Korea University

Flash Attention-3: Fast and Accurate Attention with Asynchrony and Low-precision

NeurIPS 2024

2025020686

김영혁

TIMELINE

Flash Attention 1

2022.10

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

Flash Attention 2

2023.6

Flash Attention-2: Faster Attention with Better Parallelism and Work Partitioning

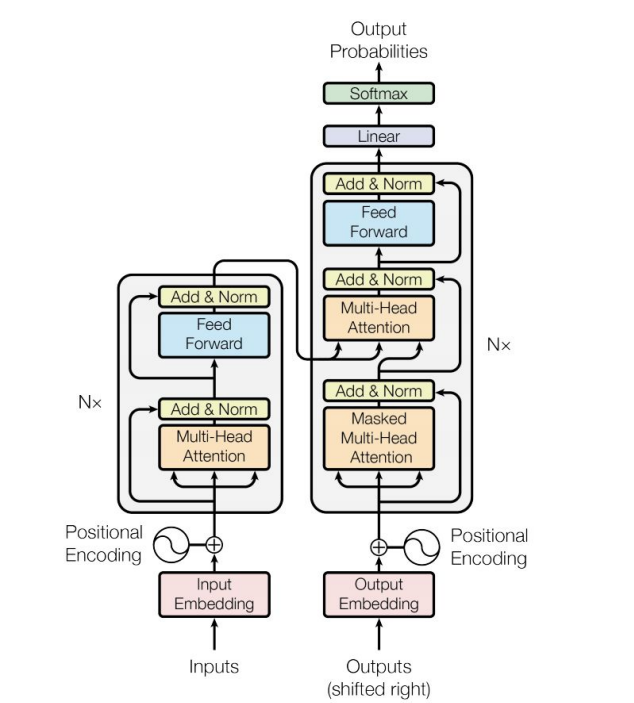
Flash Attention 3

2024.3

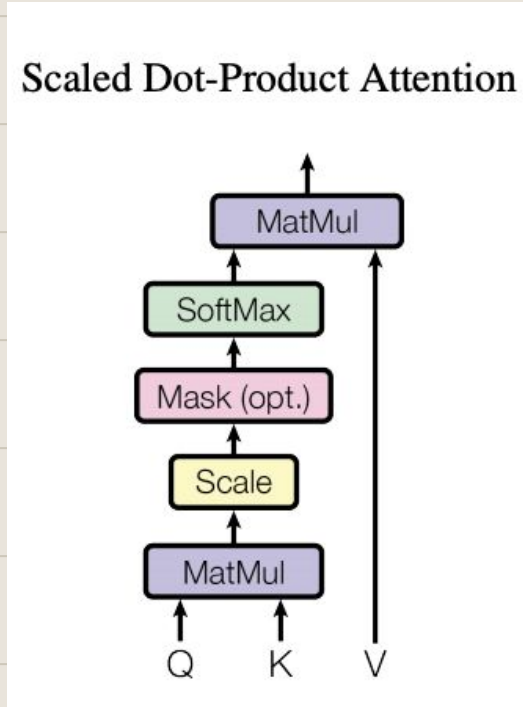
Flash Attention-3: Faster Attention with Better Parallelism and Customization

Attention

- Transformer



[Image source: Vaswani et al. (2017)]

