao, J. Zhang, D. Zheng, H. Chen, and N. Zhang, "Automind: Adaptive knowledgeable

- agent for automated data science," 2025. [Online]. Available: https://arxiv.org/abs/2506.10974
- [388] H. Li, M. D. Ma, J. tse Huang, Z. Weng, W. Wang, and J. Zhao, "Biasinspector: Detecting bias in structured data through llm agents," 2025. [Online]. Available: https://arxiv.org/abs/2504.04855
- [389] T. Chen, S. Anumasa, B. Lin, V. Shah, A. Goyal, and D. Liu, "Auto-bench: An automated benchmark for scientific discovery in llms," 2025. [Online]. Available: https://arxiv.org/abs/2502.15224
- [390] A. AhmadiTeshnizi, W. Gao, and M. Udell, "Optimus: Optimization modeling using mip solvers and large language models," 2023. [Online]. Available: https://arxiv.org/abs/2310.06116
- [391] J. M. Laurent, J. D. Janizek, M. Ruzo, M. M. Hinks, M. J. Hammerling, S. Narayanan, M. Ponnapati, A. D. White, and S. G. Rodriques, "Lab-bench: Measuring capabilities of language models for biology research," 2024. [Online]. Available: https://arxiv.org/abs/2407.10362
- [392] D. Nathani, L. Madaan, N. Roberts, N. Bashlykov, A. Menon, V. Moens, A. Budhiraja, D. Magka, V. Vorotilov, G. Chaurasia, D. Hupkes, R. S. Cabral, T. Shavrina, J. Foerster, Y. Bachrach, W. Y. Wang, and R. Raileanu, "Mlgym: A new framework and benchmark for advancing ai research agents," 2025. [Online]. Available: https://arxiv.org/abs/2502.14499
- [393] M. Tian, L. Gao, S. D. Zhang, X. Chen, C. Fan, X. Guo, R. Haas, P. Ji, K. Krongchon, Y. Li, S. Liu, D. Luo, Y. Ma, H. Tong, K. Trinh, C. Tian, Z. Wang, B. Wu, Y. Xiong, S. Yin, M. Zhu, K. Lieret, Y. Lu, G. Liu, Y. Du, T. Tao, O. Press, J. Callan, E. Huerta, and H. Peng, "Scicode: A research coding benchmark curated by scientists," 2024. [Online]. Available: https://arxiv.org/abs/2407.13168
- [394] G. Wölflein, D. Ferber, D. Truhn, O. Arandjelović, and J. N. Kather, "Llm agents making agent tools," 2025. [Online]. Available: https://arxiv.org/abs/2502.11705
- [395] I. Shi, Z. Li, F. Liu, W. Wang, L. He, Y. Yang, and T. Shi, "esapiens: A platform for secure and auditable retrieval-augmented generation," 2025. [Online]. Available: https://arxiv.org/abs/2507.09588
- [396] D.-C. Lian, R.-S. Huang, P.-E. Chen, C. Lim, Y.-K. Lin, G.-Y. Tseng, Z.-C. Yang, Z.-Y. Lin, P.-C. Chen, and S.-K. Hsieh, "Lingbench++: A linguistically-informed benchmark and reasoning framework for multi-step and cross-cultural inference with llms," 2025. [Online]. Available: https://arxiv.org/abs/2507.16809
- [397] Z. Chu, S. Wang, J. Xie, T. Zhu, Y. Yan, J. Ye, A. Zhong, X. Hu, J. Liang, P. S. Yu, and Q. Wen, "Llm agents for education: Advances and applications," 2025. [Online]. Available: https://arxiv.org/abs/2503.11733
- [398] Z. Kang, J. Gong, J. Yan, W. Xia, Y. Wang, Z. Wang, H. Ding, Z. Cheng, W. Cao, Z. Feng, S. He, S. Yan, J. Chen, X. He, C. Jiang, W. Ye, K. Yu, and X. Li, "Hssbench: Benchmarking humanities and social sciences ability for multimodal large language models," 2025. [Online]. Available: https://arxiv.org/abs/2506.03922
- [399] C. Angliss, J. Cui, J. Hu, A. Rahman, and P. Stone, "A benchmark for generalizing across diverse team strategies in competitive pokémon," 2025. [Online]. Available: https://arxiv.org/abs/2506.10326
- [400] European Parliament and Council of the European Union, "Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 april 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation)," Official Journal of the European Union, Tech. Rep., May 2016, uRL: https://eur-lex.europa.eu/eli/reg/2016/679/oj.
- [401] U.S. Congress, "Health insurance portability and accountability act of 1996," Public Law 104-191, Aug. 21 1996, 104th Congress. URL: https://www.govinfo.gov/content/pkg/PLAW-104publ191/pdf/ PLAW-104publ191.pdf.
