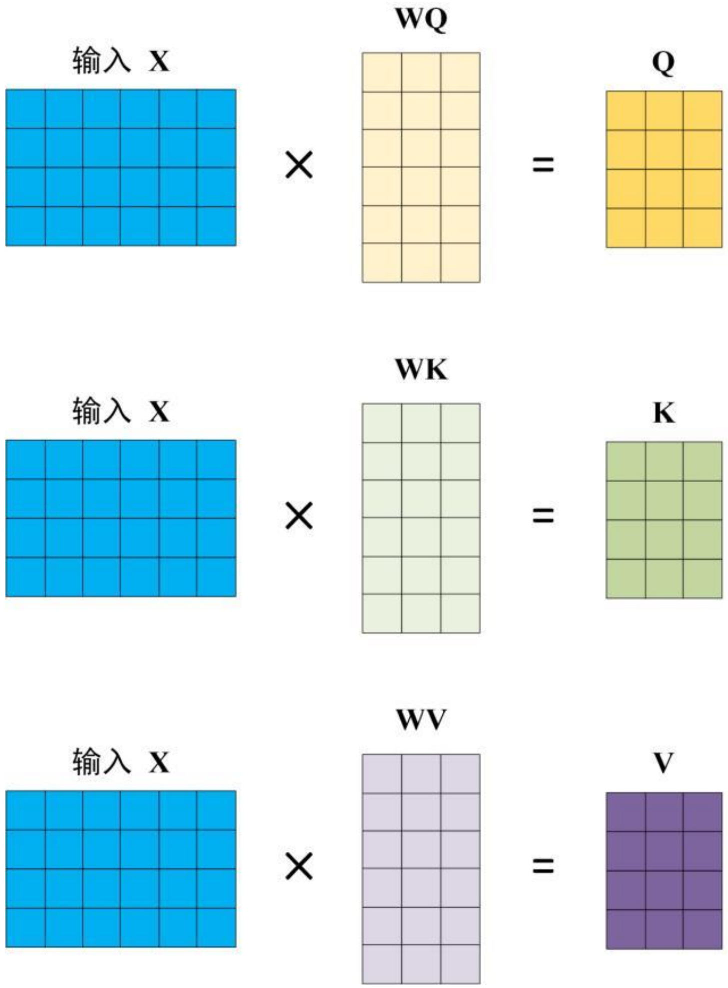
FlashAttention

Kun Yuan (袁 坤)

