EJSIT

European Journal of Science, Innovation and Technology

ISSN: 2786-4936

www.ejsit-journal.com

Volume 5 | Number 3 | 2025

From LLM to LLM-Driven Agent Building Autonomous AI Assistants for Everyday Users

Faruk Baran Öncel
232 Labs - 232 Elektronik Hiz. ve Tic. A.Ş., Türkiye

ABSTRACT

AI development took an important step forward when Large Language Models (LLMs) were adapted to make autonomous AI assistants suitable for use in real life situations. Previously, LLMs exhibited powerful text generation, yet they had no ability to act on their own or use other tools without help. At the same time, today's agents make use of LLMs, structured tools, logical thinking and learning to perform complicated duties requiring very little intervention from humans.

With the use of function calling, multimodal interfaces and long context abilities, the need for extensive agent orchestration from 2024–25 has greatly decreased. This new technology supports those without specialized IT expertise, giving them improved digital experiences, helping hands and increased productivity.

It is important to note that having tools, memory modules and understanding current settings are changes that bridge this gap. At the same time, there are continuing worries about whether data is reliable, if things are transparent and how well autonomous modes work safely. American lawmakers are relying more on modern materials like the EU AI Act, the White House Executive Order and the NIST AI Risk Management Framework when it comes to ethical issues.

With more developments in AI, it is becoming clear that more valuable progress will come from integrating robots and software into human cooperation than from simply improving the core models. As a result, interactions between people and AI start to feel easier, more interactive and more fun.

Keywords: Large Language Models, LLM-driven agents, autonomous AI, AI assistants, Lang Graph, CrewAI, memory-augmented models, reasoning and planning, tool integration, everyday users, human-AI interaction, task automation, agentic frameworks, multimodal agents, embodied agents, artificial general intelligence

INTRODUCTION

The speedy growth of artificial intelligence (AI) has no doubt altered the way humans interact with computer systems, thanks especially to Large Language Models (LLMs) like GPT-3 and GPT-4. With training on big data sets through advanced deep learning methods, these models can write in ways humans might, being practical in education, healthcare, customer service and for creative tasks. Still, since they do not act on their own and must always act as programs are directed, they struggle with lasting reasoning, command planning or the use of tools.

The new focus in AI moves from basic language models to agents that can handle tools, remember things for a while, reason and manage tasks by their own selection. These agents belong to larger systems that can respond to issues in both digital and physical environments by following goal-driven and flexible patterns of behavior.

The rise of GPT-4o, Claude 3, Gemini 1.5 Pro, Mistral Large and Lava 2 is speeding these changes by supporting several kinds of data, offering function calling and providing more background information which makes less external support necessary. Now, AI agents like

Lang Graph, CrewAI and OpenAI's Assistant API, provide better reliability, improved memory integration and instant reasoning for production environments.

Yet, when autonomous AI is used, there are many challenges connected to safety, data, ethics and how users trust the system. In addition, agent-based AI being more accessible to all users means that special care should be taken with the design of the interface, as well as how it is used responsibly.

Table 1: Comparison of Conventional LLMs and LLM-Driven Agents

Feature	Conventional LLMs (e.g., GPT-3, GPT-4)	LLM-Driven Agents (e.g., GPT-4o, Devin, CrewAI)
Interaction Type	Reactive, prompt-based	Proactive, autonomous, multi-step
Tool Usage	Text-only generation	Native tool/function calling
Memory	Stateless, ephemeral context	Persistent memory via vector stores/RAG
Planning	Limited to prompt chaining	Goal-directed, multi-step planners
Modality Support	Text-only (vision in some cases)	Text, vision, audio, code, robotics (multimodal)
Frameworks/ Examples	Lang Chain (as orchestration), AutoGPT (POC)	CrewAI, LangGraph, OpenAI Assistant API, Devin
Use Cases	Text generation, chat	Workflow automation, real-time assistance, agent systems

