Agentic AI

Sushil Kulkarni
Professor & Dean,
NSoMASA, NMIMS University.

Introduction

"Agentic" comes from the word "agency," which refers to the capacity to act independently and make decisions. Agentic AI is designed to take action independently, solving problems and improving processes without waiting for a command.

Let us start with an example of Agentic AI. Suppose two waiters Madan and Swati work in the same restaurant. Madan follows fixed protocols and obeys all instructions given by the manager. He is accurate in his work. He will go to a customer who is coming to a restaurant and then he will offer the food Menu. The customer will order a few things. He will accurately note down without giving extra suggestions or recommendations and deliver it. While Swati is an intelligent and accurate. She will provide extra suggestions. For example, if a customer is ordering Nan and Curry and the weather is cold outside then Swati will proactively suggest hot tomato soup. If a customer is frequent, then Swati will recommend their favourite dish. If a customer complains about cold food, Madan simply reports to the manager and waits for instructions. In contrast, when Swati notices food getting cold, she independently assesses the situation.

Thus in the context of AI, being 'agentic' means the system can perceive, plan, and act autonomously, enabling it to handle complex and dynamic scenarios.

Let us break down how Agentic AI operates through three key capabilities, using our restaurant waiter example:

1. **Perceive:** Swati demonstrates perception by:

   - Noticing weather conditions
   - Recognizing frequent customers
   - Detecting when food is getting cold
   - Observing customer ordering patterns

2. **Plan:** Swati's planning abilities show through:

   - Deciding which dishes to recommend based on the weather
   - Figuring out how to handle cold food situations
   - Creating strategies to prevent future issues

3. **Act Autonomously:** Swati acts independently by:

- Making proactive suggestions without manager's approval
- Taking initiative to solve problems
- Adjusting service based on circumstances
- Implementing solutions without waiting for instructions

These three capabilities work together to handle complex and dynamic scenarios. For instance, when dealing with cold food:

- Perceive: Notices the food temperature issue
- Plan: Determines steps needed (check kitchen, inform customer, offer compensation)
- Act: Implements the solution independently

Let us take another example of the football game. If a ball goes into a bush, the responsibility for retrieving it could fall to the player who kicked it there. However, introducing an agentic AI like Vinay, the robot changes the dynamics significantly.

1. **Perceive**: Vinay uses sensors and cameras to detect the ball's location and assess the environment. It can also identify obstacles, players' positions, and the flow of the game.
2. **Plan**: Instead of simply waiting for instructions, Vinay autonomously develops a plan. For instance, it might calculate the quickest route to the ball while considering safety and game strategy. Vinay could prioritize a swift retrieval to minimize disruption if the game is intense.
3. **Act**: Vinay executes its plan by navigating to the bush, retrieving the ball, and returning it to the field. It might also communicate with players, suggesting optimal positioning or timing for a quick restart of the game.

Definition of agentic AI

Thus, the formal definition of agentic AI can be as follows:

Agentic AI refers to an AI system capable of autonomously perceiving its environment, planning, action and executing tasks to achieve specific goals, all with predefined boundaries.

Or

Define an agent as a sophisticated program or robot designed to independently perform specific tasks. It can make decisions based on predefined rules or learned experiences, interacting with people or its environment to achieve its goals.

Example 1. (Helpful Robot friend): Imagine you have a robot at home. When you return from work looking tired, the robot observes this (perception), considers that you might appreciate a snack (planning), and brings you an apple (action) without you having to ask.

Example 2. (Smart alarm clock) Consider a smart alarm clock that recognizes your usual wake-up time. If it detects that you've stayed up late—perhaps because your bedroom light was on longer—it decides to let you sleep in a little longer and wakes you gently. It perceives your sleep patterns, plans the optimal time to wake you, and then acts by ringing the alarm.

The five key features of Agentic AI

The following are the key features of agentic AI:

(a) **Autonomous operation:** It means the agent can operate autonomously. For instance, an agent monitors the performance of the server. If an agent detects that the server is nearing its capacity, it can trigger a job and automatically give provision for additional resources or activate load balancing, all without requiring intervention from a system administrator.

