

Предобучение DeepSeek v3

Хрущев Михаил
руководитель группы претрейна YandexGPT

Познакомимся

- Меня зовут Михаил, я руковожу командой претрейна YandexGPT.
- Мы делаем претрейн - нагружаем модели знаниями и навыками из открытых источников.

Некоторые наши достижения:

- YaLM 100B
- YaFSDP - наш аналог FSDP, быстрее до 47%



Познакомимся

- Меня зовут Михаил, я руководитель командой претрейна YandexGPT.
- Мы делаем претрейн - нагружаем модели знаниями и навыками из открытых источников.

Некоторые наши достижения:

- YaLM 100B
- YaFSDP
- Почти все языковые модели в Яндексе



Особенности претрейнов

- Это дорого: хороший претрейн требует месяцы обучения на тысячах GPU.
- Ускорение обучений даже на 10% может привести к экономии сотен миллионов рублей в год.

План доклада

- Проблемы масштабирования претрейнов 70B+ моделей
- DeepSeek v3
- MLA
- Изменения в MoE
- Оптимизации экспертного параллелизма
- DualPipe
- Block-wise FP8
- Итоги

Проблемы масштабирования 70B+ моделей

Оптимальная схема обучения моделей до 70В

- YaFSDP
- Тензорный параллелизм для уменьшения размера активаций

Оптимальная схема обучения моделей до 70В

- YaFSDP
- Тензорный параллелизм для уменьшения размера активаций

После 70В такая схема перестает работать:

- Активации начинают занимать слишком много
- Тензорный параллелизм приводит к уменьшению числа активаций, но увеличению числа коммуникаций.
- Что делать?

Проблемы масштабирования. DP+TP

Рассмотрим схему YaFSDP + TP:

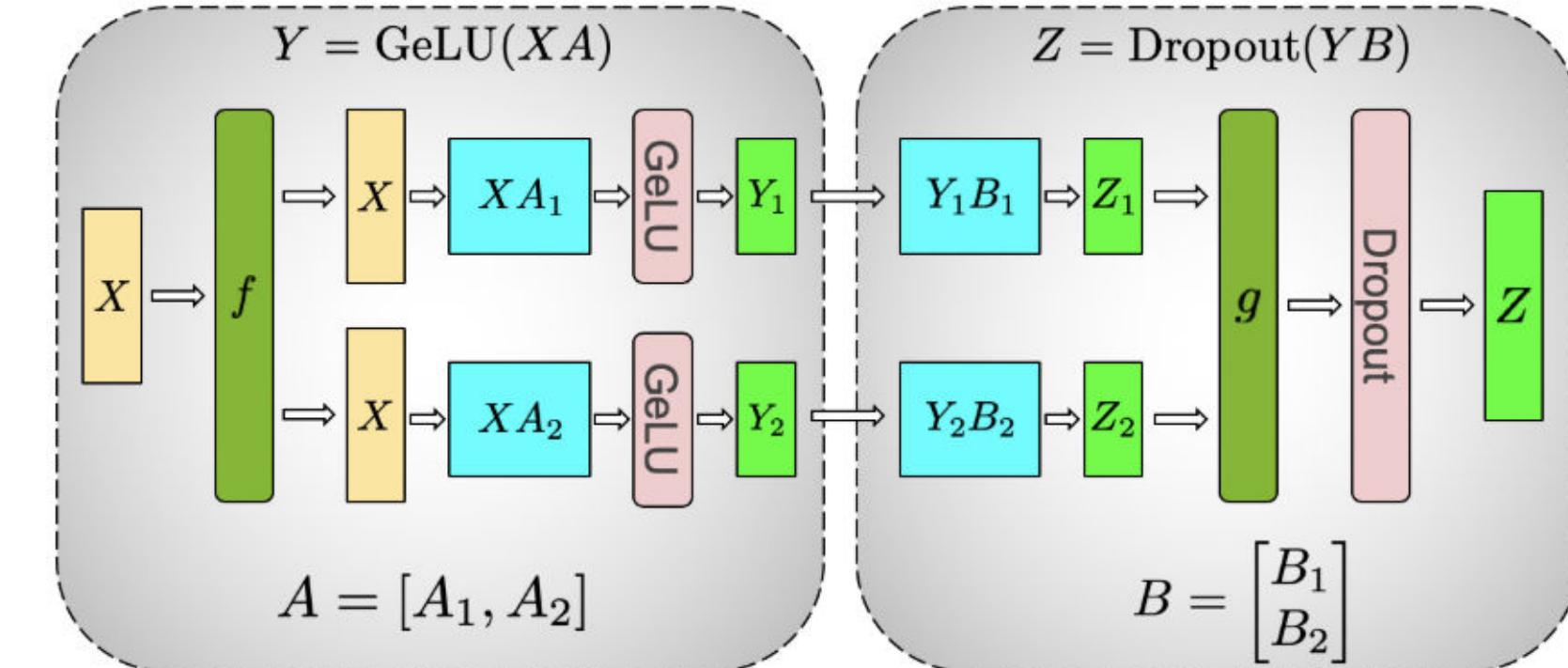
- Объем коммуникаций YaFSDP на итерацию = $6*|P|$ байт, где $|P|$ - число параметров модели.
- Компьют на итерацию линейно зависит от числа параметров: $O(|P|)$.
- Объем хранимых активаций без чекпоинта активаций растет сублинейно с увеличением размера модели.
- Тензорный параллелизм позволяет уменьшить число активаций и компьют в TP раз.

Проблемы масштабирования. DP+TP

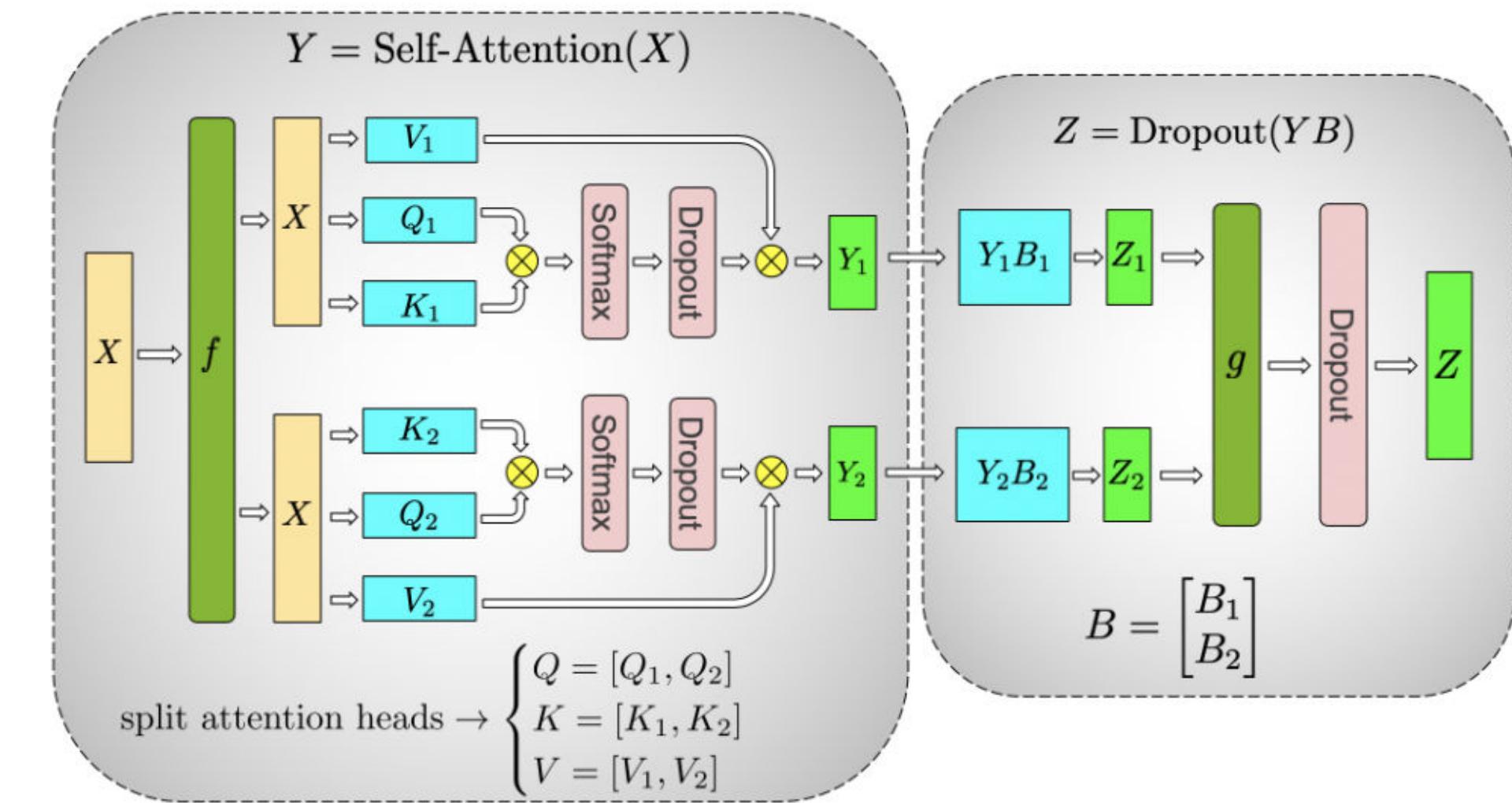
Рассмотрим схему YaFSDP + TP:

- Объем коммуникаций YaFSDP на итерацию = $6*|P|$ байт, где $|P|$ - число параметров модели.
- Компьют на итерацию линейно зависит от числа параметров: $O(|P|)$.
- Объем хранимых активаций без чекпоинта активаций растет сублинейно с увеличением размера модели.
- Тензорный параллелизм позволяет уменьшить число активаций и компьют в TP раз. **Но не коммуникаций.**

Почему при $TP > 1$ размер коммуникации не уменьшается?



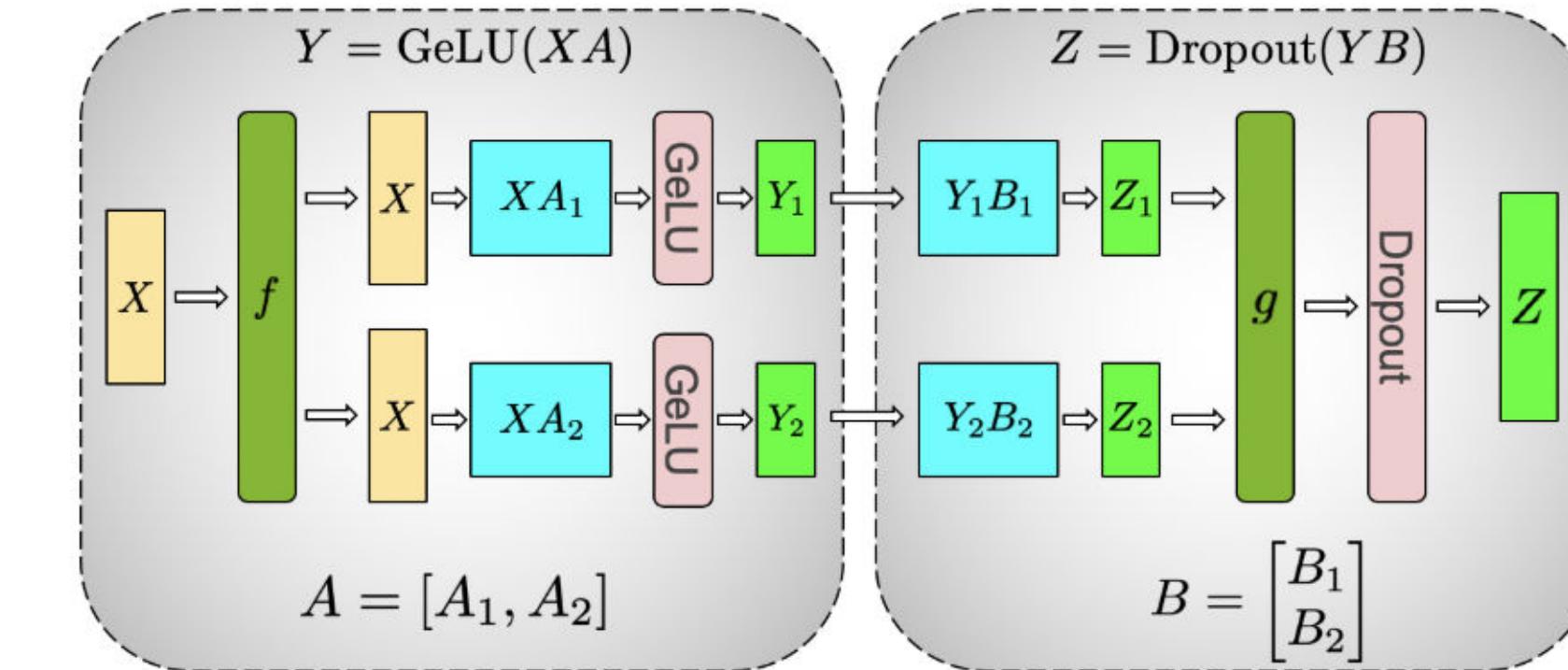
(a) MLP



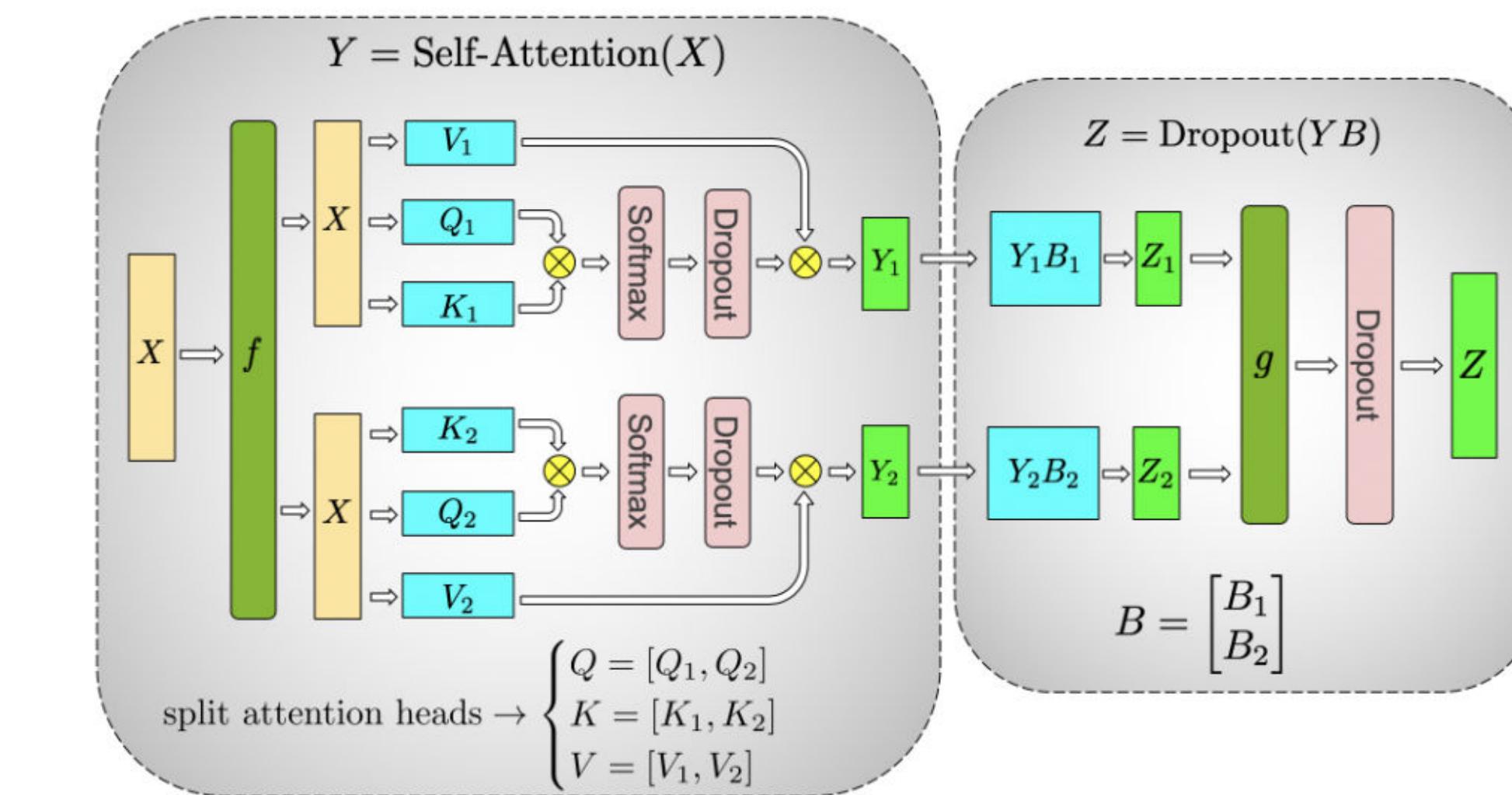
(b) Self attention

Почему при $TP > 1$ размер коммуникации не уменьшается?

