# Optimizing inference engine for large MoE language models: experience and lessons

Kun Tan

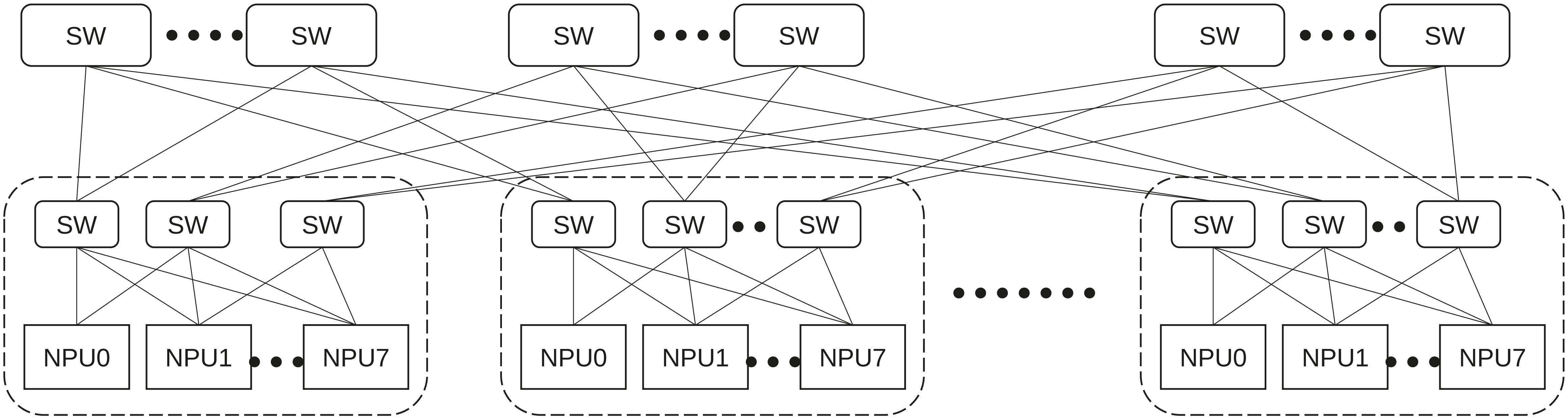
Director of Distributed and Parallel Software Lab
Huawei Technologies

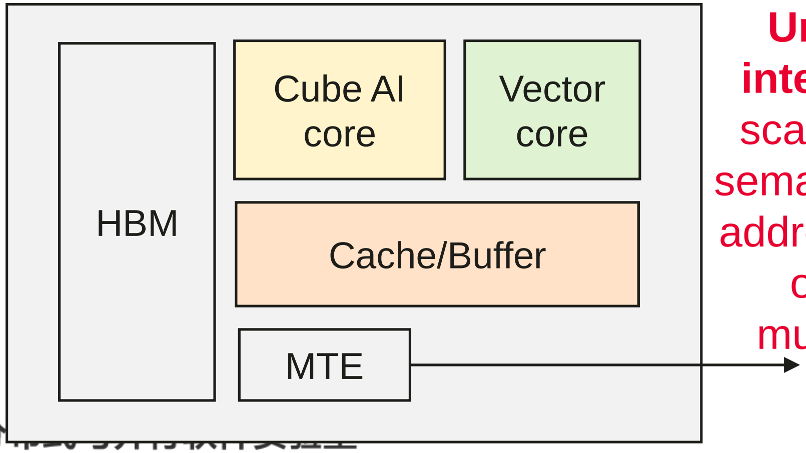
DP2E-AI 2025, Paris

分布式与并行软件实验室
Distributed and Parallel Software Laboratory

HUAWEI

# CloudMatrix Ascend Super-computer



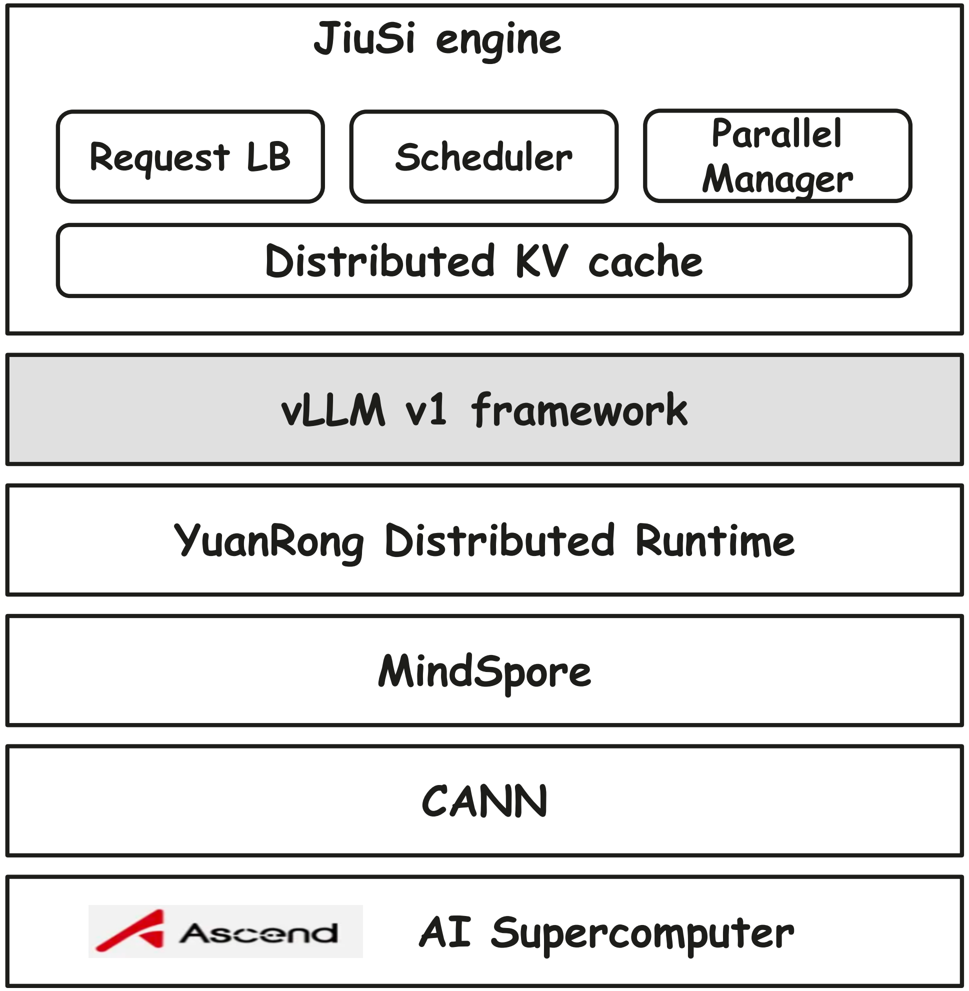




**Unified buffer interconnection**: scalable, memory semantics w/ unified address space, out-of-order and multipath native

| | NVL72 | CloudMatrix |
|---|---|---|
| # of node | 72 | 384 |
| Peak (Pflops) | 180 | 300 |
| Mem BW(TBps) | 7.9 (576) | 3.2 (1229) |
| Int. BW(TBps) | 1.8 (130) | 2.8 (269) |

