RocketKV: Accelerating Long-Context LLM Inference via Two-Stage KV Cache Compression

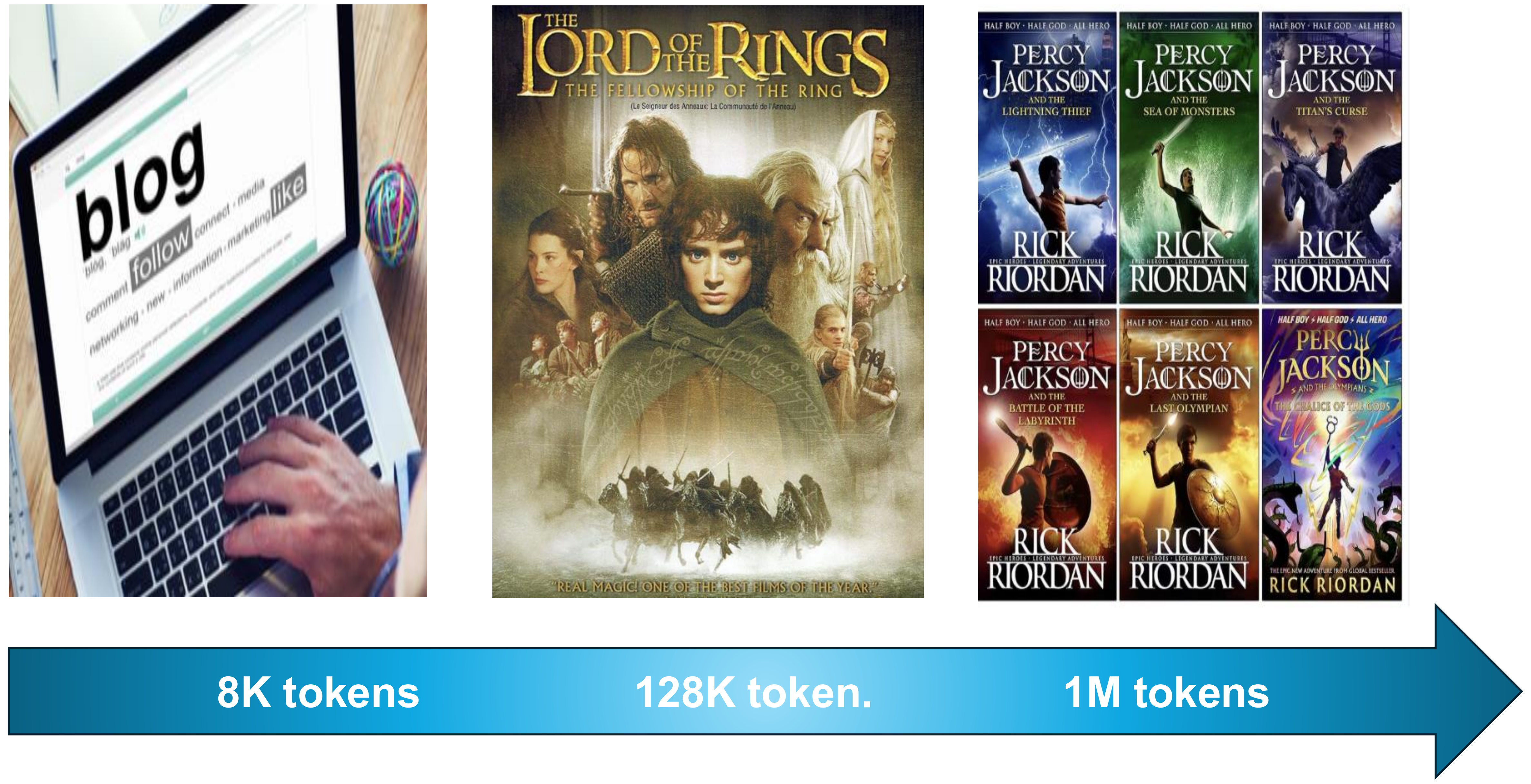
Payman Behnam[1,2]*, Yaosheng Fu[1]*, Ritchie Zhao[1], Po-An Tsai[1], Zhiding Yu[1], Alexey Tumanov[2]