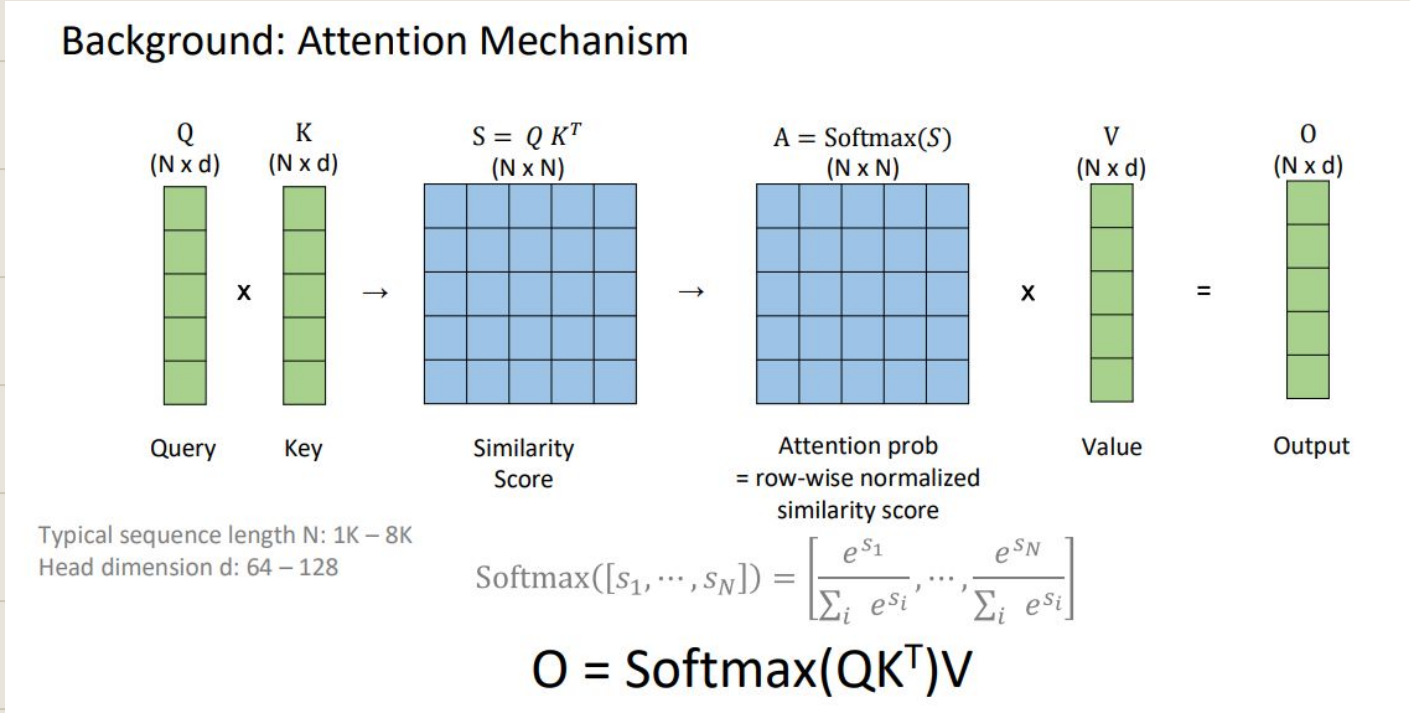


- **Scaled Dot-Product Attention**

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

Attention

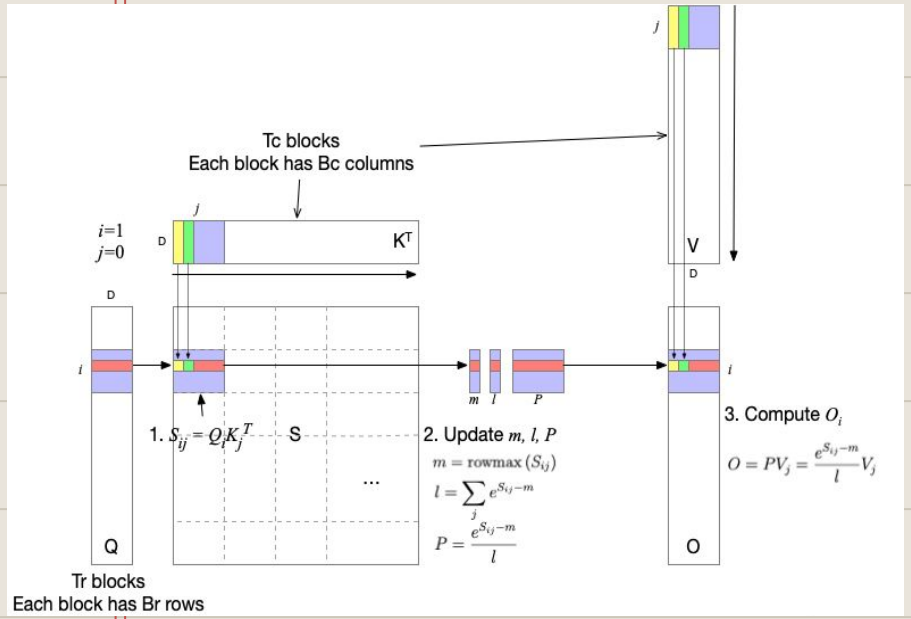
- The entire sequence of length N must be stored for the Softmax

