# 第一章　性能指标计算

## 第1条　影响性能计算的部署策略

1）**PD分离部署特性：**

Transformer大模型推理PD分离部署特性，主要是指模型推理的prefill阶段和decode阶段分别实例化部署在不同的机器资源上同时进行推理，其结合prefill阶段的计算密集型特性，以及decode阶段的访存密集型特性，通过调节PD节点数量配比来提升decode节点的batch size来充分发挥NPU卡的算力，进而提升集群整体吞吐。此外，在decode平均低时延约束场景，PD分离相比PD混合部署，更加能够发挥性能优势。

2）**TBO（Two-Batch Overlap，双批次重叠）**

是一种通过让两个微批次（Micro-Batch）的计算与通信交替执行来隐藏通信开销的并行技术：

a）执行机制

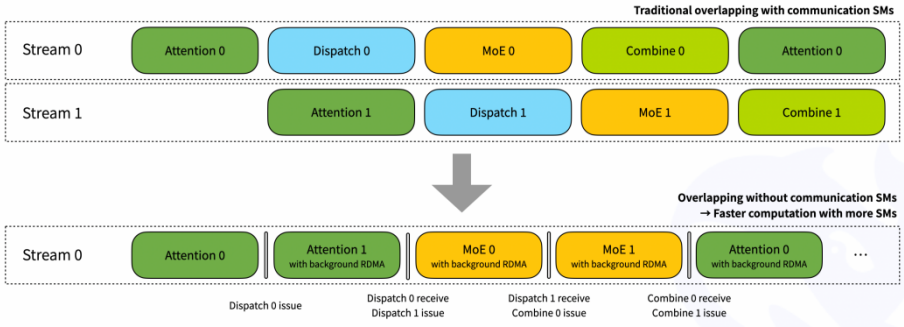


图 6.1 DeepEP 优化后的 TBO

b）执行步骤：

c）**批次分割：**将输入批次分为 Batch A 和 Batch B

d）**交替执行：**

Batch A 通信时，GPU 计算 Batch B

Batch B 通信时，GPU 计算 Batch A

e）**理想效果：**完全隐藏 All-to-All 通信开销，DeepEP可以做到。

f）必要条件：

g）**至少有两个 Batch**（单 Batch 无法使用 TBO）

h）**通信时间 ≈ 计算时间**（通信是显著瓶颈）

i）**Micro-Batch 能满足性能要求**

## 第2条　Embedding 层

1）参数量

$$P_{embed} = V \times H = 129{,}280 \times 7{,}168 \approx 927 \text{ Million}$$

2) 显存占用

$$M_{embed} = P_{embed} \times \text{Precision} = 0.93B \times 1\text{Byte (FP8)} = \mathbf{0.93\ GB}$$

3) 计算量Embedding 层本质是**查表操作**（Lookup），不涉及矩阵乘法，计算量可忽略。

4) 数据搬运量

a) Prefill阶段：$B \times S \times H \times \text{Precision}$

b) Decode阶段：$B \times H \times \text{Precision}$

5) 数据通讯量

a) **TP 场景**：Embedding 按词表维度切分，无需 AllReduce

b) **DP 场景**：无通讯

**第3条 MLA (Multi-Head Latent Attention)层**

1) MLA 参数量

a) **单层参数分解**：

| 投影矩阵 | 维度 | 参数量计算 | 数值 |
|---|---|---|---|
| $W_{q\_a}$(Q 下投影) | $[H, R_q]$ | $H \times R_q$ | 7,168 × 1,536 =**11.0M** |
| $W_{q\_b}$(Q 上投影) | $[R_q, N_h \times$ | $R_q \times N_{heads} \times ($ | 1,536 × 128 × 192 =**37.7M** |
| $W_{kv\_a}$(KV 下投影) | $[H, d_{kv}]$ | $H \times (R_{kv} + d_{ro}$ | 7,168 × 576 =**4.1M** |
| $W_{kv\_b}$(KV 上投影) | $[R_{kv}, N_h$ | $R_{kv} \times N_{heads} \times$ | 512 × 128 × 256 =**16.8M** |
| $W_o$(输出投影) | $[N_h \times d_v$ | $N_{heads} \times d_v \times$ | 128 × 128 × 7,168 =**117.4M** |
| **单层合计** | - | - | **187M** |

