AttAcc! Unleashing the Power of PIM for Batched Transformer-based Generative Model Inference

Jaehyun Park*†, Jaewan Choi*†, **Kwanhee Kyung**†, Michael Jaemin Kim†,

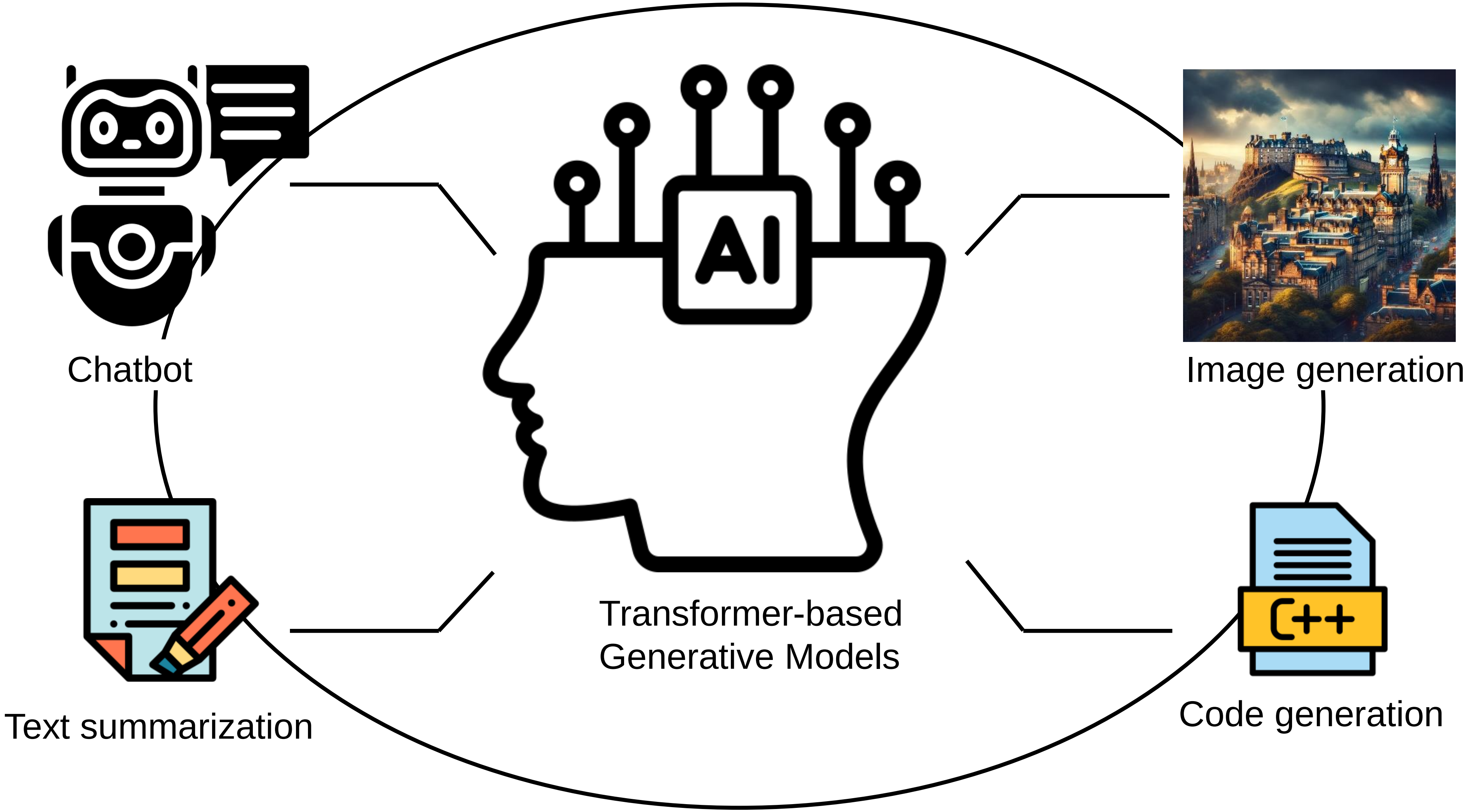
Yongsuk Kwon†, Nam Sung Kim‡, Jung Ho Ahn†

† Seoul National University, ‡ University of Illinois Urbana Champaign

* Equally contributed

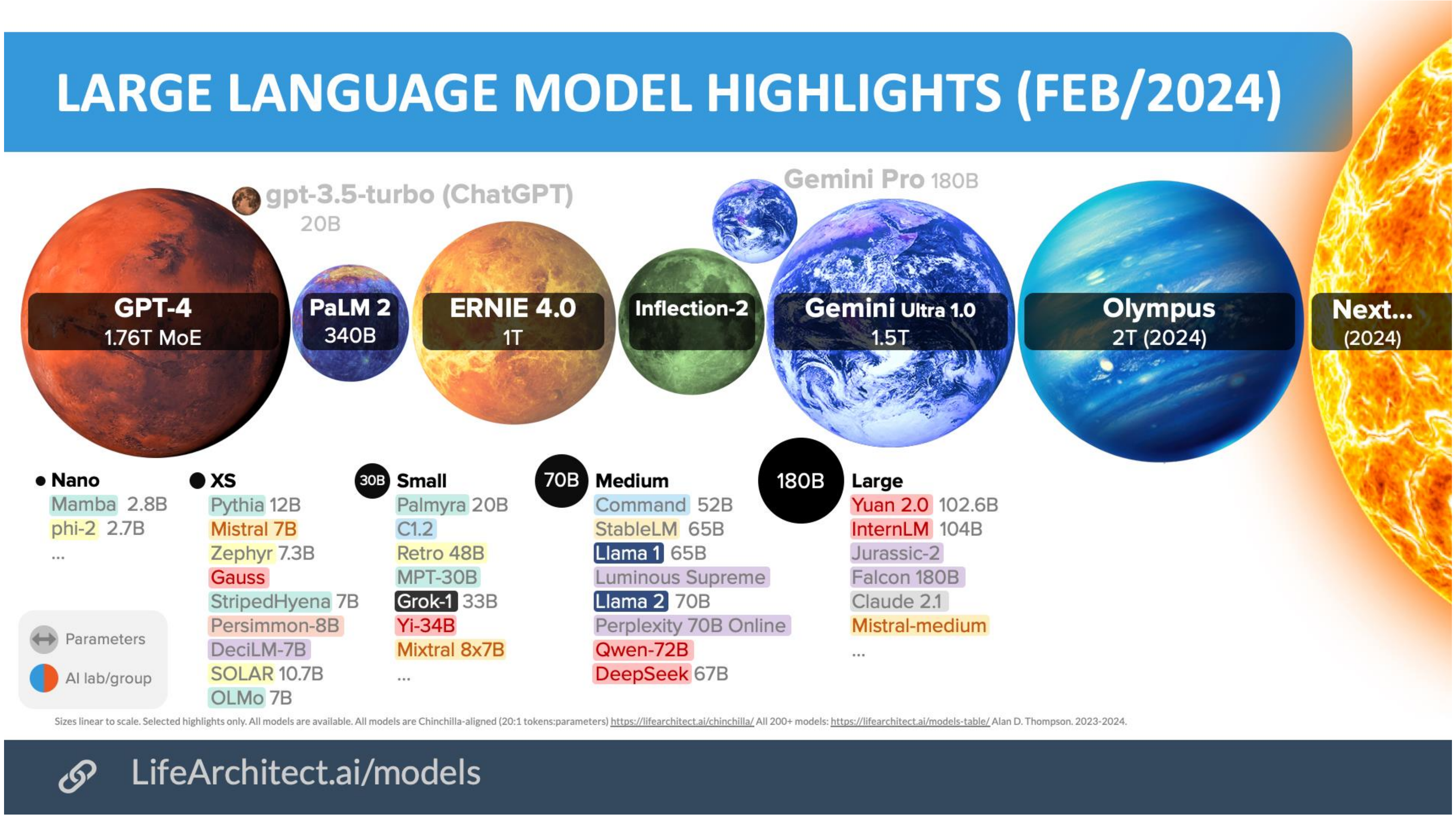
Presenter: Kwanhee Kyung (kwanhee.kyung@scale.snu.ac.kr)

Why Transformer-based Generative Model (TbGM) Inference?



Why Transformer-based Generative Model (TbGM) Inference?

- Model size

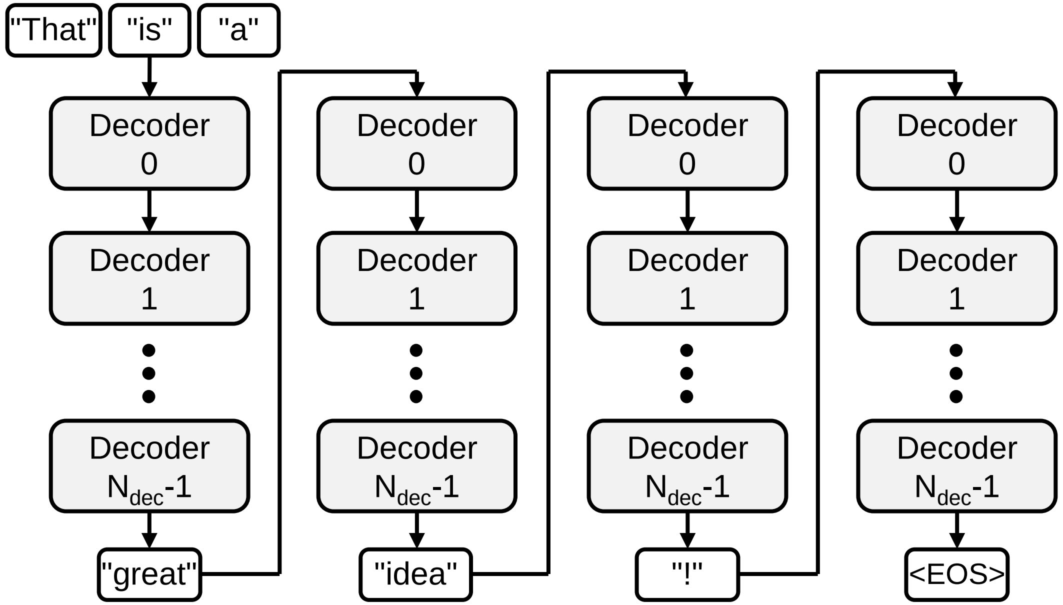


Why Transformer-based Generative Model (TbGM) Inference?

- Sequence length (L) supported by TbGM

https://hazyresearch.stanford.edu/blog/2023-03-27-long-learning

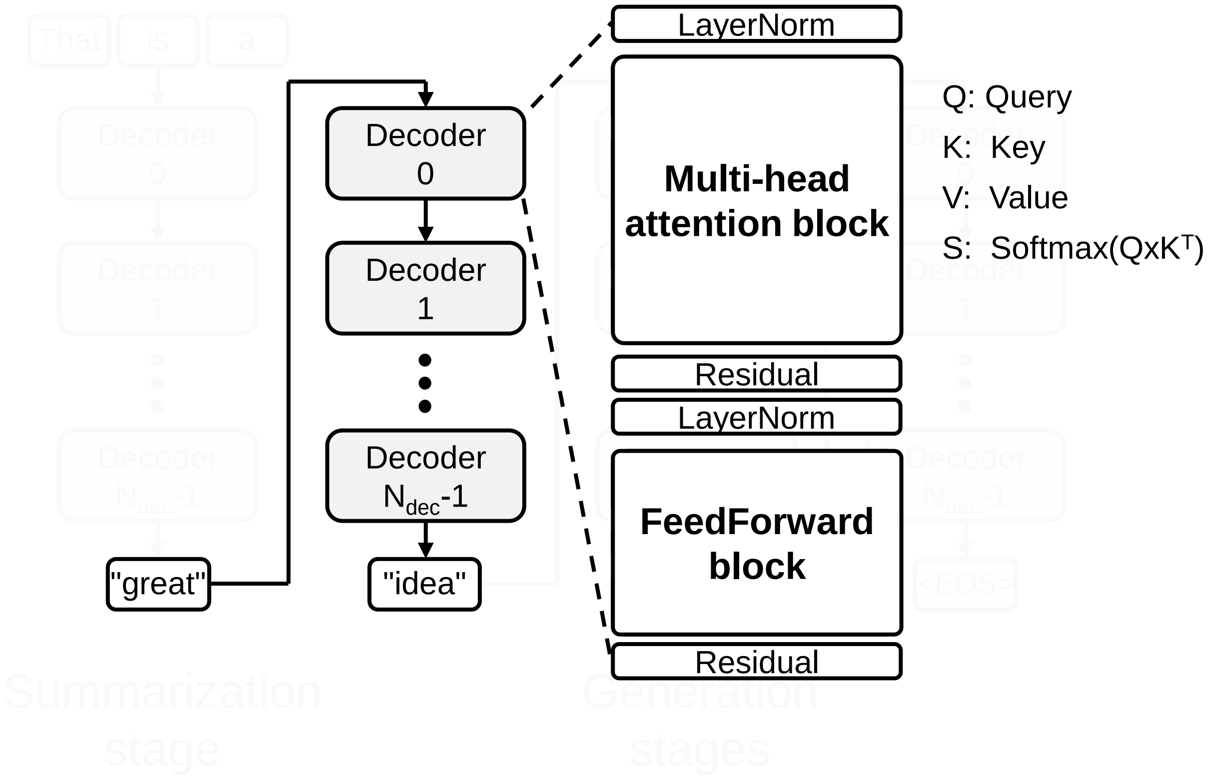
TbGM Inference



Summarization (Sum) stage (=Prefill phase)

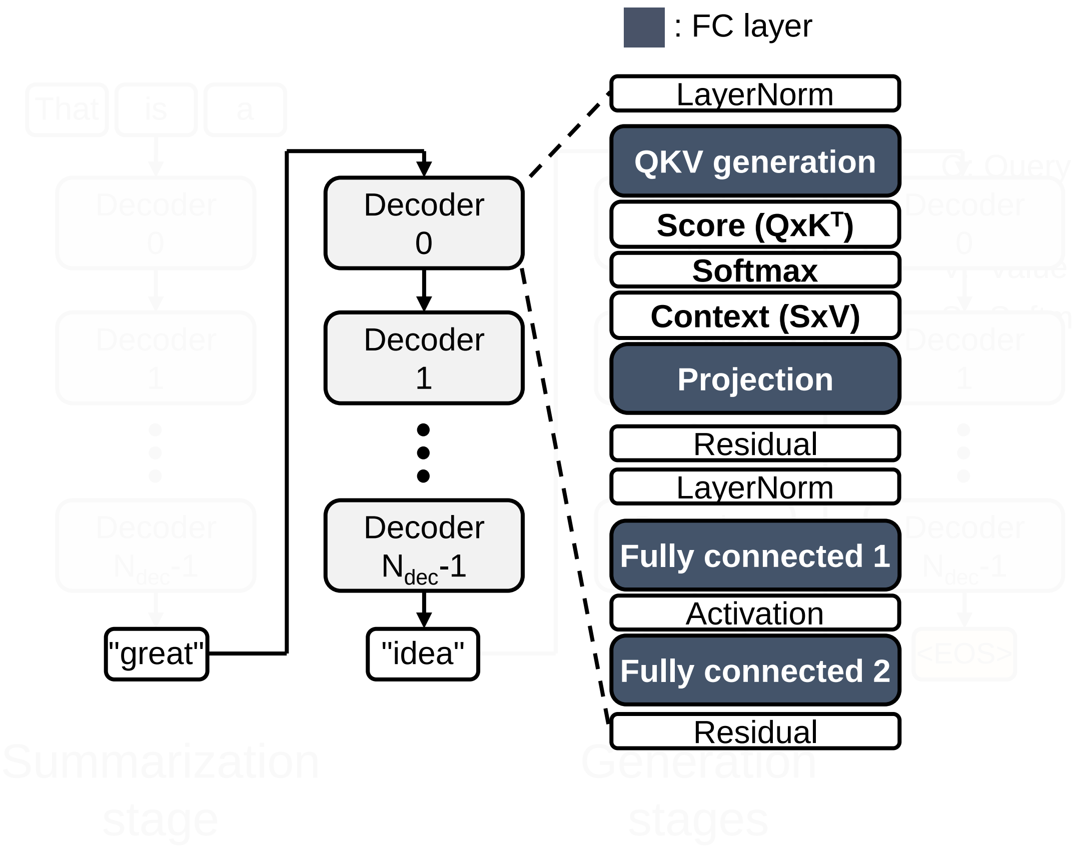
Generation (Gen) stages (=Decode phase)

