Analytics Vidhya

RAG vs. Agents vs. Agentic RAG

Instructor

Dipanjan Sarkar

Head of Community & Principal AI Scientist at Analytics Vidhya

Google Developer Expert - ML & Cloud Champion Innovator

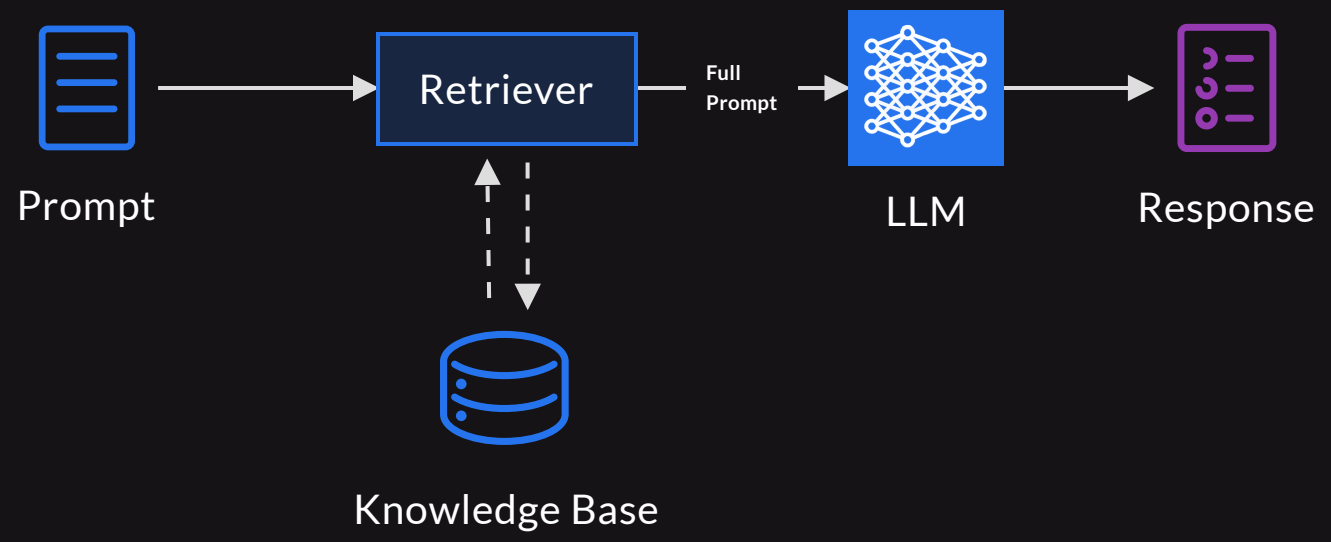
Published Author

Outline

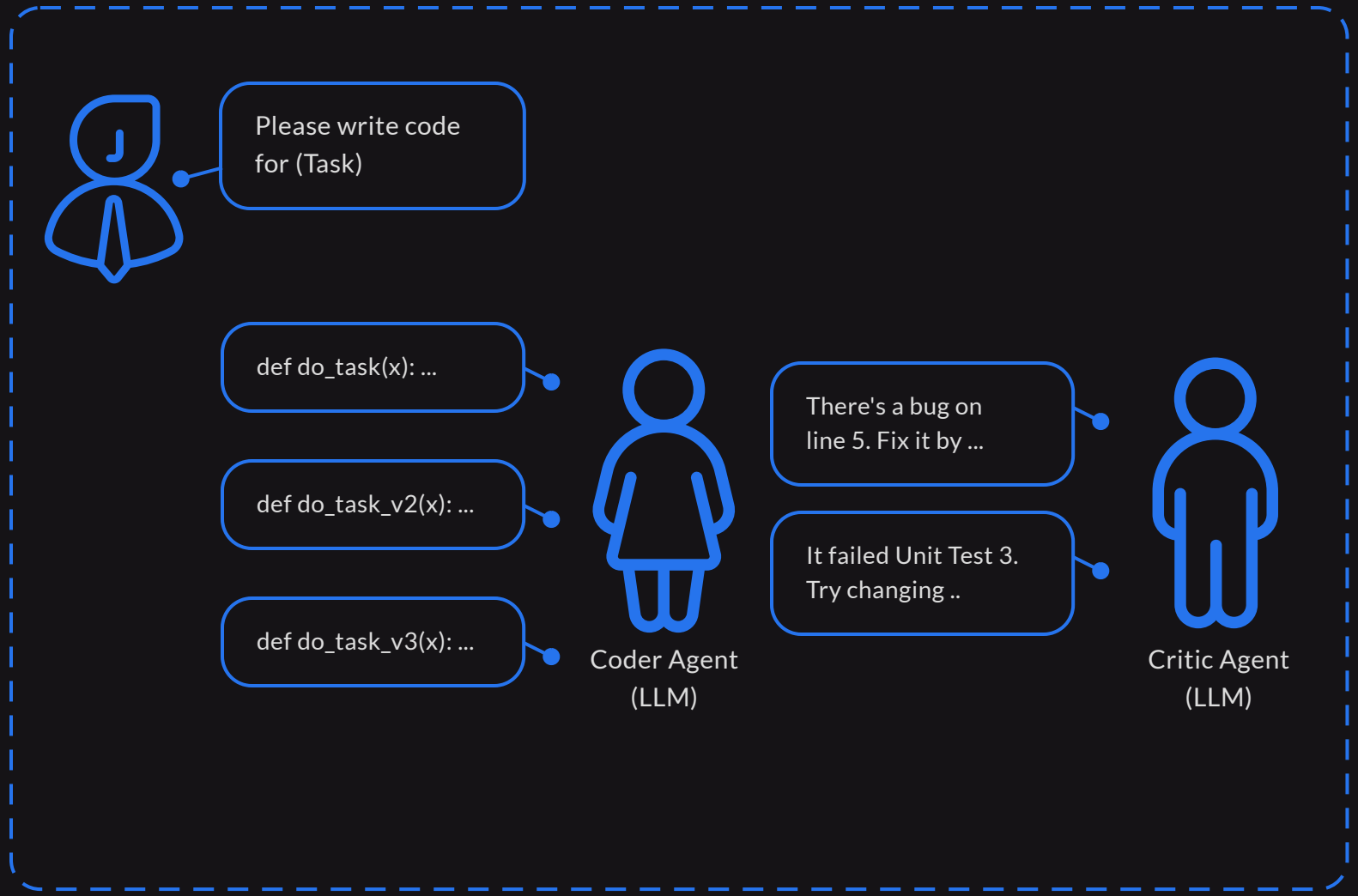
- RAG Systems
- AI Agents
- Agentic RAG
- RAG vs. Agents vs. Agentic RAG

Retrieval Augmented Generation (RAG)

- RAG connects an external knowledge base to augment the existing knowledge of a LLM
- RAG leverages a vector database to first retrieve relevant context for a query and makes the LLM use this context to answer queries
- RAG is beneficial in situations requiring the latest information or answers involving custom enterprise data on which the LLM was never trained.

