Introduction to Agentic LLMs

Lukas Gienapp

Kassel University

WS2025/26

What are Agents?

Observation

Agents are **autonomous** systems that ...

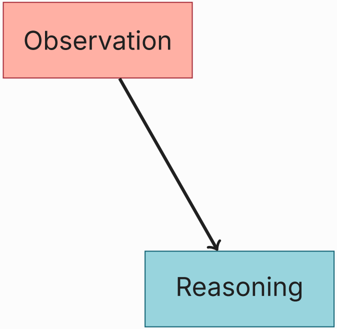
... **observe** their environment,

What are Agents?

Agents are **autonomous** systems that ...

... **observe** their environment,

... **reason** about it to make decisions,



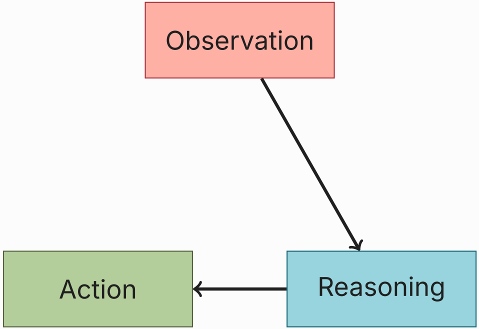
What are Agents?

Agents are **autonomous** systems that ...

... **observe** their environment,

... **reason** about it to make decisions,

... **act** to modify their environment,



What are Agents?

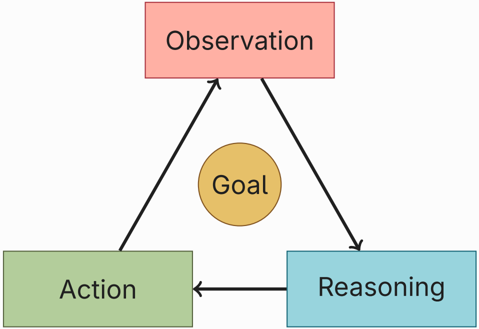
Agents are **autonomous** systems that ...

... **observe** their environment,

... **reason** about it to make decisions,

... **act** to modify their environment,

... to achieve a **goal**.



Example: Agents

Thermostat

Autonomous - requires no human input

Goal-oriented - set target temperature

Example: Agents

Thermostat

Autonomous - requires no human input

Goal-oriented - set target temperature

Observation - current and desired temperature → →

Example: Agents

Thermostat

Autonomous - requires no human input

Goal-oriented - set target temperature

Observation - current and desired temperature → **Reasoning** - decision rules, PID controller →

Example: Agents

Thermostat

Autonomous - requires no human input

Goal-oriented - set target temperature

Observation - current and desired temperature → **Reasoning** - decision rules, PID controller → **Action** - heating/cooling

Example: Agents

Thermostat

Autonomous - requires no human input

Goal-oriented - set target temperature

Observation - current and desired temperature → **Reasoning** - decision rules, PID controller → **Action** - heating/cooling

Self-Driving Car

Autonomous - requires no human input

Goal-oriented - set destination to drive to

Example: Agents

Thermostat

Autonomous - requires no human input

Goal-oriented - set target temperature

Observation - current and desired temperature → **Reasoning** - decision rules, PID controller → **Action** - heating/cooling

Self-Driving Car

Autonomous - requires no human input

Goal-oriented - set destination to drive to

Observation - sensors, cameras, GPS, ... → →

Example: Agents

Thermostat

Autonomous - requires no human input

Goal-oriented - set target temperature

Observation - current and desired temperature → **Reasoning** - decision rules, PID controller → **Action** - heating/cooling

Self-Driving Car

Autonomous - requires no human input

Goal-oriented - set destination to drive to

Observation - sensors, cameras, GPS, ... → **Reasoning** - rules of traffic, risks, route, ... →

Example: Agents

Thermostat

Autonomous - requires no human input

Goal-oriented - set target temperature

Observation - current and desired temperature → **Reasoning** - decision rules, PID controller → **Action** - heating/cooling

Self-Driving Car

Autonomous - requires no human input

Goal-oriented - set destination to drive to

Observation - sensors, cameras, GPS, … → **Reasoning** - rules of traffic, risks, route, … → **Action** - throttle, steering, brakes…

What are LLM agents?