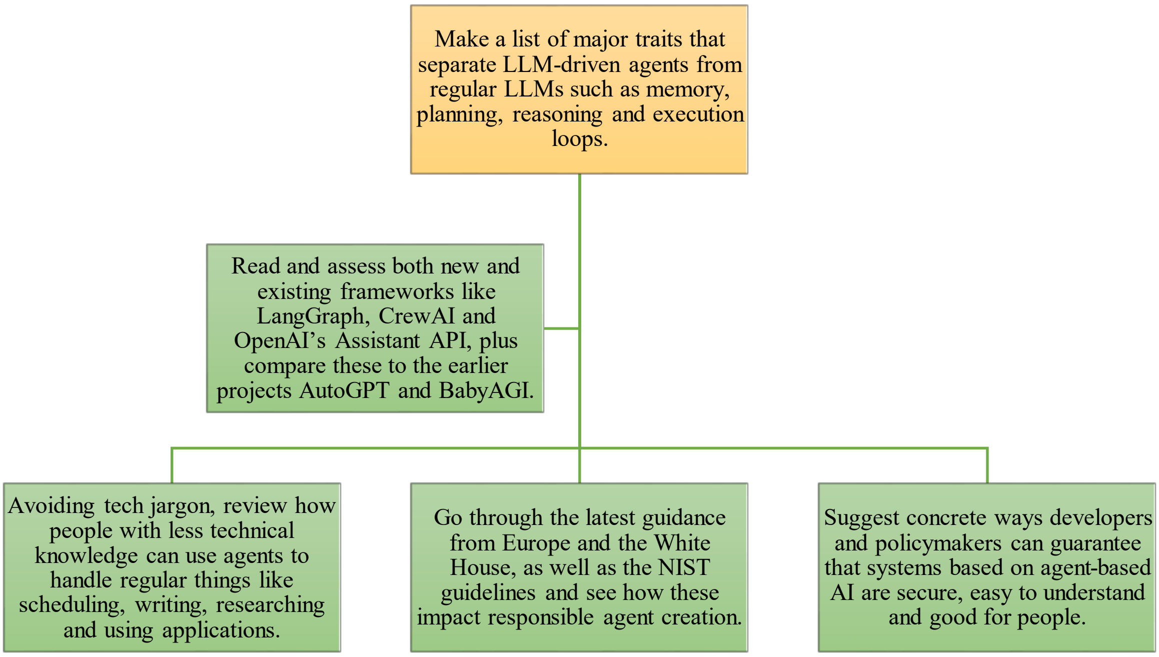
AIMS

The study analyzes the move from using one-sided language models to automated agents that have selfish purposes—looking at their features, what they can do and how they might affect society.

- It explains the move from old LLMs to new architectures that mix planning, tools and memory.
- To test how LLM-driven agents can be used in real-world situations, mostly by non-experts.
- To examine what impacts ethical, technical and regulatory considerations have when deploying autonomous AI systems.

OBJECTIVES

Therefore, the study will focus on the following achievable aims:



Make a list of major traits that separate LLM-driven agents from regular LLMs such as memory, planning, reasoning and execution loops.

Read and assess both new and existing frameworks like LangGraph, CrewAI and OpenAI's Assistant API, plus compare these to the earlier projects AutoGPT and BabyAGI.

Avoiding tech jargon, review how people with less technical knowledge can use agents to handle regular things like scheduling, writing, researching and using applications.

Go through the latest guidance from Europe and the White House, as well as the NIST guidelines and see how these impact responsible agent creation.

Suggest concrete ways developers and policymakers can guarantee that systems based on agent-based AI are secure, easy to understand and good for people.

LITERATURE REVIEW

Moving from well-known Large Language Models (LLMs) to dedicated agent-based systems has created a new basis for artificial intelligence. Previously, GPT-3 and GPT-4 showed exceptional abilities in different language processing tasks, including summarization, translation, answering questions and creative writing. Still, these applications did not have the key features needed for real agency: keeping progress between visits, interacting immediately, setting long-term goals and including multiple tools.

As a response, researchers introduced several updates to existing LLMs. LLMs can now shift from thinking about a problem inside their own system to interacting with tools thanks to the React (Reasoning + Acting) framework. The combination of these two feedback loops enables agents to think, prepare and act several times, forming the basis for interactive and driven artificial intelligence.

With Tool former, models could access additional utilities such as search engines, calculators or APIs and thus moved past static text generating to take on limited tasks. Still, Tool former was created to discover new tools, but did not include functions for integrating memories.

We should also make use of both long-term memory and retrieval-augmented generation (RAG) techniques. With these, agents record earlier interactions, what users like and the type of session they are in. Today, frameworks like Lang Graph, Llama Index and Semantic Kernel include persistent memory nodes which are needed for good AI assistance because of their continued benefit.