- Мы шардируем модель внутри хоста - поэтому ограничены пропускной способностью хоста



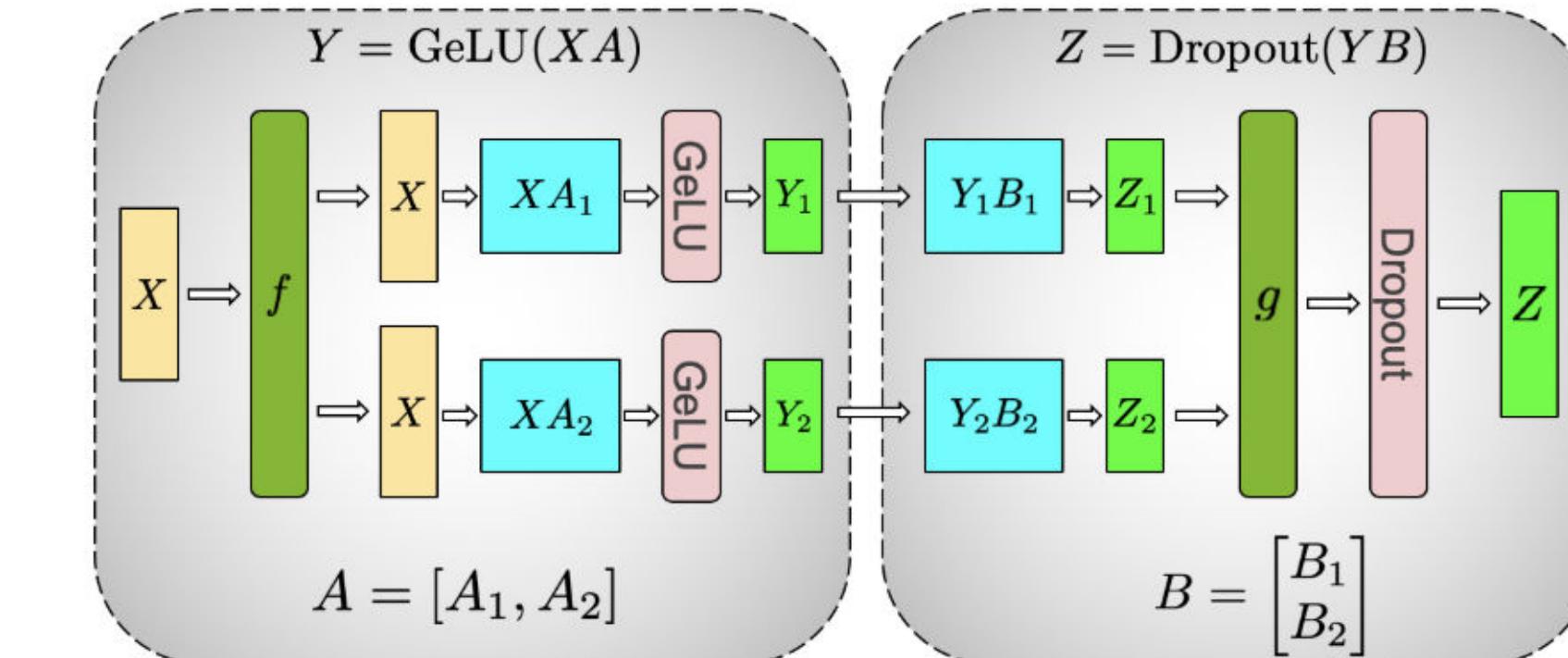
(a) MLP



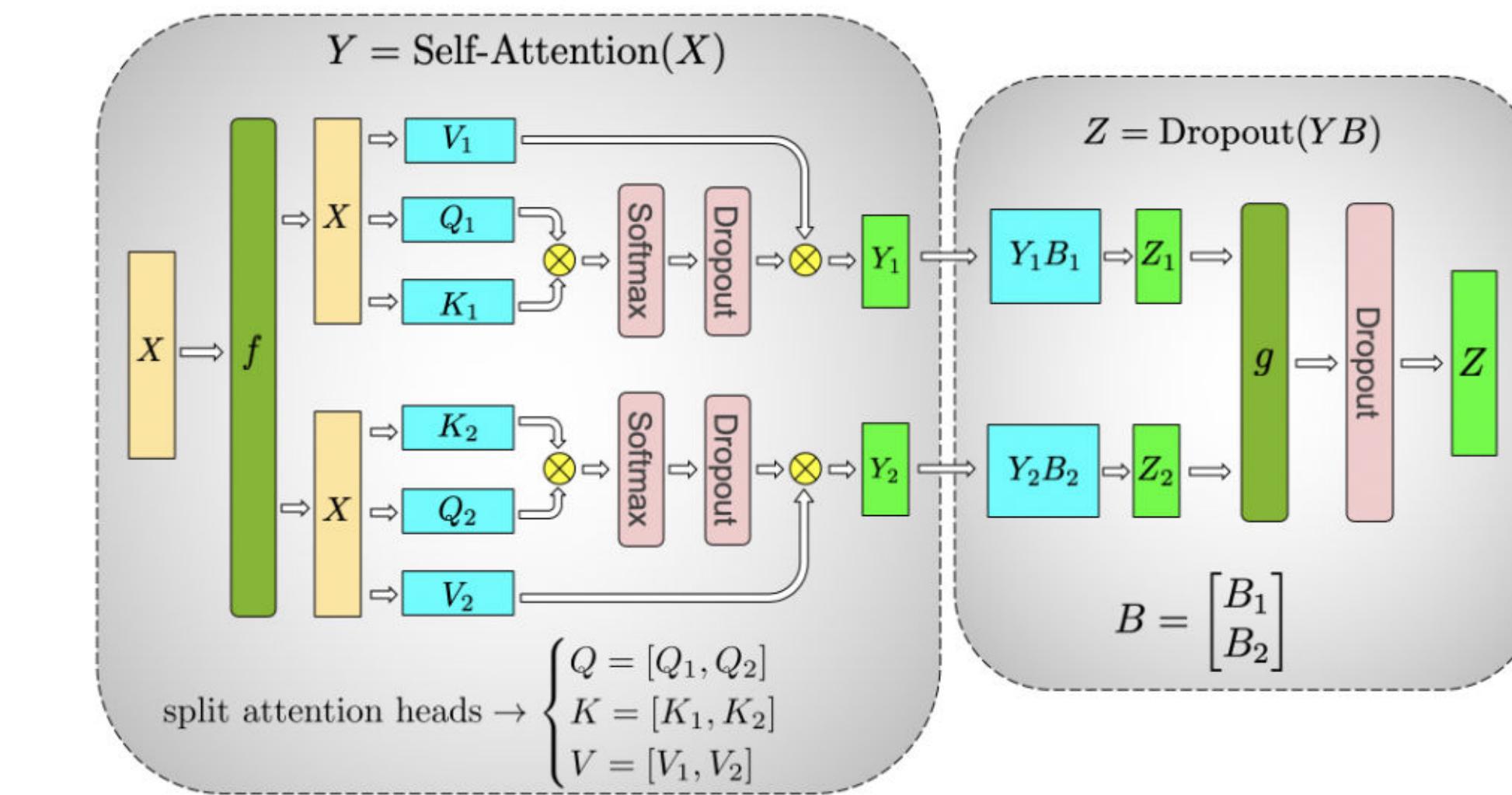
(b) Self attention

Почему при $TP > 1$ размер коммуникации не уменьшается?

- Мы шардируем модель внутри хоста - поэтому ограничены пропускной способностью хоста
- Если мы шардируем модель внутри хоста - мы все так же должны синхронизировать всю модель через узкий боттлнек: пропускную способность хоста



(a) MLP



(b) Self attention

Проблемы масштабирования. DP+TP

Compute = $O(|P|)$ / ТР

Activations = $|A|$ / ТР

YaFSDP коммуникации = $O(|P|)$

Проблемы масштабирования. DP+TP

Compute = $O(|P|)$ / ТР

Activations = $|A|$ / ТР

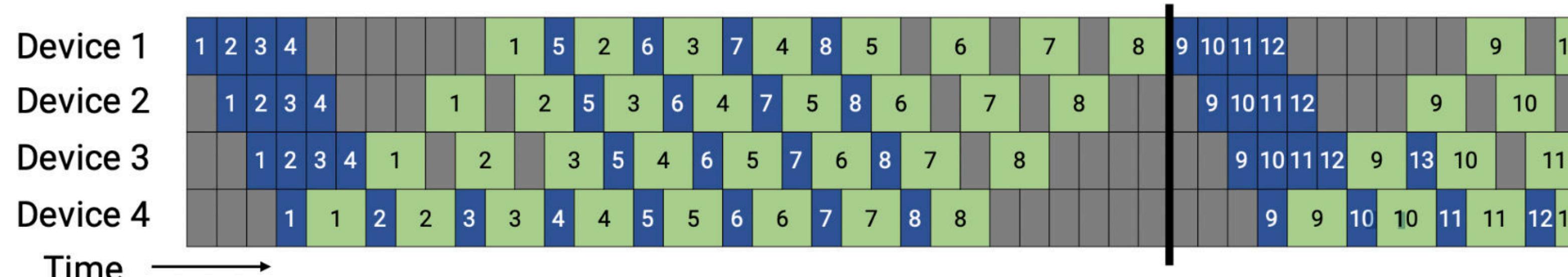
YaFSDP коммуникации = $O(|P|)$

При увеличении ТР объем коммуникаций не получится перекрыть вычислениями!

Как увеличивать модели дальше?

Шардировать модель по хостам:

- Межхостовый тензорный параллелизм - большие и частые коммуникации между хостами.
- Экспертный параллелизм - более редкие и разреженные коммуникации между хостами.
- Пайpline параллелизм - пузыри съедают производительность.



DeepSeek v3

DeepSeek v3

- Гигантская модель: 671B MoE с 32B активными параметрами.
- По умности на январь 2025 года сопоставима с топовыми моделями OpenAI и Anthropic
- На обучение было затрачено 2 месяца 2048 H800!