Center for Machine Learning Research @ Peking University

Self-Attention



$$Q = XW_Q \in \mathbb{R}^{N \times d}$$

N is the sequence length

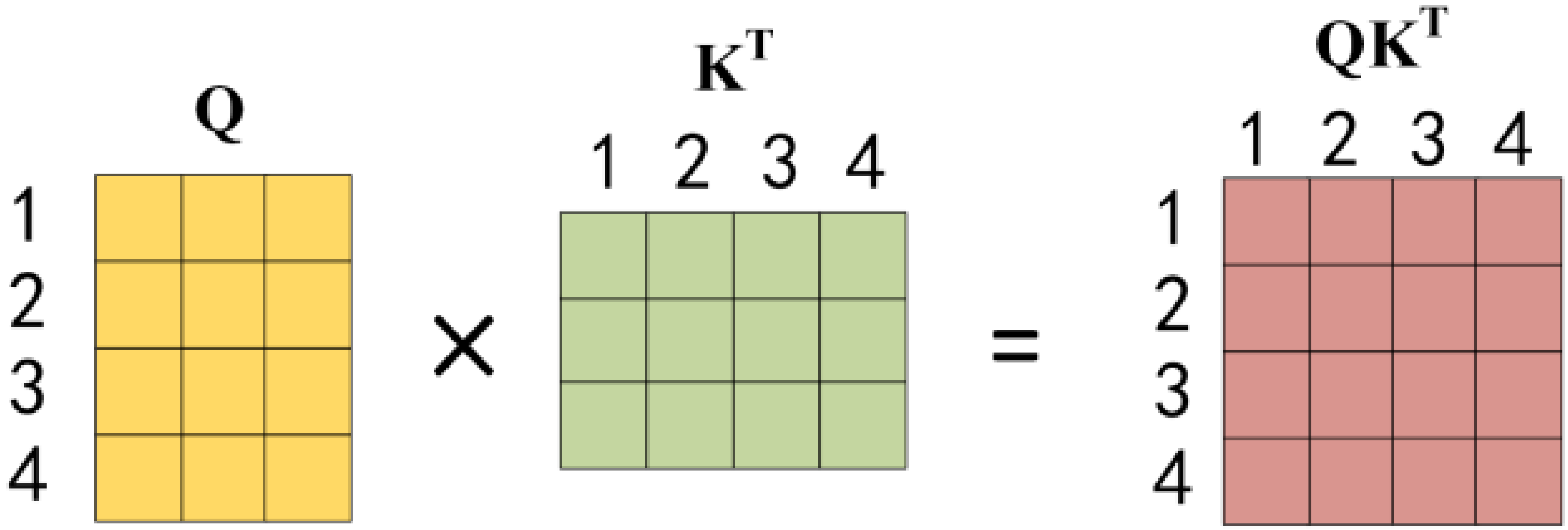
d is the embedding dimension

$$K = XW_K \in \mathbb{R}^{N \times d}$$

$$V = XW_V \in \mathbb{R}^{N \times d}$$

Self-Attention

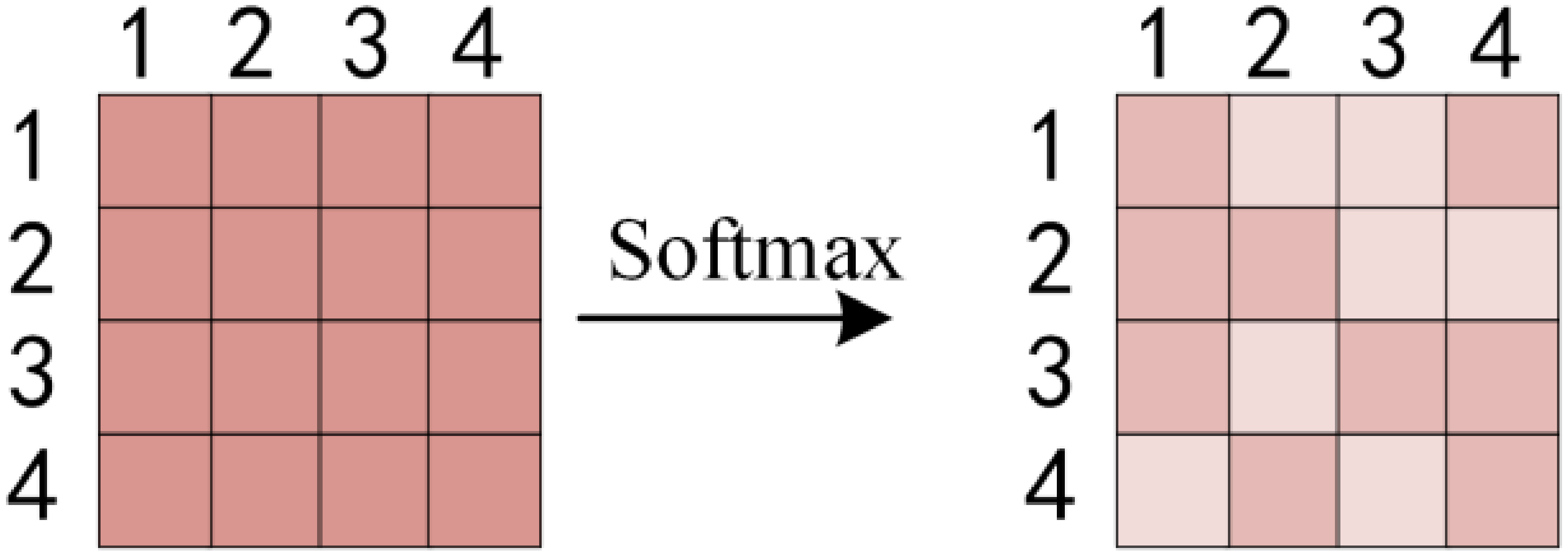
$$S = QK^\top \in \mathbb{R}^{N \times N}$$



Self-Attention

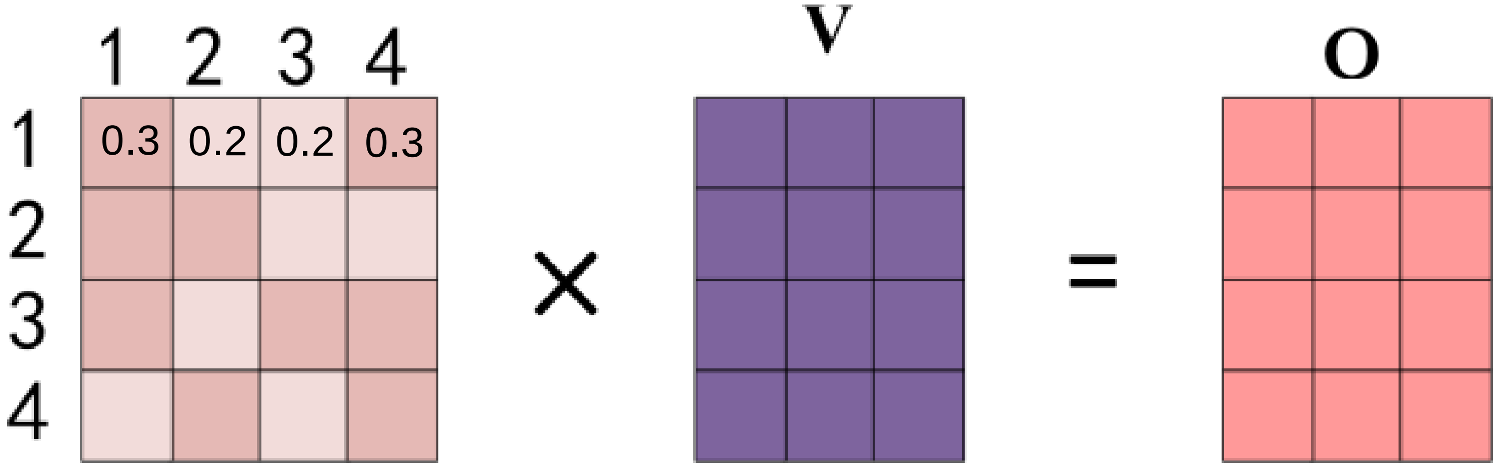
$$P = \mathrm{softmax}(S) \in \mathbb{R}^{N \times N}$$

(we ignore the scaling for simplicity)