b) 所有层参数量（61层）：

$$P_{MLA}^{total} = 61 \times 187M = \mathbf{11.4B}$$

2) 显存占用

a) **静态权重**：

$$M_{MLA}^{static} = 11.4B \times 1\text{Byte} = \mathbf{11.4\ GB}$$

b) **动态 KV Cache**：

| 对比项 | 标准 MHA | MLA (DeepSeek-V3) |
|---|---|---|
| 每 Token 存储维度 | $2 \times N_h \times d_h = 32{,}768$ | $R_{kv} + d_{rope} = 576$ |
| 压缩比 | 100% | **1.76%** |

$$M_{KV}^{MLA} = L \times B \times S \times (R_{kv} + d_{rope}) \times \text{Precision}$$

c) **示例**（4K序列，Batch=1，FP8）：

$$M_{KV} = 61 \times 1 \times 4096 \times 576 \times 1 = 143.7\text{MB}$$

3) 计算量（每层）：

a) 线性投影部分

| 操作 | 矩阵乘法形状 | FLOPs 公式 | 数值（T=1） |
|---|---|---|---|
| Q 下投影 | $[T,H]\times[H,$ | $2\times T\times H\times R_q$ | 2 × 7,168 × 1,536 = 22.0M |
| Q 上投影 | $[T,R_q]\times[R$ | $2\times T\times R_q\times N_h\times$ | 2 × 1,536 × 128 × 192 = 75.5M |
| KV 下投影 | $[T,H]\times[H,$ | $2\times T\times H\times d_{kv}$ | 2 × 7,168 × 576 = 8.3M |
| KV 上投影 | $[T,R_{kv}]\times[$ | $2\times T\times R_{kv}\times N_h\times$ | 2 × 512 × 128 × 256 = 33.6M |
| 输出投影 | $[T,N_h\times d_v]$ | $2\times T\times N_h\times d_v\times$ | 2 × 128 × 128 × 7,168 = 234.9M |
| **线性投影合计** | - | $2\times T\times P_{MLA}$ | **2 × T × 187M = 374M** |

b) Attention 部分其中 $d=d_{nope}+d_{rope}=192$ Prefill（$T=S$，$S_{kv}=S$，二次复杂度）：

$\text{FLOPs}_{Attn}^{prefill}=2\times N_h\times S^2\times(d+d_v)=2\times 128\times S^2\times 320$ Decode（$T=1$，$S_{kv}=$KVLen，线性复杂度）：

$$\text{FLOPs}_{Attn}^{decode}=2\times N_h\times 1\times \text{KVLen}\times(d+d_v)=2\times 128\times \text{KVLen}\times 320$$

| 操作 | 矩阵乘法形状 | FLOPs 公式 |
|---|---|---|
| QK^T | $[T,d]\times[d,S_{kv}]$per head | $2\times N_h\times T\times d\times S_{kv}$ |
| Score × V | $[T,S_{kv}]\times[S_{kv},d_v]$per head | $2\times N_h\times T\times S_{kv}\times d_v$ |

c) 单层总计算量

Prefill（$T=B\times S$）：

$\text{FLOPs}_{MLA}^{prefill}=2\times(B\times S)\times 187M+2\times B\times 128\times S^2\times 320$Decode（$T=B$）：

$$\text{FLOPs}_{MLA}^{decode}=2\times B\times 187M+2\times B\times 128\times \text{KVLen}\times 320$$

d) 所有层计算量

**Prefill 计算量**（61层）：

$\text{FLOPs}_{MLA}^{prefill,total}=61\times[2\times(B\times S)\times 187M+2\times B\times 128\times S^2\times 320]$

**Decode 计算量**（61层）：

$$\text{FLOPs}_{MLA}^{decode,total}=61\times[2\times B\times 187M+2\times B\times 128\times \text{KVLen}\times 320]$$

4) 数据搬运量

| 阶段 | 权重搬运 | KV Cache 搬运 |
|---|---|---|
| Prefill | $P_{MLA}^{total}\times$ Precision = 11.4GB | 无（正在生成） |
| Decode | $P_{MLA}^{total}\times$ Precision = 11.4GB | $B\times \text{KVLen}\times L\times 576\times$ Precision |

5) 数据通讯量

a) **TP 场景**（MLA 按头数切分）：每层 2 次 AllReduce：Q投影后、输

出投影后

$$\text{Comm}_{MLA}^{TP} = 2 \times L \times T \times H \times \text{Precision}$$

