Microsoft UNIVERSITY OF SURREY

ICLR

Apr. 24 10:00–12:30 #291

SCBench: A KV Cache-Centric Analysis of Long-Context Methods

Yucheng Li[◊], Huiqiang Jiang[†], Qianhui Wu, Xufang Luo, Surin Ahn, Chengruidong Zhang, Amir H. Abdi, Dongsheng Li, Jianfeng Gao, Yuqing Yang, Lili Qiu

Microsoft Corporation, [◊]University of Surrey

https://aka.ms/SCBench

Microsoft Research

KV Cache is not limited to a single turn.

- Long-context methods are designed and utilized around the KV cache, but existing benchmarks focus only on single-request scenarios, ignoring its full lifecycle in real-world use.

Repo-level Code Debugging/ Long-document QA

Multi-turn Dialogue

Self-play Reasoning

(a) Long-Context is shared in real-world scenarios.

RadixAttention

Automatic Prefix Caching

(b) Prefix caching is widely used in LLM framework.

Prompt Caching

Context Caching

Prompt Caching

(c) Prefix caching is widely used in LLM API.

Long-context methods are built around KV Cache.

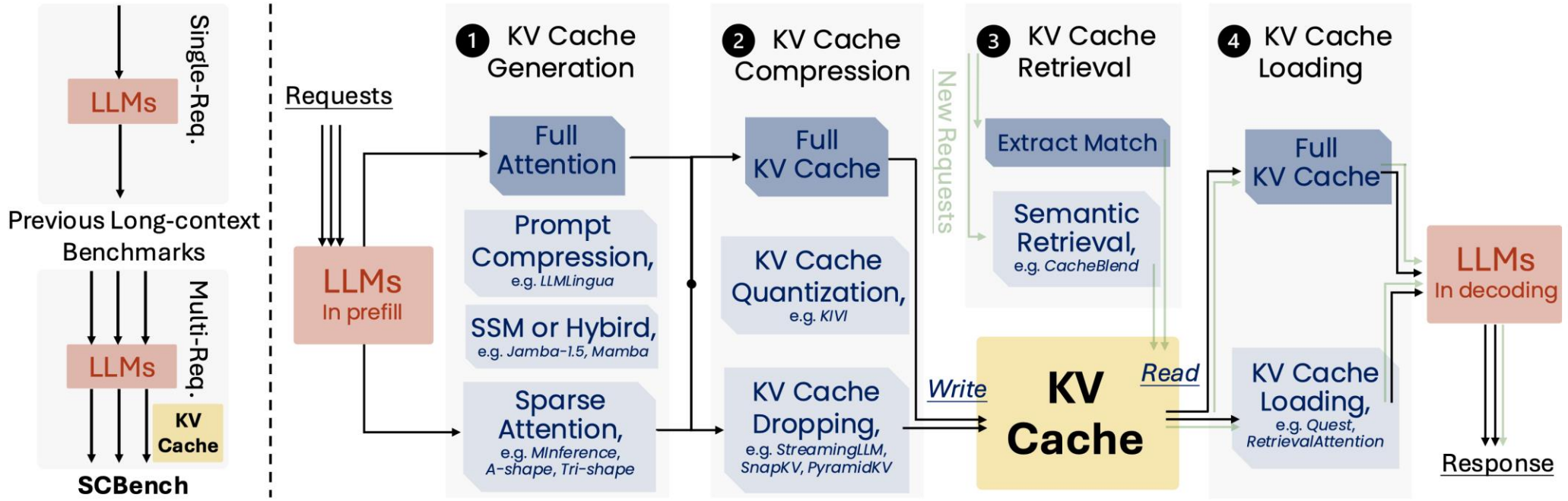
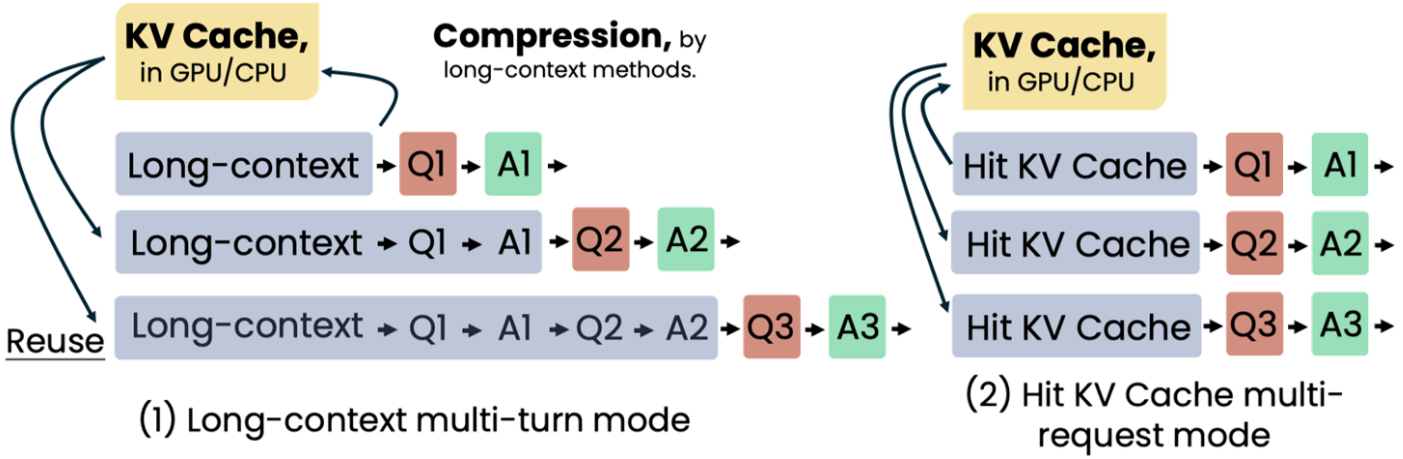