Self-Attention

$$O = PV \in \mathbb{R}^{N \times d}$$



FLOPS

北京大学 PEKING UNIVERSITY

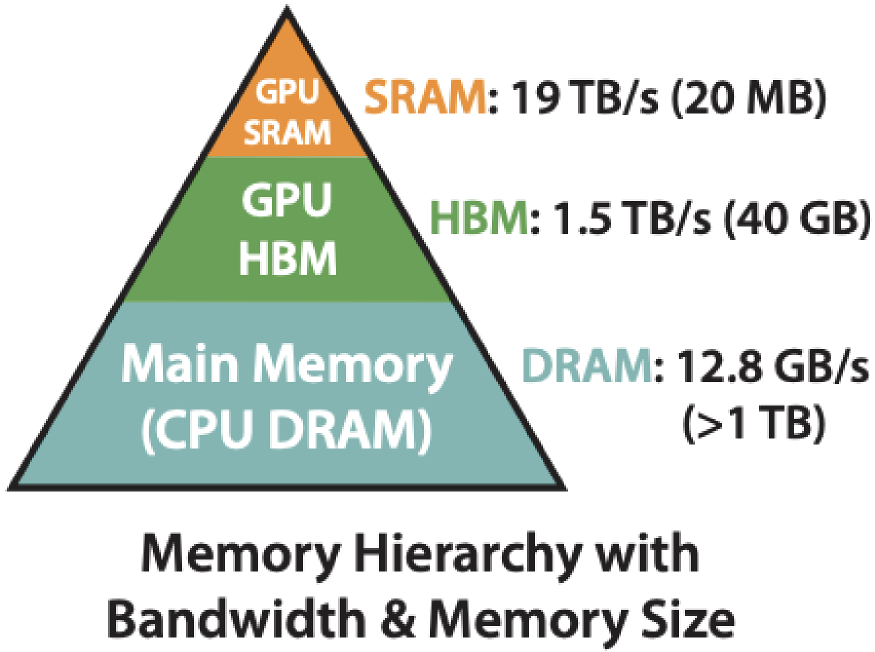
- The above attention process incurs $O(N^2d)$ FLOPS computation complexity

- Increases quadratically fast with sequence length N

- Various methods have been developed to reduce $O(N^2)$ to $O(N)$. These methods are not exact attention, and they typically fail to achieve remarkable acceleration

- The fundamental reason is that they cannot reduce **M**emory **A**ccess **C**ost (MAC)

Memory in GPU



Fast but small

Large but slow

Execution Model in GPU. Load inputs from HBM to SRAM, computes, then writes outputs to HBM.

Since HBM is slow, MAC is primarily composed of **HBM reads and writes**

[FlashAttention: Fast and Memory-Efficient Exact Attention with IO-Awareness]

MAC in standard attention implementation

北京大学 PEKING UNIVERSITY

Algorithm 0 Standard Attention Implementation

Require: Matrices $\mathbf{Q}, \mathbf{K}, \mathbf{V} \in \mathbb{R}^{N \times d}$ in HBM.

1: Load $\mathbf{Q}, \mathbf{K}$ by blocks from HBM, compute $\mathbf{S} = \mathbf{Q}\mathbf{K}^\top$, write $\mathbf{S}$ to HBM.
2: Read $\mathbf{S}$ from HBM, compute $\mathbf{P} = \mathrm{softmax}(\mathbf{S})$, write $\mathbf{P}$ to HBM.
3: Load $\mathbf{P}$ and $\mathbf{V}$ by blocks from HBM, compute $\mathbf{O} = \mathbf{P}\mathbf{V}$, write $\mathbf{O}$ to HBM.
4: Return $\mathbf{O}$.

MAC cost is

$$4N^2 + 4dN$$

Operation	MAC
Load Q and K	$2dN$
Write S	N^2
Read S	N^2
Write P	N^2
Load Q and V	$N^2 + dN$
Write O	dN

MAC consumes significant wall-clock time in transformer

- **Compute-bound operator:** computing time > accessing HBM time

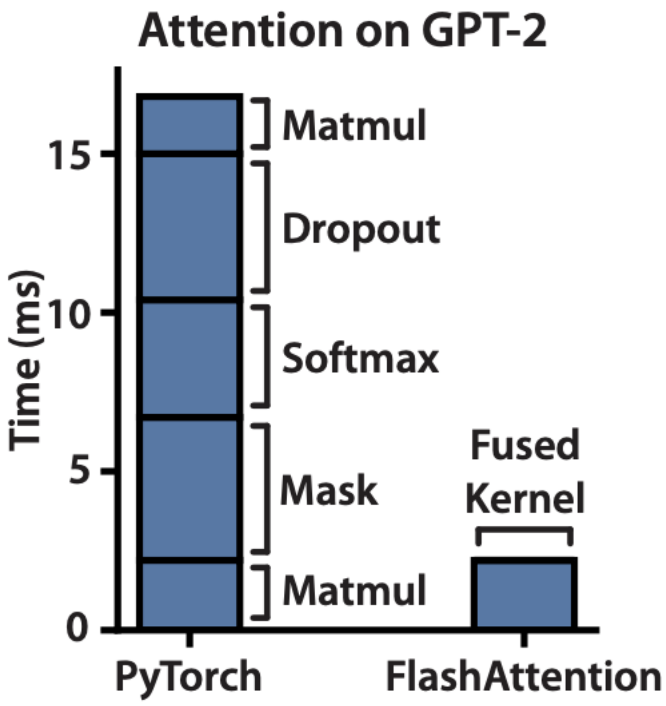
  Matrix multiplication; convolution

- **Memory-bound operator:** accessing HBM time > computing time

  Element-wise operator (activation, dropout); reduction (sum, softmax)