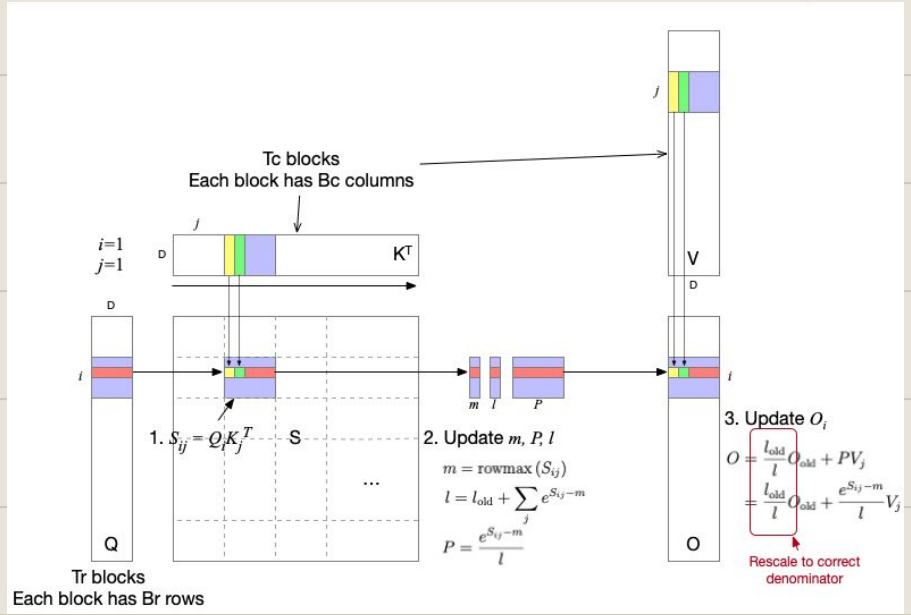


[Image source: 2024 NuerIPS, Flash Attention 3]

Flash Attention

- Softmax computed using partial scores S, with the max value
- Recomputed and compensated for later.





[Image source: https://insujang.github.io/2024-01-21/flash-attention/]