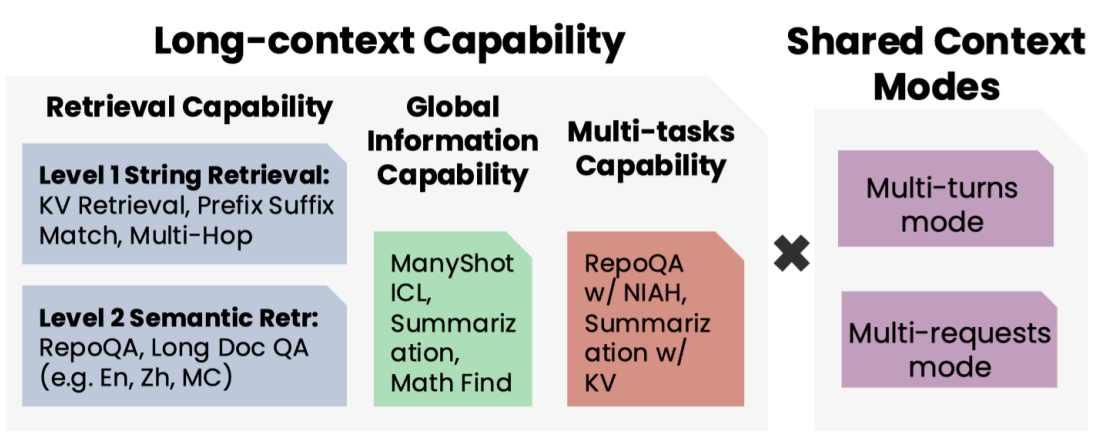
Figure 1: KV Cache lifecycle. Prior benchmarks focus on single-request, while real-world applications reuse KV cache across requests. We propose **SCBench** and categorize long-context methods into KV Cache Generation, Compression, Retrieval, and Loading from a KV-cache-centric perspective.

SCBench

- ❑ Two typical shared context modes;
- ❑ Four category long-context capability;