**第4条 FFN (Feed-Forward Network)层**

1) Dense MLP（前3层）

a) 参数量（单层）：使用SwiGLU激活函数，包含 3 个投影矩阵：

| 投影矩阵 | 维度 | 参数量 |
|---|---|---|
| $W_{gate}$ | $[H, I_{dense}]$ | 7,168 × 18,432 = 132M |
| $W_{up}$ | $[H, I_{dense}]$ | 7,168 × 18,432 = 132M |
| $W_{down}$ | $[I_{dense}, H]$ | 18,432 × 7,168 = 132M |
| **单层合计** | - | **396M** |

b) 参数量（3层）：

$$P_{DenseMLP}^{total} = 3 \times 396M = \mathbf{1.19B}$$

c) 计算量（单层）

| 操作 | 矩阵乘法形状 | FLOPs |
|---|---|---|
| Gate 投影 | $[T, H] \times [H, I]$ | $2 \times T \times H \times I_{dense}$ |
| Up 投影 | $[T, H] \times [H, I]$ | $2 \times T \times H \times I_{dense}$ |
| Down 投影 | $[T, I] \times [I, H]$ | $2 \times T \times I_{dense} \times H$ |
| **单层合计** | - | $2 \times T \times 3 \times H \times I_{dense} = 2$ |

d) 计算量（3层）**Prefill 计算量**（$T = B \times S$）：

$$\text{FLOPs}_{DenseMLP}^{prefill} = 3 \times 2 \times (B \times S) \times 396M = 2 \times (B \times S) \times 1.19B$$

**Decode 计算量**（$T = B$）：

$$\text{FLOPs}_{DenseMLP}^{decode} = 3 \times 2 \times B \times 396M = 2 \times B \times 1.19B$$

e) 显存占用

$$M_{DenseMLP} = 1.19B \times 1\text{Byte} = \mathbf{1.19\ GB}$$

f) 数据搬运量

$$\text{IO}_{DenseMLP} = P_{DenseMLP}^{total} \times \text{Precision} = 1.19\text{GB}$$

g) 数据通讯量

**TP 场景**：每层 1 次 AllReduce（Down投影后）

$$\text{Comm}_{DenseMLP}^{TP} = 3 \times T \times H \times \text{Precision}$$

2) MoE Layers（后58层）

a) 参数量（单层）：

| 组件 | 计算公式 | 数值 |
|---|---|---|
| 共享专家 | $N_s \times H \times I_{moe}$ | 1 × 7,168 × 2,048 × 3 =**44M** |
| 单个路由专家 | $H \times I_{moe} \times 3$ | 7,168 × 2,048 × 3 =**44M** |
| 所有路由专家 | $N_r \times 44M$ | 256 × 44M =**11.3B** |
| Gate 网络 | $H \times N_r$ | 7,168 × 256 =**1.8M** |

| | | |
|---|---|---|
| **单层合计** | - | **11.34B** |

b) 总激活参数量

$$P_{MoE}^{total} = 58 \times 11.34B = \mathbf{657.7B}$$

c) 激活参数量（单层）

由于 Top-K 路由，每个 Token 只激活$K_r = 8$个路由专家：

$$P_{MoE}^{active,layer} = (N_s + K_r) \times 44M = 9 \times 44M = \mathbf{396M}$$

d) 总激活参数量

$$P_{MoE}^{active} = 58 \times 396M = \mathbf{23.0B}$$

e) 计算量（单层）

| 组件 | FLOPs 公式 |
|---|---|
| Gate 网络 | $2 \times T \times H \times N_r$ |
| 共享专家 | $2 \times T \times N_s \times H \times I_{moe} \times 3$ |
| 路由专家（平均） | $2 \times T \times K_r \times H \times I_{moe} \times 3$ |
| **单层合计** | $2 \times T \times P_{MoE}^{active,layer}$ |

f) 总计算量

**Prefill 计算量**（$T = B \times S$）：

$\text{FLOPs}_{MoE}^{prefill} = 58 \times 2 \times (B \times S) \times 396M = 2 \times (B \times S) \times 23.0B$ Deco

de **计算量**（$T = B$）：

$$\text{FLOPs}_{MoE}^{decode} = 58 \times 2 \times B \times 396M = 2 \times B \times 23.0B$$