- Transformer includes many memory-bound operators

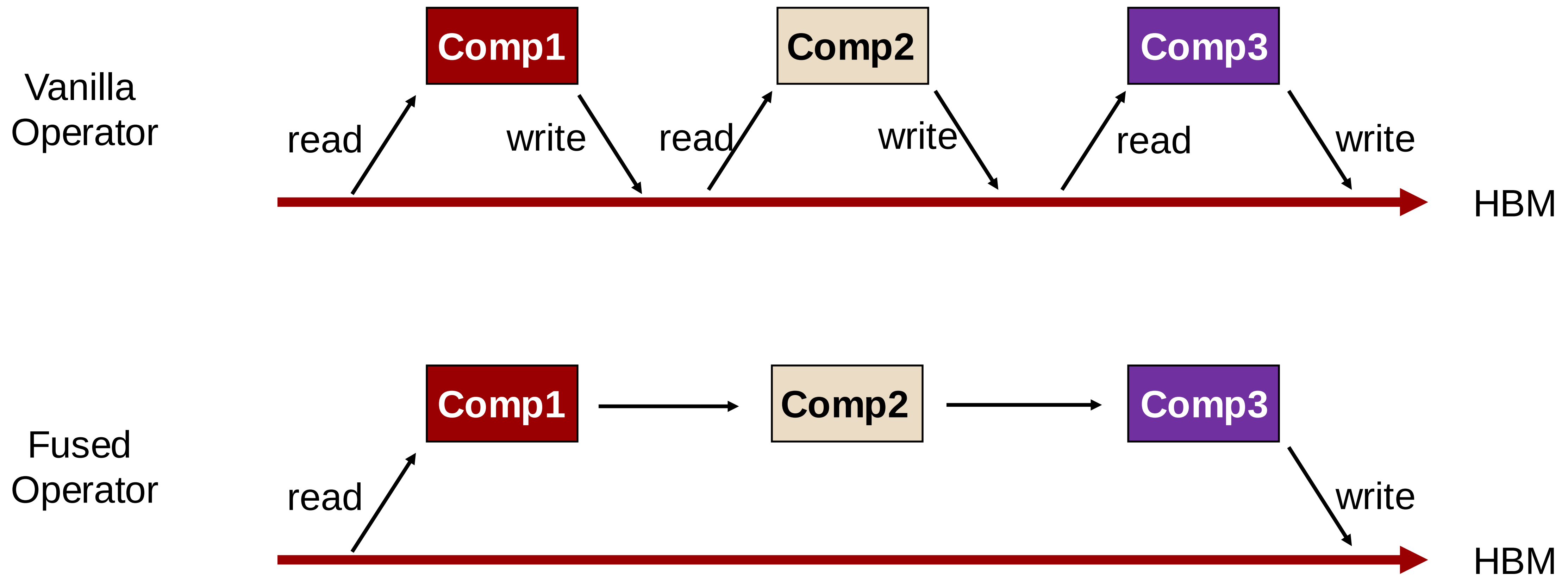
- Reducing MAC cost can significantly accelerate attention



FlashAttention: Fast and Memory-Efficient Exact Attention with IO-Awareness

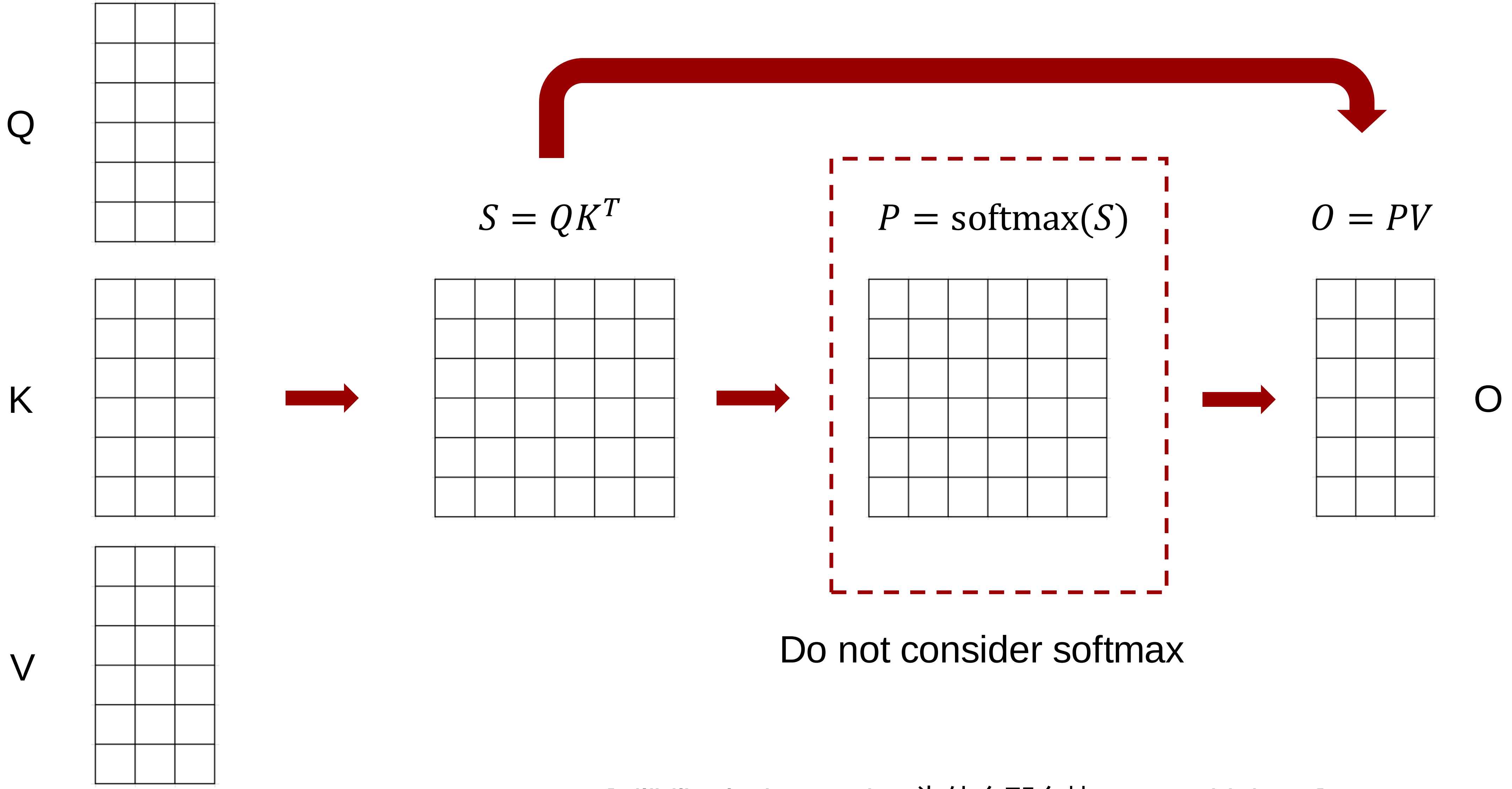
Tri Dao[†], Daniel Y. Fu [†], Stefano Ermon [†], Atri Rudra [‡], Christopher Ré [†]
[†] Department of Computer Science, Stanford University
[‡] Department of Computer Science and Engineering, University at Buffalo, SUNY
{trid,danfu}@stanford.edu, ermon@stanford.edu, atri@buffalo.edu, chrismre@cs.stanford.edu