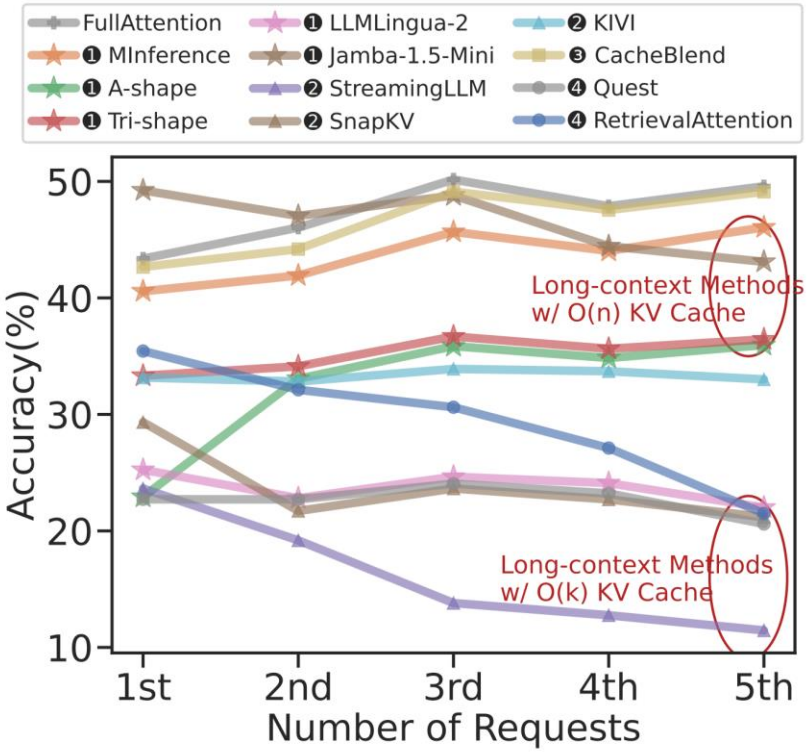
(a) Two Shared Context Modes



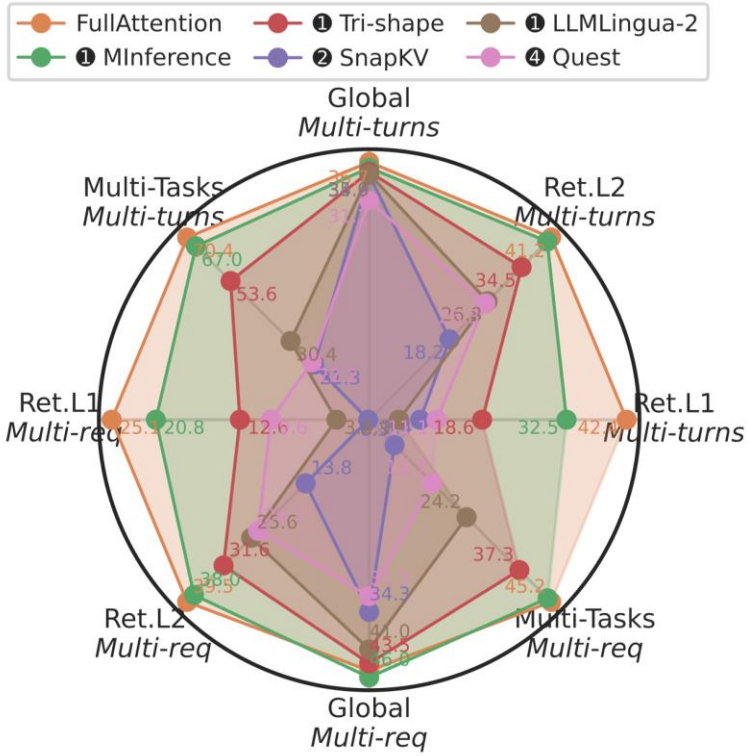
(b) Overview of SCBench

Performance in Multi-turn and Capability

- ❑ Sub-O(n) Memory is Almost Infeasible in Multi-Turn Decoding.
- ❑ Long-generation scenarios exhibit distribution shift issues.