LLM Agents
Autonomous systems that observe, reason, and take action **through text/language**.

What are LLM agents?

LLM Agents

Autonomous systems that observe, reason, and take action **through text/language**.

Isn’t any LLM an agent then?

What are LLM agents?

LLM Agents

Autonomous systems that observe, reason, and take action **through text/language**.

Isn't any LLM an agent then? Not really.

What are LLM agents?

LLM Agents

Autonomous systems that observe, reason, and take action **through text/language**.

Isn’t any LLM an agent then? Not really.

Characteristics

- **Autonomous**? Inference loops beyond zero-shot!
- **Goal-oriented**? Task ‘personas’ beyond general prompt!
- **Observation:** Capability for complex interaction
- **Reasoning**: Reasoning-tuned LLMs!
- **Action**: Tool use!

Agentic vs Non-Agentic LLMs

Example: “Write me an essay about new developments in LLM research.”

Non-agentic LLM

1. Write the essay using internal knowledge only, in one go.

Agentic vs Non-Agentic LLMs

Example: "Write me an essay about new developments in LLM research."

Non-agentic LLM

1. Write the essay using internal knowledge only, in one go.

Agentic LLM

1. **Observation**: I need to write an essay
2. **Reasoning**: Do I know enough about the topic?
3. **Action**: Search for more information using web tool.
4. **Observation**: Topic + gathered information
5. **Reasoning**: I now have all the information I need.
6. **Action**: Write the first draft of the essay.
7. **Observation**: Topic + information + draft
8. **Reasoning**: The draft can be improved.
9. **Action**: Revise the draft in writing.
10. …

Agent Autonomy

Less Autonomous

- Predetermined steps
- Hardcoded tools
- Instruction-following agents

Agent Autonomy

Less Autonomous

- Predetermined steps
- Hardcoded tools
- Instruction-following agents

Example

Research bot, that can conduct paper search on arXiv and summarize & cite the results for a given topic.

Agent Autonomy

Less Autonomous

- Predetermined steps
- Hardcoded tools
- Instruction-following agents

Example

Research bot, that can conduct paper search on arXiv and summarize & cite the results for a given topic.

More Autonomous

- Open-ended tasks
- Create its own tools
- Decision-making agents

Agent Autonomy

Less Autonomous	More Autonomous
Predetermined steps	Open-ended tasks
Hardcoded tools	Create its own tools
Instruction-following agents	Decision-making agents

Example

Research bot, that can conduct paper search on arXiv and summarize & cite the results for a given topic.

Example

Coding agent, that autonomously implements a software to spec, with full shell and filesystem access.

Benefits of Agentic LLMs

- Specialization
  - different agents for different sub-tasks instead of a single model
  - e.g., the research bot can have a *Writer*, a *Search*, a *Critic*, ...

Benefits of Agentic LLMs

- Specialization
  - different agents for different sub-tasks instead of a single model
  - e.g., the research bot can have a *Writer*, a *Search*, a *Critic*, ...
- Parallelization
  - tools can be used in parallel by many independent sub-agents
  - e.g., the research bot can conduct many different web searches in parallel

Benefits of Agentic LLMs

- Specialization
  - different agents for different sub-tasks instead of a single model
  - e.g., the research bot can have a *Writer*, a *Search*, a *Critic*, …
- Parallelization
  - tools can be used in parallel by many independent sub-agents
  - e.g., the research bot can conduct many different web searches in parallel
- Modularity
  - independent parts of the workflow can be changed without retraining, new tools can be made available dynamically
  - e.g., we can add a new search endpoint to the existing research bot

Benefits of Agentic LLMs

- Specialization
    - different agents for different sub-tasks instead of a single model
    - e.g., the research bot can have a *Writer*, a *Search*, a *Critic*, ...
- Parallelization
    - tools can be used in parallel by many independent sub-agents
    - e.g., the research bot can conduct many different web searches in parallel
- Modularity
    - independent parts of the workflow can be changed without retraining, new tools can be made available dynamically
    - e.g., we can add a new search endpoint to the existing research bot
- Adaptation
    - the LLM can self-adjust its behaviour to the specific task at hand
    - e.g., the research bot can decide which web sources to use (arXiv, Newspapers, Wiki, ...)