# CloudMatrix Ascend Super-computer

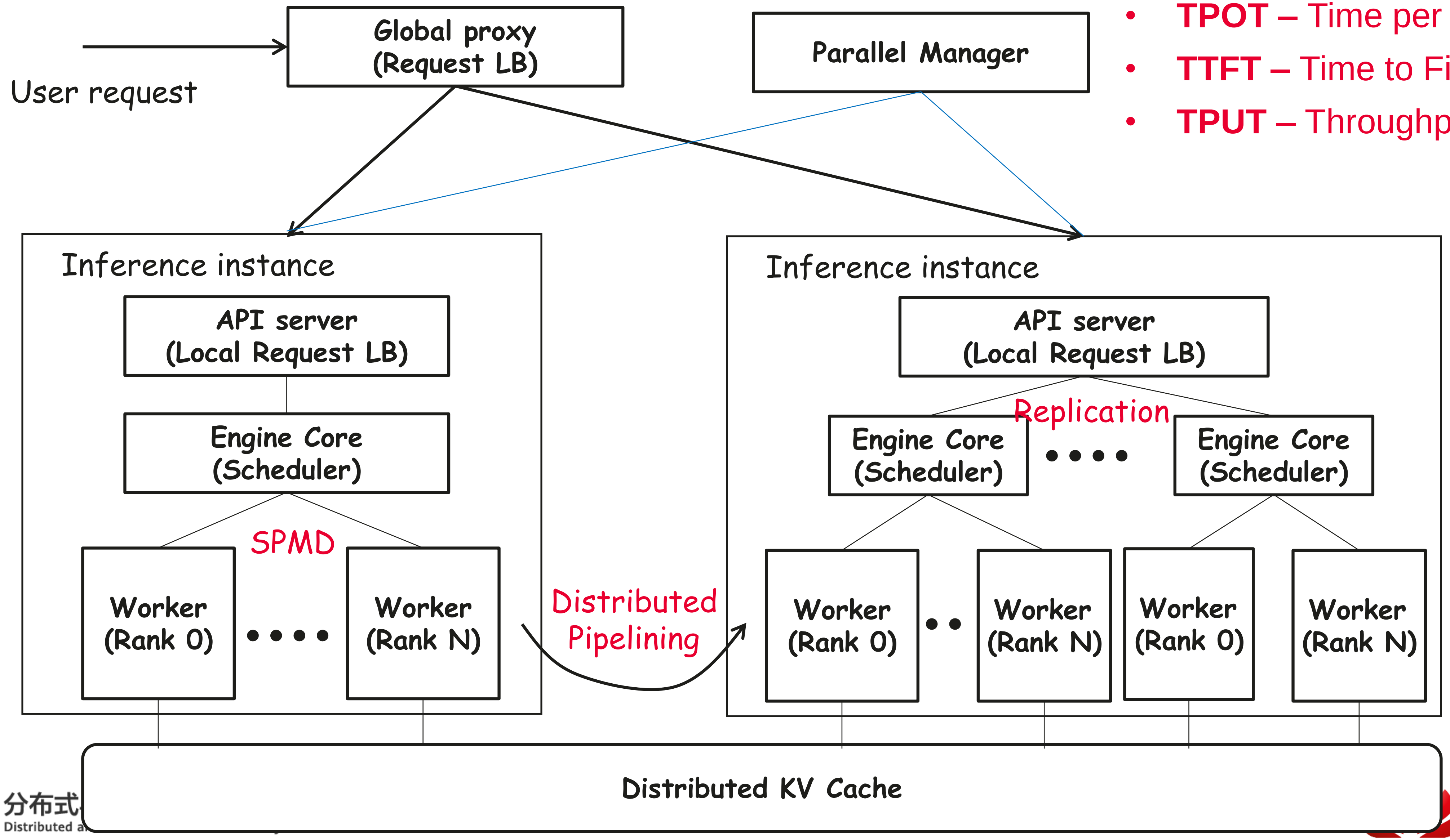
# JiuSi(九思): vLLM-based Ascend-affined inference engine




- **JiuSi (九思)**
  - Based on vLLM v1 architecture
    - An easy-hacking codebase
  - Distributed KV Cache centric
  - Enhanced request load-balancing, scheduling, and rich parallel mechanisms
  - Asynchronous processing optimizations to vLLM
- **YuanRong(元戎)**
  - A *serverless* runtime for clusters
    - Providing elasticity for JiuSi workers
  - Data system – a distributed object memory
    - An ideal substrate for distributed KVC