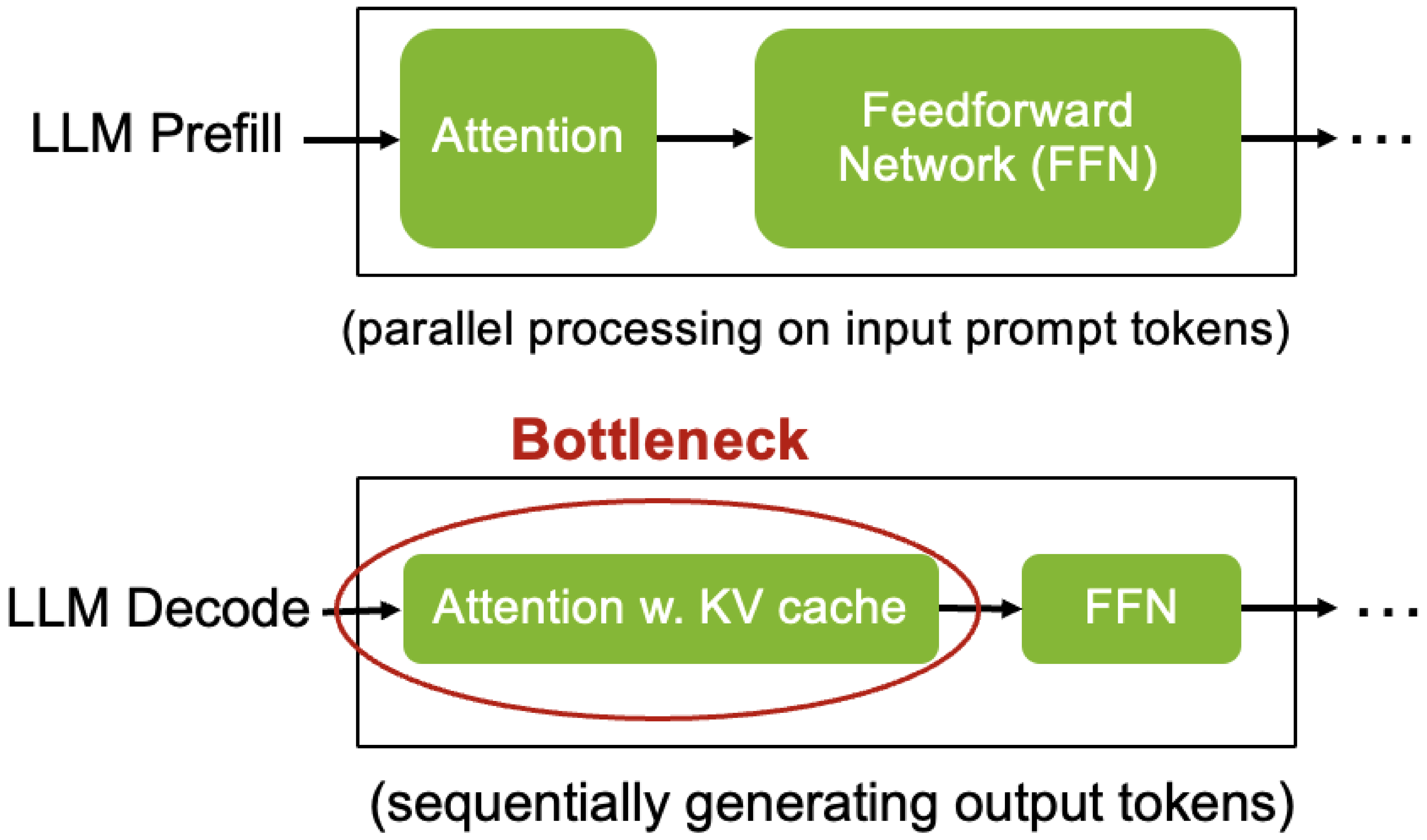
*Equal Contribution

[1]Nvidia, [2]Georgia Institute of Technology

Motivation

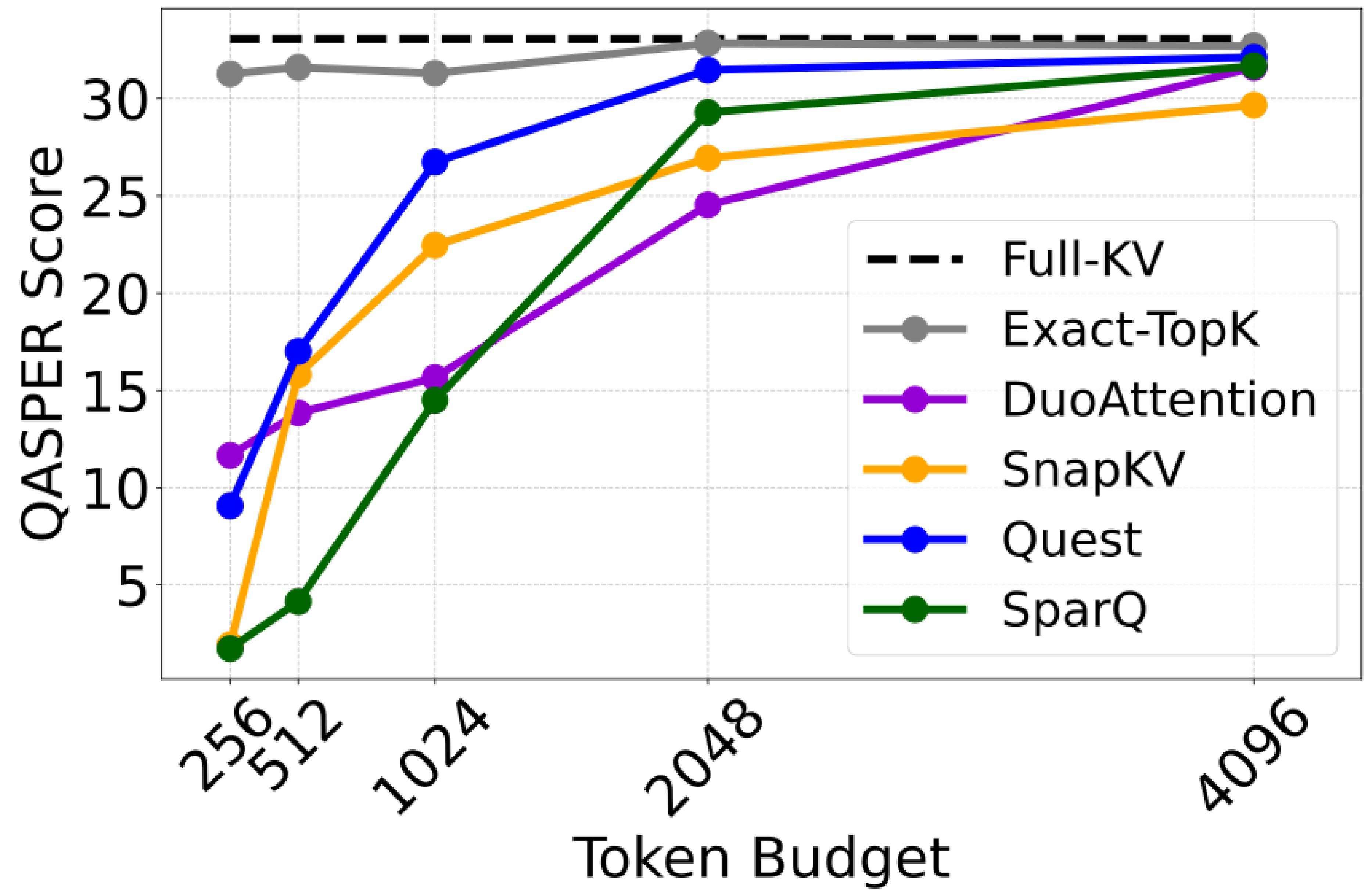
- Increasing context length allows LLM to understand longer documents and videos.
- KV cache in LLM inference stores past attention to avoid recomputation.
- KV cache becomes a major bottleneck at the decode phase in long contexts.

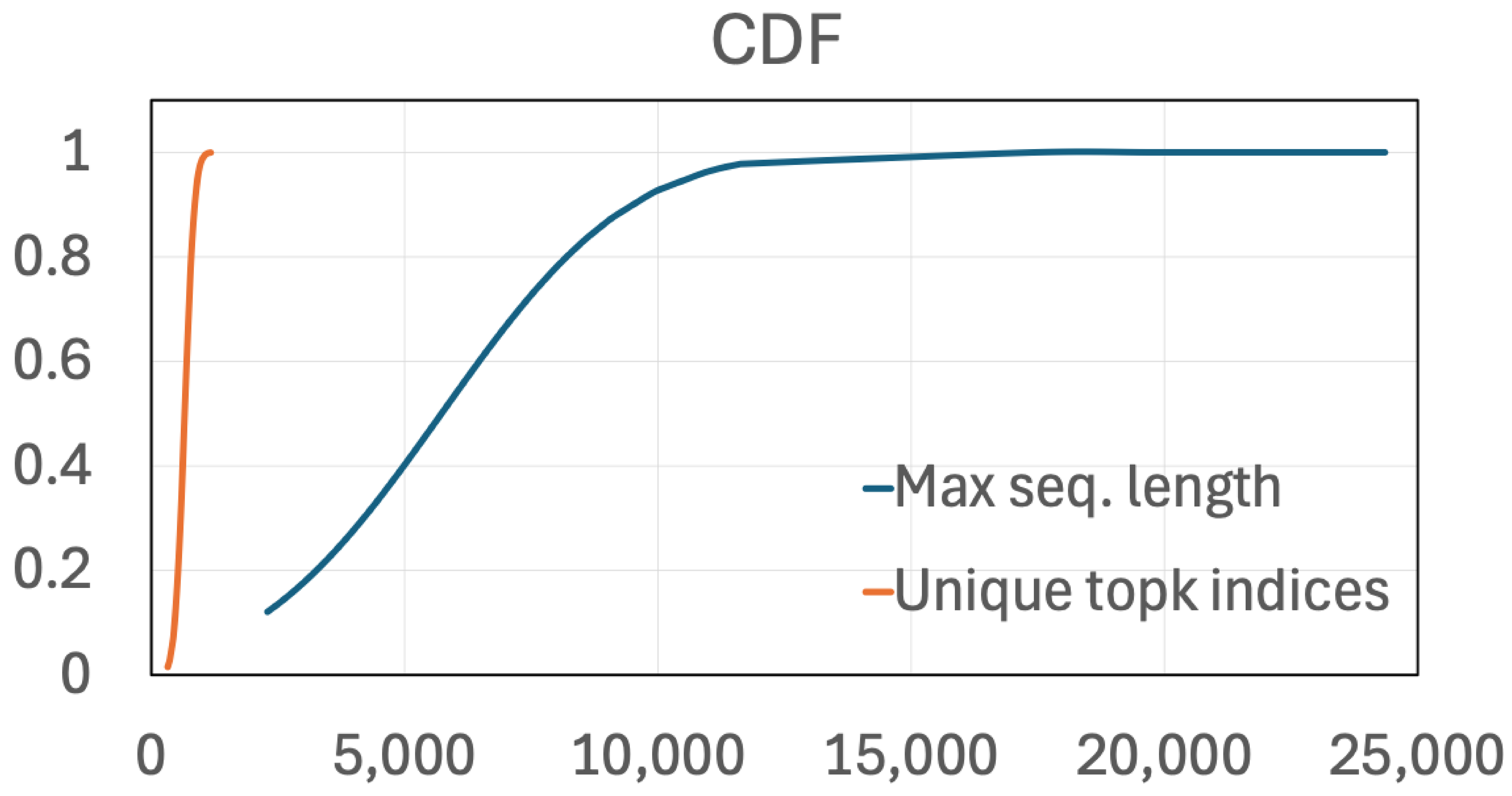




Observation

- Existing KV cache compression methods fail to match the accuracy of oracle top-k attention (Exact-TopK)
- Maximum sequence length: 25000, the number of unique top-k indices: 1200.
- We realize that dynamic KV token selection can be applied on the filtered KV token set after permanent KV token eviction.



