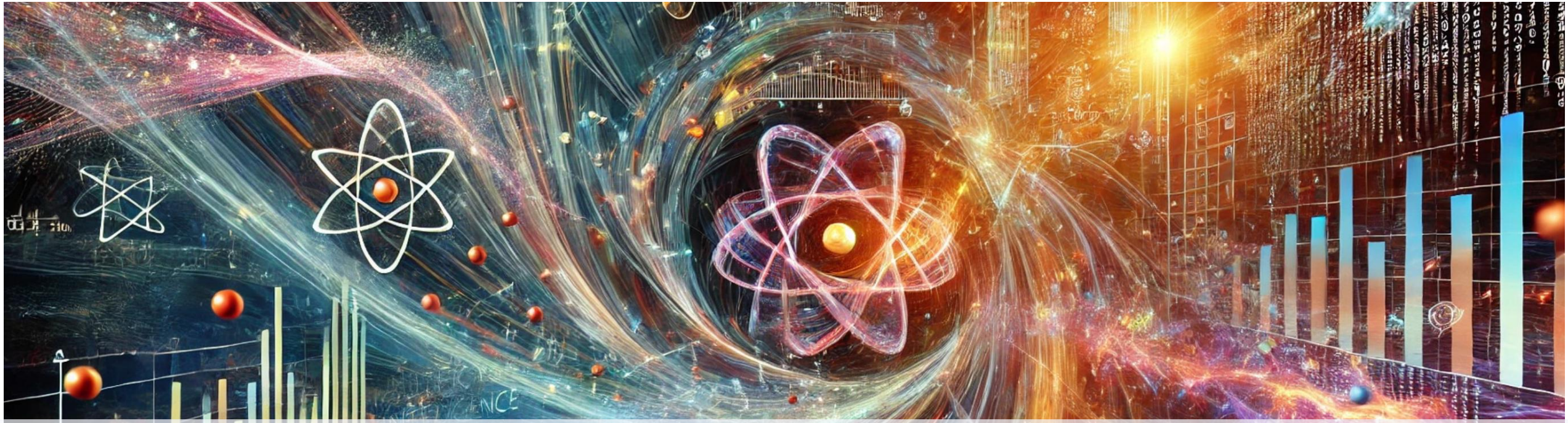
Computing Center, IHEP, CAS
National HEP Data Center