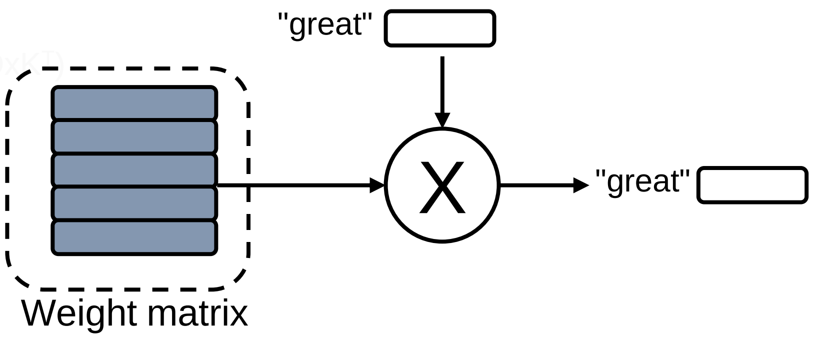
Characteristics of the Gen Stage



Characteristics of the Gen Stage

- **FC layers** are all **general matrix-vector multiplications (GEMVs)**

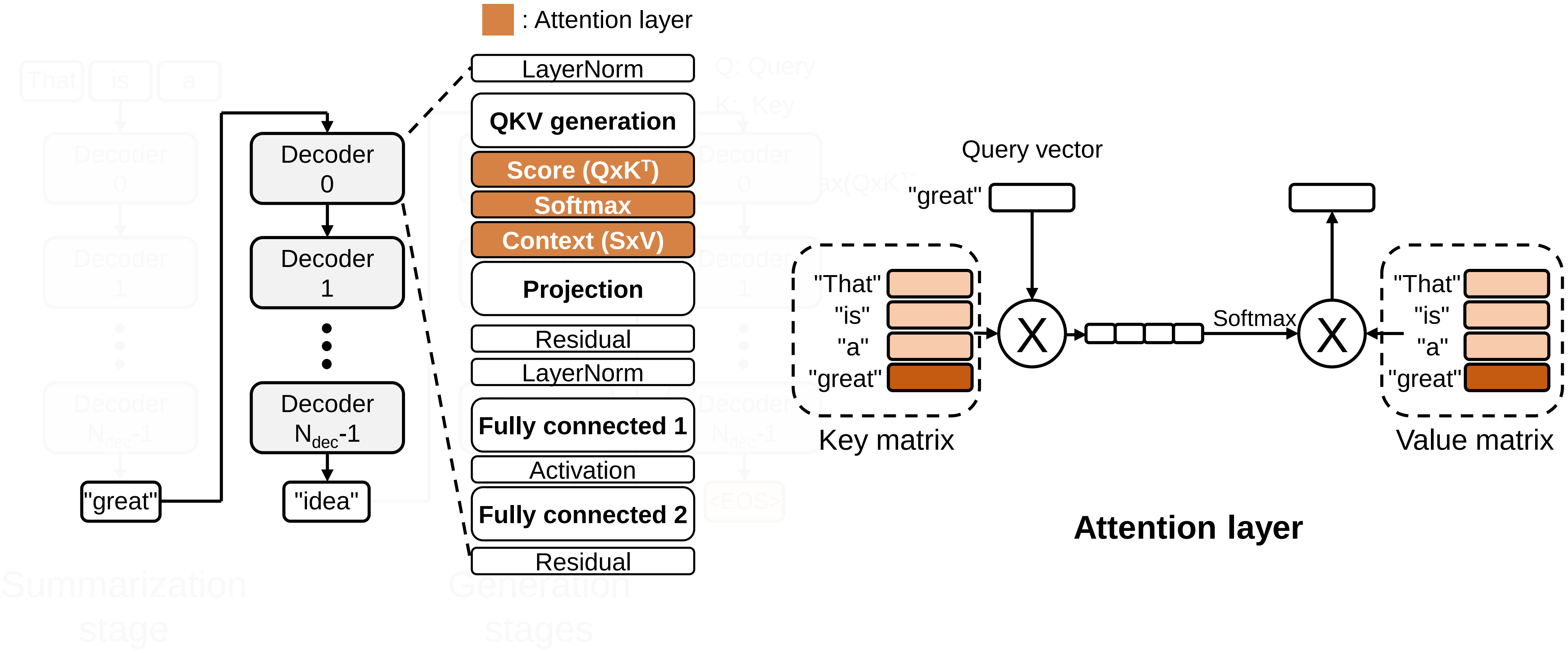




Fully Connected (FC) layer

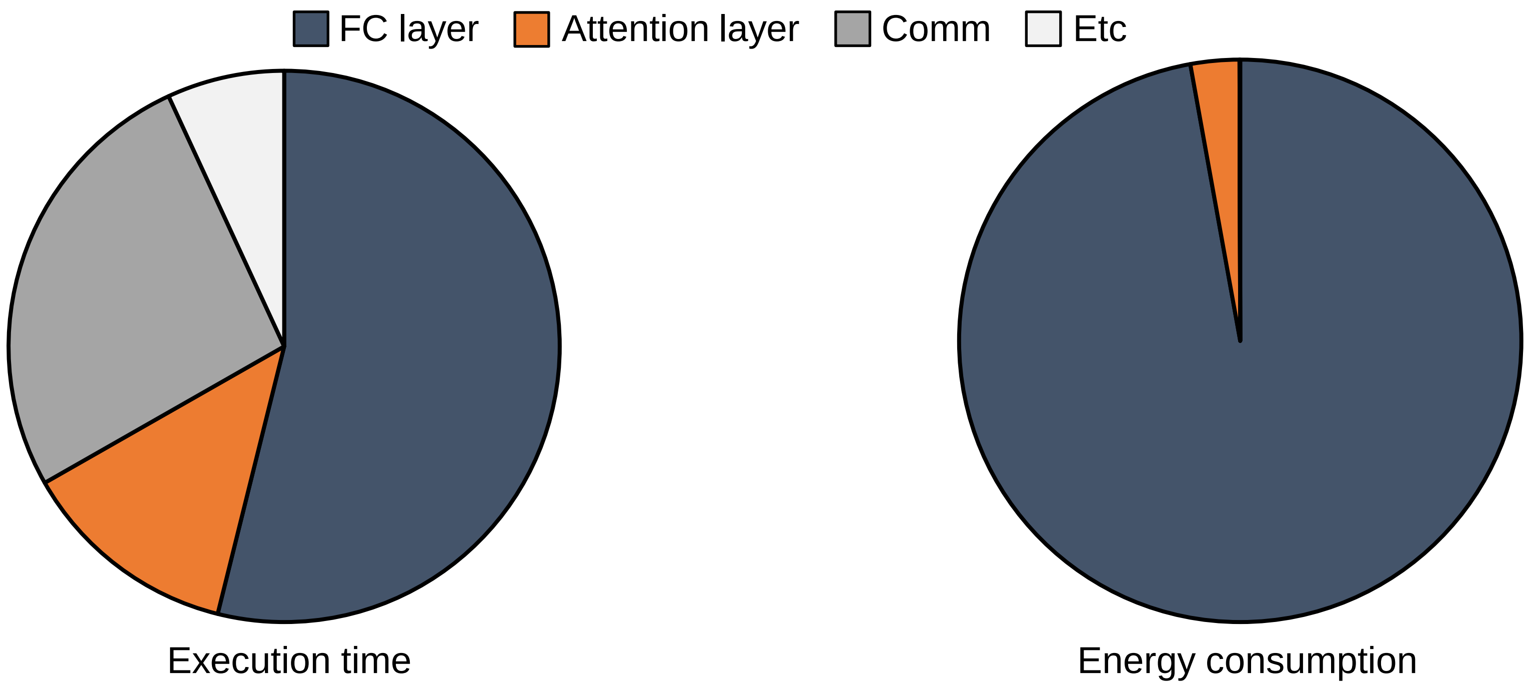
Characteristics of the Gen Stage

- The **attention layer** also has **GEMVs** (GEMV$_{score}$ and GEMV$_{context}$)



Prior TbGM Accelerators

- Many prior works [1,2,3] for TbGM accelerator focused on accelerating **FC layers**
  - FC layers account for a significant portion of execution time and energy consumption



Execution time and energy consumption breakdown of TbGM inference
(GPT3-175B on DGX-A100 with HBM3 , L_{in}=2048, L_{out}=128)

[1] S Lee et al., "Hardware Architecture and Software Stack for PIM Based on Commercial DRAM Technology," ISCA, 2021
[2] S Hong et al., "DFX: A Low-latency Multi-FPGA Appliance for Accelerating Transformer-based Text Generation," MICRO, 2022.
[3] D Kwon et al., "A 1ynm 1.25V 8Gb 16Gb/s/Pin GDDR6-Based Accelerator-in-Memory Supporting 1TFLOPS MAC Operation and Various Activation Functions for Deep Learning Application," JSSC, 2023

Prior TbGM Accelerators

- Many prior works [1,2,3] for TbGM accelerator focused on accelerating **FC layers**
  - FC layers account for a significant portion of execution time and energy consumption

However, a key assumption so far is that the batch size is 1

Execution time and energy consumption breakdown of TbGM inference
(GPT3-175B on DGX-A100 with HBM3 , L_{in}=2048, L_{out}=128)

[1] S Lee et al., "Hardware Architecture and Software Stack for PIM Based on Commercial DRAM Technology," ISCA, 2021
[2] S Hong et al., "DFX: A Low-latency Multi-FPGA Appliance for Accelerating Transformer-based Text Generation," MICRO, 2022.
[3] D Kwon et al., "A 1ynm 1.25V 8Gb 16Gb/s/Pin GDDR6-Based Accelerator-in-Memory Supporting 1TFLOPS MAC Operation and Various Activation Functions for Deep Learning Application," JSSC, 2023

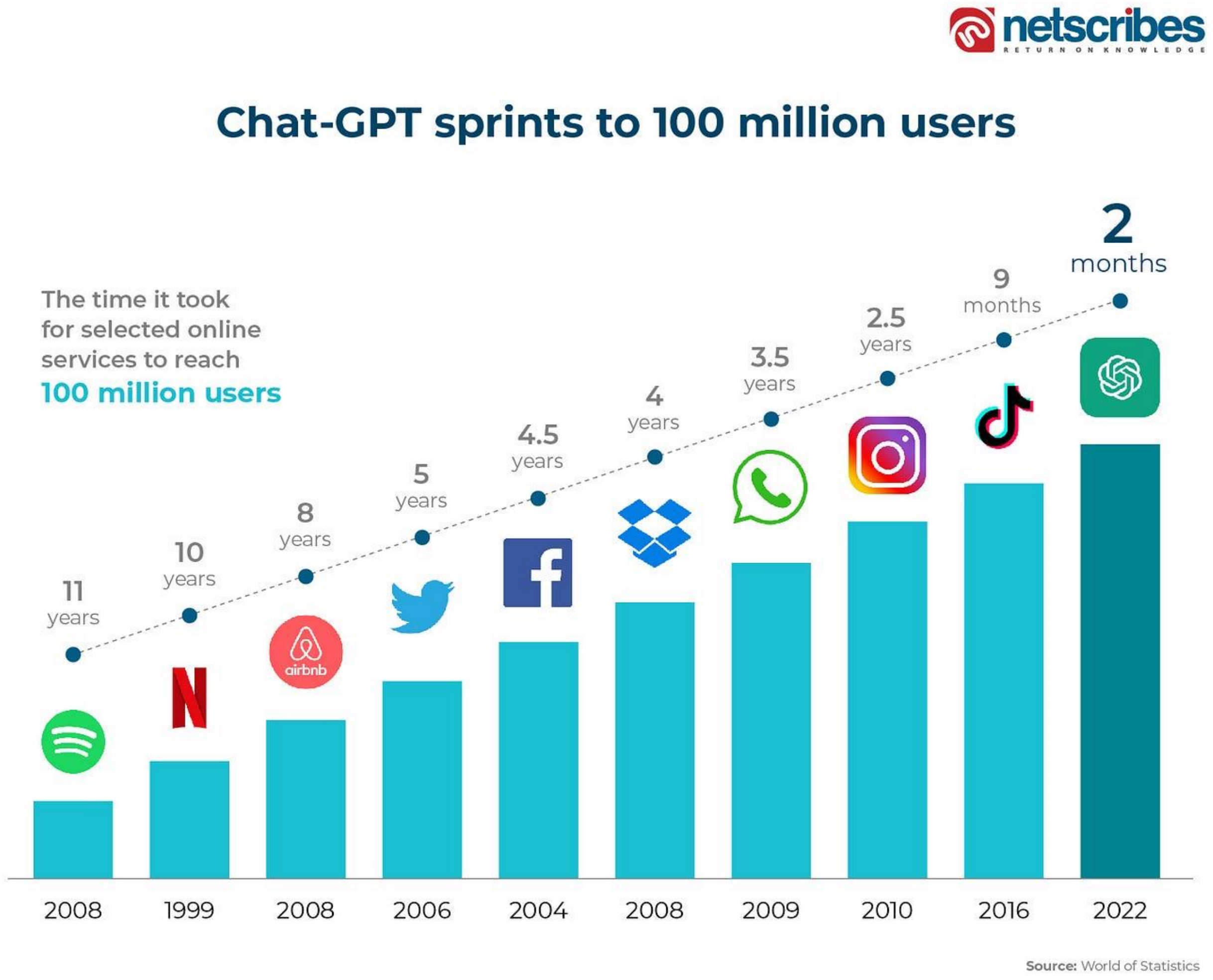
Outline

- We discover that **the attention layer**
  - becomes more important in batched TbGM inference
  - poses several challenges in conventional systems

- To address these challenges, we propose
  - processing-in-memory (PIM)-based accelerator (***AttAcc***) for the attention layer
  - heterogeneous system with ***AttAcc*** and xPU for end-to-end TbGM inference
  - optimizations that improve utilization of the heterogeneous system

Why Large Batch Size Matters

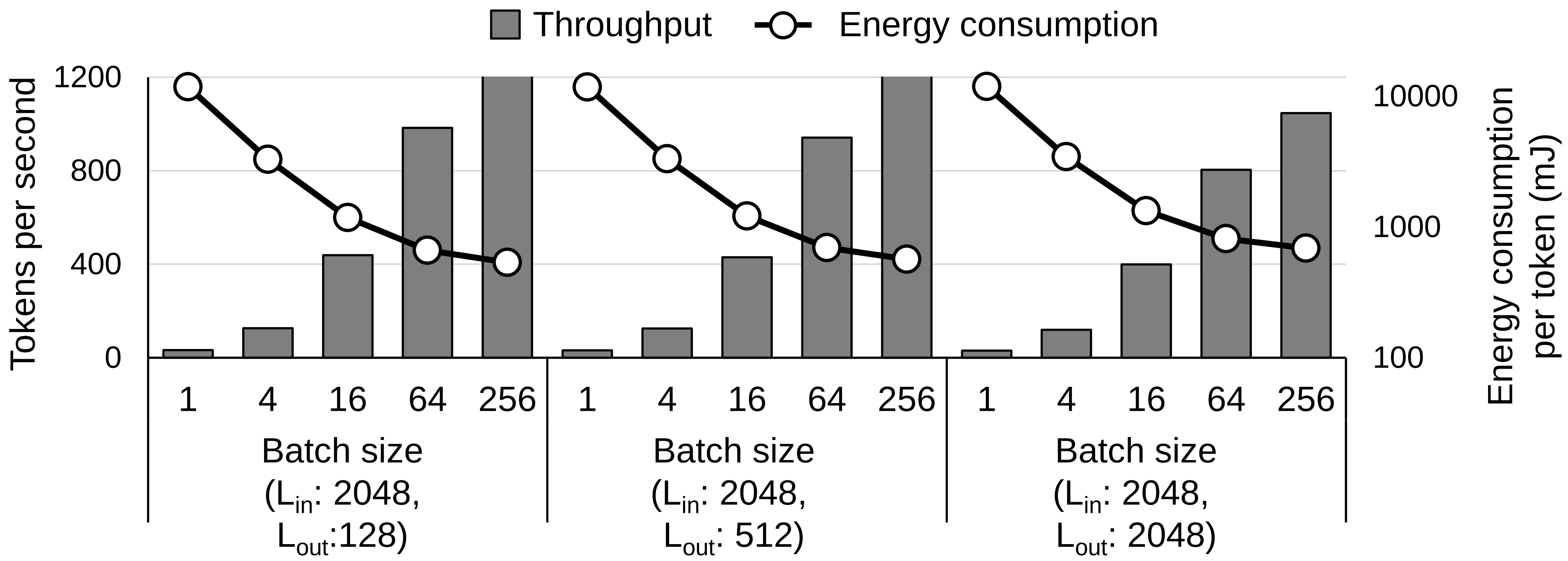
- Ensure sufficient requests from increased TbGM inference usage



https://ai.plainenglish.io/chat-gpt-achieving-100-million-users-in-just-2-month-a-deep-analysis-a453e6f85acf

Why Large Batch Size Matters

- Ensure sufficient requests from increased TbGM inference usage
- Batching technique for TbGM inference [1] enables high throughput and energy efficiency

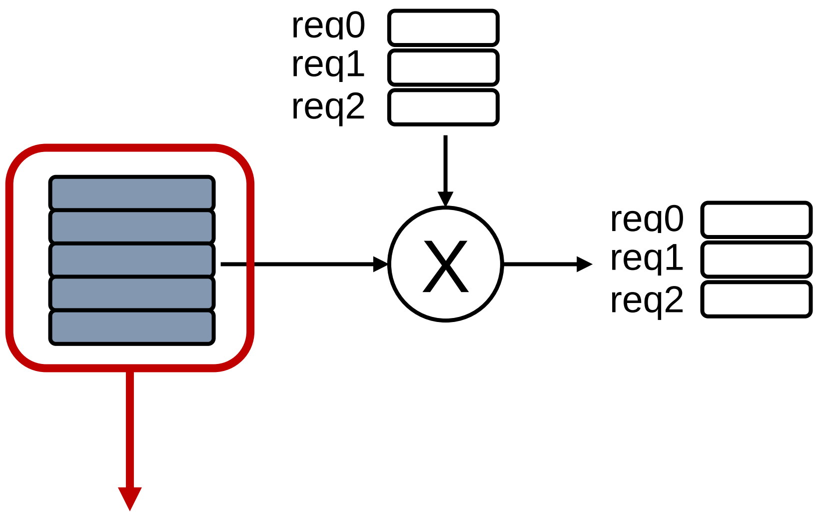


Throughput and energy consumption per output token of TbGM inference (GPT-3 175B, DGX A100 with HBM3)