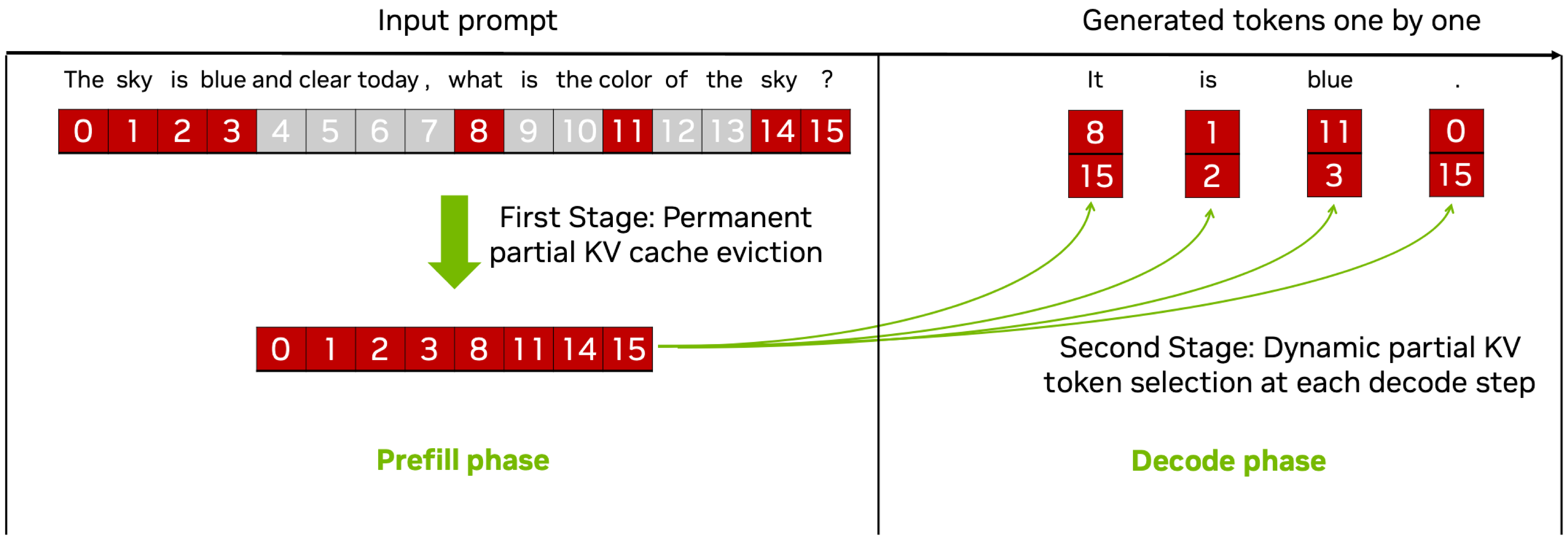


Layer 31, head0 of Top256

qasper benchmark in LongBench on Mistral-7B-Ins-v0.2

RocketKV: Two Stage KV Cache Compression

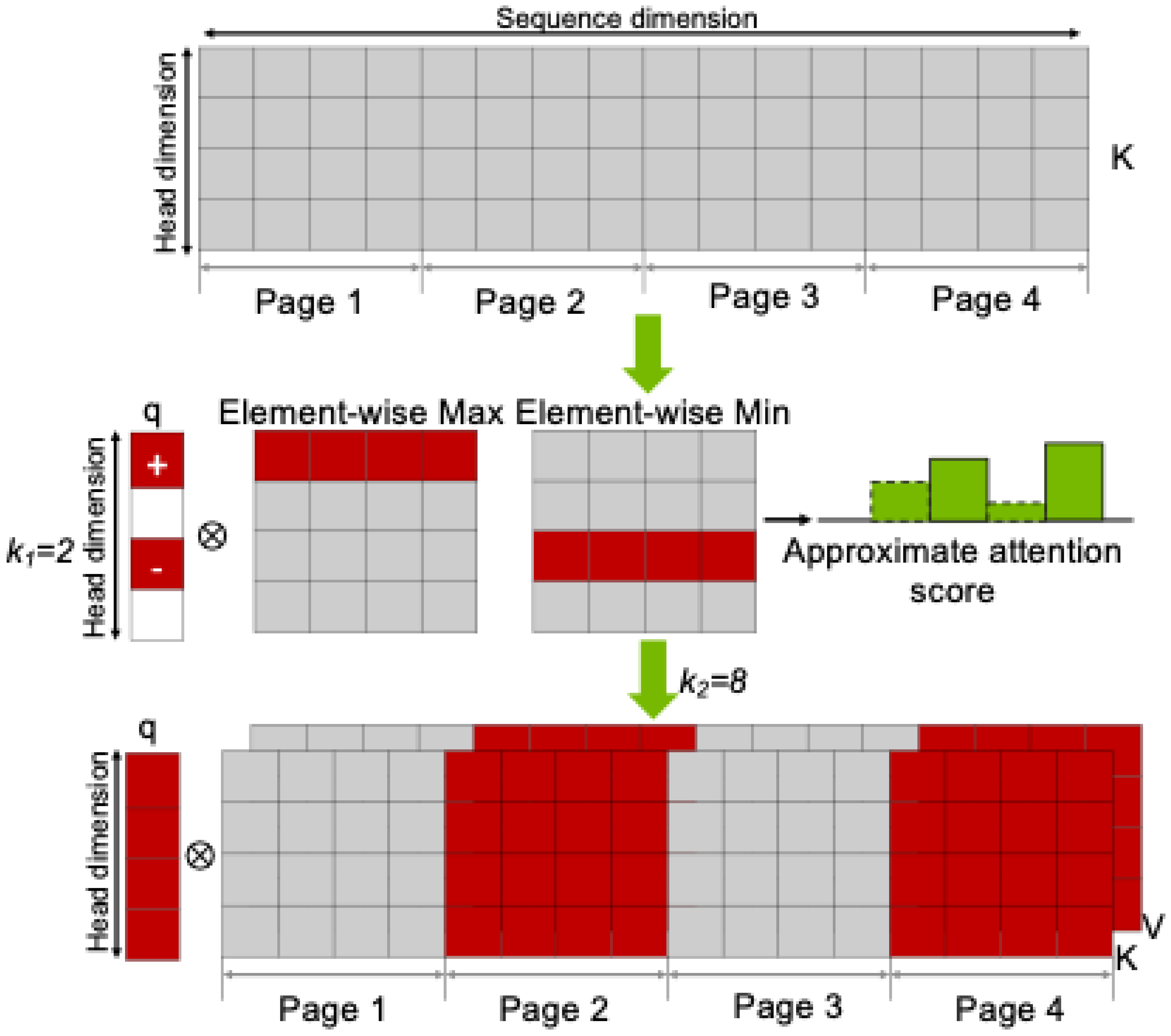
- Two-stage KV cache compression for decode acceleration.



- RocketKV enables flexible integration of a wide range of KV cache compression techniques at each stage.
- First stage (SnapKV):
  - Removes coarse-grain KV tokens with low importance.
- Second stage (Hybrid Sparse Attention):
  - Removes fine-grain KV tokens from the remaining ones.

Hybrid Sparse Attention (HSA)

- Step 1: Token Grouping & Auxiliary Storage
  - Group key tokens into pages and store per-page *Kmax/Kmin* along the head dimension to enable efficient lookup, updating them with each new key token.
- Step 2: Attention Score Approximation
  - For each query, select *top-k1* head positions by magnitude, use *Kmax/Kmin* with selected *q* to estimate max attention scores per page and select *top-k2* pages along the sequence dimension.
- Step 3: Sparse Attention Execution
  - Retrieve original key/value vectors only from the *top-k2* predicted indices and perform attention over them.



