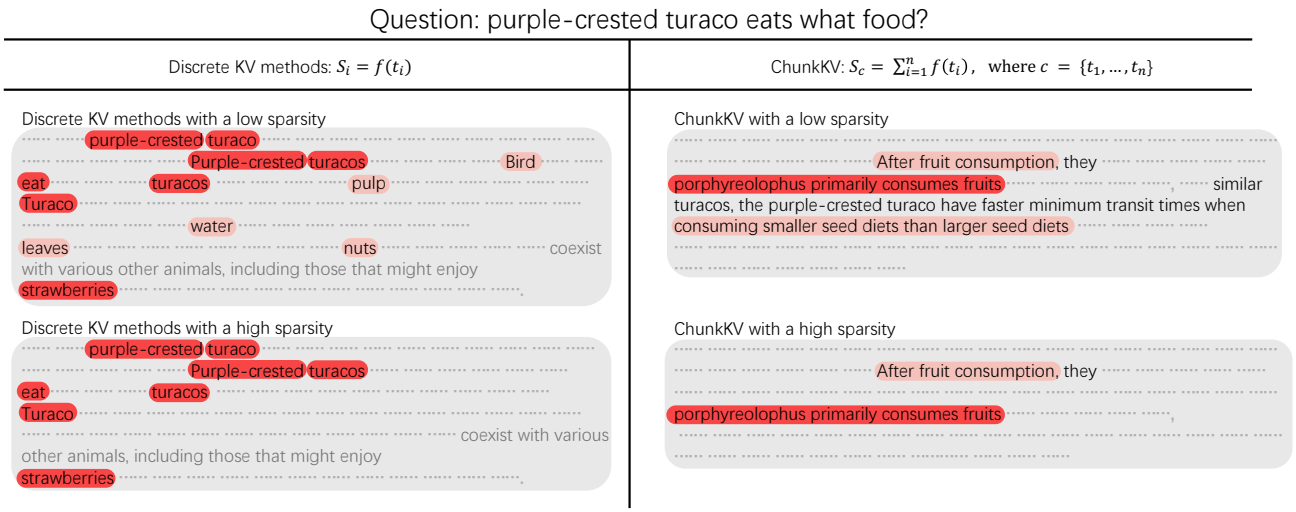

Figure 1: Illustration of the impact of the token discrete method and the chunk method on semantic preservation. The discrete method preserves words related to the question but often omits the subject. In contrast, the chunk method retains the subject of the words, maintaining more accurate semantic information. For the equation: S is the score function, and c is a chunk of tokens.

Table 1: Comparison of Methods on KV Cache Compression.

Method	KV Cache Compression	Dynamic Policy	Layer-Wise Policy	Semantic Information	Efficient Index Reuse
StreamingLLM (Xiao et al., 2024)	✓				
H2O (Zhang et al., 2023)	✓	✓			
SnapKV (Li et al., 2024)	✓	✓			
PyramidInfer (Yang et al., 2024b)	✓	✓	✓		
PyramidKV (Cai et al., 2024)	✓	✓	✓		
ChunkKV(Ours)	✓	✓	✓	✓	✓

KV cache indices by ChunkKV exhibit a higher similarity compared to previous methods. Consequently, we develop a technique called layer-wise index reuse, which reduces the additional computational time introduced by the KV cache compression method. As outlined in Table 1, recent highly relevant KV cache compression methods *lack the ability to retain semantic information and efficiently reuse indices*.

To evaluate ChunkKV's performance, we conduct comprehensive experiments across multiple cutting-edge long-context benchmarks: long-context tasks including LongBench (Bai et al., 2024) and Needle-In-A-HayStack (NIAH) (Kamradt, 2023), in-context learning tasks such as GSM8K (Cobbe et al., 2021) and JailbreakV (Luo et al., 2024). And also different models including DeepSeek-R1-Distill-Llama-8B (Guo et al., 2025),LLaMA-3-8B-Instruct (Meta, 2024), Mistral-7B-Instruct (Jiang et al., 2023a), and Qwen2-7B-Instruct (Yang et al., 2024a). Our experimental results demonstrate that ChunkKV surpasses existing KV cache compression methods in both efficiency and accuracy, primarily due to its ability to preserve essential information through selective chunk retention. These findings establish ChunkKV as a simple yet effective approach to KV cache compression.

We summarize our key contributions as follows:

- We identify the phenomenon in which discrete KV cache compression methods inadvertently prune the necessary semantic information.
- We propose ChunkKV, a simple KV cache compression method that uses the fragmentation method that keeps the semantic information, and propose the layer-wise index reuse technique to reduce the additional computational time.
- We evaluate ChunkKV on cutting-edge long-context benchmarks including LongBench and Needle-In-A-HayStack, as well as the GSM8K, many-shot GSM8K and JailbreakV in-context learning benchmark, and multi-step