LightWM: Training-Free Hierarchical Working Memory for Small Language Model Agents

Ziyi Wang [1], Haonan Jin [1], Zian Wang [1], Wendong Wang [1], Lanshan Zhang [1]

1 Beijing University of Posts and Telecommunications

ICML International Conference On Machine Learning

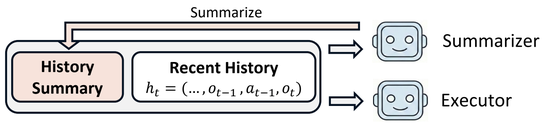
Motivation

Context Processing Strategy

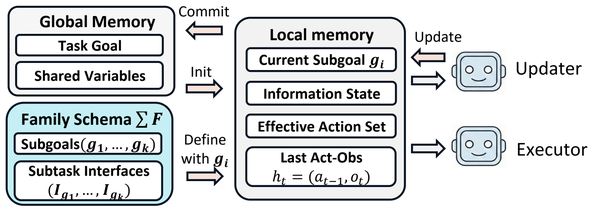
No Working Memory

Full History $h_t = (o_1, a_1, o_2, a_2, \ldots, o_{t-1}, a_{t-1}, o_t)$ → Agent

Summarization-based Working Memory



Our Working Memory (LightWM)



LLM/SLM Feasibility

LLM		• Costly Context
SLM		• Context Overflow
LLM		• Precise Summary
SLM		• Lossy Summary
Finetuned SLM		• Extra Cost
SLM		• Information-efficient Context

Key Contributions

- **POMDP-grounded memory formulation**
  We view agent memory as a compact belief/state interface that exposes goals, observations, valid actions, transition constraints, and subtask returns for decision making.
- **Hierarchical Working Memory**
  We separate task-level global memory from subtask-level local memory to maintain compact, goal-conditioned state across long-horizon execution.
- **Schema-Governed State Updates**
  We replace raw histories and free-form summaries with structured memory fields, enabling controlled updates of decision-relevant information.

Experiment

Method	Qwen3-4B (FP8)	Qwen3-4B (FP16)	Qwen3-8B (FP16)	Gemini-2.5 Flash
ReAct (Yao et al., 2023b)	0.048	0.062	0.579	0.786
Simply Summarize	0.179	0.228	0.469	0.531
HiAgent (Hu et al., 2025a)	0.235	0.248	0.503	0.558
MemGPT (Packer et al., 2024)	0.298	0.317	0.497	0.821
LightWM (Ours)	0.821	0.910	0.828	0.945

LightWM substantially improves long-horizon task success across SLM backbones.
On ALFWorld valid-unseen, Qwen3-4B FP16 reaches 0.910 success, far above ReAct, summarization-based memory, HiAgent, and MemGPT baselines.

Ablation Study

Method	HWM	Act.	Patch	Succ.	Tok/call
Ours	✓	✓	✓	0.910	723 / 32
−Act.	✓	–	✓	0.821	785 / 30
−Patch	✓	✓	–	0.766	**686** / 139
HWM only	✓	–	–	0.697	759 / 140
MemGPT	–	–	–	0.317	950 / 44

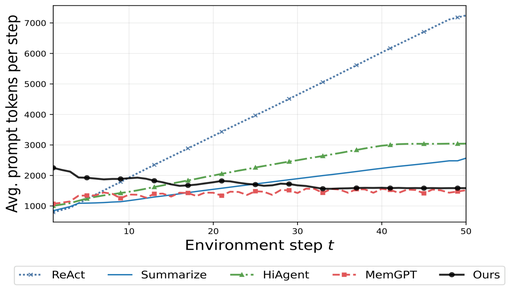
Structured hierarchical memory is the main source of performance gain.
Removing or weakening the schema-governed memory components reduces success, showing that the benefit is not only from prompting or action templates.

Token Cost

Method	Succ.	Tok/ep	Tok/act	Tok/succ
Qwen3-4B (FP16)				
ReAct	0.062	328.6K	6671	5294.1K
Simply Summarize	0.228	106.8K	2336	469.3K
HiAgent	0.248	125.2K	2750	504.2K
MemGPT	0.317	84.7K	3079	267.1K
Ours	0.910	36.2K	1848	39.8K
Gemini-2.5 Flash				
ReAct	0.786	146.3K	4088	186.1K
Simply Summarize	0.531	58.1K	1591	109.5K
HiAgent	0.559	35.3K	1190	63.2K
MemGPT	0.821	39.0K	1381	47.5K
Ours	0.945	34.1K	1891	36.1K

LightWM solves tasks in fewer environment steps.
By maintaining compact subgoal-conditioned state, the agent completes episodes more efficiently instead of repeatedly consuming the 50-step budget.

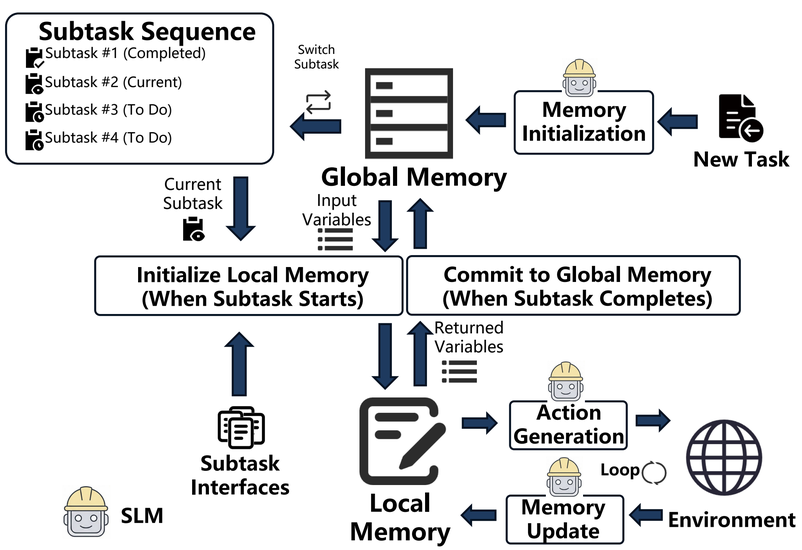
Context Growth



LightWM keeps decision context bounded over long interactions.
Raw-history and summary-based methods accumulate growing prompts, while schema-governed local memory stabilizes the context exposed to the SLM at each step.

Design

Framework Overview



Local Memory Fields

Field	Role	R/W
Core	Read-only subtask identity, goal, and natural-language termination condition	R
Context	Minimal goal-conditioned information state for the active subtask (initialized from M^G and updated online)	R/W
Actions	Subtask-specific action templates and executor constraints (offline, fixed)	R
Rules	Typed constraints for allowable updates; defines the patch grammar and guarded operators	R
Return	Structured subtask outputs committed back to M^G upon completion	R/W

LightWM converts long interaction histories into a compact, schema-governed working memory. A persistent **global memory** tracks task-level progress across subtasks, while a subgoal-specific **local memory** exposes only the information needed for the current decision: objective, context, actions, rules, and return values.

During online execution, the agent updates only selected writable fields through constrained patches, keeping memory **compact, controllable, and decision-relevant.**