Multi-Turn Scenario

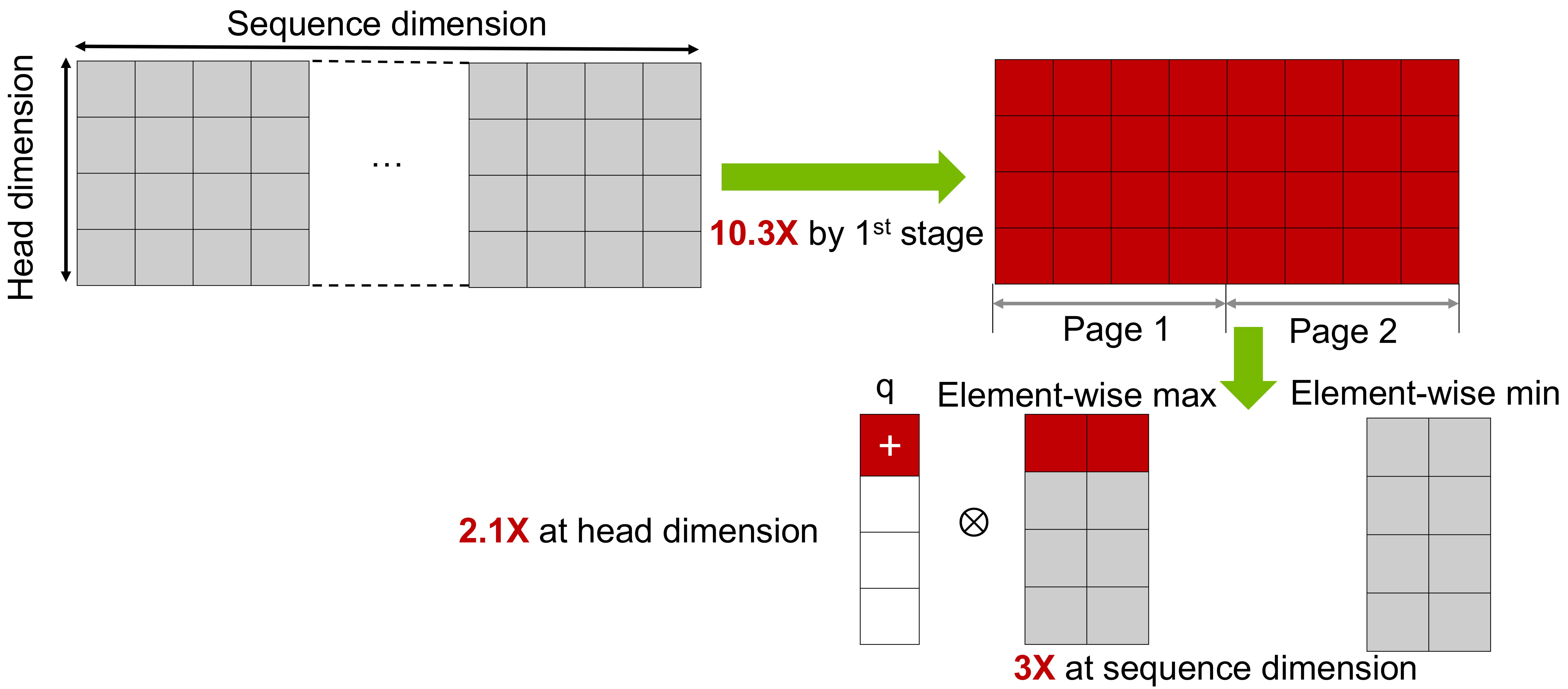
- Challenge in Multi-Turn Decoding
  - Permanent KV eviction underperforms as important tokens can differ across queries.
- RocketKV-MT Solution
  - Retains unselected KV tokens for future turns, but restricts dynamic selection to the filtered set, saving memory traffic without reducing storage.

Adaptive Compression Decomposition

- RocketKV automatically adjusts compression in each stage based on the overall compression target.
- In HSA, compression is further decomposed across head and sequence dimension.
- For compression ratio of *c*, we define a split factor *r, allocating* c^r for the first stage and $c^{(1-r)}$ for the second stage, where $0 \leq r \leq 1$ and $r = min(0.2 + 0.06 * \log_2(c), 0.8)$.
- Example: Compression ratio = 64X

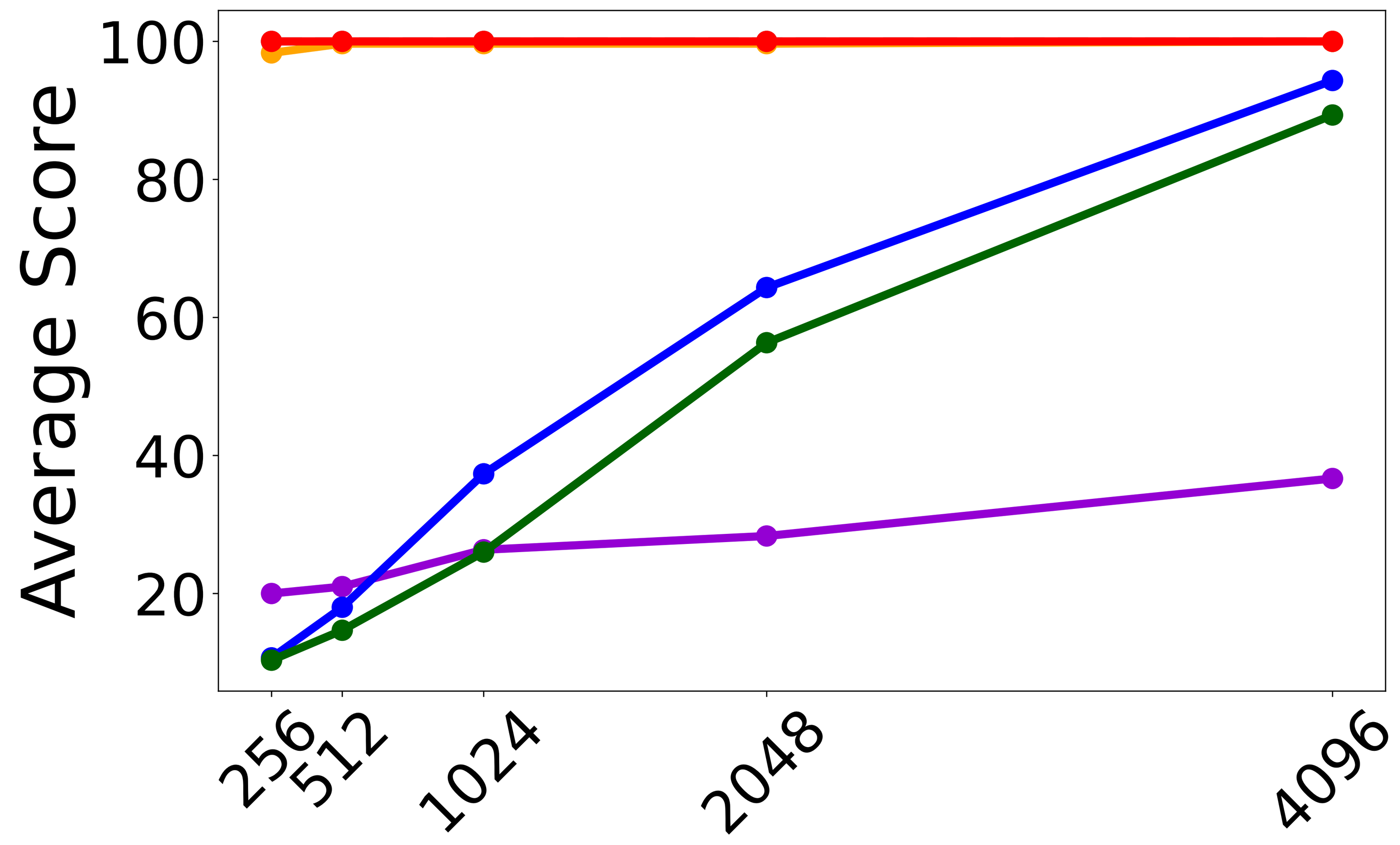
$$\mathbf{r} = 0.2 + 0.06 * \log_2(64) = 0.56$$

$$\Longrightarrow 64^{0.56} = \mathbf{10.3} \text{ and } 64^{(1\text{-}0.56)} = \mathbf{6.2}$$