Flash Attention

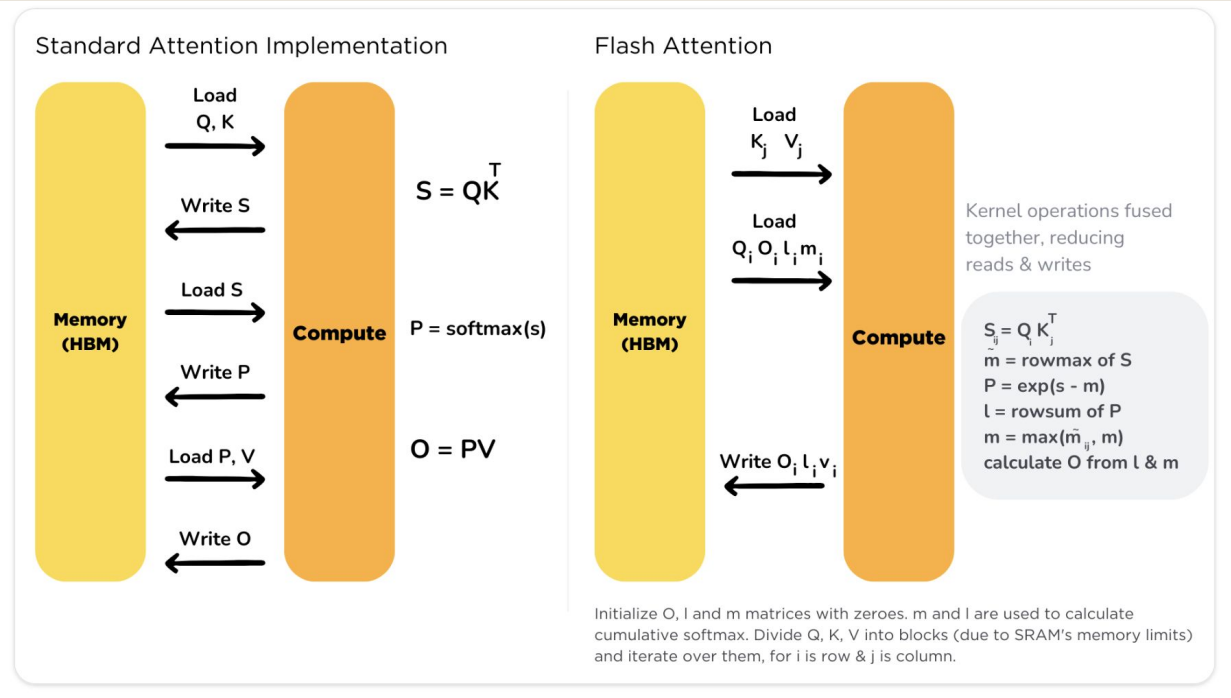
- Recompute O1 with new max M1

Loop 0	Loop 1
$S_0 = QK_0$	$S_1 = QK_1$
$m_0 = rowmax(S_0)$	$m_1 = \max(rowmax(S_0), m_0)$
$l_0 = rowsum(e^{s_0 - m_0})$	$l_1 = l_0 * e^{m_0 - m_1} + e^{s_1 - m_1}$
$P_0 = \frac{e^{s_0 - m_0}}{l_0}$	$P_1 = \frac{e^{s_1 - m_1}}{l_1}$
$O_0 = P_0 V_0$	$O_1 = O_0 * \frac{l_0}{l_1} + P_1 V_1$

[Image source: https://zhuanlan.zhihu.com/p/607364156]

Flash Attention

- Benefits : Reducing Read & Write cycle



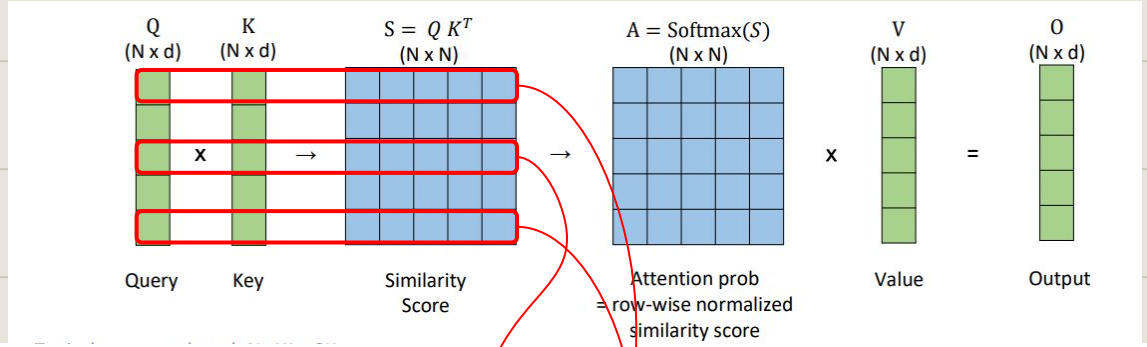
[Image source: https://velog.io/@jpseo99/Flash-Attention]

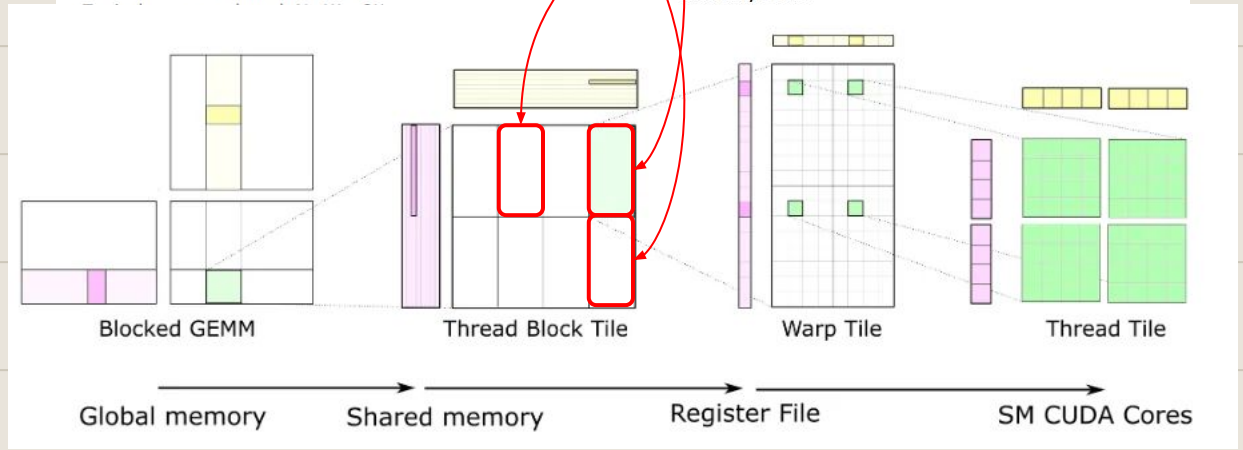
Flash Attention

- Limitation
  - **Limited inter-thread block parallelism**: For long sequences, a single CTA (Cooperative Thread Array) handles the computation, which restricts parallelization.
  - **Rigid memory layout**: The fixed structure limits flexibility and adaptability to different models or hardware.
  - **Performance benefits mainly for long sequences**: Gains are significant only for long sequences; performance is limited for short sequences or small batch sizes.

