Analytics Vidhya

Memory Management for Long Conversations

Dipanjan Sarkar

Head of Community & Principal AI Scientist at Analytics Vidhya

Google Developer Expert - ML & Cloud Champion Innovator

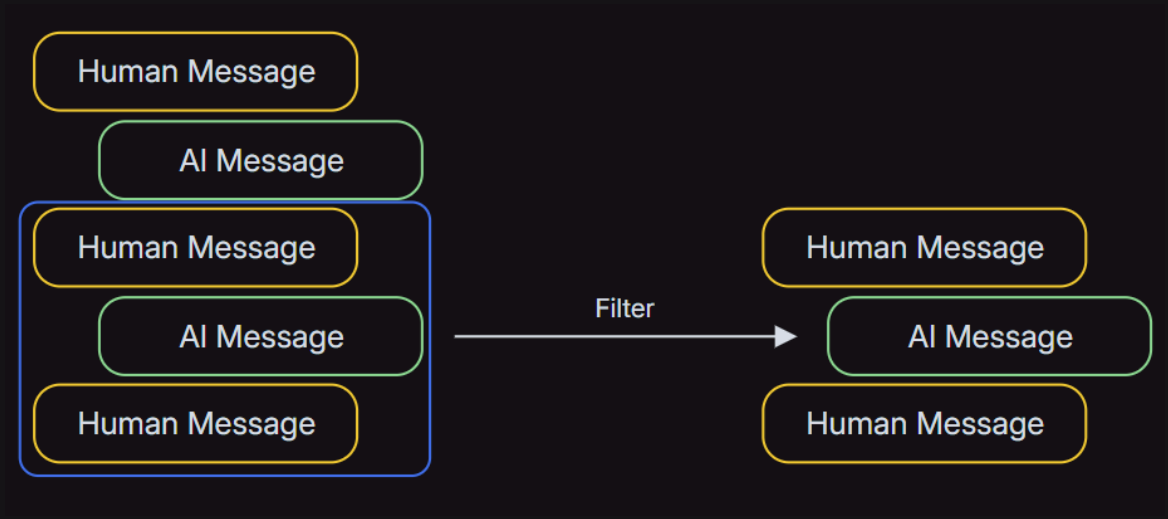
Published Author

Issues with Long Conversations in Agentic Systems

Managing memory in agents during long conversations presents challenges due to the limited context window sizes of Large Language Models (LLMs).

Challenges

- Context Window Limitations
  - LLMs have fixed context windows (e.g., 4K tokens), restricting the amount of conversation history they can process.
  - Even long context LLMs have limits
- Performance Degradation
  - Exceeding the context window can lead to errors or diminished model performance, as the LLM may become "distracted" by outdated or irrelevant information.
- Cost
  - More conversation history means more tokens which means costs increase as commercial LLMs charge you per token or open LLMs will take more computing power and time.

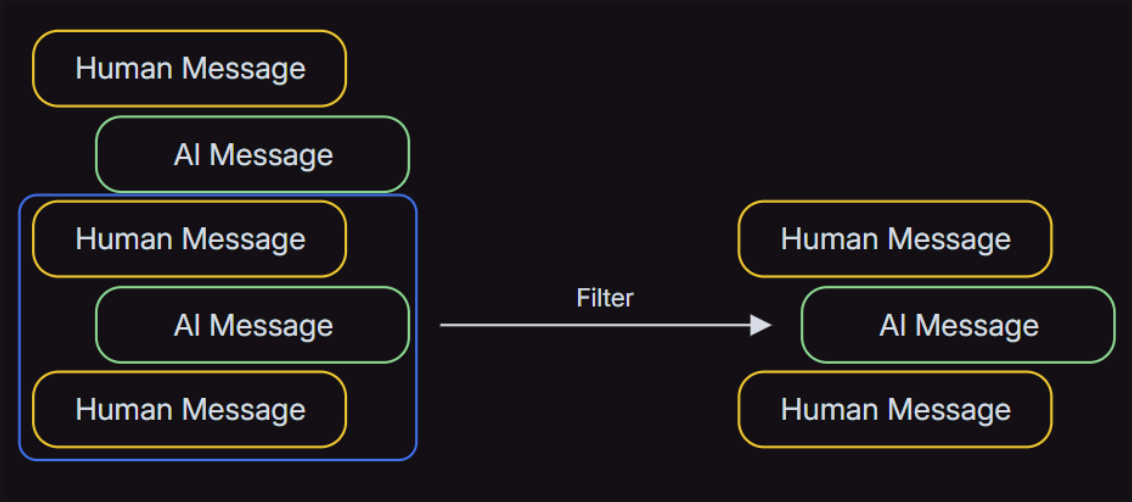


Managing Long Conversations in LangGraph

Strategies in LangGraph

Editing Message Lists

- Manually removing or forgetting stale information from the agent state message list to prevent context window overflow.
  - RemoveMessage can remove specific messages in the history
  - trim_messages can remove messages from the history based on number of tokens
- *We will be using some of these in our hands-on projects in this module*

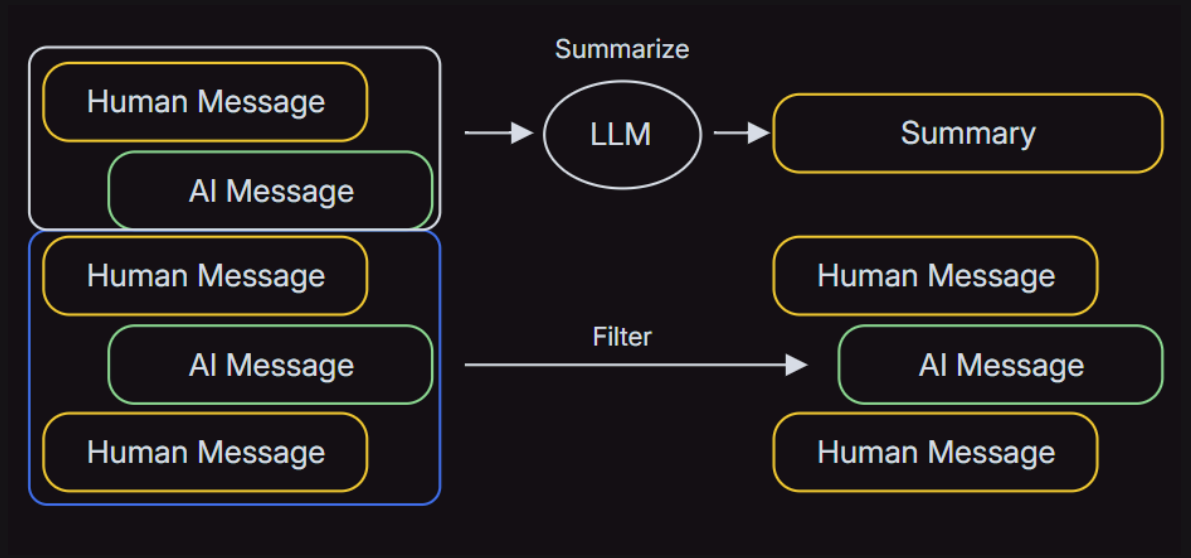


Managing Long Conversations in LangGraph

Strategies in LangGraph

Summarizing Past Conversations

- Generating concise summaries of previous interactions to retain essential context without exceeding token limits.
- Custom node function can be created that uses an LLM to take the existing agent messages, summarize them and delete the original messages



Thanks