



MOSAIC: Memory-Oriented Skill-powered Agent with Interactive Coordination

Wensong Wang^{1†}, Yushan Cheng^{1†}, Fangyi Gan^{1†}, Haiqiang Liu^{1†}

Group 8, School of Computer Science and Engineering, Northeastern University¹



Module 1 — 刘海强

研究问题

- No unified message orchestration layer
- No multi-turn conversation support
- No checkpoint recovery on interruption

解决方法

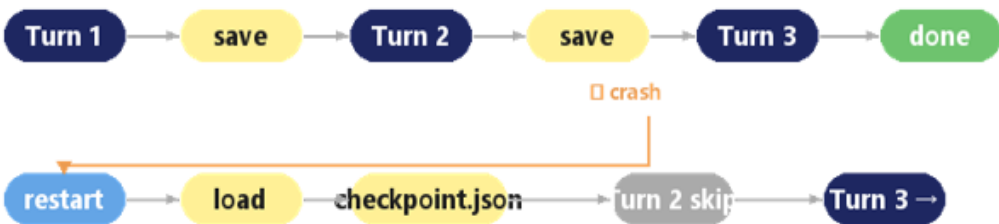
message orchestration:



multi-turn dialogue:



resume from checkpoint:



实验结果

- **Message orchestration** — Closed-loop chain (system → user → assistant → tool → assistant) with per-turn trace logs
- **Multi-turn dialog** — Batch and CLI modes; messages accumulate across turns
- **Checkpoint recovery** — Auto-save before each turn; resume from breakpoint, skip completed turns

B2方向Skill 工具函数模块 — 淦方毅

研究问题

Local LLMs cannot reliably perform external actions such as calculation, file reading, table analysis, web search, and code validation. Therefore, the Agent needs a set of safe, unified, and callable Skill functions.

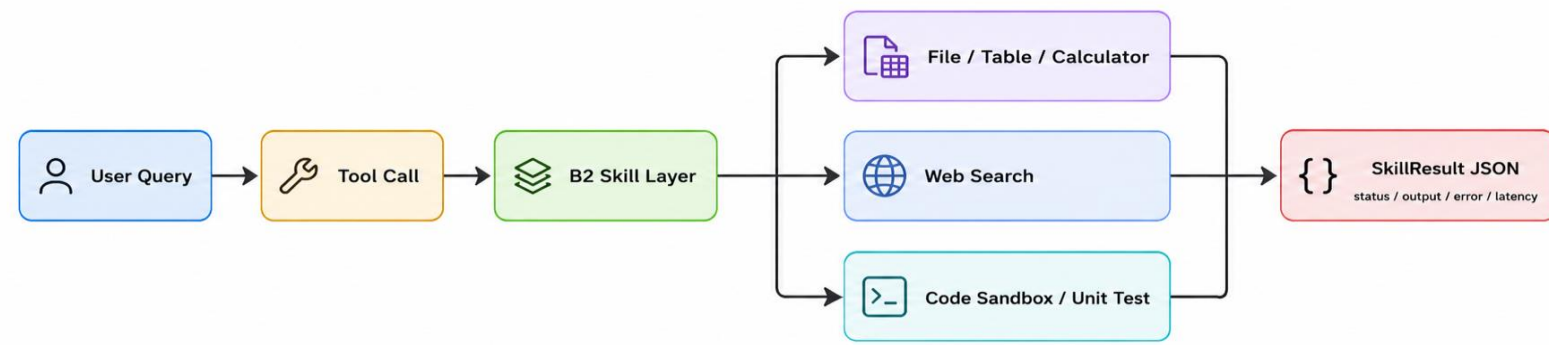
解决方法

I implemented the B2 Skill execution layer with a unified entry point, b2_run_skill.py, which wraps the outputs of 9 Skills into a standard SkillResult. Basic Skills cover calculation, file reading, local search, table analysis, and format conversion. Extended Skills support combined conversion, Python sandbox execution, unit test sandboxing, and web search. Risky operations are protected with path restrictions, AST checks, timeout limits, and subprocess isolation.