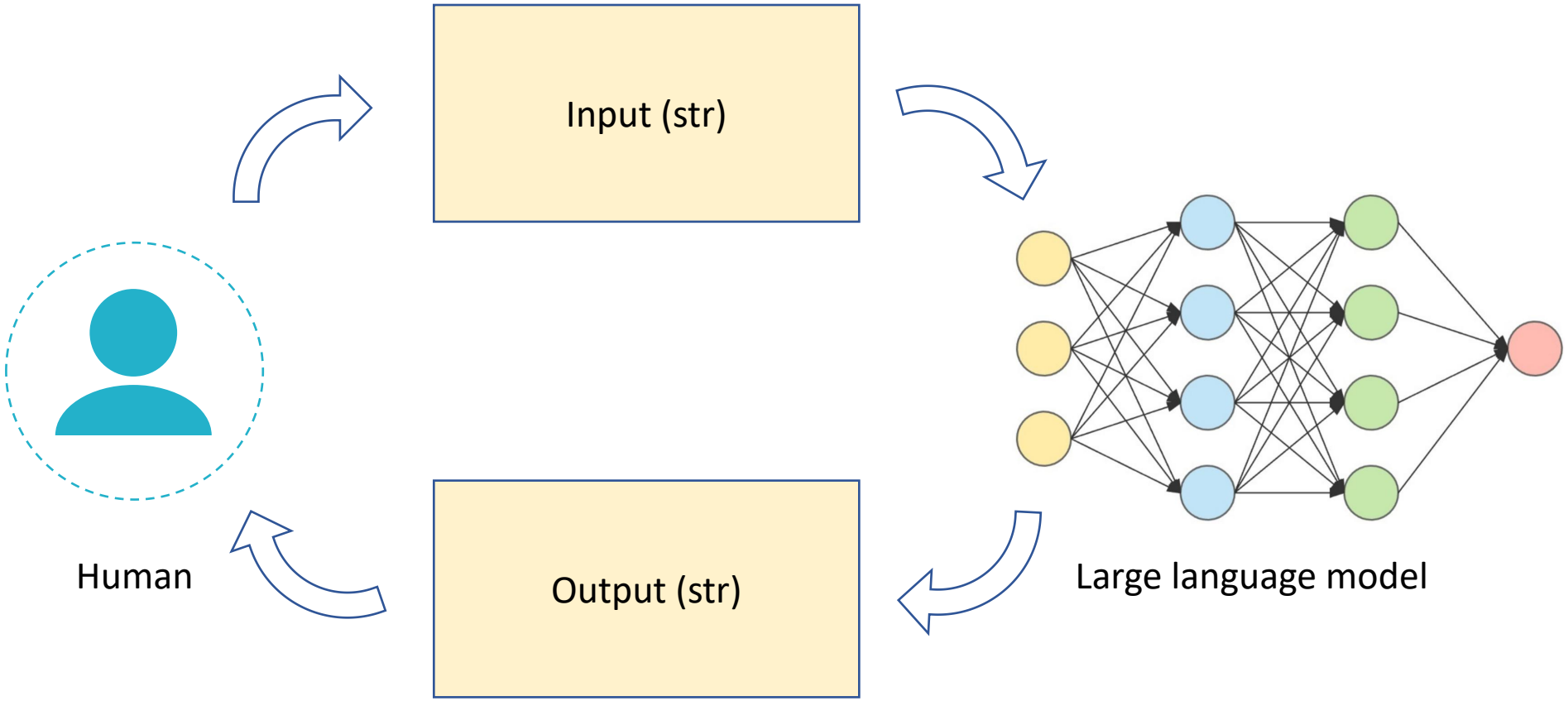


An Introduction to Context Engineering

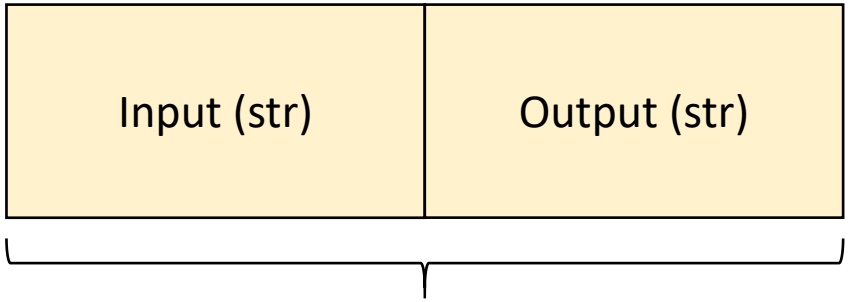
Bolun Zhang
On behalf of Dr.Sai working group
2026.04.14 in IHEP, Beijing

LLM: Text Generator

- LLM is an ”intelligent” text generator, it only understands and responds in text.
- its capabilities go far beyond simple text generation.



Limited Context Window



= 128K tokens | 1:

~ 96K words | 1:0.75

~ 128K chinese characters | 1:1

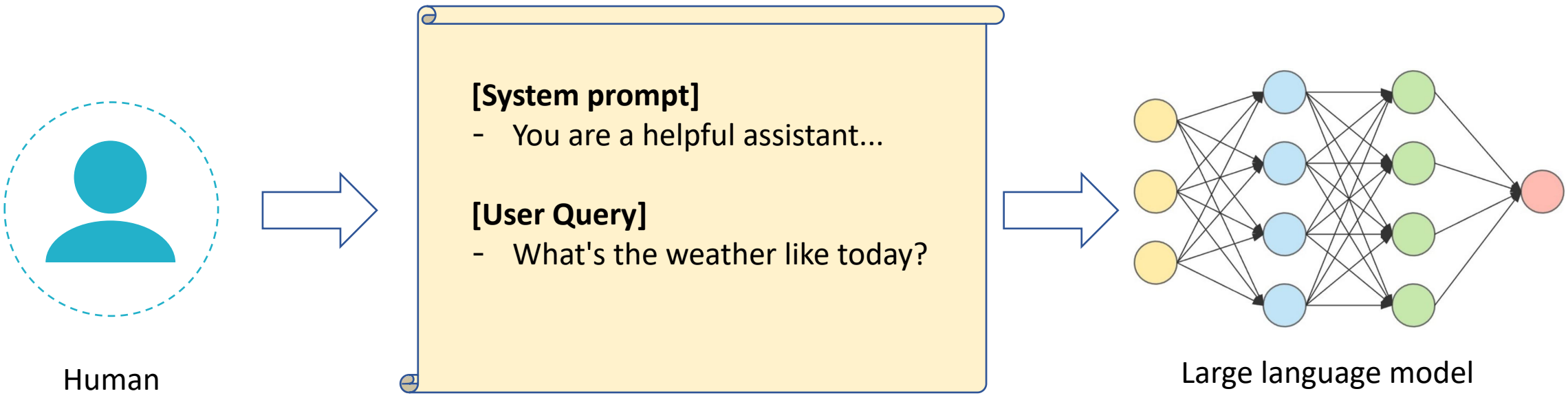
Category	Context Window	Example Models
Ultra-Long Context	2M - 10M+ tokens	Llama 4 Scout (10M), xAI Grok 4 (2M), Gemini 2.5 Pro (1M)
Long Context	1M tokens	Gemini 2.5 Flash, GPT-4.1 / 4.1 Mini, Llama 4 Maverick
Mid-Range Context	200K - 400K tokens	OpenAI o3 (200K), Claude 3.5 Sonnet (200K), Kimi (200K)
Standard Context	128K tokens	DeepSeek V3/R1, Mistral Large 3, Qwen3-235B, GPT-4o, Llama 3.1 70B
Small LLMs	8K tokens	Qwen-8B, Llama2-7B
Very Small LLMs	< 8K tokens	TinyLLaMA 1.1B (2K), GPT-2 (1024)

more tokens ≠ better results! - Even within the window, performance gains diminish as context fills up.