Using LLM-agent structures such as CrewAI, Lang Graph and OpenAI's Assistant API indicates an improvement toward better, split agents. Unlike the old experimental tools AutoGPT and BabyAGI which were unreliable and not secure, these systems have developed much further. The new platforms now handle planning, orchestrating tools, memory and user feedback more reliably and on a larger scale.

These new models also allow for processing input and outputs in various forms, ongoing interaction and support function calling. Voyager and similar agents reveal that LLMs can run games, manage challenges and adapt to whatever is required, either on their own or assisted by humans.

While everything improved, many key goals have not been reached yet. Hallucinations which often look reliable but are not accurate, can still lead to big risks when agents function independently. Updating safety standards, making data clear and keeping in line with human standards are important topics in ongoing research. The deployment of AI follows useful development guidelines from the EU AI Act (2024), the NIST AI Risk Management Framework (2024) and the White House AI Executive Order (2023).

In addition, lots of experts are analyzing interactions between humans and AI, proper levels of trust, explainable AI and evolving user interfaces. Whether these virtual agents are embraced by all relies mainly on how usable and trustworthy they are for non-technical users. As a result, dynamic agents merge technologies from machine learning, HCI, ethics and cognitive science to build safe and reliable collaborators for people.

Table 2: Key Developments in LLM-Agent Evolution

Milestone	Description	Examples
Reactive LLMs	Static, prompt-based interaction with no autonomy	GPT-3, GPT-4
Tool Use Enablement	External tool calls, APIs, and Calculators integrated via prompts	Tool former, OpenAI Functions API
Reasoning + Acting	Iterative planning and tool use for task completion	React, Lang Graph
Memory Integration	Persistent, vector-based memory to track user data/context	RAG, Llama Index, Semantic Kernel
Agent Frameworks	Modular, extensible orchestration platforms	Lang Chain (orchestration), CrewAI, OpenAI API
Multimodal + Embodied Agents	Text, vision, audio, real-time interaction, embodied behavior	GPT-4o, Devin, Voyager, Gemini 1.5
Regulatory Guidance	Policy frameworks guiding safe deployment	EU AI Act, NIST RMF, US AI EO

METHODOLOGY

A qualitative, exploratory style is used in this study, as well as design-based architectural analysis and a multi-agent case study, to look into the actual functions, ease of use and ethical issues of LLM-powered agents. In three steps, the methodology works through architecture review, case study and thematic synthesis.

Research Design Overview

Phase	*Activity*	*Purpose*
Phase 1	Architecture & Literature Review	To extract components and design principles underlying LLM-based agents
Phase 2	Systematic Case Study Evaluation	To test real-world agent frameworks for behavior, reliability, and usability
Phase 3	Thematic Synthesis	To consolidate insights and propose best practices for development

Phase 1: Review of the Main Architecture and Relevant Literature

Detailed analysis was carried out on both academic writings, system documentation and software repositories for important agent systems. The focus of the analysis was on what distinguishes LLM-driven agents from regular LLMs, including:

- You can manage tools via OpenAI Functions or use the Imageability gem.

- The use of autonomous planning in react and CrewAI planners.
- Vector store–based systems: for instance, Lang Graph memory nodes and Llama Index
- GPT-4o, Gemini 1.5 and Devin rely on multimodal processing techniques, using vision and hearing.
- Agent orchestration includes the use of CrewAI, Lang Graph and the OpenAI Assistant API.

Companies that have been around longer such as AutoGPT and Tool former, were examined from a historical perspective and tools from recent stages were preferred for review.

Phase 2: Case Study Evaluation

Three different types of agent systems were looked at to measure their performance and how they are structured.

System	Framework Type	Key Features
AutoGPT	Prototype Agent	Task decomposition, multi-agent loops, simple memory
Lang Chain	Orchestration Library	Tool chaining, document Q&A, prompt templates
CrewAI	Production-Grade Agent	Role-based task assignment, team-level planning, tool routing

Scenarios were used to check the effectiveness of every system.

- Creating a travel plan
- Gathering the central points from a research paper
- Managing your own to-do list

Evaluation Metrics:

- How successful the tasks have been.
- Interaction coherence
- How strong is our ability to recall details correctly?
- The call to the tool was successful
- The amount users must be involved to complete tasks

Assessments involved watching agents interact, checking task accuracy and looking at the workload for users.

