International Journal of
Network Dynamics and Intelligence (IJNDI)

Scilight

Review

Check for updates

Toward Collaborative AI: A Framework for the Transition from Autonomous Agents to Adaptive Human-AI Partners

Nalan Karunanayake [1,*], Savindu Nanayakkara [2], Kasun Gayashan Hettihewa [3], and Upaka Rathnayake [4,*]

[1] Department of Radiology, Memorial Sloan Kettering Cancer Center, New York, NY 10065, USA
[2] Department of Computational Mathematics, University of Moratuwa, Moratuwa 10400, Sri Lanka
[3] Faculty of Engineering, Curtin University Colombo, Colombo 00200, Sri Lanka
[4] Faculty of Engineering and Design, Atlantic Technological University, F91YW50 Sligo, Ireland
* Correspondence: karunan@mskcc.org (N.K.); upaka.rathnayake@atu.ie (U.R.)
Received: 9 May 2026; Revised: 26 June 2026; Accepted: 19 August 2026; Published: 4 September 2026

Abstract: Agentic AI systems that reason, plan, and act on complex goals have advanced rapidly across software engineering, scientific discovery, drug development, healthcare, finance, and social simulation. Across these domains a single failure pattern recurs: current systems can execute tasks competently but often struggle to determine when to act, when to pause, when to change strategy, and when to involve a human. Existing reviews catalog agentic architectures, taxonomies, and limitations, but none specify what capabilities these systems must acquire to support dynamic human-AI collaboration. We address that gap. We define collaborative AI as a class of systems that combine generative exploration with autonomous action and calibrate between them based on context, uncertainty, and task demands. We identify four required capabilities: metacognition, contextual mode-switching, uncertainty-aware action, and adaptive human collaboration. We relate these capabilities to established multi-agent systems foundations, including belief-desire-intention architectures, adjustable autonomy, mixed-initiative interaction, and decentralized decision-theoretic control, while specifying the distinct challenges that LLM-based agents introduce. Across the six domains reviewed here, these gaps appear repeatedly and are not solved by current architectures, which positions collaborative AI as a concrete near-term research objective.

Keywords: large language models; Agentic AI; collaborative AI framework; metacognition; multi-agent systems; uncertainty-aware action

1. Introduction

Early artificial intelligence (AI) progressed from fixed rules to statistical, data-driven inference, yet both approaches remained limited to responding to explicitly defined input [1–5]. The introduction of the transformer architecture [6] enabled large language models (LLMs) capable of generative sequence modeling and generalization across tasks [7–9]. Despite these advances, systems remained limited by their reactive nature and a lack of persistent goal representation, rendering them unable to autonomously pursue objectives over extended decision sequences [10,11].

Agentic AI moves beyond reactive generation toward independent action. Unlike prompt-based models, agentic architectures maintain an evolving task state to execute multi-step plans, leveraging external tools and environmental feedback to iteratively adapt their behavior [12–17]. In practice, this means browsing external information sources, executing code, querying databases, and coordinating with other AI agents to accomplish objectives that extend beyond a single prompt-response exchange. For example, autonomous software engineering systems resolve real-world programming tasks, and AI-driven laboratory platforms design and execute chemical experiments with minimal human intervention [18,19].

Yet evidence from deployed systems reveals a consistent failure pattern that cuts across domains. Current systems tend to fall between two limited operating patterns: reactive systems that remain idle until prompted, and rigidly autonomous agents that pursue predefined goals without adapting when conditions shift, confidence degrades, or human oversight is needed. They execute reasonably well within narrow parameters. However, they often struggle to balance open-ended exploration against committed execution, to adjust their initiative to changing

situational demands, and to recognize when autonomous action should give way to human oversight. We argue that this is one of the main near-term challenges for Agentic AI, and we refer to the missing architectural category as collaborative AI: systems that integrate generative exploration with autonomous action within a unified framework, dynamically calibrating between them based on context, uncertainty, and task demands.

For clarity, we distinguish four related terms. Agentic AI refers to systems that pursue multi-step goals through planning, tool use, and environmental feedback. Autonomous agents are the subset that execute end-to-end with minimal human involvement. Human-AI collaboration refers broadly to any arrangement in which humans and AI systems share a task. Collaborative AI, as used in this review, refers more specifically to systems that dynamically calibrate between exploration, execution, and human involvement based on context, uncertainty, and task demands. A system with these capabilities functions as an adaptive human-AI partner: neither a fixed assistant nor a fully independent agent, but one whose operating mode and degree of human involvement are selected at runtime. Collaborative AI builds on three established concepts and departs from each in a specific way. Human-in-the-loop design fixes human checkpoints at predetermined points in a workflow, whereas collaborative AI makes the timing and form of human involvement a runtime decision. Mixed-initiative interaction [20] arbitrates whether the system or the user should act next, weighing the expected utility of acting against an explicit model of the user's goals. Collaborative AI does not assume such an explicit model. Instead, it must infer when to act, pause, or defer from learned representations. It must also resolve an additional decision that is less central in classic mixed-initiative formulations, which is whether to continue exploring alternatives or commit to execution. Adjustable autonomy [21] shifts the locus of control between agent and human according to externally specified conditions, whereas collaborative AI must infer when and how to shift from learned, subsymbolic representations rather than relying on externally specified rules.

Several recent surveys have examined aspects of the Agentic AI transition, including broad overviews of concepts, methodologies, and ethical considerations [10], taxonomies distinguishing task-specific agents from orchestrated agentic systems [22], systematic assessments across emerging application areas [23], and structured reviews of definitions, frameworks, and evaluation metrics [24]. These works comprehensively map the current landscape and identify critical technical limitations such as deficits in causal reasoning, multi-step planning, and transparency. However, these reviews share a common limitation. They recognize advanced human-AI collaboration as an important future direction but do not fully specify how agentic systems should support it. To our knowledge, none provides a structured framework that defines the required capabilities, relates them to established multi-agent systems foundations, and identifies the architectural gaps that must be closed. This review addresses that open problem.

We organize this review around a central argument: the next productive transition in AI is not simply toward greater autonomy, but toward systems that balance autonomous action with adaptive human collaboration. We first surveyed the architectural foundations of Agentic AI (Section 2) and analyzed evidence across six applied domains (Section 3), identifying recurring capability gaps in metacognition, mode-switching, uncertainty management, and human collaboration. We then examine the main limitations of current systems (Section 4). Building on this evidence base, we develop collaborative AI as a proposed architectural framework for agentic systems (Section 5). We conclude by situating this framework within broader debates about artificial general intelligence (AGI) definitions and timelines (Section 6).

The contributions of this review are threefold. First, we define collaborative AI as an architectural framework between reactive generative systems and rigidly autonomous agents, organized around four capabilities: metacognition, contextual mode-switching, uncertainty-aware action, and adaptive human collaboration. Second, we relate these capabilities to established multi-agent systems foundations, including belief-desire-intention (BDI) architectures [25], adjustable autonomy [21], mixed-initiative interaction [20], and decentralized decision-theoretic frameworks [26], while identifying the distinct challenges introduced by LLM-based agents. Third, we synthesize evidence from six applied domains to show that these capability gaps recur across contexts and remain structurally unaddressed by current architectures, positioning collaborative AI as a concrete near-term research objective rather than an abstract aspiration.

2. Foundations of Agentic AI

Four functional layers can be identified across Agentic AI systems. A reasoning layer forms and revises plans, an execution layer acts on the environment through tools, a perception layer grounds the system in external information, and a coordination layer manages interactions among multiple agents. These layers are typically organized around a reasoning-action loop. Two additional dimensions characterize Agentic AI systems. The first is

the level of autonomy, ranging from assistive suggestions to supervised execution and end-to-end operation. The second is the mode of operation, ranging from reactive response to deliberative search and hybrid combinations.

In practice, this loop allows an LLM to form an intermediate hypothesis, translate it into a plan, execute actions through tools or environmental interactions, observe the outcome, and update its internal state before selecting the next step [12,15,27,28]. A central conceptual influence on this architecture is the ReAct (reasoning and acting) framework, which explicitly combined reasoning and acting within a single iterative process [15].

By anchoring intermediate reasoning steps in external observations, ReAct reduced ungrounded generations relative to pure chain-of-thought (CoT) reasoning and outperformed action-only strategies across several knowledge-intensive and decision-making benchmarks [15,29]. The ReAct framework thus established how deliberation and environmental interaction could be combined within a single system.

Within the reasoning layer, CoT prompting [29] provided an important early step before ReAct by showing that models often perform better on complex multi-step tasks when they generate intermediate reasoning steps. Tree of Thoughts (ToT) extended this idea by allowing models to explore multiple candidate reasoning trajectories and search among them using breadth-first or depth-first strategies [30]. While these techniques do not guarantee reliable agentic planning, they provide a basis for breaking down complex tasks, exploring alternatives, and tracking intermediate states within agentic workflows. More recently, large reasoning models (LRMs) have extended these ideas by combining reinforcement learning (RL) with increased test-time computation, training LLMs to generate longer chains of intermediate reasoning and explore multiple solution paths before committing to an output [28]. These models achieve stronger performance on complex mathematical, scientific, and programming tasks, broadening the reasoning capacity available to agentic systems.

At the execution layer, external tool integration has been equally important for agentic systems, extending their functional capacity beyond static text generation. Frameworks such as Toolformer [17] and TALM [31] demonstrated that models can autonomously learn to invoke application programming interfaces (APIs), an approach scaled by ToolLLM [32] to over 16,000 real-world APIs and refined by Gorilla [33] to reduce hallucinations in API call generation. For logic-intensive tasks, PAL [34] and MRKL Systems [35] showed that offloading computation to external interpreters and neuro-symbolic components yields verifiable reasoning beyond the reach of pure language modeling, with TPTU [36] further framing tool selection as an integral component of agent planning. These developments moved LLMs from isolated text generators toward systems capable of interacting with digital environments through executable interfaces.

Operating in the perception layer, Retrieval-Augmented Generation (RAG) established a foundational mechanism for grounding LLM outputs in external knowledge. First proposed by Lewis et al. [37], RAG augments the generative process by retrieving relevant sources from an external knowledge base at inference time and conditioning the model's output on both the input query and the retrieved evidence. This architecture addresses a core limitation of parametric LLMs: their reliance on static, training-time knowledge that may be incomplete, outdated, or fabricated during generation. In knowledge-intensive tasks such as open-domain question answering, fact verification, and information synthesis, RAG achieved considerable improvements in factual accuracy over closed-book baselines [37]. For Agentic AI, RAG serves a broader role than single-query grounding. When embedded within agentic control loops, retrieval mechanisms enable agents to access expert knowledge bases, verify intermediate reasoning steps against authoritative sources, and update their informational context dynamically as tasks evolve.