Context for Base Model (Single Turn)

The required information for **a single call**

- System prompt - optional?
- User query - supports techniques like Chain-of-Thought and Few-Shot prompting



Context for Base Model (Multi-Turn)

The required information for **multiple rounds of dialogue**

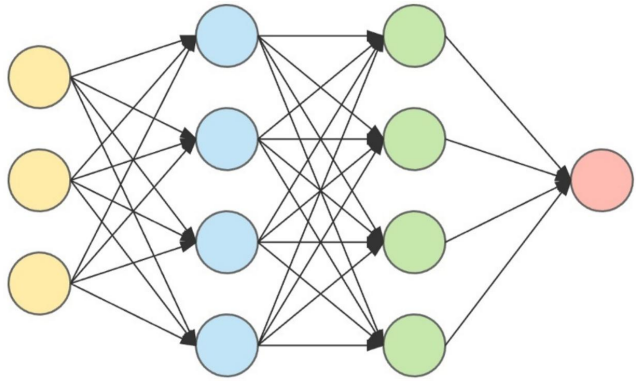
- System prompt
- Chat history - dynamically extending!
- User query

Human

[System prompt]
- You are a helpful assistant...

[Chat history]
- What's the weather like today?
- The weather is sunny today.

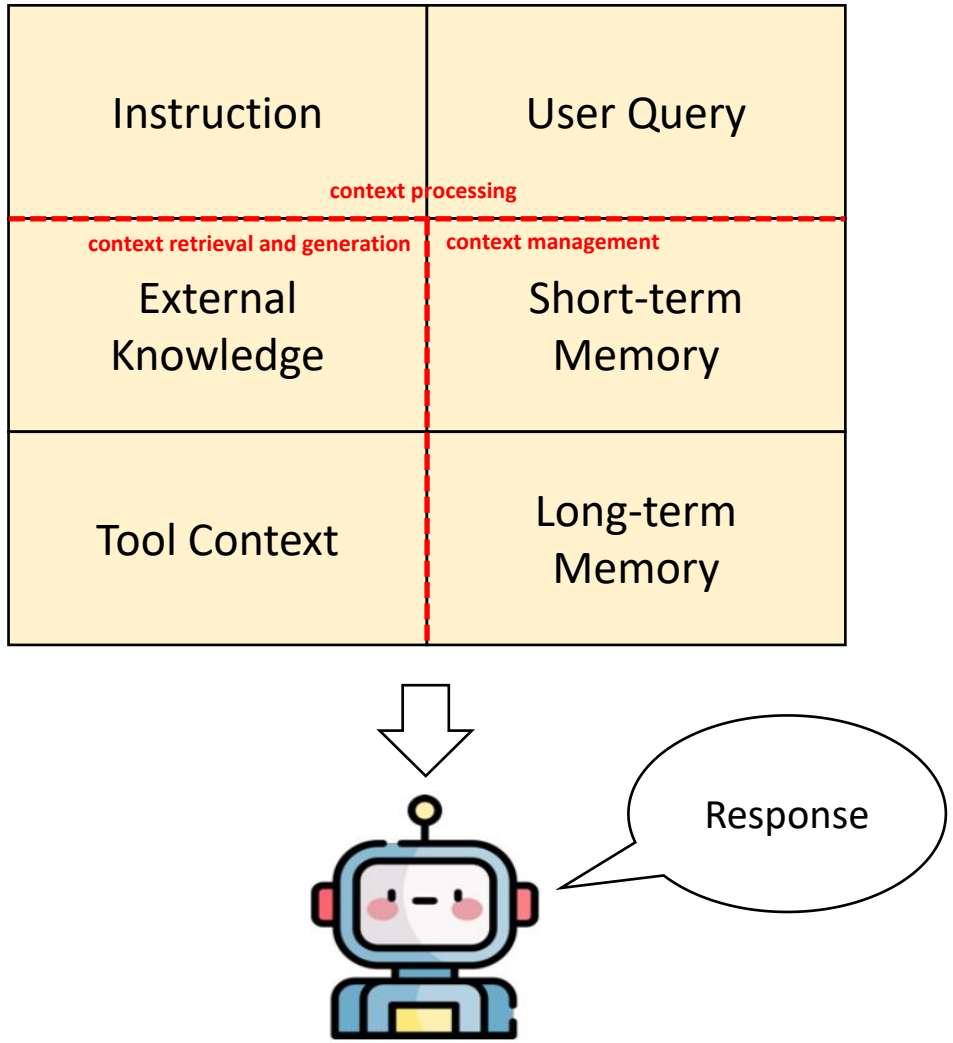
[User Query]
- Should I wear sunscreen?