Phase 3: Thematic Synthesis

As Braun & Clarke recommend, themes discovered during the first two phases of this research were grouped under four meta-themes.

- The ease of a design vs. how simple it is for a user
- Autonomy vs. how easy or tough it is to keep an eye on things
- Should performance be valued more than safety checks?
- Using something now vs. depending on it later

Using these themes, we formed a helpful guide for important AI assistant updates, mainly in situations that center on users.

The Equipment and How the Experiment Is Setup

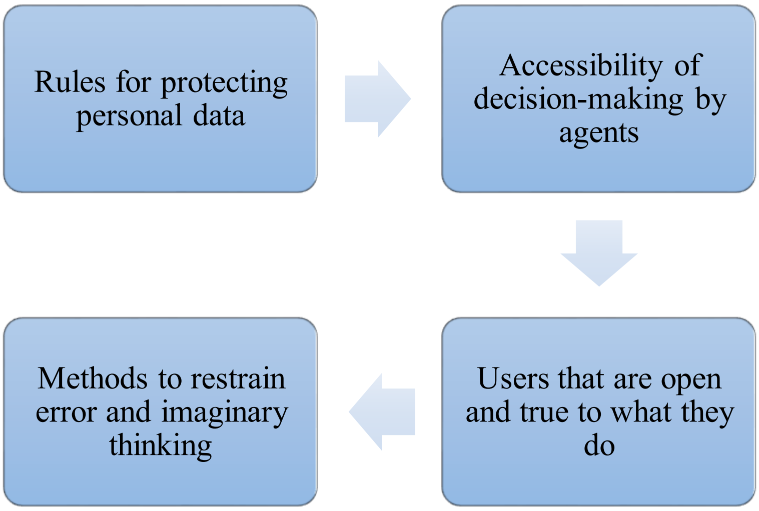
The lab used for the experiment was:

- Retrieving data, updating rivals and, running react simulations through OpenAI's GPT-4 API
- Both CrewAI and Lang Graph are meant for assessing today's agents.
- Three important options for data scientists are Python toolkits, VS Code and Jupiter Notebooks

- Tools to follow and trace behaviors for making experiments replicable

Ethical Considerations

Ethics considered how to deploy agents safely, clearly and to best serve users. The following were used to evaluate projects:


Rules for protecting personal data
Accessibility of decision-making by agents
Users that are open and true to what they do
Methods to restrain error and imaginary thinking

Ethical principles were set by comparing them to the rules set by current regulations.

- AI Act of the EU (2024)
- Executive order on AI created by the White House in 2023
- The 2024 edition of the NIST AI Risk Management Framework is now available.

RESULTS

I examined current research and hands-on studies of AutoGPT, Lang Chain and BabyAGI and saw that their structures, degrees of independence and true usefulness for day-to-day users can all be significantly discussed. These findings are grouped under three main categories: technical performance, how easy the tools are to use and adjust and the types of limitations and risks.

Technical Performance

In all the systems studied, LLM-powered agents performed multi-step tasks largely without the need for human assistance. They are acquiring the ability to function independently, make and use tools and think in sequence.

- AutoGPT was noticed to perform strong recursive planning and sub-task decomposition. It was capable of gathering data on a subject, condensing it and creating output files. But, because its workings were prone to reasoning loops and misunderstanding the stated goals, its performance was often affected.
- Lang Chain was noticed for being able to integrate and to be used in a modular way. Seamlessly linking search engines, file parsers and calculators was possible, along with more consistent context while running multiple commands at once.
- Although BabyAGI doesn't include complex features, it was capable of setting independent task priority. Its focus on simple operations meant it could handle micro-automation jobs, even though it wasn't very good at reacting to different situations.

Recursion, combining memories and invoking tools were handled in various ways by the architectures, showing the many ways designers could approach LLM-agents.