(a) Performance Across Different Requests

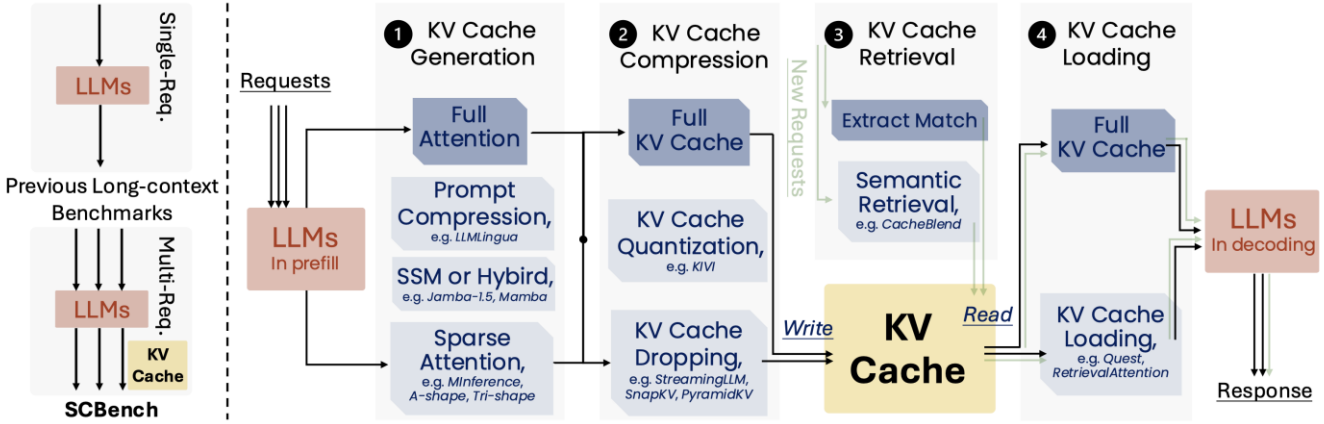


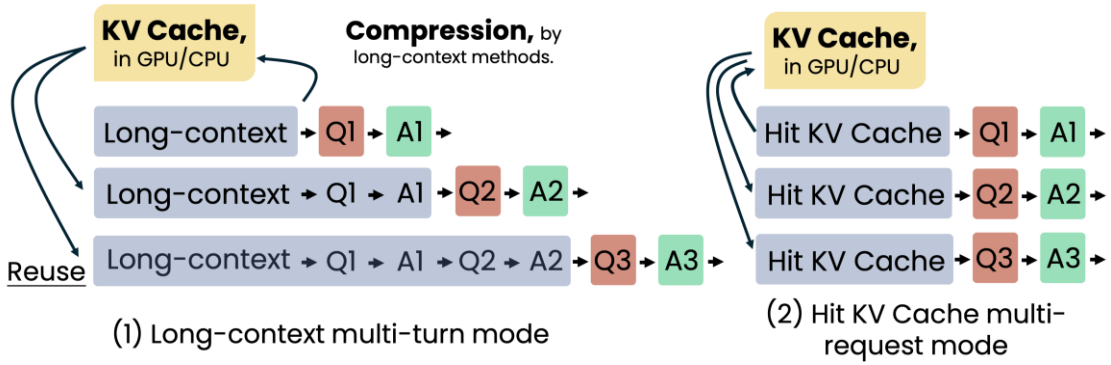
(b) Performance in Different Abilities

SCBench: A KV Cache-Centric Analysis of Long-Context Methods

Project Page | Code

Long-context methods are designed and utilized around the KV cache, but existing benchmarks focus only on single-request scenarios, ignoring its full lifecycle in real-world use.

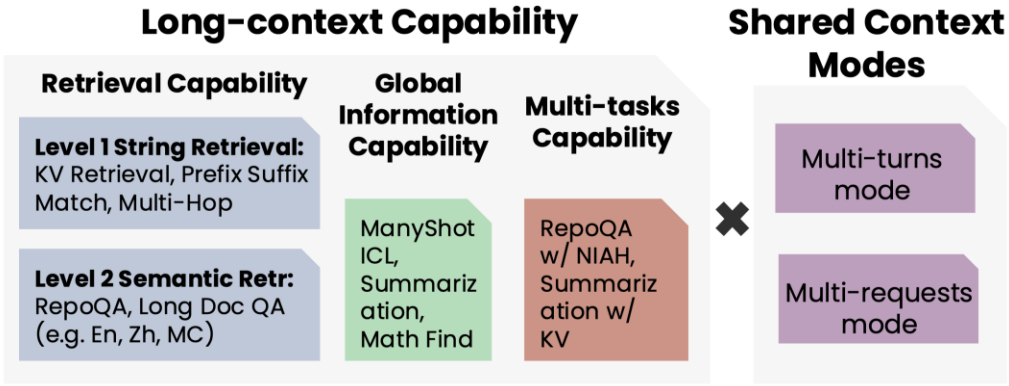




(b) Two Shared Context Modes

We propose ***SCBench***, a **KV cache-centric** benchmark for analyzing long-context methods, covering KV cache generation, compression, retrieval, and loading. It includes **four capability tasks** and **two shared-context modes**, from which we derive the following **insights**:

- *Sub-O(n) memory is almost infeasible in multi-turn decoding;*
- *Task performance shows varying decline trends;*
- *All long-context methods experience performance degradation as the compression rate decreases;*
- *Long-generation scenarios exhibit distribution shift issues.*



(c) Overview of SCBench

https://aka.ms/SCBench

🔍 **SCBench: A KV cache-centric analysis of long-context methods**

🤗: https://huggingface.co/datasets/microsoft/SCBench