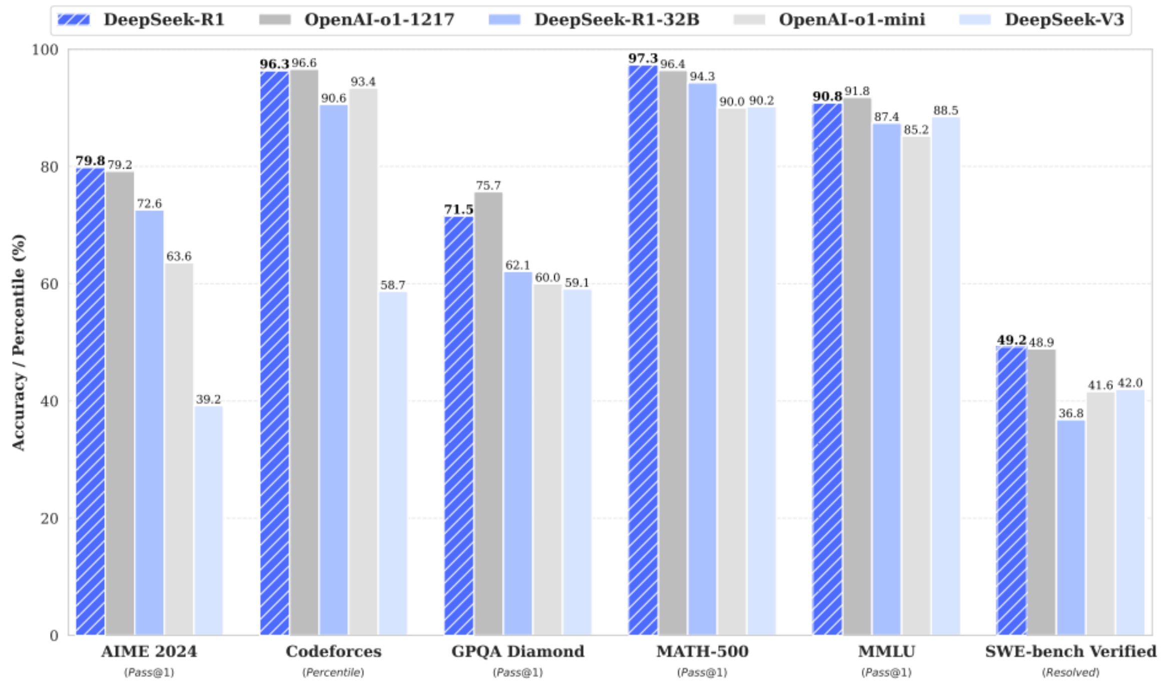
# Serving LLM

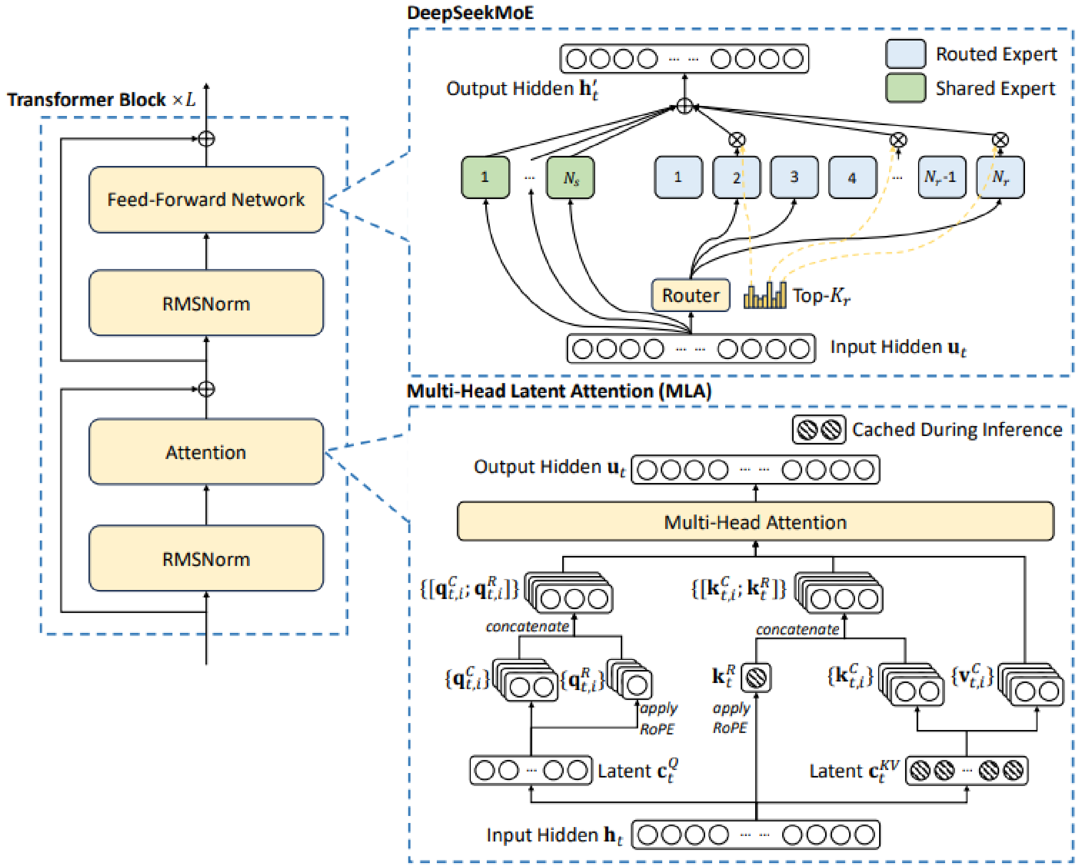
- **TPOT –** Time per Output Token
- **TTFT –** Time to First Token
- **TPUT** – Throughput per NPU



# DeepSeek V3/R1

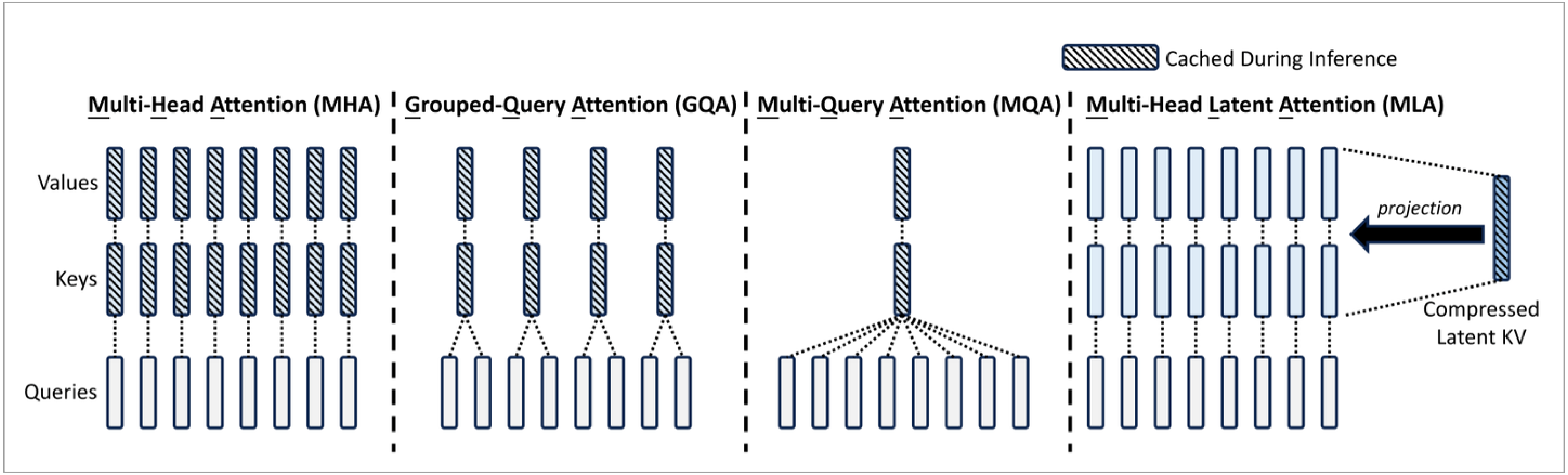
- A game-changer open source (none-)reasoning model that is on par with SOTA proprietary LLMs
- Boosting new surge of demands on LLM inferencing and agents
- 671B model but with many innovations for efficiency and high performance
  - MLA, MoE, MTP



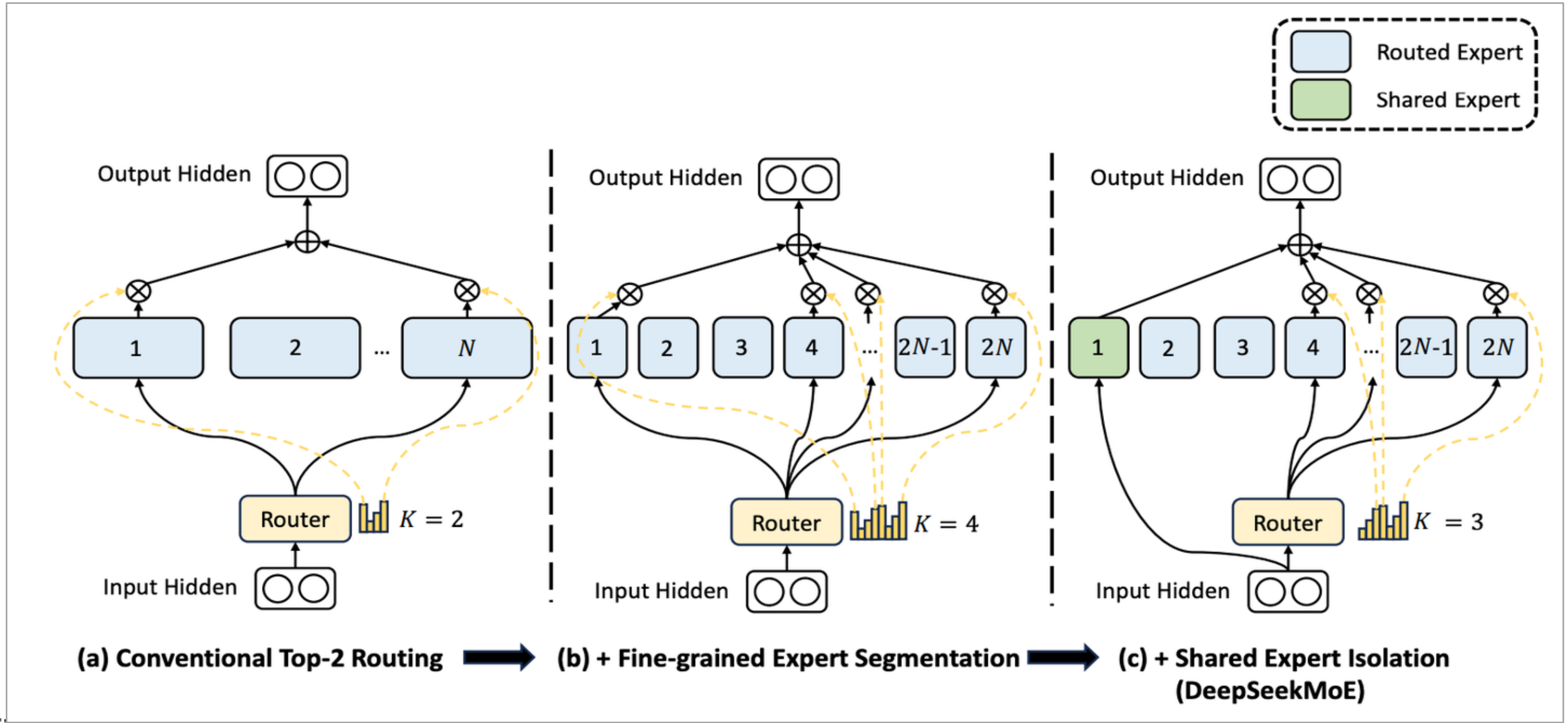
# Multi-head latent attention

- **Compress KV cache into a latent space**
  - Save up to 93.3% KVC space compared to original MHA （70KB per token!)