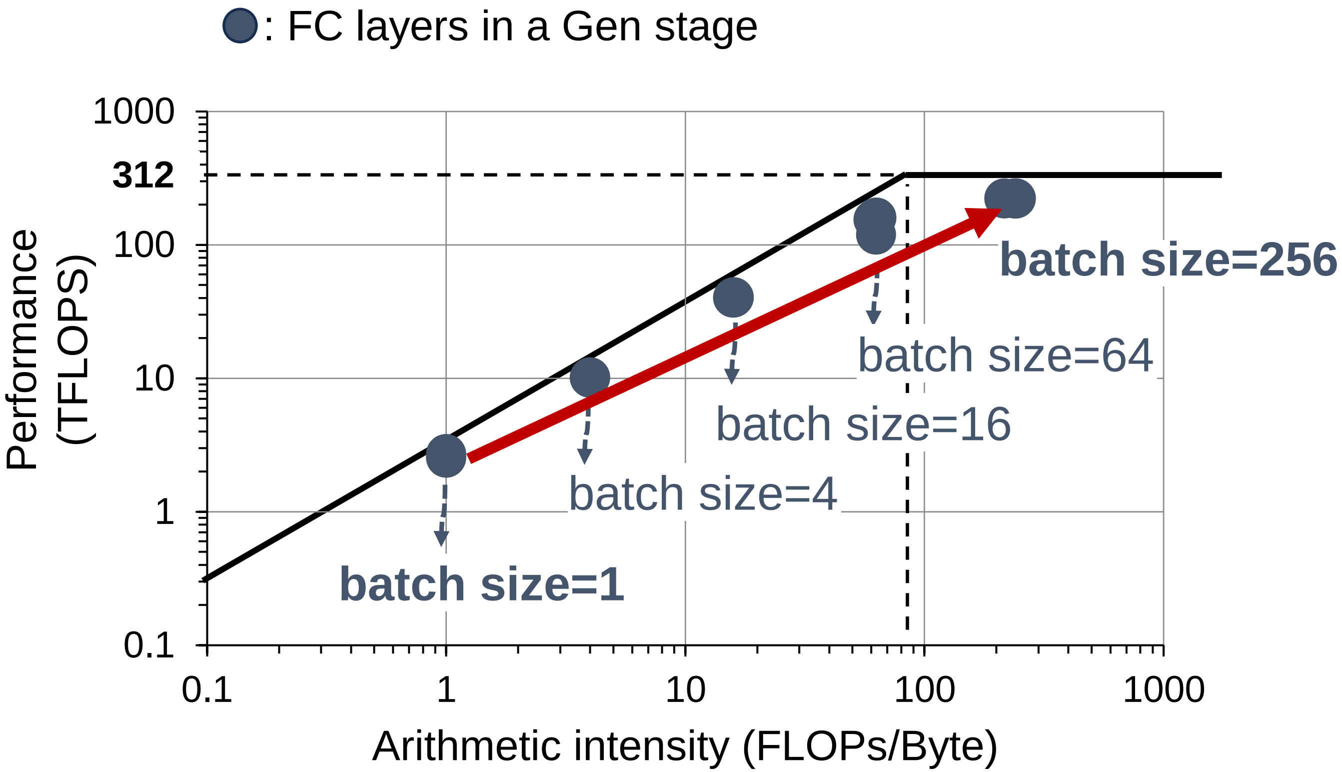
[1] G Yu et al., "Orca: A Distributed Serving System for Transformer-Based Generative Models," OSDI, 2022

What If Batch Size Increases?

- **FC layers** become more **compute-intensive**
  - Weight matrices are shared across different requests

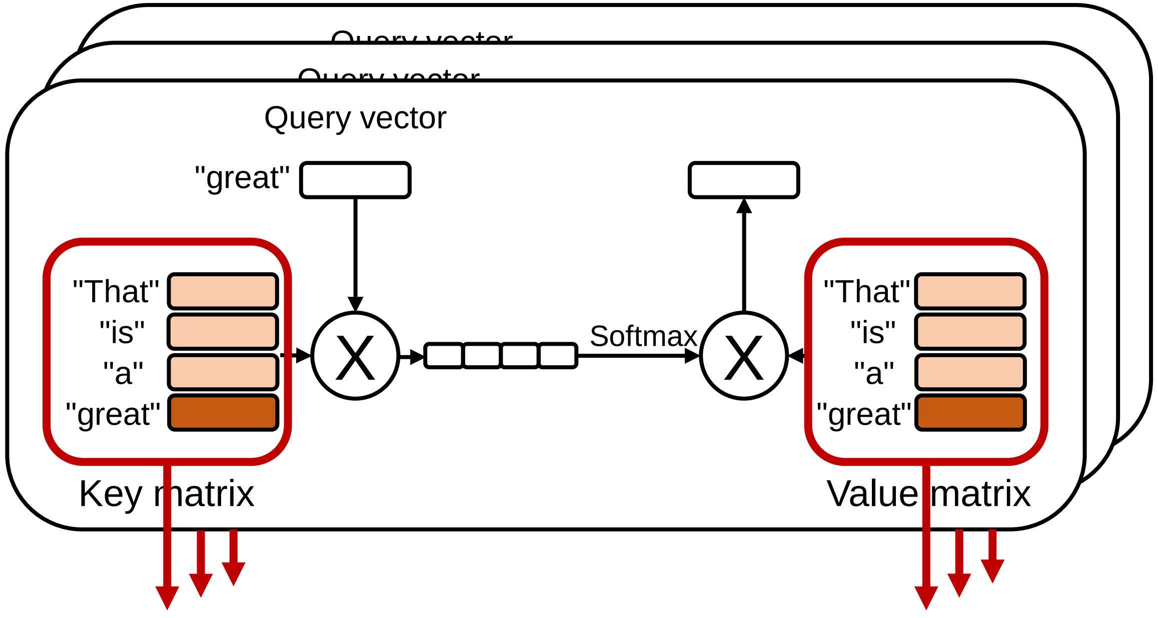


Weights reused across multiple requests

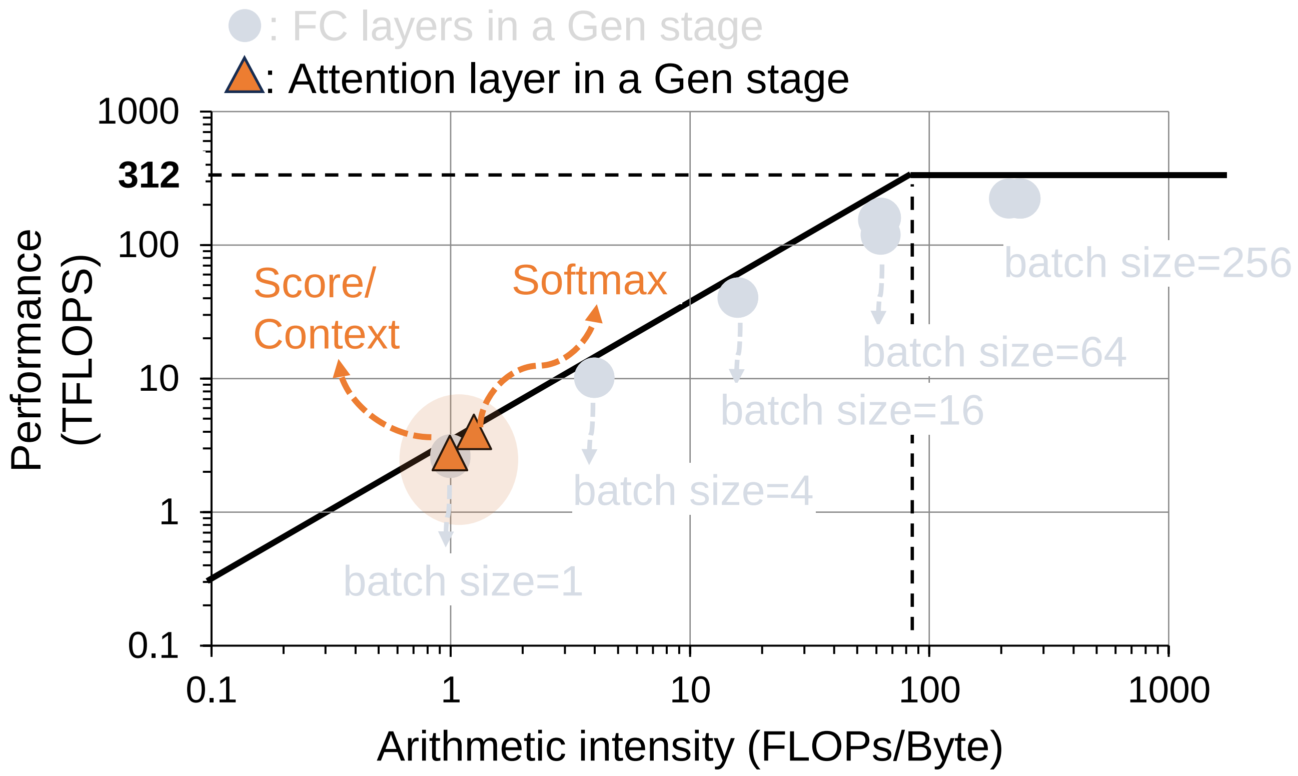


What If Batch Size Increases?

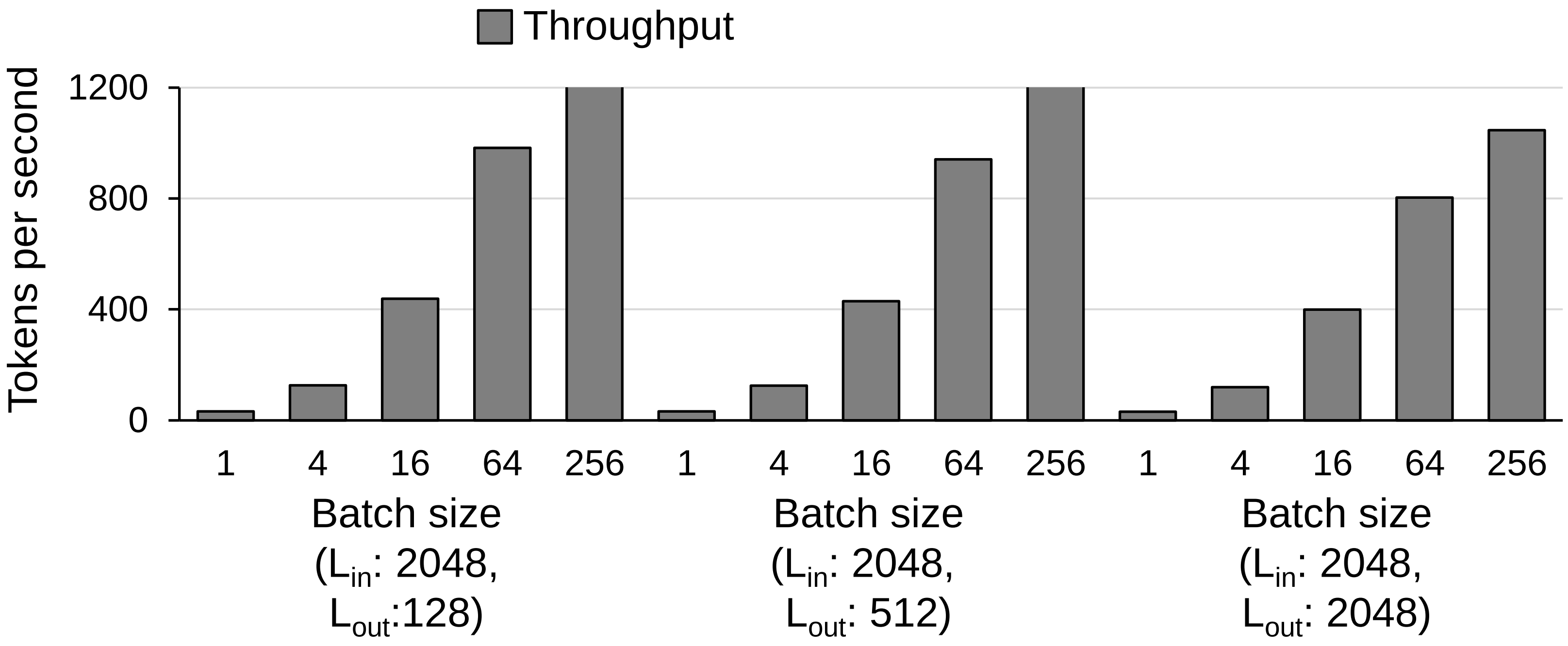
- **The attention layer** is still **memory-intensive**
  - The attention layer has unique **KV matrices per request**
  - The arithmetic intensity **remains nearly 1** regardless of the batch size



Key and Value (KV) matrices per request



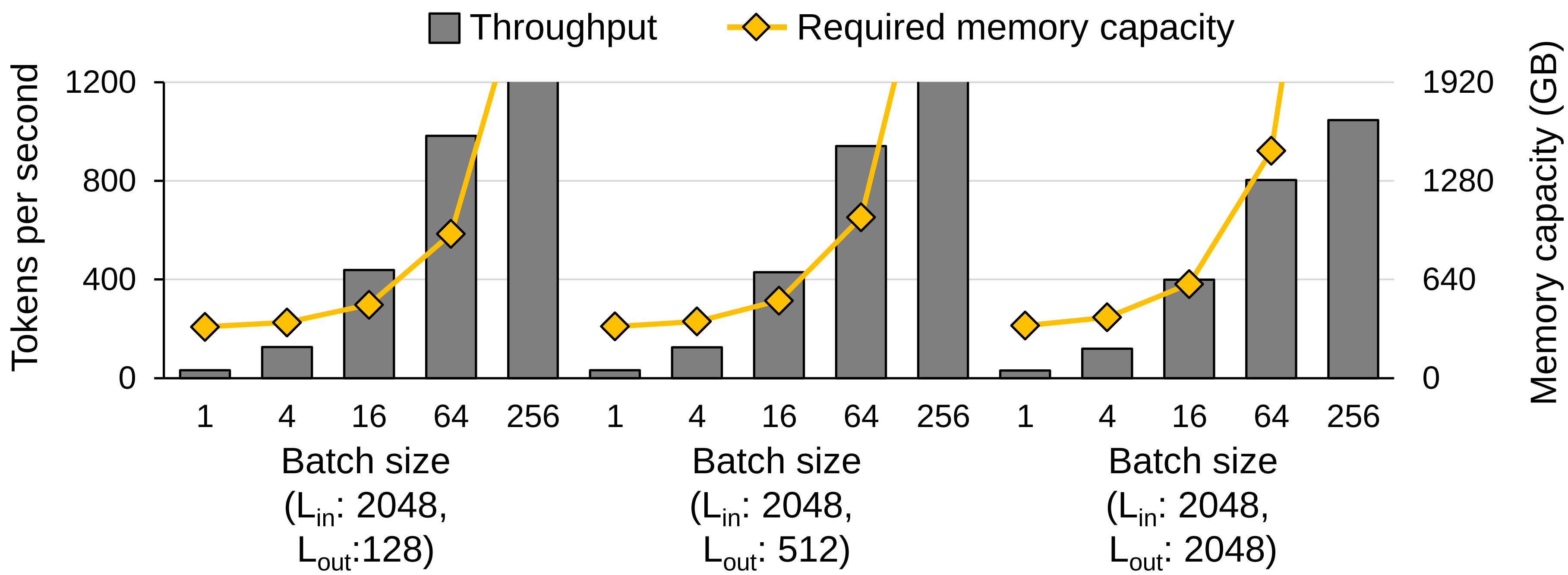
Challenges of Large Batch Size on Conventional Systems



Throughput per output token of TbGM inference
(GPT-3 175B, DGX A100 with HBM3)

Challenges of Large Batch Size on Conventional Systems

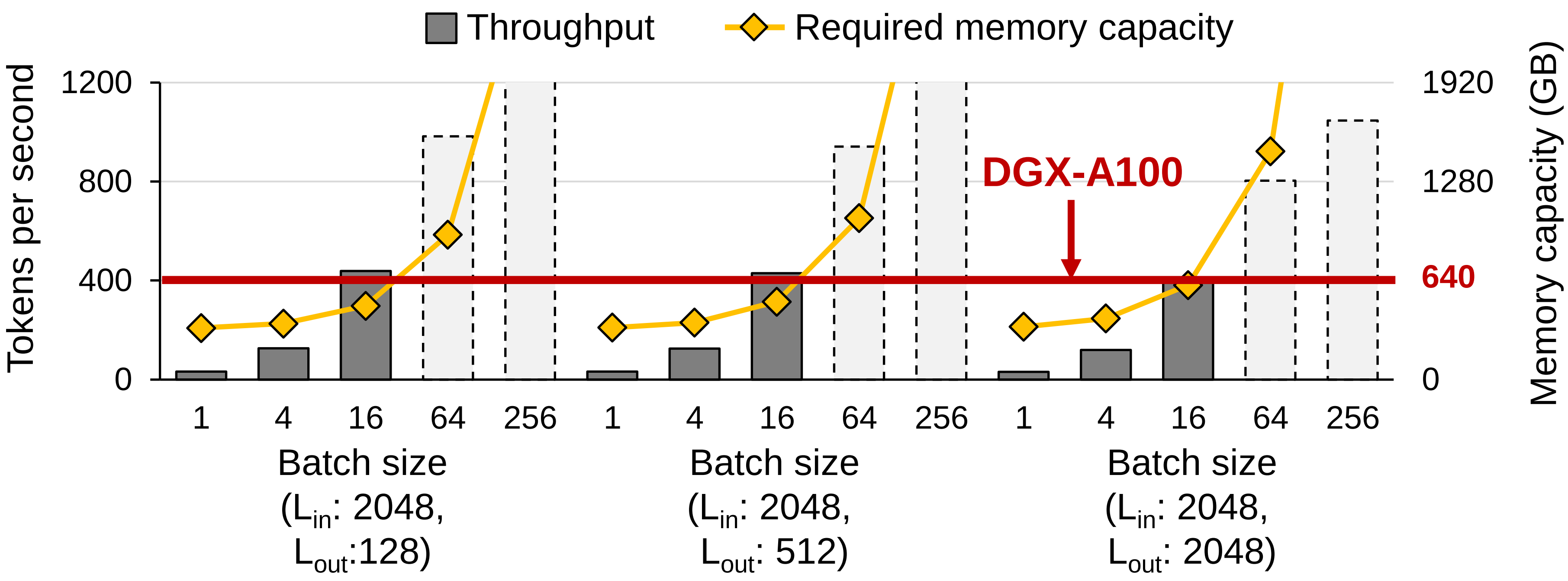
- **Large memory capacity requirement** from KV matrices
  - KV matrices require more memory capacity in proportion to batch size.