Core idea in FlashAttention: Kernal fusion



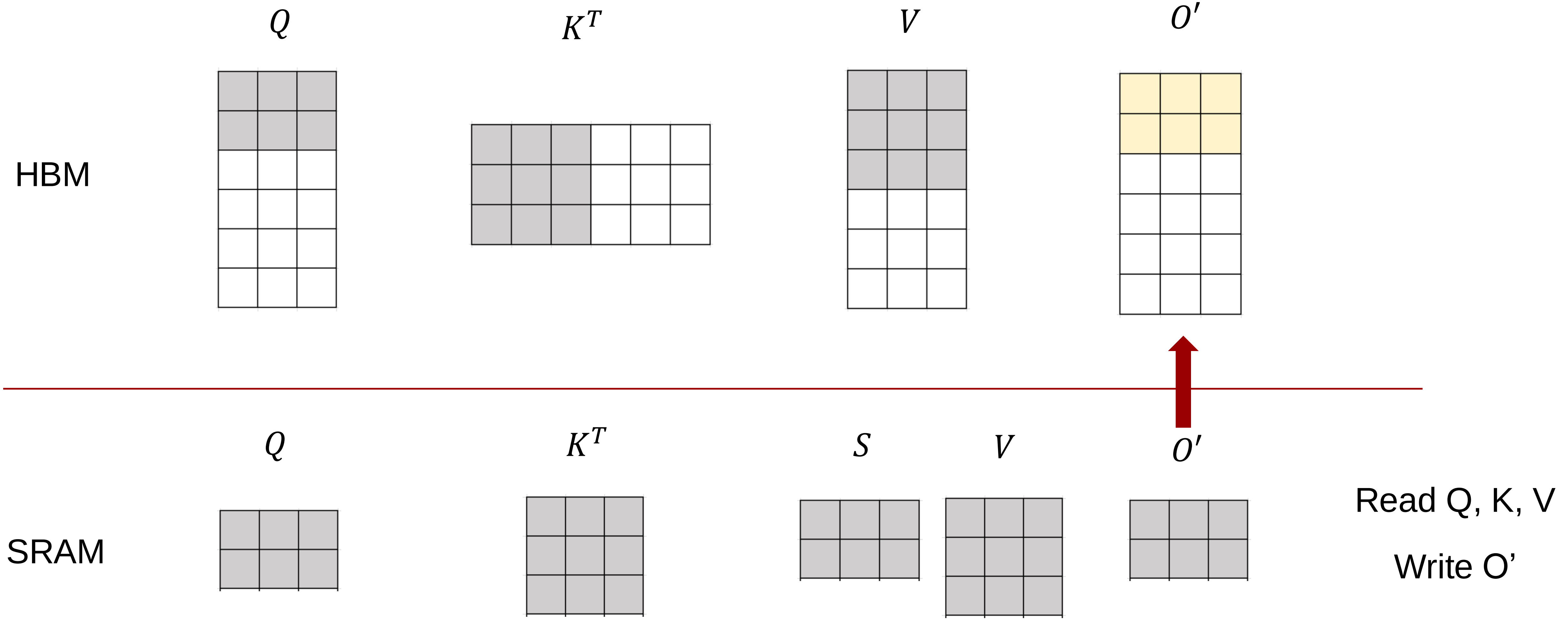
Reduce MAC significantly

A simplified attention without softmax



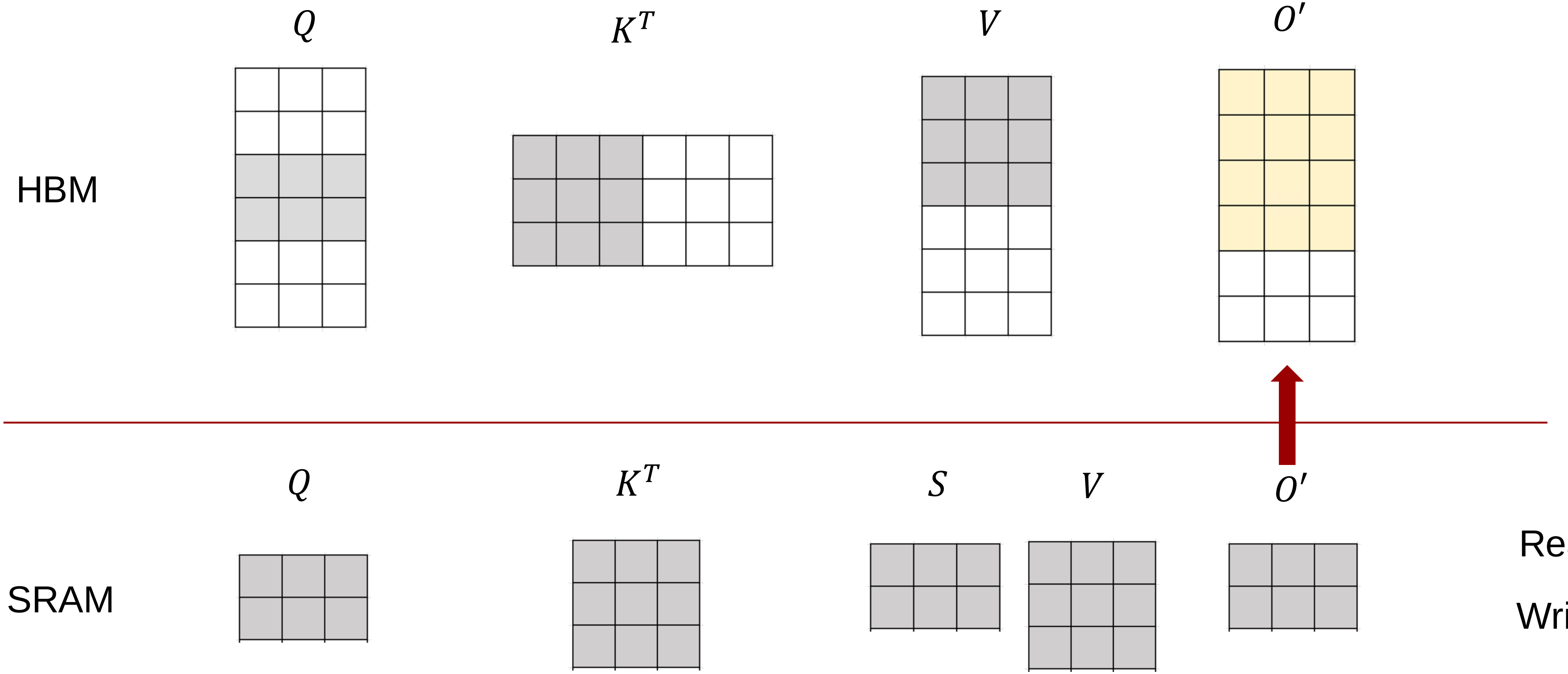