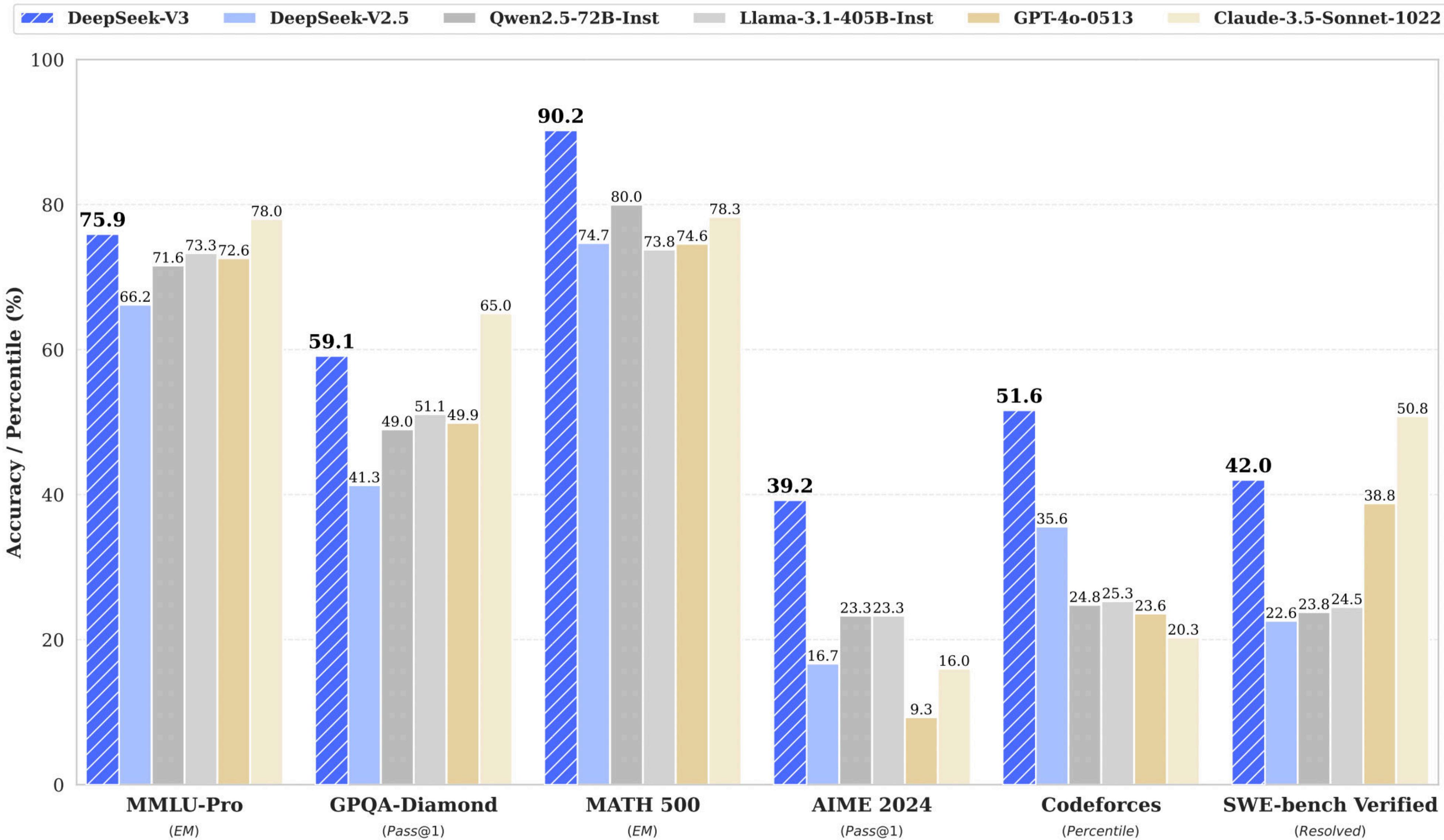


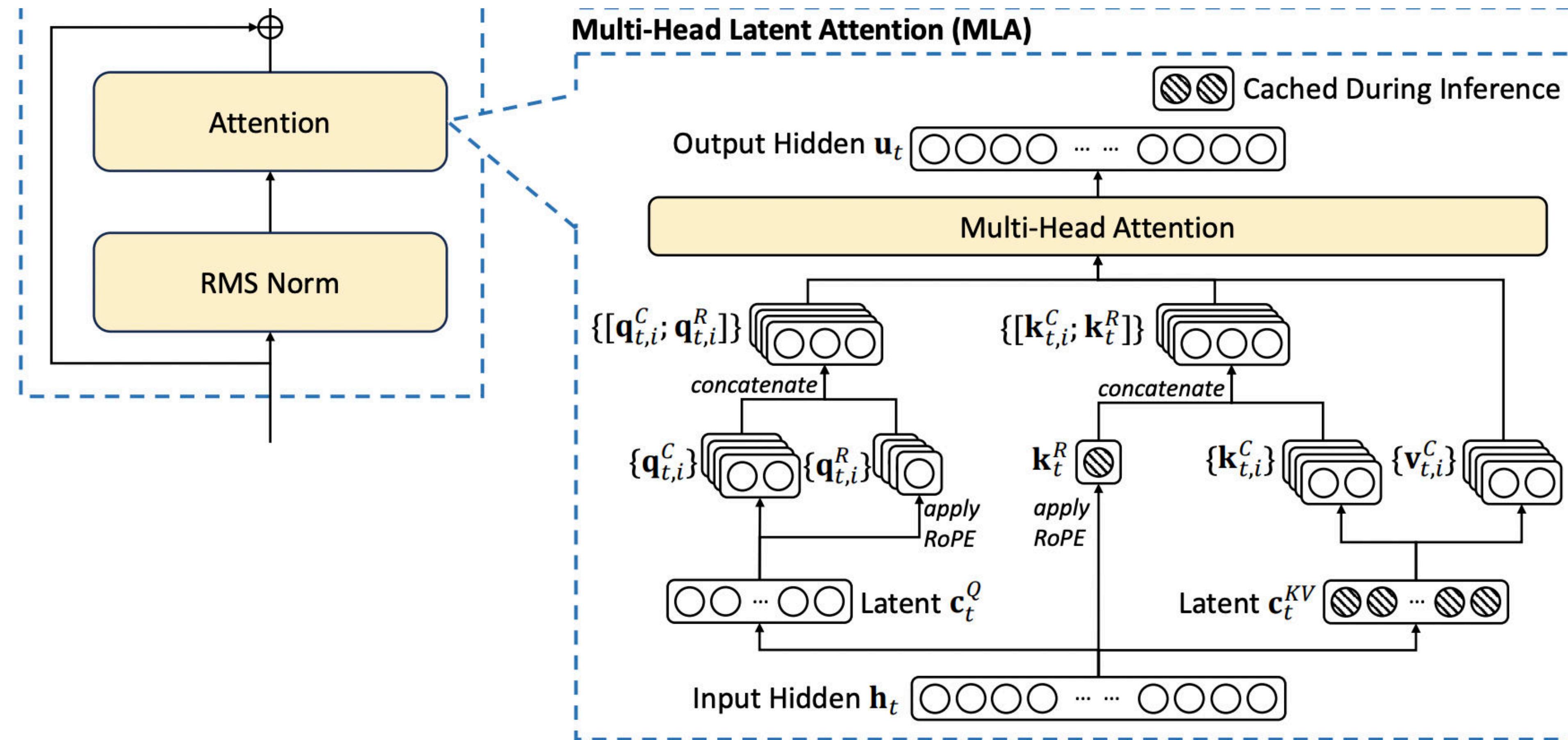
Figure 1 | Benchmark performance of DeepSeek-V3 and its counterparts.

Особенности предобучения DeepSeek v3

- MLA - Attention с потреблением памяти, как у GQA, но качеством МНА
- Block-wise FP8 - более стабильная версия FP8. Это первый известный большой претрейн на FP8.
- MoE (8 active, 256 total) - более масштабируемый и контролируемый роутинг
- Expert parallelism - более легковесная и управляемая схема параллелизма (есть вопросы)
- DualPipe (PP=16) - схема пайpline параллелизма с перекрытием EP-коммуникаций (есть очень много вопросов =))
- Multi-token prediction

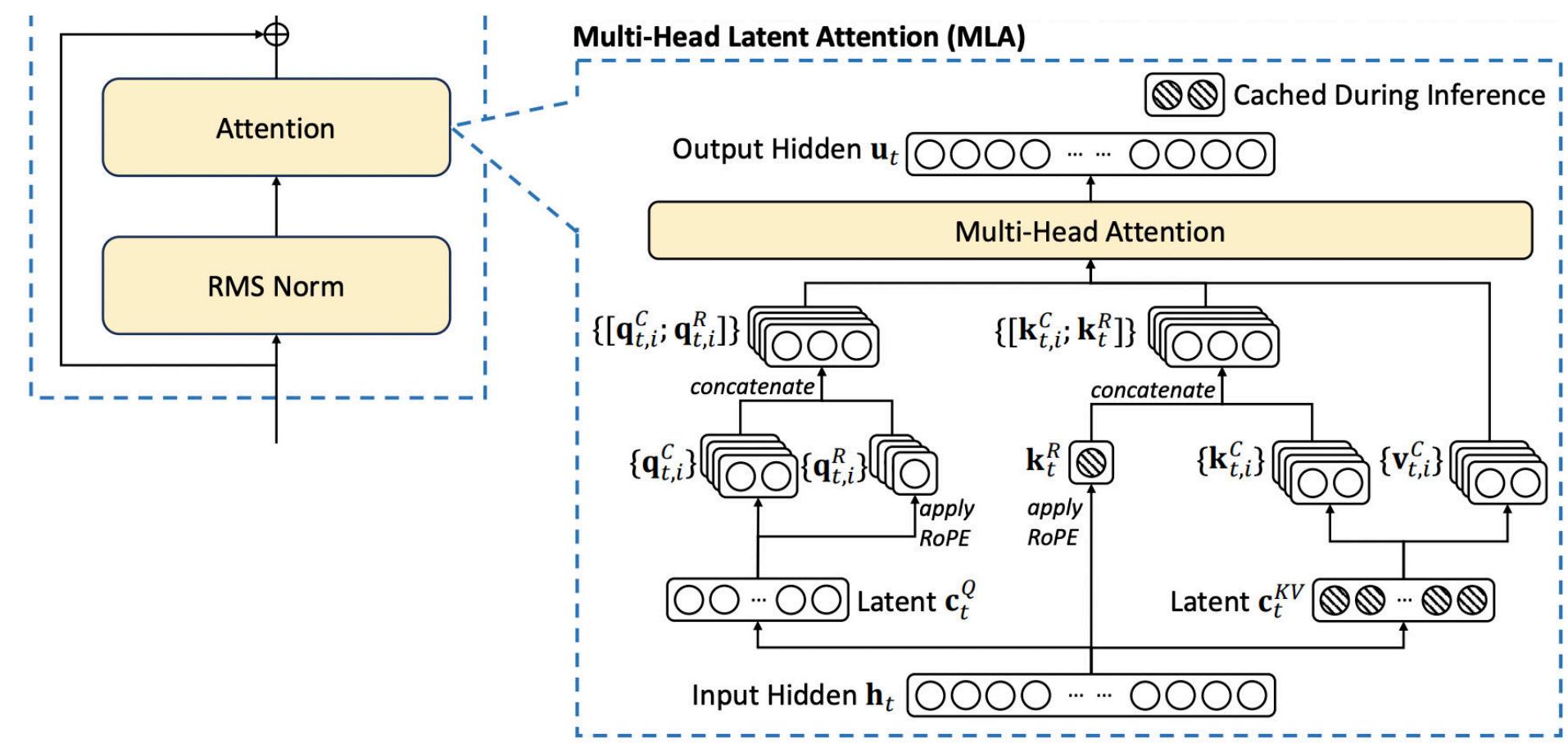
MLA

Multi-head Latent Attention (MLA)



MLA

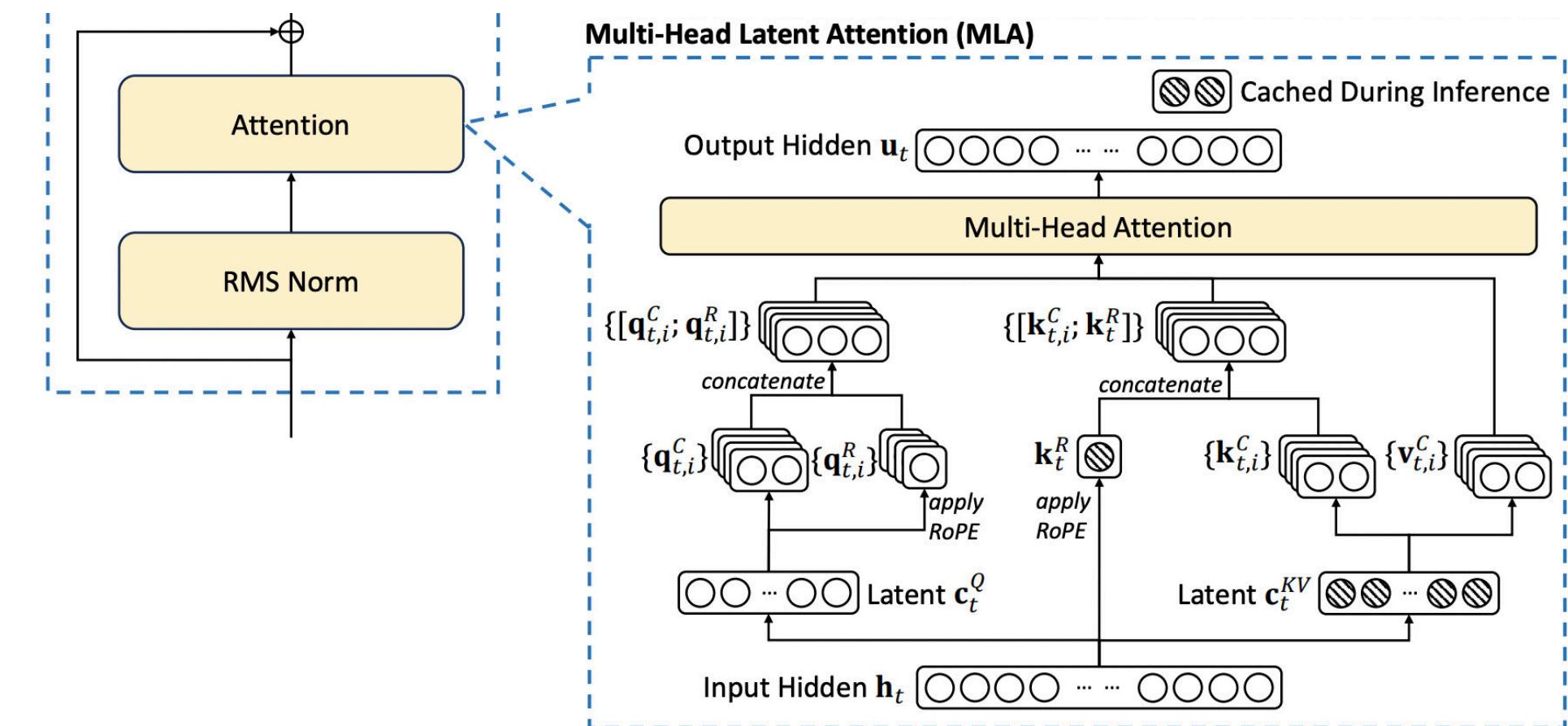
$$\begin{aligned}
 \mathbf{c}_t^{KV} &= W^{DKV} \mathbf{h}_t, \\
 [\mathbf{k}_{t,1}^C; \mathbf{k}_{t,2}^C; \dots; \mathbf{k}_{t,n_h}^C] &= \mathbf{k}_t^C = W^{UK} \mathbf{c}_t^{KV}, \\
 \mathbf{k}_t^R &= \text{RoPE}(W^{KR} \mathbf{h}_t), \\
 \mathbf{k}_{t,i} &= [\mathbf{k}_{t,i}^C; \mathbf{k}_t^R], \\
 [\mathbf{v}_{t,1}^C; \mathbf{v}_{t,2}^C; \dots; \mathbf{v}_{t,n_h}^C] &= \mathbf{v}_t^C = W^{UV} \mathbf{c}_t^{KV},
 \end{aligned}$$



MLA

$$\begin{aligned} \mathbf{c}_t^{KV} &= W^{DKV} \mathbf{h}_t, \\ [\mathbf{k}_{t,1}^C; \mathbf{k}_{t,2}^C; \dots; \mathbf{k}_{t,n_h}^C] &= \mathbf{k}_t^C = W^{UK} \mathbf{c}_t^{KV}, \\ \mathbf{k}_t^R &= \text{RoPE}(W^{KR} \mathbf{h}_t), \\ \mathbf{k}_{t,i} &= [\mathbf{k}_{t,i}^C; \mathbf{k}_t^R], \\ [\mathbf{v}_{t,1}^C; \mathbf{v}_{t,2}^C; \dots; \mathbf{v}_{t,n_h}^C] &= \mathbf{v}_t^C = W^{UV} \mathbf{c}_t^{KV}, \end{aligned}$$

$$\begin{aligned} \mathbf{c}_t^Q &= W^{DQ} \mathbf{h}_t, \\ [\mathbf{q}_{t,1}^C; \mathbf{q}_{t,2}^C; \dots; \mathbf{q}_{t,n_h}^C] &= \mathbf{q}_t^C = W^{UQ} \mathbf{c}_t^Q, \\ [\mathbf{q}_{t,1}^R; \mathbf{q}_{t,2}^R; \dots; \mathbf{q}_{t,n_h}^R] &= \mathbf{q}_t^R = \text{RoPE}(W^{QR} \mathbf{c}_t^Q), \\ \mathbf{q}_{t,i} &= [\mathbf{q}_{t,i}^C; \mathbf{q}_{t,i}^R], \end{aligned}$$