g) 显存占用

**静态权重**（必须全部加载）：

$$M_{MoE}^{static} = 657.7B \times 1\text{Byte} = \mathbf{657.7\ GB}$$

h) 数据搬运量**MoE 的 Sort & Batch 机制**：

当 Batch 中多个 Token 命中同一个专家时，硬件采用优化策略：

**Gating & Routing**：计算 Batch 中所有 Token 的路由分配

**Permutation （重排）**：将指向同一专家的 Token 聚拢成 Micro-Batch

**Grouped GEMM**：加载专家权重到 SRAM（**只搬运一次**），计算该专家的所有 Token这意味着：Small Batch 时权重无法复用（Memory-bound），Large Batch 时权重被多次复用、计算密度提高（可能转为 Compute-bound）。

| Batch Size | 激活专家数 | 权重搬运量 | 瓶颈类型 |
|---|---|---|---|
| 小 (≤8) | $\approx K_r = 8$ | $\approx 58 \times 9 \times 44M =$ | Memory-bound |
| 大 (>4096) | $\rightarrow N_r = 256$ | →657.7GB | Compute-bound |

i） 数据通讯量

| 通讯操作 | 公式 |
| --- | --- |
| Dispatch (All-to-All) | $T \times \frac{H}{\text{MOE}_{TP}} \times \text{FP8}$ |
| Combine (All-to-All) | $T \times \frac{H}{\text{MOE}_{TP}} \times \text{BF16}$ |
| **58 层总通讯** | $58 \times T \times \frac{H}{\text{MOE}_{TP}} \times (\text{FP8}$ |

**第5条 LM Head 层**

1） 参数量

$$P_{lmhead} = H \times V = 7{,}168 \times 129{,}280 = \mathbf{0.93B}$$

2） 计算量

a） 线性投影矩阵乘：$[T, H] \times [H, V]$

FLOPs：$2 \times T \times H \times V = 2 \times T \times 0.93B$

b） **Prefill**（通常只计算最后一个 Token）：

$$\text{FLOPs}_{lmhead}^{prefill} = 2 \times B \times H \times V = 2 \times B \times 0.93B$$

c） **Decode**（每个 Token 都需要）：

$$\text{FLOPs}_{lmhead}^{decode} = 2 \times B \times H \times V = 2 \times B \times 0.93B$$

3） 显存占用

$$M_{lmhead} = 0.93B \times 1\text{Byte} = \mathbf{0.93\ GB}$$

4） 数据搬运量

$$\text{IO}_{lmhead} = 0.93\text{GB}$$

5） 数据通讯量

**TP 场景**：按词表切分后 AllGather

$$\text{Comm}_{lmhead} = T \times V \times \text{Precision}$$

**第6条 RMSNorm**

1） **位置**：每层 MLA 前、FFN 前各一个 + 最终输出前一个 $=2 \times L + 1 = 123$ 个

2） **参数量**：

$$P_{RMSNorm} = 123 \times H = 123 \times 7{,}168 = \mathbf{0.88M}$$

3） **计算量**（每个 RMSNorm）：

$$\text{FLOPs}_{RMSNorm} = T \times H \times 5(\text{rsqrt},\ \text{mul},\ \text{add等})$$

**第7条 总计（Total）**

1） 参数量汇总

| 组件 | 参数量 | 占比 |
|---|---|---|
| Embedding | 0.93B | 0.14% |
| MLA (61 层) | 11.4B | 1.70% |
| Dense MLP (3 层) | 1.19B | 0.18% |
| MoE (58 层) | 657.7B | **98.0%** |
| LM Head | 0.93B | 0.14% |
| RMSNorm | 0.001B | <0.01% |
| **总参数量** | **671B** | 100% |

2） 激活参数量汇总**说明**：激活参数量是每次前向传播实际参与计算的参数量，决定了计算量。对于 MoE 模型，由于采用 Top-K 路由，每次只有部分专家参与计算。尽管总参数量为 671B，但推理计算量相当于一个 37B 的稠密模型。

| 组件 | 激活参数量 |
|---|---|
| Embedding | 0（查表） |
| MLA (61 层) | 11.4B |
| Dense MLP (3 层) | 1.19B |
| MoE (58 层，Top-8) | 23.0B |
| LM Head | 0.93B |
| **激活参数量** | **≈ 37B** |

3） 计算量汇总

a） **Prefill 阶段（$T = B \times S$）：简化公式验证：**

$\text{FLOPs}_{prefill}^{linear} = 2 \times (B \times S) \times P_{active} \approx 2 \times (B \times S) \times 37B$ **数值示例**（B=1，S=4096）：线性部分：