Challenges in Building Agentic Systems

- How do we break down the task so that LLM agents can solve it? → **Design Patterns**

Challenges in Building Agentic Systems

- How do we break down the task so that LLM agents can solve it? → **Design Patterns**
- How do we instruct LLM agents to follow the their task(s)? → **Prompt Engineering**

Challenges in Building Agentic Systems

- How do we break down the task so that LLM agents can solve it? → **Design Patterns**
- How do we instruct LLM agents to follow the their task(s)? → **Prompt Engineering**
- How do we describe the world to LLM agents? → **Model Context Protocol**

Challenges in Building Agentic Systems

- How do we break down the task so that LLM agents can solve it? → **Design Patterns**
- How do we instruct LLM agents to follow the their task(s)? → **Prompt Engineering**
- How do we describe the world to LLM agents? → **Model Context Protocol**
- How do we give LLM agents access to tools to interact with the world? → **Tool Usage**

Challenges in Building Agentic Systems

- How do we break down the task so that LLM agents can solve it? → **Design Patterns**
- How do we instruct LLM agents to follow the their task(s)? → **Prompt Engineering**
- How do we describe the world to LLM agents? → **Model Context Protocol**
- How do we give LLM agents access to tools to interact with the world? → **Tool Usage**
- How do we improve agent effectiveness? → **Memory & Multi-agent systems**

Challenges in Building Agentic Systems

- How do we break down the task so that LLM agents can solve it? → **Design Patterns**
- How do we instruct LLM agents to follow the their task(s)? → **Prompt Engineering**
- How do we describe the world to LLM agents? → **Model Context Protocol**
- How do we give LLM agents access to tools to interact with the world? → **Tool Usage**
- How do we improve agent effectiveness? → **Memory & Multi-agent systems**

Goal of the course: enable you to address these challenges in a group project.

Course Organization

- First part: Lectures (Session 1 – 3)
  - Foundational knowledge for understanding LLMs
  - Architectures, training, prompt engineering patterns, fine-tuning, …
  - **Goal**: Learn basic LLM concepts and methods
- Second part: Exercises (Session 4 – 10)
  - Engineering concepts and development patterns for agents
  - Python tools, model APIs, …
  - **Goal**: Learn to apply concepts in practice and prepare you for the group projects
- Third part: Group Work (Session 11 - 14)
  - Conceptualize, implement, and evaluate your own agent
  - Give a short (15min) presentation and demo at the end of the semester
  - Hand in a research report (6 pages) about your findings

Syllabus

Week	Date	Topic	Type	Deliverables
1	16.10.2025	Introduction	Lecture	
2	23.10.2025	Training & Tuning LLMs	Lecture	Agent Ideas
3	30.10.2025	Prompting & Reasoning	Lecture	Group Formation
4	06.11.2025	Agents & Tools	Lab	
5	13.11.2025	Multi-Agent Patterns	Lab	Group Topic
6	20.11.2025	Agent Memory	Lab	
7	27.11.2025	Model Context Protocol (I)	Lab	
8	04.12.2025	Model Context Protocol (II)	Lab	
9	11.12.2025	Evaluation	Lab	
10	18.12.2025	Scientific Writing	Lecture	
		Winter Break		
11	22.01.2026		Consultation	
12	29.01.2026		Consultation	
13	05.02.2026	Presentations	Presentations	Project Presentation
14	12.02.2026		Consultation	
	? (TBD.)			Project Report

Homework: Agent Ideas

Come up with your own agent idea!

- What is the goal?
- What agent roles are needed?
- What tools could be needed?
- What would a typical workflow look like?

Shortly present these questions on a single slide start of next session!