Day1: Flash MLA 深度解读

ZOMI

DeepSeek
@deepseek_ai

🚀 Day 1 of #OpenSourceWeek: FlashMLA

Honored to share FlashMLA - our efficient MLA decoding kernel for Hopper GPUs, optimized for variable-length sequences and now in production.

✅ BF16 support
✅ Paged KV cache (block size 64)
⚡ 3000 GB/s memory-bound & 580 TFLOPS compute-bound on H800

🔗 Explore on GitHub: github.com/deepseek-ai/Fl...

How does FlashMLA manage sequence lengths? | What optimizations targe

7:04 AM · Feb 24, 2025 · **446.3K** Views

DeepSeek 开源 Flash-MLA

- Flash-MLA 适用于 Hopper GPU 高效 MLA 内核，针对可变长度序列服务进行优化。速度在 H800 SXM5 GPU 上具有 3000 GB/s 内存速度 & 580 TFLOPS 计算上限。

- FlashMLA Github 开源地址：https://github.com/deepseek-ai/FlashMLA
- 本 PPT 开源：https://github.com/chenzomi12/AIInfra/tree/main/06AlgoData
- 夸克链接：https://pan.quark.cn/s/374bc7960241（代码、论文、注释）

视频目录大纲

1. DeepSeek V2： MLA 论文解读
2. DeepSeek V2： MLA 原理概况
3. Flash MLA：代码注释与解读
4. 思考与小结

01

DeepSeek V2：MLA 论文解读

02

DeepSeek V2：MLA 原理概况

Vision Guild of MLA

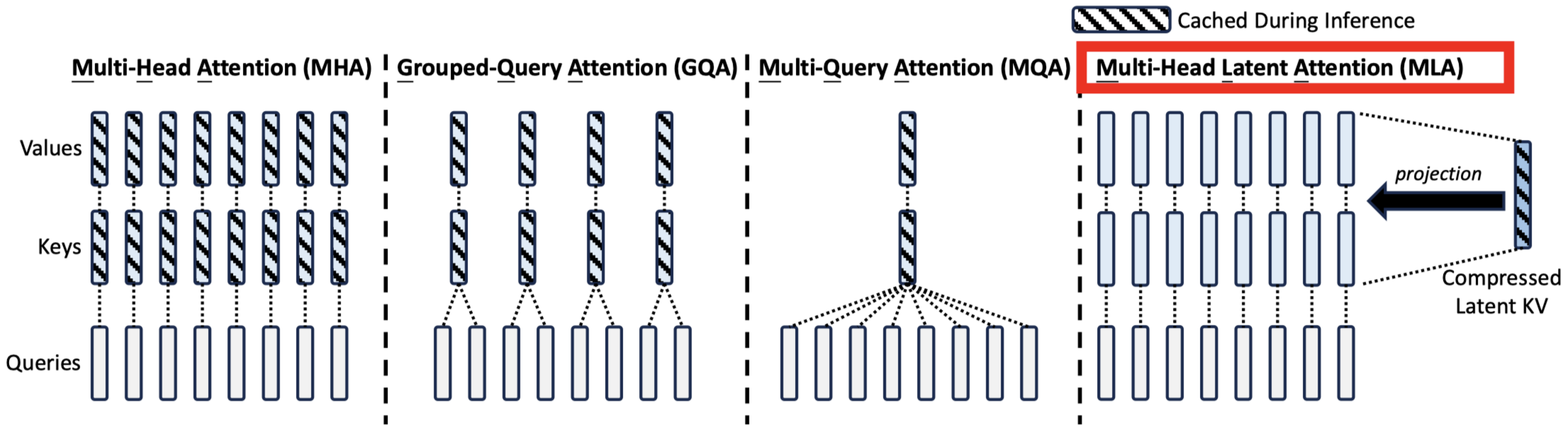
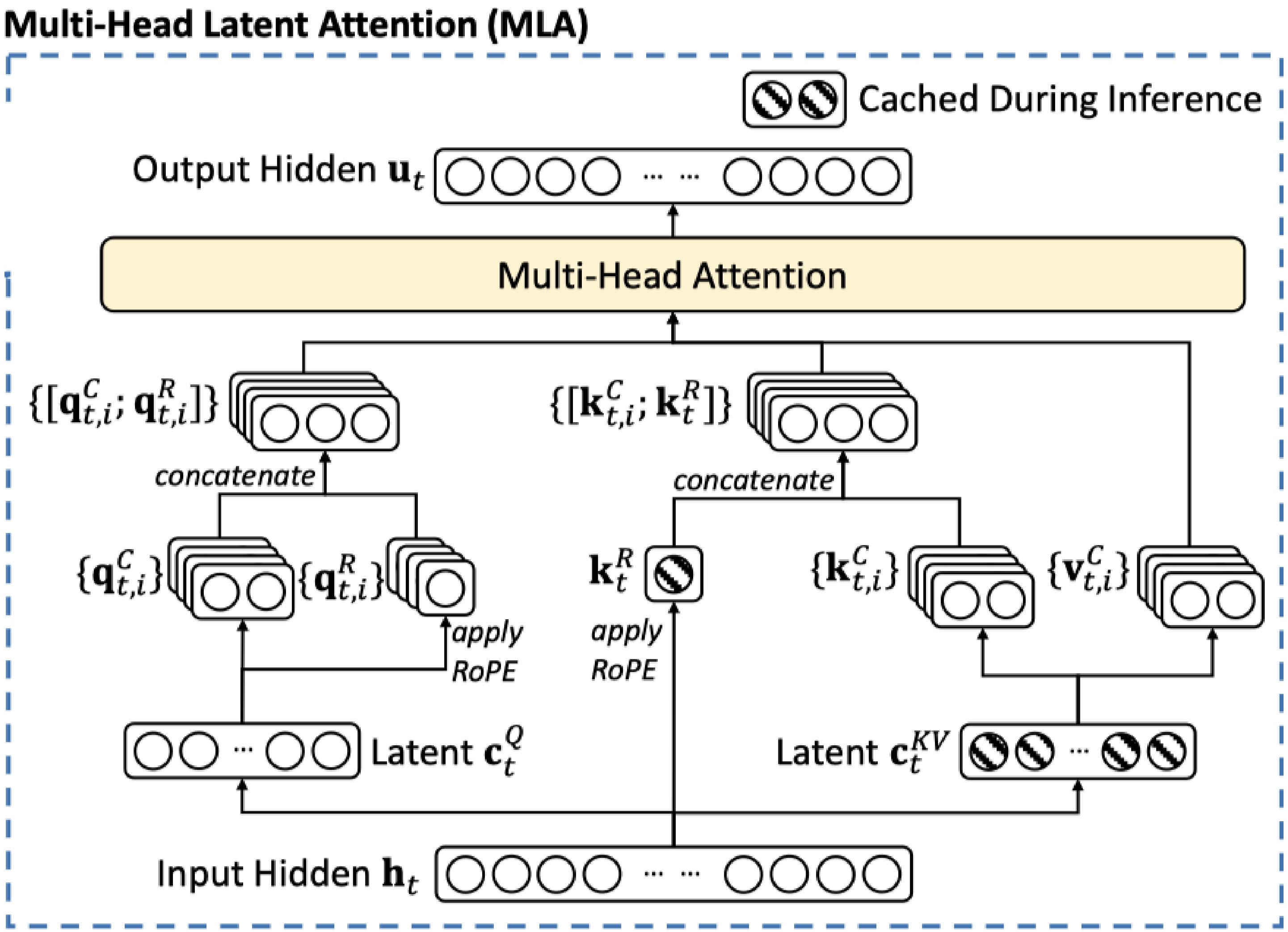
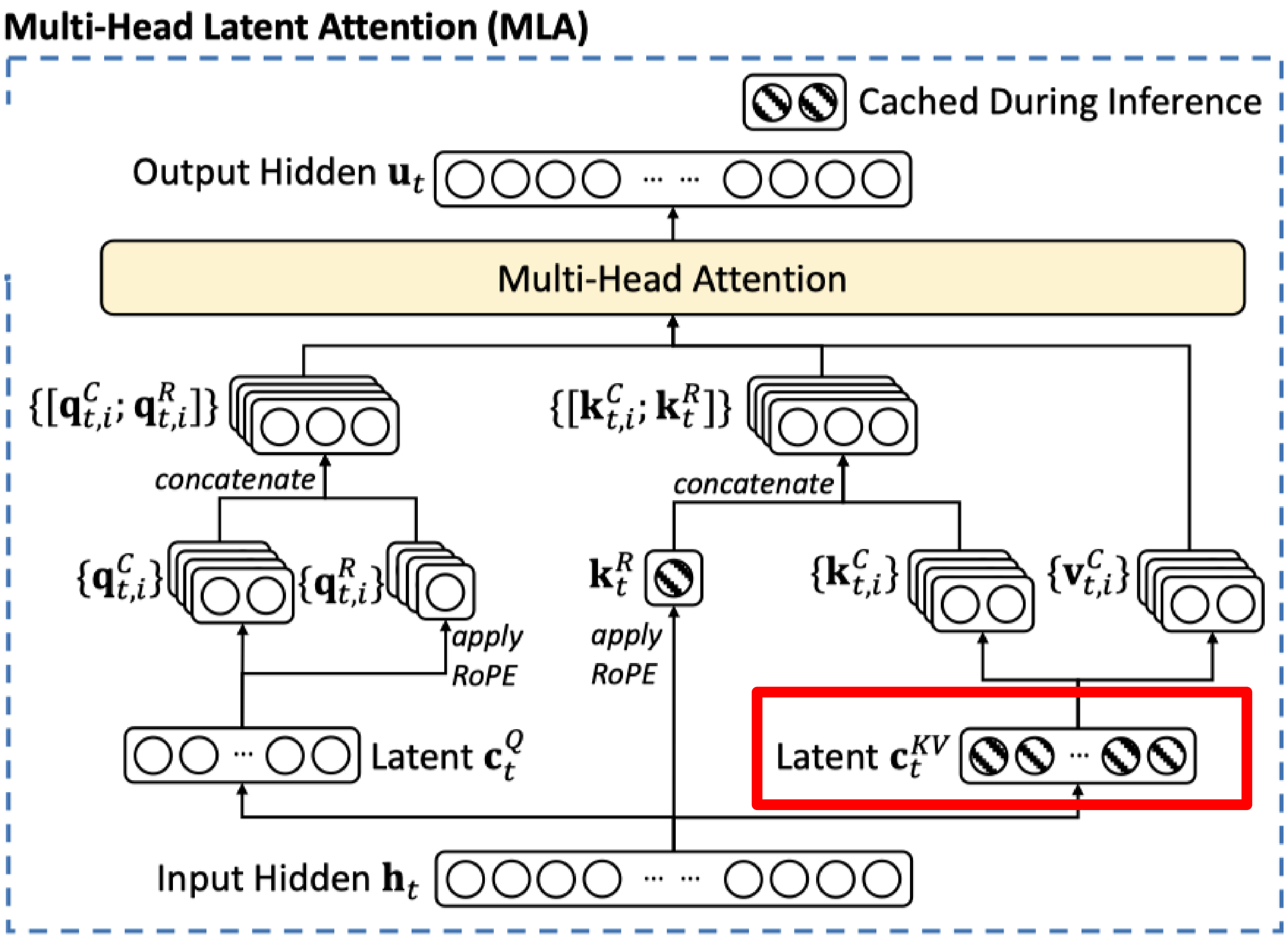