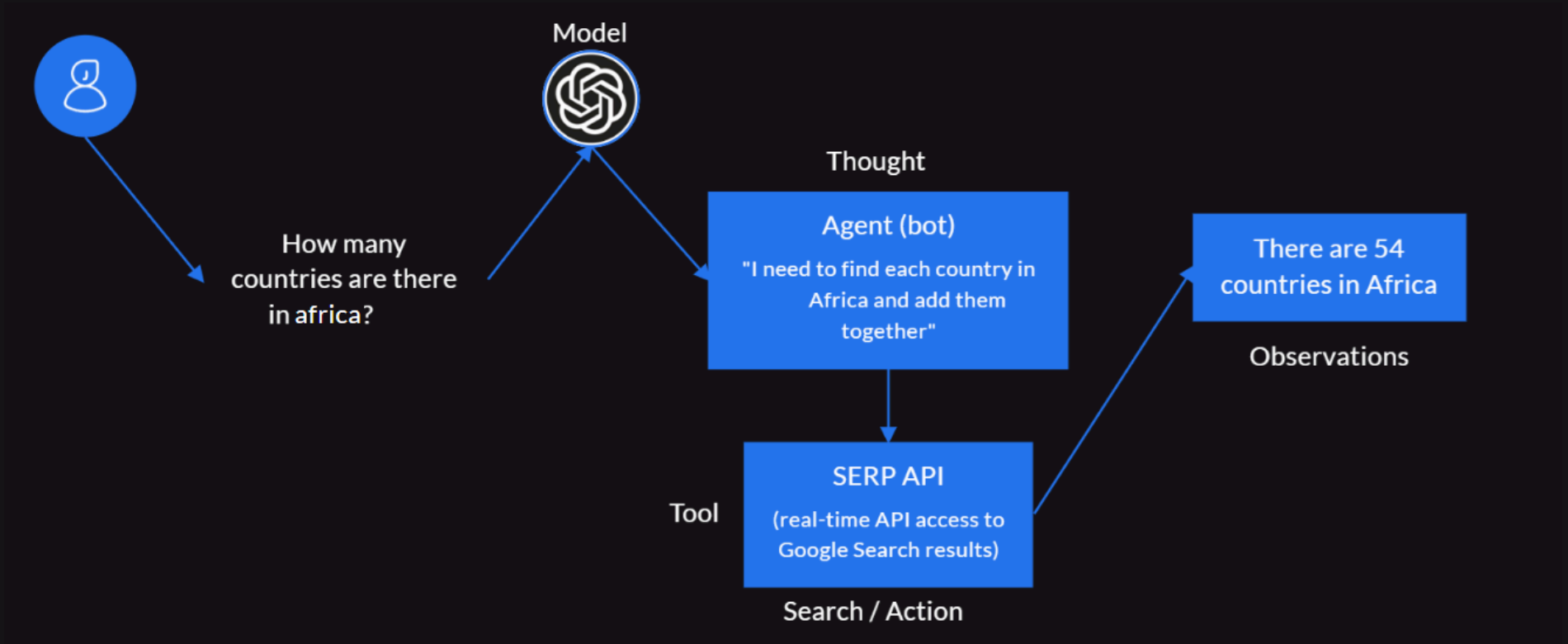


AI Agents



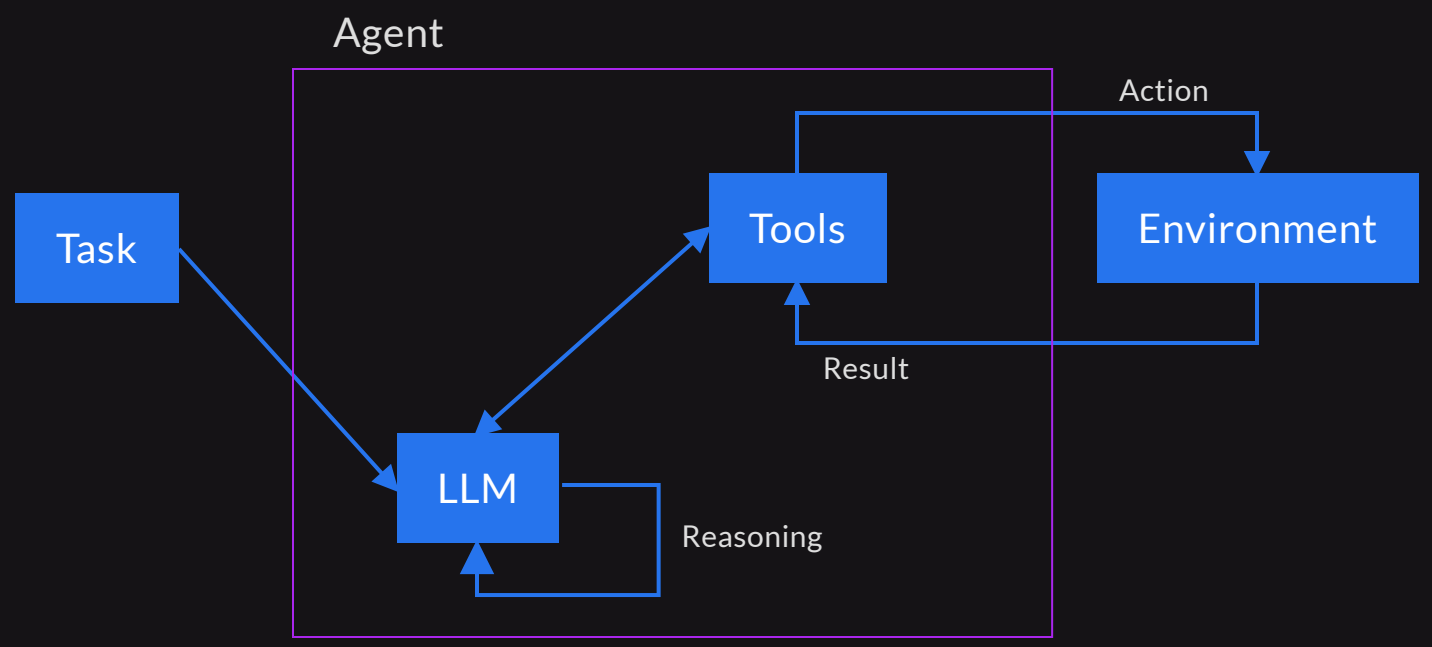
- AI Agents also known as Agentic AI Systems or autonomous AI, represents a fundamental shift in artificial intelligence
- Designed to autonomously understand and manage complex workflows with minimal human intervention
- Functions quite similarly to humans
- They can grasp nuanced contexts, set and pursue goals, reason through tasks, and adapt their actions based on changing conditions