Large language model

Context for Agent

1. **System & Instruction**
   - system prompt (incl. few-shot, CoT instructions)
   - output format constraints
   - agent state, skills
2. **User Query**
   - current user input
3. **Short-term Memory**
   - full chat history of current session
4. **Long-term Memory**
   - user profile, preferences
   - cross-session history
5. **External Knowledge (RAG)**
   - retrieved document chunks from knowledge base
6. **Tool Context**
   - tool definitions
   - intermediate results from tool calls



Why Context Engineering?

1. Limited context window & diminishing returns: Input length has a hard limit, performance gains drop off as you fill it up.

- Chunking, context compression, pruning, RAG

2. Lost in the middle: Information in the middle of a long context is often ignored by the model.

- Put critical information at the beginning and the end, not in the middle.

3. Hallucinations: The model generates false or ungrounded information.

- RAG, in-context learning (ICL), prompt calibration, grounding with tools

4. Knowledge timeliness (outdated info): The model cannot know events after its training cutoff.

- RAG with live data sources, tool use (web search / API), fine-tuning on fresh data

5. Incomplete background information: The system lacks user-specific or task-specific context.

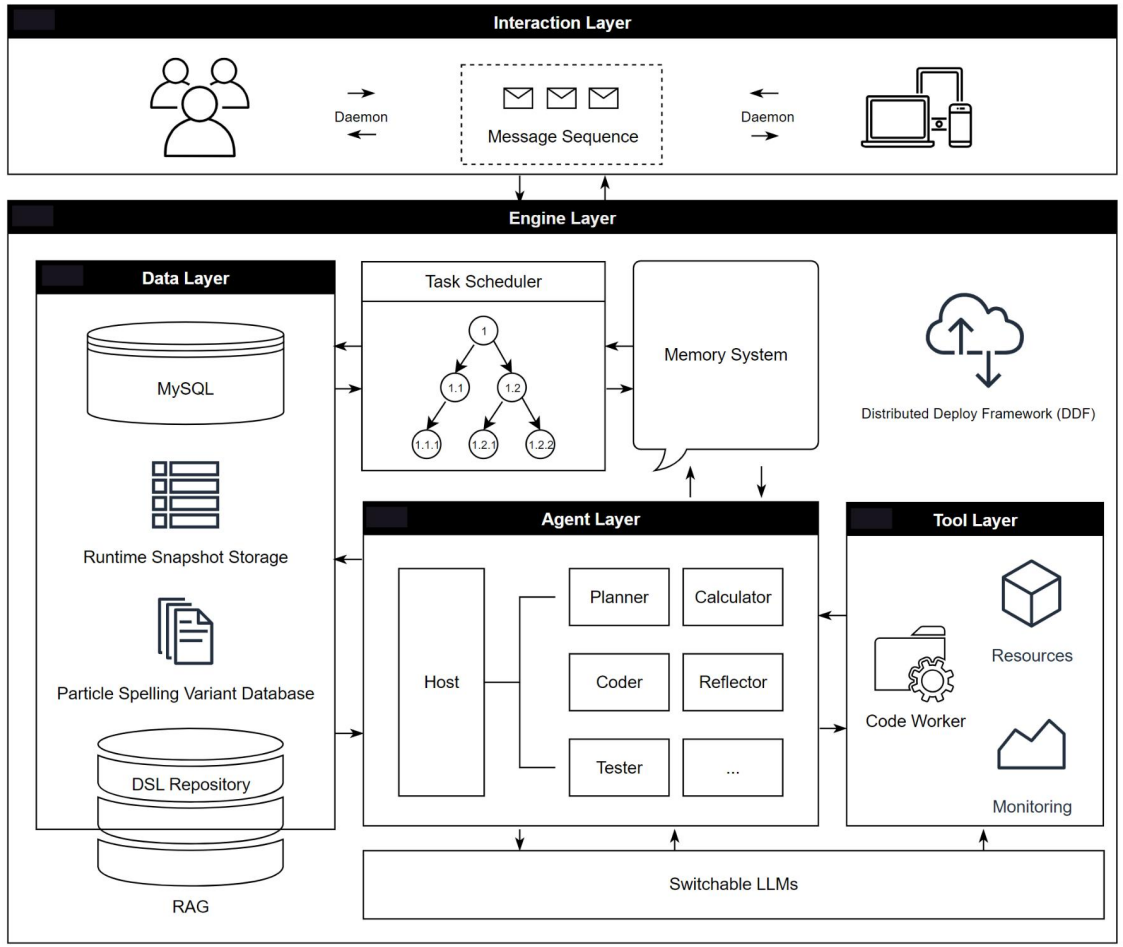
- Long-term memory (user profile), short-term memory (session history), tool use to fetch missing context

