LightMem: Lightweight and Efficient Memory-Augmented Generation

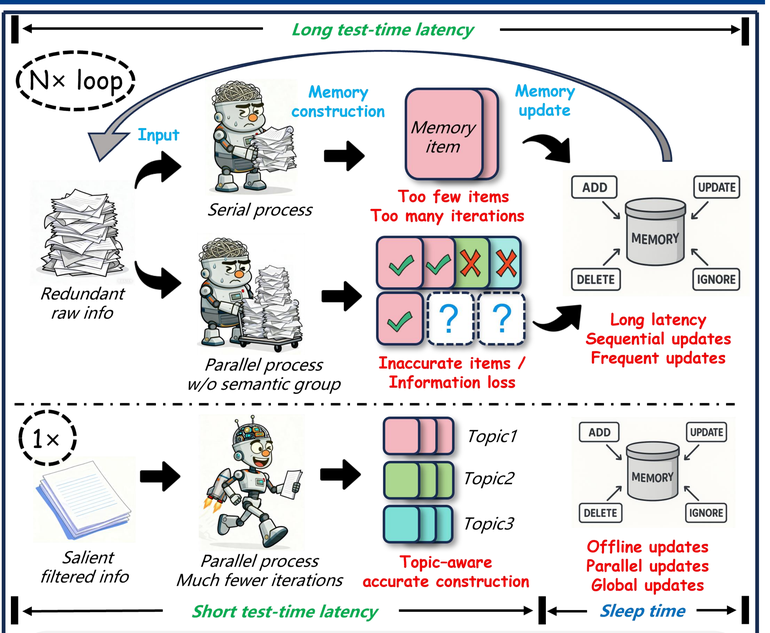
Jizhan Fang[1], Xinle Deng[1], Haoming Xu[1], Ziyan Jiang[1], Yuqi Tang[1], Ziwen Xu[1], Shumin Deng[2], Yunzhi Yao[1], Mengru Wang[1], Shuofei Qiao[1], Huajun Chen[1], Ningyu Zhang[1†]

[1] Zhejiang University [2] National University of Singapore

GitHub Paper

ICLR

Limitations of Current LLM Memory Systems



Motivation 1: Noisy & Redundant Memory Input

Long interactions contain substantial redundant information.
Existing methods directly store raw context without filtering.
→ Leads to token inefficiency and even degraded in-context learning.

☞ Memory systems lack information selection mechanisms.

Motivation 2: Fragmented & Shallow Memory Construction

Current approaches treat turns independently or rely on fixed context windows.
Fail to capture cross-turn semantic connections.
→ Results in incomplete or entangled memory representations.

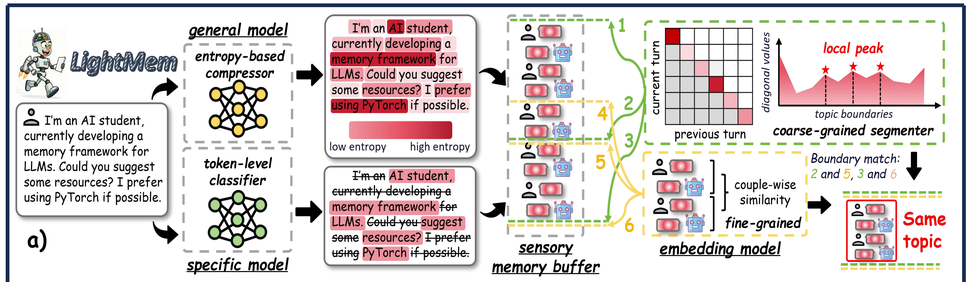
☞ Memory construction lacks global semantic coherence.

Motivation 3: Inefficient & Myopic Memory Updating

Memory updates/forgetting happen during inference (test-time coupling).
→ Causes high latency in long-horizon tasks.
→ Prevents deep, reflective processing of past experiences.

☞ Memory systems lack decoupled and reflective update mechanism.