Throughput and required memory capacity per output token of TbGM inference (GPT-3 175B, DGX A100 with HBM3)

Challenges of Large Batch Size on Conventional Systems

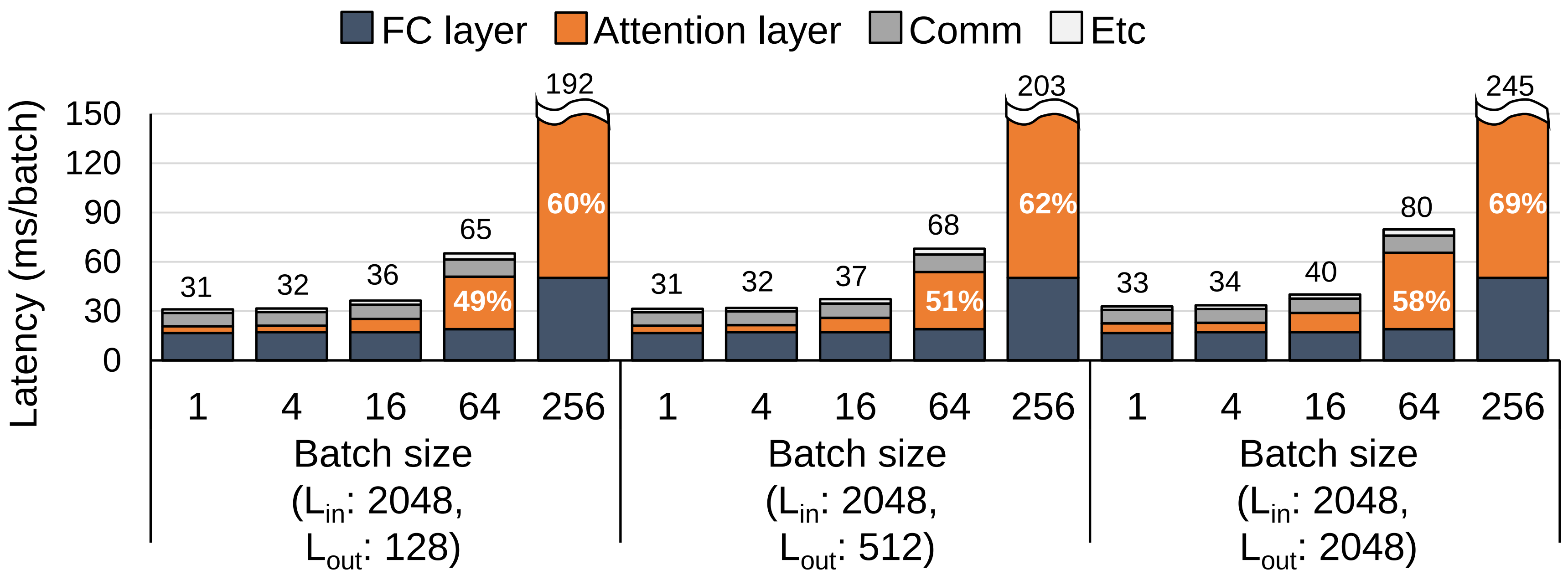
- **Large memory capacity requirement** from KV matrices
  - KV matrices require more memory capacity in proportion to batch size.



Throughput and required memory capacity per output token of TbGM inference
(GPT-3 175B, DGX A100 with HBM3)

Challenges of Large Batch Size on Conventional Systems

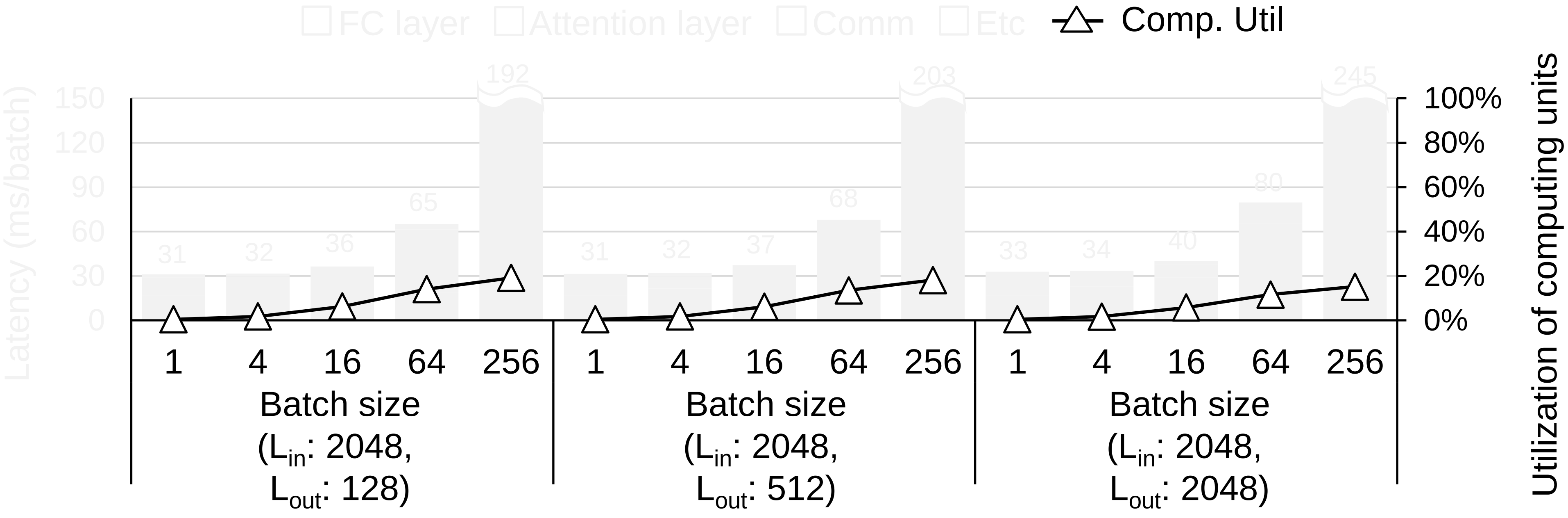
- **Long latency** of the attention layer
  - The latency of the attention layer increases linearly with batch size.
  - It can limit batch sizes under service level objectives (SLOs).



The Gen stage time breakdown and compute utilization
(GPT-3 175B, DGX-unlimited memory capacity)

Challenges of Large Batch Size on Conventional Systems

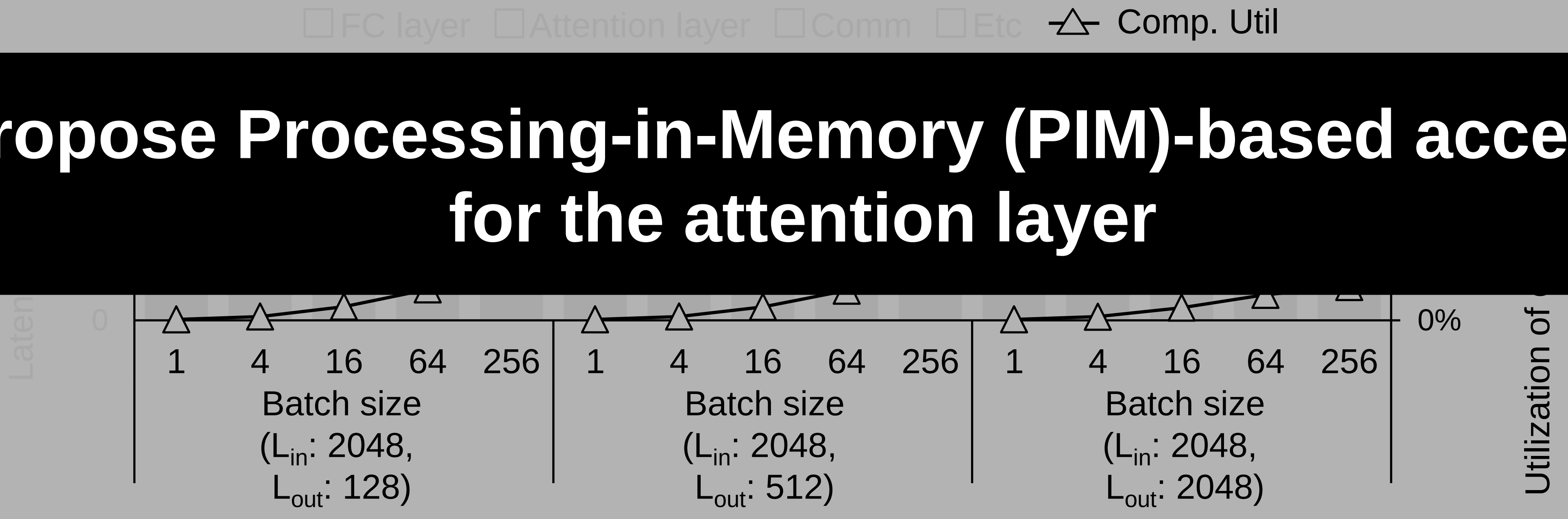
- **Low utilization of computing units** from attention layer



The Gen stage time breakdown and compute utilization
(GPT-3 175B, DGX-unlimited memory capacity)

Challenges of Large Batch Size on Conventional Systems

- **Low utilization of computing units** from attention layer

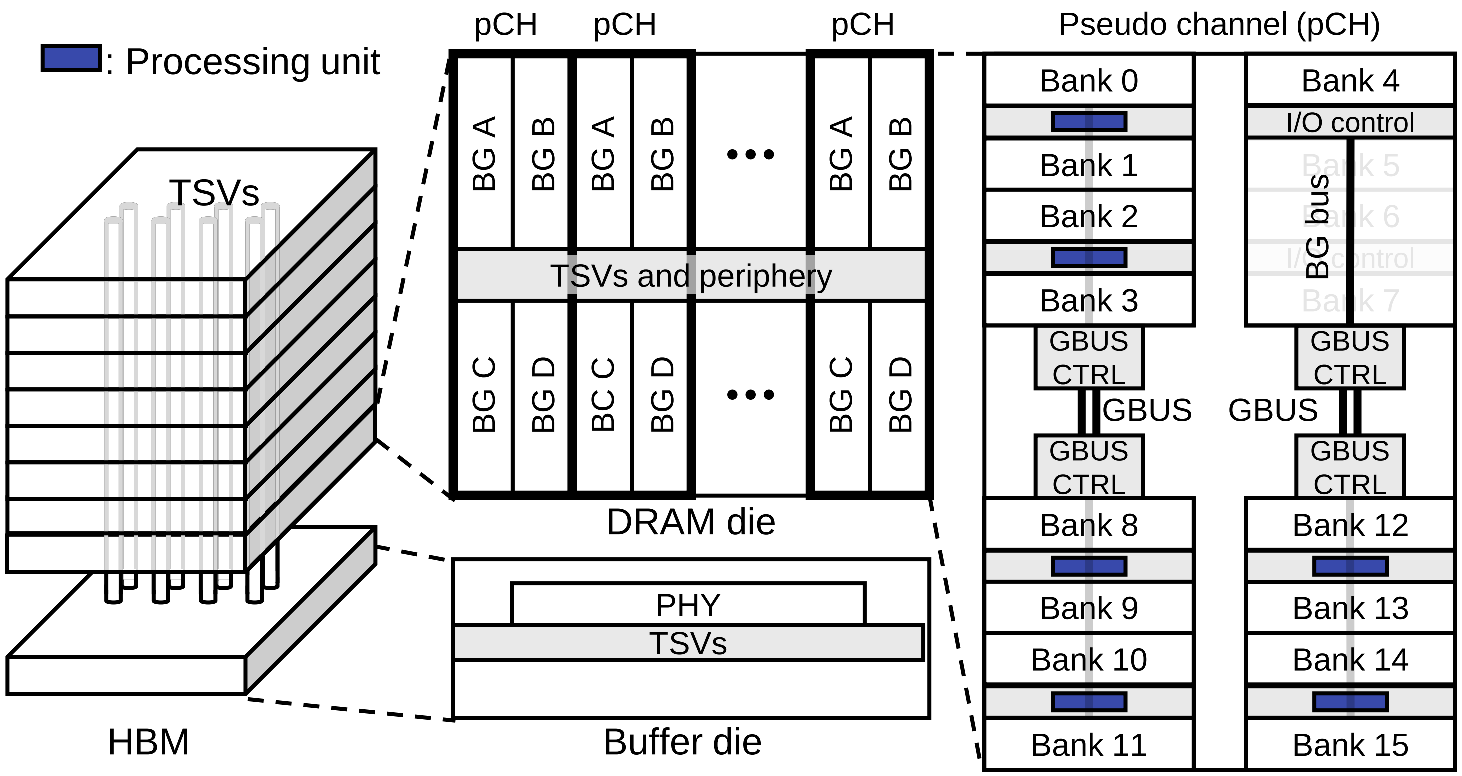


We propose Processing-in-Memory (PIM)-based accelerator for the attention layer

The Gen stage time breakdown and compute utilization
(GPT-3 175B, DGX-unlimited memory capacity)

Processing-In-Memory (PIM)

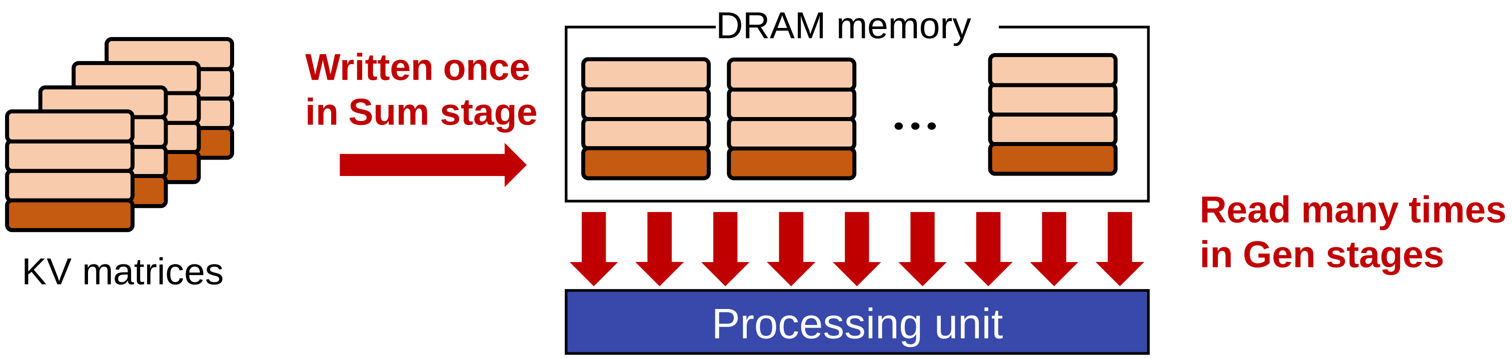
- PIM exploits abundant internal bandwidth to processing units (PUs) closer to the memory.



Why Processing-In-Memory (PIM) for the Attention Layer?

- High memory bandwidth requirement
  - The attention layer is **memory-intensive GEMV**
  - The size of the KV matrices is **too large to be cached**

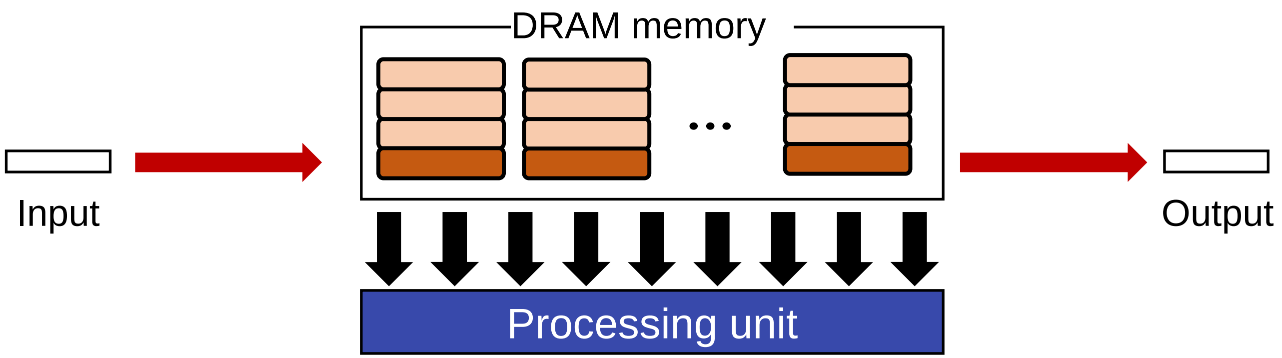
- Relatively low external bandwidth requirement
  - KV matrices are **written once** in the Sum stage and **read many** times in Gen stages



Why Processing-In-Memory (PIM) for the Attention Layer?