Flash Attention 2

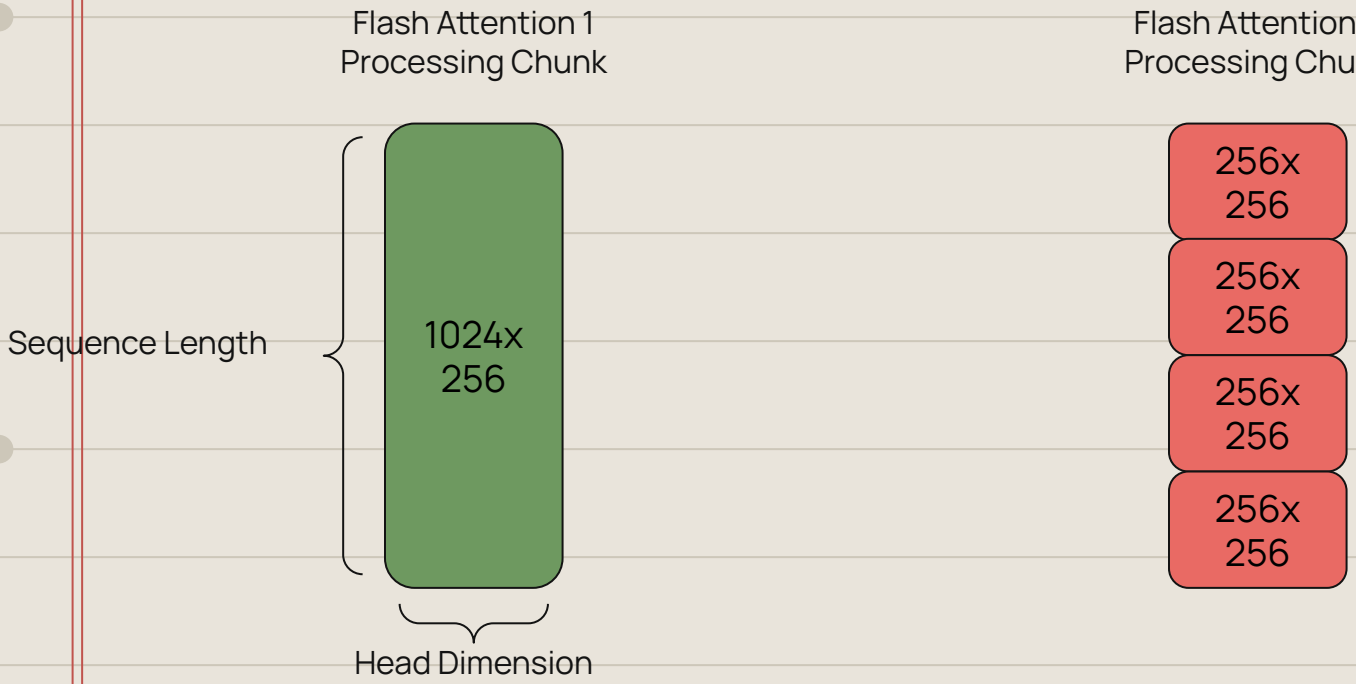
- Main idea : Thread Block parallelism (Global max and update later)