(b) **Goal-oriented behaviour:** Agents are goal-oriented, meaning they establish objectives and devise strategies to achieve them. For example, consider an AI-driven software deployment agent whose goal is to ensure that all applications in the organization are up to date. It schedules updates during low-traffic periods, resolves dependency issues, and verifies successful installations.

(c) **Learning and evolution:** Agents can learn and evolve by incorporating new information and improving over time. For instance, a cybersecurity AI agent monitors network traffic to identify threats. As it encounters new types of cyberattacks, it learns from them and adapts its algorithms to recognize and block emerging threats more effectively.

(d) **Contextual understanding:** The agent possesses contextual understanding, enabling it to make informed decisions based on various situations. For example, consider an agent that interprets user queries in an IT support system. When an employee reports "email issues," the agent takes into account factors such as recent email server updates, network connectivity, and user account status to offer accurate troubleshooting steps or escalate the issue as needed.

(e) **Multi-domain utility:** Agents have multi-domain utility, meaning they can work across various areas or topics. For example, a virtual IT assistant can handle tasks ranging from network monitoring, system diagnostics, and automating backups, to assisting users with software issues. It seamlessly operates across different domains to support the IT infrastructure efficiently.

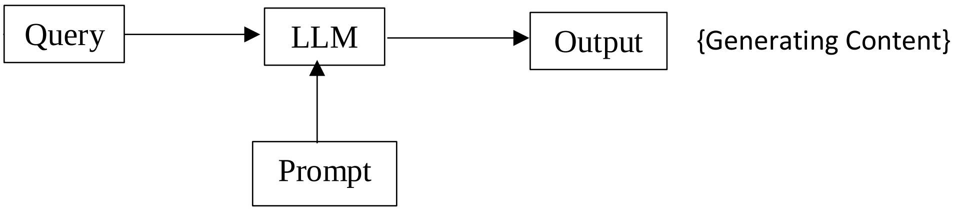
Difference between traditional AI and Agentic AI

The following table shows the difference between traditional AI and Agentic AI:

Feature	Traditional AI	Agentic AI
Decision-making	Follows pre-programmed instructions	Makes independent decisions based on goals
Learning Ability	Learns only from data fed to it	Continuously learns and adapts
Adaptability	Limited to predefined scenarios	Adapts to new situations in real-time
Action Orientation	Focuses mainly on data analysis	Takes actions to improve processes or goals
Complexity Handling	Works best with structured, simple data	Handles complex, dynamic environments

Difference between Generative AI and Agentic AI

Generative AI refers to a class of artificial intelligence models that can generate new content based on the query and prompt they receive. Artificial intelligence models can be Large Language Models (LLMs) like ChatGPT (GPT-4), Claude, Mistral 7B, Gemini, and LLaMA



A **query** refers to a specific request or question posed to the model. It is often concise and can be thought of as a direct inquiry that the user wishes the model to address. A **prompt**, on the other hand, is a broader and more context-rich input that guides the model toward generating a specific type of output.

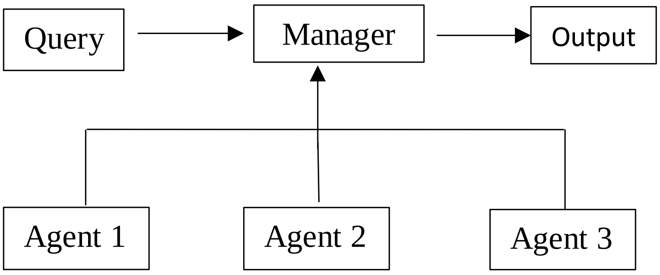
For example, the Query may be "What's the capital of India?" and the Prompt can be "Provide a brief overview of India, including its history and significance as the capital of India".

On the other hand, **agentic AI** is an autonomous AI system that makes decisions. Currently, LLMs are not trained with the most recent data. For instance, if we ask an LLM about the top five current

AI news stories, it won't be able to answer. However, if an LLM could access external sources or tools that could operate autonomously, it would qualify as agentic AI. These tools could include services like DuckDuckGo for finding recent news online, Wikipedia for information retrieval, or archival databases.