MLA

Attention Mechanism	KV Cache per Token (# Element)	Capability
Multi-Head Attention (MHA)	$2n_h d_{hl}$	Strong
Grouped-Query Attention (GQA)	$2n_g d_{hl}$	Moderate
Multi-Query Attention (MQA)	$2d_{hl}$	Weak
MLA (Ours)	$(d_c + d_h^R)l \approx \frac{9}{2}d_{hl}$	Stronger

Ablations

Benchmark (Metric)	# Shots	Dense 7B w/ MQA	Dense 7B w/ GQA (8 Groups)	Dense 7B w/ MHA
# Params	-	7.1B	6.9B	6.9B
BBH (EM)	3-shot	33.2	35.6	37.0
MMLU (Acc.)	5-shot	37.9	41.2	45.2
C-Eval (Acc.)	5-shot	30.0	37.7	42.9
CMMLU (Acc.)	5-shot	34.6	38.4	43.5

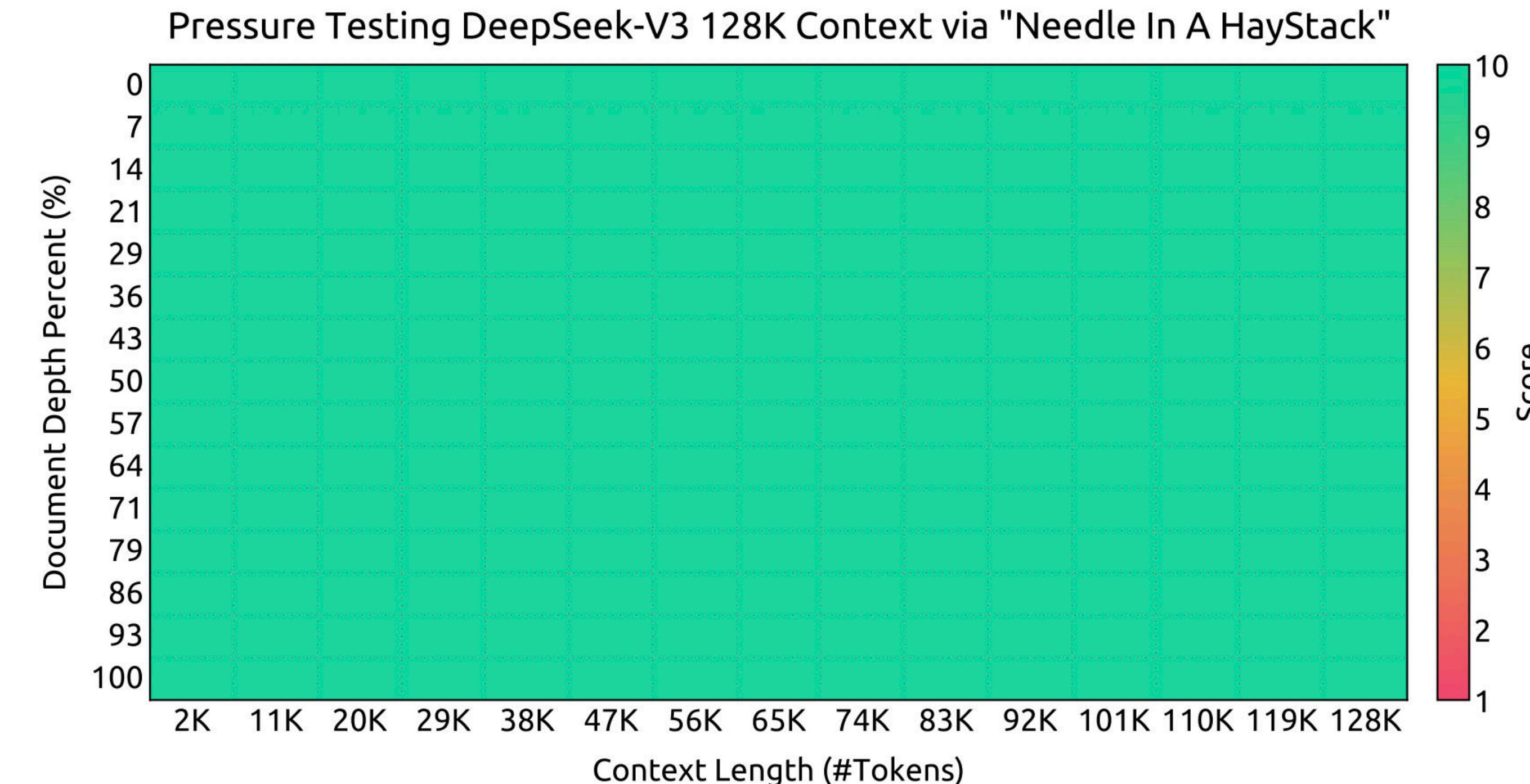
Table 8 | Comparison among 7B dense models with MHA, GQA, and MQA, respectively. MHA demonstrates significant advantages over GQA and MQA on hard benchmarks.

Benchmark (Metric)	# Shots	Small MoE w/ MHA	Small MoE w/ MLA	Large MoE w/ MHA	Large MoE w/ MLA
# Activated Params	-	2.5B	2.4B	25.0B	21.5B
# Total Params	-	15.8B	15.7B	250.8B	247.4B
KV Cache per Token (# Element)	-	110.6K	15.6K	860.2K	34.6K
BBH (EM)	3-shot	37.9	39.0	46.6	50.7
MMLU (Acc.)	5-shot	48.7	50.0	57.5	59.0
C-Eval (Acc.)	5-shot	51.6	50.9	57.9	59.2
CMMLU (Acc.)	5-shot	52.3	53.4	60.7	62.5

Table 9 | Comparison between MLA and MHA on hard benchmarks. DeepSeek-V2 shows better performance than MHA, but requires a significantly smaller amount of KV cache.

Расширение контекста

- Используется YARN-подход
- Достигается 100% качество на иголке



MoE

Изменения в MoE

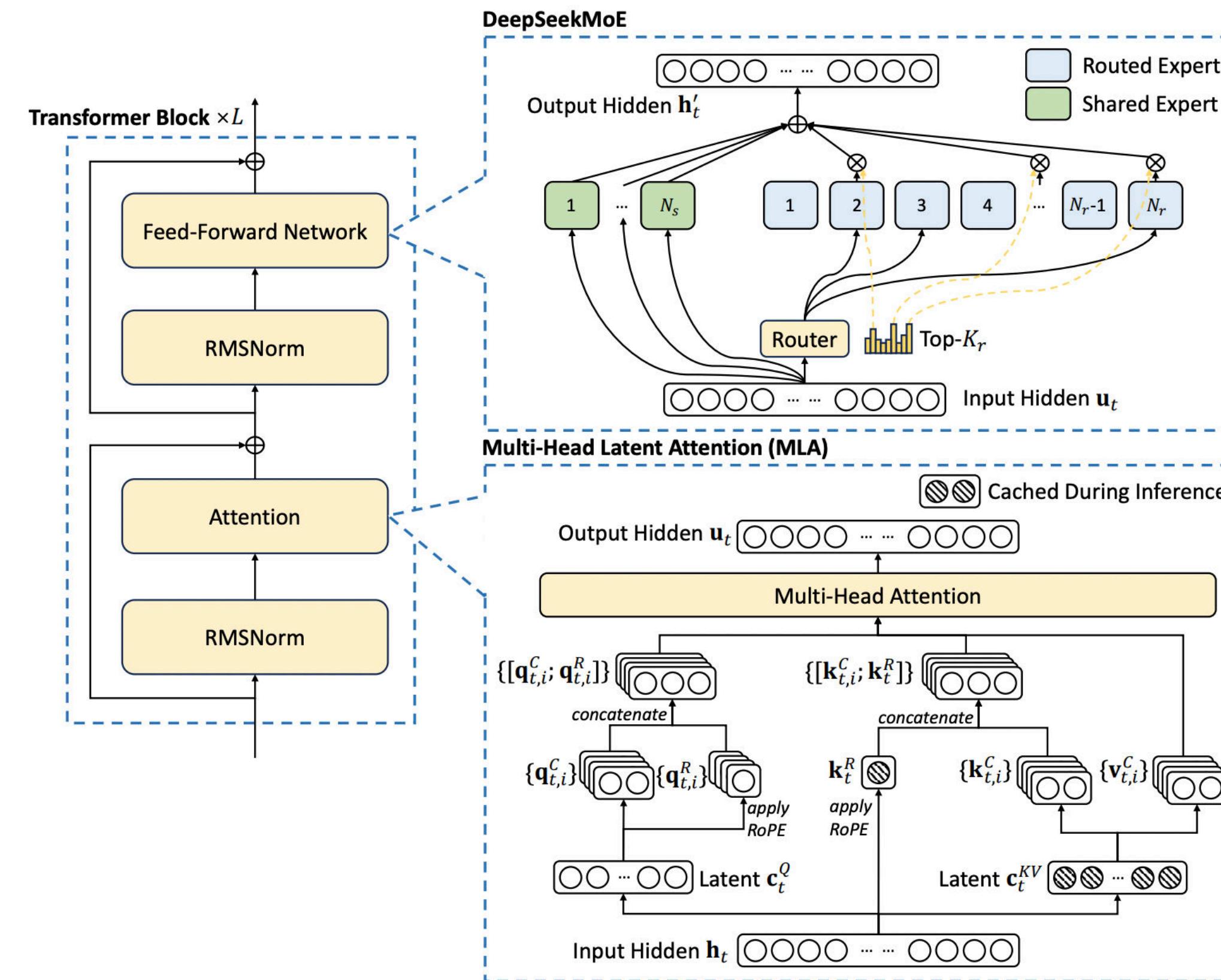
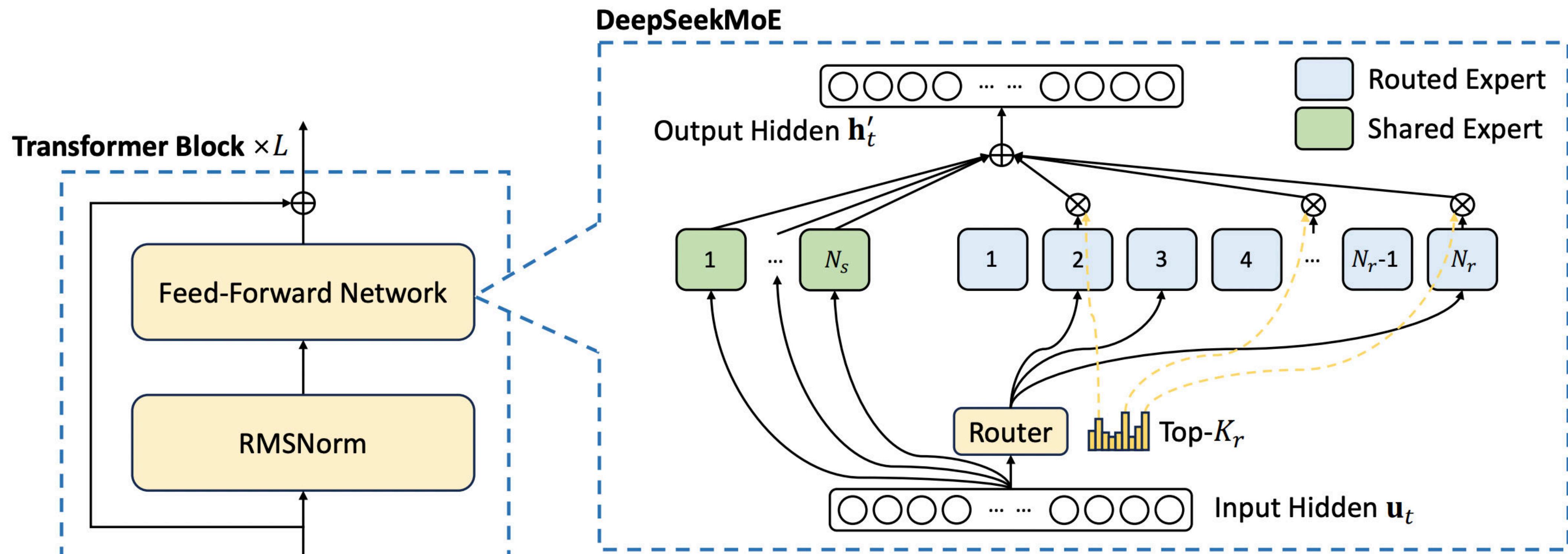


Figure 2 | Illustration of the basic architecture of DeepSeek-V3. Following DeepSeek-V2, we adopt MLA and DeepSeekMoE for efficient inference and economical training.

Изменения в MoE



Роутинг

$$\mathbf{h}'_t = \mathbf{u}_t + \sum_{i=1}^{N_s} \text{FFN}_i^{(s)}(\mathbf{u}_t) + \sum_{i=1}^{N_r} g_{i,t} \text{FFN}_i^{(r)}(\mathbf{u}_t),$$

$$g_{i,t} = \frac{g'_{i,t}}{\sum_{j=1}^{N_r} g'_{j,t}},$$

$$g'_{i,t} = \begin{cases} s_{i,t}, & s_{i,t} \in \text{Topk}(\{s_{j,t} | 1 \leq j \leq N_r\}, K_r), \\ 0, & \text{otherwise,} \end{cases}$$

$$s_{i,t} = \text{Sigmoid}(\mathbf{u}_t^T \mathbf{e}_i),$$

Роутинг

$$\mathbf{h}'_t = \mathbf{u}_t + \sum_{i=1}^{N_s} \text{FFN}_i^{(s)}(\mathbf{u}_t) + \sum_{i=1}^{N_r} g_{i,t} \text{FFN}_i^{(r)}(\mathbf{u}_t),$$

$$g_{i,t} = \frac{g'_{i,t}}{\sum_{j=1}^{N_r} g'_{j,t}},$$

$$g'_{i,t} = \begin{cases} s_{i,t}, & s_{i,t} \in \text{Topk}(\{s_{j,t} | 1 \leq j \leq N_r\}, K_r), \\ 0, & \text{otherwise,} \end{cases}$$

$$s_{i,t} = \text{Sigmoid}(\mathbf{u}_t^T \mathbf{e}_i),$$

Softmax плохо масштабируется на большом числе экспертов.

