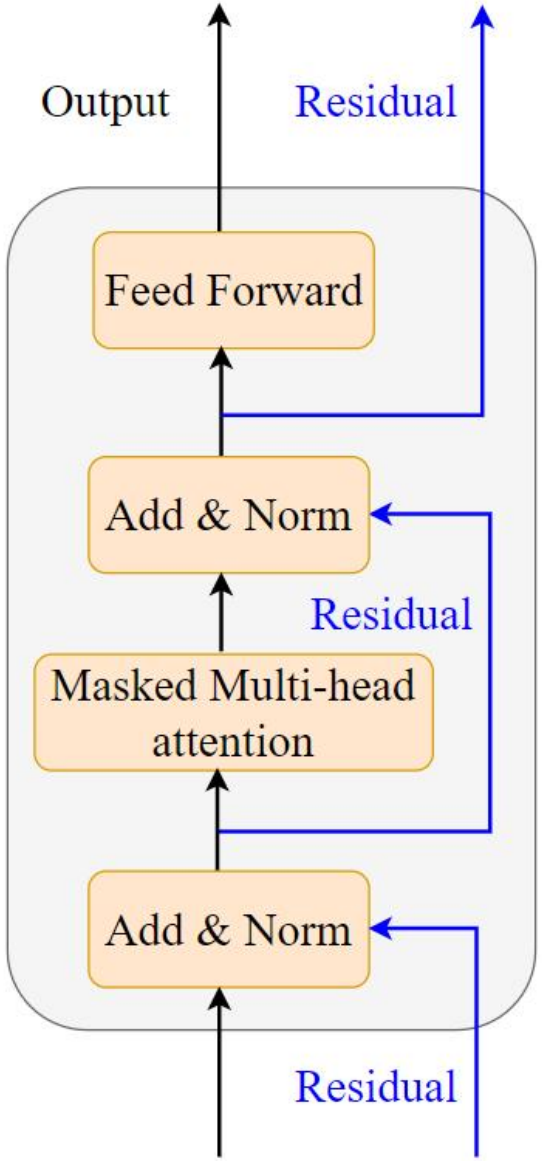
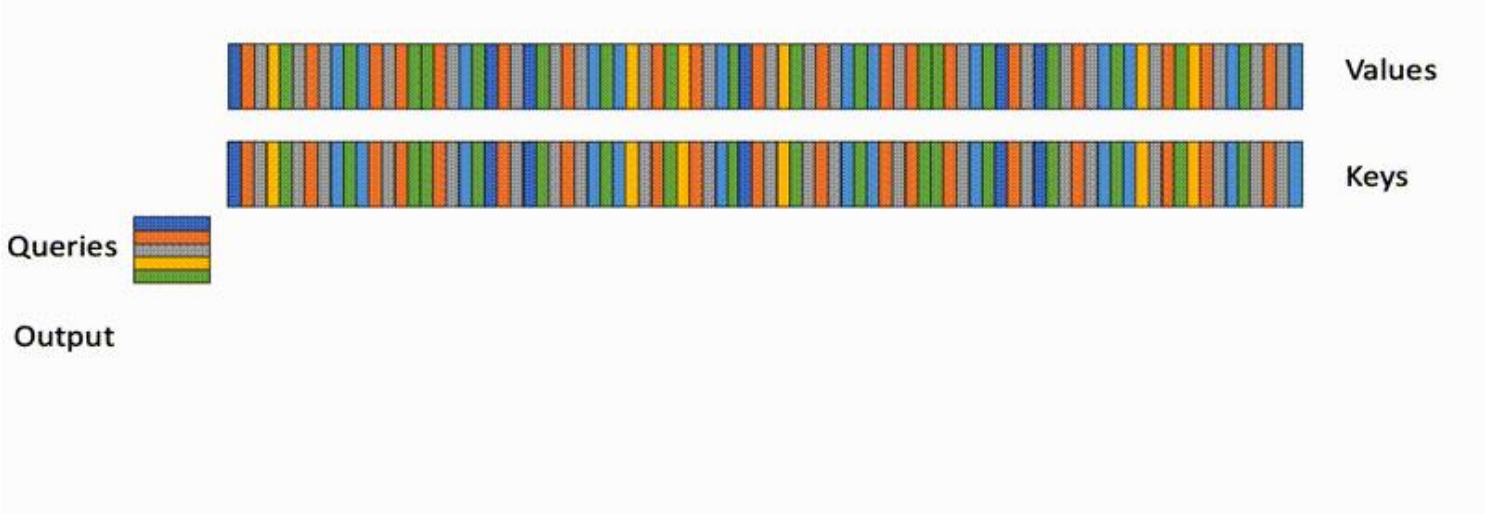
CrossKV : Reuse KV Cache Across Requests