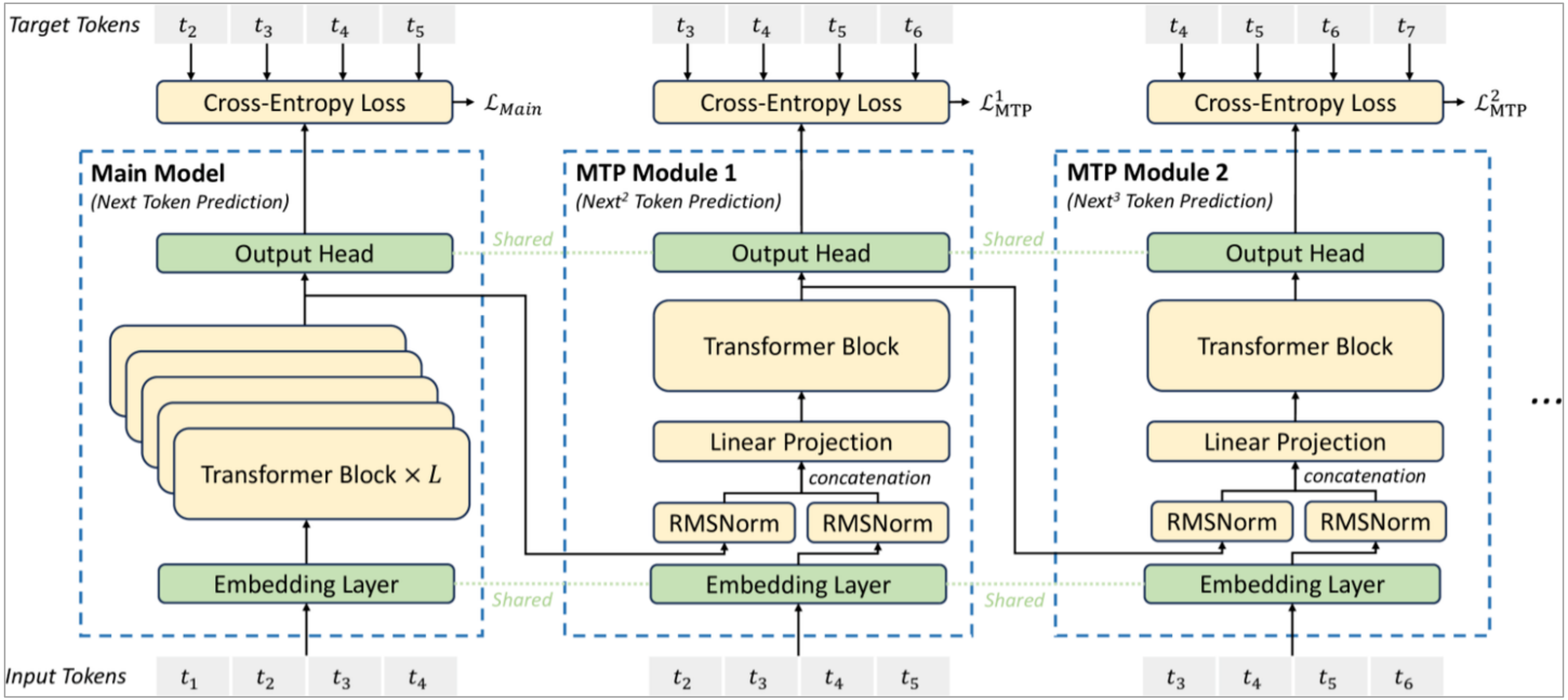
# DeepSeek Mixture-of-Experts

- Fine-grained experts – 256 experts (activation 37B)
- Shared experts vs. routed experts
  - 1+256 experts



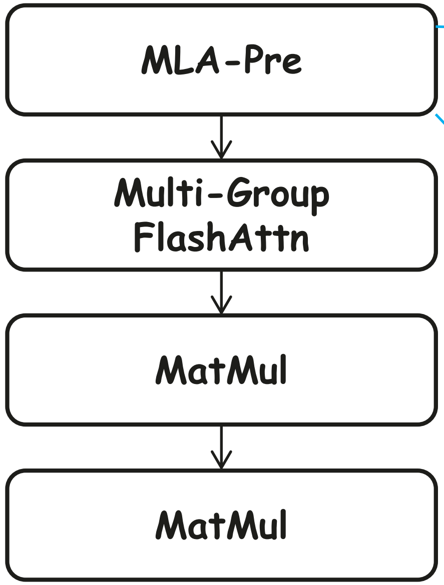
# Multiple Token Prediction

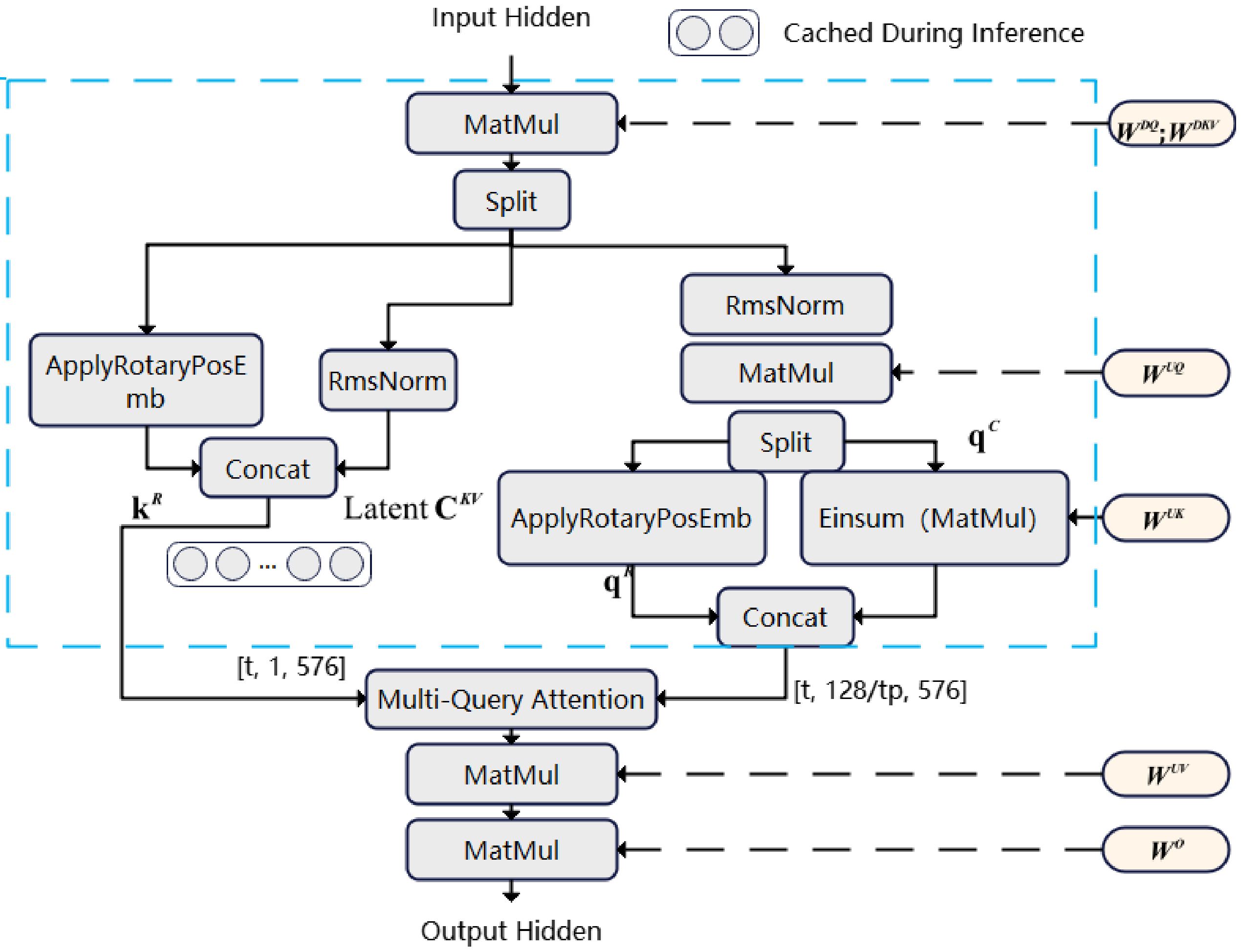
- Involving multiple token prediction at training stage
- Increase stability at when training, and also increasing throughput as speculative decoding



