

# Recap of Conversational Systems & Memory

Dipankar Sarkar

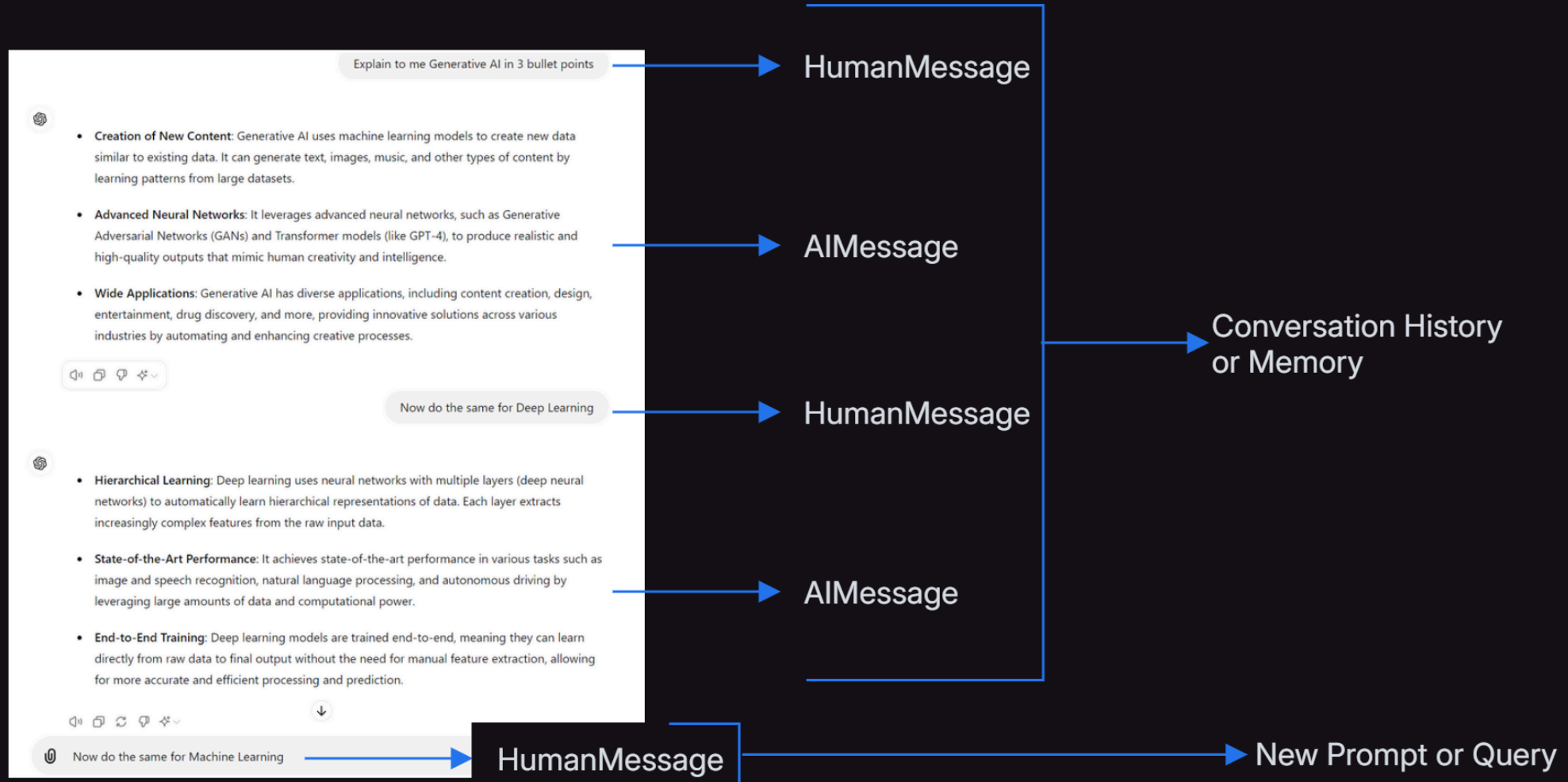
Head of Community & Principal AI Scientist at Analytics Vidhya

Google Developer Expert - ML & Cloud Champion Innovator

Published Author



# What is a Conversation?



# What is a Conversation?

## With Conversational Memory

I am interested in integrating LLMs with external knowledge

LLMs are great at generating human-like text. Yet, integrating external knowledge can enhance their capabilities even more.