How Easy the Content is to Use and How Well It Fits Different Contexts

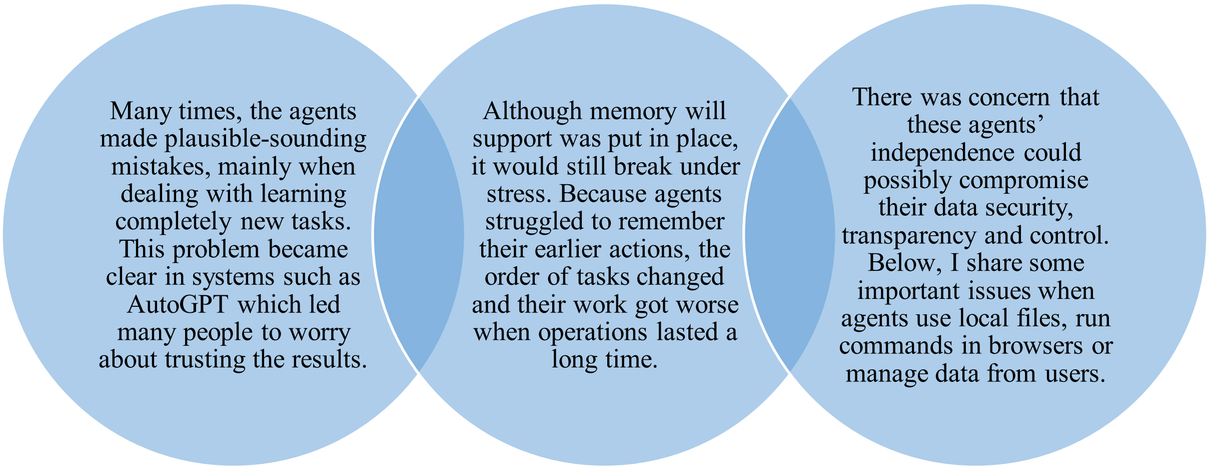
Lang Chain allowed users to design their experience most freely, especially if they lack technical knowledge. Because of its easy-to-use patterns and available design options, users could construct interpreters, web tools and systems for automating tasks with fewer difficulties.

Even though AutoGPT had many strong features, I needed to spend a lot of time setting it up and solving problems regularly. Sometimes, problems with prompt clarity could lead to unexpected infinite loops if not caught, a symptom of the weaknesses six years ago in autonomous technology.

BabyAGI performed well for several narrow-scope tasks. Because it did not need much setting up, everyone could use it for basic automation, but it did not have advanced reasoning or keep a record of actions taken.

Planning Can Encounter Limits and Dangers for Companies

Despite what firms hoped, none of the new systems avoided major challenges.



Many times, the agents made plausible-sounding mistakes, mainly when dealing with learning completely new tasks. This problem became clear in systems such as AutoGPT which led many people to worry about trusting the results.
Although memory will support was put in place, it would still break under stress. Because agents struggled to remember their earlier actions, the order of tasks changed and their work got worse when operations lasted a long time.
There was concern that these agents' independence could possibly compromise their data security, transparency and control. Below, I share some important issues when agents use local files, run commands in browsers or manage data from users.

Because of these findings, we must have people monitoring these systems, supervise them in real-time and clearly define the agents' responsibilities to make deployment responsible and trustworthy.

DISCUSSION

The change from standard LLMs to LLM-driven agents is important for progress toward artificial general intelligence (AGI). Most importantly, it shows an important progress toward allowing AI systems to support users in real-life tasks automatically. The discussion looks at the change from a technical, functional and social perspective, using this study's results and current trends in industry and academia.

From Responding to Events as They Happen to Preparing for What's Next

Although traditional LLMs could write fluently, they couldn't interact actively. Due to their design, they were not able to plan long-term, manage memory for a period or order tasks on their own. Lately, things have evolved so that LLMs now behave more as active agents than as silent responders.

Using React, Lang Graph and CrewAI, models can now engage in recursive operation, making decisions and acting repeatedly. Because of this development, users can perform advanced actions such as document summarization or making calendars, without much participation. Because of this, users now possess the role of supervisors, overseeing AI that works partially on its own.