实验结果

All 9 Skills were registered in extended_tools. The B3 schema check returned tool_count=9 and aligned_tools=9. B2 standalone tests, B3 tool call execution, and the B1 full demo all passed. web_search supports live web retrieval, and python_unit_test_sandbox reports passed and failed test cases.

B2 Skill Tool Execution Flow



Module 4 — 程宇山

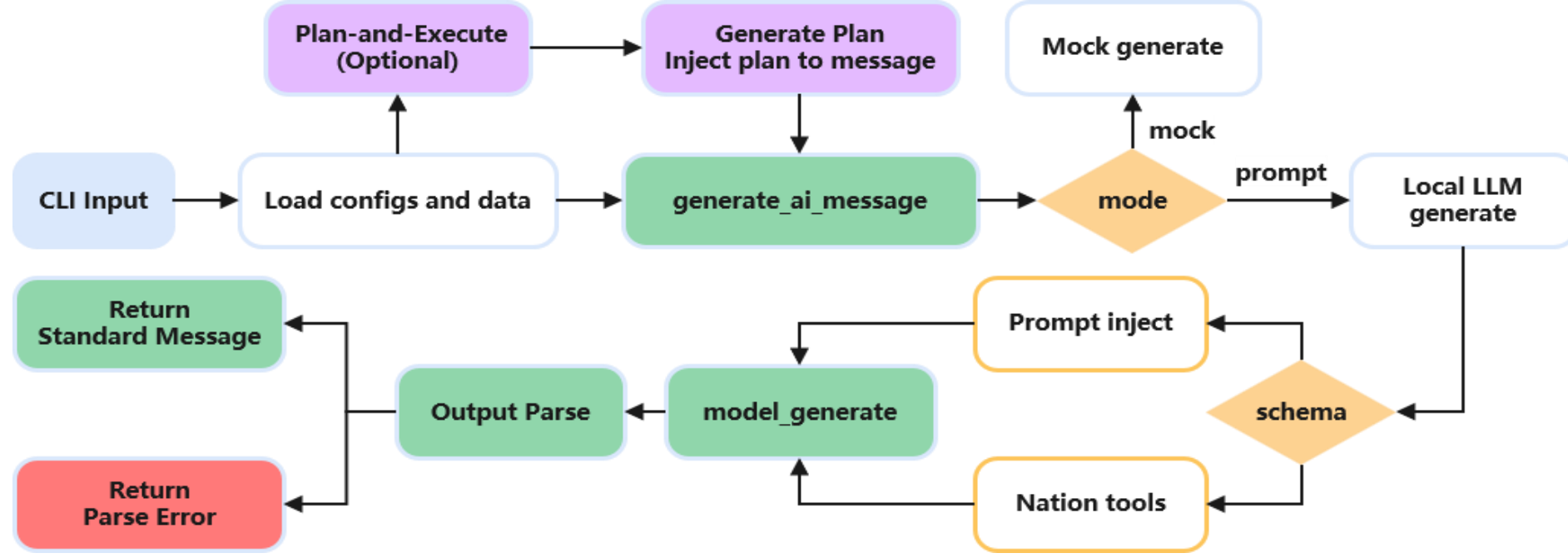
研究问题

- How to enable local large language models to participate in the Agent tool calling workflow stably and standardly.
 - The output format of the local model is unstable.
 - Tasks may require collaborative tool invocation.