What are the different possible methods for doing this?

You could use pre-existing knowledge graphs, allow LLMs access to tools like APIs, or retrieval augmentation with vector DBs!

..... Conversation History .....

Interesting! What was it I wanted to know about?

You were interested in integrating LLMs with external knowledge.

## Without Conversational Memory

(No conversation history is stored)

..... Conversation History .....

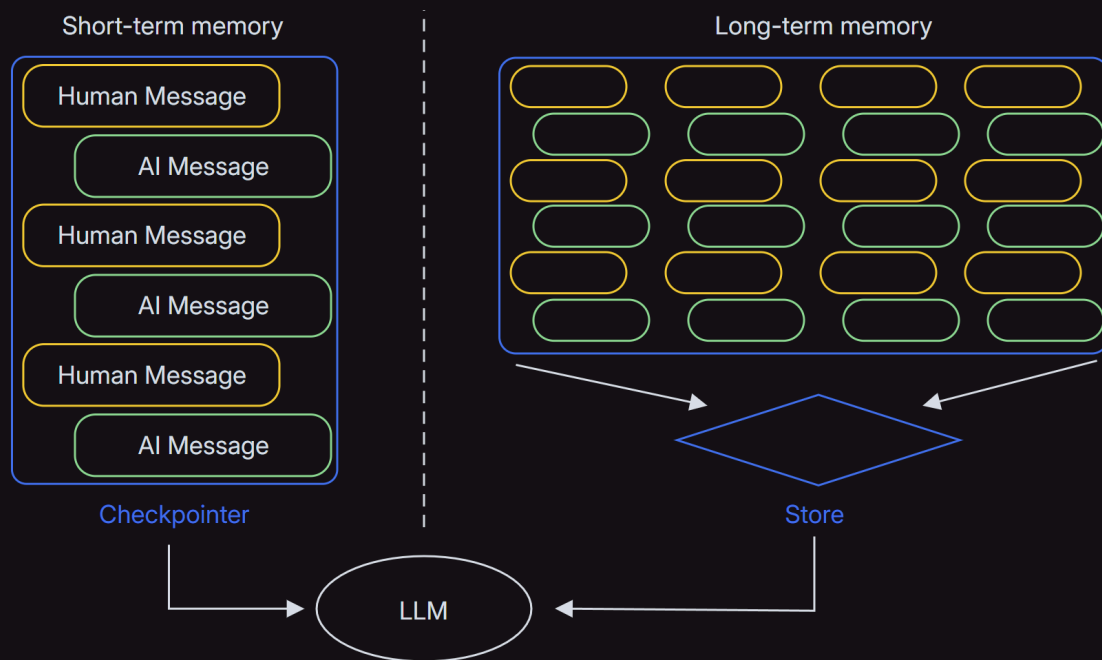
Interesting! What was it I wanted to know about?

Sorry, I have no idea what you're talking about!