In the coordination layer, multi-agent architectures address tasks that benefit from specialization, coordination, and parallelism. MetaGPT [27] mirrors the structure of a software development team, assigning distinct roles such as product manager, architect, and engineer to separate agents and enforcing coordinated intermediate outputs to reduce miscommunication. However, CAMEL [38] takes a different approach, using inception prompting to let two agents autonomously assign each other roles and collaborate on tasks without human steering. In addition, AutoGen [39] provides a more general-purpose infrastructure, allowing developers to configure multi-agent conversations with customizable backends, tool use, and optional human-in-the-loop participation.

These frameworks show that coordination among multiple AI agents can enhance problem decomposition and refinement beyond what a single agent can achieve. Multi-agent systems also introduce complexities, including inter-agent misalignment, redundant action sequences, and cascading error propagation [40–42]. Reasoning, execution, perception, and coordination, integrated by a reasoning-action loop, form the architectural basis of current Agentic AI systems (Figure 1), though each layer carries limitations that Section 4 will address.

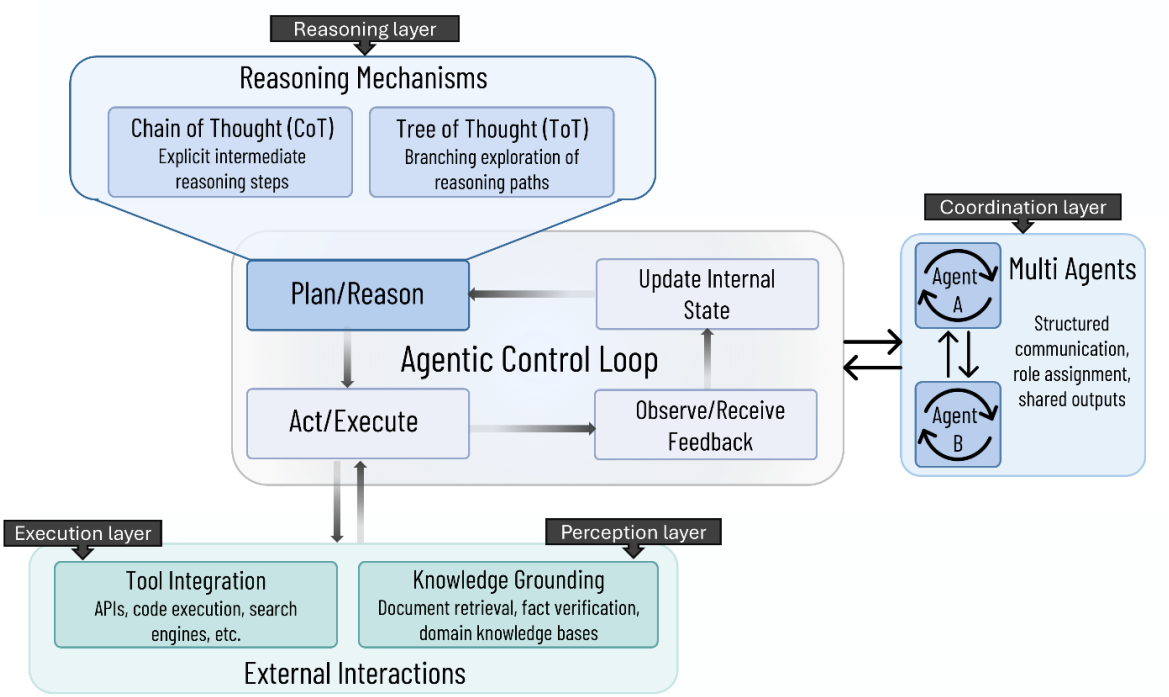

Figure 1. The main parts of current Agentic AI systems. A central control loop plans, acts, observes, and updates state. It draws on four layers: reasoning, execution through external tools, perception through knowledge grounding and retrieval, and coordination across multiple agents.

3. Agentic AI Across Domains

The frameworks described above are no longer only conceptual. They have begun to produce measurable results across several fields. To show how Agentic AI is being used in practice, we examine six domains in which these systems have moved beyond proof of concept. The examples were chosen because they have influenced subsequent work, represent common architectural patterns within their domains, or report evaluable outcomes. Because the evidence includes peer-reviewed studies, preprints, and industry reports, we indicate the evidence type in Table 1 and treat self-reported or preprint findings as provisional.

3.1. Software Engineering

Software engineering provides a clear test case for autonomous execution, because agents can be evaluated against concrete repository-level tasks while still exposing failures in metacognition and uncertainty-aware action. In 2024, Cognition Labs introduced Devin, described as the first AI software engineer [19]. Unlike assistive tools such as GitHub Copilot, which provide inline code suggestions for a user to accept or reject, Devin operates end-to-end. Given a natural language description of a task, it formulates a plan, writes code, runs tests, debugs errors, searches online documentation, and deploys finished applications. It integrates an LLM with a shell, a code editor, and a web browser within a sandboxed compute environment, giving it access to the same tools a human developer would use.

SWE-bench [43] evaluates systems on resolving real-world GitHub issues from popular open-source Python repositories. Cognition reported Devin's performance on SWE-bench as 13.86% issues resolved end-to-end, compared with a previously reported best unassisted baseline of 1.96% [19]. In an assisted setting where models were given the exact files to edit, Cognition reported a best prior result of 4.80%. SWE-bench has since introduced additional benchmark variants, including SWE-bench Verified, SWE-bench Multilingual, and SWE-Bench Pro [44].

SWE-bench was subsequently adopted in a number of studies targeting repository-level issue resolution. Yang et al. introduced SWE-agent [45], arguing that LLMs benefit from interfaces designed specifically for AI agents. By designing an Agent-Computer Interface (ACI) for streamlined file navigation, editing, and execution, SWE-agent achieved 12.47% on SWE-bench, compared with 3.8% reported for a prior non-interactive retrieval-augmented baseline. Zhang et al. presented AutoCodeRover [46], which leverages program-structure signals and can incorporate spectrum-based fault localization when tests are available. AutoCodeRover reports 19% efficacy on SWE-bench Lite at an average cost of $0.43 per issue and a runtime of roughly 4 min, compared with an average of 2.68 days for manual fixes reported in the same work. Tao et al. proposed MAGIS [47], a four-role multi-agent framework comprising a manager, repository custodian, developer, and QA engineer, reporting 13.94% resolved

issues on SWE-bench and an eight-fold improvement over GPT-4 [48] in their experiments. Later versions of Devin gained additional capabilities including multi-agent operation, where one AI agent dispatches tasks to other AI agents, and self-assessed confidence evaluation, in which the system asks for clarification when it is not confident enough to proceed. The open-source community also developed OpenHands, an open platform for building and evaluating software-development agents [49]. Despite these advances, resolution rates on standard benchmarks remain limited, and performance varies considerably across repositories, test coverage levels, and benchmark variants. Some reported results remain self-assessed, with limited independent replication, highlighting the need for standardized evaluation protocols as the field matures. Notably, the later introduction of self-assessed confidence evaluation and clarification-seeking in Devin represents an early, ad hoc response to the absence of metacognition and uncertainty-aware action, two capabilities that the collaborative AI framework proposed in Section 5 identifies as architectural requirements rather than optional add-ons.

3.2. Scientific Discovery

Because experimental actions consume physical resources and cannot always be undone, scientific discovery raises the cost of acting without metacognitive awareness of one's limits. Agentic AI systems have begun to autonomously conduct scientific research across multiple disciplines. Coscientist [18] provides an early example of this architecture applied to experimental chemistry. The system integrates LLM reasoning with internet search, code execution, and robotic laboratory automation to design and carry out chemical experiments. GPT-4 [48] was used within the planning component of the architecture. In validation studies, Coscientist designed and executed palladium-catalyzed cross-coupling reactions, including Suzuki–Miyaura and Sonogashira reactions, using reagents provided in multiwell plate format. The authors report that the system used molecular information encoded in simplified molecular-input line-entry system representations to guide reaction planning and adjust experimental procedures based on structural features of candidate compounds. ChemCrow [50] extended this approach by embedding GPT-4 within a toolkit of 18 expert-designed tools spanning organic synthesis, drug discovery, and materials science. The system relies not only on free-form language generation but also invokes domain-specific computational chemistry utilities during task execution. Across multiple benchmarks, ChemCrow produced improved performance relative to language models operating without tool integration, further showing that structured tool access improves autonomous scientific reasoning.

Agentic architectures have also been applied beyond laboratory science. FunSearch [51] combines a pretrained LLM with evolutionary search and automated evaluation to generate new constructions in extremal combinatorics. Applied to the cap set problem, the system produced constructions that improved asymptotic lower bounds, representing the largest improvement reported in over 20 years. The key design choice is that FunSearch searches over programs that generate solutions, not the solutions themselves, enabling repeated evaluation and refinement within a feedback loop that mirrors experimental exploration. In materials science, autonomous laboratories have emerged as a promising approach for accelerating solid-state inorganic materials synthesis and discovery, integrating computational predictions, machine learning, robotics, and automated characterization [52]. The system integrates computational screening data from the Materials Project [53] with thermodynamics-guided active learning, and robotic execution. Over 17 days of continuous autonomous operation, the A-Lab evaluated 57 computationally predicted inorganic targets and successfully synthesized 36 compounds. Results from each experimental cycle were fed back into the planning system, forming a closed-loop workflow that progressively refined synthesis decisions.

These systems share a structural pattern: the LLM coordinates an autonomous research pipeline, embedded in planning, tool use, execution, and feedback rather than generating text in isolation. Yet they still lack general mechanisms for monitoring their own competence and involving humans when their limits are reached.

3.3. Drug Discovery

Drug discovery extends autonomous scientific workflows into a regulated, multi-stage pipeline, where graded autonomy turns adaptive human collaboration from a convenience into a requirement. While AI systems such as AlphaFold [54] have transformed computational biology through prediction capabilities, drug discovery has begun adopting Agentic AI that autonomously reasons, plans, and iterates through complex research workflows. Gao et al. published a perspective in 2024 proposing a conceptual framework for biomedical AI agents that operate across hypothesis generation, experiment design, and knowledge refinement [55]. The authors describe graded levels of autonomy, culminating in systems that can formulate testable hypotheses, design experimental protocols, and update internal representations through closed-loop feedback. The framework does not position AI as a replacement for scientists but emphasizes collaboration between human expertise and machine-guided exploration of large hypothesis spaces. Early implementations of such architectures have appeared in the Design-Make-Test-

Analyze (DMTA) cycle. Fehlis et al. introduced Tippy [56], a multi-agent framework that coordinates five specialized agents responsible for molecular design, laboratory execution, analysis, reporting, and supervisory oversight. The system automates components of the DMTA workflow and incorporates safety guardrails to regulate agent interactions. Reported evaluations indicate improvements in workflow coordination and automation within controlled laboratory settings.