LightMem: Three-stage Memory Framework



Light1: Sensory Memory (Pre-processing)

LightMem first filters raw interaction streams by a **sensory memory module** *to reduce redundancy. A pre-compression model or a generative model* **selectively retains informative tokens** *based on* **retention probability or entropy**.

$$P(\text{retain } x_i \mid \mathbf{x}; \theta) = \text{softmax}(\ell_i)_1$$

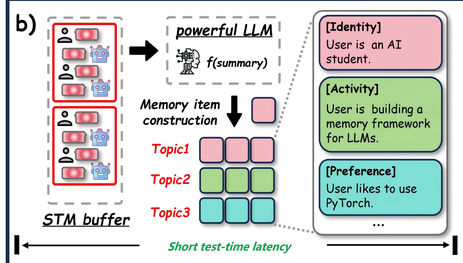
$$P(\text{retain } x_i \mid \mathbf{x}; \theta) = -\sum_{x_i \in \mathcal{V}} q(x_i) \log P(x_i \mid \mathbf{x}; \theta)$$

Light1: Sensory Memory (Topic Segment)

A hybrid topic segmentation mechanism combines **attention patterns** *and* **semantic similarity**. *This produces coherent, topic-level segments that serve as clean and structured inputs for* **downstream memory construction**.

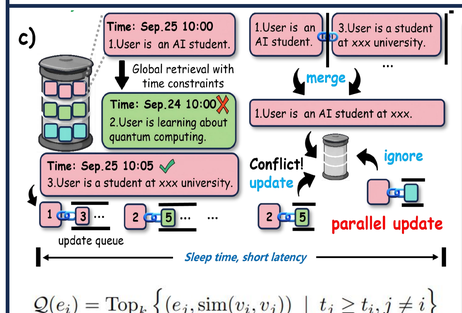
$$\mathcal{B}_1 = \{k \mid M_{k,k-1} > M_{k-1,k-2},\ M_{k,k-1} > M_{k+1,k},\ 1 < k < n\},$$

$$\mathcal{B}_2 = \left\{k \mid \text{sim}(s_{k-1}, s_k) < \tau, 1 \le k < n\right\}, \quad \mathcal{B} = \mathcal{B}_1 \cap \mathcal{B}_2.$$



Light2: Short-Term Memory (Topic-aware Summarization)

A backbone LLM generates concise summaries for each topic, forming **structured memory entries**. *This topic-aware design* **avoids noisy mixing** *across sessions while* **minimizing API calls** *and preserving summarization accuracy.*



$$\mathcal{Q}(e_i) = \text{Top}_k\left\{(e_j, \text{sim}(v_i, v_j)) \mid t_j \ge t_i, j \ne i\right\}_{:n}$$

Light3: Long-Term Memory (Decoupled & Parallel Update)

LightMem adopts a decoupled memory update strategy by inserting entries into long-term memory during inference and deferring updates to **a sleep-time phase**. *A similarity-based* **update queue** *is constructed for each entry with* **temporal constraints**, *enabling independent and* **parallel offline updates**.

Experiments & Evaluation

Dataset1: LongMemEval

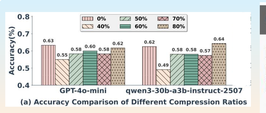
Method	ACC (%)	Summary Tokens (k) In	Summary Tokens (k) Out	Update Tokens (k) In	Update Tokens (k) Out	Total (k)	Calls	Runtime (s)
GPT-4o-mini								
FullText	56.80	–	–	–	–	105.07	–	–
NaiveRAG	61.00	–	–	–	–	–	–	867.38
LangMem	37.20	–	–	982.68	119.48	1,102.16	520.62	2,293.70
A-MEM	62.60	214.66	42.82	1,157.52	190.81	1,605.81	986.55	5,132.06
MemoryOS	44.80	2,302.35	304.18	350.02	35.19	2,991.75	2,938.41	8,030.04
Mem0	53.61	424.13	17.76	560.17	150.56	1,152.62	811.57	4,248.49
LightMem								
r=0.5, th=256	64.29	20.80	10.01	–	–	30.81	25.67	302.69
(OP-update)	64.69	–	–	**44.46**	**2.56**	47.02	70.23	342.63
r=0.6, th=256	67.78	24.58	10.53	–	–	35.11	30.47	329.61
(OP-update)	65.39	–	–	53.98	3.18	57.16	85.07	411.56
r=0.7, th=512	**68.64**	**18.88**	**9.37**	–	–	**28.25**	**18.43**	**283.76**
(OP-update)	67.07	–	–	79.38	4.06	83.44	125.47	496.03
Qwen3-30B-A3B-Instruct-2507								
FullText	54.80	–	–	–	–	105.07	–	–
NaiveRAG	60.80	–	–	–	–	–	–	659.09
LangMem	50.80	–	–	1,311.96	118.06	1,430.02	495.12	3,237.16
A-MEM	65.20	219.21	66.98	1,260.54	318.20	1,864.93	989.30	5,367.51
MemoryOS	49.60	2,101.54	510.88	305.12	27.43	2,944.97	2,922.28	8,721.78
Mem0	39.51	424.20	**15.34**	411.50	111.35	1001.90	722.76	2,239.94
LightMem								
r=0.4, th=768	61.95	**9.01**	16.14	–	–	**25.15**	16.54	357.13
(OP-update)	62.34	–	–	111.13	7.88	119.01	176.02	1036.47
r=0.6, th=768	**70.20**	13.19	19.21	–	–	32.40	19.97	417.13
(OP-update)	65.14	–	–	**97.11**	**5.92**	103.03	152.93	1023.56
r=0.8, th=1024	68.69	14.82	18.49	–	–	33.31	**9.43**	**355.71**
(OP-update)	67.34	–	–	106.91	6.20	113.11	168.37	1026.90