解决方法

- A unified AIMessage generation process has been designed to standardize output formats.
- Implement two transmission methods for tools_schema
- Introduce the Plan-and-Execute paradigm

执行流程



结果呈现

- ai_message.json: Standardized model output
- raw_model_output.json: Raw model outputs, parsing results and error logs
- plan.json: Plans generated by Plan-and-Execute

Module 3 & 5 — 王稳淞

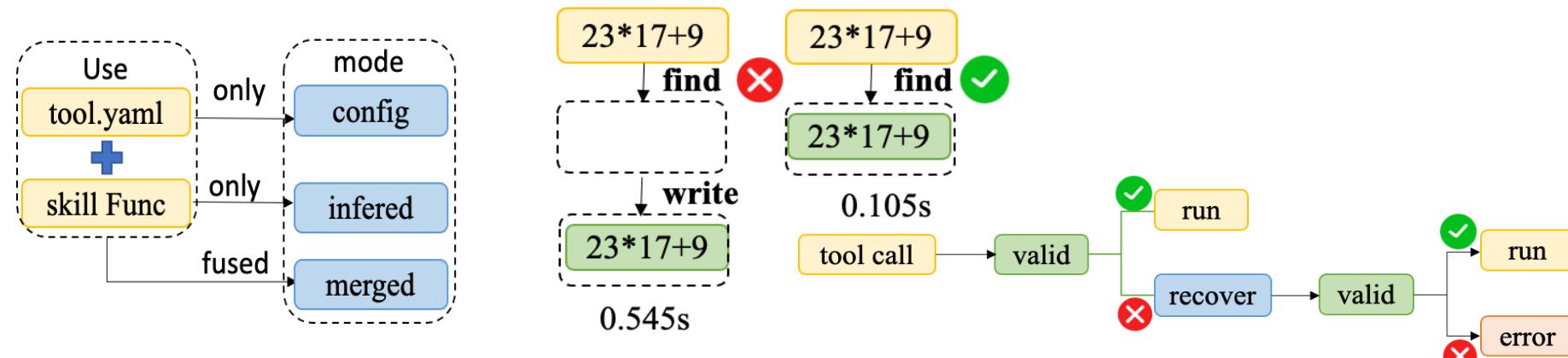
研究问题

- B3: How can we make local LLM tool calls robust, executable, and measurable?
- B5: How can we automatically retrieve, compress, and inject relevant memories for local Agent execution?