Demystify the Decoder Layer and Self Attention (LLama2-7b-chat)

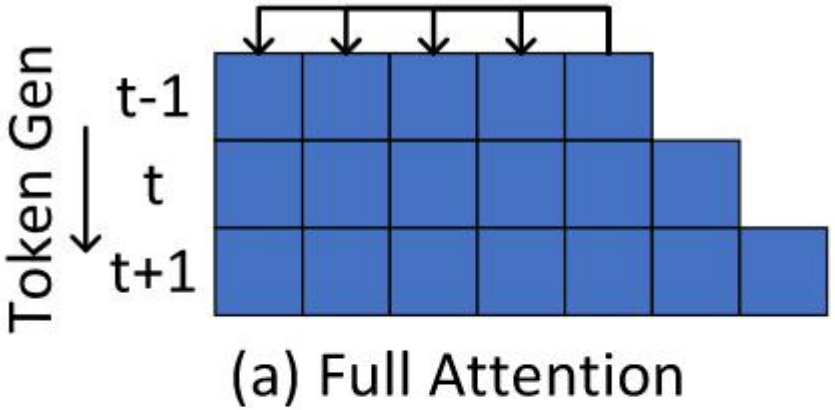


$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V$$

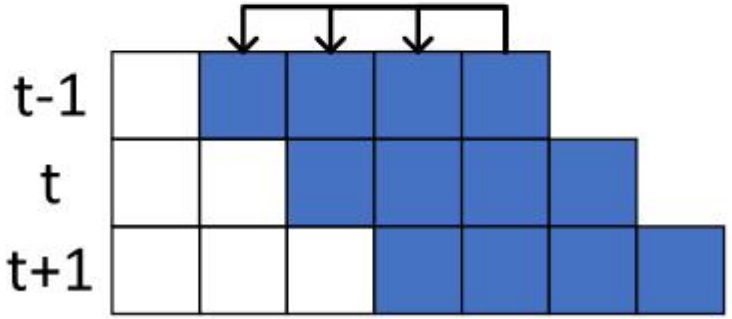


Demystify the Decoder Layer and Self Attention (LLama2-7b-chat)

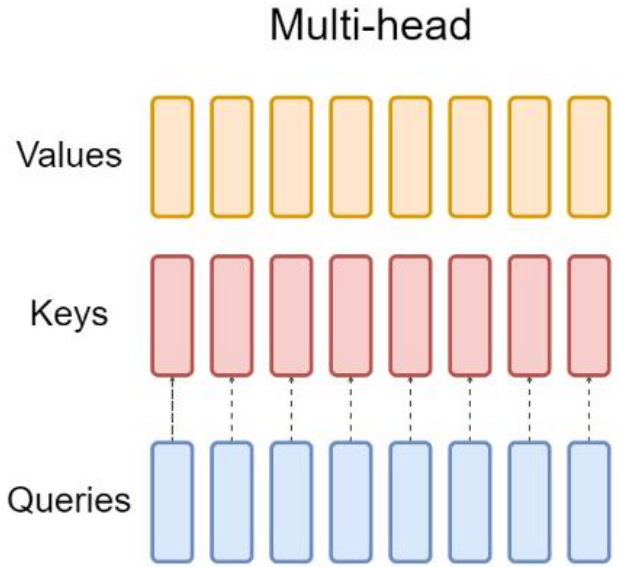
How does LLM have the ability to memorize context ?