# Popular LangChain Built-in Memory Constructs

Memory Type	Description
<a href="#"><u>ConversationBufferMemory</u></a>	This memory allows for storing messages of conversation history and then extracts the messages in a variable
<a href="#"><u>ConversationBufferWindowMemory</u></a>	Stores only the the last K messages of conversation history. This can be useful for keeping a sliding window of the most recent interactions with the LLM
<a href="#"><u>ConversationSummaryMemory</u></a>	Summarizes the conversation history and stores the current summary in memory instead of actual conversation messages
<a href="#"><u>VectorStoreRetrieverMemory</u></a>	Stores conversation history in a vector database and queries the top-K most relevant docs every time a new message or query is sent to the LLM. Doesn't care about the order of conversations
<a href="#"><u>ChatMessageHistory</u></a>	Simple but highly configurable wrapper to store messages in an in-memory list. Easy to manage memory for multiple users and sessions.
<a href="#"><u>SQLChatMessageHistory</u></a>	Same as ChatMessageHistory but persists conversations in a SQL database. Very useful when managing multiple long conversations

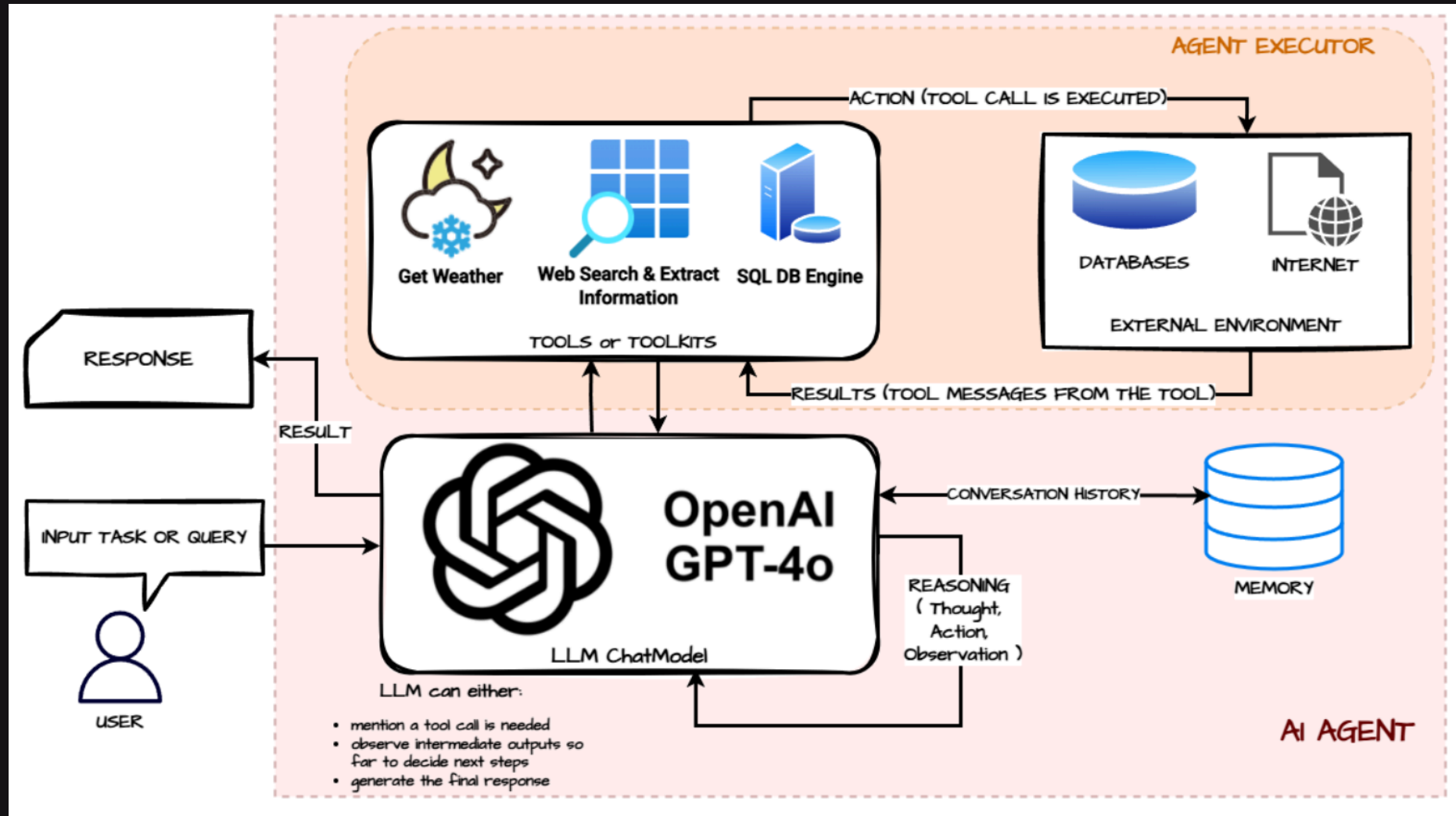
# Key Memory Constructs in LangGraph



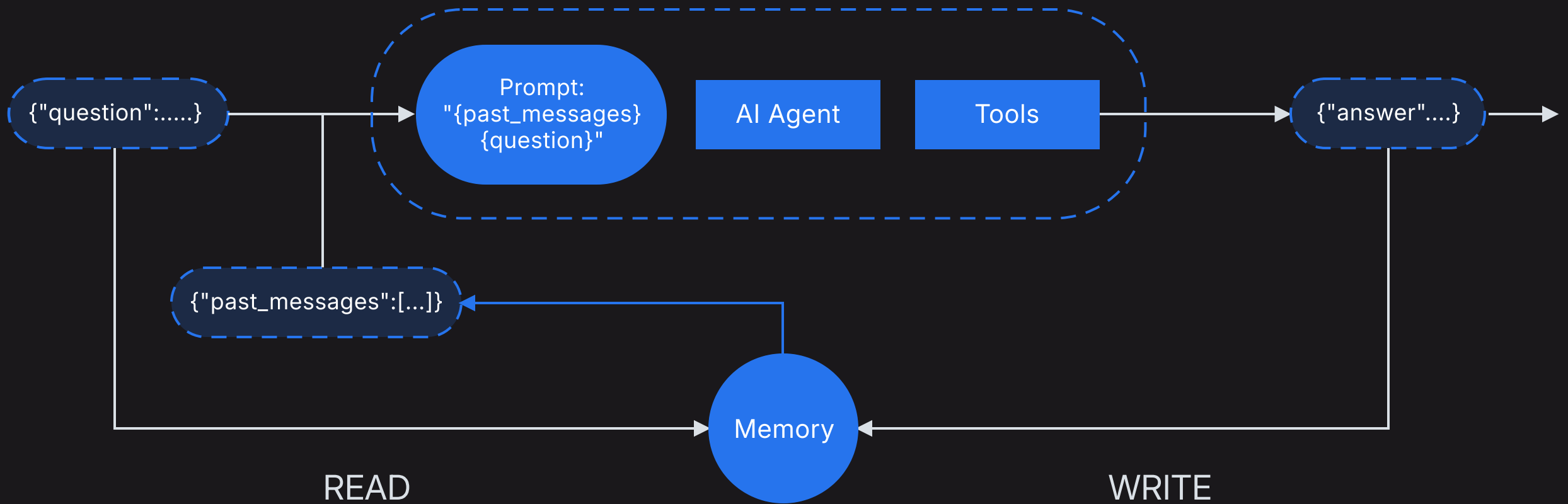
Method	Description
LangGraph Checkpointer	Manages the persistence of an agent's state per user session (thread) interaction by saving checkpoints of the graph's state at each step. This enables session memory, error recovery, human-in-the-loop interactions, and the ability to resume conversations from specific points. Can be transient (in-memory) or persistent (on-disk).
LangGraph Memory Store	Provides a system for managing long-term (cross-thread) memory within conversational agents. Short-term memory is managed as part of the agent's state, while long-term memory is stored as JSON documents in a store, organized under custom namespaces for hierarchical organization. Useful for personalized memories.



# Conversational AI Agent



# Conversational AI Agent



**Thanks**