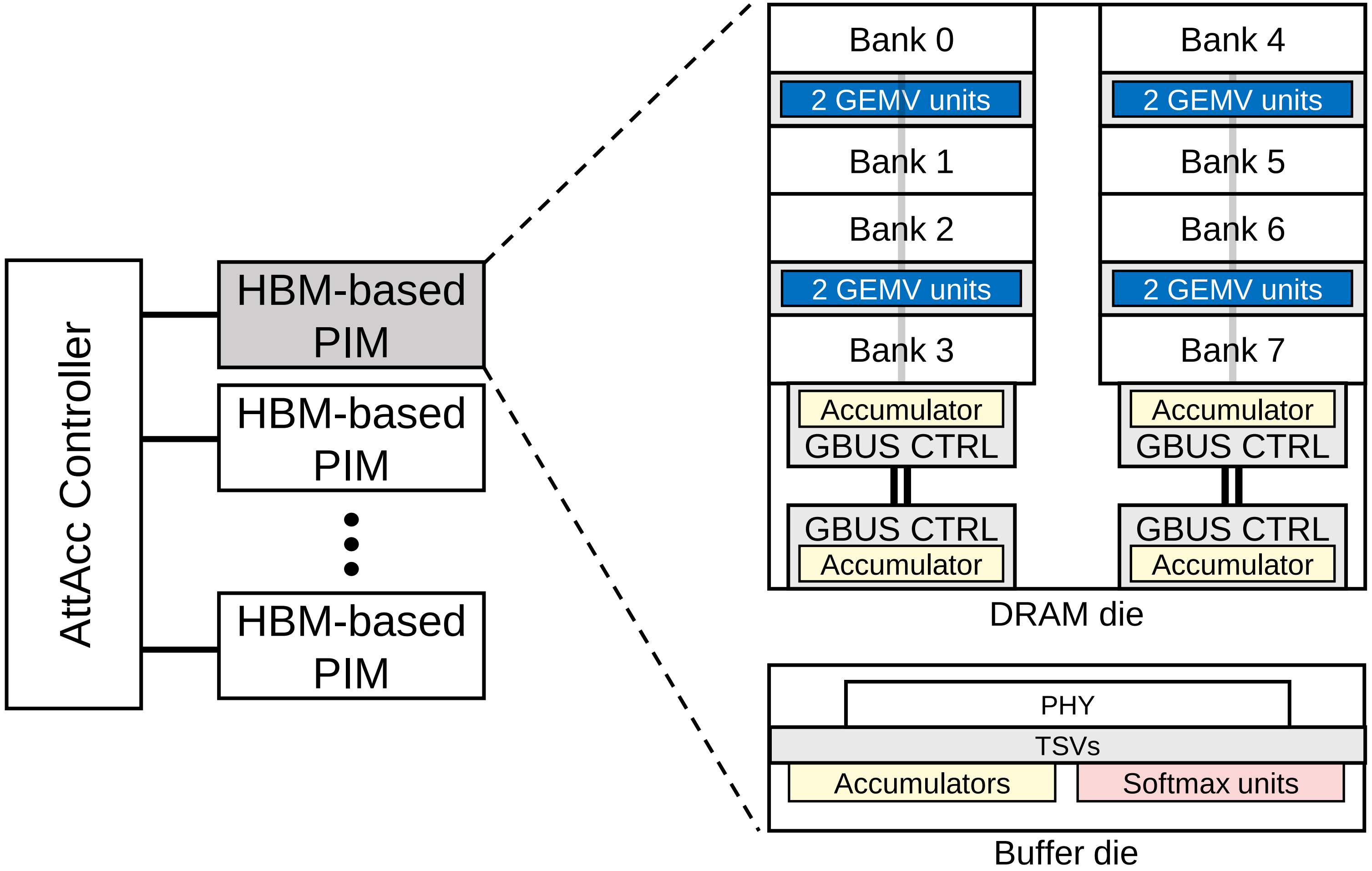
- High memory bandwidth requirement
  - The attention layer is **memory-intensive GEMV**
  - The size of the KV matrices is **too large to be cached**

- Relatively low external bandwidth requirement
  - KV matrices are **written once** in the Sum stage and **read many** times in Gen stages
  - The input and output of the attention layer are vectors that are **much smaller than KV matrices**.



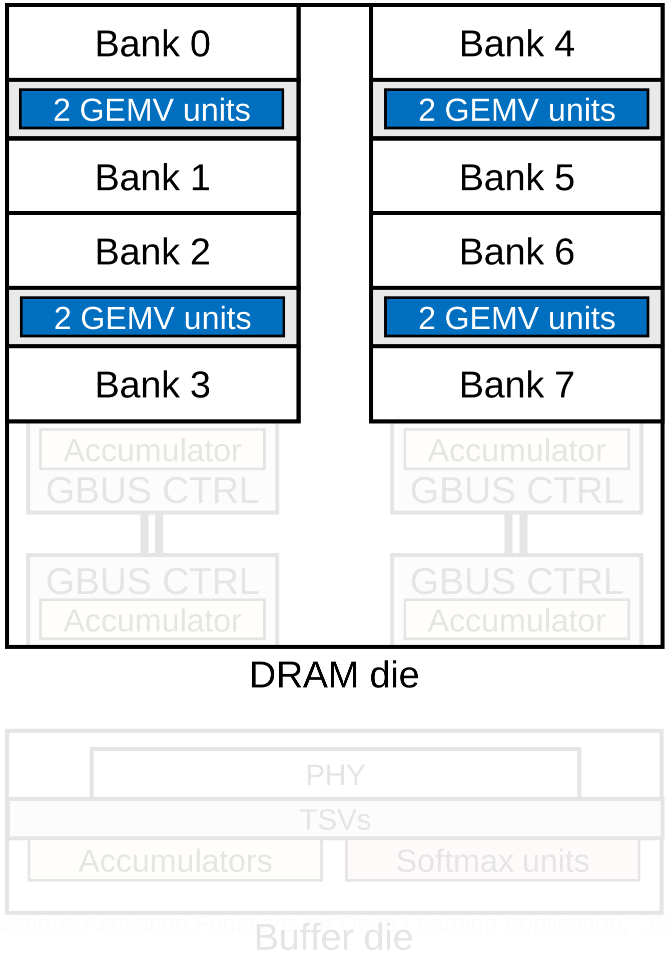
AttAcc: PIM-based Attention Accelerator

- We propose ***AttAcc***, which consists of HBM-based PIMs and a controller
- HBM-based PIM has
  - GEMV units
  - Softmax unit
  - Accumulators



AttAcc: GEMV Unit

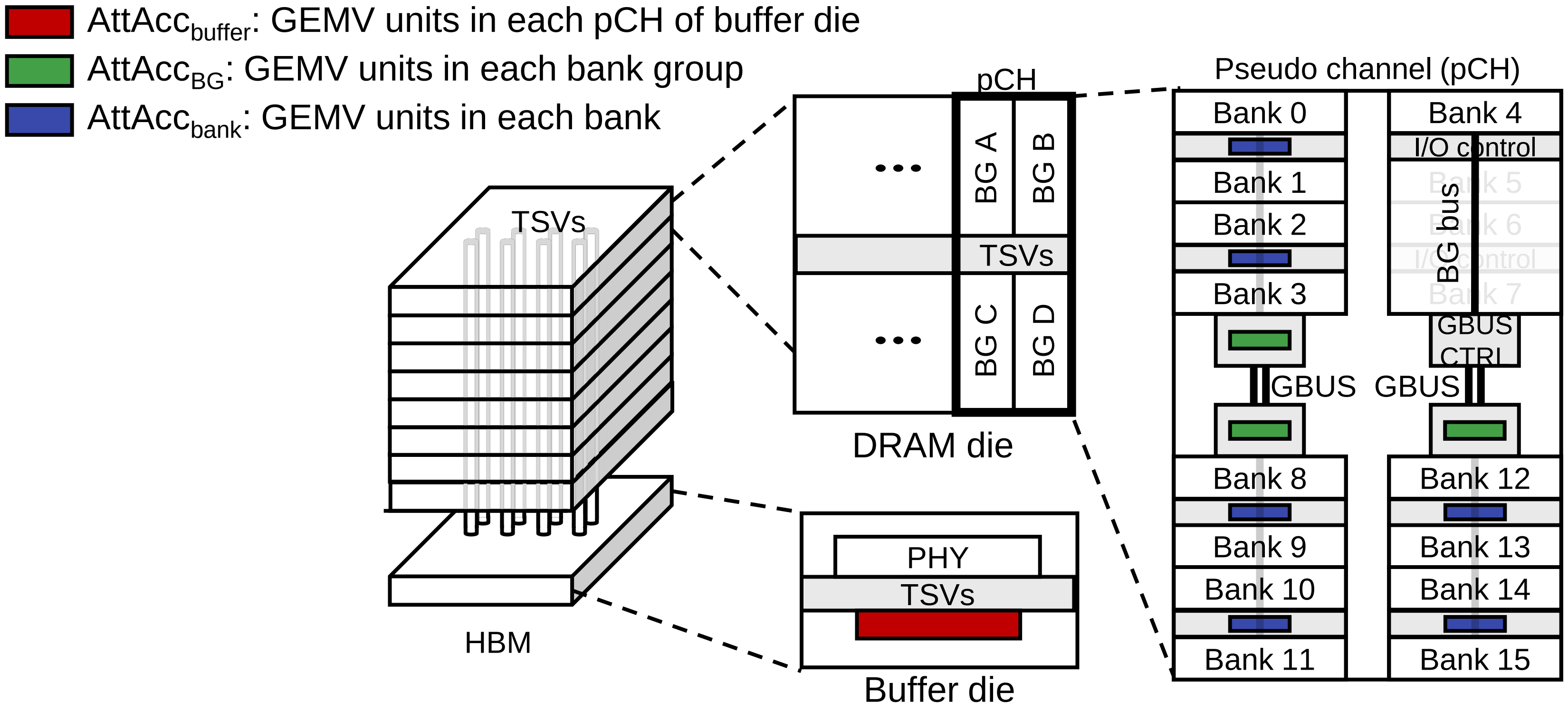
- Placed on each bank similar to Samsung HBM-PIM [1] and Hynix AiM [2]



[1] S Lee et al., "Hardware Architecture and Software Stack for PIM Based on Commercial DRAM Technology," ISCA, 2021
[2] D Kwon et al., "A 1ynm 1.25V 8Gb 16Gb/s/Pin GDDR6-Based Accelerator-in-Memory Supporting 1TFLOPS MAC Operation a ... 2023

AttAcc: GEMV Unit

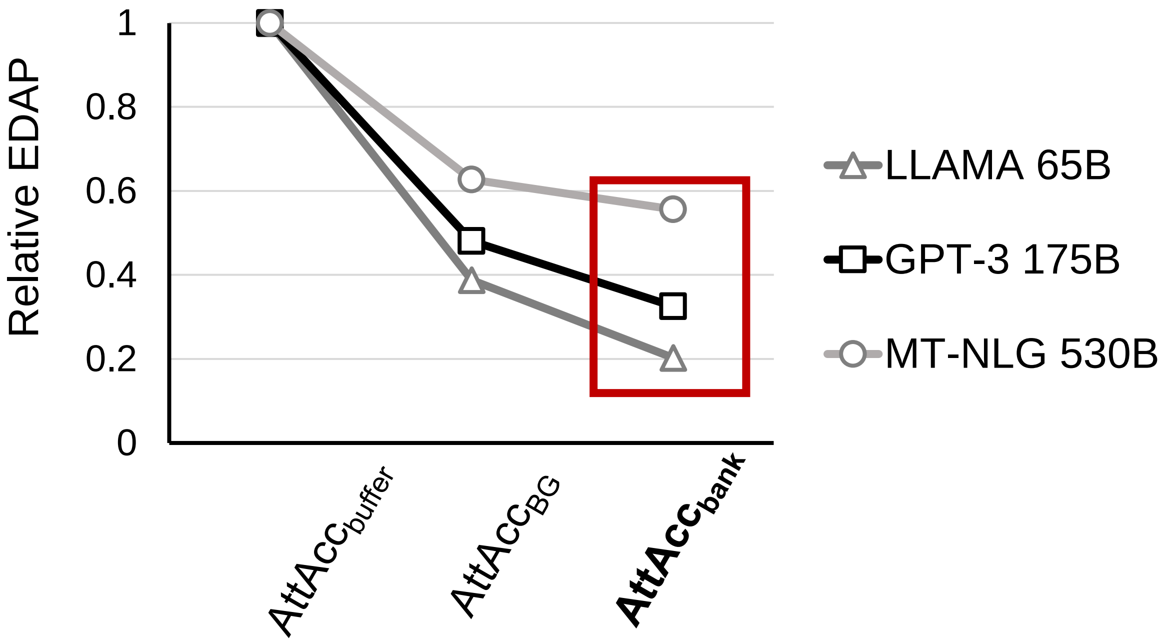
- Placed on each bank similar to Samsung HBM-PIM [1] and Hynix AiM [2]
  - *$AttAcc_{Buffer}$ vs $AttAcc_{BG}$ vs $AttAcc_{Bank}$*



[1] S Lee et al., "Hardware Architecture and Software Stack for PIM Based on Commercial DRAM Technology," ISCA, 2021
[2] D Kwon et al., "A 1ynm 1.25V 8Gb 16Gb/s/Pin GDDR6-Based Accelerator-in-Memory Supporting 1TFLOPS MAC Operation and Various Activation Functions for Deep Learning Application," JSSC, 2023

AttAcc: GEMV Unit

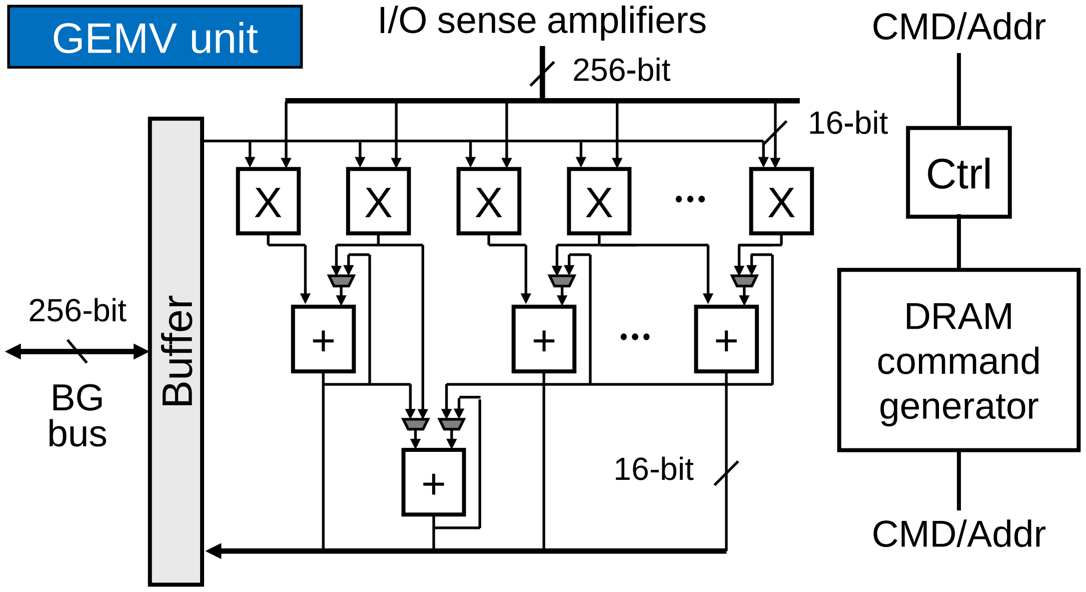
- Placed on each bank similar to Samsung HBM-PIM [1] and Hynix AiM [2]
  - *AttAcc$_{Buffer}$ vs AttAcc$_{BG}$ vs AttAcc$_{Bank}$*

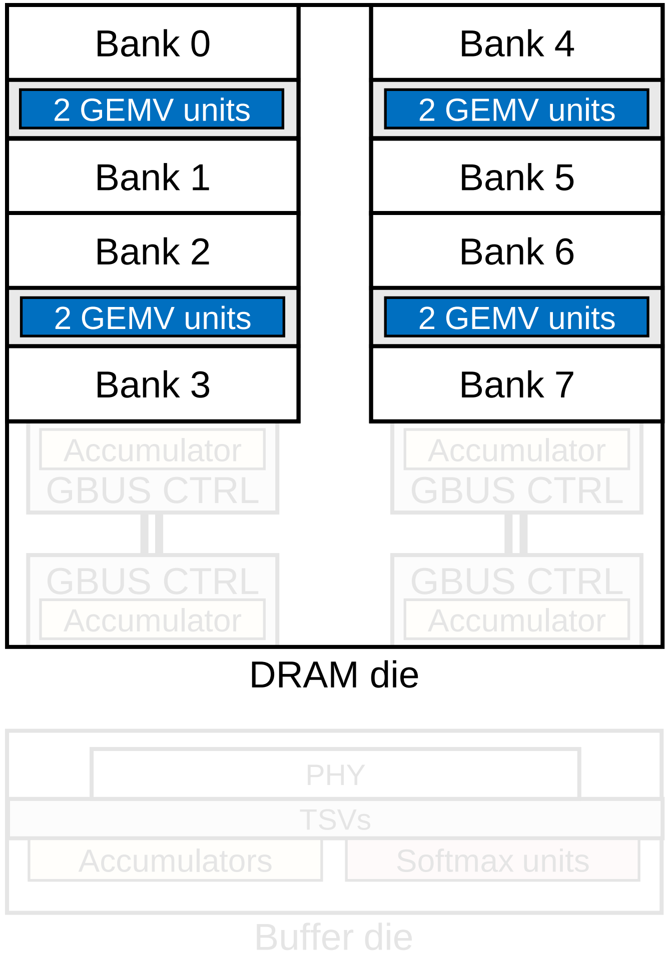


[1] S Lee et al., "Hardware Architecture and Software Stack for PIM Based on Commercial DRAM Technology," ISCA, 2021
[2] D Kwon et al., "A 1ynm 1.25V 8Gb 16Gb/s/Pin GDDR6-Based Accelerator-in-Memory Supporting 1TFLOPS MAC Operation and Various Activation Functions for Deep Learning Application," JSSC, 2023

AttAcc: GEMV Unit

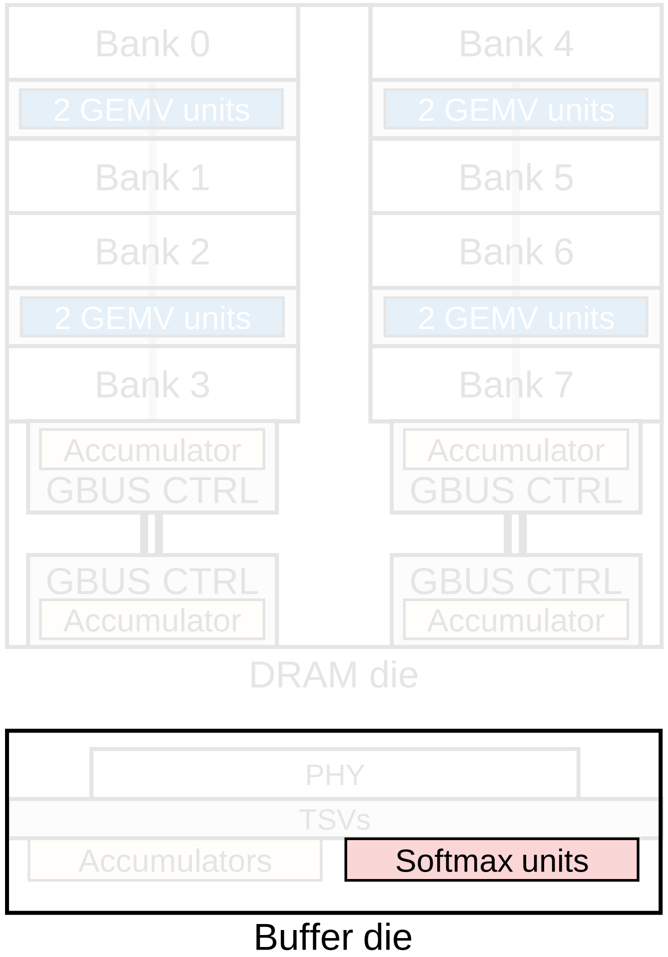
- Placed on each bank similar to Samsung HBM-PIM [1] and Hynix AiM [2]
- FP16 multipliers, FP16 adders, buffer for input vectors, and control unit.