# Optimizing DeepSeek infer. on Ascend – MLA Kernel

- **Our strategy: leverage existing FlashAttention implementation**

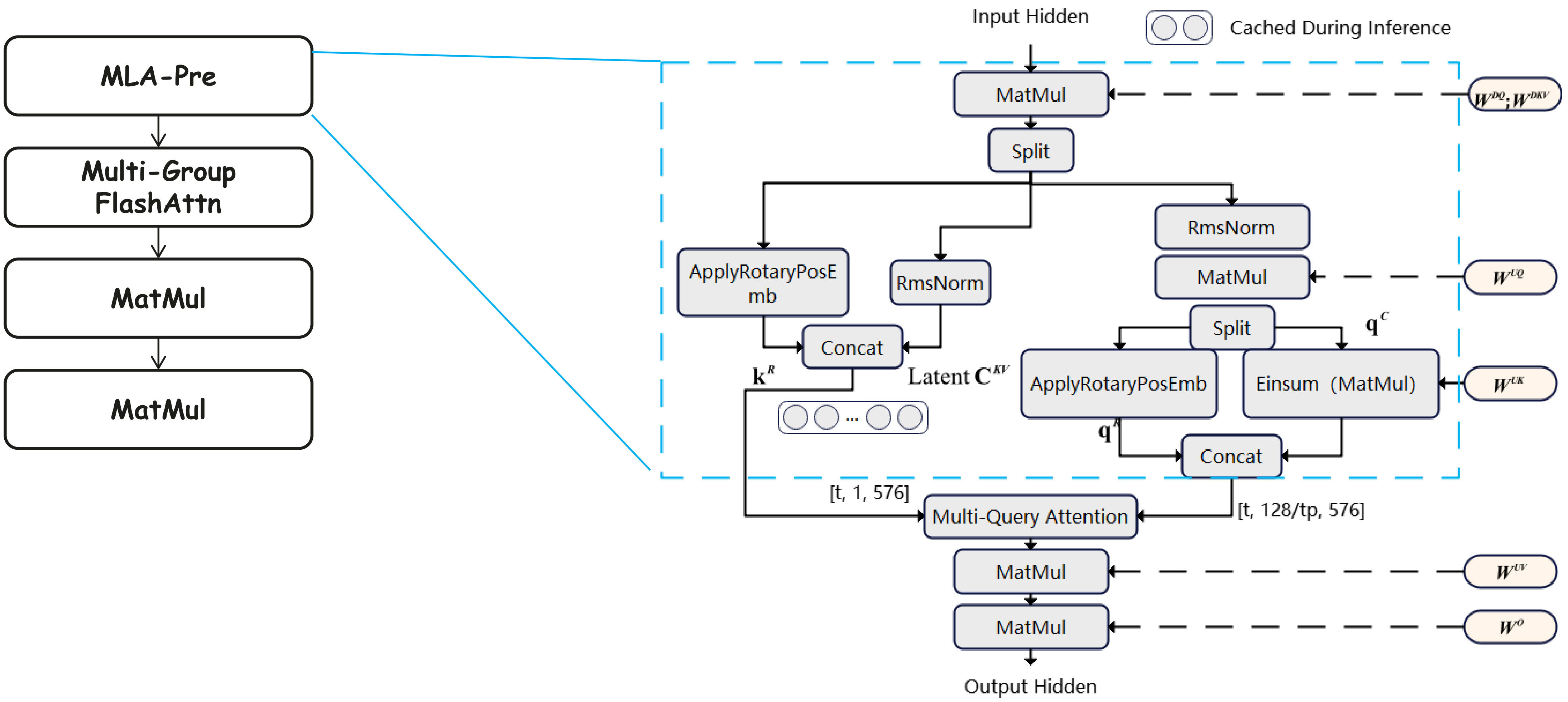



| (batch, seq) | MLA-Pre | w/o. fusion |
|---|---|---|
| (16,1K) | 65.133 | 130.846 |
| (16,2K) | 61.852 | 122.766 |
| (32,2K) | 72.134 | 163.113 |