6. Performance–cost trade-off: More tokens ≠ better results. Longer context or larger models increase latency and cost.

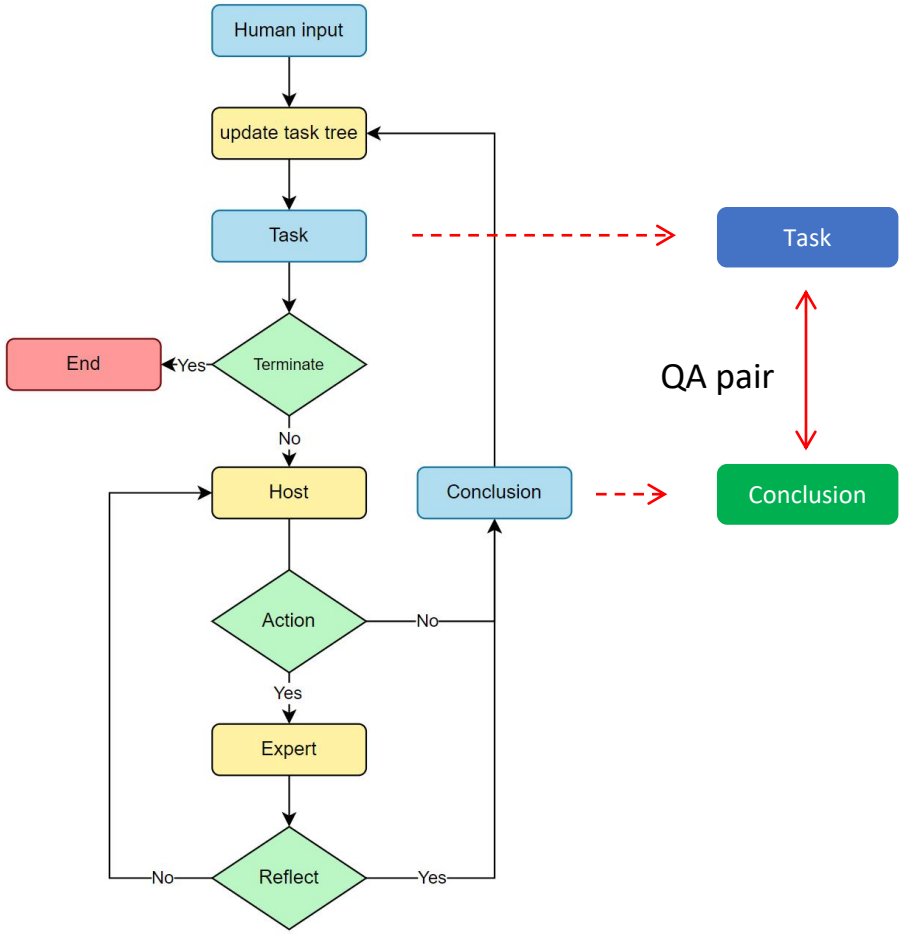
- Compress, prune, or use RAG instead of blindly expanding context.

Context Engineering in MAS

Taking Dr. Sai-BESIII as an example, it employs an orchestrator-worker architecture with five workers in the MAS (Multi-Agent System).