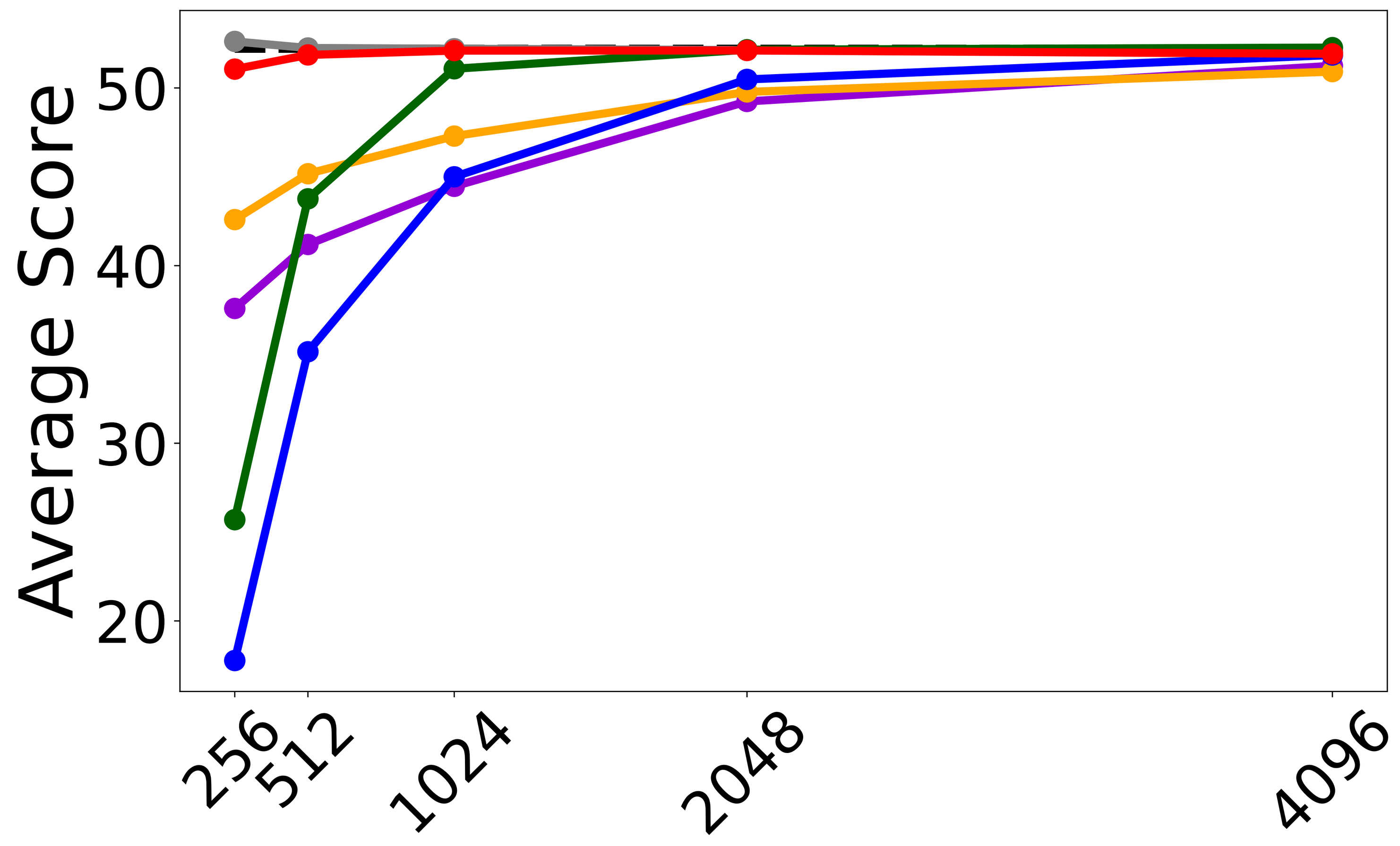


Experimental Results-Accuracy

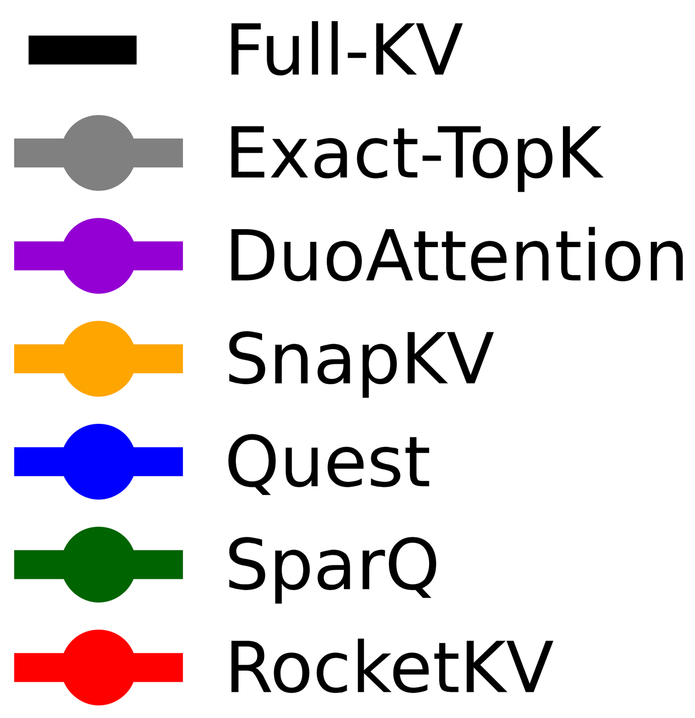
- RocketKV achieves up to **400X** KV compression while maintaining accuracy comparable to full KV cache attention across various models and datasets.
- RocketKV outperforms all SOTA methods, especially in lower token budgets (**Up to 90%**).