At the computational modeling stage, Gómez-Tamayo et al. introduced MolAgent in 2025 [57], presenting an agent-based framework for molecular property estimation. MolAgent integrates automated feature generation, model selection, ensemble construction, and validation within a pipeline. By embedding these steps within an agentic control loop, the system formalizes expert modeling practices into an autonomous workflow. Kodela proposed a related multi-agent architecture for molecular design and optimization that coordinates specialized agents across stages of compound generation and refinement [58]. Recent evaluations have also examined the limitations of current agentic drug discovery systems. Wijaya assessed six agentic frameworks across 15 task categories and identified capability gaps, including limited integration of protein language models, insufficient bridging between in silico and in vivo data, and reliance on single-objective optimization strategies [59]. Despite this growing activity, significant challenges remain in achieving broad generalization and end-to-end pharmaceutical integration. The emphasis on graded levels of autonomy in Gao et al.'s framework [55] and the supervisory oversight agent in Tippy [56] gesture toward the adaptive human collaboration that collaborative AI requires, yet these remain static design choices rather than dynamic, context-sensitive capabilities.

3.4. Healthcare and Medical Imaging

Healthcare places especially high demands on oversight, testing adaptive human collaboration in its most consequential form: autonomous reasoning must be balanced against clinical accountability at every step. In healthcare, autonomous reasoning systems must be balanced with clinical oversight, making it a demanding test case for Agentic AI. TissueLab [60] is a co-evolving agentic framework for medical imaging. The system allows clinicians to submit research questions in natural language, after which it generates analytical workflows and invokes specialized tools spanning pathology, radiology, and spatial omics. An LLM serves as the orchestration layer that determines which expert models to use and how intermediate outputs are incorporated into subsequent steps. The architecture separates orchestration from image-level computation, with tailored models performing direct image analysis. VoxelPrompt adopts a different strategy by embedding agency within volumetric image processing [61]. The system employs a language-based controller that iteratively predicts executable instructions for processing 3D medical volumes such as MRI and CT scans. The controller interacts with a vision network to encode image features, generate volumetric outputs including segmentations, and interpret intermediate results to guide subsequent actions. This iterative predict-execute-interpret loop enables the system to decompose complex radiological tasks into sequential analytical operations rooted in image data.

Agentic approaches have also been applied to broader clinical reasoning tasks. Agent hospital introduced a simulated hospital environment in which medical agents interact and evolve through scenarios [62]. Zuo et al. proposed KG4Diagnosis, a hierarchical multi-agent diagnostic framework enhanced by knowledge graphs to coordinate reasoning across clinical information sources [63]. Zhou et al. proposed Zodiac, a multi-agent LLM framework for cardiovascular diagnosis and reported performance comparable to expert-level benchmarks under defined evaluation settings [64].

What emerges across these systems is a common pattern: LLMs function as coordinating components within structured clinical pipelines, not as standalone predictors. Across imaging and diagnostic contexts, agentic architectures center on planning, external tool use, and stepwise reasoning while preserving separation between coordination and domain-specific computation. The central challenge remains the safe integration of autonomous reasoning with rigorous validation, transparency, and human oversight in high-stakes medical environments [14]. Healthcare thus illustrates the collaborative AI challenge in its most demanding form: systems must not only execute clinical reasoning competently, but must also recognize the boundaries of their competence and calibrate human involvement accordingly, precisely the metacognitive and mode-switching capabilities that Section 5 formalizes.

3.5. Finance

Finance shifts the challenge to continuous decisions under market uncertainty, where contextual mode-switching between cautious and committed action is priced in real time. Financial systems have long incorporated automated decision models, particularly in trading, portfolio optimization, and risk assessment. Recent research has begun to explore agentic architectures that extend these systems beyond static prediction toward adaptive, multi-step decision frameworks. A recent review synthesized developments in RL agents, multi-agent financial

systems, and LLM augmented financial reasoning across applications including algorithmic trading, fraud detection, credit risk modeling, and regulatory compliance [65]. These approaches focus on ongoing interaction with market environments, tool invocation, and dynamic policy updates beyond single-pass optimization.

More recent work has applied LLM-based multi-agent architectures directly to trading. Xiao et al. introduced TradingAgents [66], a framework that mirrors the organizational structure of real-world trading firms by deploying seven specialized agent roles including fundamental analysts, sentiment analysts, technical analysts, Bull and Bear researchers, traders, and risk managers. The framework coordinates agents through reports and adversarial debates in which Bull researchers argue for investment opportunities while Bear researchers highlight risks, instead of routing all available information through a single model. Trading decisions emerge from this deliberative process and are subject to review by a risk management team before execution. In backtesting experiments on individual equities, TradingAgents showed improvements in cumulative returns, Sharpe ratio, and maximum drawdown compared to traditional baselines. The adversarial debate mechanism deserves particular attention because it introduces a form of internal quality control that goes beyond simple task decomposition.

Agentic architectures have also been applied to financial modeling and model risk management. Okpala et al. constructed two multi-agent crews using the CrewAI framework [67] to automate end-to-end workflows in financial services [68]. The modeling crew coordinates a judge agent with specialized agents responsible for exploratory data analysis, feature engineering, model selection and hyperparameter tuning, model training, model evaluation, and documentation writing. A separate model risk management crew independently performs compliance checking of modeling documentation, model replication, conceptual soundness assessment, outcome analysis, and documentation. Both crews incorporate a human-in-the-loop module that enables expert oversight at critical decision points. The authors validated the effectiveness of these crews across credit card fraud detection, credit card approval, and portfolio credit risk modeling tasks. This work illustrates how Agentic AI can automate both predictive modeling and the governance and validation processes that surround it, a capability with particular relevance to regulated industries. The adversarial debate mechanism in TradingAgents and the human-in-the-loop checkpoints in Okpala et al.'s framework represent partial solutions to the mode-switching and human collaboration challenges, but these remain fixed architectural choices rather than capabilities that adapt to shifting task demands and uncertainty levels.

3.6. Social Simulation and Emergent Behavior

Social simulation removes the pressure of task completion and instead reveals what agents do when left to interact, exposing coordination that arises with no metacognitive support beneath it. Recent work has explored how Agentic AI gives rise to social behavior when deployed in multi-agent environments. Park et al. introduced a simulated town populated by 25 LLM-driven agents equipped with memory, planning, and reflection mechanisms [69]. Each agent maintained a stream of observations, retrieved contextually relevant memories, synthesized higher-level reflections, and generated planned actions based on its internal state. Within this environment, agents initiated conversations, formed relationships, disseminated information, and coordinated social activities. In one documented scenario, an agent organized a social event, and other agents made attendance decisions based on prior interactions and scheduling constraints. These behaviors emerged without explicit hard-coded social rules, instead arising from memory retrieval, reflection, and planning processes mediated by language model calls.

Subsequent studies have used LLM-based agents as experimental substrates to investigate emergent social phenomena. Ashery et al. showed that populations of interacting LLM agents can spontaneously develop shared conventions and collective behavioral patterns through repeated interaction, without pre-specified rules [70]. Takata et al. reported that sustained interaction within LLM agent communities gives rise to emergent individuality and role differentiation shaped by accumulated interaction histories [71]. These studies do not implement the full agentic control loop described in Section 2, as the agents lack explicit tool integration and structured planning-execution cycles. Nevertheless, they demonstrate that LLM-based agents embedded in social environments can exhibit coordination and behavioral differentiation beyond what is programmed, extending the implications of agentic architectures into social simulation. Table 1 consolidates the systems reviewed across all six domains, summarizing their approaches, evaluation settings, key results, and limitations. These emergent coordination behaviors are notable because they arise without the metacognitive or uncertainty-aware mechanisms that collaborative AI would require, raising the question of whether more deliberate architectural support for self-monitoring and adaptive collaboration could yield richer and more controllable forms of multi-agent social interaction.

Table 1. Agentic AI systems across six domains, with architecture, evidence type, principal result, and main limitation.

System (Year)	Approach	Evidence	Key Result	Limitations
Software Engineering				
Devin (2024) [19]	Single autonomous agent, plan-code-test-debug loop	Industry report, self-reported	13.86% SWE-bench end-to-end vs 1.96% baseline	No independent replication, sensitive to test coverage
SWE-agent (2024) [45]	LLM agent with agent–computer interface	Peer-reviewed (NeurIPS)	12.47% SWE-bench, 18% on Lite	Sensitive to repository structure and test quality
AutoCodeRover (2024) [46]	Program-structure-aware agent with fault localization	Peer-reviewed (ISSTA)	19.0% SWE-bench Lite at $0.43/issue	Correct localization still yields incorrect patches
MAGIS (2024) [47]	Four-role multi-agent (manager, custodian, developer, QA)	Peer-reviewed (NeurIPS)	13.94% SWE-bench, 25.33% on Lite	Declines as code-change complexity rises
Scientific Discovery				
Coscientist (2023) [18]	LLM planner with web search, code, lab-automation APIs	Peer-reviewed (Nature)	Designed and ran Suzuki–Miyaura and Sonogashira reactions	Proof of concept, partial manual intervention
ChemCrow (2024) [50]	LLM with 18 expert chemistry tools	Peer-reviewed (Nat. Mach. Intell.)	Outperformed tool-free LLMs across chemistry tasks	Depends on tool quality, limited reproducibility under closed APIs
FunSearch (2024) [51]	LLM with evolutionary program search and evaluation	Peer-reviewed (Nature)	Improved cap set bounds, beat first/best-fit bin packing	Needs efficient evaluators; results vary across runs
A-Lab (2023) [52]	Closed-loop lab: screening, active learning, robotics	Peer-reviewed (Nature)	Synthesized 36 of 57 targets in 17 days (63%)	Limited by reaction kinetics and DFT stability errors
Drug Discovery				
Tippy (2025) [56]	Five-agent framework across the DMTA cycle	Preprint	Automated DMTA cycle, 95.3% purity, 72% yield in a demo	Controlled demos; little comparison to baseline workflows
MolAgent (2025) [57]	Agentic pipeline for molecular property estimation	Peer-reviewed (J. Chem. Inf. Model.)	Competitive on ADMET; first on lipophilicity MAE	Retrospective benchmarks, high compute cost
Healthcare and Medical Imaging				
TissueLab (2025) [60]	LLM orchestration over domain expert models	Preprint	Beat VLM baselines, 0.931 accuracy for metastasis classification	Needs clinician supervision, depends on segmentation-tool quality
VoxelPrompt (2024) [61]	Joint LLM and 3D CNN generating executable code	Preprint	Top zero-shot lesion Dice, 89% characterization accuracy	Brain imaging only, no multi-organ clinical deployment
Zodiac (2024) [64]	Multi-agent LLM for cardiovascular diagnosis	Preprint	Top scores on 8 expert-rated metrics over GPT-4o, Gemini Pro	Single specialty, expert rating, not prospective deployment
Finance				
TradingAgents (2025) [66]	Seven-role multi-agent firm with Bull/Bear debate	Preprint	Improved returns, Sharpe ratio, max drawdown in backtests	Short backtest window, multi-agent latency and cost
Financial Modeling and MRM Crews (2025) [68]	Two CrewAI crews with judge agent and human-in-the-loop	Preprint	End-to-end modeling and risk management across three tasks	Standard datasets, not tested in production
Social Simulation and Emergent Behavior				
Generative Agents (2023) [69]	25 agents with memory, reflection, planning	Peer-reviewed (UIST)	Emergent conversations, relationships, event coordination	Single simulated environment, no real-world transfer
LLM Convention Studies (2025) [70]	Agent populations in repeated decentralized interaction	Peer-reviewed (Science Advances)	Shared conventions emerged without pre-specification	No planning/tool loop, generalizability untested