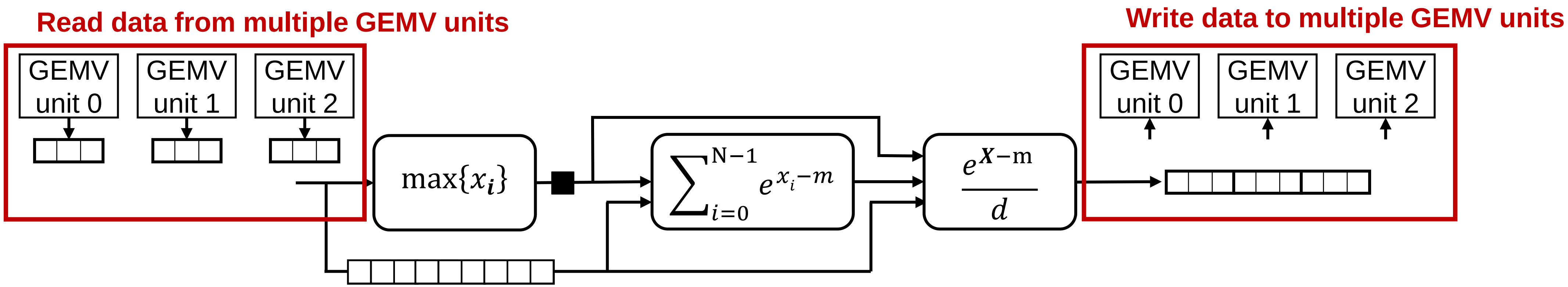
AttAcc: Softmax Unit

- Placed on buffer die



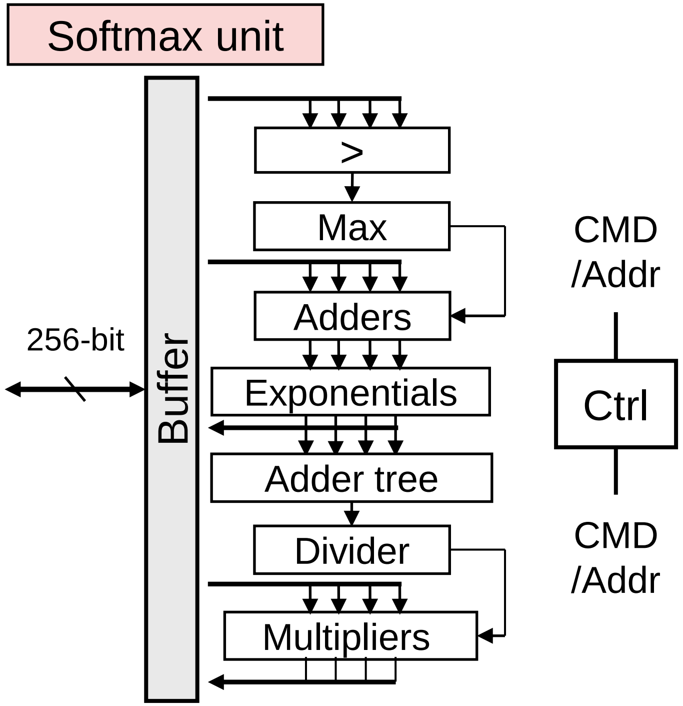
AttAcc: Softmax Unit

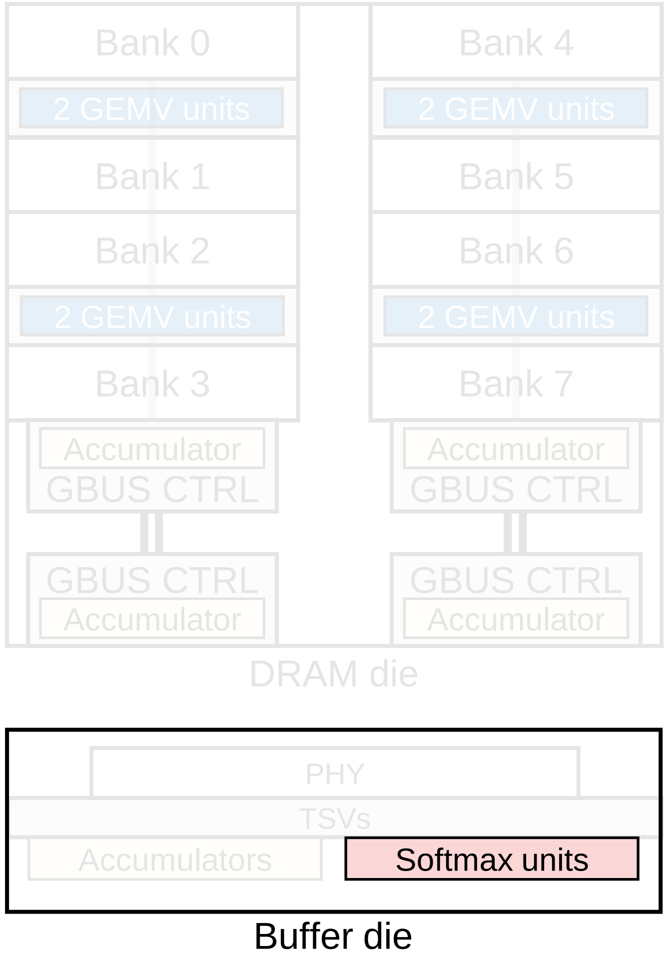
- Placed on buffer die
  - Communication with multiple GEMV units is required
  - Complex processing units and requirement for large SRAM buffers for intermediate vectors
  - Placing softmax unit on DRAM die is overkill.
    - = For GPT-3 175B, the FLOPs of softmax is **50 times smaller** than GEMVs in the attention layer



AttAcc: Softmax Unit

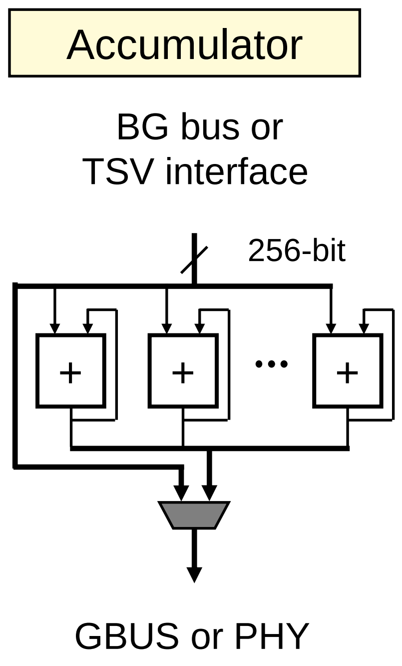
- Placed on buffer die
- Processing units such as exponents, multipliers, and adders supporting FP32
- Buffer for intermediate vectors and control unit.

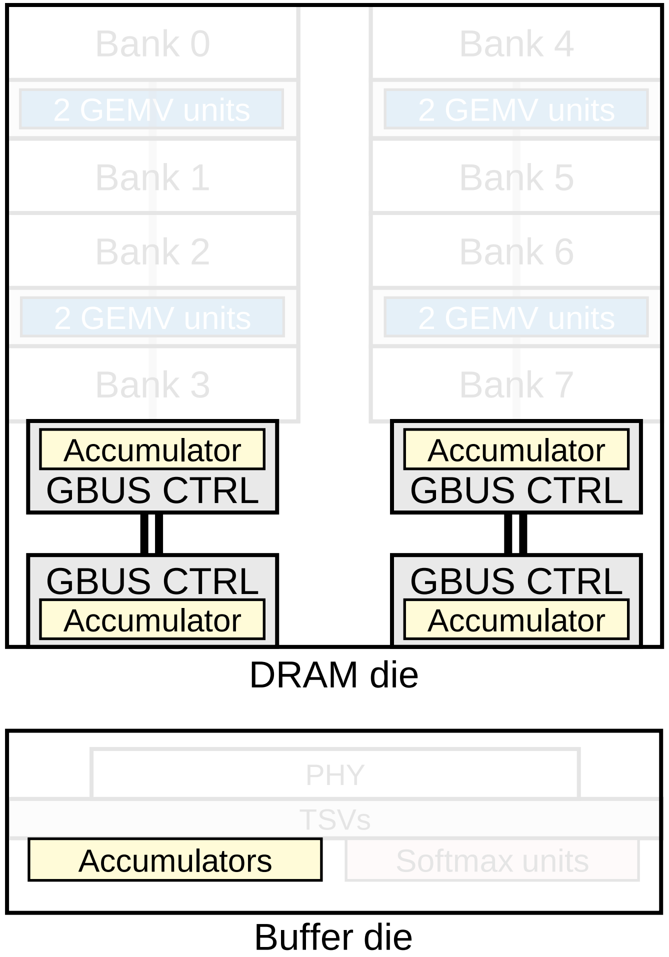




AttAcc: Accumulator

- Placed hierarchically between the GEMV and the softmax units
- Supports the reduction of partial results from different GEMV units

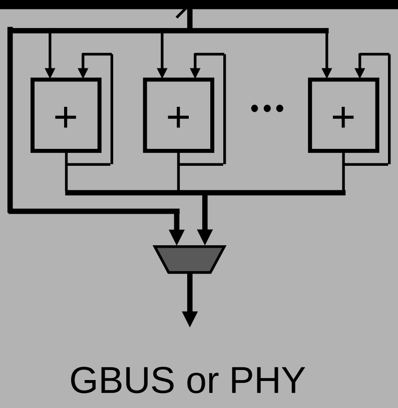


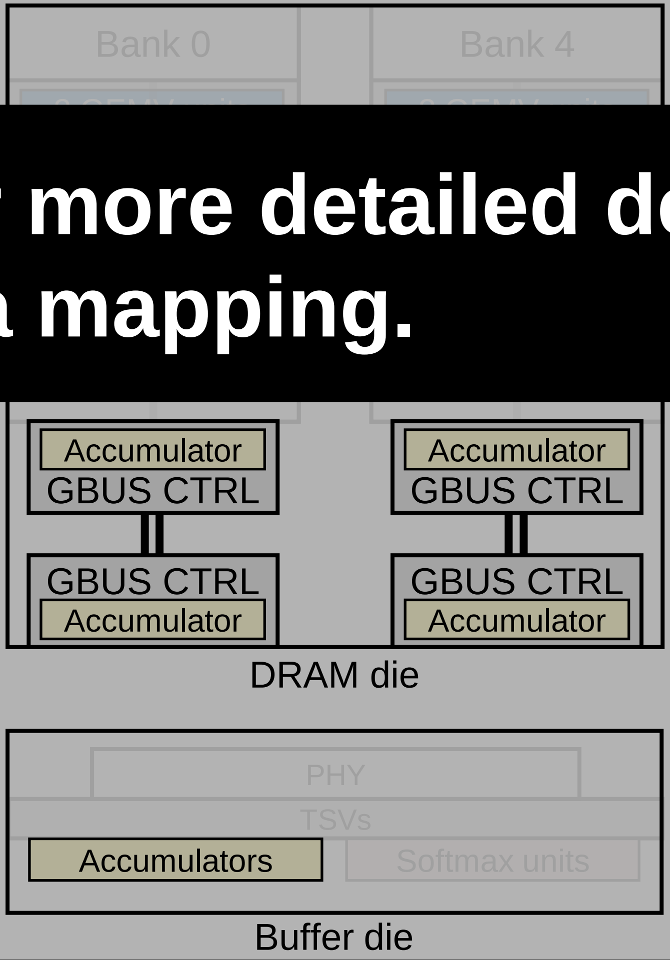


AttAcc: Accumulator

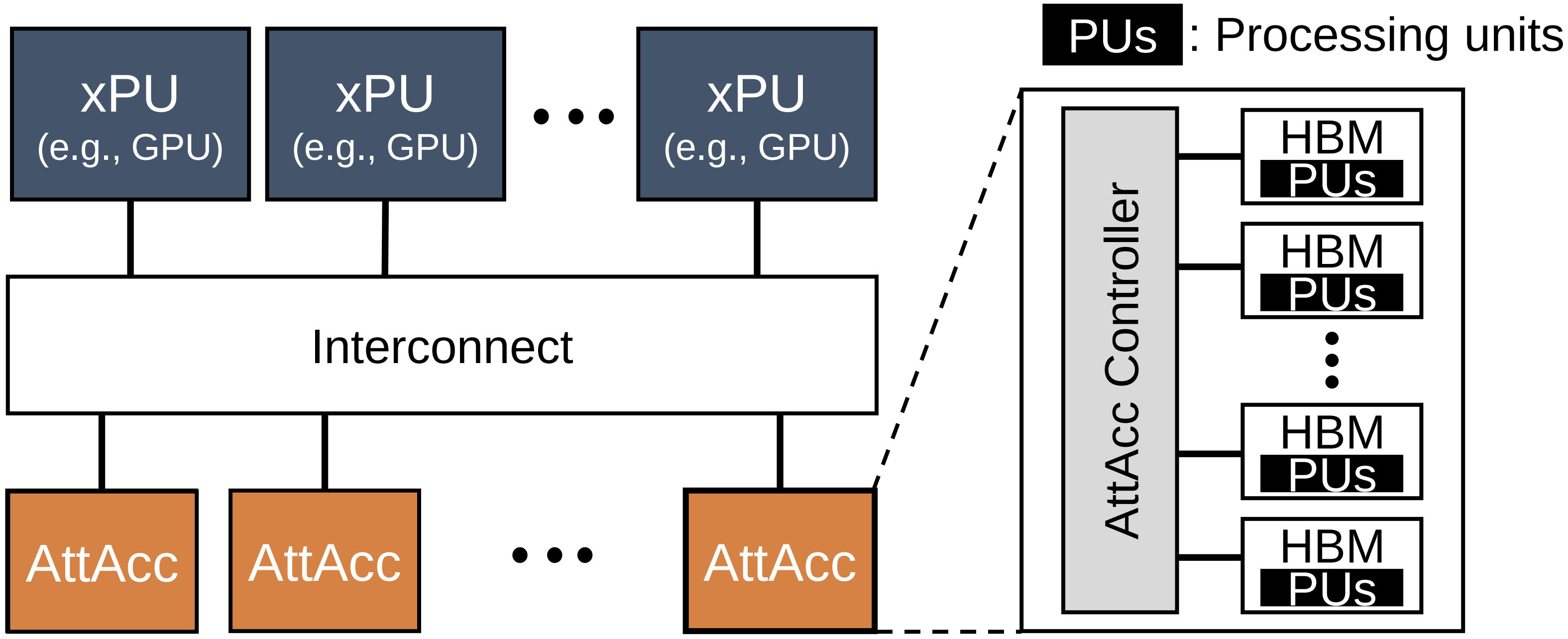
- Placed hierarchically between the GEMV and the softmax units
- Supports the reduction of partial results from different GEMV units

Please refer to the full paper for more detailed design exploration and data mapping.