Figure 3 | Simplified illustration of Multi-Head Attention (MHA), Grouped-Query Attention (GQA), Multi-Query Attention (MQA), and Multi-head Latent Attention (MLA). Through jointly compressing the keys and values into a latent vector, MLA significantly reduces the KV cache during inference.

Vision Guild of MLA

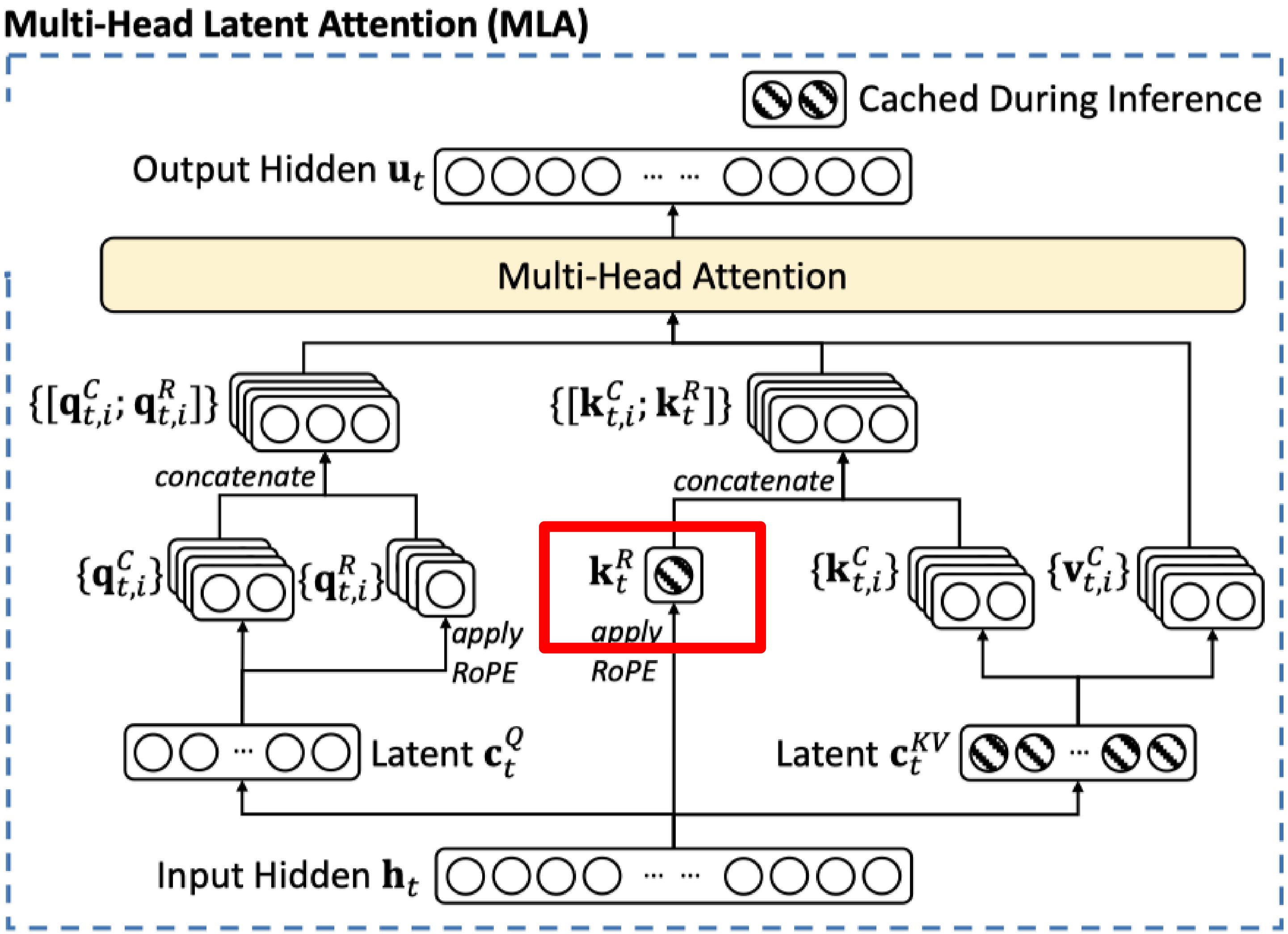


Vision Guild of MLA



- c_t 是输入 h_t 低秩投影向量，长度比 h_t 短。
- 尤为重要的是，c_t^{KV} 是所有 head 共享，因此 MHA 中需要缓存所有 $k_{t,i}(s)$ 和 $v_{t,i}(s)$ 的操作变成了只需要缓存 c_t 。

Vision Guild of MLA



- 为了适配 RoPE，所有 head 共用一个 k_t^R ，并且在设计时让 W_{kr} 的列数 d_r 也比较小。

- MLA 采用了 MQA 的思想，构造了所有 head 共享的 cache 变量 c_t 和 k_t^R ，这样才大幅降低了KV Cache。

Full Formulas of MLA

$$\mathbf{c}_t^Q = W^{DQ}\mathbf{h}_t, \tag{37}$$

$$[\mathbf{q}_{t,1}^C; \mathbf{q}_{t,2}^C; ...; \mathbf{q}_{t,n_h}^C] = \mathbf{q}_t^C = W^{UQ}\mathbf{c}_t^Q, \tag{38}$$