3.7. Cross-Domain Synthesis

The same gaps recur across the six domains despite very different tasks. The architectures converge on a common progression, from the single reasoning-action loop of Coscientist [18], through the role-based multi-agent

designs of MAGIS [47] and TradingAgents [66], to the closed-loop feedback of A-Lab [52] and TissueLab [60]. Yet none of these systems monitors its own competence or adjusts its autonomy as a task changes. Devin's later addition of confidence-based clarification and the supervisory agent in Tippy are the closest attempts, and both are fixed design choices rather than the metacognition and adaptive human collaboration the systems would need to make those judgments at runtime. The level of human involvement varies by domain: software engineering tends toward near-autonomous execution in sandboxed environments, finance relies on threshold-based monitoring, and healthcare requires human approval for high-stakes decisions. These differences broadly follow the reversibility and cost of error. However, in each case, the oversight model is largely fixed at design time rather than adjusted dynamically as uncertainty rises or task stakes change. Table 1 catalogues the individual systems and Table 2 organizes them by domain, and together they show that this is the recurring deficit that motivates the framework in Section 5.

In Table 2 each domain is summarized by its typical task, dominant architecture, primary bottleneck, oversight model, and evidence maturity, showing that the binding constraint and the tolerated level of autonomy track the reversibility and cost of error rather than the task itself.

Table 2. Cross-domain synthesis of Agentic AI.

Domain	Task	Dominant Architecture	Primary Bottleneck	Oversight Model	Evidence Maturity
Software engineering	Repository-level issue resolution	Single-agent loop to multi-agent roles	Resolution rate, wrong patches despite correct localization	Near-autonomous, sandboxed	Public benchmarks, conference and industry, self-reported
Scientific discovery	Experiment design and execution	LLM planner with tools and closed-loop robotics	Safe boundary of autonomous planning under physical cost	Supervised execution	Peer-reviewed, proof-of-concept scale
Drug discovery	DMTA cycle and property modeling	Multi-agent pipelines with graded autonomy	in silico to in vivo gap, end-to-end integration	Supervised, with oversight agent	Mixed journal and preprint, retrospective or demo
Healthcare and imaging	Image analysis and clinical diagnosis	LLM orchestration over expert models	Integrating autonomy with validation and accountability	Human approval of actions	Mostly preprint, expert rating, no prospective deployment
Finance	Trading, modeling, risk management	Role-based multi-agent with debate and oversight	Cost of error under market uncertainty, latency	Human monitoring, intervention at thresholds	Preprint, backtests, not production
Social simulation	Emergent multi-agent social behavior	Memory-reflection-planning agents, no tool loop	Control and interpretability of emergent behavior	Observational	Peer-reviewed, single environment, untested transfer

4. Current Limitations and Challenges

Despite these advances, current Agentic AI systems face considerable limitations. Hallucination remains a persistent problem [72], and in agentic systems it is compounded by error propagation across multi-step chains [14,73]. When an agent makes a factual error in an early reasoning step, that error can cascade through subsequent actions, leading to confidently executed but fundamentally flawed outcomes [74]. Several frameworks address this challenge. The ReAct framework mitigates hallucination to some extent through access to external information, but it does not eliminate the problem [15,75]. Reflexion [16] takes a different approach by enabling agents to verbally reflect on task feedback and maintain an episodic memory of self-reflective experiences that improve decision making in subsequent trials. While ReAct alone showed hallucination rates converging at roughly 22% with no signs of recovery, combining it with Reflexion reduced that rate and enabled continued learning across trials.

The CRITIC framework [76] allows agents to validate and progressively revise their own outputs through tool-interactive critiquing, using search engines for fact-checking and code interpreters for debugging in iterative verify-then-correct loops. Chain-of-Verification (CoVe) [77] introduced a four stage pipeline in which the model drafts a response, plans verification questions, answers them independently to avoid bias, and generates a final verified output, achieving reductions in hallucination across diverse tasks.

Beyond these specific frameworks, researchers deploy a range of complementary strategies. As discussed in Section 2, RAG [37] provides a further layer of grounding by conditioning agent outputs on evidence retrieved from external knowledge bases, reducing reliance on parametric memory and anchoring multi-step reasoning in verifiable sources. Strict prompting with explicit output schemas and deterministic external tool usage constrains the agent's action space to reduce the likelihood of fabricated intermediate steps. Multi-agent verification

architectures assign separate agents to generate and critique outputs, creating adversarial checks that catch errors a single agent might miss. Fine-tuning on domain-specific data combined with low-temperature sampling reduces output variance and steers generation toward higher-confidence predictions. Human-in-the-loop checkpoints at critical decision points provide a final layer of oversight, allowing experts to intercept and correct errors before they propagate through downstream actions. Even so, early agentic systems such as AutoGPT were frequently reported to get stuck in unproductive loops and to fail to complete non-trivial tasks reliably [78], a reminder that no single mitigation strategy has yet solved these reliability and hallucination-related failure modes.

Reliability and consistency across repeated trials remain a significant challenge. When presented with the same task under nominally identical conditions, agentic systems may generate divergent action trajectories and outcomes, complicating validation and deployment in mission-critical settings [79,80]. This variability also limits reproducibility, because an agent's output can depend on stochastic sampling, external tool behavior, retrieval results, and model versions that change over time. The problem is compounded when results are self-reported, evaluated on developer-designed benchmarks, or not independently replicated, with limited ability to assess training-data contamination when training corpora are undisclosed. Stronger comparisons would require fixed model versions, reporting of performance distributions across repeated runs, clear separation of developer-reported and independently replicated results, and greater transparency about evaluation data.

Safety raises additional concerns, particularly around dual-use risks. The Coscientist authors explicitly discussed these risks and evaluated whether the system could be coerced into generating synthesis guidance for hazardous or controlled chemicals, illustrating the security challenges posed by powerful autonomous chemistry agents [18]. Computational cost and latency also limit practical deployment, because multi-step agentic workflows require repeated model calls, tool invocations, retrieval steps, and verification procedures [24].

A central gap concerns the evaluation ecosystem itself. Many benchmarks emphasize task success rates while leaving important dimensions such as process quality, uncertainty recognition, and human collaboration less systematically assessed [44,81]. Industry data further suggest that confidence in Agentic AI remains limited, with only a minority of IT application leaders actively considering or deploying such systems and many identifying them as potential security risks [82]. These trust and governance gaps are not isolated technical problems; they are symptoms of a deeper architectural deficit (Figure 2). Across the six domains reviewed in Section 3, agentic systems demonstrate competent task execution within narrow operational envelopes but consistently lack mechanisms to monitor their own confidence, shift between exploration and execution as conditions change, or calibrate human involvement based on evolving task demands. The limitations documented in this section converge on a common conclusion: the next productive transition is not toward greater autonomy alone, but toward systems that can manage the boundary between autonomous action and adaptive human collaboration. Section 5 develops this argument into a concrete architectural framework.

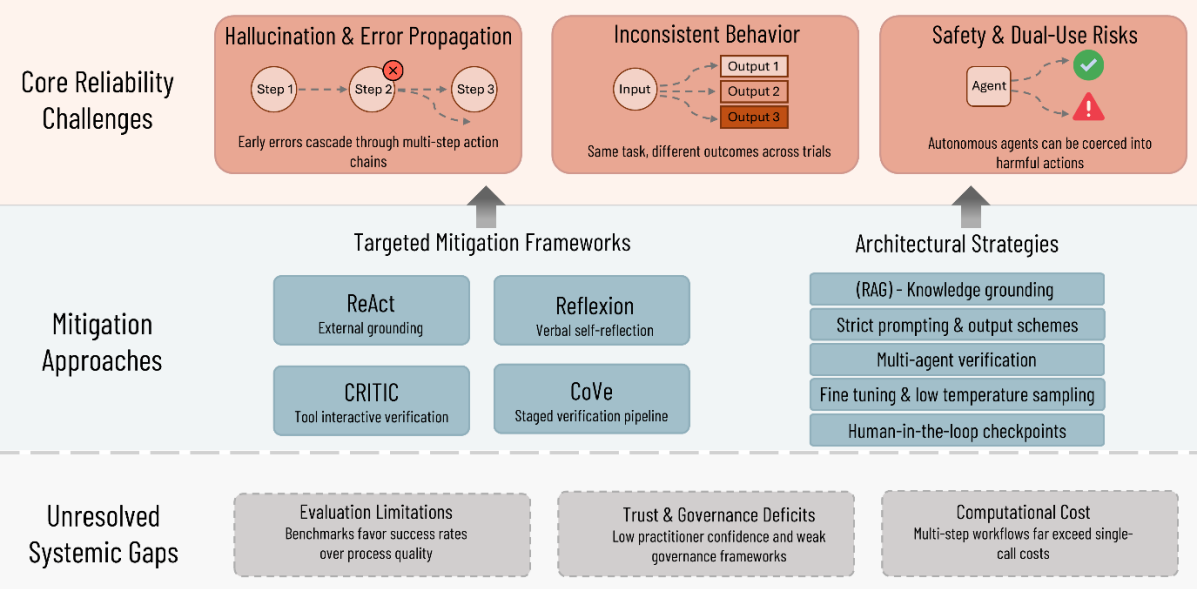

Figure 2. Limitations and mitigation strategies in current Agentic AI systems. Hallucination, inconsistent behavior, and safety risks are each partly addressed by targeted frameworks and architectural strategies, yet evaluation, trust, and cost gaps remain. The mitigations treat each problem separately, and none lets a system judge its own competence.

5. The Transition to Collaborative AI

5.1. Beyond Generative vs. Agentic AI

As discussed in the Introduction, current AI systems built on LLMs operate in one of two configurations. In their default mode, LLMs function as next-token predictors [83] that generate responses reactively to user prompts. In agentic configurations, external frameworks wrap around the same predictive mechanism to orchestrate multi-step goal pursuit, but these systems remain limited in their ability to adjust autonomy, strategy, and human involvement as conditions shift. Neither configuration captures what effective human-AI collaboration requires, which should involve negotiation between exploration and commitment, knowing when to brainstorm, when to narrow options, and when to execute automatically. Current systems do not manage this negotiation [84,85], pointing to the need for architectures that integrate generative flexibility with calibrated decision control.

