Memory OS of AI Agent

arxiv.org/abs/2506.06326

Jiazheng Kang
Beijing University of Posts and Telecommunications
kjz@bupt.edu.cn

Mingming Ji
Tencent AI Lab
matthhewj@tencent.com

Zhe Zhao
Tencent AI Lab
nlpzhezhao@tencent.com

Ting Bai *
Beijing University of Posts and Telecommunications
baiting@bupt.edu.cn

开源代码：https://github.com/BAI-LAB/MemoryOS

简介

本文提出MemoryOS，一种借鉴操作系统中段页式内存管理思想的memory管理方案。MemoryOS构建了由短期记忆(STM)、中期记忆(MTM)和长期记忆(LPM)组成的分层结构，分别用于保持当前对话上下文、聚合历史话题信息与建构用户个性画像，以页(page)为基本单位表示单轮对话内容，并在中期记忆中通过语义聚类将多个相关对话页组织成段(segment)，实现基于话题的记忆组织与存储。为了支持动态的记忆更新，MemoryOS引入了基于热度评分(heat score)的淘汰与转移机制，根据记忆的访问频率、交互活跃度与时间衰减等因素，将中期记忆转为长期记忆或者去掉某个中期记忆。MemoryOS同时从STM、MTM和LPM 三类记忆中检索相关信息，辅助生成上下文一致且个性化的response。

背景

Memory在Agent系统中的重要性已经不需要多说了，近期已经有越来越多的研究工作开始关注memory方向，因为发展演化的快，所以还没有比较统一确定的memory系统架构。本文作者受到操作系统领域中段页式存储结构(segmented paging的启发，提出了一种基于层级memory架构的存储与组织方案，让我们一起学习下。