$$[\mathbf{q}_{t,1}^R; \mathbf{q}_{t,2}^R; ...; \mathbf{q}_{t,n_h}^R] = \mathbf{q}_t^R = \mathrm{RoPE}(W^{QR}\mathbf{c}_t^Q), \tag{39}$$

$$\mathbf{q}_{t,i} = [\mathbf{q}_{t,i}^C; \mathbf{q}_{t,i}^R], \tag{40}$$

$$\boxed{\mathbf{c}_t^{KV}} = W^{DKV}\mathbf{h}_t, \tag{41}$$

$$[\mathbf{k}_{t,1}^C; \mathbf{k}_{t,2}^C; ...; \mathbf{k}_{t,n_h}^C] = \mathbf{k}_t^C = W^{UK}\mathbf{c}_t^{KV}, \tag{42}$$

$$\boxed{\mathbf{k}_t^R} = \mathrm{RoPE}(W^{KR}\mathbf{h}_t), \tag{43}$$

$$\mathbf{k}_{t,i} = [\mathbf{k}_{t,i}^C; \mathbf{k}_t^R], \tag{44}$$

$$[\mathbf{v}_{t,1}^C; \mathbf{v}_{t,2}^C; ...; \mathbf{v}_{t,n_h}^C] = \mathbf{v}_t^C = W^{UV}\mathbf{c}_t^{KV}, \tag{45}$$

$$\mathbf{o}_{t,i} = \sum_{j=1}^{t} \mathrm{Softmax}_j\left(\frac{\mathbf{q}_{t,i}^T \mathbf{k}_{j,i}}{\sqrt{d_h + d_h^R}}\right)\mathbf{v}_{j,i}^C, \tag{46}$$

$$\mathbf{u}_t = W^O[\mathbf{o}_{t,1}; \mathbf{o}_{t,2}; ...; \mathbf{o}_{t,n_h}], \tag{47}$$