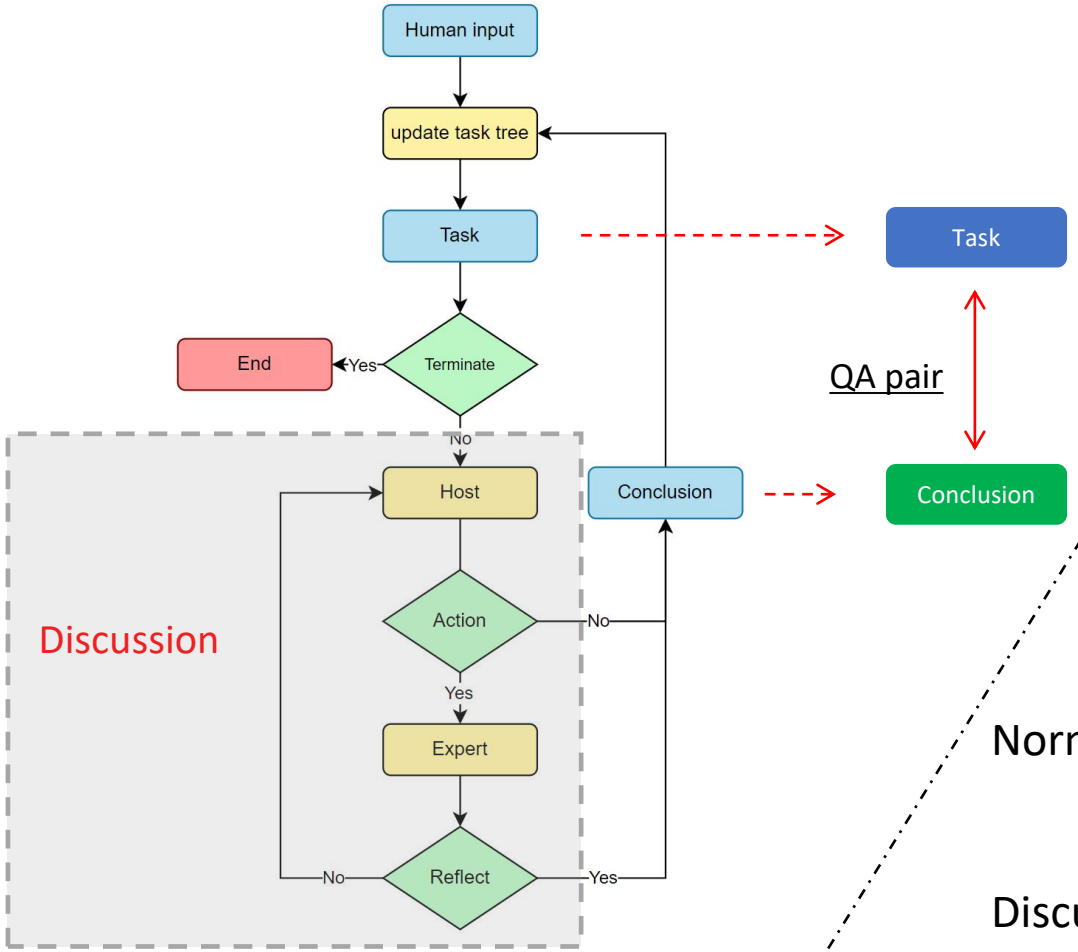
The architecture of Dr.Sai-BESIII

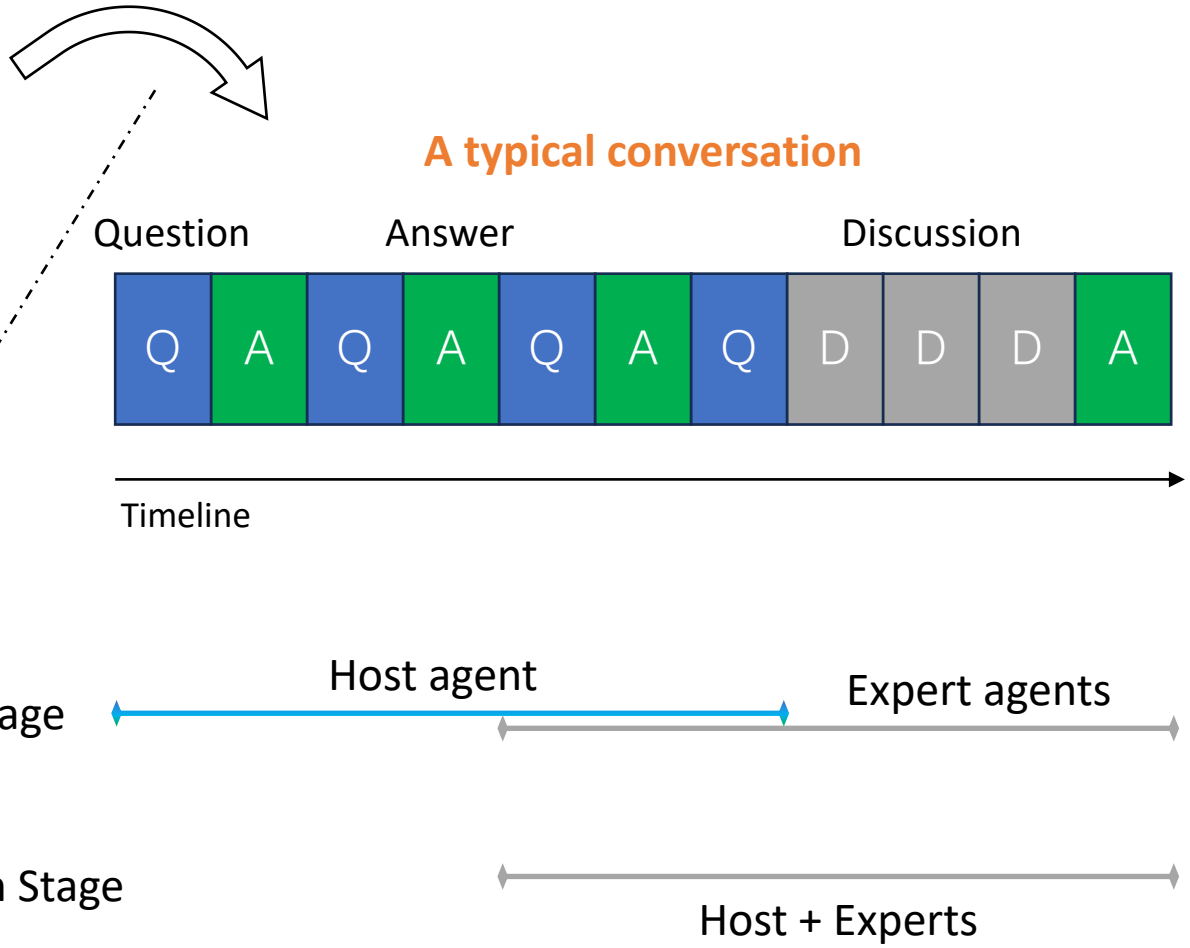


Workflow

Context Engineering in MAS