[Bilibili: Flash Attention 为什么那么快? By RethinkFun]

Kernal fusion in simplified attention



[Bilibili: Flash Attention 为什么那么快? By RethinkFun]

Kernal fusion in simplified attention

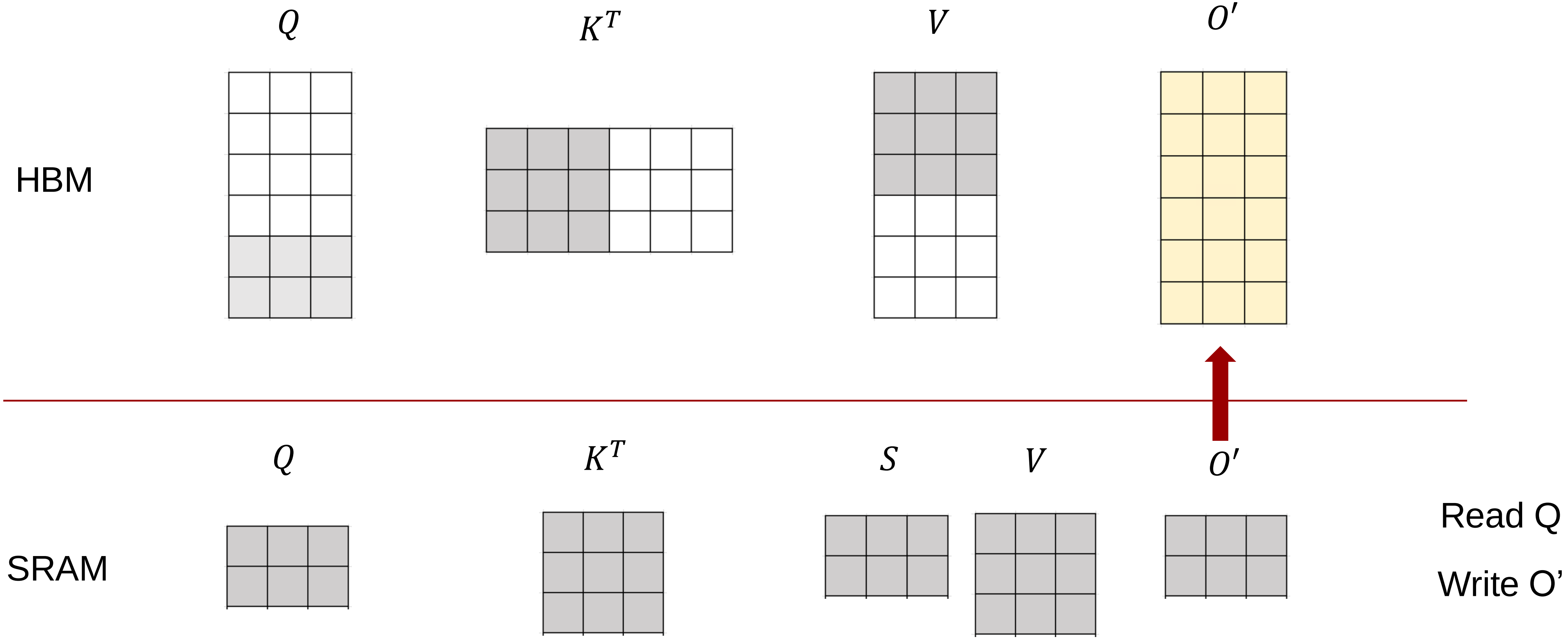


Read Q

Write O’