5.2. Defining Collaborative AI

The concept of human-AI collaboration has a long history in the multi-agent systems and human-computer interaction communities, spanning work on mixed-initiative interaction [20], adjustable autonomy [21], and human-agent coordination [86]. More recently, Holter and El-Assady [84] deconstructed human-AI collaboration around agency, interaction, and adaptation, while Feng et al. [85] explored LLM-based human-agent collaboration with conditional autonomy. Building on these foundations, we use the term collaborative AI in this review to describe a specific architectural concept: an emerging class of AI systems that bring together generative exploration and autonomous action within a unified framework, dynamically calibrating between them based on the system's assessment of context, uncertainty, and task demands. This usage is distinct from broader applications of the term in areas such as AI literacy and educational settings, and focuses specifically on the architectural capabilities required for LLM-based agents to function as effective partners in open-ended tasks.

These systems move beyond fixed-mode operation, adaptively shifting between reasoning and execution (similar to human thought and action). They may explore alternatives and invite human input in one phase, then commit to a course of action and carry it forward in the next, with each transition driven by the system's assessment of the task state. Realizing such systems requires capabilities that are not yet reliably supported in current architectures. The evidence reviewed in Sections 3 and 4 reveals a consistent pattern: current agentic systems fail not primarily in task execution, but in knowing when to act, when to pause, when to shift strategy, and when to involve a human partner. These recurring failure modes point to four capability gaps: metacognition, contextual mode-switching, uncertainty-aware action, and adaptive human collaboration.

Metacognition: awareness of the system's own uncertainty and the boundaries of its competence. A collaborative AI system must recognize when it is operating in a well-understood scenario where autonomous action is appropriate, and when it has entered a domain where confidence is low and exploration or human input would be more productive. A related requirement is contextual mode-switching: transitioning fluidly between generative reasoning and goal-directed execution based on task complexity, the stakes involved, and the novelty of the situation. Routine tasks with clear parameters and low risk may warrant rapid autonomous execution. Novel, high-stakes, or ambiguous tasks warrant slower, more exploratory engagement with human oversight. Collaborative AI also demands uncertainty-aware action, where confidence estimates actively drive decisions about whether to explore further or commit to a path. This goes beyond producing calibrated outputs. It requires maintaining and updating a running estimate of how well the system understands the task, how likely its planned actions are to succeed, and how costly failure would be. Finally, these systems require adaptive human collaboration. The AI system must learn how and when to involve human partners in the decision-making process. This is not a static setting but a dynamic relationship that evolves based on the human's expertise, preferences, and availability, as well as the nature of the task itself.

These four capabilities are not without precedent in the multi-agent systems literature. Metacognition draws on a long tradition of epistemic reasoning in agent architectures, most notably the BDI model [25], which formalized how agents maintain and revise beliefs about their own knowledge states. However, BDI architectures operated over symbolic, hand-crafted knowledge representations, whereas LLM-based agents must derive metacognitive assessments from learned, subsymbolic representations without explicit belief structures. Contextual mode-switching relates to the classical distinction between deliberative and reactive agent architectures, and particularly to hybrid approaches that sought to combine both within a single system [87]. The adjustable autonomy literature [21] addressed a similar challenge: dynamically shifting the locus of control between agent and human based on situational demands, and adaptive decision-making frameworks [88] demonstrated that agents can dynamically restructure their organizational roles at runtime to improve performance. LLM-based systems reintroduce this problem in a less structured form, where mode transitions must be inferred from natural language context rather than formally specified

conditions. Uncertainty-aware action connects to decision-theoretic agent frameworks, including decentralized partially observable Markov decision processes (DEC-POMDPs) [26], which provide formal methods for optimal action selection under uncertainty in multi-agent settings. Current agentic systems lack equivalent formal grounding, relying instead on indirect proxies for uncertainty such as token-level predictions, verbalized self-assessments, or output consistency across repeated sampling, none of which constitute formal uncertainty quantification at the task level. Finally, adaptive human collaboration extends work on mixed-initiative interaction [20] and the foundational roadmap for agent research [86], which established that effective collaboration requires dynamic negotiation of roles and responsibilities rather than fixed allocation. The collaborative AI framework proposed here builds on these foundations while addressing the distinct challenges posed by LLM-based agents: the absence of explicit symbolic representations, the stochastic nature of language-based reasoning, and the need to operate across open-ended task domains rather than predefined environments.

Table 3 summarizes these four capabilities, their current implementation status, and the structural gaps that remain. Figure 3 illustrates how they operate together within a unified architecture.

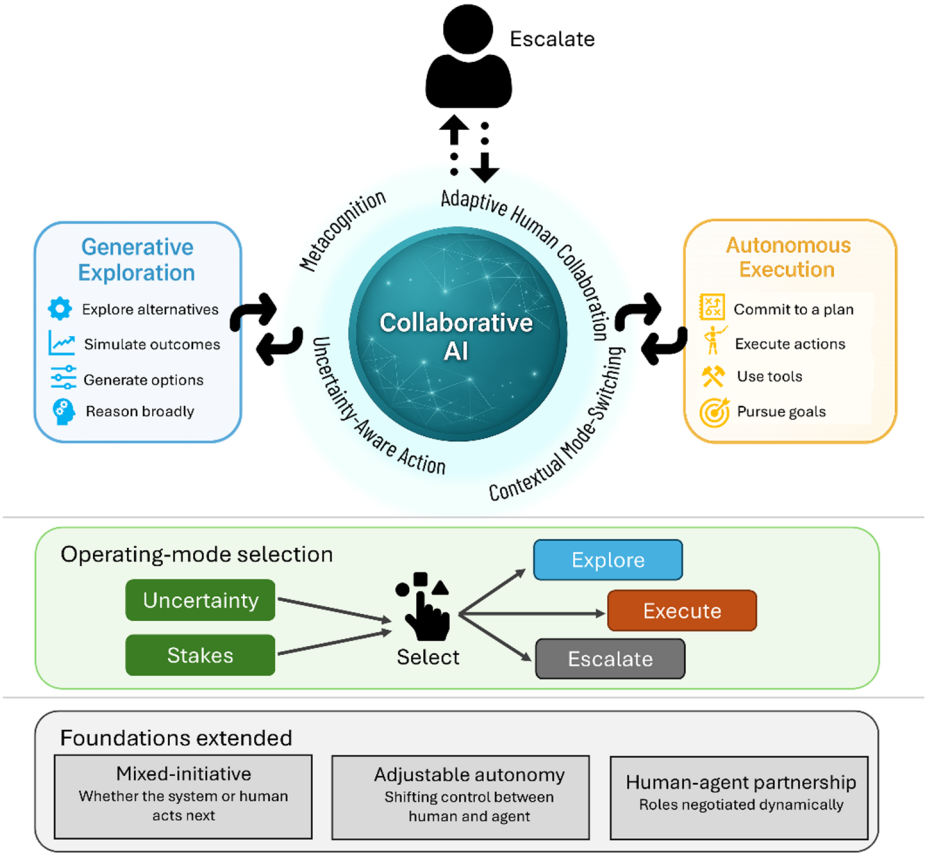

Figure 3. The collaborative AI architecture and its decision structure. The four required capabilities surround the core that integrates generative exploration with autonomous execution. At each step the system assesses its own uncertainty and the stakes of acting, and on that basis selects one operating mode: explore, execute, or escalate to a human partner. The framework extends the mixed-initiative, adjustable-autonomy, and human-agent-partnership foundations shown below by driving this choice from inferred uncertainty rather than fixed rules.

Described individually, the four capabilities above read as separate competencies. Their force comes from how they combine into one decision the system makes at each step: which mode to operate in. We distinguish three modes. In exploration the system reasons broadly, generates alternatives, and may invite human input. In execution it commits to a plan and carries it out autonomously. In escalation it hands the decision, in whole or in part, to a human partner. The choice among the three rests on two assessments the system maintains as the task unfolds: its uncertainty about whether its intended action will succeed, and the stakes, meaning the cost and reversibility of acting wrongly. Low uncertainty and low stakes favor execution; high uncertainty with moderate stakes favors exploration; high stakes relative to confidence favor escalation. Seen this way, the four capabilities are the parts of a single decision rather than independent features: metacognition produces the uncertainty assessment, uncertainty-aware action weighs it against the stakes, contextual mode-switching makes the selection, and adaptive human collaboration

governs the escalation branch. The framework specifies this structure rather than fixed thresholds; where the boundaries between modes fall depends on the cost structure of each task and domain.

Table 3. Collaborative AI: required capabilities, current status, gaps, and limitations.

Required Capability	Description	Typical Implementation Today	Structural Limitation
Metacognition	Awareness of the system's own uncertainty and competence boundaries	Confidence derived from token-level probabilities or post-hoc calibration, not task-level understanding	Limited mechanisms for estimating uncertainty at the level of plans, multi-step reasoning, or action outcomes
Contextual Mode-Switching	Shifting between exploration and execution based on task complexity, uncertainty, and stakes	Systems operate in fixed interaction modes with limited autonomous adaptation	Absence of architecture-level mechanisms that dynamically adjust operational policy in response to changing task context
Uncertainty-Aware Action	Confidence estimates drive whether to act, explore, defer, or seek input	Verification or retrieval is often applied reactively at the output stage or embedded within predefined loops, rather than integrated into pre-action commitment decisions	Limited integration of explicit cost-sensitive decision frameworks that weigh the risk of acting incorrectly against the cost of delay or escalation
Adaptive Human Collaboration	Dynamic allocation of initiative between human and AI based on evolving task demands	Human involvement at predefined checkpoints with limited modeling of user needs	Lack of adaptive initiative management that continuously updates collaboration strategy as tasks and human states evolve

5.3. Evidence from Existing Systems

The concept of collaborative AI is still in its formative phase, yet several recent systems exhibit partial implementations of its core capabilities. Fang et al. describe an emerging class of self-evolving agents that automatically enhance agent behavior based on interaction data and environmental feedback, bridging the static capabilities of foundation models and the continual adaptability needed for real-world deployment [81]. These systems are instructive because they learn not only to complete tasks, but to improve how tasks are approached over time.

TissueLab [60], discussed in Section 3.4, provides a particularly relevant example. It is presented as a co-evolving agentic system in which experts can visualize intermediate analysis results and refine them through real-time feedback, enabling the system to improve its workflows and decision strategies through continued interaction. This reflects an early form of the adaptive human collaboration that collaborative AI will require.