# Optimizing DeepSeek infer. on Ascend – MLA Kernel

- Our strategy: leverage existing FlashAttention implementation

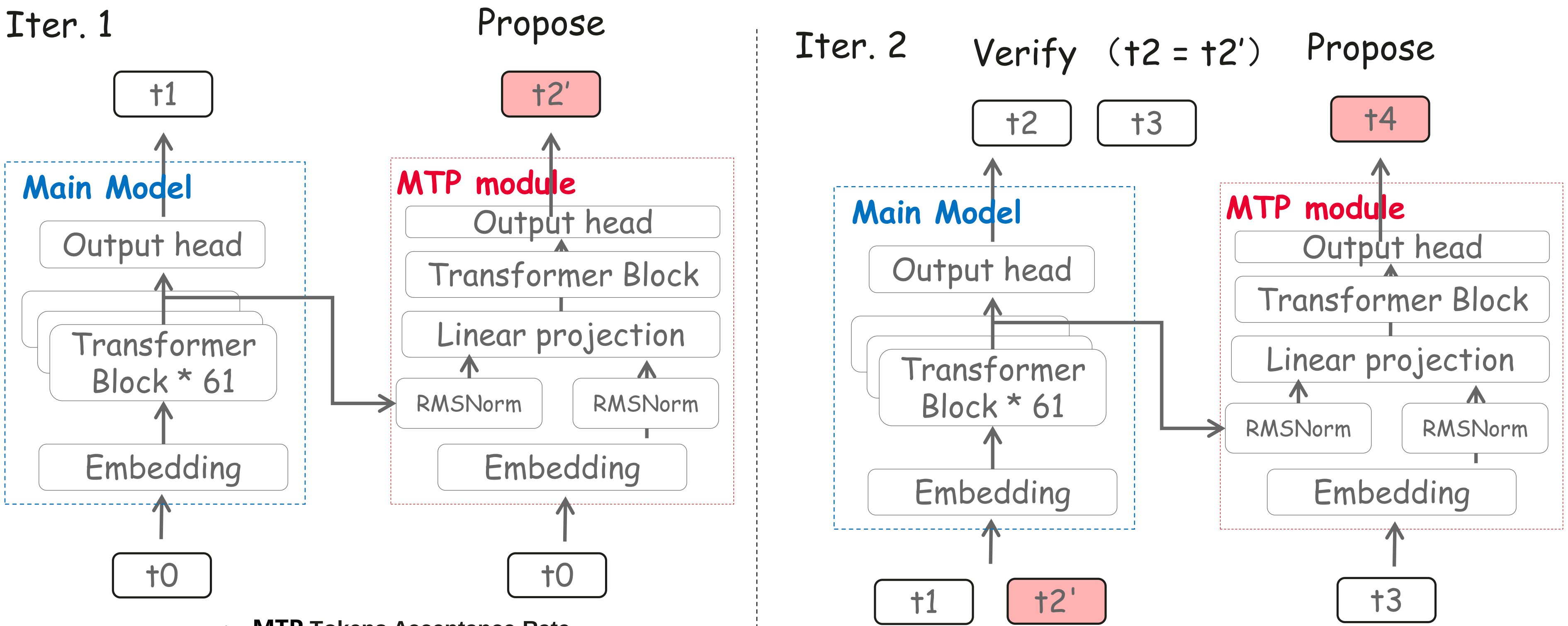



分布式与并行软件实验室
Distributed and Parallel Software Laboratory

HUAWEI

# Optimization 2 – MTP decoding

- **MTP-based speculative decoding is helpful**
  - Improving decoding throughput by 1.8x




| Dataset | MTP Accuracy |
|---|---|
| C-Eval | 81.4% |
| GSM8K | 88.7% |
| MMLU | 90.7% |

- **MTP Tokens Acceptance Rate**

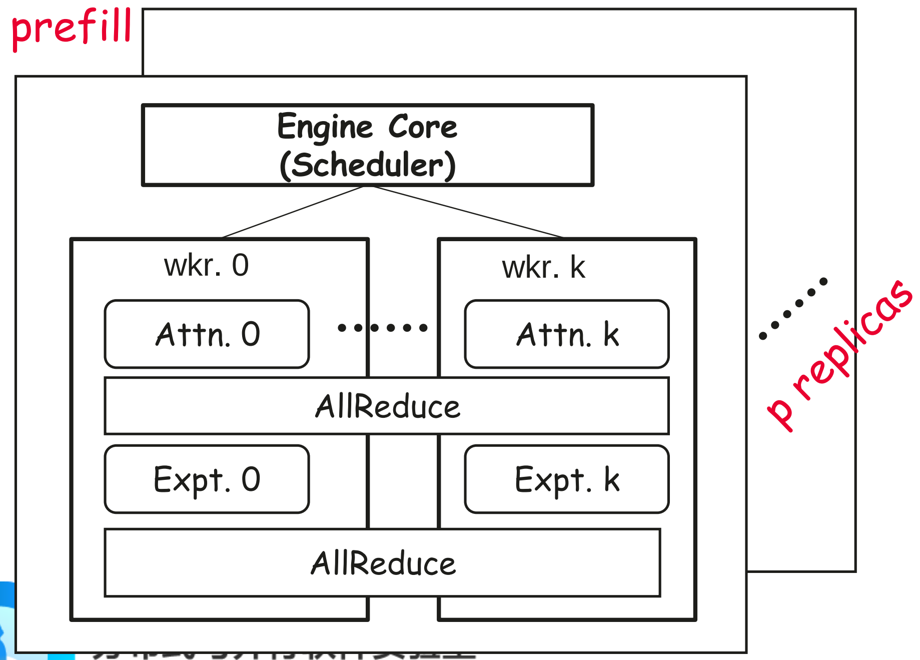
分布式与并行软件实验室
Distributed and Parallel Software Laboratory

HUAWEI

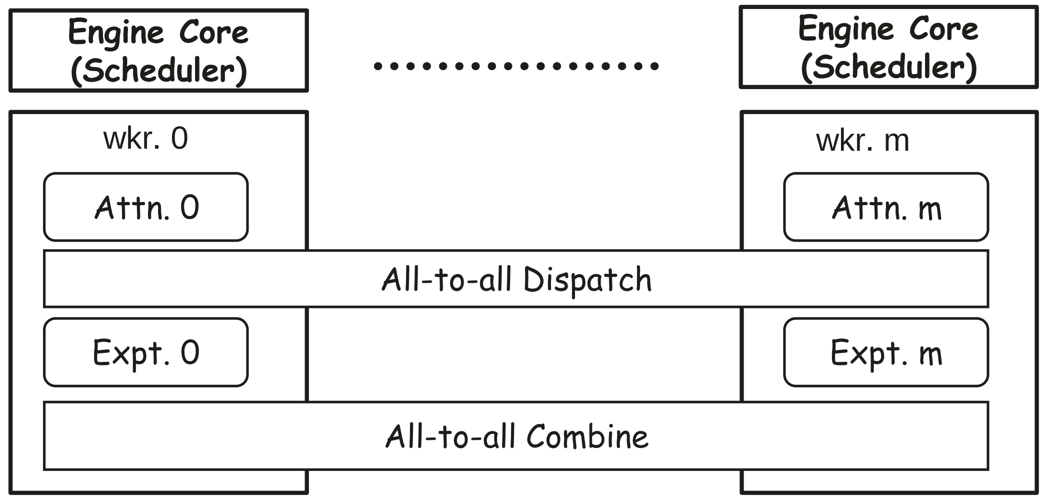
# Optimization 3 – Prefill-decode disaggregation

- **Latency orientated deployment – minimizing the end-to-end delay**
  - TTFT < 1s; TPOT < 50ms ~ 20 tps throughput per user
  - Batching as more requests as possible to increase NPU utilization
- **Adopt different strategies to maximize parallelism at prefill- and decode-stage**
  - Prefill – Tensor-Parallel + Data-Parallel
  - Decode – Expert-Parallel + Fine-grained Data-Parallel