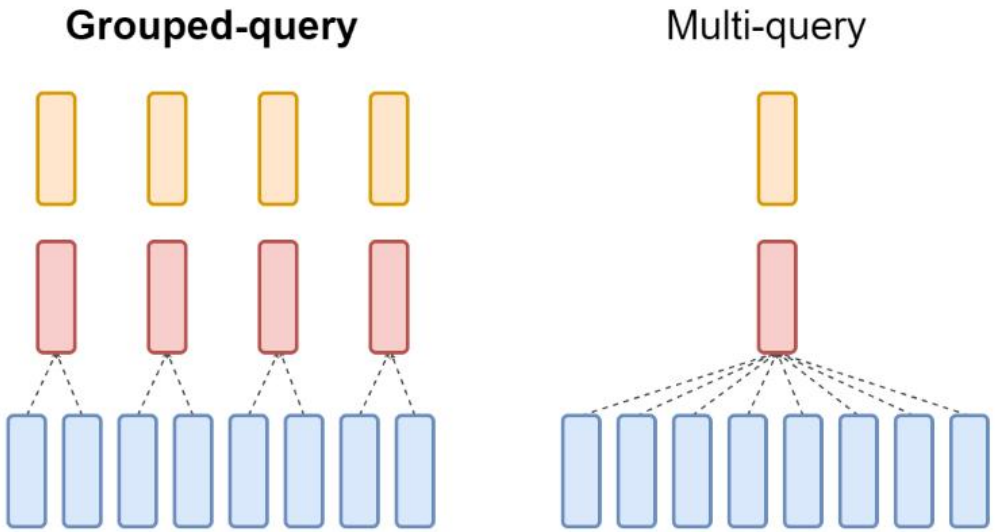


(a) Full Attention



(b) Window Attention





Demystify the Procedure of LLM Inference——Prefill (LLama2-7b-chat)

1. Prompt: “I love SYSU”

2. Chat Template: “[INST] I love SYSU [/INST]”

3. Token IDs: [1,518,25580,29962,306,5360,28962,14605,518,29914,25580,29962]

4. Embedding: ==> Input Tensor(12,4096)

5. LLama Model(32 Decoder Layers) : ==> Output Tensor(12,4096)

5. Logits: Output[-1](1,4096) * Embedding.t(4096,32000) ==> Tensor(1,32000)

6. Sample: Select one tokens based on Logits (Temperature, Top-P, Top-K) ==> [29871]

Demystify the Procedure of LLM Inference——Decode (LLama2-7b-chat)

1. Token ID: [29871]

2. Embedding: Input Tensor(1,4096)

3. LLama Model(32 Decoder Layers) : ==> Output Tensor(1,4096)

4. Logits: Output(1,4096) * Embedding.t(4096,32000) ==> Tensor(1,32000)

5. Sample: Select one tokens based on Logits (Temperature, Top-P, Top-K)

What's the effect of KV cache?——Reduce redundant computation.

KV cache across requests

The intention of KV cache is used to accelerate the execution of one inference request.

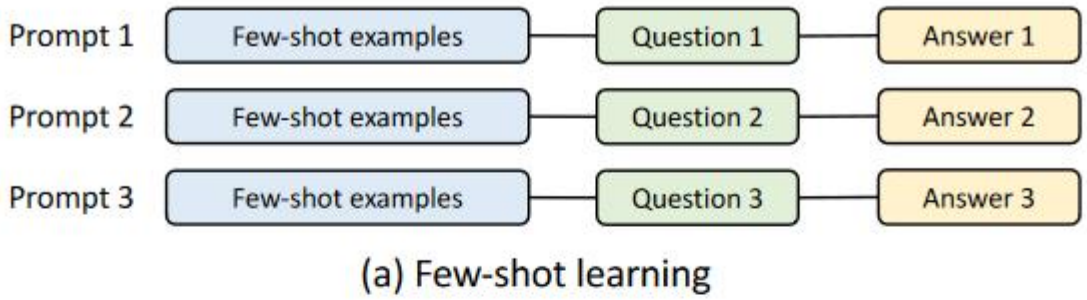
In what situation, KV cache can be reused accross requests?

Rethink the computing procedure of self attention: Query will only attend current token's and previous tokens' Key Value.

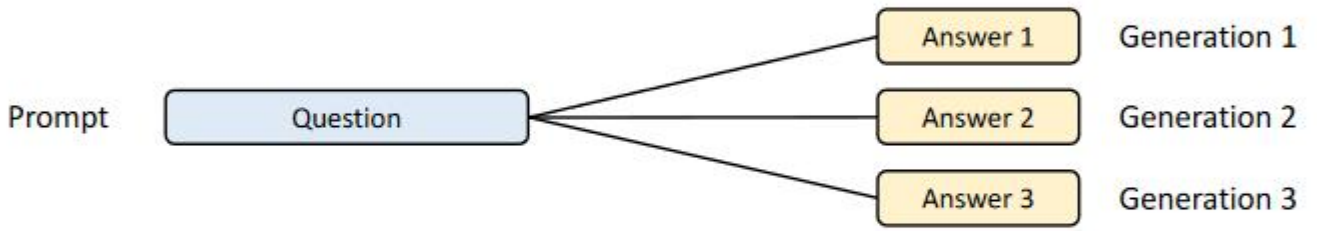
So if two prompts have the common prefix, they will share the same KV cache.

What is the use case of KV cache across requests ?