$\text{FLOPs}_{prefill}^{linear} = 2 \times 4096 \times 36.5B = 299$ TFLOPsAttention 二次项（与$S^2$相关）：

$$\text{FLOPs}_{attention} = 61 \times 2 \times 1 \times 128 \times 4096^2 \times 320 = 61 \times 2 \times 128 \times 320 \times S^2 = 5{,}013{,}504 \times S^2 \approx 5 \times 10^6 \times S^2$$

当$S = 4096$时：

$\text{FLOPs}_{attention} = 5{,}013{,}504 \times 16{,}777{,}216 \approx$ **84.1 TFLOPs** Prefill **总计：**

$$\text{FLOPs}_{prefill}^{total} = 299 + 84.1 = \mathbf{383.1\ TFLOPs}$$

| 组件 | FLOPs 公式 |
|---|---|
| Embedding | 0 |
| MLA 线性投影 | $2 \times T \times 11.4B$ |
| MLA Attention | $61 \times 2 \times B \times 128 \times S^2 \times 320$ |
| Dense MLP | $2 \times T \times 1.19B$ |
| MoE | $2 \times T \times 23.0B$ |

| | |
|---|---|
| LM Head | $2 \times B \times 0.93B$ |
| **线性部分合计** | $2 \times T \times (11.4 + 1.19 + 23.0 + 0.93)B =$ |

b) **Decode 阶段**（$T = B$，每次生成 1 个 Token）：**简化公式验证**：

$\text{FLOPs}_{decode}^{linear} = 2 \times B \times P_{active} \approx 2 \times B \times 37B$ **数值示例**（B=64, KVLen=4096）：线性部分：

$\text{FLOPs}_{decode}^{linear} = 2 \times 64 \times 36.5B = 4.67$ TFLOPs Attention 线性项（与 KVLen 相关）：

$$\text{FLOPs}_{attention} = 61 \times 2 \times B \times 128 \times \text{KVLen} \times 320$$
$$= 61 \times 2 \times 128 \times 320 \times B \times \text{KVLen} = 5{,}013{,}504 \times B \times \text{KVLen}$$

当$B = 64$,$\text{KVLen} = 4096$时：

$\text{FLOPs}_{attention} = 5{,}013{,}504 \times 64 \times 4096 \approx \mathbf{1.31\ TFLOPs}$Decode 总计：

$\text{FLOPs}_{decode}^{total} = 4.67 + 1.31 = \mathbf{5.98\ TFLOPs}$**注意**：Decode 阶段的 Attention 计算量与 KVLen 线性相关，随着生成过程的进行，KVLen 会逐渐增加。

| **组件** | **FLOPs 公式** |
|---|---|
| Embedding | 0 |
| MLA 线性投影 | $2 \times B \times 11.4B$ |
| MLA Attention | $61 \times 2 \times B \times 128 \times \text{KVLen} \times 320$ |
| Dense MLP | $2 \times B \times 1.19B$ |
| MoE | $2 \times B \times 23.0B$ |
| LM Head | $2 \times B \times 0.93B$ |
| **线性部分合计** | $2 \times B \times 36.5B$ |

4) 显存占用汇总

a) **静态权重**：

$$M_{static} = P_{total} \times \text{Precision} = 671B \times 1\text{Byte} = \mathbf{671\ GB}$$

b) **动态 KV Cache**（4K序列，Batch=64，FP8）：

$$M_{KV} = 61 \times 64 \times 4096 \times 576 \times 1 = 9.2\text{GB}$$

c) **总显存**：

$$M_{total} = M_{static} + M_{KV} = 671 + 9.2 = \mathbf{680.2\ GB}$$

d) 并行场景单芯片显存不同部分的权重是**分别存储**的，不能简单地将所有并行度乘在一起：

$\text{DRAM}_{chip} = \frac{M_{MLA}}{\text{MLA}_{TP} \times \text{MLA}_{DP}} + \frac{M_{MoE}}{\text{MOE}_{TP} \times \text{MOE}_{EP}} + \frac{M_{other}}{\text{TP} \times \text{DP}} + \frac{M_{KV}}{\text{MLA}_{TP} \times \text{MLA}_{DP}}$**示例**（64卡，TP=8, EP=8）：

$$\text{DRAM}_{chip} = \frac{11.4}{8} + \frac{657.7}{8} + \frac{2.9}{8} + \frac{9.2}{8} = 1.4 + 82.2 + 0.4 + 1.2 \approx \mathbf{85\ GB}$$ **注意**：使用 Sequence Parallelism (SP) 时，KV Cache 会在 TP 维度上切分；Decode 阶段 KV Cache 会随生成 token 增加而增长。