Но он и не нужен =)

Роутинг

$$\mathbf{h}'_t = \mathbf{u}_t + \sum_{i=1}^{N_s} \text{FFN}_i^{(s)}(\mathbf{u}_t) + \sum_{i=1}^{N_r} g_{i,t} \text{FFN}_i^{(r)}(\mathbf{u}_t),$$

$$g_{i,t} = \frac{g'_{i,t}}{\sum_{j=1}^{N_r} g'_{j,t}},$$

$$g'_{i,t} = \begin{cases} s_{i,t}, & s_{i,t} + b_i \in \text{Topk}(\underline{s_{j,t} + b_j} | 1 \leq j \leq N_r, K_r), \\ 0, & \text{otherwise.} \end{cases}$$

$$s_{i,t} = \text{Sigmoid}(\mathbf{u}_t^T \mathbf{e}_i),$$

Роутинг

$$\mathbf{h}'_t = \mathbf{u}_t + \sum_{i=1}^{N_s} \text{FFN}_i^{(s)}(\mathbf{u}_t) + \sum_{i=1}^{N_r} g_{i,t} \text{FFN}_i^{(r)}(\mathbf{u}_t),$$

$$g_{i,t} = \frac{g'_{i,t}}{\sum_{j=1}^{N_r} g'_{j,t}},$$

$$g'_{i,t} = \begin{cases} s_{i,t}, & s_{i,t} + b_i \in \text{Topk}(\underline{s_{j,t} + b_j} | 1 \leq j \leq N_r), K_r), \\ 0, & \text{otherwise.} \end{cases}$$

$$s_{i,t} = \text{Sigmoid}(\mathbf{u}_t^T \mathbf{e}_i),$$

- Вводим bias для каждого эксперта. Через него мы можем влиять на балансировку экспертов

Балансировка экспертов

$$\mathbf{h}'_t = \mathbf{u}_t + \sum_{i=1}^{N_s} \text{FFN}_i^{(s)}(\mathbf{u}_t) + \sum_{i=1}^{N_r} g_{i,t} \text{FFN}_i^{(r)}(\mathbf{u}_t),$$

$$g_{i,t} = \frac{g'_{i,t}}{\sum_{j=1}^{N_r} g'_{j,t}},$$

$$g'_{i,t} = \begin{cases} s_{i,t}, & s_{i,t} + b_i \in \text{Topk}(\{s_{j,t} + b_j | 1 \leq j \leq N_r\}, K_r), \\ 0, & \text{otherwise.} \end{cases}$$

$$s_{i,t} = \text{Sigmoid}(\mathbf{u}_t^T \mathbf{e}_i),$$

- Если эксперт i вызывается слишком часто, уменьшаем b_i на гамма
- Иначе: увеличиваем b_i на гамма

Complementary Sequence-Wise Auxiliary Loss

Complementary Sequence-Wise Auxiliary Loss. Although DeepSeek-V3 mainly relies on the auxiliary-loss-free strategy for load balance, to prevent extreme imbalance within any single sequence, we also employ a complementary sequence-wise balance loss:

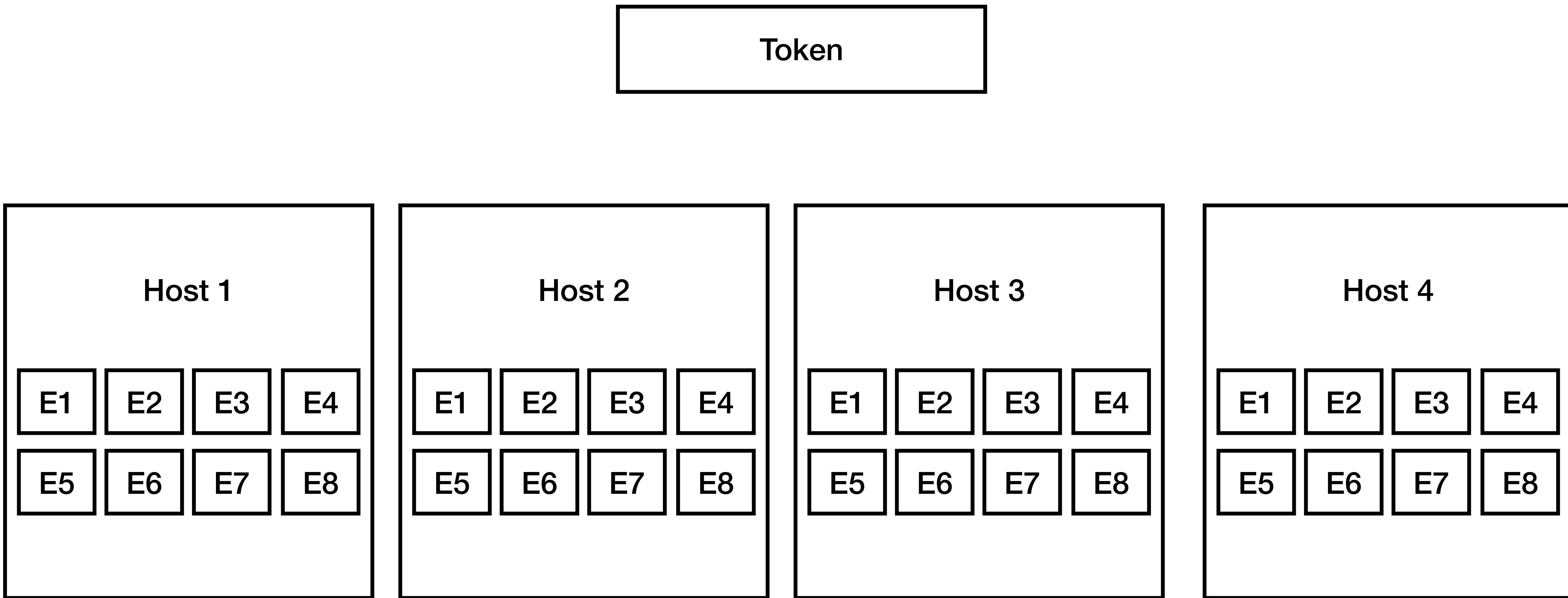
$$\mathcal{L}_{\text{Bal}} = \alpha \sum_{i=1}^{N_r} f_i P_i,$$

$$f_i = \frac{N_r}{K_r T} \sum_{t=1}^T \mathbb{1} (s_{i,t} \in \text{Topk}(\{s_{j,t} | 1 \leq j \leq N_r\}, K_r)),$$

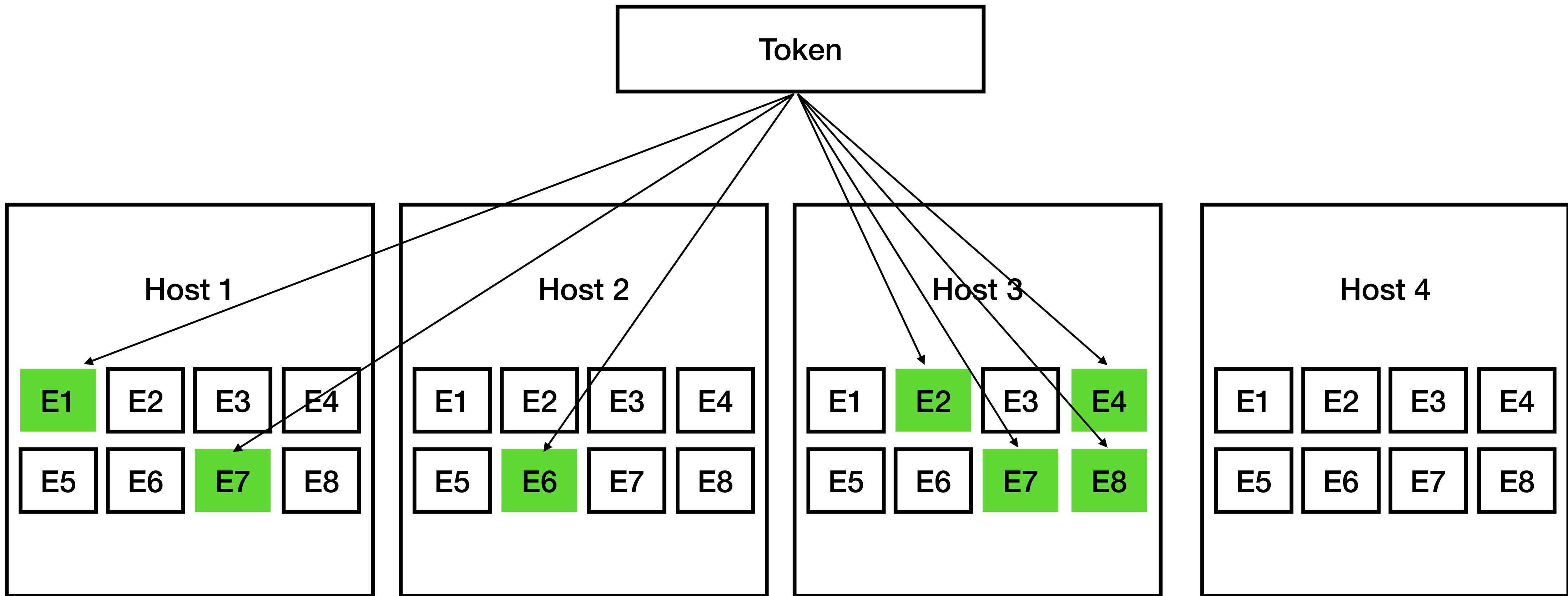
$$s'_{i,t} = \frac{s_{i,t}}{\sum_{j=1}^{N_r} s_{j,t}},$$

$$P_i = \frac{1}{T} \sum_{t=1}^T s'_{i,t},$$

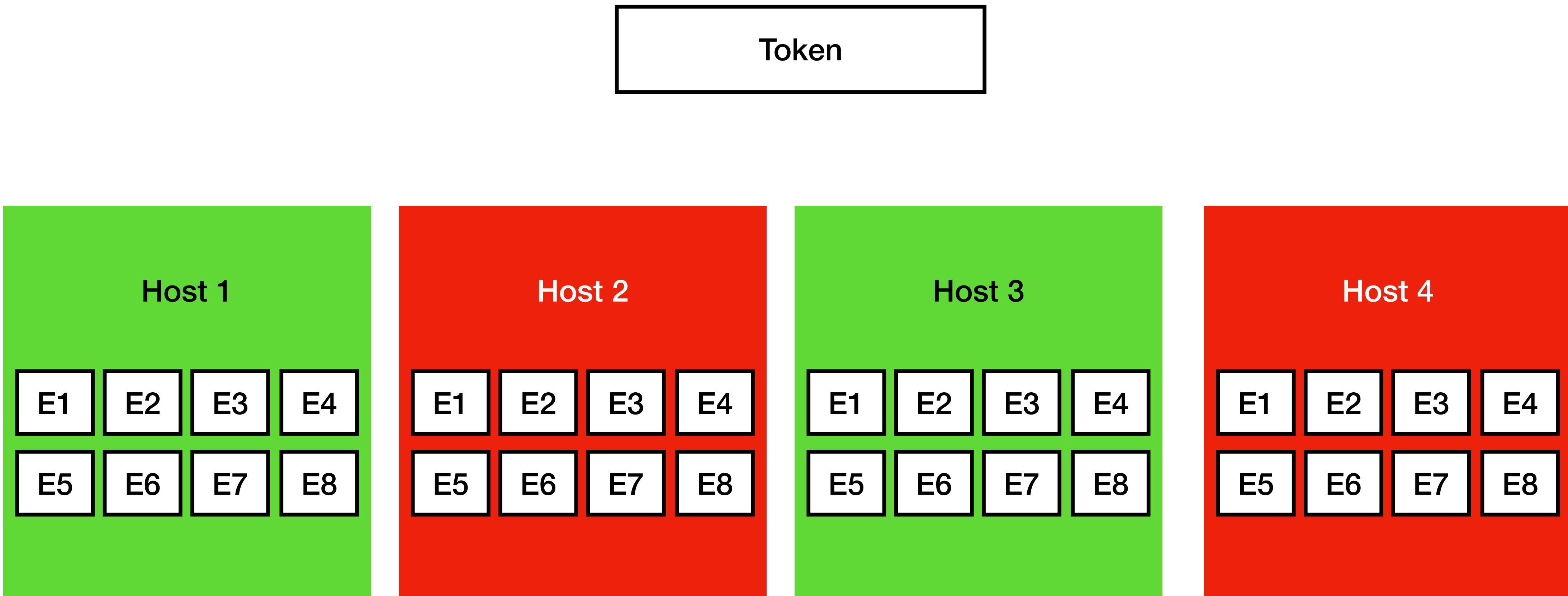
Иерархический роутинг



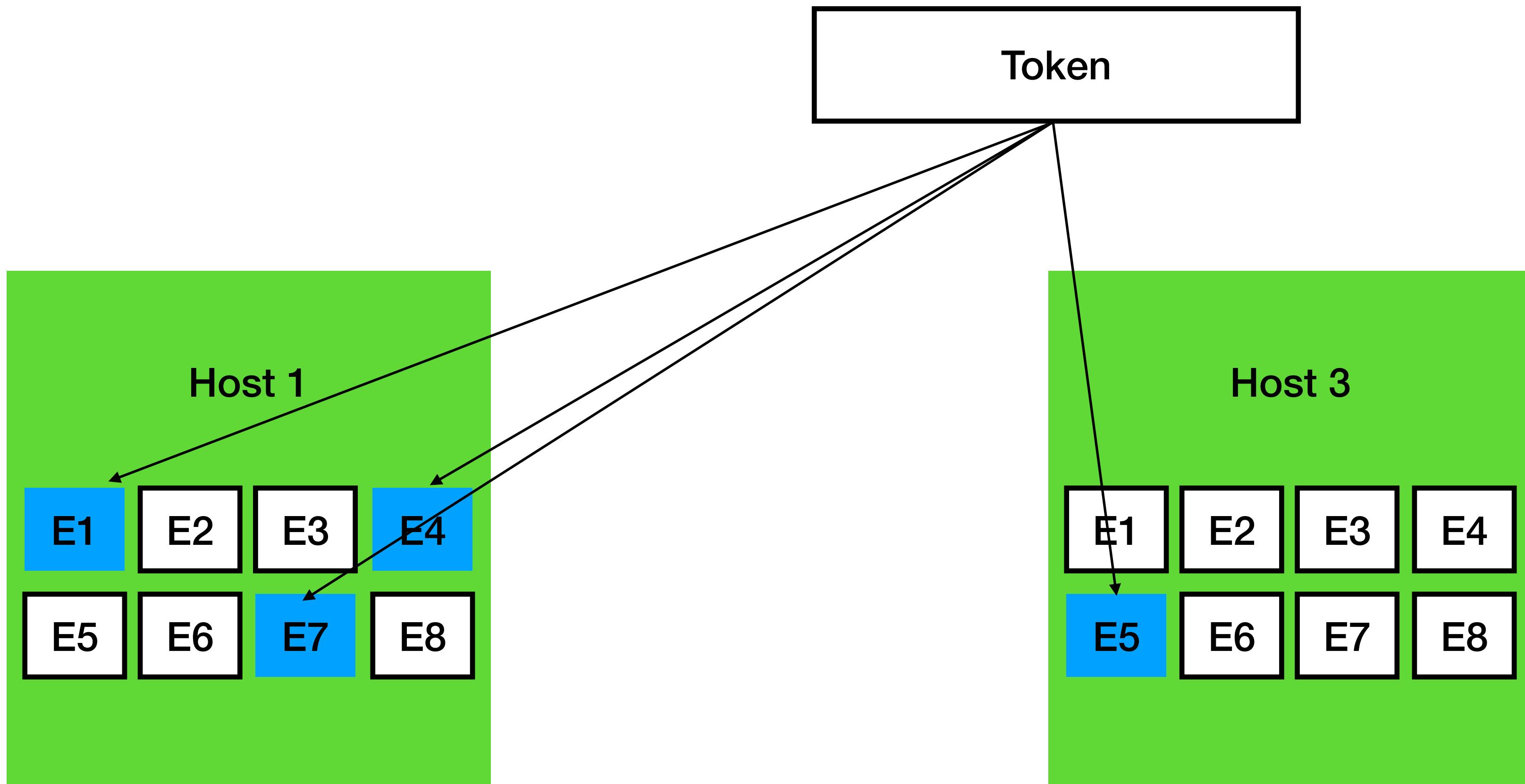
Иерархический роутинг



Иерархический роутинг



Иерархический роутинг



Ablation

Benchmark (Metric)	# Shots	Small MoE		Large MoE	
		Aux-Loss-Based	Aux-Loss-Free	Aux-Loss-Based	Aux-Loss-Free
# Activated Params	-	2.4B	2.4B	20.9B	20.9B
# Total Params	-	15.7B	15.7B	228.7B	228.7B
# Training Tokens	-	1.33T	1.33T	578B	578B
Pile-test (BPB)	-	0.727	0.724	0.656	0.652
BBH (EM)	3-shot	37.3	39.3	66.7	67.9
MMLU (EM)	5-shot	51.0	51.8	68.3	67.2
DROP (F1)	1-shot	38.1	39.0	67.1	67.1
TriviaQA (EM)	5-shot	58.3	58.5	66.7	67.7
NaturalQuestions (EM)	5-shot	23.2	23.4	27.1	28.1
HumanEval (Pass@1)	0-shot	22.0	22.6	40.2	46.3
MBPP (Pass@1)	3-shot	36.6	35.8	59.2	61.2
GSM8K (EM)	8-shot	27.1	29.6	70.7	74.5
MATH (EM)	4-shot	10.9	11.1	37.2	39.6

Table 5 | Ablation results for the auxiliary-loss-free balancing strategy. Compared with the purely auxiliary-loss-based method, the auxiliary-loss-free strategy consistently achieves better model performance on most of the evaluation benchmarks.