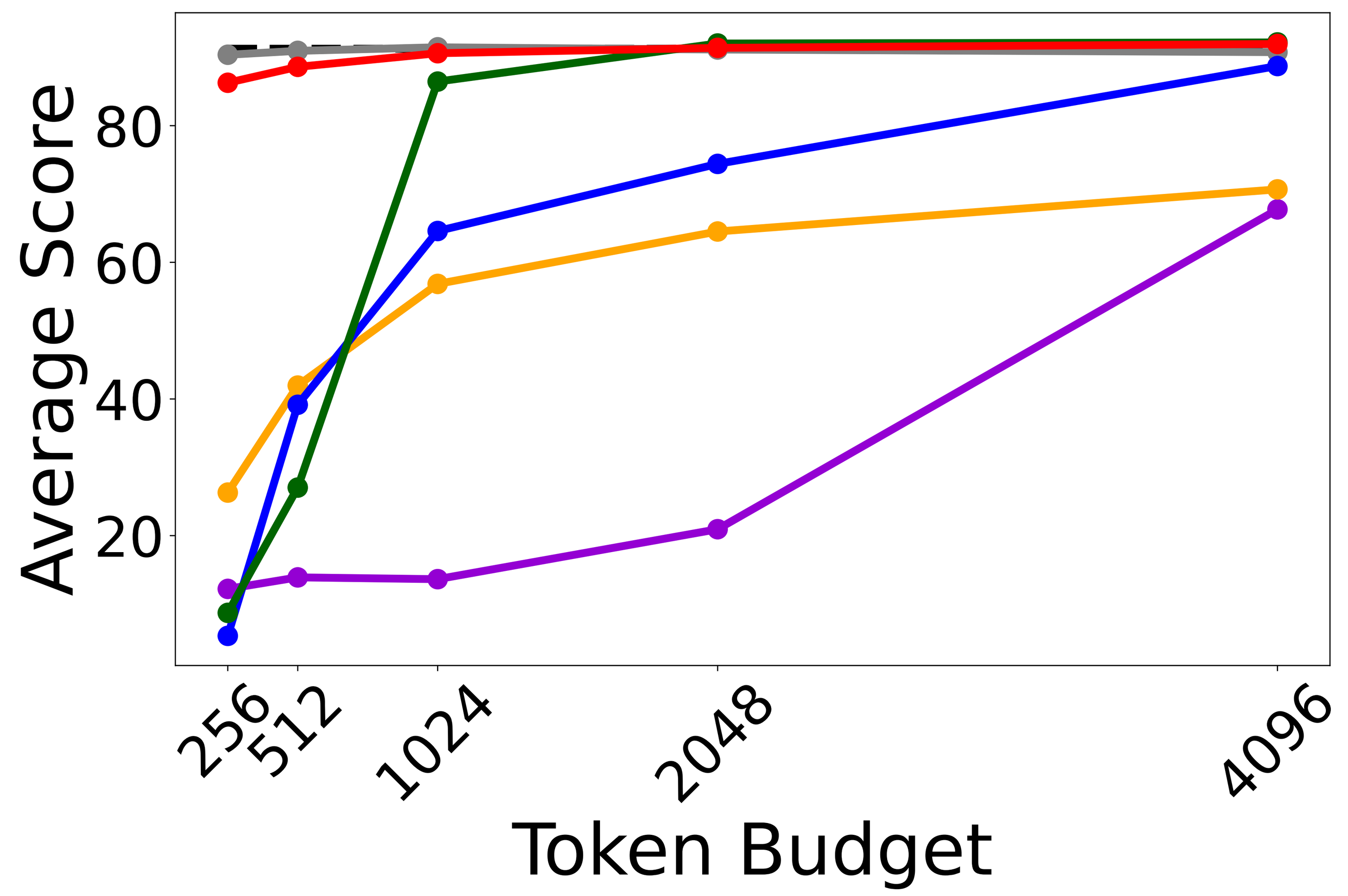


LongBench, Llama3.1-8B-Ins

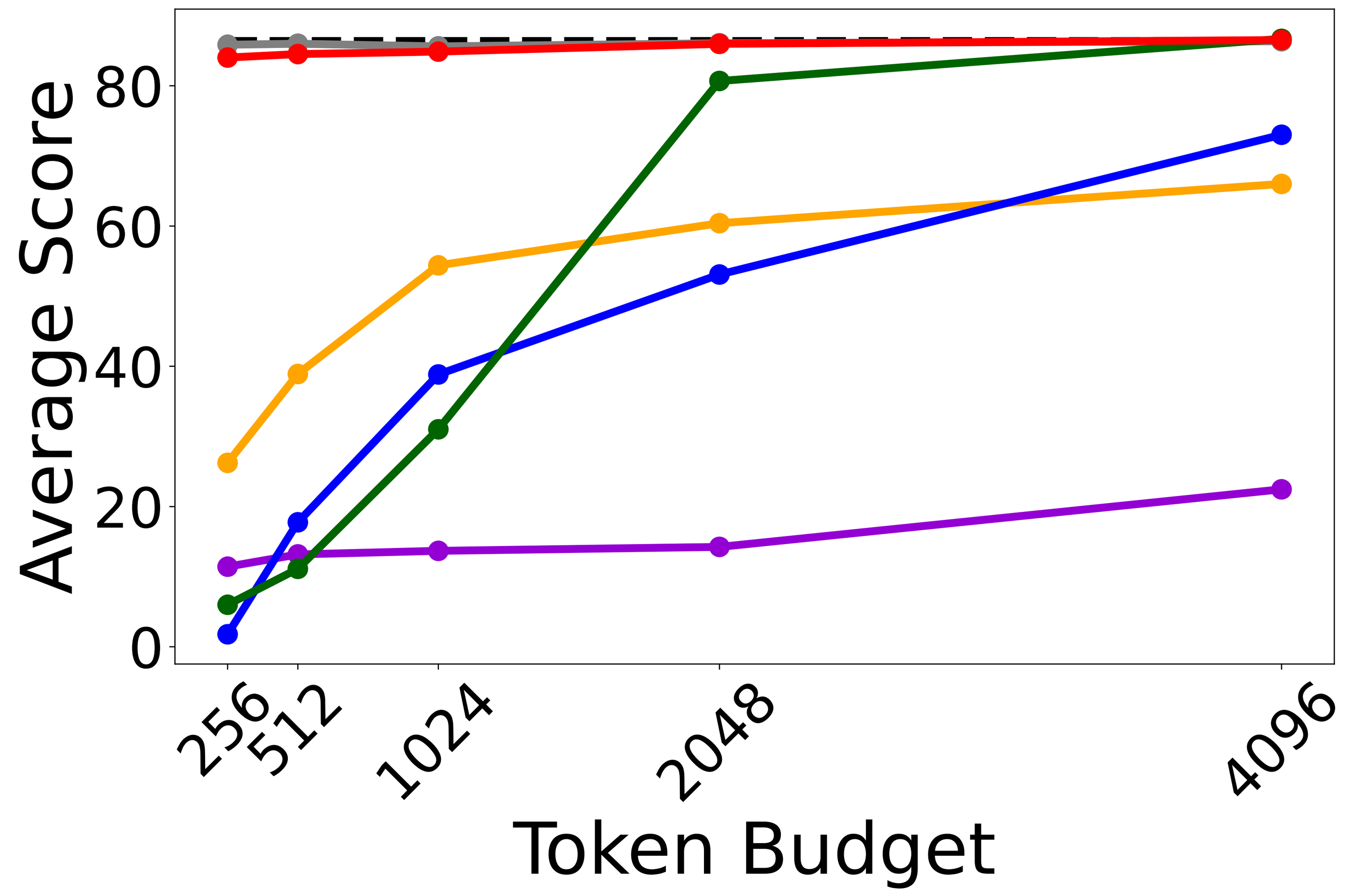


NIAH, Llama3.1-8B-Ins

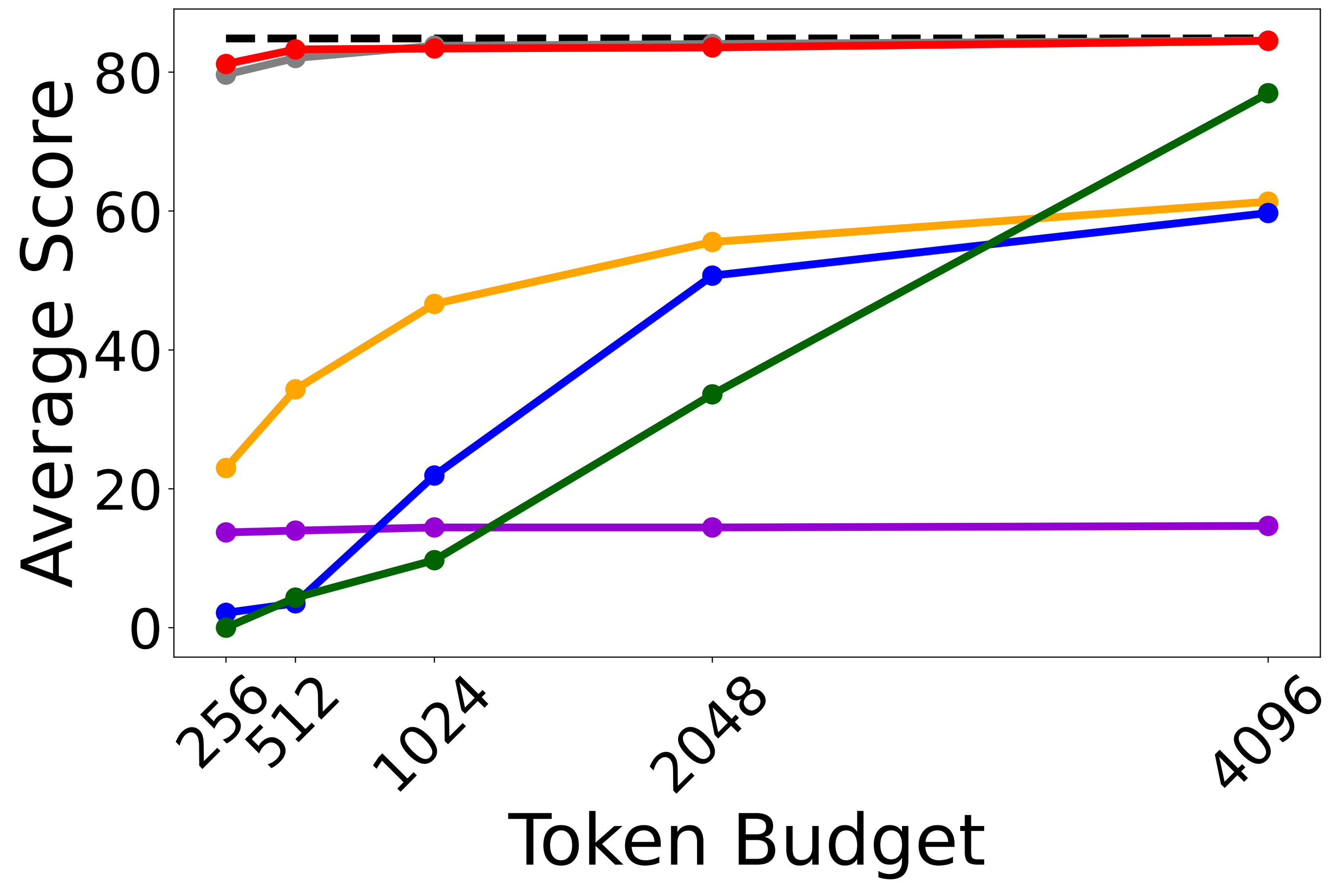




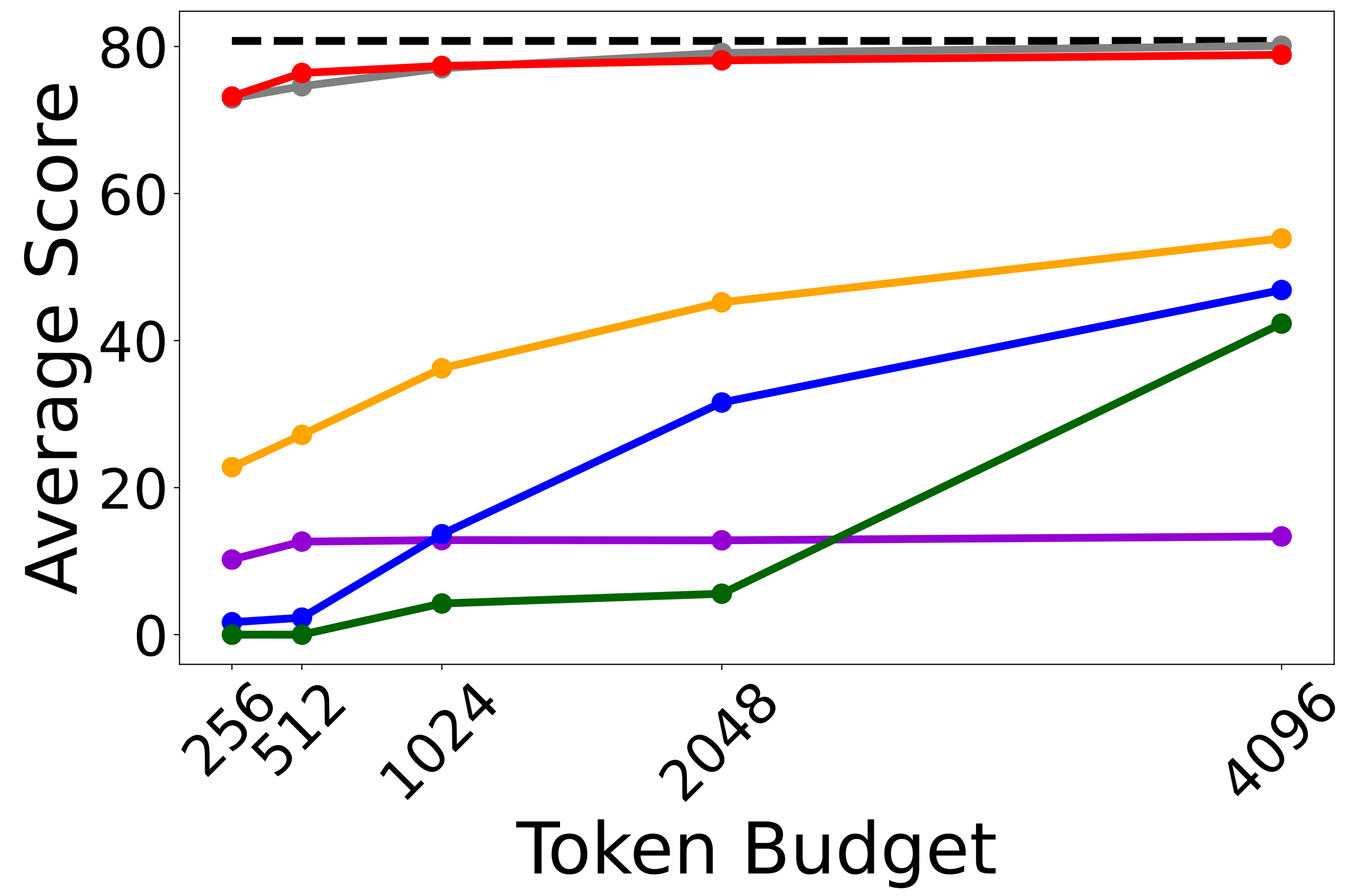
Llama3.1-8B-Ins, SeqLen=16K



Llama3.1-8B-Ins, SeqLen=32K



Llama3.1-8B-Ins, SeqLen=64K

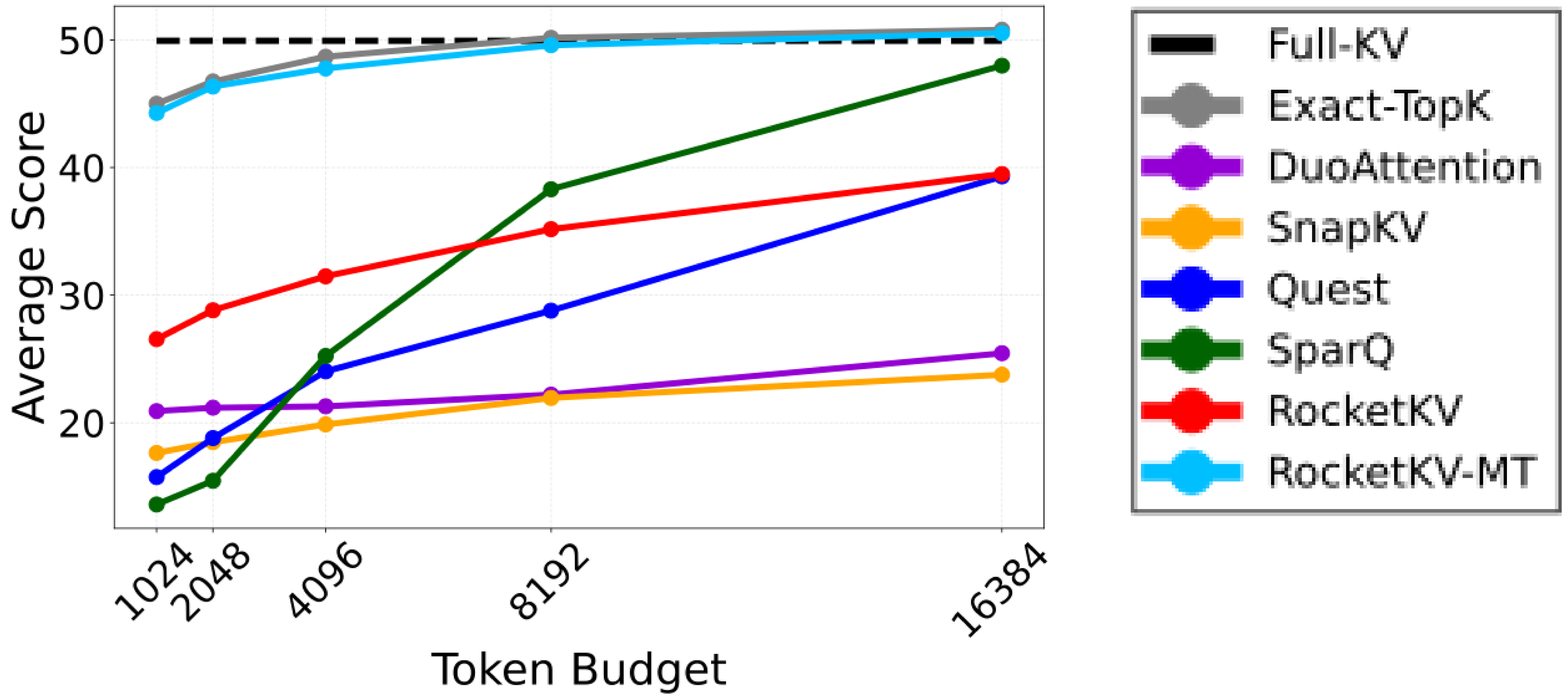


Llama3.1-8B-Ins, SeqLen=96K

NVIDIA

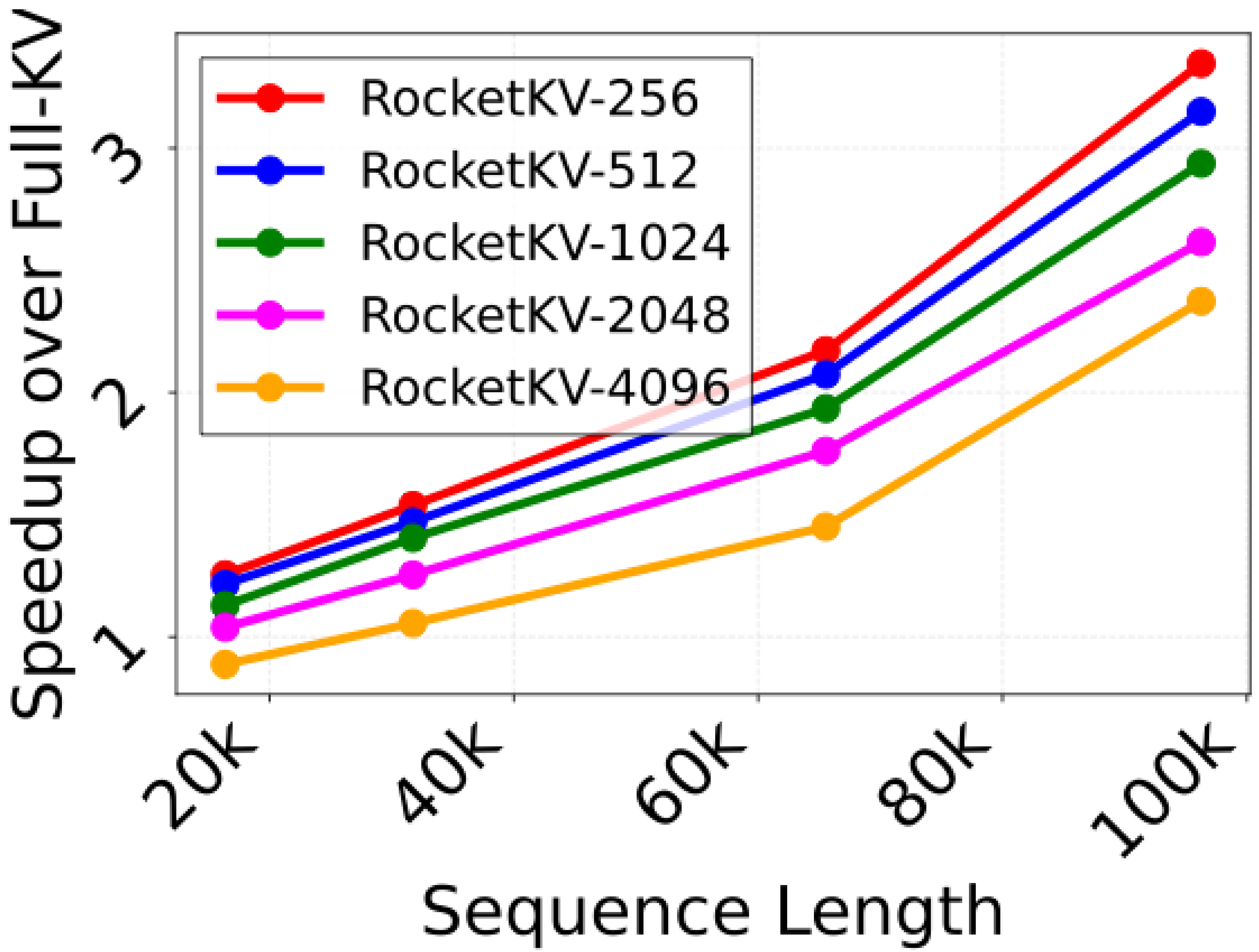