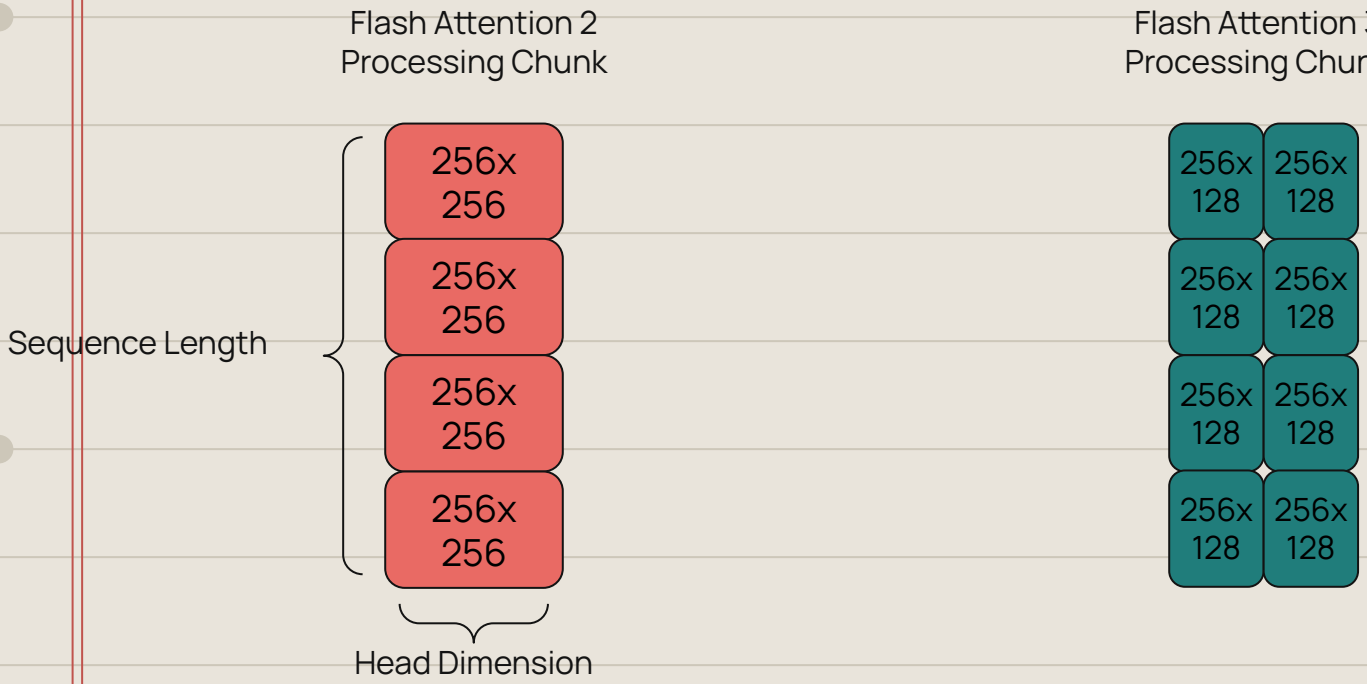
Flash Attention 2

- Main idea : Thread Block parallelism (Global max and update later)



Flash Attention 3

- Main idea : M(Head dim) parallelism + Re-ordering + Block Quantization
- M(Head dim) parallelism