| K*p | M |
|---|---|
| 16*2 | 32 |
| 16*4 | 64 |
| 16*16 | 288 |

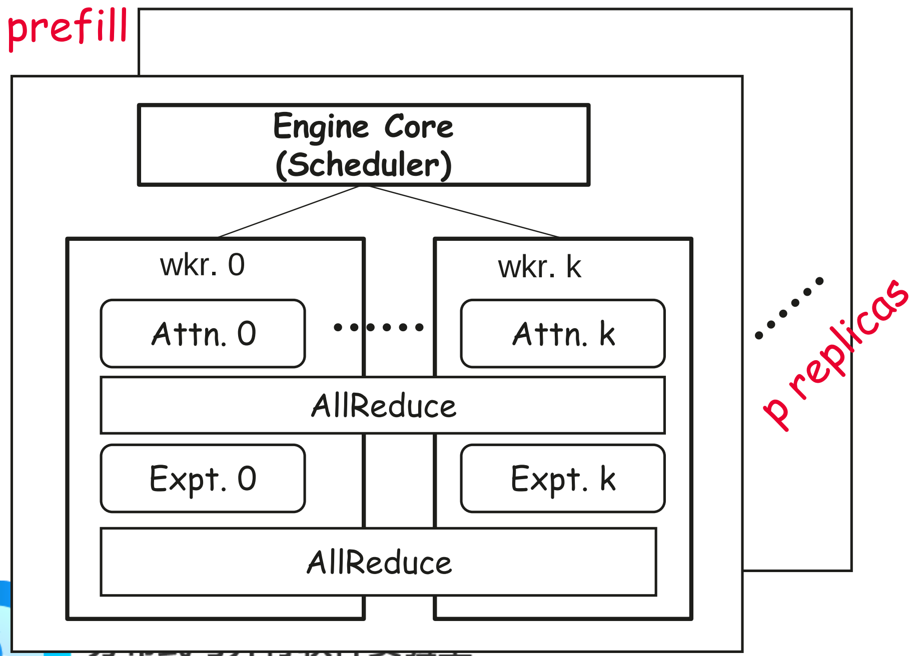



decode

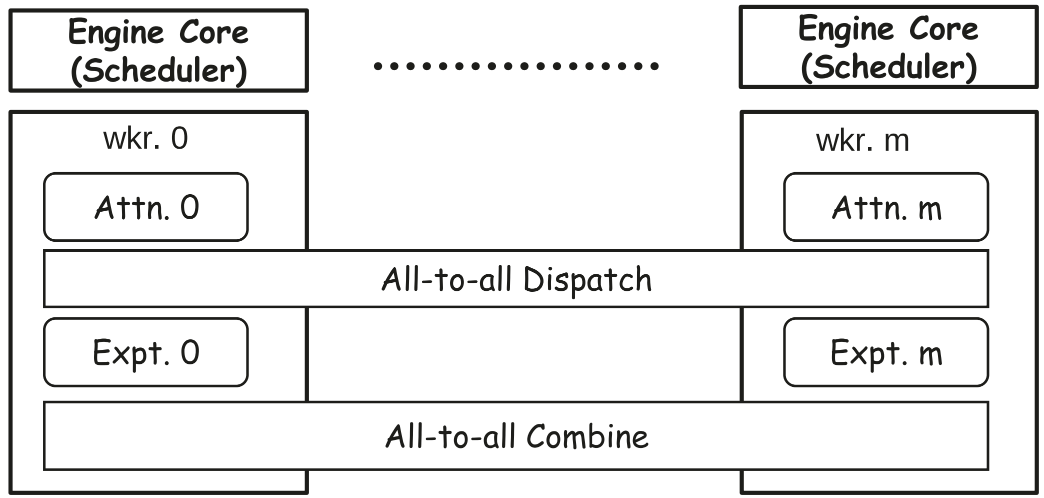


# Optimization 3 – Prefill-decode disaggregation

| K*p | M |
|---|---|
| 16*2 | 32 |
| 16*4 | 64 |
| 16*16 | 288 |

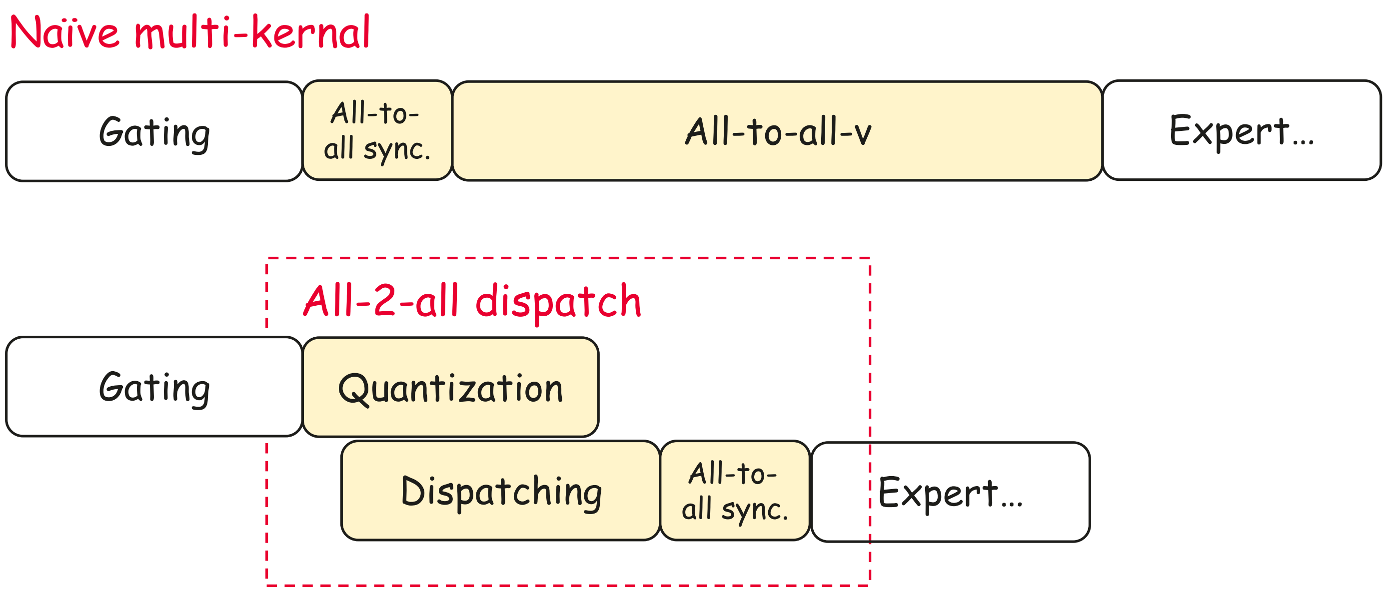


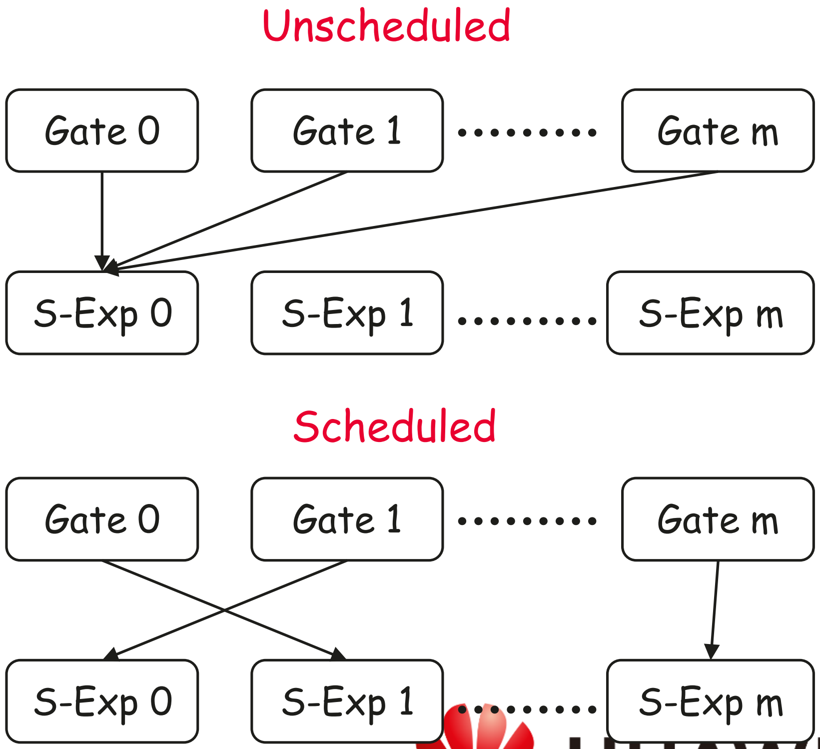

decode



# Optimization 4 – Dispatch/Combine for MoE EP

- **Kernels fusing two all-to-all collectives for EP**
  - Pre-allocated communication buffer and dispatching/combining
  - Finer compute-communication overlapping with MEM WRITE
  - Transmission scheduling for shared experts – avoiding in-casting
  - Communication-time quantization



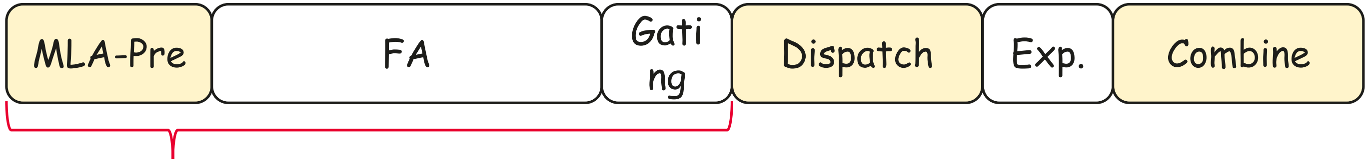



# Put them all together

Performance
w. seqlen = 2K

| Cluster size | Batch | TPOT (us) | TPUT (tpsc) |
|---|---|---|---|
| 64 | 1 | 34 | 29.4 |
| 128 | 32 | 50 | 640 |
| 288 | 96 | 55 | 1745 |