- [402] J. Lim, B. Vogel-Heuser, and I. Kovalenko, "Large language model-enabled multi-agent manufacturing systems," 2024. [Online]. Available: https://arxiv.org/abs/2406.01893
- [403] Y. Jadhav and A. B. Farimani, "Large language model agent as a mechanical designer," 2025. [Online]. Available: https://arxiv.org/abs/2404.17525
- [404] Y. Han and Z. Guo, "Regulator-manufacturer ai agents modeling: Mathematical feedback-driven multi-agent llm framework," 2024. [Online]. Available: https://arxiv.org/abs/2411.15356
- [405] S. H. Tóth, Z. J. Viharos, Á. Bárdos, and Z. Szalay, "Sim-to-real application of reinforcement learning agents for autonomous, real vehicle drifting," *Vehicles*, vol. 6, no. 2, pp. 781–798, 2024.
- [406] A. Jonnarth, O. Johansson, J. Zhao, and M. Felsberg, "Sim-to-real transfer of deep reinforcement learning agents for online coverage path planning," *IEEE Access*, vol. 13, p. 106883–106905, 2025. [Online]. Available: http://dx.doi.org/10.1109/ACCESS.2025.3581035

- [407] M. W. Lauer-Schmaltz, P. Cash, J. P. Hansen, and A. Maier, "Towards the human digital twin: Definition and design – a survey," 2024. [Online]. Available: https://arxiv.org/abs/2402.07922
- [408] X. Liu and I. David, "Ai simulation by digital twins: systematic survey, reference framework, and mapping to a standardized architecture," Software and Systems Modeling, Aug. 2025. [Online]. Available: http://dx.doi.org/10.1007/s10270-025-01306-0
- [409] J. Harper, "Autogenesisagent: Self-generating multi-agent systems for complex tasks," 2024. [Online]. Available: https://arxiv.org/abs/2404. 17017
- [410] B. Niu, Y. Song, K. Lian, Y. Shen, Y. Yao, K. Zhang, and T. Liu, "Flow: Modularized agentic workflow automation," 2025. [Online]. Available: https://arxiv.org/abs/2501.07834
- [411] S. Han, Q. Zhang, Y. Yao, W. Jin, and Z. Xu, "Llm multi-agent systems: Challenges and open problems," 2025. [Online]. Available: https://arxiv.org/abs/2402.03578
- [412] L. Applis, Y. Zhang, S. Liang, N. Jiang, L. Tan, and A. Roychoudhury, "Unified software engineering agent as ai software engineer," 2025. [Online]. Available: https://arxiv.org/abs/2506.14683
- [413] J. Zhang, J. Xiang, Z. Yu, F. Teng, X. Chen, J. Chen, M. Zhuge, X. Cheng, S. Hong, J. Wang, B. Zheng, B. Liu, Y. Luo, and C. Wu, "Aflow: Automating agentic workflow generation," 2025. [Online]. Available: https://arxiv.org/abs/2410.10762
- [414] Z. Sha, H. Tian, Z. Xu, S. Cui, C. Meng, and W. Wang, "Agent safety alignment via reinforcement learning," 2025. [Online]. Available: https://arxiv.org/abs/2507.08270
- [415] F. He, T. Zhu, D. Ye, B. Liu, W. Zhou, and P. S. Yu, "The emerged security and privacy of llm agent: A survey with case studies," 2024. [Online]. Available: https://arxiv.org/abs/2407.19354
- [416] F. Berdoz and R. Wattenhofer, "Can an ai agent safely run a government? existence of probably approximately aligned policies," 2024. [Online]. Available: https://arxiv.org/abs/2412.00033
- [417] T. Osogami, "Ai agents should be regulated based on the extent of their autonomous operations," 2025. [Online]. Available: https://arxiv.org/abs/2503.04750
- [418] Y. Bai, S. Kadavath, S. Kundu, A. Askell, J. Kernion, A. Jones, A. Chen, A. Goldie, A. Mirhoseini, C. McKinnon, C. Chen, C. Olsson, C. Olah, D. Hernandez, D. Drain, D. Ganguli, D. Li, E. Tran-Johnson, E. Perez, J. Kerr, J. Mueller, J. Ladish, J. Landau, K. Ndousse, K. Lukosuite, L. Lovitt, M. Sellitto, N. Elhage, N. Schiefer, N. Mercado, N. DasSarma, R. Lasenby, R. Larson, S. Ringer, S. Johnston, S. Kravec, S. E. Showk, S. Fort, T. Lanham, T. Telleen-Lawton, T. Conerly, T. Henighan, T. Hume, S. R. Bowman, Z. Hatfield-Dodds, B. Mann, D. Amodei, N. Joseph, S. McCandlish, T. Brown, and J. Kaplan, "Constitutional ai: Harmlessness from ai feedback," 2022. [Online]. Available: https://arxiv.org/abs/2212.08073
- [419] A. S. Schnur. (2025) Agentic ai in workflow automation. Medium blog post. Accessed: 2025-09-18. [Online]. Available: https://medium.com/ @schnur/agentic-ai-in-workflow-automation-e109bb366a1a
- [420] D. Kokotajlo, S. Alexander, T. Larsen, E. Lifland, and R. Dean, "Ai 2027: Scenario forecast for the next decade of ai," https://ai-2027.com/, 2025, published April 3, 2025.