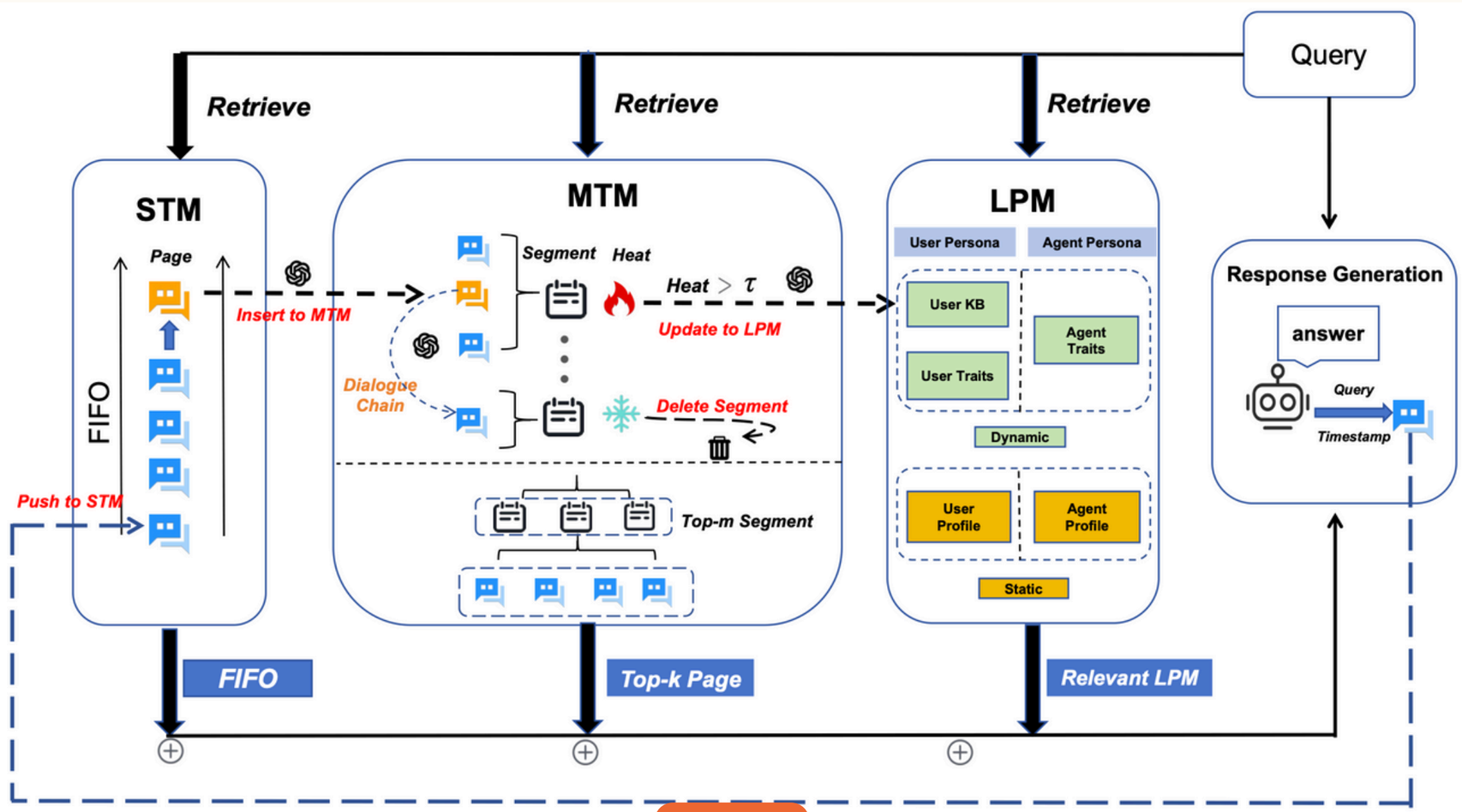
实验设置

- 数据集：GVD和LoCoMo
- backbone模型：GPT-4o-mini和Qwen2.5-7B
- 用如下方式计算page和segment相似度，将page添加到最相似的segment中。考虑向量语义相似度和Jacard相似度

$$\mathcal{F}_{\text{score}} = \cos(\mathbf{e}_s, \mathbf{e}_p) + \mathcal{F}_{Jacard}(K_s, K_p),$$

MemoryOS架构

- 短期记忆(STM)用一个固定大小的队列(queue)实现，每个队列元素称为一个page，包含(query, response, 时间戳和可选的llm生成的上下文摘要)
- 中期记忆(MTM)负责对page进行按主题分类存储，换句话说，MemoryOS是按照主题(topic)分类管理历史交互信息的
- 长期记忆(LTM)主要是用户/Agent个性化因素



部分实验结果

Table 2: LoCoMo dataset comparison with per-category scores and average ranks. A-Mem refers to the results reported in the original paper. A-Mem* represents our implementation results under the same experimental environment as our model.

Model	Method	Single Hop F1 ↑	Single Hop BLEU-1 ↑	Multi Hop F1 ↑	Multi Hop BLEU-1 ↑	Temporal F1 ↑	Temporal BLEU-1 ↑	Open Domain F1 ↑	Open Domain BLEU-1 ↑	Avg. Rank ↓ (F1)	Avg. Rank ↓ (BLEU-1)
GPT-4o-mini	TiM	16.25	13.12	18.43	17.35	8.35	7.32	23.74	22.05	3.8	4.0
	MemoryBank	5.00	4.77	9.68	6.99	5.56	5.94	6.61	5.16	5.0	5.0
	MemGPT	26.65	17.72	25.52	19.44	9.15	7.44	41.04	34.34	2.2	2.5
	A-Mem	27.02	20.09	45.85	36.67	12.14	12.00	44.65	37.06	–	–
	A-Mem*	22.61	15.25	33.23	29.11	8.04	7.81	34.13	27.73	3.0	2.5
	Ours	**35.27**	**25.22**	**41.15**	**30.76**	**20.02**	**16.52**	**48.62**	**42.99**	**1.0**	**1.0**
Improvement (%)		32.35%↑	42.33%↑	23.83%↑	5.67%↑	118.80%↑	111.52%↑	18.47%↑	25.19%↑	–	–

思考

从消融实验结果来看，中期记忆对效果提升最大，说明本文提出的以话题(topic)的方式组织历史交互数据是有效的，并且重要程度比用户/Agent persona信息更重要。我们是否可以再发散思考下，有没有其他更好的方式组织这么多轮的(query, response)数据呢？比如事件(event)？知识图谱？或者balabala。另一个问题是memory的基本单元用哪种数据结构更合理，是原始的(query, response)还是需要对(q, r)做一步提取/摘要？