AI Agents



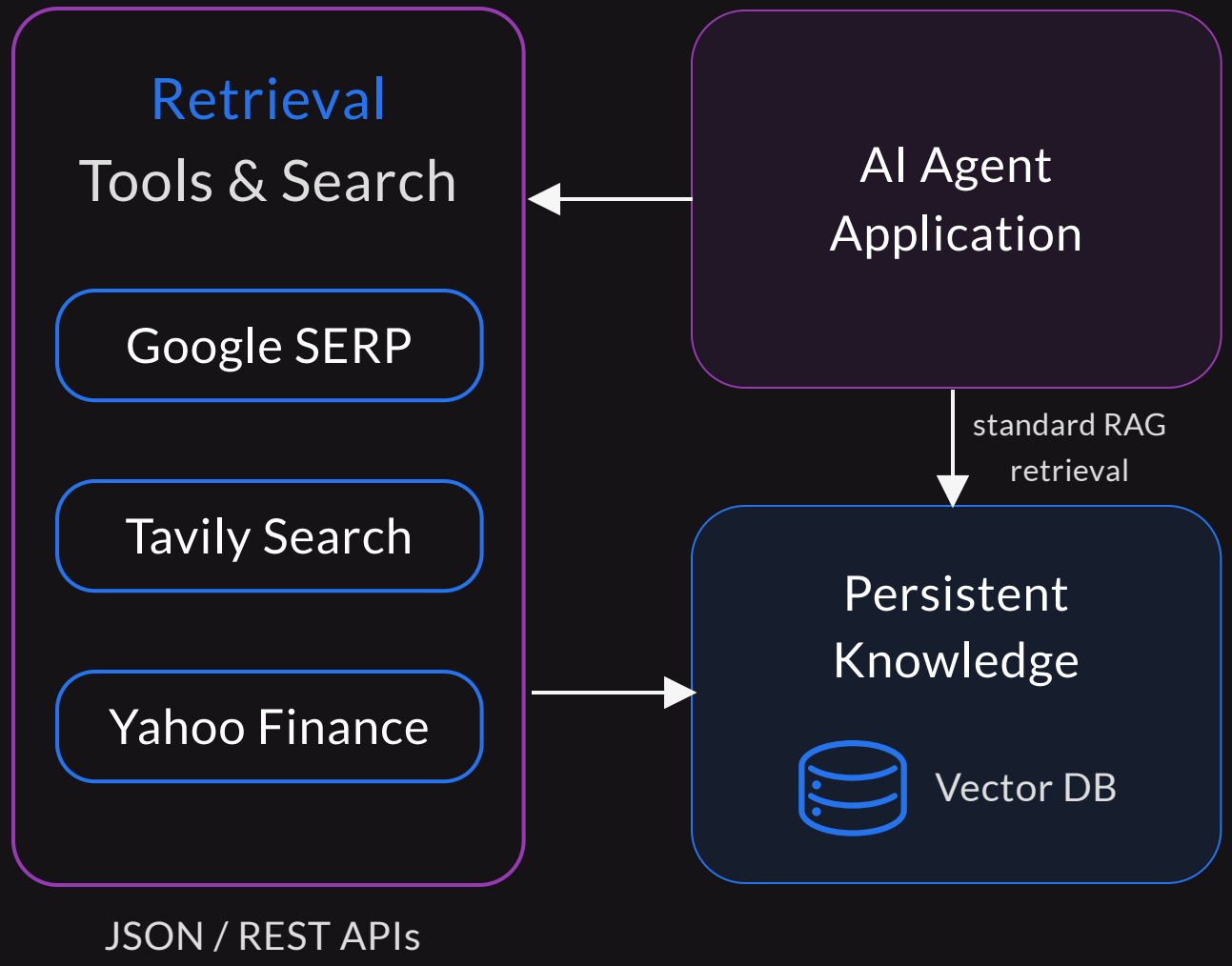
- Agents are systems that use an LLM as a reasoning engine to determine which actions to take and what the inputs to those actions should be.
- The results of those actions can then be fed back into the agent and it determines whether more actions are needed, or whether it is okay to stop.