Flash Attention 3

- Main idea : M parallelism + Re-ordering + Block Quantization
- Re-ordering

Flash attention 1&2

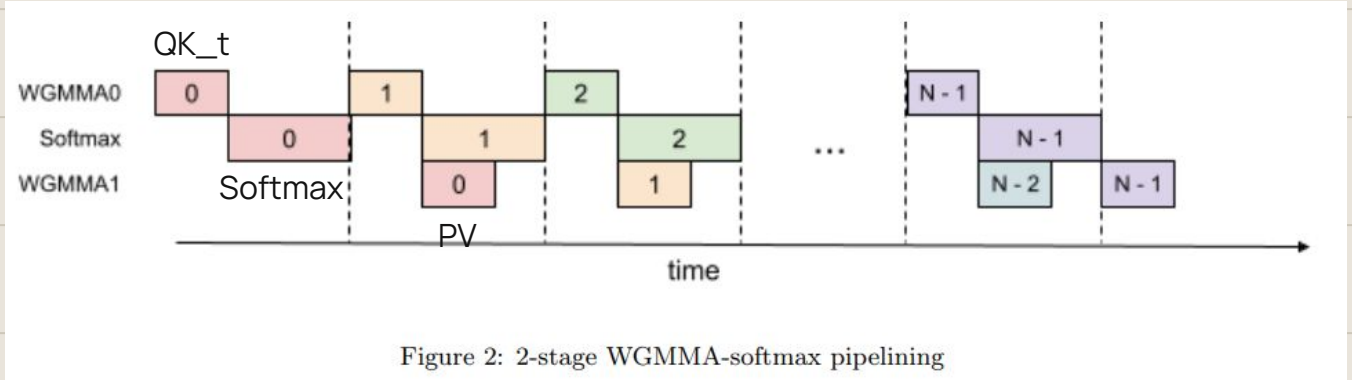
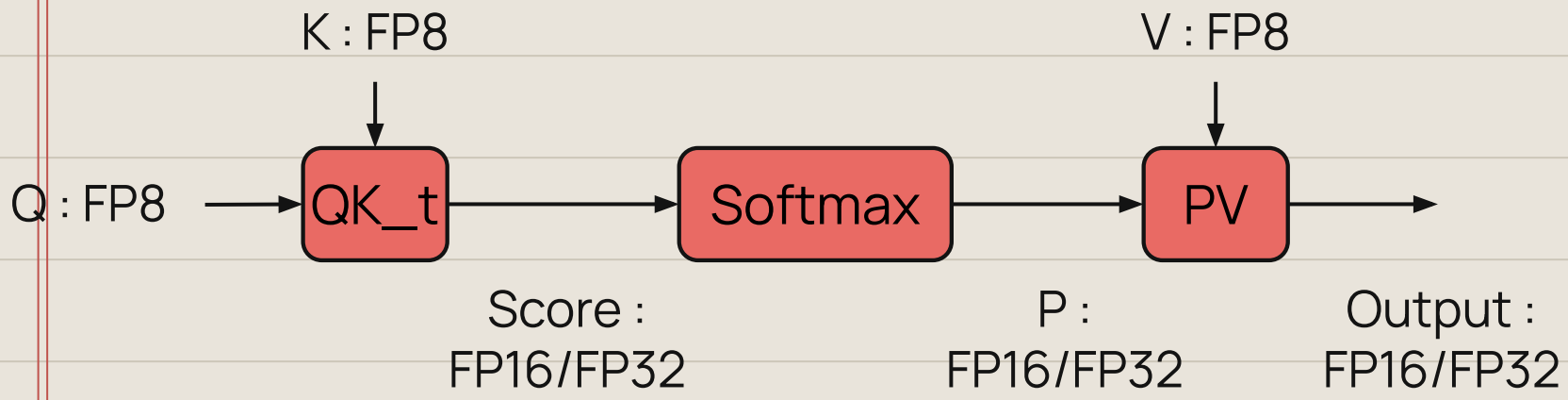


Figure 2: 2-stage WGMMA-softmax pipelining

[Image source: Flash Attention 3]