Dataset2: LoCoMo

Method	ACC (%)	Summary Tokens (k) In	Summary Tokens (k) Out	Update Tokens (k) In	Update Tokens (k) Out	Total (k)	Calls	Runtime (s)
GPT-4o-mini								
FullText	71.83	–	–	–	–	–	–	–
NaiveRAG	63.64	–	–	–	–	–	–	–
LangMem	57.20	–	–	898.27	111.95	1010.22	920.62	2229.37
A-MEM	64.16	182.74	49.29	729.89	187.52	1149.43	1175.47	6060.73
MemoryOS(locomo)[!]	58.25	110.98	33.40	78.08	64.54	287.00	553.45	2422.05
MemoryOS(regular)	54.87	226.86	46.61	177.66	75.34	526.48	1016.06	3332.59
Mem0	61.69	851.32	20.53	632.12	189.42	1693.39	1602.20	4432.87
LightMem(0.7,512)	71.95	73.19	20.13	6.05	0.40	99.76	41.65	848.49
LightMem(0.7,768)	70.26	**57.54**	18.92	**3.79**	**0.23**	**80.48**	**29.55**	**737.80**
LightMem(0.8,768)	**72.99**	62.82	**17.95**	4.14	0.28	85.19	29.83	815.32
Qwen3-30B-A3B-Instruct-2507								
FullText	**74.87**	–	–	–	–	–	–	–
NaiveRAG	66.95	–	–	–	–	–	–	–
LangMem	60.53	–	–	1004.35	138.02	1142.37	1005.37	2268.57
A-MEM	56.10	158.29	60.85	924.19	483.51	1626.80	1175.40	5543.90
MemoryOS(locomo)	61.04	122.21	53.12	104.43	81.75	361.51	414.70	1269.70
MemoryOS(regular)	51.30	228.85	51.60	242.27	143.63	666.35	1004.60	1982.20
Mem0	43.31	827.09	**18.64**	763.88	189.80	1799.40	1614.50	4540.70
LightMem(0.6,768)	71.36	**56.68**	34.14	**8.31**	**0.74**	**99.87**	**29.10**	**815.70**
LightMem(0.8,1024)	72.60	61.38	36.33	9.86	0.88	108.45	32.00	1079.40

LongMemEval: +2.09%–6.40% (GPT) and up to +7.67% (Qwen) accuracy, while achieving up to 38×/21.8× token reduction, 30×/17.1× fewer API calls, and 12.4×/6.3× speedup.

LoCoMo: LightMem improves accuracy by 6.10%–29.29% and delivers consistent efficiency gains (up to 20.92× tokens, 55.48× API calls, 8.21× speedup), with sleep-time consolidation further enhancing memory reliability.



sentence, token-level compression and summarization *should be used flexibly.*

Case Study: Memory Update Mechanism Comparison

History1: {'Monday, 2 PM': User is planning a trip to Tokyo.}
History2: {'Monday, 4 PM': User asks about trains to Kyoto.}

Hard Update: Overwrites memory
-> "User plans Kyoto trip"
⚠ Tokyo context lost

LightMem Soft Update: Appends info
-> "Tokyo trip + Kyoto inquiry"
✓ Full context preserved

This case explains why LightMem's **soft update** *mechanism works.*
soft update vs traditional hard update???