AI Agents



- Technically an AI Agent is a combination of LLMs, prompts and tools
- Flow starts with a user query or task
- The LLM usually reasons about what to do next in the cycle like Chain of Thought
- The LLM might call one or more tools to get relevant information from external sources
- The above steps might happen multiple times till the LLM has enough information to give a response

Agentic RAG



- Agentic RAG is a combination of AI Agents and RAG Systems
- Leverages retrieval and search tools to access live real-time data besides the vector database
- Can be extended to add in multiple levels of complex flows to validate retrieval, response generation and check for hallucinations
- Examples include Agentic Corrective RAG, Self-Reflective RAG and more

RAG vs. Agents vs. Agentic RAG

Feature	RAG	**Agents**	**Agentic RAG**
Key Role	Combines LLMs with external data retrieval to generate responses	Combines LLMs, tools, and instructions for autonomous task management	Enhances RAG by using agents for intelligent retrieval, response generation, grading, critiquing, and more
Real-Time Data Retrieval	Not possible in native RAG	Not a core feature but possible with tools	Designed for real-time data retrieval and integration
Integration with Retrieval Systems	Tied to static retrieval from pre-defined vector databases	Not specifically tied to retrieval, can work with search tools	Deeply integrated with diverse retrieval systems, agents control the process
Context-Awareness	Limited by the static vector database, no advanced or real-time context-awareness	Moderate, based on the agent's logic and tools	High, agents adapt to user query and retrieve context, including real-time data

RAG vs. Agents vs. Agentic RAG

Feature	RAG	Agents	Agentic RAG
Task Complexity	Handles simple query-based tasks but lacks advanced decision-making	Handles complex, multi-step tasks with multiple agents and tools working in coordination if needed	Handles complex, multi-step tasks with multiple tools and agents as needed for retrieval, reasoning, answering, grading, and more
Decision-Making	Limited, no autonomous decision-making involved	Autonomous decisions based on environment and task, not tied to data retrieval	Agents autonomously decide what data to retrieve, how to retrieve, grade, reason, reflect, and generate responses
Multi-Step Reasoning	*Limited to single-step queries and responses*	Capable of multi-step reasoning if designed for complex tasks	Excels at multi-step reasoning, especially after retrieval with grading, hallucination, and response evaluation

Thank You