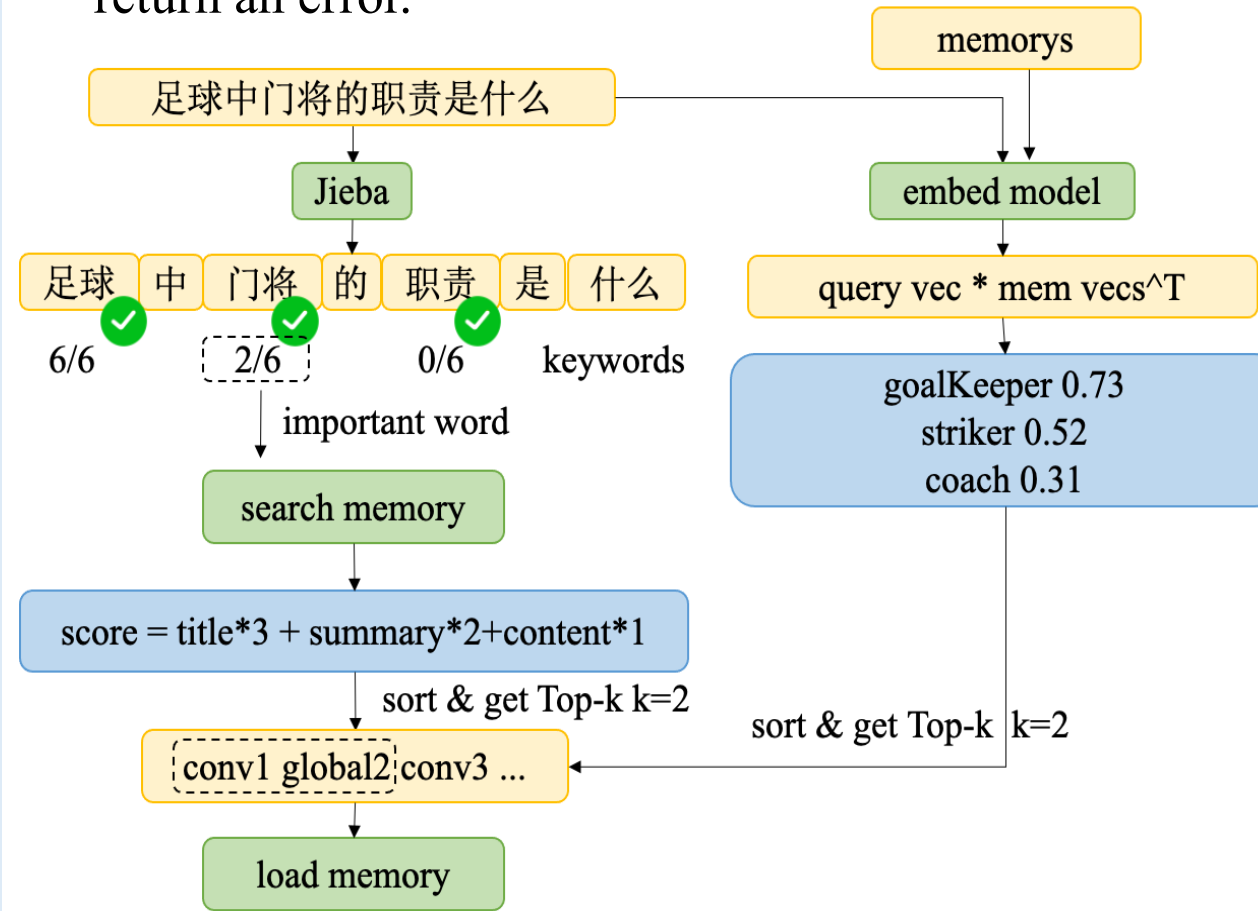
解决方法

B3: exploration 1 2 3

- Generate tool schemas from three sources: Skill functions, tools.yaml, or a merged version of both.
- Cache lookup before tool execution; cache miss triggers execution and result storage.



- Recover malformed tool calls by trimming names/arguments, parsing JSON strings, and coercing numeric or boolean strings; validate again before execution, otherwise return an error.



B5: exploration 1 2 4

- Keyword retrieval uses jieba keywords and weighted title / summary / content hits to rank top-k memories.
- Vector retrieval uses localfastemded embeddings to rank memories by query-document similarity.
- Long memories are compressed when they exceed max_doc_chars, using LLM summarization with rule-based fallback

实验结果

Experiments show that B3 reliably converts Skills into executable tool calls: schema auditing aligned all 5 tools with 0 warnings, recovery fixed and executed 4/4 malformed tool calls, and caching reduced repeated execution with 3/6 hits in one batch and 6/6 hits on a shared-cache rerun. For B5, keyword retrieval scanned 6 memory documents and returned the top-2 relevant results, long-memory compression reduced 6339 characters to 1847 with LLM summarization and 1722 with rule-based fallback, and vector retrieval with local fastemded returned top similarities of 0.631656 and 0.604013.

整体系统

- ✓ 系统描述: A local tool-using Agent framework where B1 orchestrates execution, B2 provides 9 Skills, B3 manages tool calls, B4 standardizes local LLM outputs, and B5 handles memory. It supports multi-turn dialogue, checkpoint resume, prompt switching, compression, and logging.
- ✓ 系统表现: The B1-B5 end-to-end workflow has been successfully verified. The system can generate standard artifacts such as messages.json, trace.json, final_answer.md, tool_messages.json, tools_schema.json, and memory records. All 9 extended Skills are registered; B3 validates aligned tool schemas, recovers malformed tool calls, and caches repeated calls. B4 handles single-tool, multi-tool, final-answer, and error cases, while B5 supports keyword/vector memory retrieval and long-memory compression.