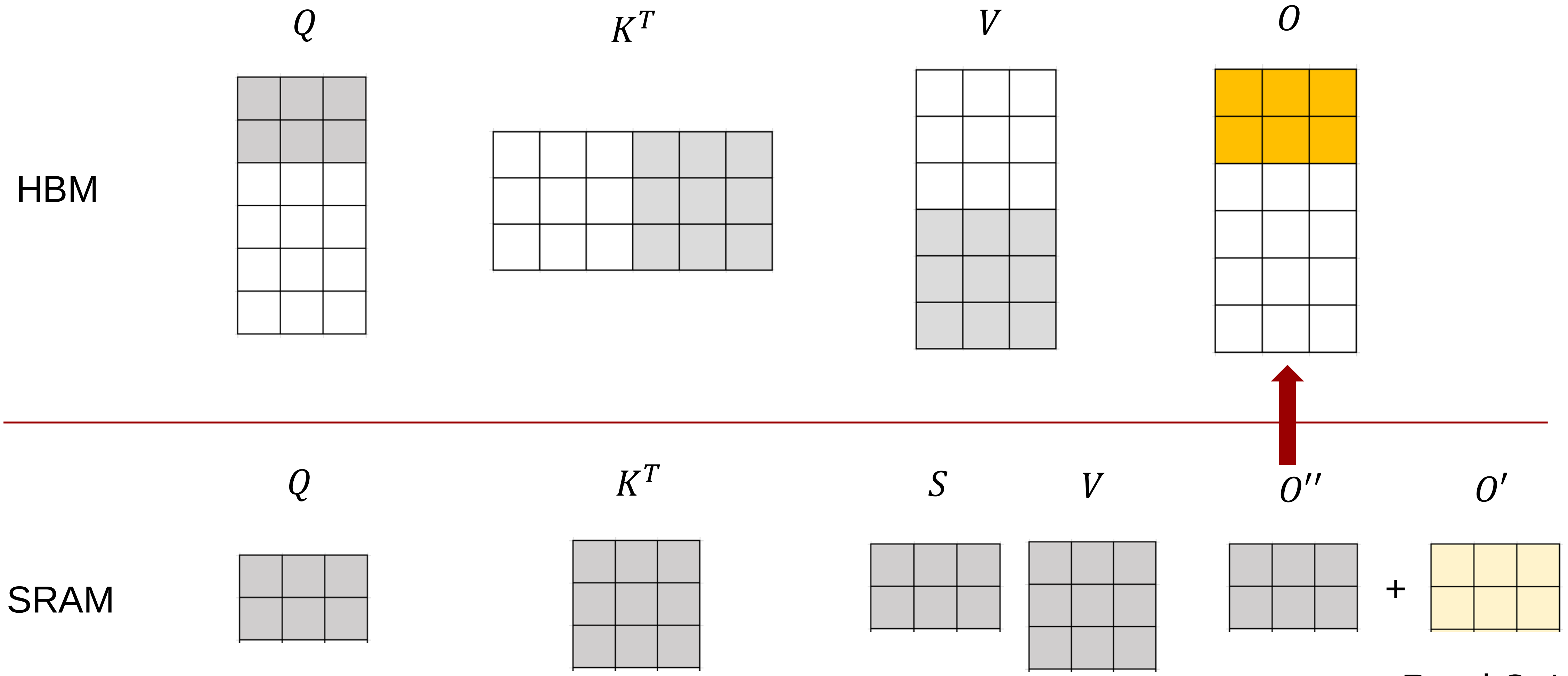
[Bilibili: Flash Attention 为什么那么快? By RethinkFun]

Kernal fusion in simplified attention



[Bilibili: Flash Attention 为什么那么快? By RethinkFun]