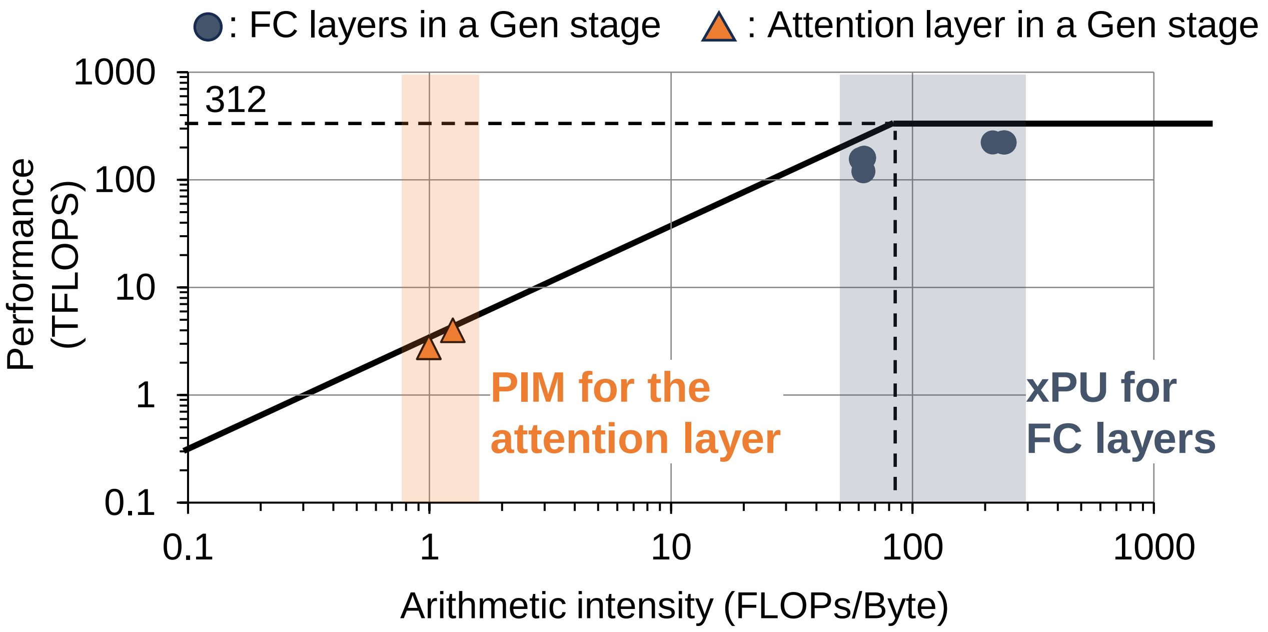
Heterogeneous System with *AttAcc*



Heterogeneous system with xPUs and *AttAccs*

Heterogeneous System with *AttAcc*

- High performance
  - High computing power of **xPU for batched FC layers** with high FLOPs/Byte
  - Amplified memory bandwidth of **PIM for the attention layer** with low FLOPs/Byte



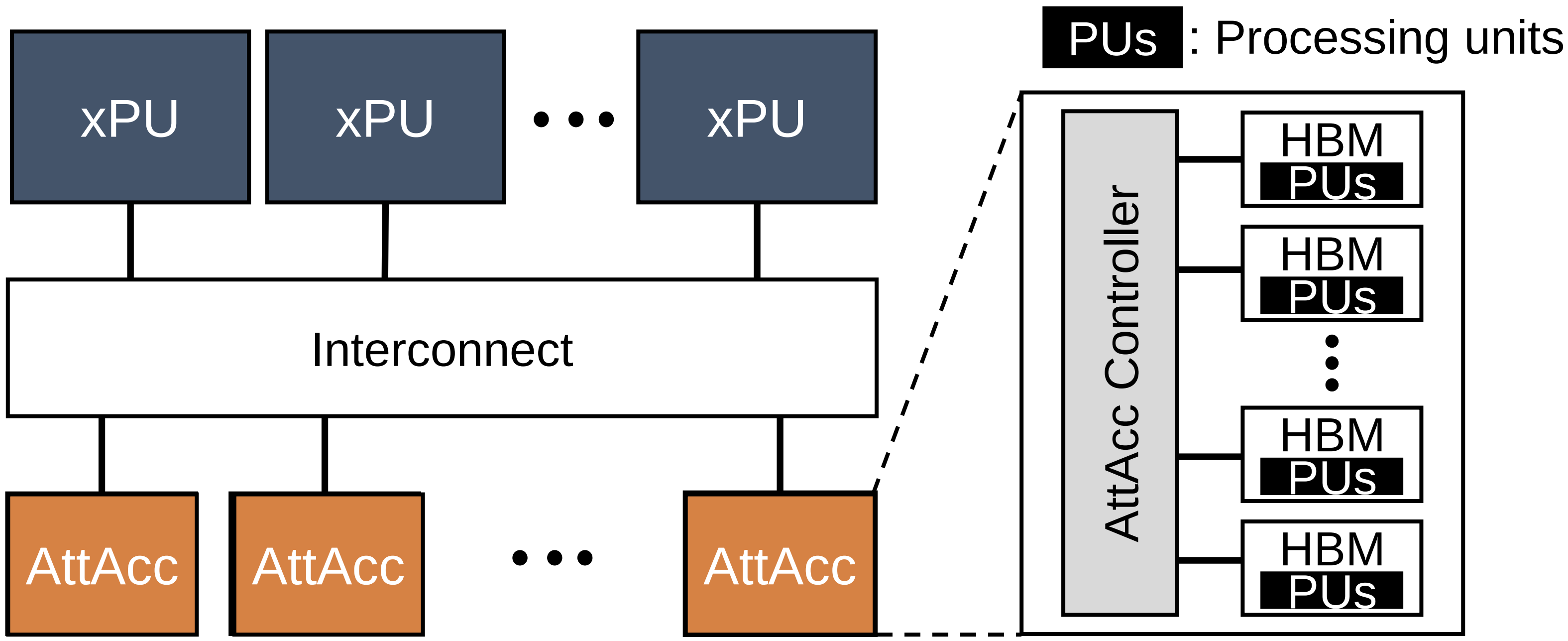
Roofline model of the DGX-A100 with HBM3

Heterogeneous System with *AttAcc*

- High performance
  - High computing power of **xPU for batched FC layers** with high FLOPs/Byte
  - Amplified memory bandwidth of **PIM for the attention layer** with low FLOPs/Byte
- High energy efficiency
  - Leveraging **high reusability of weights** of batched FC layers through on-chip caches in xPU
  - Benefit from **short data transfer** in PIM for the attention layer

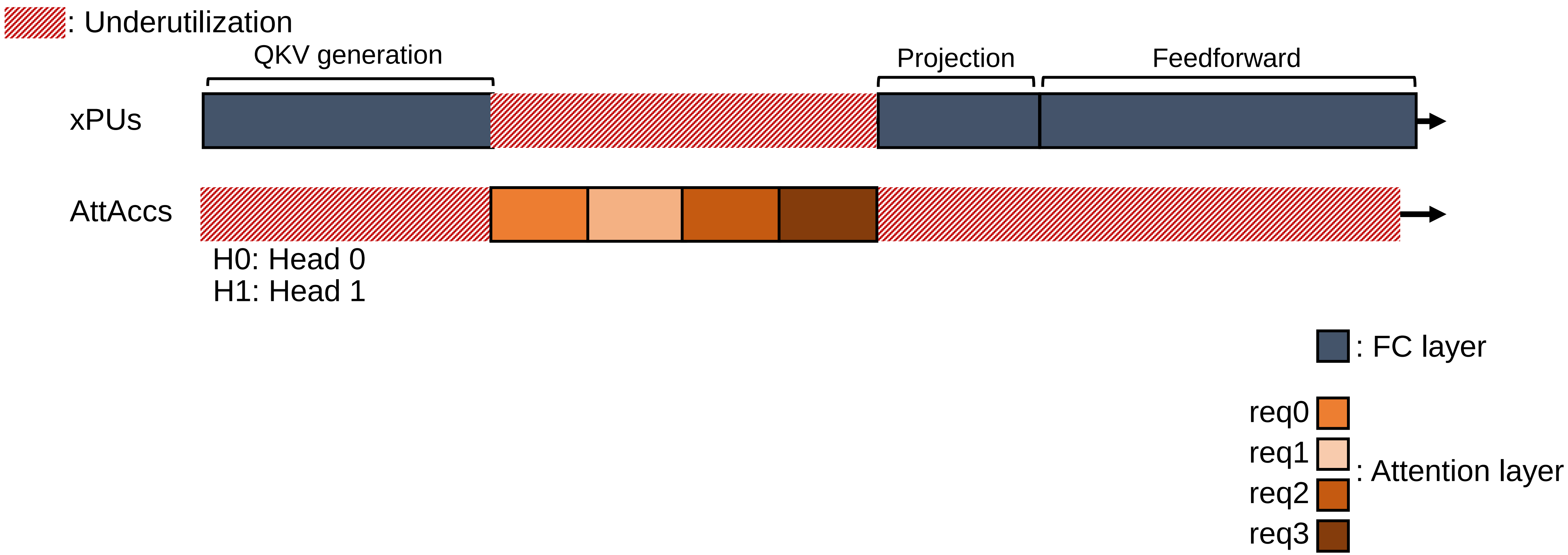
Heterogeneous System with ***AttAcc***

- Proposed system consists of multiple xPUs (e.g., GPU, TPU) and attention accelerators ***AttAcc***s
  - Batched FC layers on multiple xPUs
  - Attention layers on ***AttAcc***s
- ***AttAcc***s and xPUs can be connected via an interface such as NVLINK, PCIe, and CXL.

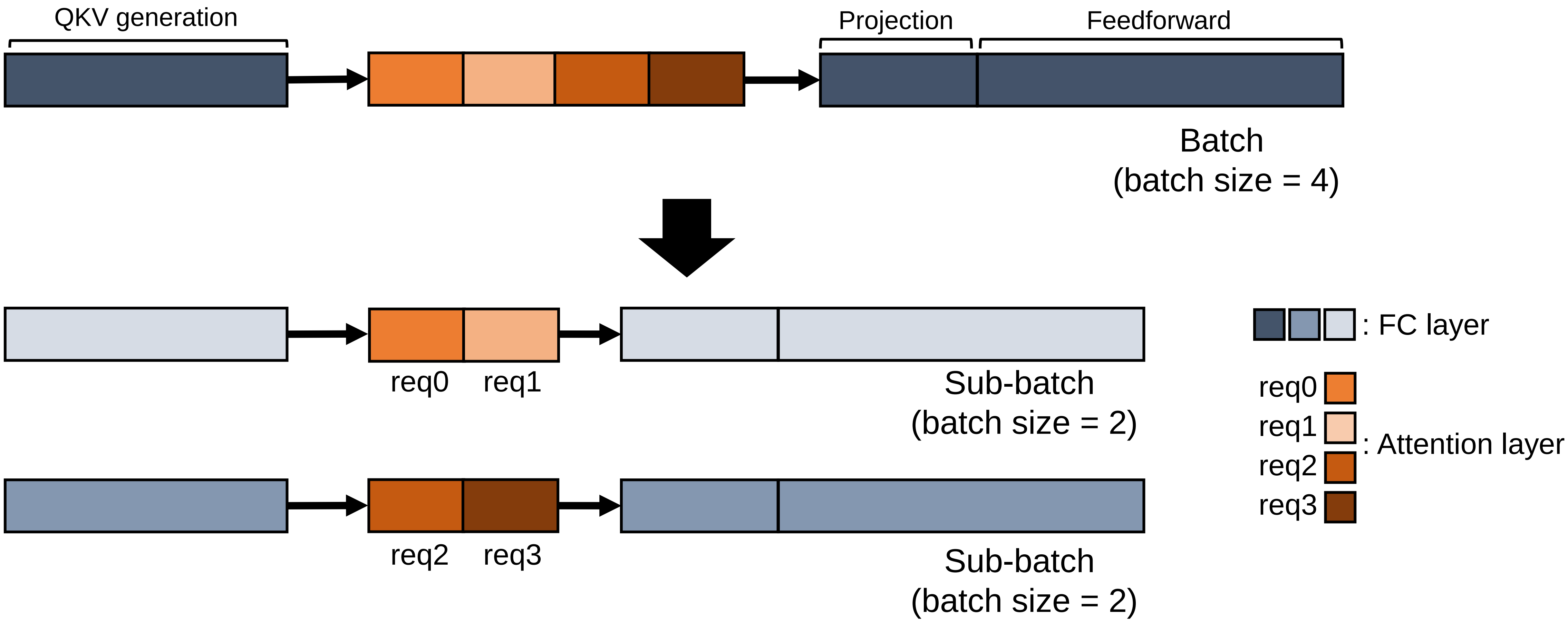


Heterogeneous system with xPUs and *AttAccs*

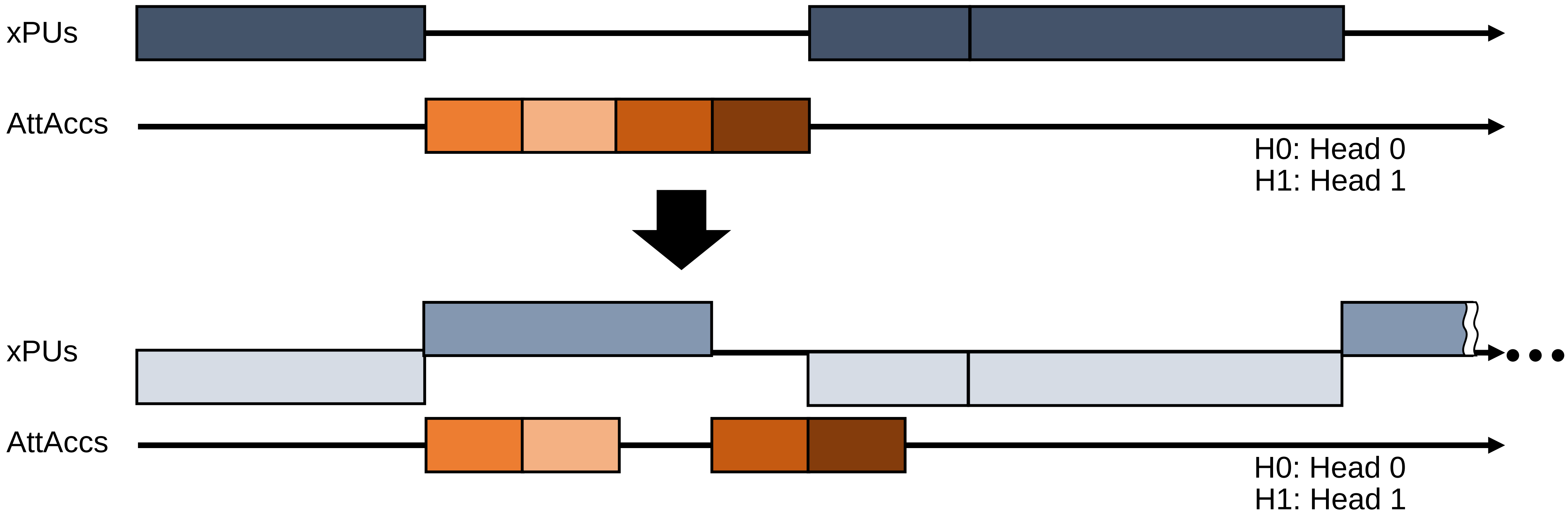
Execution Flow of the Heterogeneous System