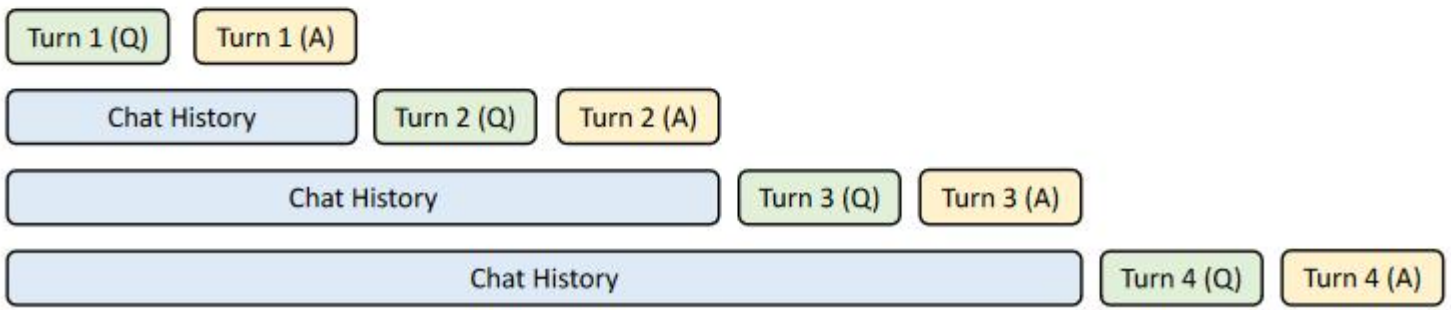
The use case of KV cache



(a) Few-shot learning

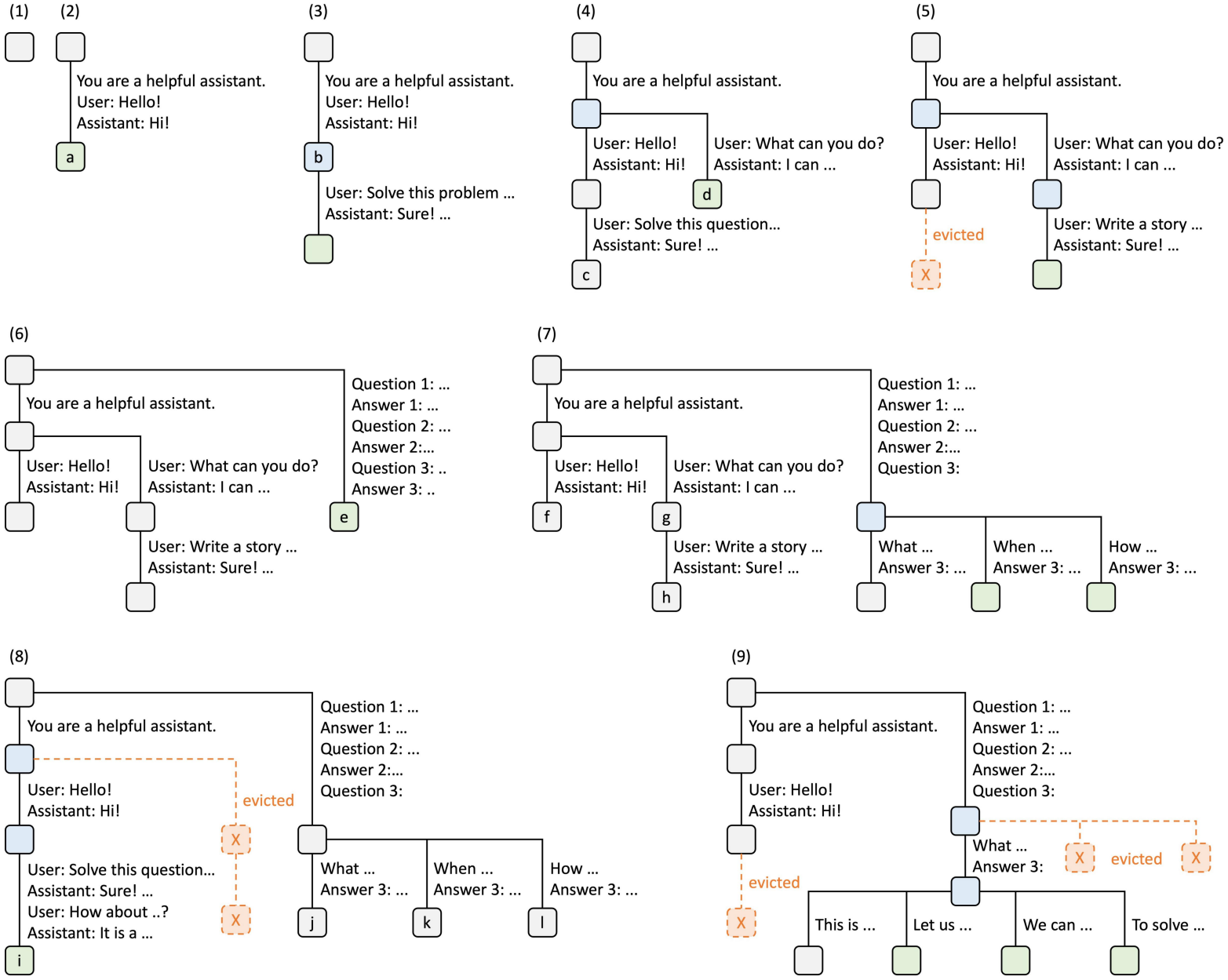


(b) Self-consistency



(c) Multi-turn chat

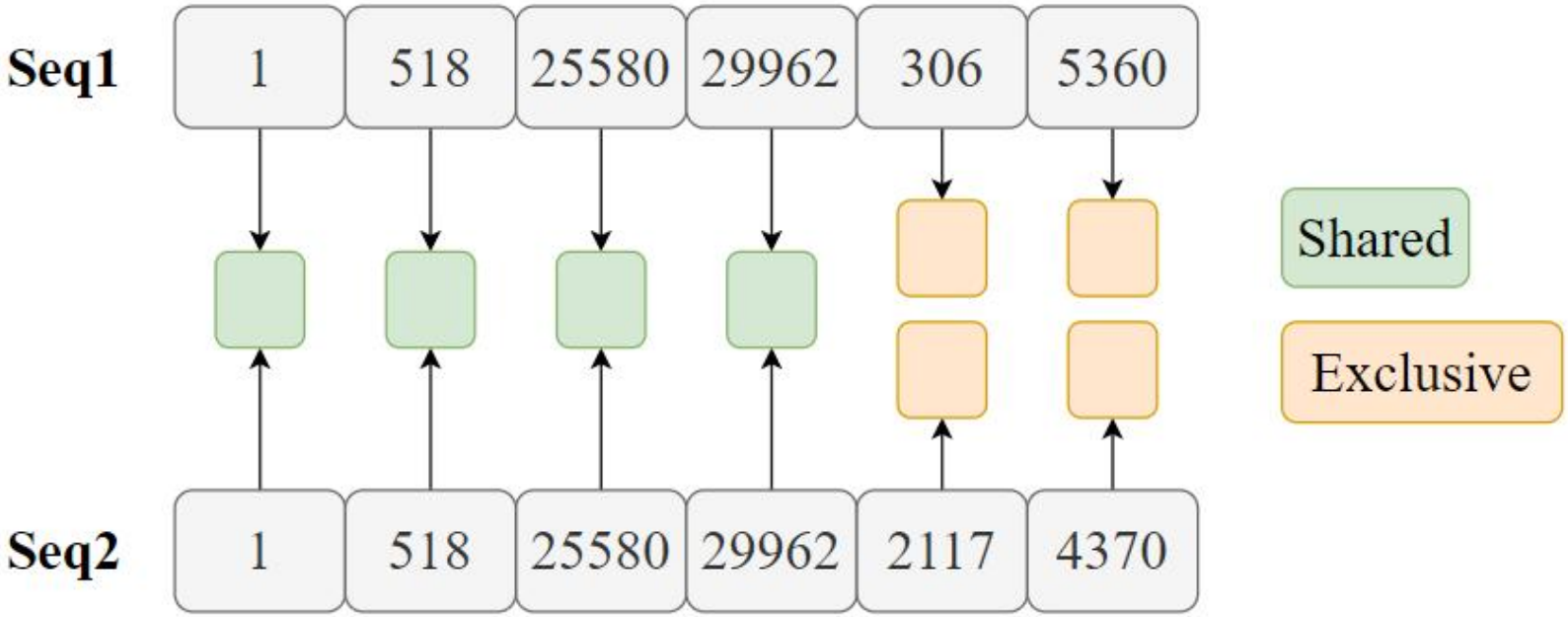
Efficiently Programming Large Language Models using SGLang



VLLM —— Automatic Prefix Caching

Before: Logical block table => Physical block table

After: Logical block table => Hash table => Physical block table



The same hash indicate that they share the common prefix, so the KV cache can be reused for them.

https://github.com/vllm-project/vllm/pull/2762

END