Full Formulas of MLA

- 每个Transformer层，只缓存蓝框向量： c_t^{KV} 和 k_t^R ：

$$
\begin{aligned}
\mathbf{c}_t^Q &= W^{DQ}\mathbf{h}_t, & (37)\\
[\mathbf{q}_{t,1}^C;\mathbf{q}_{t,2}^C;\ldots;\mathbf{q}_{t,n_h}^C] = \mathbf{q}_t^C &= W^{UQ}\mathbf{c}_t^Q, & (38)\\
[\mathbf{q}_{t,1}^R;\mathbf{q}_{t,2}^R;\ldots;\mathbf{q}_{t,n_h}^R] = \mathbf{q}_t^R &= \mathrm{RoPE}(W^{QR}\mathbf{c}_t^Q), & (39)\\
\mathbf{q}_{t,i} &= [\mathbf{q}_{t,i}^C;\mathbf{q}_{t,i}^R], & (40)\\
\boxed{\mathbf{c}_t^{KV}} &= W^{DKV}\mathbf{h}_t, & (41)\\
[\mathbf{k}_{t,1}^C;\mathbf{k}_{t,2}^C;\ldots;\mathbf{k}_{t,n_h}^C] = \mathbf{k}_t^C &= W^{UK}\mathbf{c}_t^{KV}, & (42)\\
\boxed{\mathbf{k}_t^R} &= \mathrm{RoPE}(W^{KR}\mathbf{h}_t), & (43)\\
\mathbf{k}_{t,i} &= [\mathbf{k}_{t,i}^C;\mathbf{k}_t^R], & (44)\\
[\mathbf{v}_{t,1}^C;\mathbf{v}_{t,2}^C;\ldots;\mathbf{v}_{t,n_h}^C] = \mathbf{v}_t^C &= W^{UV}\mathbf{c}_t^{KV}, & (45)\\
\mathbf{o}_{t,i} &= \sum_{j=1}^{t}\mathrm{Softmax}_j\left(\frac{\mathbf{q}_{t,i}^T\mathbf{k}_{j,i}}{\sqrt{d_h+d_h^R}}\right)\mathbf{v}_{j,i}^C, & (46)\\
\mathbf{u}_t &= W^O[\mathbf{o}_{t,1};\mathbf{o}_{t,2};\ldots;\mathbf{o}_{t,n_h}], & (47)
\end{aligned}
$$

Full Formulas of MLA

- 每个Transformer层，只缓存蓝框向量： c_t^{KV} 和 k_t^R ：

$$\mathbf{c}_t^Q = W^{DQ}\mathbf{h}_t, \tag{37}$$

$$[\mathbf{q}_{t,1}^C; \mathbf{q}_{t,2}^C; \ldots; \mathbf{q}_{t,n_h}^C] = \mathbf{q}_t^C = W^{UQ}\mathbf{c}_t^Q,$$

$$[\mathbf{q}_{t,1}^R; \mathbf{q}_{t,2}^R; \ldots; \mathbf{q}_{t,n_h}^R] = \mathbf{q}_t^R = \mathrm{RoPE}(W^{QR}\mathbf{c}_t^Q),$$

$$\mathbf{q}_{t,i} = [\mathbf{q}_{t,i}^C; \mathbf{q}_{t,i}^R], \tag{40}$$

$$\boxed{\mathbf{c}_t^{KV}} = W^{DKV}\mathbf{h}_t,$$

$$[\mathbf{k}_{t,1}^C; \mathbf{k}_{t,2}^C; \ldots; \mathbf{k}_{t,n_h}^C] = \mathbf{k}_t^C = W^{UK}\mathbf{c}_t^{KV}, \tag{42}$$

$$\boxed{\mathbf{k}_t^R} = \mathrm{RoPE}(W^{KR}\mathbf{h}_t),$$

$$\mathbf{k}_{t,i} = [\mathbf{k}_{t,i}^C; \mathbf{k}_t^R], \tag{44}$$

$$[\mathbf{v}_{t,1}^C; \mathbf{v}_{t,2}^C; \ldots; \mathbf{v}_{t,n_h}^C] = \mathbf{v}_t^C = W^{UV}\mathbf{c}_t^{KV}, \tag{45}$$

$$\mathbf{o}_{t,i} = \sum_{j=1}^{t} \mathrm{Softmax}_j\left(\frac{\mathbf{q}_{t,i}^T \mathbf{k}_{j,i}}{\sqrt{d_h + d_h^R}}\right)\mathbf{v}_{j,i}^C, \tag{46}$$

$$\mathbf{u}_t = W^O[\mathbf{o}_{t,1}; \mathbf{o}_{t,2}; \ldots; \mathbf{o}_{t,n_h}], \tag{47}$$