The agentic AI process is given below:

A query goes to the manager who interacts with all agents for processing the data. Each agent is having certain goal and tasks. Tasks of agent may connect sequentially. This is the agent framework philosophy where there are many agents working together. Agent 1 might do some task and pass the result to agent 2 but then agent 3 might need something from agent 1 then it goes back to agent 1 and eventually if everyone agrees then passes to the manage and get the output. AutoGen and LangChain for building agents follow similar processes.



Agentic AI Frameworks

Following are a few popular frameworks used:

(a) **LangChain:** It is a framework designed to build applications with language models. It provides tools for chaining components like prompts, data retrieval, and APIs, facilitating the development of complex agentic systems.

(b) **Langflow:** Simplifies the creation and management of language models, enabling developers to design conversational agents and language-based applications efficiently.

(c) **LangGraph:** Provides a framework for representing language models as graphs, allowing for more nuanced interactions and relationships in language processing tasks relevant to agentic behaviour.

(d) **Phidata:** Supports data processing and integration for AI applications, facilitating the analysis and management of data that agents rely on for decision-making and learning.

(e) **Haystack:** An open-source framework for building NLP-powered search systems. It enables the creation of question-answering systems and chatbots, making it suitable for agentic AI applications that require information retrieval.

(f) **LlamaIndex (formerly GPT Index):** A tool that helps manage and query large datasets with large language models. It simplifies the integration of structured data with language models, enhancing their capability in agentic tasks.

(g) **Hugging Face Transformers:** A widely-used library for natural language processing that supports various pre-trained models. It's essential for developing conversational agents and integrating machine learning into agentic systems.

(h) **CrewAI:** It is an advanced artificial intelligence platform designed to streamline team collaboration and project management. It leverages machine learning algorithms to enhance productivity by automating repetitive tasks and improving communication among team members. CrewAI provides real-time analytics and insights, helping organizations make data-driven decisions. Its user-friendly interface and integration capabilities with existing tools make it a valuable asset for any team aiming for efficiency and effectiveness.

Agentic AI architecture

One of the Agentic AI architectures is given below and consists of the following components:

1. **Core AI System:** It consists of three components. **(a)** Foundation AI is a Large Language Model (LLM), which processes inputs and generates responses. **(b)** The planning module creates action plans and sequences tasks. **(c)** Memory Module stores context, previous interactions, and learned patterns. These interconnected components form a cohesive system capable of understanding, planning, and engaging in meaningful interactions.
2. **The Tool Integration Layer** is a vital component that enables an AI agent to effectively interact with and utilize various tools to accomplish tasks autonomously. (a) **The Tool Executive** functions as the agent's control centre, making decisions about which tools to use when, and orchestrating their execution in a strategic sequence to achieve given objectives, **(b) The API Connectors** serve as standardized communication bridges that translate the agent's commands into a format that external tools can understand. For example, converting an agent's request to search for information into the specific format required by a search engine's API, or translating a

request to analyze data into commands that a spreadsheet program can execute. **(c) The Tool Cache** enhances the agent's efficiency by storing frequently used tool configurations and previous results, allowing the agent to learn from past interactions and quickly access proven tool combinations for common tasks. This integrated approach enables the AI agent to act independently in complex environments, making informed decisions about tool selection and usage while adapting to new scenarios.

3. **Execution Environment:** It is a safe workspace where the AI does its tasks. Components of the Executive Environment contain **(a) The Task Manager** is like a supervisor who makes sure everything is done in the right order. **(b) The Safety Monitor** is like a security guard, checking that everything the AI wants to do is safe and allowed. **(c) The Feedback Module** collects and processes execution results.