Recent agentic systems incorporate confidence-aware clarification behaviors [89,90], pausing to request additional information when internal uncertainty exceeds a threshold. Although still rudimentary, such mechanisms represent early steps toward the metacognitive mode-switching that collaborative AI requires.

5.4. Adaptive Human Collaboration Versus Static Oversight

The fourth capability is the one most often confused with existing practice, so it is worth separating clearly. Most deployed systems involve humans through static human-in-the-loop design, where the points of consultation are fixed when the system is built. The agent pauses at the same predetermined steps on every run, regardless of how confident it is or how much is at stake. This is predictable and easy to audit, but it spends human attention evenly across situations that do not warrant it and withholds it from situations that do.

Adaptive human collaboration replaces the fixed checkpoint with a decision the agent makes as each situation arises. It rests on three questions. The agent settles when to involve a human by weighing its own uncertainty and the cost of acting wrongly against the value of proceeding alone, so a routine, low-stakes step proceeds untouched while a step the agent is unsure of, or one whose consequences are hard to reverse, triggers involvement. It settles how to involve them by matching the form of contact to the need: a request for clarification when information is missing, a joint review of intermediate results when judgment is needed, or a full handoff of control when the action exceeds the agent's competence. It settles why by reference to the reversibility of the pending action and the boundary of its own reliable operation, the two factors that determine whether human judgment is worth its cost at that moment. The same task may then draw no human involvement on one run and an early handoff on another, depending on the uncertainty and stakes the agent meets along the way.

5.5. Open Challenges and Technical Gaps

Bringing collaborative AI into real-world practice will require progress across several interrelated dimensions. Technically, current LLM-based systems lack robust mechanisms for representing and reasoning about uncertainty at the level of actions and decisions [90]. Token-level probability estimates provide only a narrow view of confidence, and they do not fully capture whether a system understands the broader context of a task. More expressive forms of uncertainty modeling will be needed if systems are to decide responsibly when to act and when to pause. Learning mechanisms also need to advance beyond bounded task adaptation; collaborative AI requires learning across experiences, understanding not only what works, but why and when it may fail.

Deciding when the AI should lead and when the human should take over is a related problem. This challenge maps onto three conventional oversight models: human-in-the-loop (HITL), where a human approves each consequential action; human-on-the-loop (HOTL), where the system acts autonomously but a human monitors and can intervene at any stage; and human-out-of-the-loop (HOOTL), where the system executes end-to-end with only initial specification. The appropriate modality depends on the reversibility of actions, severity of potential harm, and maturity of verification mechanisms. As example scenarios, healthcare represents a clear HITL case, given direct patient safety implications and regulatory mandates for clinician accountability. Finance suits HOTL oversight, where AI systems execute routine transactions but human experts intervene at threshold events. Software engineering offers conditions more amenable to HOOTL: tasks can be precisely specified, outputs are testable in sandboxed environments, errors are reversible through version control, and multi-agent verification provides built-in quality assurance before deployment. Current systems rely on fixed escalation rules, but collaborative AI will need to adjust this balance dynamically as tasks unfold and as the human partner's needs and expertise change. Designing such collaboration protocols requires both technical advances and deeper integration with insights from human–computer interaction and cognitive science. Evaluation presents a parallel challenge. Most benchmarks measure task completion, but collaboration quality also depends on how decisions are made, how uncertainty is managed, and whether human involvement occurs at the right time. Without metrics that capture interaction quality and escalation decisions, progress toward collaborative AI will be difficult to measure.

6. From Agentic AI to General Intelligence: Definitions, Forecasts, and Open Questions

Artificial General Intelligence (AGI) refers to machines capable of matching human cognitive performance across a broad range of domains [91]. Despite its prominence in technical discourse, the concept remains imprecisely defined, and competing frameworks attempt to operationalize progress in different ways. DeepMind researchers have proposed a taxonomy distinguishing levels of generality based on both the breadth and depth of an AI system's competence [92], while OpenAI has articulated capability thresholds within its preparedness framework to assess increasingly advanced systems across defined risk domains [93].

Large-scale surveys of AI researchers reveal substantial dispersion in predicted timelines, with median estimates varying markedly depending on whether AGI is defined as outperforming humans across tasks, fully automating occupations, or meeting other operational criteria [94]. Compared with earlier surveys [95,96], more recent evidence suggests timelines have shifted earlier [94], yet forecasts still span decades and remain sensitive to framing and methodology. No clear consensus has emerged.

This definitional and forecasting uncertainty has practical implications for the field. As systems become more capable and autonomous, challenges related to alignment, oversight, and safe deployment grow in complexity [97]. These concerns do not require assumptions about superintelligent systems; they apply to the trajectory of increasingly capable Agentic AI that this review has documented. Advancing alignment research and governance frameworks in parallel with capability development remains a pressing and evidence-supported priority.

7. Conclusions

Agentic AI has progressed rapidly from conceptual architectures to deployed systems capable of writing software, designing experiments, supporting medical diagnosis, managing financial strategies, and modeling complex social dynamics. The evidence presented here shows that these capabilities are empirically demonstrated and benchmarked across multiple domains. Meanwhile, critical issues like hallucination and error propagation remain unsolved across workflows. Because current evaluation frameworks still favor task completion over process quality, practitioner trust remains severely limited. No single mitigation strategy has fully addressed these reliability gaps.

This review argues that the next transition is more subtle but potentially more consequential. The move from Agentic AI to collaborative AI represents a central near-term challenge because it requires systems to integrate generative reasoning with goal-directed action through context-sensitive judgment, represents a central near-term

challenge. Achieving this shift will require advances in metacognition, uncertainty modeling, adaptive mode-switching, and dynamic human-AI collaboration. These capabilities extend beyond current implementations but build directly on established multi-agent systems foundations, including BDI architectures, adjustable autonomy, and mixed-initiative interaction, while addressing the distinct challenges posed by LLM-based agents, including the absence of explicit symbolic representations and the stochastic nature of language-based reasoning. As Section 6 discusses, definitions of AGI remain contested and expert forecasts span decades, yet alignment and governance challenges are already pressing concerns for the increasingly capable agentic systems documented here.

Beyond identifying this transition, the review puts it in a form that can guide design. It reduces the four capabilities to one runtime decision: at each step a system chooses among exploration, committed execution, and escalation to a human, based on its own assessment of uncertainty and stakes. It maps that choice onto the established oversight models, human-in-the-loop, human-on-the-loop, and human-out-of-the-loop, by the reversibility and cost of error in each domain. Evaluation must follow: benchmarks scoring only task completion cannot reveal whether a system knows when it is uncertain or when to involve a person.

Several concrete directions follow, each advancing one capability. Uncertainty representation must move from token-level signals to estimates over multi-step plans. Learning must extend from within-task adaptation to learning across tasks, including why an approach failed before. Initiative management must become adaptive, updating when a system involves a human as the task and the partner's needs change. And benchmarks must measure collaboration quality and escalation timing directly, so progress can be observed rather than inferred.

This review has focused on LLM-based agentic systems across six application domains. Other areas, including robotics, education, and legal reasoning, involve distinct problems that merit separate analysis. As a narrative review, the scope and selection of studies reflect the authors' judgment rather than a systematic search protocol. Given the rapid pace of development in this field, some findings reported here may have been superseded by the time of publication.

Across all stages, however, the evolution of AI cannot be separated from the structure of the human-AI relationship. The collaborative AI framework proposed here offers one path forward, grounding questions of autonomy, trust, and shared responsibility in concrete architectural requirements instead of leaving them to abstract debate. The long-term impact of AI will depend not only on what these systems can do, but on the institutional, technical, and ethical frameworks we build around them.

Author Contributions: N.K. performed the literature search, reviewed the studies, and wrote the first draft of the manuscript. U.R., S.N., and K.G.H. contributed to the conceptualization of the review structure and provided critical revisions and final editing.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

Use of AI and AI-Assisted Technologies: No AI tools were utilized for this paper.

Abbreviations

ACI	Agent-Computer Interface
ADMET	Absorption, Distribution, Metabolism, Excretion, and Toxicity
AGI	Artificial General Intelligence
AI	Artificial Intelligence
API	Application Programming Interface
BDI	Belief-Desire-Intention
CNN	Convolutional Neural Network
CoT	Chain-of-Thought
CoVe	Chain-of-Verification
CT	Computed Tomography
DEC-POMDP	Decentralized Partially Observable Markov Decision Processes
DMTA	Design-Make-Test-Analyze
ECG	Electrocardiogram
HPLC	High-Performance Liquid Chromatography
HITL	Human-in-the-Loop
HOTL	Human-on-the-Loop
HOOTL	Human-out-of-the-Loop
LLM	Large Language Model
LRM	Large Reasoning Model
MAE	Mean Absolute Error
MRI	Magnetic Resonance Imaging
MRM	Model Risk Management
QA	Quality Assurance
RAG	Retrieval-Augmented Generation

ReAct	Reasoning and Acting
RL	Reinforcement Learning
ROI	Region of Interest
ToT	Tree of Thoughts
VLM	Vision-Language Model