d_h： 单个 head 向量维度

d_c ： MLA 低秩压缩的维度

c_t^{KV} ：维度为 $d_c = 4 \times d_h = 512$

k_t^R ： 维度为 $d_h^R = d_h/2 = 64$

Full Formulas of MLA

- 每个Transformer层，只缓存

- 公式（37），（38）类似KV的逻辑，通过两个矩阵（$W^{DQ}, W^{UQ} \in \mathbb{R}^{d_h n_h \times d_q}$）也做了一层低秩变换，这一步Q的变换看着趋是为了减少模型的参数的数量。在Deepseek-V3里 $d_q = 1536$ 。是KV压缩维度 d_c 的3倍。但相对于 $d = 7168$ 还是压缩了不少。

$$\mathbf{c}_t^Q = W^{DQ}\mathbf{h}_t, \tag{37}$$

$$[\mathbf{q}_{t,1}^C; \mathbf{q}_{t,2}^C; ...; \mathbf{q}_{t,n_h}^C] = \mathbf{q}_t^C = W^{UQ}\mathbf{c}_t^Q, \tag{38}$$

$$[\mathbf{q}_{t,1}^R; \mathbf{q}_{t,2}^R; ...; \mathbf{q}_{t,n_h}^R] = \mathbf{q}_t^R = \mathrm{RoPE}(W^{QR}\mathbf{c}_t^Q), \tag{39}$$

$$\mathbf{q}_{t,i} = [\mathbf{q}_{t,i}^C; \mathbf{q}_{t,i}^R], \tag{40}$$

$$\boxed{\mathbf{c}_t^{KV}} = W^{DKV}\mathbf{h}_t, \tag{41}$$

$$[\mathbf{k}_{t,1}^C; \mathbf{k}_{t,2}^C; ...; \mathbf{k}_{t,n_h}^C] = \mathbf{k}_t^C = W^{UK}\mathbf{c}_t^{KV}, \tag{42}$$

$$\boxed{\mathbf{k}_t^R} = \mathrm{RoPE}(W^{KR}\mathbf{h}_t), \tag{43}$$

$$\mathbf{k}_{t,i} = [\mathbf{k}_{t,i}^C; \mathbf{k}_t^R], \tag{44}$$

$$[\mathbf{v}_{t,1}^C; \mathbf{v}_{t,2}^C; ...; \mathbf{v}_{t,n_h}^C] = \mathbf{v}_t^C = W^{UV}\mathbf{c}_t^{KV}, \tag{45}$$

$$\mathbf{o}_{t,i} = \sum_{j=1}^{t} \mathrm{Softmax}_j\left(\frac{\mathbf{q}_{t,i}^T\mathbf{k}_{j,i}}{\sqrt{d_h + d_h^R}}\right)\mathbf{v}_{j,i}^C, \tag{46}$$

$$\mathbf{u}_t = W^O[\mathbf{o}_{t,1}; \mathbf{o}_{t,2}; ...; \mathbf{o}_{t,n_h}], \tag{47}$$

Full Formulas of MLA

- 每个Transformer层，只缓存蓝框向量： c_t^{KV} 和 k_t^R ：

$$\mathbf{c}_t^Q = W^{DQ}\mathbf{h}_t, \tag{37}$$

$$[\mathbf{q}_{t,1}^C; \mathbf{q}_{t,2}^C; ...; \mathbf{q}_{t,n_h}^C] = \mathbf{q}_t^C = W^{UQ}\mathbf{c}_t^Q \tag{38}$$

$$[\mathbf{q}_{t,1}^R; \mathbf{q}_{t,2}^R$$

首先公式（41）对输入 h_t 做一个低秩压缩，将 d维的输入经过 W^{DKV} 变换后压缩成 d_c维的 c_t^{KV} 。DeepSeek-V3中 $d = 7168$， $d_c = 512$

$$\boxed{\mathbf{c}_t^{KV}} = W^{DKV}\mathbf{h}_t, \tag{41}$$

$$[\mathbf{k}_{t,1}^C; \mathbf{k}_{t,2}^C; ...; \mathbf{k}_{t,n_h}^C] = \mathbf{k}_t^C = W^{UK}\mathbf{c}_t^{KV}, \tag{42}$$

$$\boxed{\mathbf{k}_t^R} = \mathrm{RoPE}(W^{KR}\mathbf{h}_t), \tag{43}$$

$$\mathbf{k}_{t,i} = [\mathbf{k}_{t,i}^C; \mathbf{k}_t^R], \tag{44}$$

$$[\mathbf{v}_{t,1}^C; \mathbf{v}_{t,2}^C; ...; \mathbf{v}_{t,n_h}^C] = \mathbf{v}_t^C = W^{UV}\mathbf{c}_t^{KV}, \tag{45}$$

$$\mathbf{o}_{t,i} = \sum_{j=1}^{t} \mathrm{Softmax}_j\left(\frac{\mathbf{q}_{t,i}^T \mathbf{k}_{j,i}}{\sqrt{d_h + d_h^R}}\right)\mathbf{v}_{j,i}^C, \tag{46}$$

$$\mathbf{u}_t = W^O[\mathbf{o}_{t,1}; \mathbf{o}_{t,2}; ...; \mathbf{o}_{t,n_h}], \tag{47}$$