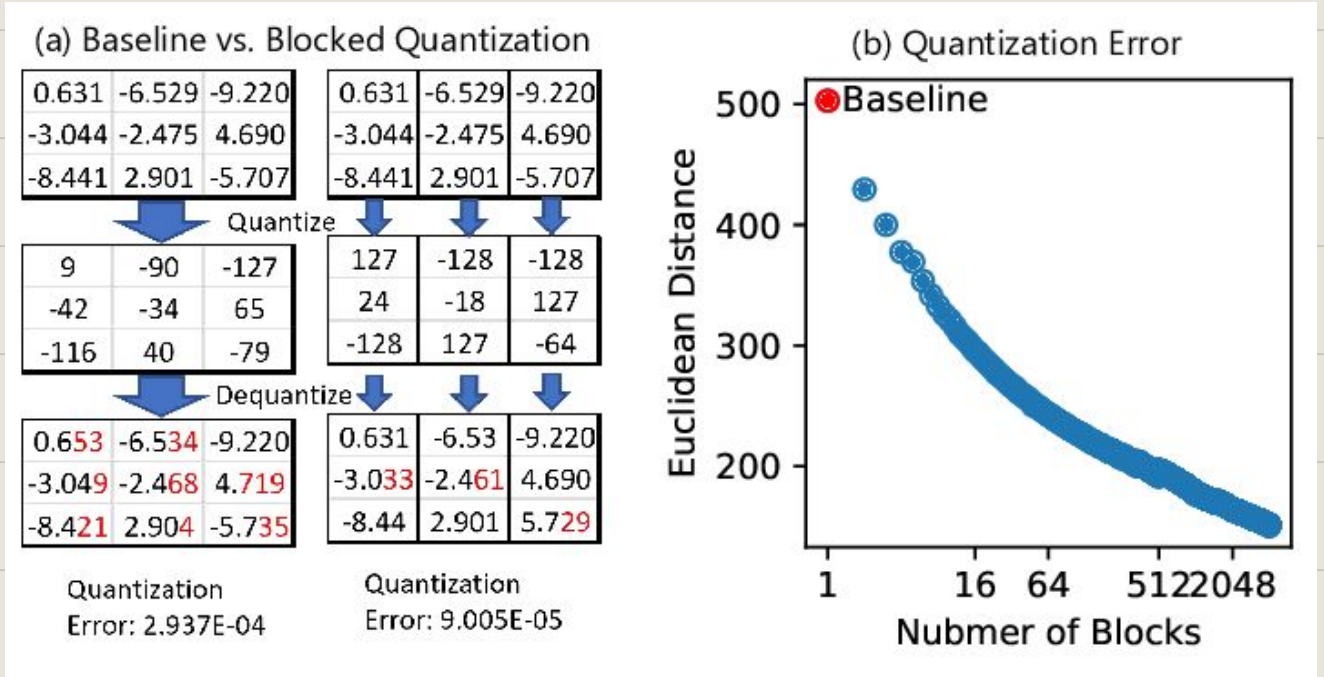
Flash Attention 3

- Main idea : M parallelism + Re-ordering + Block Quantization
- FP8 Block Quantization



Flash Attention 3

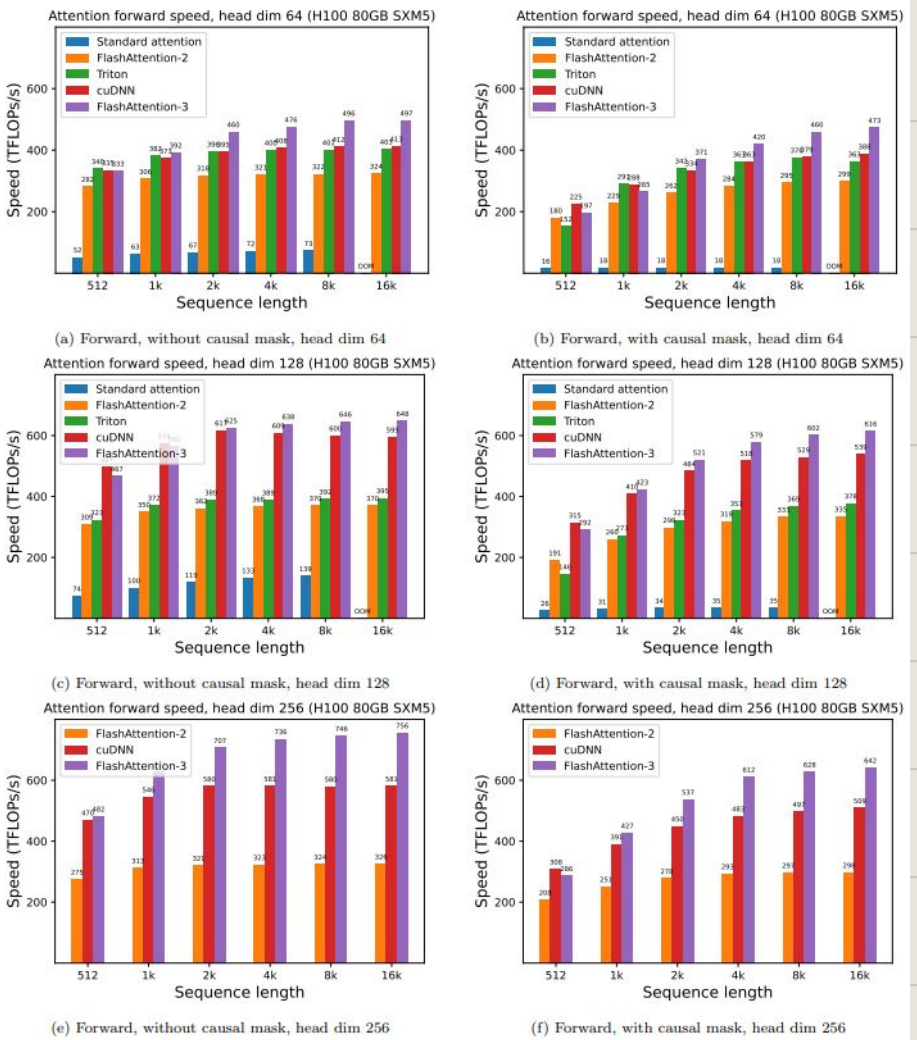
- Main idea : M parallelism + Re-ordering + Block Quantization
- FP8 Block Quantization



[Image source: https://www.researchgate.net/figure/llustration-example-of-block-based-quantization-vs-baseline_fig2_371728248]

Flash Attention 3

- Validation
- GPU Optimize library
  - cuDNN : CUDA Deep Neural Network
  - Triton : GPU Programming for Neural Networks



[Image source: Flash Attention - 3]

THANK YOU!