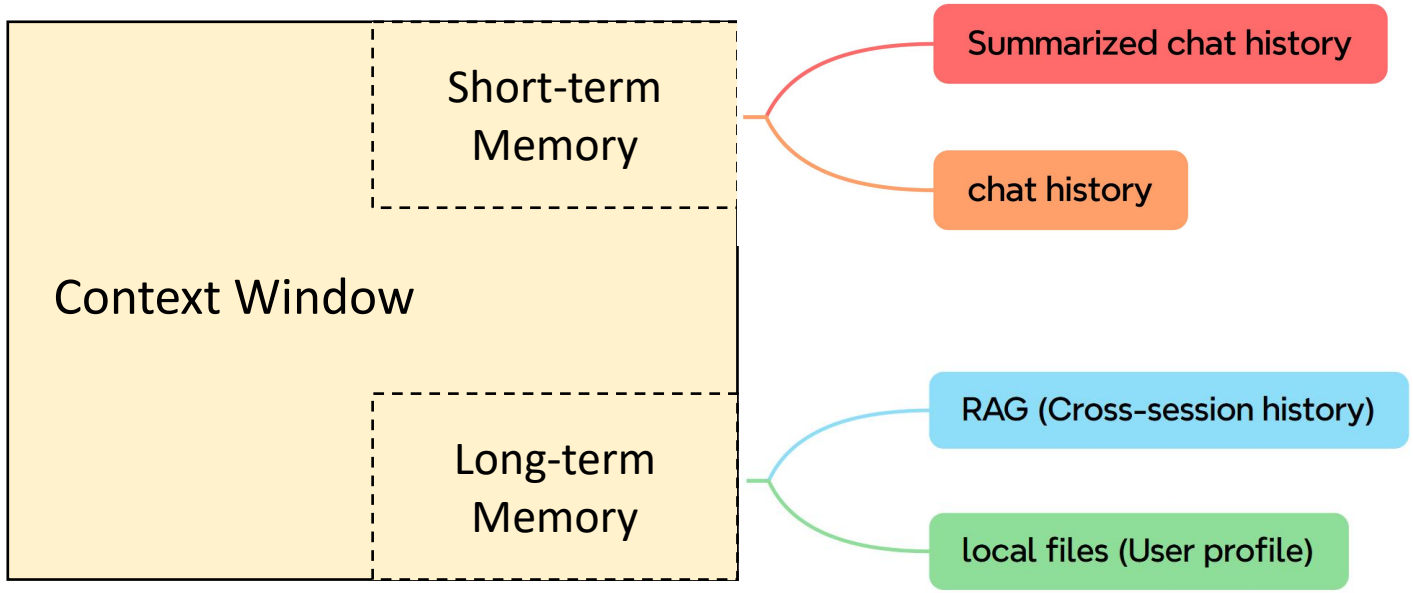
A task-based memory system is adopted to enable context engineering across long-running, interruptible analytical tasks.