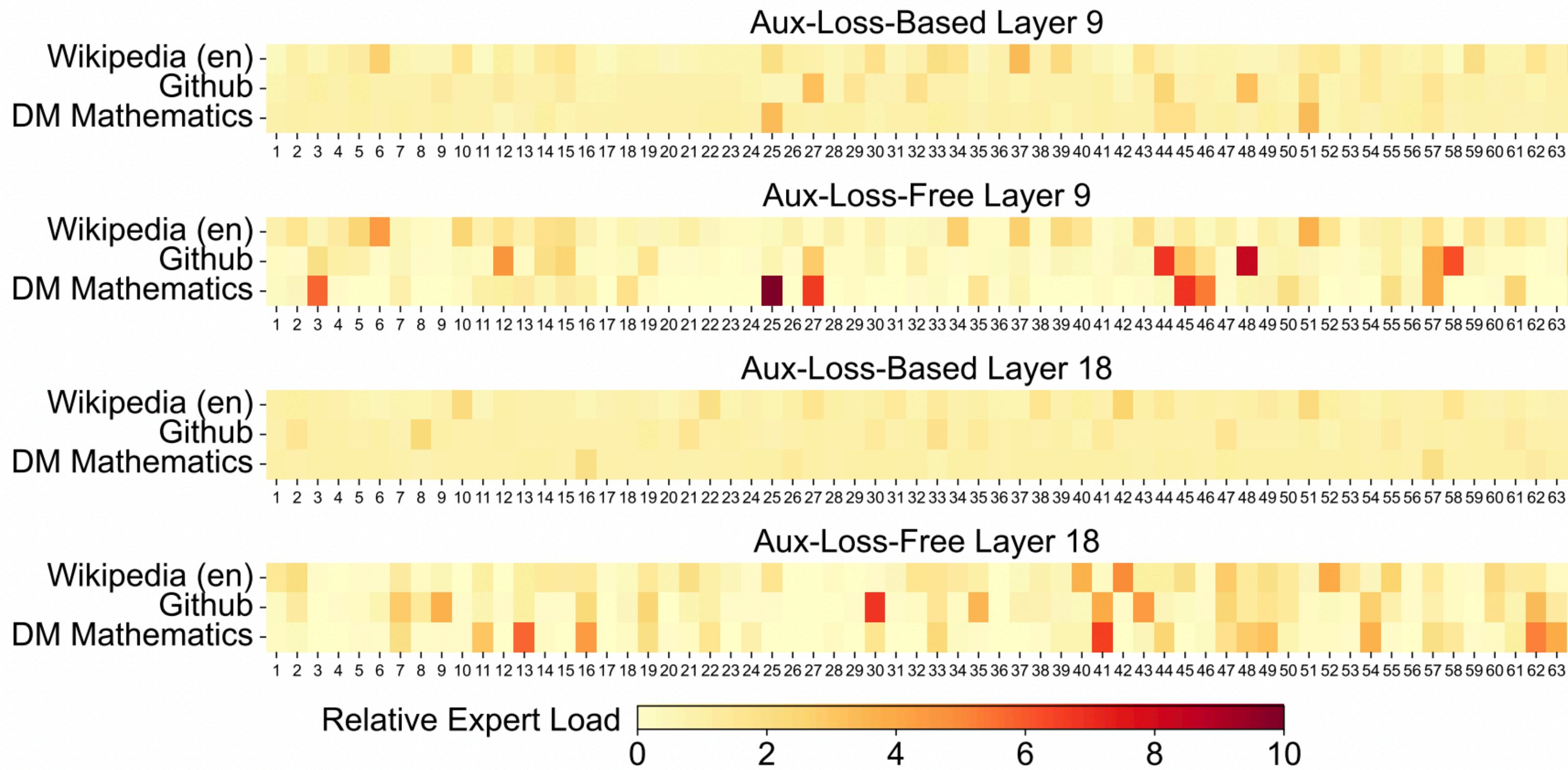


Figure 9 | Expert load of auxiliary-loss-free and auxiliary-loss-based models on three domains in the Pile test set. The auxiliary-loss-free model shows greater expert specialization patterns than the auxiliary-loss-based one. The relative expert load denotes the ratio between the actual expert load and the theoretically balanced expert load. Due to space constraints, we only present the results of two layers as an example, with the results of all layers provided in Appendix C.

Оптимизации экспертного параллелизма

Суть задачи

- Есть EP/8 хостов по 8 GPU, |E|/EP экспертов на каждой
- Каждая GPU работает со своим куском данных (data parallelism)
- Каждый токен (всего $EP * batch_size * seq_len$) выбирает своих 8 экспертов и летит к ним.
- После вычисления экспертов результаты для токенов летят на родительские GPU для суммирования.
- Вводные DeepSeek v3:
 - 8 экспертов на токен
 - 256 экспертов
 - $EP = 64$ (8 хостов)
 - Каждый токен выбирает не больше 4 хостов за раз

Варианты стратегий

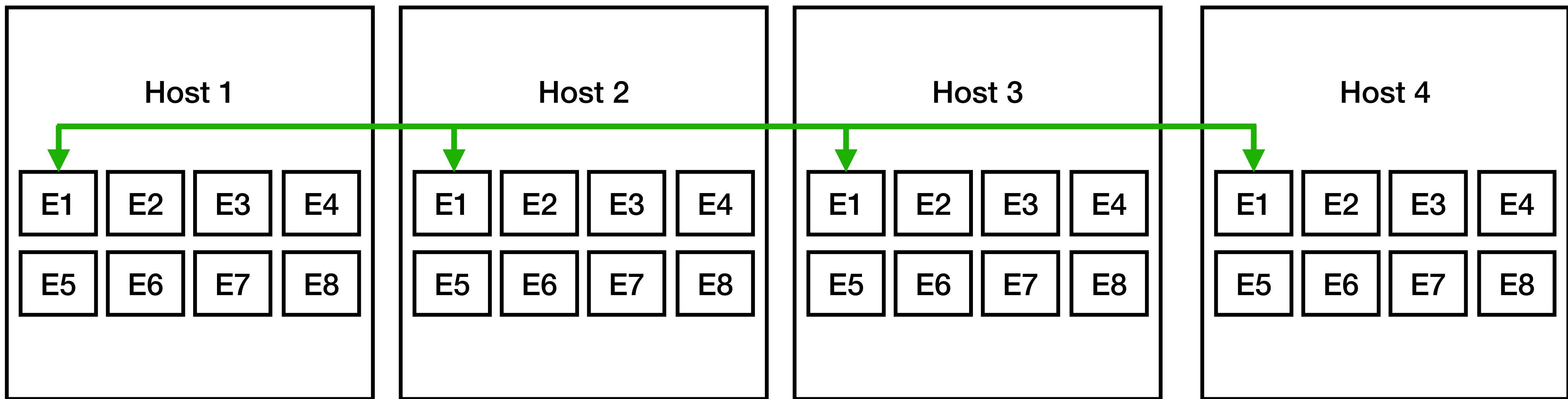
- All gather
 - Объем коммуникаций: $|EP| * \text{batch_size} * \text{seq_len} * \text{hidden} = 1.9$ млрд на хост. ($\text{hidden}=7168$, $\text{seq_len}=4096$, $\text{batch_size}=1$)

Варианты стратегий

- All gather - собираем все токены на всех GPU, потом распределяем.
 - Объем коммуникаций: $|EP| * \text{batch_size} * \text{seq_len} * \text{hidden} = 1.9$ млрд на хост. ($\text{hidden}=7168$, $\text{seq_len}=4096$, $\text{batch_size}=1$). Возьмем этот объем коммуникаций за C.
- Dispatch - отправляем с GPU на GPU токен в случае, если токен выбрал этого эксперта
 - Среднее количество GPU, которые выберет токен на хосте: $0.96*C$.
 - Объем коммуникаций в 0.96 раз меньше, но реализовать оптимально такое сложно

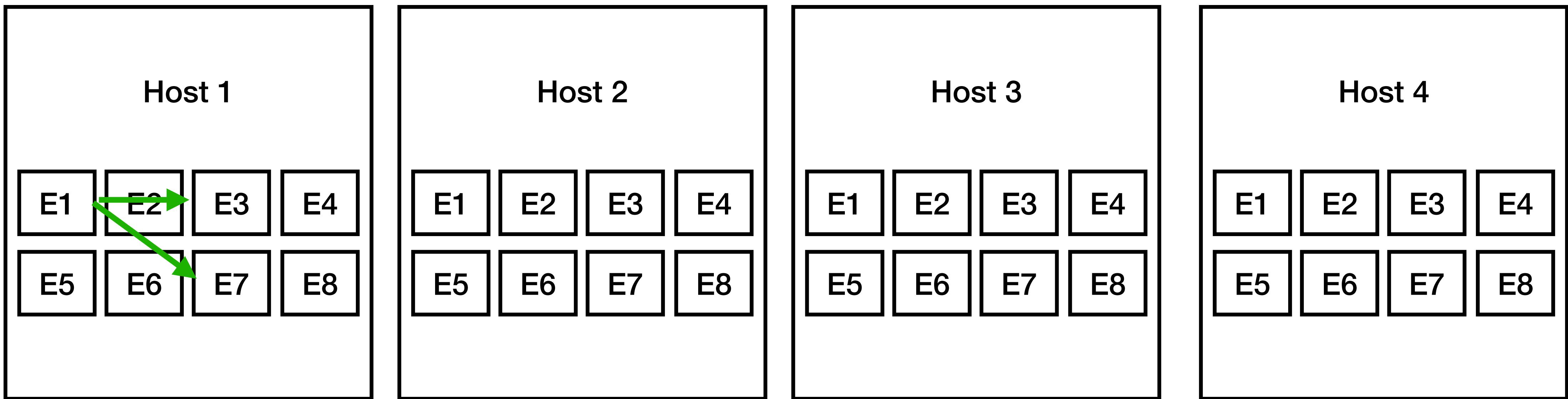
Варианты стратегий

- DeepSeek v3 dispatch:
 - Сначала пересылаются токены между GPU одного ранка.



Варианты стратегий

- DeepSeek v3 dispatch:
 - Сначала пересылаются токены между GPU одного ранка.
 - Затем: внутри хоста.



Варианты стратегий

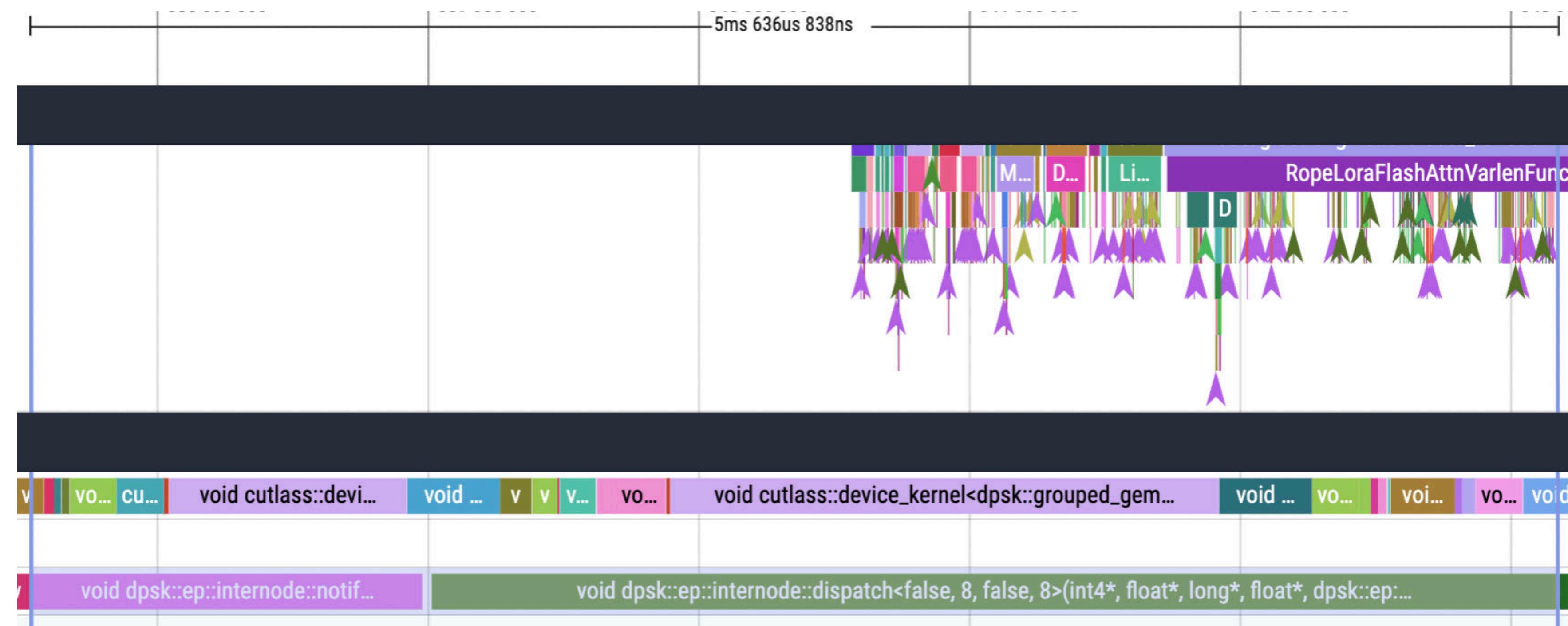
- DeepSeek v3 dispatch:
 - Сначала пересылаются токены между GPU одного ранка.
 - Затем: внутри хоста.
 - Мы гарантированно не пересылаем токены дважды, поэтому объем коммуникаций меньше: 0.67^*C .