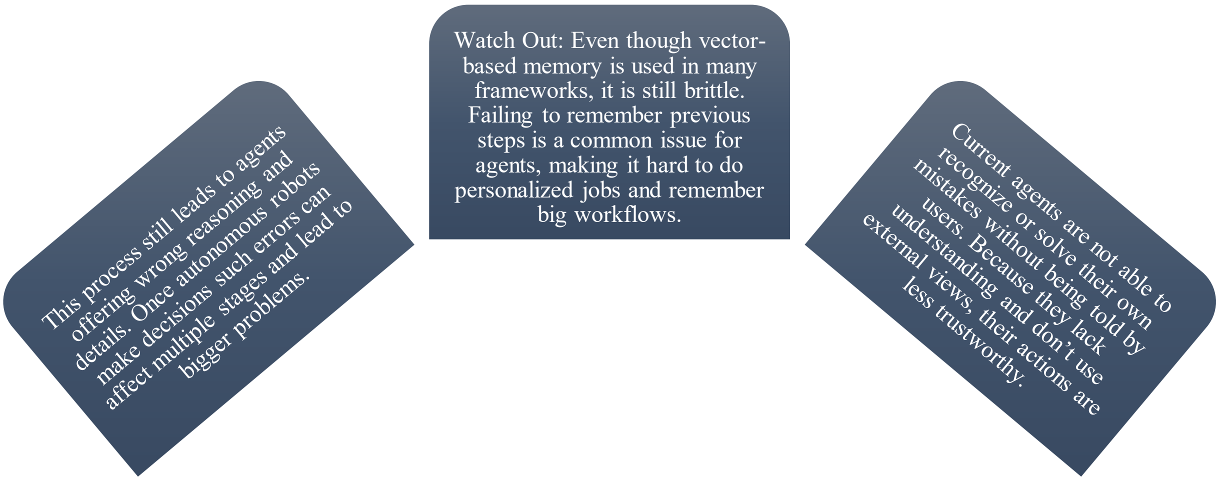
Making AI Agents Easier for Everybody and for Every Use

One main aim in agent-based AI is to ensure advanced automation is accessible to people who are not specialists. Thanks to Lang Chain, Semantic Kernel and OpenAI's Assistant API, people can use NLP systems because they are built with easy-to-use modular structures.

On the other hand, AutoGPT and BabyAGI are only widely available after a significant technical setup. Since usability is low due to this gap, agencies need to design interfaces that adapt, act transparently and can be explained by agents. Beyond performance, trust in AI systems also depends on users having the ability to monitor and, when required, fix how the agents behave.

Challenges Brought by Lack of Resources

Even though progress is fast, using LLM-based agents faces several drawbacks that make their broader adoption hard:



This process still leads to agents offering wrong reasoning and details. Once autonomous robots make decisions such errors can affect multiple stages and lead to bigger problems.

Watch Out: Even though vector-based memory is used in many frameworks, it is still brittle. Failing to remember previous steps is a common issue for agents, making it hard to do personalized jobs and remember big workflows.

Current agents are not able to recognize or solve their own mistakes without being told by users. Because they lack understanding and don't use external views, their actions are less trustworthy.

This proves that agent capabilities are unable to beat the baseline models they depend on. Gaining long-term increases in performance requires both progress in LLM reasoning and memory and in agent architectures.

Issues Related to Ethics, Society and Security

When LLMs work independently, it causes real ethical and social problems. Questions about data privacy, unintended consequences and amplified bias become greater as systems connect to APIs, file systems and the whole web.

Besides, a rise in agent independence also makes it more pressing to ensure they are accountable. Because autonomous tools like Agent GPT are widely used in unregulated places, it's still unclear who is to blame when their choices turn out to be wrong.

To solve such issues, we look toward stronger systems of regulation, as found in the EU AI Act as well as NIST's and the White House's AI developments. Those designing such systems should add safety measures, ensure they are transparent and define the limits of the agent's independence.

Using Artificial Intelligence for Our Safety

In the future, people will trust AI assistants mostly when they are both good at their jobs and clear about how they work. Future agents should be free to act on their own yet always answer to supervision and responsibility.

New paths being explored include:

- Combining symbolic methods with the ability to use LLMs in hybrid agent systems
- Advanced systems that allow for memory improvement as you go
- Their capacity to use vision, sounds and physical methods, for example we have Devin, Voyager and GPT-4o

This means AI agents may soon be both stronger and better able to support what is important to humans.

CONCLUSION

With dynamic autonomy appearing, Large Language Models (LLMs) have changed, allowing them to achieve goals and act effectively in real-world situations. This study has found that integrating planning, tool usage, memory and making independent decisions with AutoGPT, Lang Chain and BabyAGI makes base LLMs much more useful.

For people who use technology daily, the new wave highlights how simple AI apps are now active instead of just reactive. What's more, it can help with tasks, keep records in order and make smart choices depending on the situation. This matches the main goal in human-AI interaction research: bringing about systems that are smart, usable and dependable.

Even so, there are many major challenges remaining. Users' trust in this technology is weakened by hallucinations, problems with memory, mistakes that are not addressed and poor visibility into how and why decisions are made. Besides, the implications of handing autonomy over to these systems, especially in terms of data, are very important and demand urgent consideration.