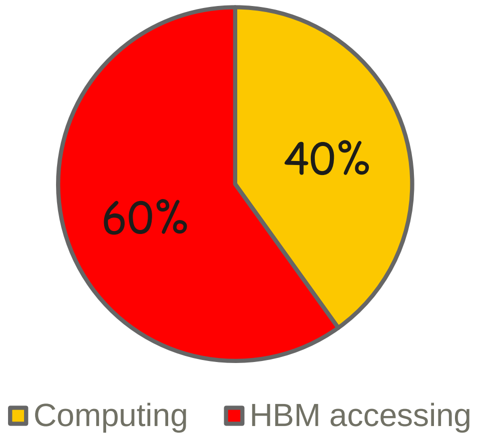
Breakdown




| Attention | Dispatch | Exp. | Combine |
|---|---|---|---|
| 846 | 184 | 234 | 216 |

# Insights and future directions

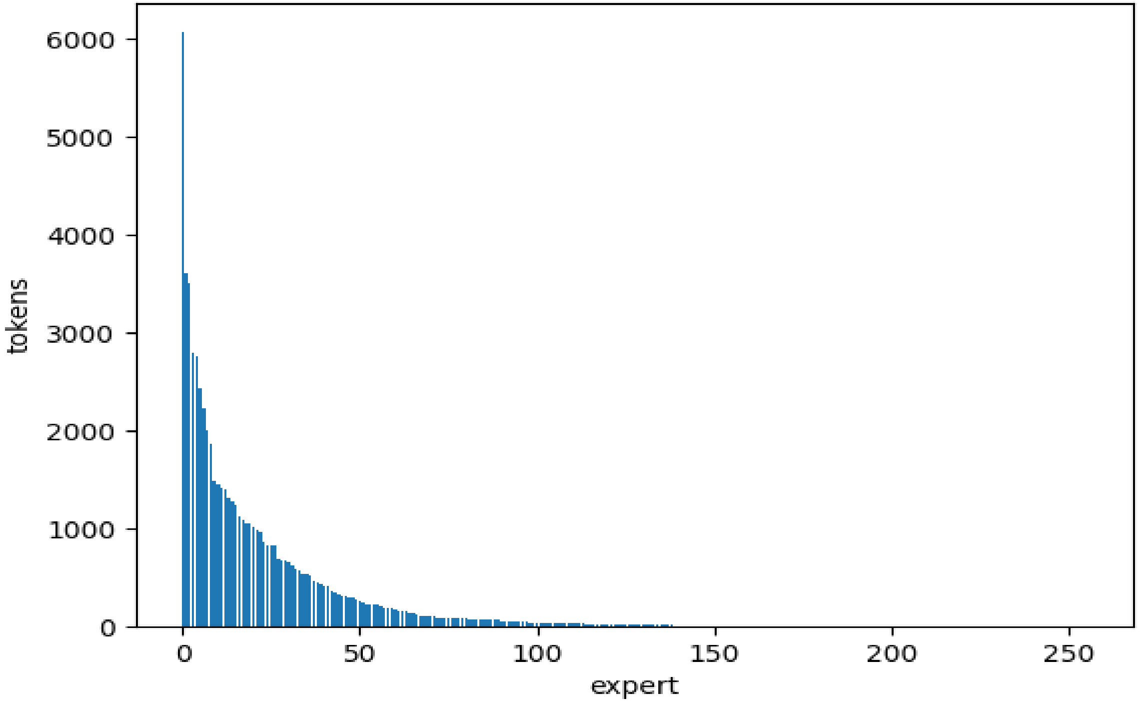
- **Better model and hardware designs close the gap among computing, memory, and communication**
  - e.g., MLA, UB interconnection
  - **Memory bandwidth is still bottleneck**, but the gap is much smaller
  - Tricks (like better quantization) can still be applied to even close the gap
  - Future optimization needs to consider both in memory and compute
    - **Sparsity in time (sequence) dimension is promising**

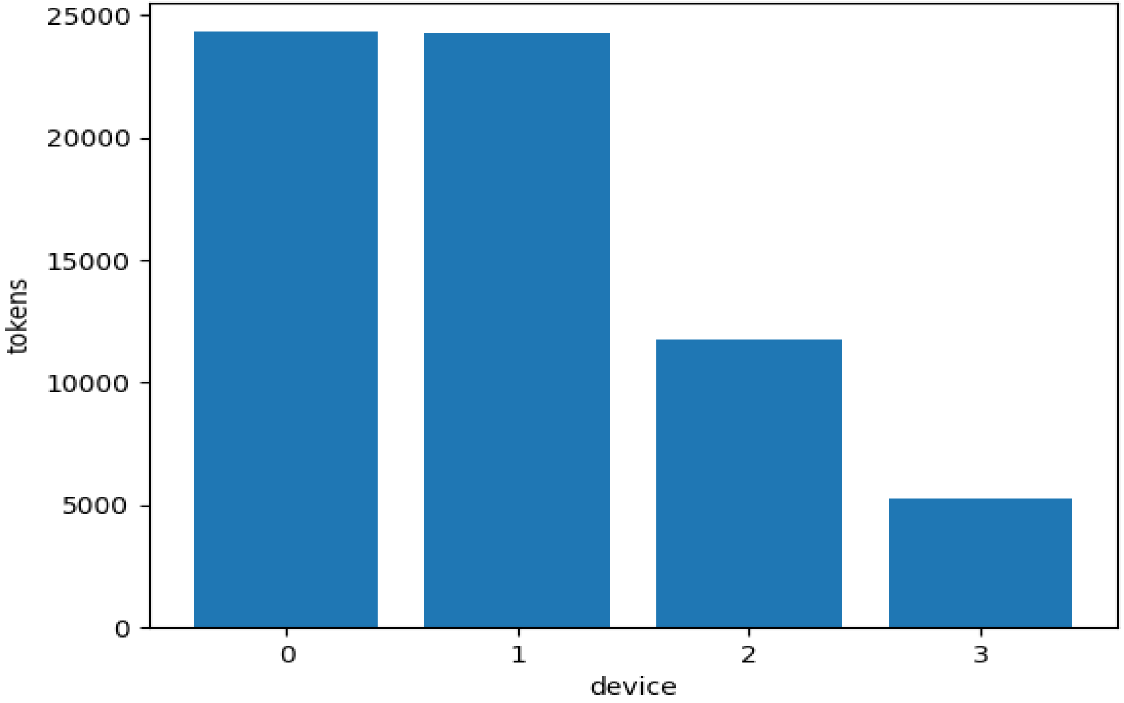



- **Deeper fused kernels**
  - Enable finer pipelining and better computing-communication overlapping
  - **FusedDeepMLA -** MLA-Pre + FlashAttn + Gating
  - **FusedDeepMOE –** Dispatch + Exp. + Combine
  - Expecting 20+% performance boosting
  - **Better frameworks to build fused kernels**

# Insights and future directions (cont.)

- **Benchmark vs. deployment**
  - Canary deployment on production environment
  - Only ~ 50% of benchmarked throughput
  - **Unbalancing among experts calls for dynamic experts scheduling**



MINES PARIS | PSL

# Thank you!

分布式与并行软件实验室
Distributed and Parallel Software Laboratory

HUAWEI