Kernal fusion in simplified attention

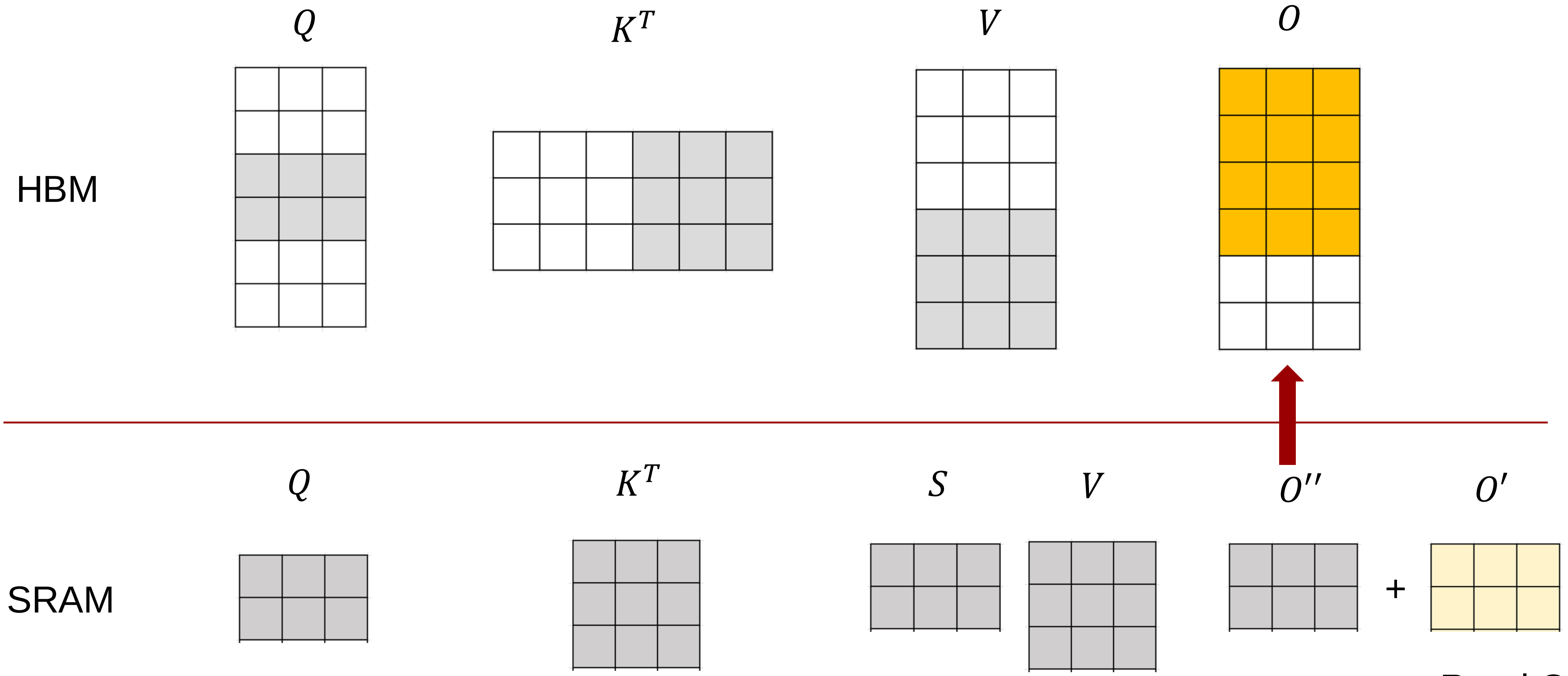


Read Q, K, V,O’

Write O

[Bilibili: Flash Attention 为什么那么快? By RethinkFun]

Kernal fusion in simplified attention

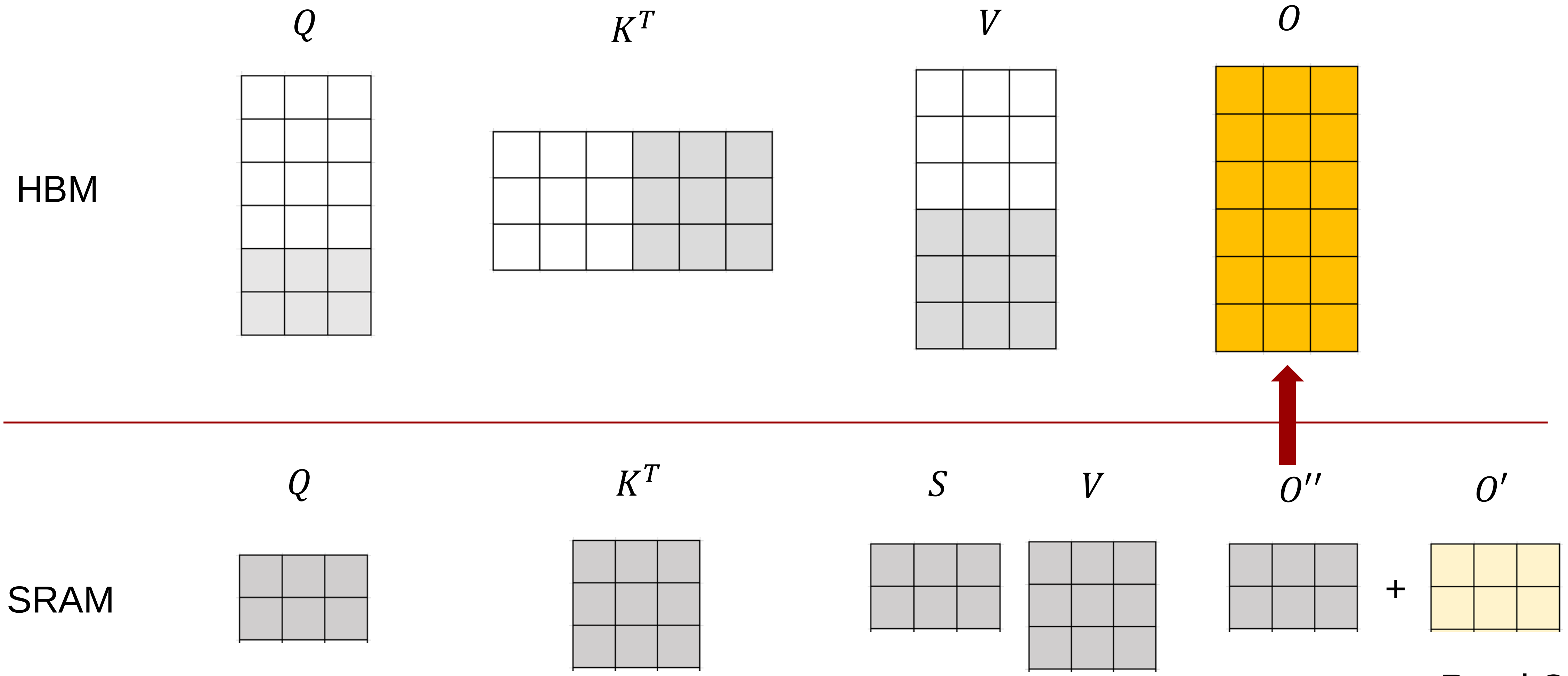


Read Q, O'

Write O

[Bilibili: Flash Attention 为什么那么快? By RethinkFun]

Kernal fusion in simplified attention



Read Q, O'

Write O

[Bilibili: Flash Attention 为什么那么快? By RethinkFun]

HBM accessing comparison

Vanila Attention

Operation	MAC
Load Q and K	$2dN$
Write S	N^2
Read S	N^2
Write P	N^2
Load Q and V	$N^2 + dN$
Write O	dN

$$\mathbf{4N^2 + 4dN}$$

Flash Attention

Operation	MAC
Load Q twice	$2dN$
Load K, V	$2dN$
Write O’	dN
Read O’	dN
Write O	dN

$$\mathbf{7dN}$$

Kernal fusion significantly saves MAC

- When $N \gg d$, FlashAttention significantly saves MAC $\boldsymbol{4N^2 + 4dN \gg 7dN}$

- The longer the sequence length is, the better that FlashAttention is

- The fundamental reason is that we fusion the intermediate operators, e.g., **do not store $\boldsymbol{S}$**

Thank you!

Kun Yuan homepage: https://kunyuan827.github.io/