Варианты стратегий

- DeepSeek v3 dispatch:
 - Сначала пересылаются токены между GPU одного ранка.
 - Затем: внутри хоста.
 - Мы гарантированно не пересылаем токены дважды, поэтому объем коммуникаций меньше: 0.67^*C .
 - Но: мы рассылаем каждый токен только на половину хостов, поэтому объем коммуникаций примерно равен 0.4^*C .
 - Бонус: такой подход работает в схеме с порезанным NVLink.

Странности

- Суммарное время коммуникаций в FP8 в итоге должно составлять $0.4 \cdot 64 \cdot 4096 \cdot 7168 / 400e9$ - 2ms на dispatch.
 - По выложенным профилям время коммуникации: 5.5ms.

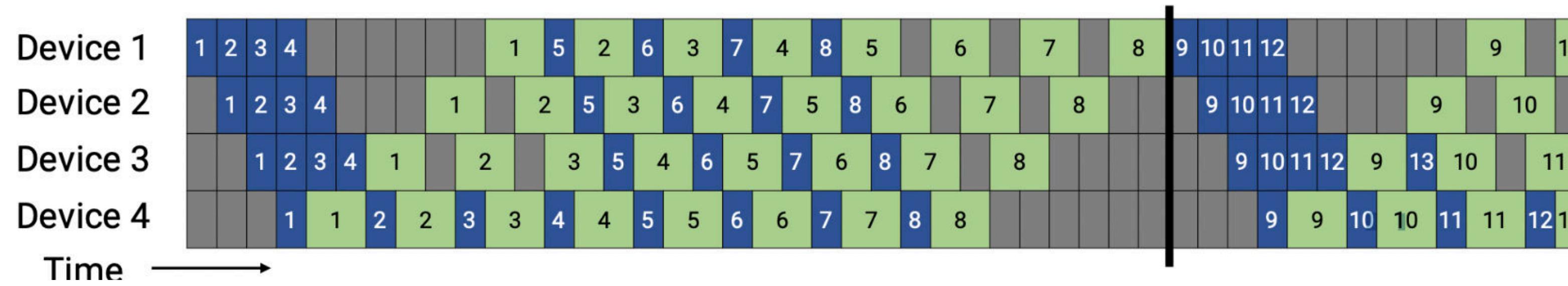


Код

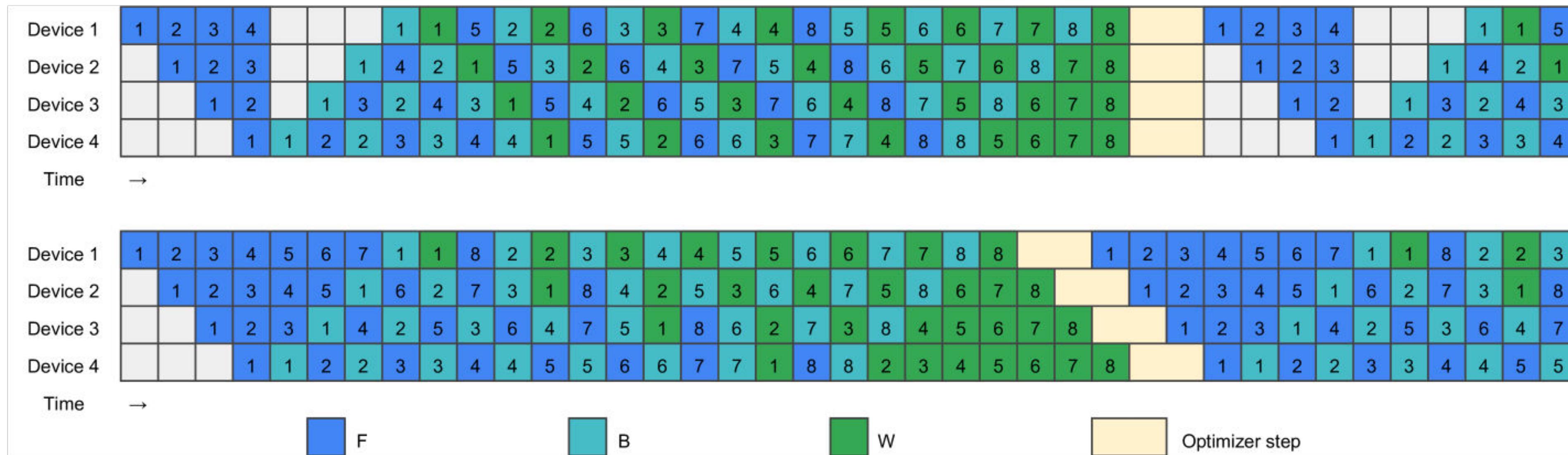
- DeepSeek выложил свой вариант экспертного параллелизма.
- Ссылка: <https://github.com/deepseek-ai/DeepEP>

DualPipe

Pipeline parallelism



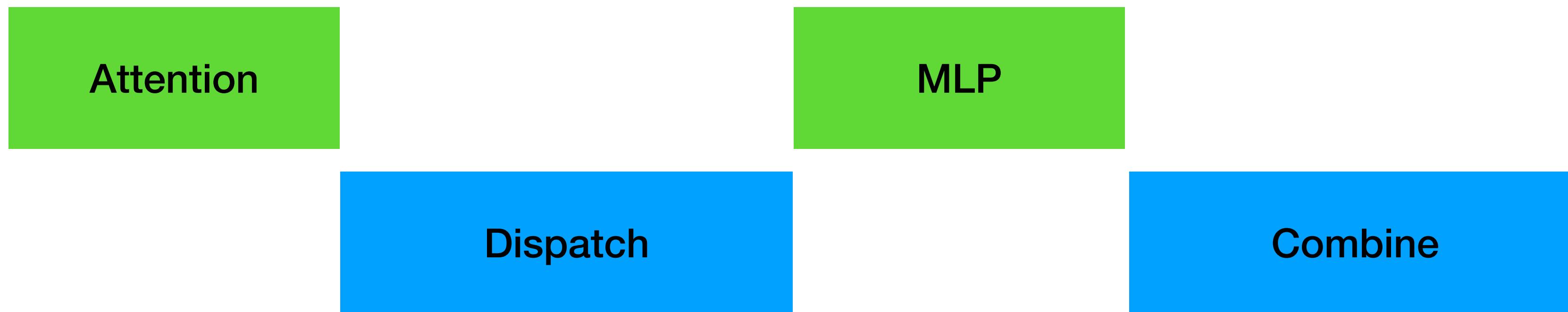
Zero Bubble



Forward B DeepSeek v3

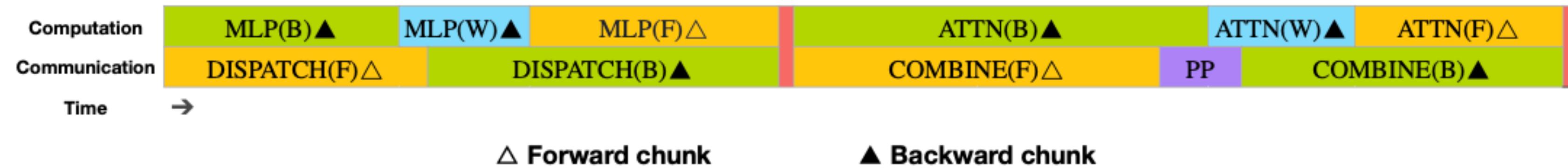


Forward в DeepSeek v3

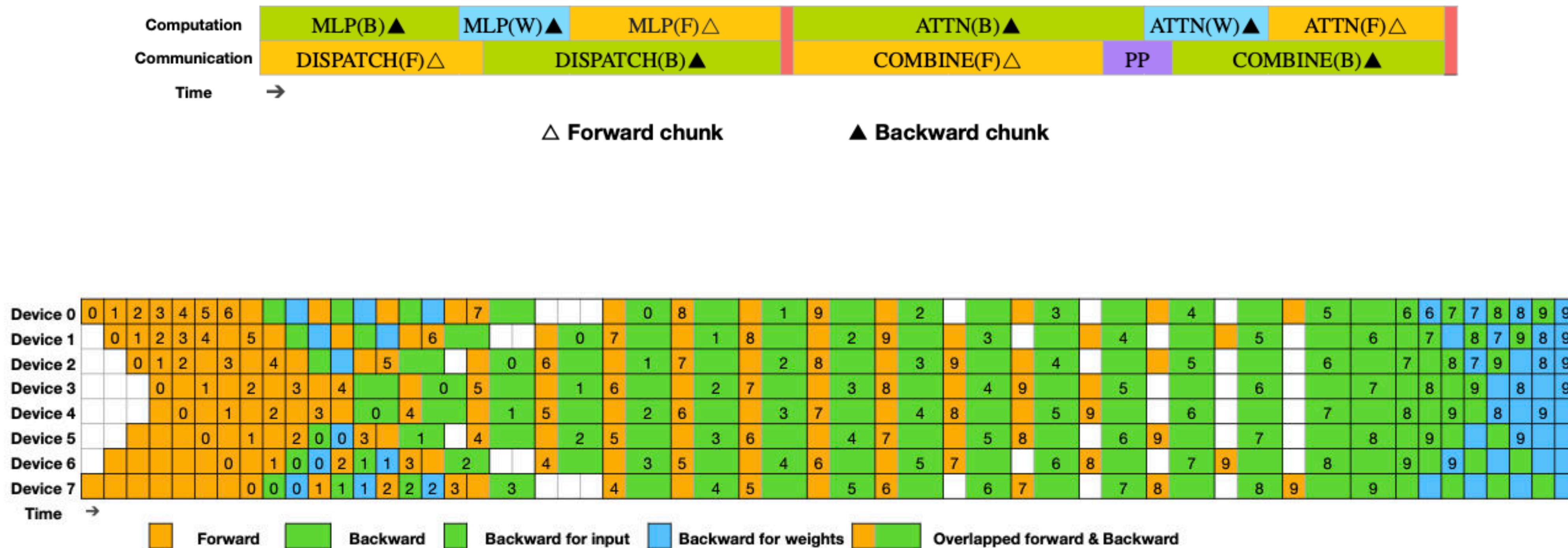


- Наивная реализация РР будет неэффективной из-за микропузьрей внутри Forward и Backward!