Naïve Approach: Batch-level Pipelining

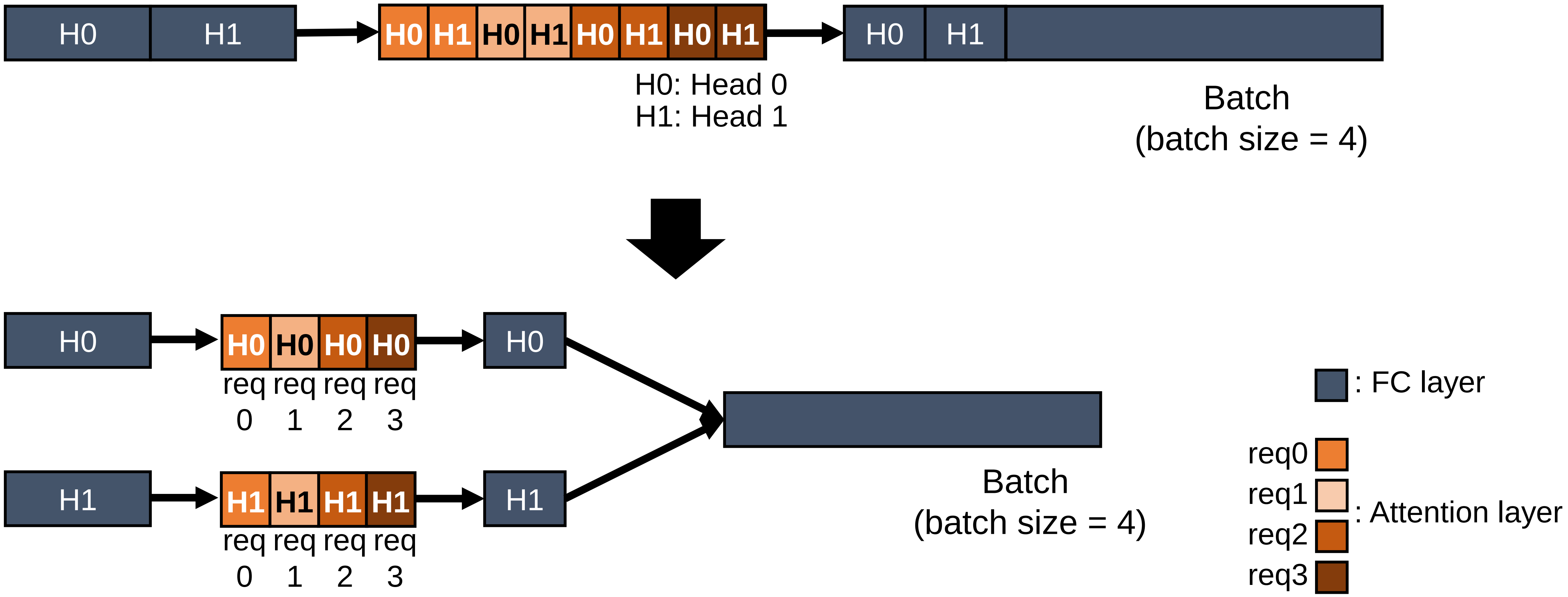


Naïve Approach: Batch-level Pipelining

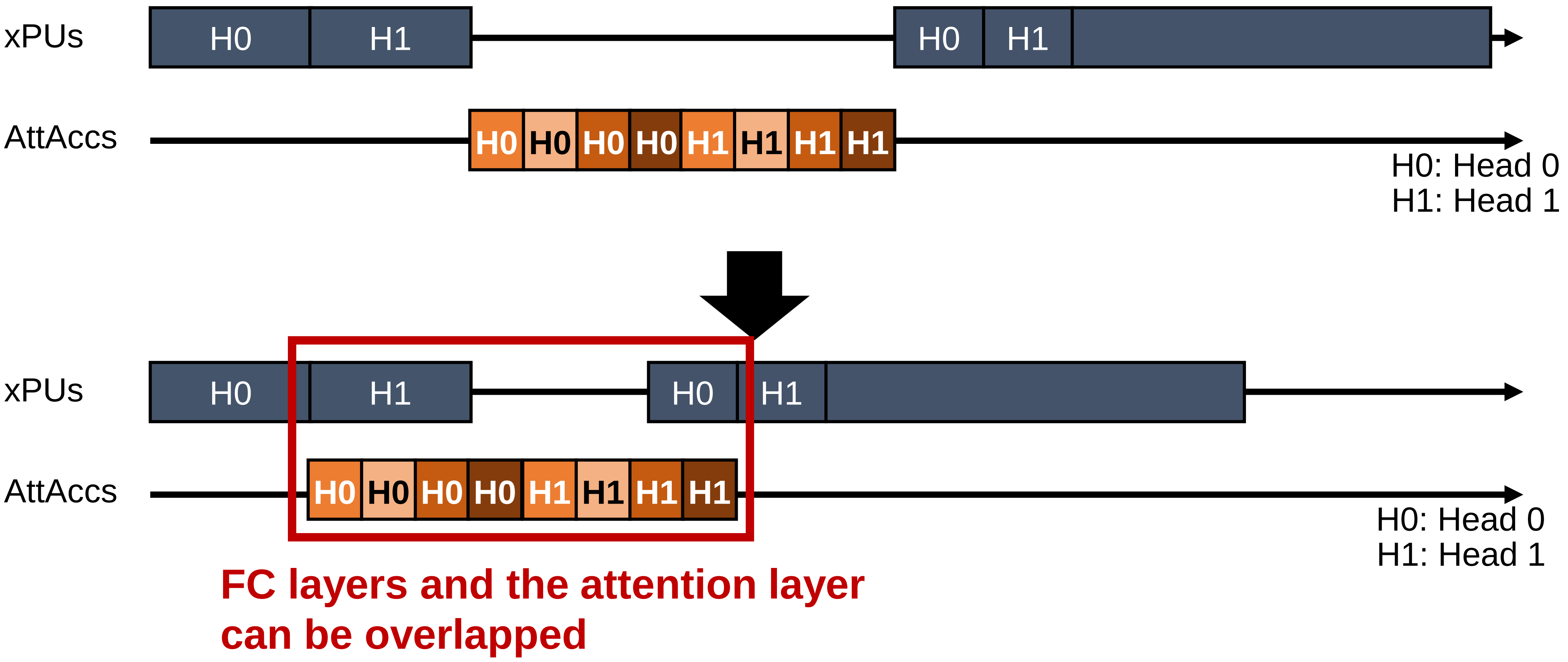


Head-level Pipelining

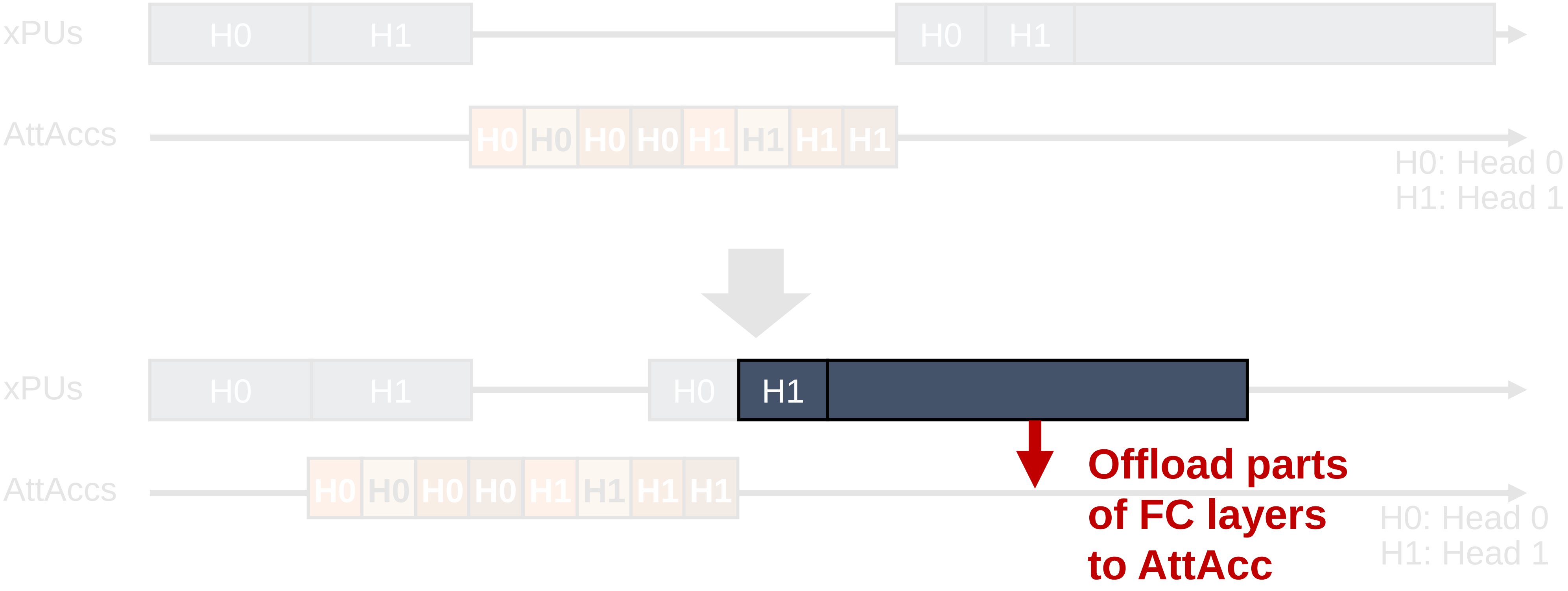
- FC layers that precede or follow the attention layer can be divided into heads.



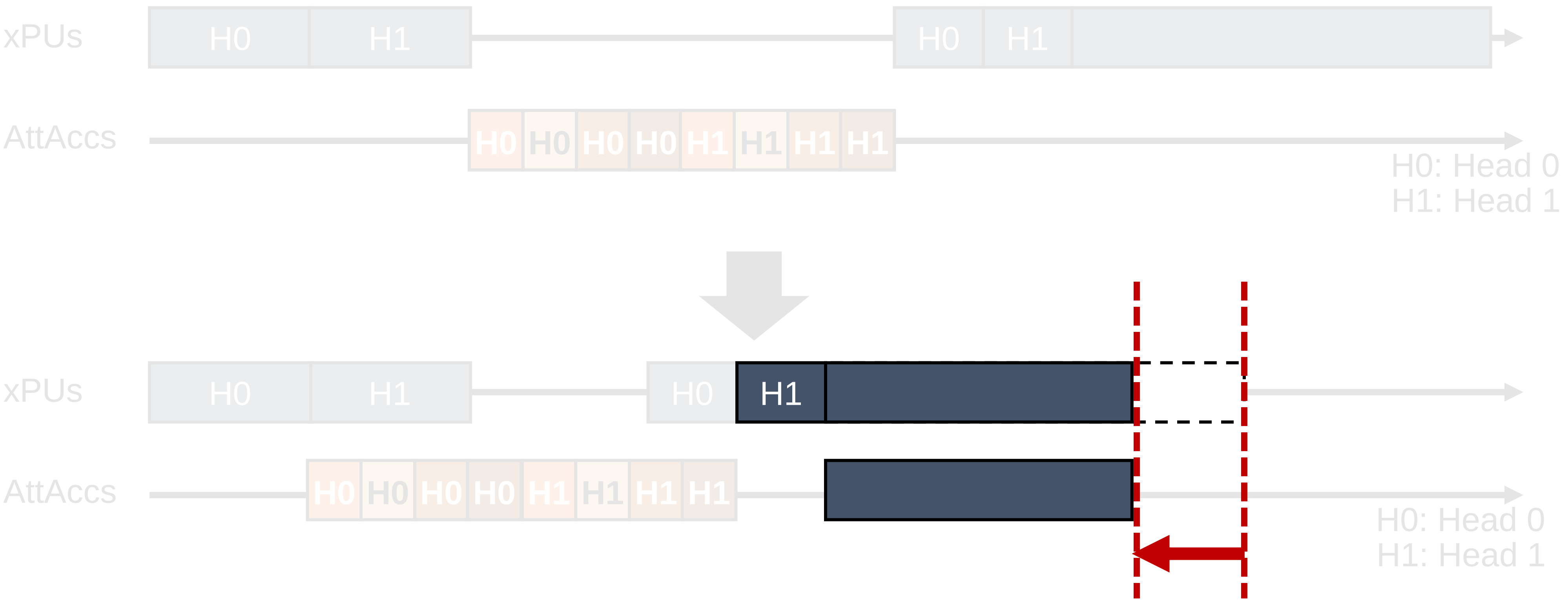
Head-level Pipelining



FeedForward Co-processing



FeedForward Co-processing



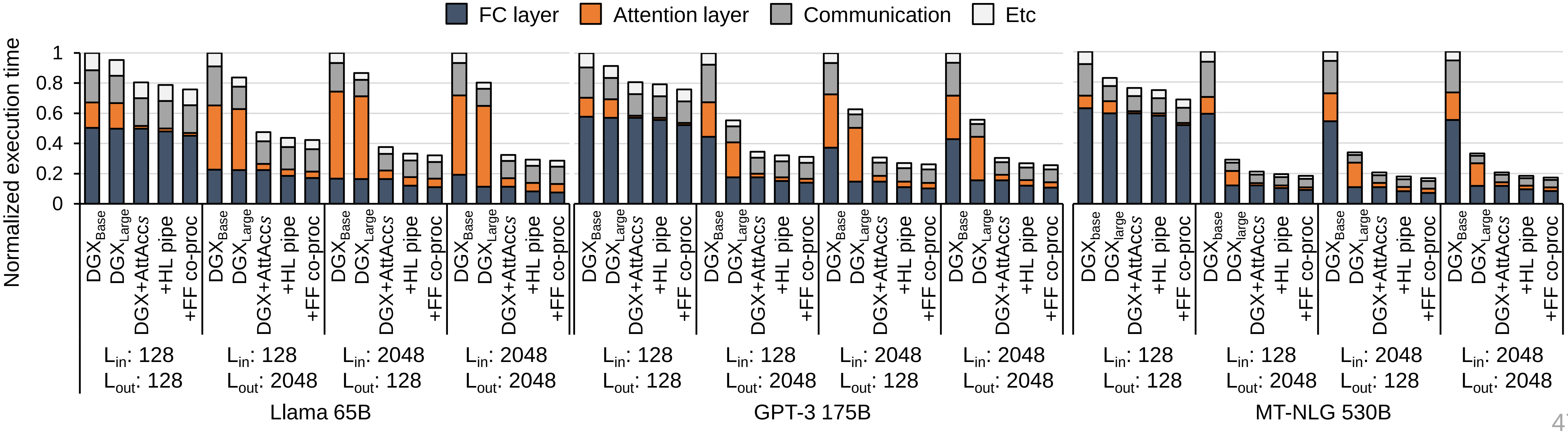
Experimental Setup

- Performance
  - Ramulator2 [1] and in-house simulator to evaluate ***AttAcc*** and DGX, respectively
- Energy and area
  - RTL synthesis for compute units and CACTI for buffer
  - The area overhead of ***AttAcc***s is 10.84% of a HBM.
    - = Scaling the area to DRAM process for units in DRAM die
- Target model
  - Various size of TbGMs: Llama 65B, GPT-3 175B, and MT-NLG 530B
- Comparison
  - DGX_{Base}: DGX-A100 having 40 HBM stacks
  - DGX_{Large}: DGX-A100 having 80 HBM stacks
  - *DGX*+***AttAcc***: DGX_{Base} + 8 ***AttAcc***s with 5 HBM stacks each

[1] H Luo et al., “Ramulator 2.0: A Modern, Modular, and Extensible DRAM Simulator,” arXiv, 2023.

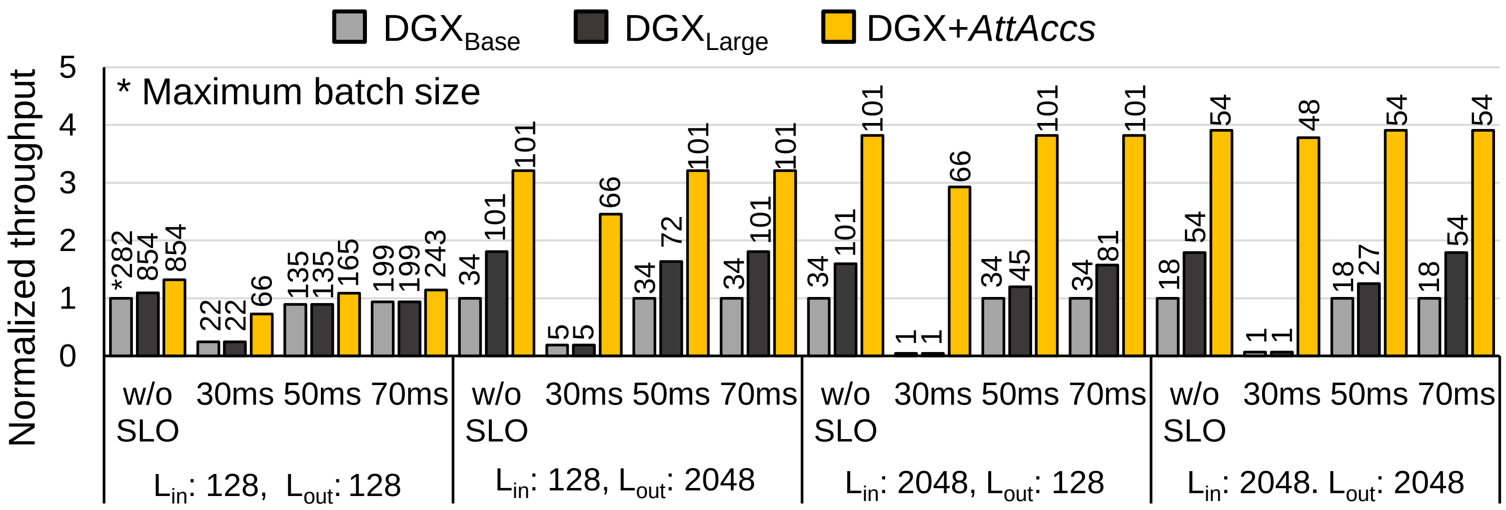
Evaluation (Performance)

- *DGX+**AttAcc**s* outperforms DGX_{Base} and even DGX_{Large} up to by 5.93x and 2.81x, repectively
  - 4.84x and 2.48x from ***AttAcc***
  - 1.15x from head-level pipelining
  - 1.10x from feedforward co-processing



Evaluation (Performance)

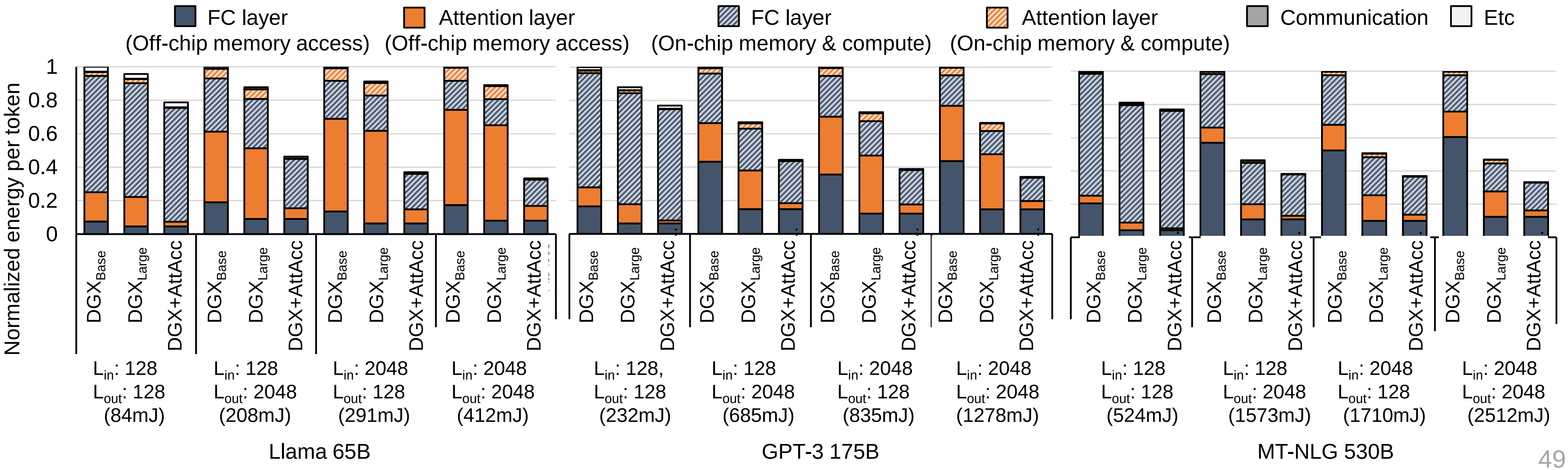
- *DGX+**AttAccs*** achieves further throughput improvement under SLO constraint
  - Performance improvement from relieving the batch size constraints caused by SLO



Normalized throughput of GPT-3 175B inference for various SLOs

Evaluation (Energy Efficiency)

- Energy consumption of *DGX+**AttAcc**s* compared to *DGX$_{Base}$* (*DGX$_{Large}$*) is reduced by up to
  - 66.7% (62.6%) for Llama 65B
  - 65.9% (48.8%) for GPT-3 175B
  - 66.8% (29.1%) for MT-NLG 530B



Conclusion

- We discovered that **the attention layer** poses a constraint on the batch size in conventional systems (e.g., DGX) due to the **long latency** and **memory capacity requirements**.

- We proposed a **heterogeneous system** (*DGX* + ***AttAcc****s*) with the conventional system for the batched FC layer and ***AttAcc****s* for the attention layer, leveraging PIM architecture.

- We explored GEMV unit placement and data mapping in the PIM architecture and proposed efficient pipelining and co-processing optimizations to improve system utilization.

- *DGX+AttAccs* achieved higher throughput (up to 2.81×) and energy efficiency (up to 2.67×) compared to the monolithic GPU system.

Thank you!

Question?