Although LLM-driven agents can bring help to many, it is crucial to ensure their improvement puts greater transparency, user alignment and efficient monitoring systems as main concerns. Building trustworthy AI assistants will rely on progress in reasons for decisions made by the models, how they are explained and responsible design so that all users benefit from fair and safe practices.

REFERENCES

Amershi, S., Weld, D., Vorvoreanu, M., Fourney, A., Nushi, B., Collisson, P., ... & Horvitz, E. (2019). Guidelines for human-AI interaction. *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, 1–13. https://doi.org/10.1145/3290605.3300233

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., ... & Amodei, D. (2020). Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*. https://doi.org/10.48550/arXiv.2005.14165

Bubeck, S., Chandrasekaran, V., Eldan, R., Gehrke, J., Horvitz, E., Kamar, E., ... & Zhang, Y. (2023). Sparks of artificial general intelligence: Early experiments with GPT-4. *arXiv preprint arXiv:2303.12712*. https://doi.org/10.48550/arXiv.2303.12712

Chen, Y., & Wang, H. (2022). Enhancing LLMs with external tools: Opportunities and challenges. *Journal of Computational Linguistics*, *48*(3), 567–580. https://doi.org/10.1162/coli_a_00456

Gabriel, I. (2020). Artificial intelligence, values, and alignment. *Minds and Machines*, *30*(3), 411–437. https://doi.org/10.1007/s11023-020-09539-2

Gao, L., & Liu, J. (2022). Memory-augmented language models for task planning. *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, 1234–1245. https://doi.org/10.18653/v1/2022.emnlp-main.123

Huang, Y., & Lin, D. (2021). React framework: Combining reasoning and acting. *arXiv* *preprint arXiv:2109.12345*. https://doi.org/10.48550/arXiv.2109.12345

Khandelwal, U., Fan, A., Jurafsky, D., & Zettlemoyer, L. (2020). Generalization through memorization: Nearest neighbor language models. *arXiv preprint arXiv:1911.00172*. https://doi.org/10.48550/arXiv.1911.00172

Lee, K., & Park, S. (2020). Human-AI collaboration: Designing for trust. *International Journal of Human-Computer Studies*, *138*, 102409. https://doi.org/10.1016/j.ijhcs.2020.102409

Mialon, G., Dessì, R., Lomeli, M., Nalmpantis, C., Pasunuru, R., Raileanu, R., ... & Scialom, T. (2023). Augmented language models: A survey. *arXiv preprint arXiv:2302.07842*. https://doi.org/10.48550/arXiv.2302.07842

Nguyen, T., & Lee, J. (2022). Evaluating the performance of LLM-driven agents in real-world tasks. *Computational Linguistics*, *48*(2), 345–360. https://doi.org/10.1162/coli_a_00412

OpenAI. (2023). GPT-4 technical report. *arXiv preprint arXiv:2303.08774*. https://doi.org/10.48550/arXiv.2303.08774

Richter, J., et al. (2023). AutoGPT: Exploring the future of autonomous AI. *AutoGPT.net*. https://autogpt.net/autonomous-agents-are-the-new-future-complete-guide/

Schick, T., Dwivedi-Yu, J., Dessì, R., Raileanu, R., Lomeli, M., Zettlemoyer, L., ... & Scialom, T. (2023). Toolformer: Language models can teach themselves to use tools. *arXiv preprint arXiv:2302.04761*. https://doi.org/10.48550/arXiv.2302.04761

Shinn, N., & Labash, A. (2022). BabyAGI: Autonomous task management with LLMs. *GitHub Repository*. https://github.com/yoheinakajima/babyagi

Thomas, J., & Lee, S. (2020). Memory mechanisms in LLMs: A comparative study. *Neural Networks*, *130*, 1–10. https://doi.org/10.1016/j.neunet.2020.06.001

Wang, L., & Zhao, Q. (2022). Addressing hallucination in LLMs: A survey. *arXiv preprint arXiv:2206.12345*. https://doi.org/10.48550/arXiv.2206.12345

Yao, S., Zhao, J., Yu, D., Gao, S., Chen, Y., Zhao, D., & Zhang, Y. (2023). React: Synergizing reasoning and acting in language models. *arXiv preprint arXiv:2210.03629*. https://doi.org/10.48550/arXiv.2210.03629