Full Formulas of MLA

- 每个Transformer层，只缓存蓝框向量： c_t^{KV} 和 k_t^R ：

$$\mathbf{c}_t^Q = W^{DQ}\mathbf{h}_t, \tag{37}$$

$$[\mathbf{q}_{t,1}^C; \mathbf{q}_{t,2}^C; ...; \mathbf{q}_{t,n_h}^C] = \mathbf{q}_t^C = W^{UQ}\mathbf{c}_t^Q, \tag{38}$$

$$[\mathbf{q}_{t,1}^R; \mathbf{q}_{t,2}^R; ...; \mathbf{q}_{t,n_h}^R] = \mathbf{q}_t^R = \mathrm{RoPE}(W^{QR}\mathbf{c}_t^Q), \tag{39}$$

- 然后通过公式（42）和公式（45）两个变换矩阵（ $W^{UK}, W^{UV} \in \mathbb{R}^{d_h n_h \times d_c}$ ），将KV的维度扩展回$d = d_h n_h$ ，也就是每个Head有一个单独的 k, v （跟MHA的KV数量一致）

$$[\mathbf{k}_{t,1}^C; \mathbf{k}_{t,2}^C; ...; \mathbf{k}_{t,n_h}^C] = \mathbf{k}_t^C = W^{UK}\mathbf{c}_t^{KV}, \tag{42}$$

$$\boxed{\mathbf{k}_t^R} = \mathrm{RoPE}(W^{KR}\mathbf{h}_t), \tag{43}$$

$$\mathbf{k}_{t,i} = [\mathbf{k}_{t,i}^C; \mathbf{k}_t^R], \tag{44}$$

$$[\mathbf{v}_{t,1}^C; \mathbf{v}_{t,2}^C; ...; \mathbf{v}_{t,n_h}^C] = \mathbf{v}_t^C = W^{UV}\mathbf{c}_t^{KV}, \tag{45}$$

$$\mathbf{o}_{t,i} = \sum_{j=1}^{t} \mathrm{Softmax}_j\left(\frac{\mathbf{q}_{t,i}^T \mathbf{k}_{j,i}}{\sqrt{d_h + d_h^R}}\right)\mathbf{v}_{j,i}^C, \tag{46}$$

$$\mathbf{u}_t = W^O[\mathbf{o}_{t,1}; \mathbf{o}_{t,2}; ...; \mathbf{o}_{t,n_h}], \tag{47}$$

Full Formulas of MLA

- 每个Transformer层，只缓存蓝框向量： c_t^{KV} 和 k_t^R ：

- 我们注意到在增加RoPE⁺位置编码并没有在上述计算出的 q_t^C, k_t^C 的基础上乘以Rope的对角矩阵。而是单独计算了两个带着位置编码的 q_t^R, k_t^R 如公式（39）和公式（43）所示

$[\mathbf{q}_{t,1}^C; \mathbf{q}_{t,2}^C$

$$[\mathbf{q}_{t,1}^R; \mathbf{q}_{t,2}^R; \ldots; \mathbf{q}_{t,n_h}^R] = \mathbf{q}_t^R = \mathrm{RoPE}(W^{QR}\mathbf{c}_t^Q), \tag{39}$$

$$\mathbf{q}_{t,i} = [\mathbf{q}_{t,i}^C; \mathbf{q}_{t,i}^R], \tag{40}$$

$$\boxed{\mathbf{c}_t^{KV}} = W^{DKV}\mathbf{h}_t, \tag{41}$$

$$[\mathbf{k}_{t,1}^C; \mathbf{k}_{t,2}^C; \ldots; \mathbf{k}_{t,n_h}^C] = \mathbf{k}_t^C = W^{UK}\mathbf{c}_t^{KV}, \tag{42}$$

$$\boxed{\mathbf{k}_t^R} = \mathrm{RoPE}(W^{KR}\mathbf{h}_t), \tag{43}$$

$$\mathbf{k}_{t,i} = [\mathbf{k}_{t,i}^C; \mathbf{k}_t^R], \tag{44}$$

$$[\mathbf{v}_{t,1}^C; \mathbf{v}_{t,2}^C; \ldots; \mathbf{v}_{t,n_h}^C] = \mathbf{v}_t^C = W^{UV}\mathbf{c}_t^{KV}, \tag{45}$$

$$\mathbf{o}_{t,i} = \sum_{j=1}^{t} \mathrm{Softmax}_j\left(\frac{\mathbf{q}_{t,i}^T \mathbf{k}_{j,i}}{\sqrt{d_h + d_h^R}}\right)\mathbf{v}_{j,i}^C, \tag{46}$$

$$\mathbf{u}_t = W^O[\mathbf{o}_{t,1}; \mathbf{o}_{t,2}; \ldots; \mathbf{o}_{t,n_h}], \tag{47}$$