Experimental Results-Accuracy (Multi-Turn)

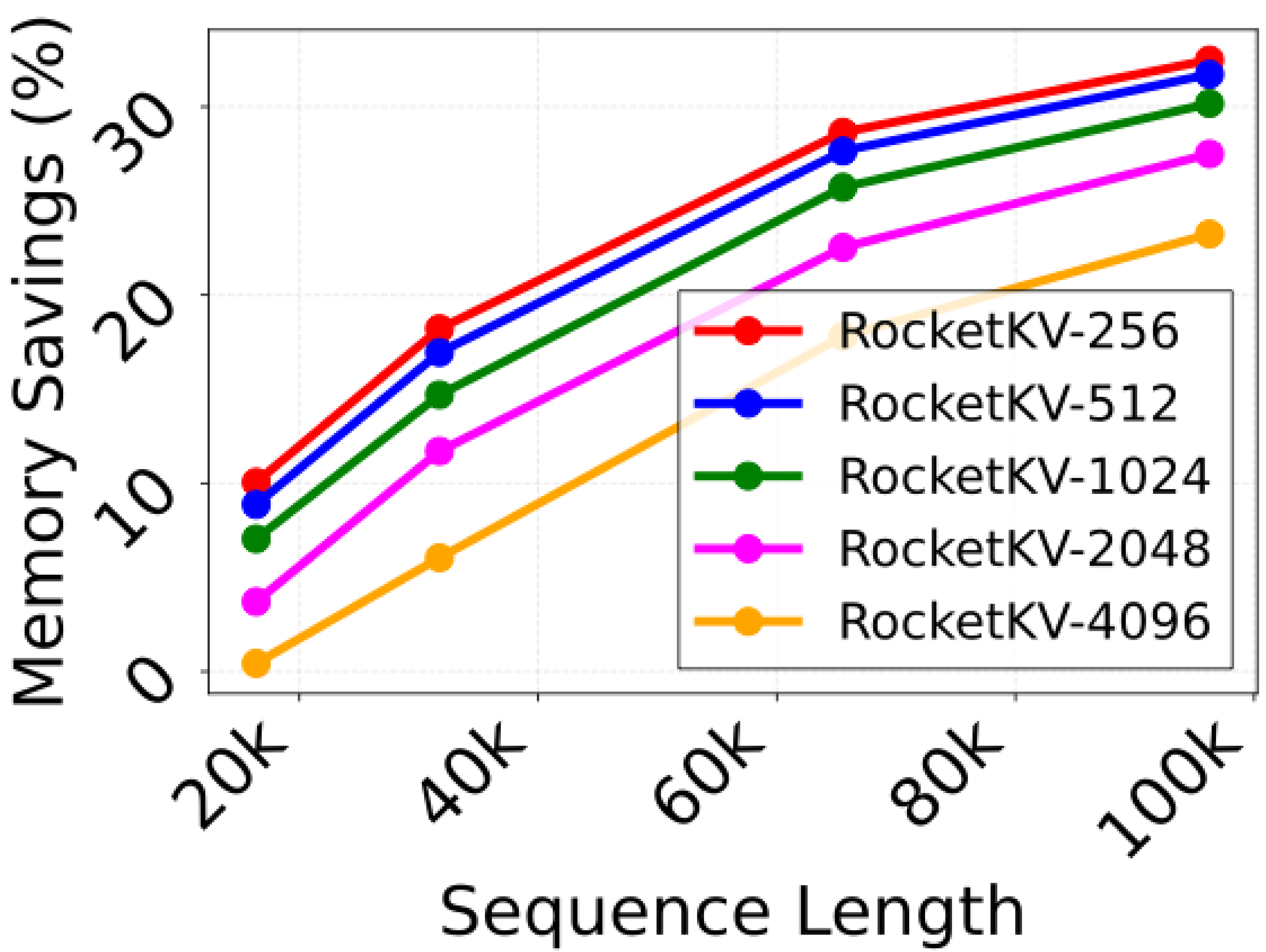
- RocketKV underperforms Exact-TopK due to early KV evictions, but RocketKV-MT retains important tokens and matches Exact-TopK accuracy across all budgets.



Experimental Results-Efficiency

- By running on A100, RocketKV delivers up to **3.7X end-to-end speedup** and **32.6% peak memory reduction** during decoding phase.





Conclusion

- **Training-Free Compression:** RocketKV reduces KV cache size without retraining, targeting decode-phase bottlenecks through two stage compression.

- **High Efficiency:** RocketKV achieves up to **400X** compression, and end-to-end **3.7X** speedup, and **32.6%** memory savings with minimal accuracy loss.

- **Multi-Turn Support**: RocketKV-MT extends the proposed method to multi-turn tasks.