For example, consider email management:

Input Processing: User request: "Sort my emails by priority and draft responses to urgent ones." LLM understands the task, the Planning Module creates a strategy for email classification, Memory Module recalls the user's previous email preferences

Tool Orchestration:

- Tool Executive selects email API and natural language processing tools
- API Connectors link to email server and drafting tools
- Tool Cache uses saved email templates and response pattern

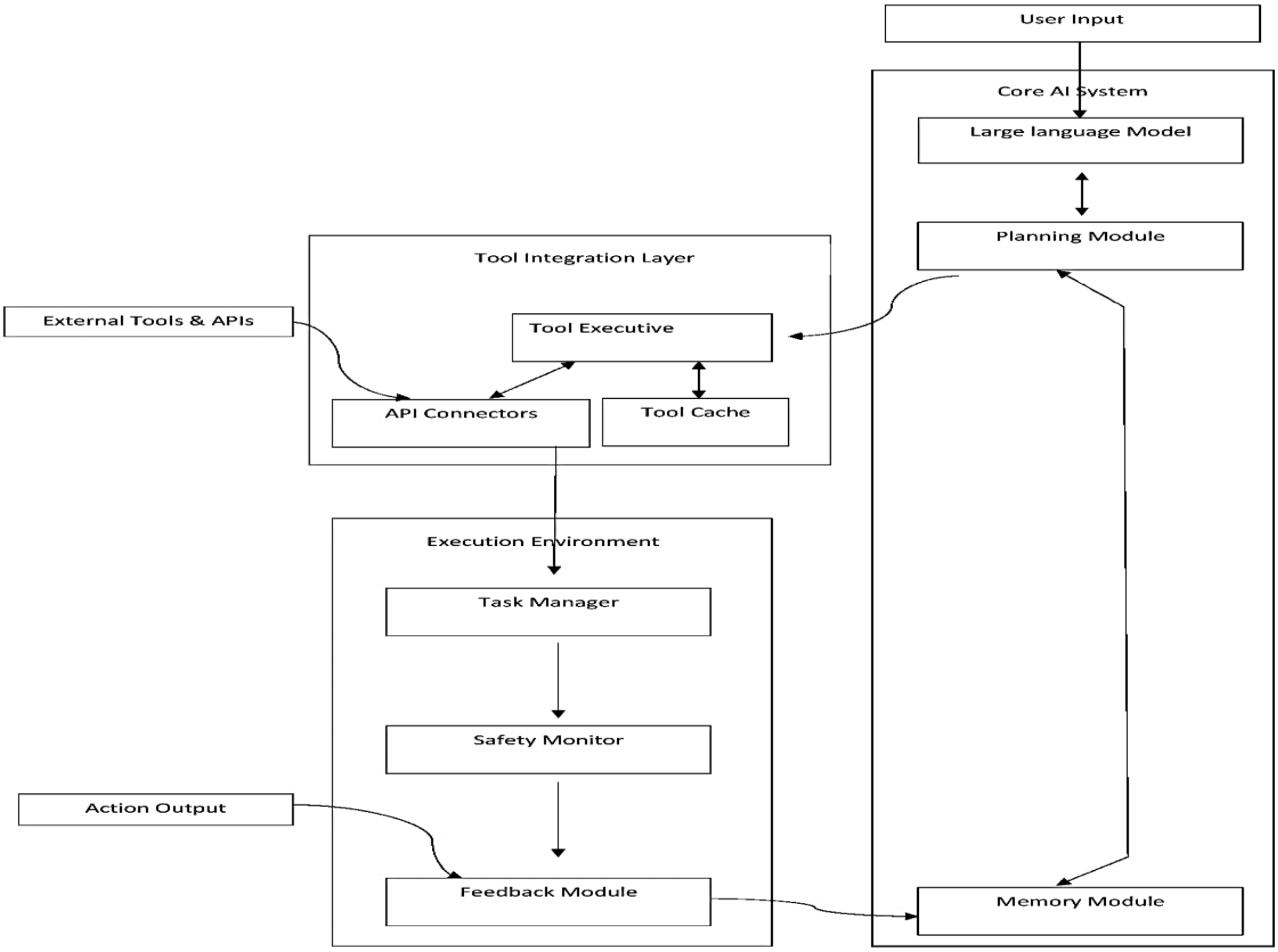
Action Execution:

- Task Manager coordinates email scanning, classification, and response drafting
- Safety Monitor ensures email access permissions and content appropriateness
- Feedback Module tracks response quality and user edits
- Memory Module records new email patterns and preferences

Continuous Learning:

- System learns which emails user typically marks as urgent
- Refines email classification based on user behavior

- Improves response drafting based on user edits



****