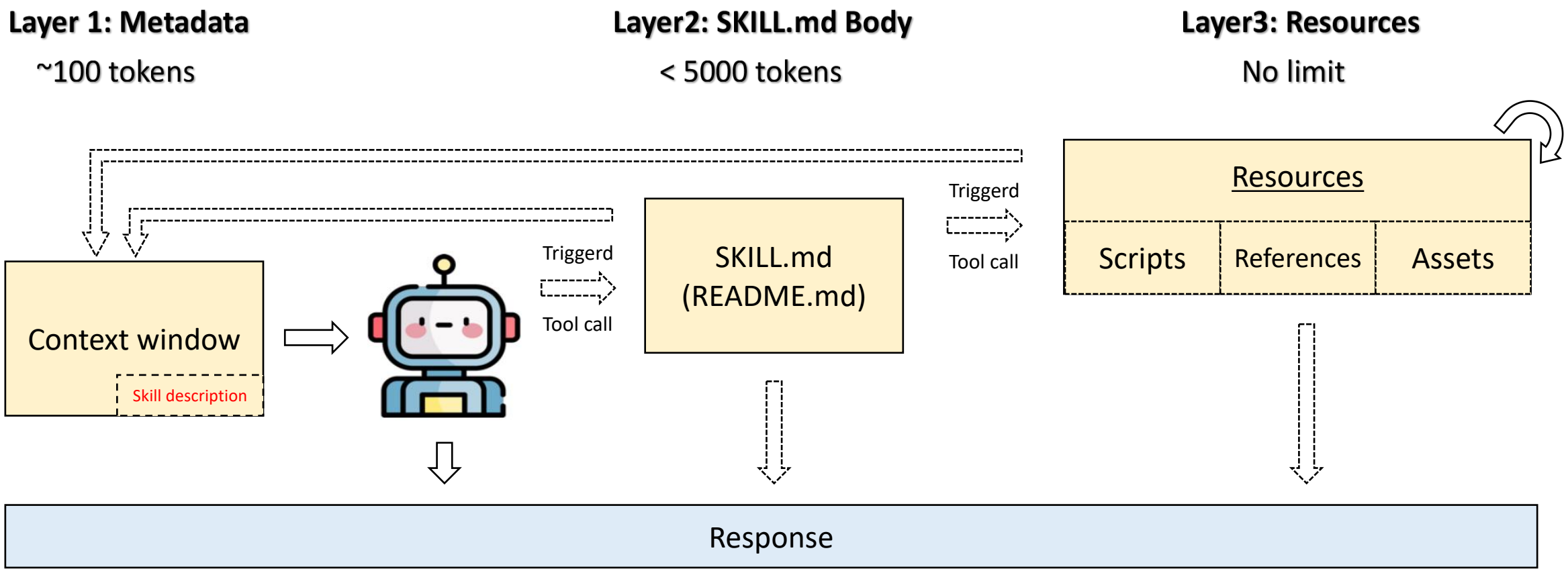
Context Engineering in MAS

- Short-term Memory
  - **Summarized chat history:** triggered upon reaching the token limit
  - **Chat history:** accumulated incrementally, orchestrated with message offloading mechanisms and other strategies
- Long-term Memory
  - **RAG:** retrieve contents either always or on demand, returning the most relevant results
  - **Local files:** always triggered during agent invocation



Skill: Progressive Disclosure

- **Description:** a structured function package designed for a specific task. It features a progressive disclosure mechanism: The model evaluates the metadata to determine whether to initiate a skill. If so, it then loads the body and resources.
- **Components:** consists of descriptions, tools, and resources.



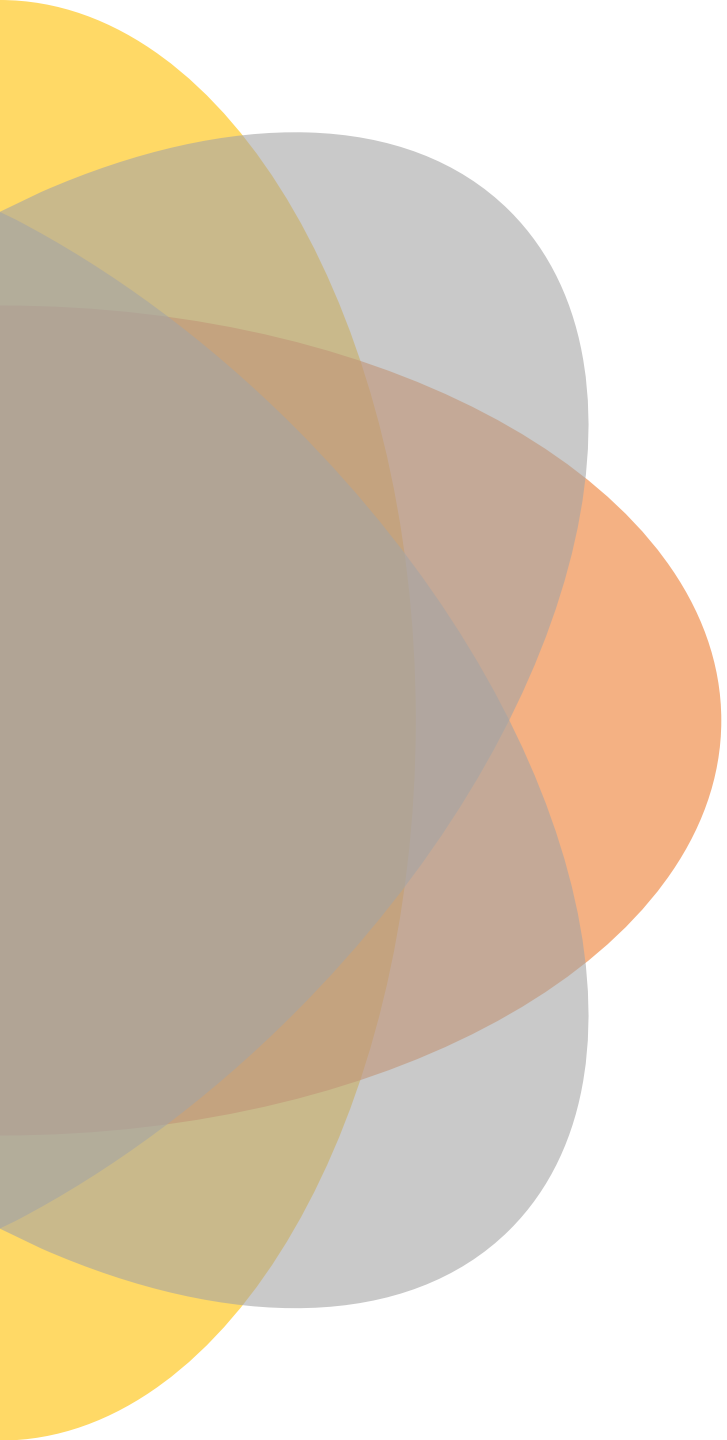
Summary

- Context engineering means strategically deciding what to present, how to structure it, and when to introduce it — all within a limited context window.
- Be cautious in using your 'attention budget'
- Don’t just fill the window — engineer it.

Future Directions:

- Agentic Context Engineering (ACE): Let the agent manage its own context dynamically.
- Context Asymmetry: Models understand well but generate poorly — a key research gap.

Thanks



Backup