Full Formulas of MLA

- 每个Transformer层，只缓存蓝框向量： c_t^{KV} 和 k_t^R ：

$$\mathbf{c}_t^Q = W^{DQ}\mathbf{h}_t, \tag{37}$$

$[\mathbf{q}_{t,1}^C; \mathbf{q}_{t,}^C$ …

$[\mathbf{q}_{t,1}^R; \mathbf{q}_{t,}^R$ …

- 然后按如下公式（40），（44）跟已经计算的 q_t^C , k_t^C 拼接，构成完整的 q_t , k_t 向量。

$$\mathbf{q}_{t,i} = [\mathbf{q}_{t,i}^C; \mathbf{q}_{t,i}^R], \tag{40}$$

$$\boxed{\mathbf{c}_t^{KV}} = W^{DKV}\mathbf{h}_t, \tag{41}$$

$$[\mathbf{k}_{t,1}^C; \mathbf{k}_{t,2}^C; ...; \mathbf{k}_{t,n_h}^C] = \mathbf{k}_t^C = W^{UK}\mathbf{c}_t^{KV}, \tag{42}$$

$$\boxed{\mathbf{k}_t^R} = \mathrm{RoPE}(W^{KR}\mathbf{h}_t), \tag{43}$$

$$\mathbf{k}_{t,i} = [\mathbf{k}_{t,i}^C; \mathbf{k}_t^R], \tag{44}$$

$$[\mathbf{v}_{t,1}^C; \mathbf{v}_{t,2}^C; ...; \mathbf{v}_{t,n_h}^C] = \mathbf{v}_t^C = W^{UV}\mathbf{c}_t^{KV}, \tag{45}$$

$$\mathbf{o}_{t,i} = \sum_{j=1}^{t} \mathrm{Softmax}_j\left(\frac{\mathbf{q}_{t,i}^T\mathbf{k}_{j,i}}{\sqrt{d_h + d_h^R}}\right)\mathbf{v}_{j,i}^C, \tag{46}$$

$$\mathbf{u}_t = W^O[\mathbf{o}_{t,1}; \mathbf{o}_{t,2}; ...; \mathbf{o}_{t,n_h}], \tag{47}$$

03

Flash MLA：代码注释与解读

flash_fwd_mla_kernel.h

1. **共享内存布局优化**：
    - 根据输入数据的维度选择合适的共享内存布局，以优化内存访问模式。
2. **矩阵乘法和 Softmax 计算：**
    - 实现高效的矩阵乘法和 Softmax 计算，支持因果掩码（Causal Mask）和分块计算。
3. **Split-K 优化：**
    - 通过 Split-K 将计算任务分配到多个线程块中，提高并行度和计算效率。
4. **结果存储：**
    - 将计算结果存储到全局内存或中间缓存中，支持 Split 和非 Split 的存储策略。

End

总结与思考

优化特性

1. **分页KV缓存管理：**
   - 针对长序列推理中显存碎片严重问题，Flash-MLA 实现基于 64-block Paged KV Cache，极大提高了显存利用率，缓解内存访问瓶颈。
2. **异步内存拷贝：**
   - 利用 NVIDIA Hopper SM90 架构特性，借助 Tensor Memory Accelerator (TMA) 异步内存拷贝指令，实现显存（HBM/GDDR）到 SRAM 零拷贝传输，接近理论峰值带宽。
3. **双模式执行引擎：**
   - 为适应不同输入序列长度场景，Flash-MLA 采用动态负载均衡算法，设计了双缓冲模式，短序列下采用计算优先模式，长序列下采用内存优先模式，使得整体延迟大幅降低。

Question

- Flash MLA 主要提供推理（前向）在 Hopper 架构加速，未来会被支持异构推理框架集成。
- 那么这对推理框架意味着什么呢？对中间加速库/推理框架的公司有哪些启示？
- 对算力的到底是利好还是什么格局呢？

引用与参考

- FlashMLA Github 开源地址：https://github.com/deepseek-ai/FlashMLA
- 本 PPT 开源：https://github.com/chenzomi12/AIInfra/tree/main/06AlgoData
- 夸克链接：https://pan.quark.cn/s/374bc7960241（代码、论文、注释）

Thank you

把AI系统带入每个开发者、每个家庭、每个组织，构建万物互联的智能世界

Bring AI System to every person, home and organization for a fully connected, intelligent world.

ZOMI

GitHub github.com/chenzomi12/AIInfra