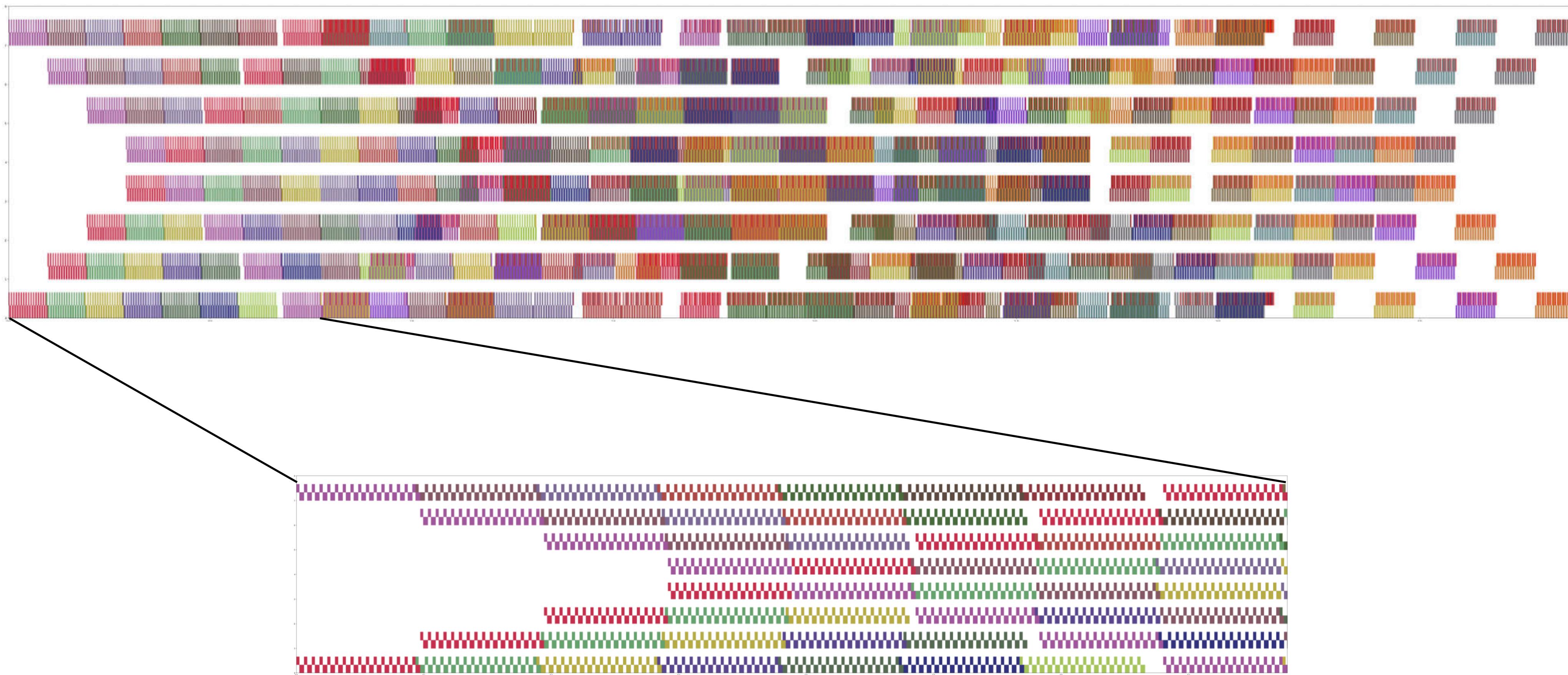
Перекрытие Forward и Backward



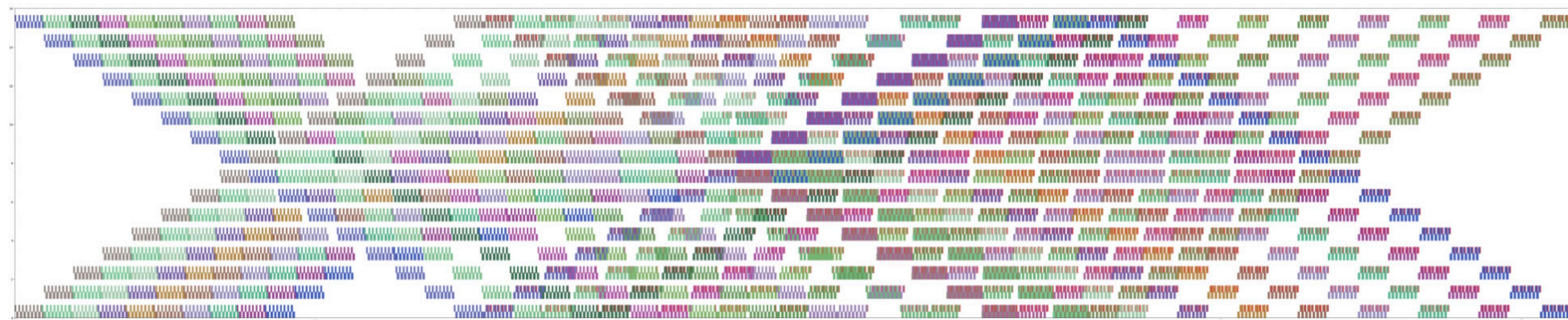
Перекрытие Forward и Backward



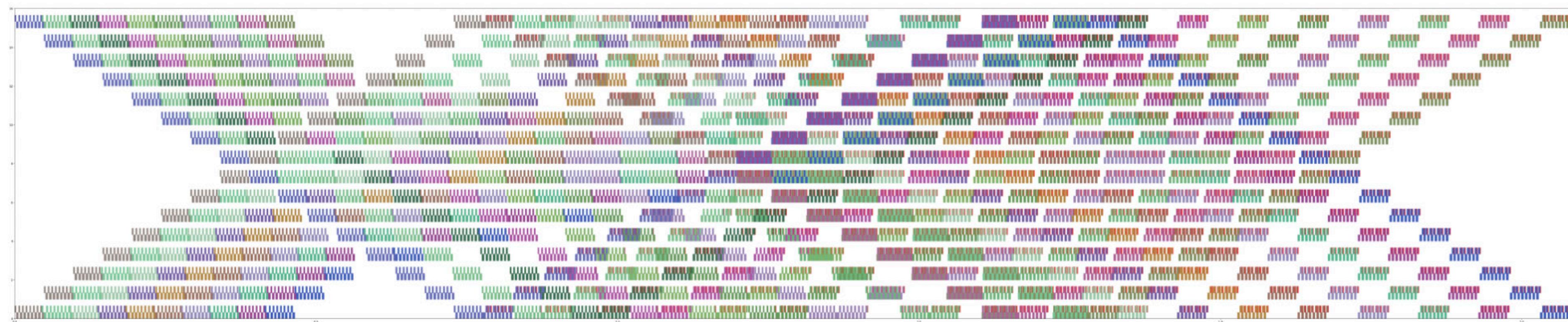
Реальная картина ($PP = 8$)



Реальная картина ($PP = 16$)



Реальная картина (PP = 16)



Утилизация: $\leq 42\% = ($

Код

- DeepSeek выложил свой вариант DualPipe.
- Ссылка: <https://github.com/deepseek-ai/DualPipe>

FP8 квантизация

FP8 квантизация. Что хотим

- Ускорение GEMM до x2
- Ускорение коммуникаций x2
- Уменьшение памяти на активации до x2

FP8 квантизация

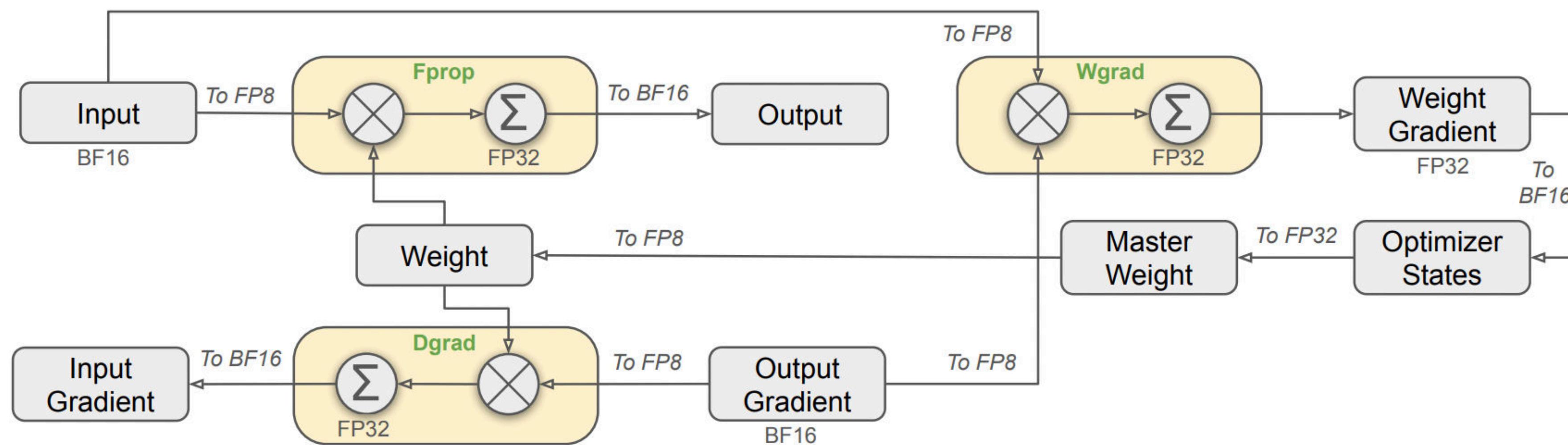


Figure 6 | The overall mixed precision framework with FP8 data format. For clarification, only the **Linear** operator is illustrated.

FP8 квантизация

```
scale = weight.abs().max()
```

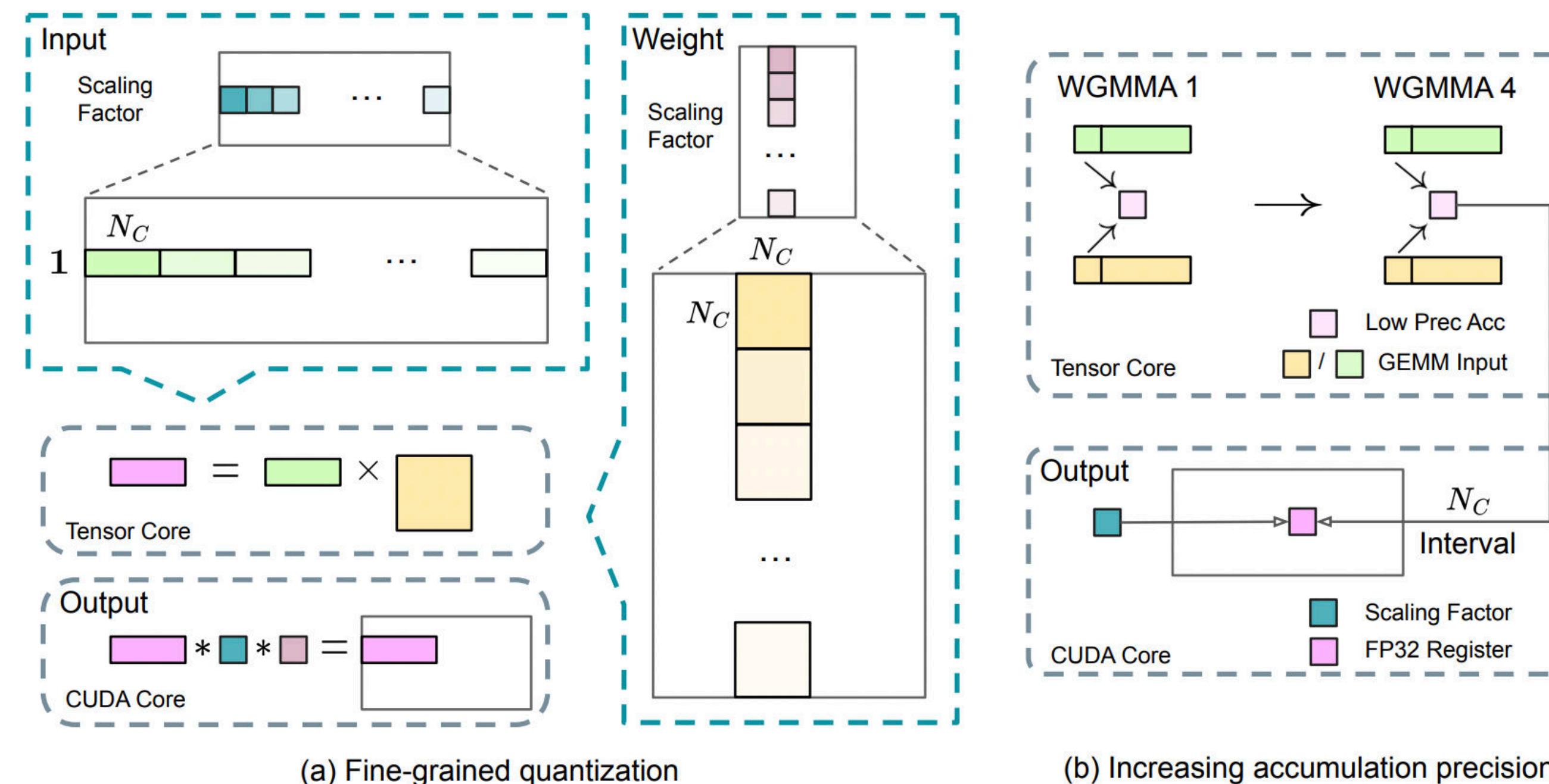
```
fp_8_weight = cast_to_fp8(weight / scale)
```

Проблемы

- FP8 квантизация может приводить к потере качества из-за аутлаеров.

Решение: блочная квантизация

- Делаем квантизацию для блоков 128x128 для весов
- Делаем квантизацию для блоков 1x128 для активаций



Проблемы с точностью

Increasing Accumulation Precision. Low-precision GEMM operations often suffer from underflow issues, and their accuracy largely depends on high-precision accumulation, which is commonly performed in an FP32 precision (Kalamkar et al., 2019; Narang et al., 2017). However, we observe that the accumulation precision of FP8 GEMM on NVIDIA H800 GPUs is limited to retaining around 14 bits, which is significantly lower than FP32 accumulation precision. This problem will become more pronounced when the inner dimension K is large (Wortsman et al., 2023), a typical scenario in large-scale model training where the batch size and model width are increased. Taking GEMM operations of two random matrices with $K = 4096$ for example, in our preliminary test, the limited accumulation precision in Tensor Cores results in a maximum relative error of nearly 2%. Despite these problems, the limited accumulation precision is still the default option in a few FP8 frameworks (NVIDIA, 2024b), severely constraining the training accuracy.

Проблемы с точностью

In order to address this issue, we adopt the strategy of promotion to CUDA Cores for higher precision (Thakkar et al., 2023). The process is illustrated in Figure 7 (b). To be specific, during MMA (Matrix Multiply-Accumulate) execution on Tensor Cores, intermediate results are accumulated using the limited bit width. Once an interval of N_C is reached, these partial results will be copied to FP32 registers on CUDA Cores, where full-precision FP32 accumulation is performed. As mentioned before, our fine-grained quantization applies per-group scaling factors along the inner dimension K. These scaling factors can be efficiently multiplied on the CUDA Cores as the dequantization process with minimal additional computational cost.

Другие особенности

- Мантисса важнее: все операции производятся в Е4М3
- Повышенная точность для состояний оптимизатора, мастер-весов, RMSNorm, Router, etc.
- AdamW в bf16
- fp8 активации в MLP
- MoE dispatch-all-to-all коммуникации в fp8

Ablation

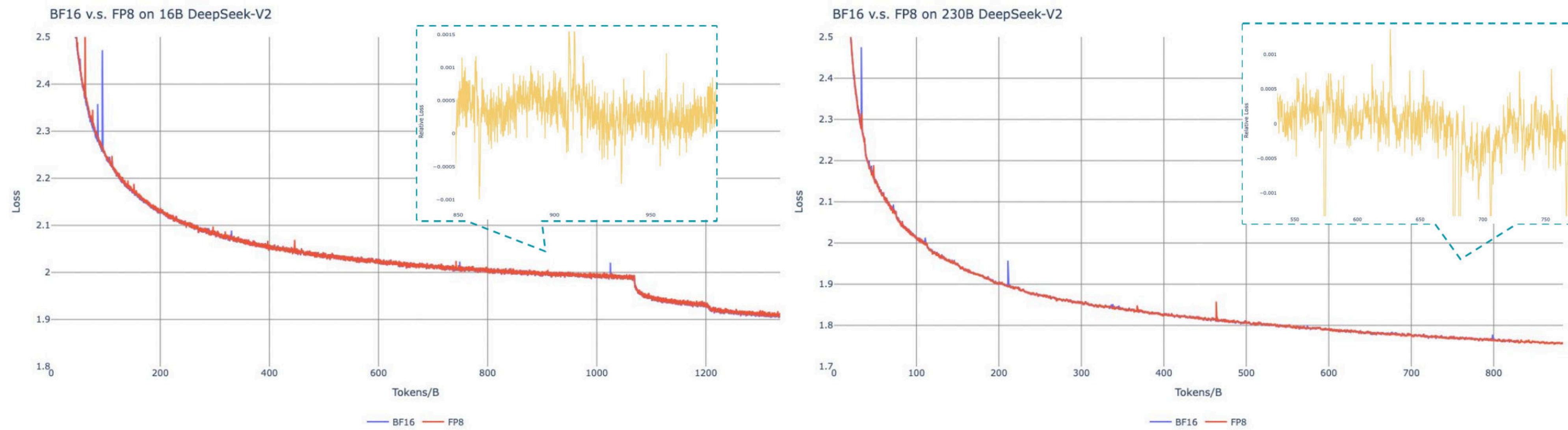


Figure 10 | Loss curves comparison between BF16 and FP8 training. Results are smoothed by Exponential Moving Average (EMA) with a coefficient of 0.9.

Код

- DeepSeek выложил свою реализацию блочной квантизации: <https://github.com/deepseek-ai/DeepGEMM>

4096	7168	16384	1358 TFLOPS	343 GB/s	1.2x
4096	4096	7168	1304 TFLOPS	500 GB/s	1.1x
4096	7168	2048	1025 TFLOPS	697 GB/s	1.1x

Grouped GEMMs for MoE models (contiguous layout)						
#Groups	M per group	N	K	Computation	Memory bandwidth	Speedup
4	8192	4096	7168	1297 TFLOPS	418 GB/s	1.2x
4	8192	7168	2048	1099 TFLOPS	681 GB/s	1.2x
8	4096	4096	7168	1288 TFLOPS	494 GB/s	1.2x
8	4096	7168	2048	1093 TFLOPS	743 GB/s	1.1x

Итоги

Итоги

- DeepSeek придумали способ, как лучше обучать гигантские MoE модели с точки зрения качества
- Тем не менее, их подход