References

1. Gevarter, W.B. Expert systems: Artificial intelligence applied. *Telemat. Inform.* **1984**, *1*, 239–251. https://doi.org/10.1016/s0736-5853(84)80030-1.
2. Jordan, M.I.; Mitchell, T.M. Machine learning: Trends, perspectives, and prospects. *Science* **2015**, *349*, 255–260. https://doi.org/10.1126/science.aaa8415.
3. Alzubaidi, L.; Zhang, J.; Humaidi, A.J.; et al. Review of deep learning: concepts, CNN architectures, challenges, applications, future directions. *J. Big Data* **2021**, *8*, 53. https://doi.org/10.1186/s40537-021-00444-8.
4. Yu, Y.; Si, X.; Hu, C.; et al. A review of recurrent neural networks: LSTM cells and network architectures. *Neural Comput.* **2019**, *31*, 1235–1270. https://doi.org/10.1162/neco_a_01199.
5. Breiman, L. Statistical modeling: The two cultures (with comments and a rejoinder by the author). *Stat. Sci.* **2001**, *16*, 199–231. https://doi.org/10.1214/ss/1009213726.
6. Vaswani, A.; Shazeer, N.; Parmar, N.; et al. Attention is all you need. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 5998–6008.
7. Budnikov, M.; Bykova, A.; Yamshchikov, I.P. Generalization potential of large language models. *Neural Comput. Appl.* **2025**, *37*, 1973–1997. https://doi.org/10.1007/s00521-024-10827-6.
8. Brown, T.; Mann, B.; Ryder, N.; et al. Language models are few-shot learners. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 1877–1901.
9. Bommasani, R.; Hudson, D.A.; Adeli, E.; et al. On the opportunities and risks of foundation models. *arXiv* **2021**, arXiv:2108.07258. https://doi.org/10.48550/arxiv.2108.07258.
10. Acharya, D.B.; Kuppan, K.; Divya, B. Agentic AI: Autonomous intelligence for complex goals—A comprehensive survey. *IEEE Access* **2025**, *13*, 18912–18936. https://doi.org/10.1109/access.2025.3532853.
11. Schneider, J. Generative to Agentic AI: Survey, Conceptualization, and Challenges. *arXiv* **2025**, arXiv:2504.18875. https://doi.org/10.48550/arxiv.2504.18875.
12. Abou Ali, M.; Dornaika, F.; Charafeddine, J. Agentic AI: a comprehensive survey of architectures, applications, and future directions. *Artif. Intell. Rev.* **2025**, *59*, 11. https://doi.org/10.1007/s10462-025-11422-4.
13. Srinivasu, P.N.; Aruna Kumari, G.L.; Ahmed, S.; et al. Exploring Agentic AI in Healthcare: A Study on Its Working Mechanism. *Front. Med.* **2026**, *12*, 1753443. https://doi.org/10.3389/fmed.2025.1753443.
14. Karunanayake, N. Next-generation agentic AI for transforming healthcare. *Inform. Health* **2025**, *2*, 73–83. https://doi.org/10.1016/j.infoh.2025.03.001.
15. Yao, S.; Zhao, J.; Yu, D.; et al. ReAct: Synergizing reasoning and acting in language models. In Proceedings of the Eleventh International Conference on Learning Representations, Kigali, Rwanda, 1–5 May 2023.
16. Shinn, N.; Cassano, F.; Gopinath, A.; et al. Reflexion: Language agents with verbal reinforcement learning. *Adv. Neural Inf. Process. Syst.* **2023**, *36*, 8634–8652. https://doi.org/10.52202/075280-0377.
17. Schick, T.; Dwivedi-Yu, J.; Dessi, R.; et al. Toolformer: Language Models Can Teach Themselves to Use Tools. In Proceedings of the Advances in Neural Information Processing Systems 36, New Orleans, LA, USA, 10–16 December 2023; pp. 68539–68551. https://doi.org/10.52202/075280-2997.
18. Boiko, D.A.; MacKnight, R.; Kline, B.; et al. Autonomous chemical research with large language models. *Nature* **2023**, *624*, 570–578. https://doi.org/10.1038/s41586-023-06792-0.
19. Wu, S. Introducing Devin, the first AI software engineer. Cognition Labs Blog. 2024. Available online: https://cognition.com/blog/introducing-devin (accessed on 12 January 2026).
20. Horvitz, E. Principles of mixed-initiative user interfaces. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Pittsburgh, PA, USA, 15–20 May 1999. https://doi.org/10.1145/302979.303030.
21. Scerri, P.; Pynadath, D.V.; Tambe, M. Towards adjustable autonomy for the real world. *J. Artif. Intell. Res.* **2002**, *17*, 171–228. https://doi.org/10.1613/jair.1037.
22. Sapkota, R.; Roumeliotis, K.I.; Karkee, M. AI agents vs. agentic AI: A conceptual taxonomy, applications and challenges. *Inf. Fusion* **2026**, *126*, 103599. https://doi.org/10.1016/j.inffus.2025.103599.
23. Hosseini, S.; Seilani, H. The role of agentic AI in shaping a smart future: A systematic review. *Array* **2025**, *26*, 100399. https://doi.org/10.1016/j.array.2025.100399.
24. Bandi, A.; Kongari, B.; Naguru, R.; et al. The Rise of Agentic AI: A Review of Definitions, Frameworks, Architectures, Applications, Evaluation Metrics, and Challenges. *Future Internet* **2025**, *17*, 404. https://doi.org/10.3390/fi17090404.
25. Rao, A.S.; Georgeff, M.P. BDI agents: from theory to practice. In Proceedings of the First International Conference on Multiagent Systems (ICMAS), San Francisco, CA, USA, 12–14 June 1995.

26. Bernstein, D.S.; Givan, R.; Immerman, N.; et al. The complexity of decentralized control of Markov decision processes. *Math. Oper. Res.* **2002**, *27*, 819–840. https://doi.org/10.1287/moor.27.4.819.297.
27. Hong, S.; Zhuge, M.; Chen, J.; et al. MetaGPT: Meta Programming for A Multi-Agent Collaborative Framework. In Proceedings of the Twelfth International Conference on Learning Representations, Vienna, Austria, 7–11 May 2024.
28. Xu, F.; Hao, Q.; Shao, C.; et al. Toward large reasoning models: A survey of reinforced reasoning with large language models. *Patterns* **2025**, *6*, 101370. https://doi.org/10.1016/j.patter.2025.101370.
29. Wei, J.; Wang, X.; Schuurmans, D.; et al. Chain-of-thought prompting elicits reasoning in large language models. *Adv. Neural Inf. Process. Syst.* **2022**, *35*, 24824–24837. https://doi.org/10.52202/068431-1800.
30. Yao, S.; Yu, D.; Zhao, J.; et al. Tree of thoughts: Deliberate problem solving with large language models. *Adv. Neural Inf. Process. Syst.* **2023**, *36*, 11809–11822. https://doi.org/10.52202/075280-0517.
31. Parisi, A.; Zhao, Y.; Fiedel, N. TALM: Tool Augmented Language Models. *arXiv* **2022**, arXiv:2205.12255. https://doi.org/10.48550/arxiv.2205.12255.
32. Qin, Y.; Liang, S.; Ye, Y.; et al. ToolLLM: Facilitating Large Language Models to Master 16000+ Real-world APIs. *arXiv* **2023**, arXiv:2307.16789.
33. Patil, S.G.; Zhang, T.; Wang, X.; et al. Gorilla: Large Language Model Connected with Massive APIs. In Proceedings of the Advances in Neural Information Processing Systems 37, Vancouver, BC, Canada, 10–15 December 2024; pp. 126544–126565. https://doi.org/10.52202/079017-4020.
34. Gao, L.; Madaan, A.; Zhou, S.; et al. Pal: Program-aided language models. In Proceedings of the 40th International Conference on Machine Learning, Honolulu, HI, USA, 23–29 July 2023.
35. Karpas, E.; Abend, O.; Belinkov, Y.; et al. MRKL Systems: A modular, neuro-symbolic architecture that combines large language models, external knowledge sources and discrete reasoning. *arXiv* **2022**, arXiv:2205.00445. https://doi.org/10.48550/arxiv.2205.00445.
36. Ruan, J.; Chen, Y.; Zhang, B.; et al. TPTU: Large Language Model-based AI Agents for Task Planning and Tool Usage. In Proceedings of the Foundation Models for Decision Making Workshop, New Orleans, LA, USA, 15 December 2023. https://doi.org/10.48550/arxiv.2308.03427.
37. Lewis, P.; Perez, E.; Piktus, A.; et al. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 9459–9474.
38. Li, G.; Hammoud, H.; Itani, H.; et al. Camel: Communicative agents for "mind" exploration of large language model society. *Adv. Neural Inf. Process. Syst.* **2023**, *36*, 51991–52008. https://doi.org/10.52202/075280-2264.
39. Wu, Q.; Bansal, G.; Zhang, J.; et al. Autogen: Enabling next-gen LLM applications via multi-agent conversations. In Proceedings of the First Conference on Language Modeling, Philadelphia, PA, USA, 7–9 October 2024.
40. Li, X.; Wang, S.; Zeng, S.; et al. A survey on LLM-based multi-agent systems: workflow, infrastructure, and challenges. *Vicinagearth* **2024**, *1*, 9. https://doi.org/10.1007/s44336-024-00009-2.
41. Radeva, I.; Popchev, I.; Doukovska, L.; et al. Multi-Agent Coordination Strategies vs. Retrieval-Augmented Generation in LLMs: A Comparative Evaluation. *Electronics* **2025**, *14*, 4883. https://doi.org/10.3390/electronics14244883.
42. Xu, X.; Wu, J. Mitigating LLM Hallucination Snowballing in Multiagent Systems via Context-Aware Semantic Consistency Reasoning. *IEEE Trans. Neural Netw. Learn. Syst.* **2026**, *37*, 3782–3796. https://doi.org/10.1109/tnnls.2026.3655508.
43. Jimenez, C.E.; Yang, J.; Wettig, A.; et al. Swe-bench: Can language models resolve real-world github issues? In Proceedings of the International Conference on Learning Representations, Vienna, Austria, 7–11 May 2024.
44. Deng, X.; Da, J.; Pan, E.; et al. SWE-Bench Pro: Can AI Agents Solve Long-Horizon Software Engineering Tasks? *arXiv* **2025**, arXiv:2509.16941. https://doi.org/10.48550/arxiv.2509.16941.
45. Yang, J.; Jimenez, C.E.; Wettig, A.; et al. SWE-agent: Agent-Computer Interfaces Enable Automated Software Engineering. *Adv. Neural Inf. Process. Syst.* **2024**, *37*, 50528–50652. https://doi.org/10.52202/079017-1601.
46. Zhang, Y.; Ruan, H.; Fan, Z.; et al. AutoCodeRover: Autonomous Program Improvement. In Proceedings of the 33rd ACM SIGSOFT International Symposium on Software Testing and Analysis, Vienna, Austria, 16–20 September 2024; pp. 1592–1604. https://doi.org/10.1145/3650212.3680384.
47. Tao, W.; Zhou, Y.; Wang, Y.; et al. MAGIS: LLM-Based Multi-Agent Framework for GitHub Issue Resolution. *Adv. Neural Inf. Process. Syst.* **2024**, *37*, 51963–51993. https://doi.org/10.52202/079017-1647.
48. GPT-4. Available online: https://openai.com/research/gpt-4 (accessed on 10 February 2026).
49. Wang, X.; Li, B.; Song, Y.; et al. Openhands: An open platform for AI software developers as generalist agents. In Proceedings of the International Conference on Learning Representations, Singapore, Singapore, 24–28 April 2025.
50. Bran, A.M.; Cox, S.; Schilter, O.; et al. Augmenting large language models with chemistry tools. *Nat. Mach. Intell.* **2024**, *6*, 525–535. https://doi.org/10.1038/s42256-024-00832-8.
51. Romera-Paredes, B.; Barekatain, M.; Novikov, A.; et al. Mathematical discoveries from program search with large language models. *Nature* **2024**, *625*, 468–475. https://doi.org/10.1038/s41586-023-06924-6.