5) 数据搬运量汇总

a) **Decode**（主要瓶颈）：

$$\text{IO}_{decode} = P_{active} \times \text{Precision} + M_{KV} + B \times H \times L \times \text{Precision}$$

6) 数据通讯量汇总

| 通讯类型 | 公式 | 场景 |
|---|---|---|
| TP AllReduce | $2 \times L \times T \times H \times$ Pre | MLA + FFN |
| EP All-to-All | $58 \times T \times H \times$ (FP8 + | MoE |
| SP AllGather | $T \times H \times$ Precision | 长序列 |

**第8条　延迟与吞吐量指标**

1) 计算时间**MFU (Model FLOPs Utilization) 影响因素：TP 并行：**

$$T_{comp}^{TP} = \frac{T_{comp}}{\text{TP}}$$

$$T_{comp} = \frac{\text{FLOPs}}{\text{Peak FLOPS} \times \text{MFU}}$$

| 阶段 | MFU 典型值 | 瓶颈类型 |
|---|---|---|
| Prefill | 50%-60% | Compute-bound |
| Decode | 30%-40% | Memory-bound |

a) **数据类型**：FP8/INT8 > BF16 > FP32

b) **矩阵形状**：大矩阵利用率高，小矩阵利用率低

c) **并行策略**：TP/DP 影响单芯片计算量

d) **访存瓶颈**：访存成为瓶颈时 MFU 降低

2) 访存时间

$$T_{mem} = \frac{\text{Total IO}}{\text{Memory Bandwidth}}$$

3) 通讯时间**通讯掩盖**：

$T_{comm}^{effective} = \max(0, T_{comm} - T_{overlap})$现代框架支持通讯与计算重叠：**Dispatch**与**MLA 计算 + Shared Expert 计算**重叠，**Combine**与**Routed Expert 计算**重叠，**AllReduce**与**下一层前向计算**重叠。

$$T_{comm} = \frac{\text{Comm Volume}}{\text{Network Bandwidth}}$$

4) 单层延迟

$$T_{layer} = \max(T_{comp}, T_{mem}) + T_{comm}^{effective}$$

5） 总延迟**Decode**：

$$T_{decode} = L \times T_{layer} + T_{overhead}$$

6） 吞吐量（TPS）**TPS per Chip 推导**：

$\text{TPS}_{per\_chip} = \frac{\text{TotalBatch}}{\text{DecodeTime (s)} \times \text{NumChips}} = \frac{B \times \text{TP} \times \text{DP}}{T_{decode} \times \text{TP} \times \text{DP}} = \frac{B}{T_{decode}}$**优化目标**：在满足 SLO 约束（TPS per Batch ≥ 10）的前提下，最大化 TPS per Chip。

| **指标** | **公式** | **含义** |
|---|---|---|
| TPS per Batch | $\frac{1000}{T_{decode}\ (\text{ms})}$ | 用户体验 |
| TPS per Chip | $\frac{B}{T_{decode}\ (\text{s})}$ | 成本效益 |
| Total TPS | $\text{TPS}_{chip} \times \text{NumChip}$ | 集群吞吐 |

7） 首 Token 延迟（FTL）

$$\text{FTL} = T_{prefill} \approx \frac{\text{FLOPs}_{prefill}}{\text{Peak FLOPS} \times \text{MFU}} + T_{comm}^{prefill}$$

## 第9条 关键公式速查

1） 参数量 → 计算量**核心关系**：对于线性层，$\text{FLOPs} = 2 \times T \times P$

| **组件** | **单层参数量** | **单层 FLOPs（线性部分）** |
|---|---|---|
| MLA | 187M | $2 \times T \times 187M$ |
| Dense MLP | 396M | $2 \times T \times 396M$ |
| MoE (激活) | 396M | $2 \times T \times 396M$ |
| LM Head | 0.93B | $2 \times T \times 0.93B$ |

2） Prefill vs Decode

| **指标** | **Prefill** | **Decode** |
|---|---|---|
| Token 数 T | $B \times S$ | B |
| FLOPs | $2 \times (B \times S) \times P_{active}$ | $2 \times B \times P_{active}$ |
| Attention | $O(S^2)$ | $O$(KVLen) |
| 瓶颈 | Compute-bound | Memory-bound |