52. Leeman, J.; Liu, Y.; Stiles, J.; et al. Challenges in High-Throughput Inorganic Materials Prediction and Autonomous Synthesis. *PRX Energy* **2024**, *3*, 011002. https://doi.org/10.1103/PRXEnergy.3.011002.
53. Jain, A.; Ong, S.P.; Hautier, G.; et al. Commentary: The Materials Project: A materials genome approach to accelerating materials innovation. *APL Mater.* **2013**, *1*, 011002. https://doi.org/10.1063/1.4812323.
54. Jumper, J.; Evans, R.; Pritzel, A.; et al. Highly accurate protein structure prediction with AlphaFold. *Nature* **2021**, *596*, 583–589. https://doi.org/10.1038/s41586-021-03819-2.
55. Gao, S.; Fang, A.; Huang, Y.; et al. Empowering biomedical discovery with AI agents. *Cell* **2024**, *187*, 6125–6151. https://doi.org/10.1016/j.cell.2024.09.022.
56. Fehlis, Y.; Crain, C.; Jensen, A.; et al. Accelerating Drug Discovery Through Agentic AI: A Multi-Agent Approach to Laboratory Automation in the DMTA Cycle. *arXiv* **2025**, arXiv:2507.09023.
57. Gómez-Tamayo, J.C.; Tavernier, J.; Aerts, R.; et al. MolAgent: Biomolecular Property Estimation in the Agentic Era. *J. Chem. Inf. Model.* **2025**, *65*, 10808–10818. https://doi.org/10.1021/acs.jcim.5c01938.
58. Kodela, K.C. Autonomous Agentic AI Systems for Pharmaceutical Drug Discovery: A Multi-Agent Framework for Molecular Design and Optimization. *SSRN* **2025**, 5382801. https://doi.org/10.2139/ssrn.5382801.
59. Wijaya, E. Beyond SMILES: Evaluating Agentic Systems for Drug Discovery. *arXiv* **2026**, arXiv:2602.10163.
60. Li, S.; Xu, J.; Bao, T.; et al. A co-evolving agentic AI system for medical imaging analysis. *arXiv* **2025**, arXiv:2509.20279. https://doi.org/10.48550/arxiv.2509.20279.
61. Hoopes, A.; Dey, N.; Butoi, V.I.; et al. VoxelPrompt: A Vision Agent for End-to-End Medical Image Analysis. *arXiv* **2024**, arXiv:2410.08397. https://doi.org/10.48550/arxiv.2410.08397.
62. Li, J.; Lai, Y.; Li, W.; et al. Agent Hospital: A Simulacrum of Hospital with Evolvable Medical Agents. *arXiv* **2024**, arXiv:2405.02957.
63. Zuo, K.; Jiang, Y.; Mo, F.; et al. KG4Diagnosis: A Hierarchical Multi-Agent LLM Framework with Knowledge Graph Enhancement for Medical Diagnosis. In Proceedings of the AAAI Bridge Program on AI for Medicine and Healthcare, Philadelphia, PA, USA, 25 February 2025.
64. Zhou, Y.; Zhang, P.; Song, M.; et al. Zodiac: A cardiologist-level llm framework for multi-agent diagnostics. *arXiv* **2024**, arXiv:2410.02026.
65. Rizinski, M.; Trajanov, D. AI Agents in Finance and Fintech: A Scientific Review of Agent-Based Systems, Applications, and Future Horizons. *Comput. Mater. Contin.* **2026**, *86*, 1–34. https://doi.org/10.32604/cmc.2025.069678.
66. Xiao, Y.; Sun, E.; Luo, D.; et al. TradingAgents: Multi-Agents LLM Financial Trading Framework. *arXiv* **2024**, arXiv:2412.20138.
67. crewAIInc. crewAIInc/crewAI. Available online: https://github.com/crewAIInc/crewAI (accessed on 1 February 2026).
68. Okpala, I.; Golgoon, A.; Kannan, A.R. Agentic AI Systems Applied to tasks in Financial Services: Modeling and model risk management crews. *arXiv* **2025**, arXiv:2502.05439.
69. Park, J.S.; O'Brien, J.; Cai, C.J.; et al. Generative agents: Interactive simulacra of human behavior. In Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology, San Francisco, CA, USA, 29 October 2023–1 November 2023. https://doi.org/10.1145/3586183.3606763.
70. Ashery, A.F.; Aiello, L.M.; Baronchelli, A. Emergent social conventions and collective bias in LLM populations. *Sci. Adv.* **2025**, *11*, eadu9368. https://doi.org/10.1126/sciadv.adu9368.
71. Takata, R.; Masumori, A.; Ikegami, T. Spontaneous Emergence of Agent Individuality Through Social Interactions in Large Language Model-Based Communities. *Entropy* **2024**, *26*, 1092. https://doi.org/10.3390/e26121092.
72. Iyidogan, E.; Ozkes, A.I. Agentic AI and hallucinations. *Econ. Lett.* **2025**, *255*, 112520.
73. Salehi, S.; Singh, Y.; Horst, K.K.; et al. Agentic AI and Large Language Models in Radiology: Opportunities and Hallucination Challenges. *Bioengineering* **2025**, *12*, 1303. https://doi.org/10.3390/bioengineering12121303.
74. Guo, T.; Chen, X.; Wang, Y.; et al. Large language model based multi-agents: A survey of progress and challenges. *arXiv* **2024**, arXiv:2402.01680.
75. Rahman, S.S.; Islam, M.A.; Alam, M.M.; et al. Hallucination to truth: a review of fact-checking and factuality evaluation in large language models. *Artif. Intell. Rev.* **2026**, *59*, 1–54. https://doi.org/10.1007/s10462-025-11454-w.
76. Gou, Z.; Shao, Z.; Gong, Y.; et al. CRITIC: Large Language Models Can Self-Correct with Tool-Interactive Critiquing. *arXiv* **2023**, arXiv:2305.11738. https://doi.org/10.48550/arxiv.2305.11738.
77. Dhuliawala, S.; Komeili, M.; Xu, J.; et al. Chain-of-Verification Reduces Hallucination in Large Language Models. In Proceedings of the Findings of the Association for Computational Linguistics: ACL 2024, Bangkok, Thailand, 11–16 August 2024; pp. 3563–3578. https://doi.org/10.18653/v1/2024.findings-acl.212.
78. Yang, H.; Yue, S.; He, Y. Auto-GPT for Online Decision Making: Benchmarks and Additional Opinions. *arXiv* **2023**, arXiv:2306.02224. https://doi.org/10.48550/arxiv.2306.02224.
79. Mustahsan, Z.; Lim, A.; Anand, M.; et al. Stochasticity in Agentic Evaluations: Quantifying Inconsistency with Intraclass Correlation. *arXiv* **2025**, arXiv:2512.06710.

80. Mehta, A. When Agents Disagree With Themselves: Measuring Behavioral Consistency in LLM-Based Agents. *arXiv* **2026**, arXiv:2602.11619. https://doi.org/10.48550/arxiv.2602.11619.
81. Fang, J.; Peng, Y.; Zhang, X.; et al. A comprehensive survey of self-evolving AI agents: A new paradigm bridging foundation models and lifelong agentic systems. *arXiv* **2025**, arXiv:2508.07407. https://doi.org/10.48550/arxiv.2508.07407.
82. Gartner Survey Finds Just 15% of IT Application Leaders Are Considering, Piloting, or Deploying Fully Autonomous AI Agents. Available online: https://www.gartner.com/en/newsroom/press-releases/2025-09-30-gartner-survey-finds-just-15-percent-of-it-application-leaders-are-considering-piloting-or-deploying-fully-autonomous-ai-agents (accessed on 10 February 2026).
83. Radford, A.; Wu, J.; Child, R.; et al. Language models are unsupervised multitask learners. Available online: https://cdn.openai.com/better-language-models/language_models_are_unsupervised_multitask_learners.pdf (accessed on 31 January 2026).
84. Holter, S.; El-Assady, M. Deconstructing Human-AI Collaboration: Agency, Interaction, and Adaptation. In *Computer Graphics Forum*; Wiley: Hoboken, NJ, USA, 2024.
85. Feng, X.; Chen, Z.Y.; Qin, Y.; et al. Large language model-based human-agent collaboration for complex task solving. In Proceedings of the Findings of the Association for Computational Linguistics: EMNLP 2024, Miami, FL, USA, 12–16 November 2024; pp. 1336–1357. https://doi.org/10.18653/v1/2024.findings-emnlp.72.
86. Jennings, N.R.; Sycara, K.; Wooldridge, M. A roadmap of agent research and development. *Auton. Agents Multi-Agent Syst.* **1998**, *1*, 7–38. https://doi.org/10.1023/a:1010090405266.
87. Wooldridge, M.; Jennings, N.R. Intelligent agents: Theory and practice. *Knowl. Eng. Rev.* **1995**, *10*, 115–152. https://doi.org/10.1017/s0269888900008122.
88. Martin, C.; Barber, K.S. Adaptive decision-making frameworks for dynamic multi-agent organizational change. *Auton. Agents Multi-Agent Syst.* **2006**, *13*, 391–428. https://doi.org/10.1007/s10458-006-0009-8.
89. Suri, M.; Mathur, P.; Lipka, N.; et al. Structured Uncertainty guided Clarification for LLM Agents. *arXiv* **2025**, arXiv:2511.08798. https://doi.org/10.48550/arxiv.2511.08798.
90. Han, J.; Buntine, W.; Shareghi, E. Towards uncertainty-aware language agent. In Proceedings of the Findings of the Association for Computational Linguistics: ACL 2024, Bangkok, Thailand, 11–16 August 2024. https://doi.org/10.18653/v1/2024.findings-acl.398.
91. Sarikaya, R. Path to artificial general intelligence: past, present, and future. *Annu. Rev. Control.* **2025**, *60*, 101021.
92. Morris, M.R.; Sohl-Dickstein, J.; Fiedel, N.; et al. Levels of AGI for Operationalizing Progress on the Path to AGI. *arXiv* **2023**, arXiv:2311.02462. https://doi.org/10.48550/arxiv.2311.02462.
93. Preparedness Framework. Available online: https://openai.com/index/updating-our-preparedness-framework/ (accessed on 31 January 2026).
94. Grace, K.; Sandkühler, J.F.; Stewart, H.; et al. Thousands of AI authors on the future of AI. *J. Artif. Intell. Res.* **2025**. https://doi.org/10.1613/jair.1.19087.
95. Müller, V.C.; Bostrom, N. Future progress in artificial intelligence: A survey of expert opinion. In *Fundamental Issues of Artificial Intelligence*; Springer: Cham, Switzerland, 2016; pp. 555–572. https://doi.org/10.1007/978-3-319-26485-1_33.
96. Grace, K.; Salvatier, J.; Dafoe, A.; et al. Viewpoint: When Will AI Exceed Human Performance? Evidence from AI Experts. *J. Artif. Intell. Res.* **2018**, *62*, 729–754. https://doi.org/10.1613/jair.1.11222.
97. Bengio, Y.; Hinton, G.; Yao, A.; et al. Managing extreme AI risks amid rapid progress. *Science* **2024**, *384*, 842–845. https://doi.org/10.1126/science.adn0117.