HOW TO TRAIN YOUR TINY MOE

Or: how to optimize your training loop as far as you can

MOTIVATION

- With the rise of small models, research at home should be a thing

MOTIVATION

- With the rise of small models, research at home should be a thing
- Personally, pretraining is something I’ve always wanted to do

MOTIVATION

- With the rise of small models, research at home should be a thing
- Personally, pretraining is something I've always wanted to do
- I have a genuine love for making things faster

HOW FAST IS FAST ENOUGH?

HOW FAST IS FAST ENOUGH?

- MFU: How well your model utilizes your GPU
- MFU at home (~30-35% is doing VERY good)
- 4x RTX 6000 Blackwell Max-Q's
- No NVLink/fancy tricks

APPENDIX A: Blackwell GB202 GPU

Table 4. RTX A6000 vs RTX 6000 Ada vs RTX PRO 6000 Blackwell Full Specs

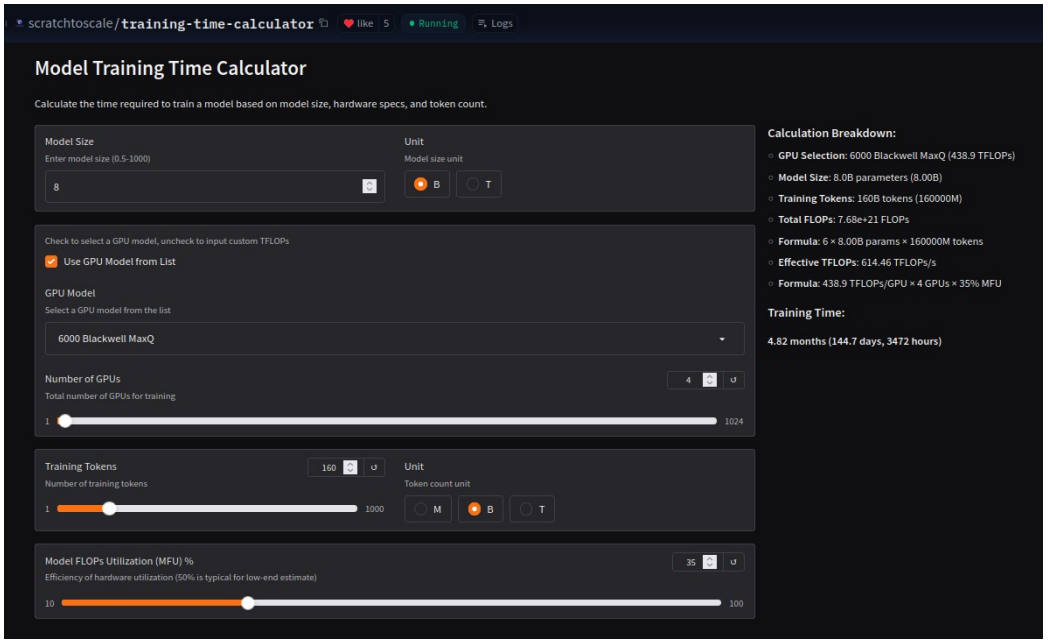
Graphics Card	RTX A6000	RTX 6000 Ada Generation	RTX PRO 6000 Blackwell Max-Q Workstation Edition	RTX PRO 6000 Blackwell Workstation Edition
GPU Codename	GA102	AD102	GB202	GB202
GPU Architecture	NVIDIA Ampere	NVIDIA Ada Lovelace	NVIDIA Blackwell	NVIDIA Blackwell
GPCs	7	12	12	12
TPCs	42	71	94	94
SMs	84	142	188	188
CUDA Cores / SM	128	128	128	128
CUDA Cores / GPU	10752	18176	24064	24064
Tensor Cores / SM	4 (3rd Gen)	4 (4th Gen)	4 (5th Gen)	4 (5th Gen)
Tensor Cores / GPU	336	568 (4th Gen)	752 (5th Gen)	752 (5th Gen)
Peak FP16 Tensor TFLOPS with FP32 Accumulate [1]	154.8/309.6 [2]	364/728 [2]	438.9/877.9 [2]	503.8/1007.6 [2]
Peak BF16 Tensor TFLOPS with FP32 Accumulate [1]	154.8/309.6 [2]	364/728 [2]	438.9/877.9 [2]	503.8/1007.6 [2]

HOW FAST IS FAST ENOUGH?

- ~12 hours total (to Chinchilla)
  - Chinchilla == 20 * XB params
  - Fast enough to test data iterations
  - 2x experiments per day
  - Vastly limits the size of the model I can use
- What models can I use?

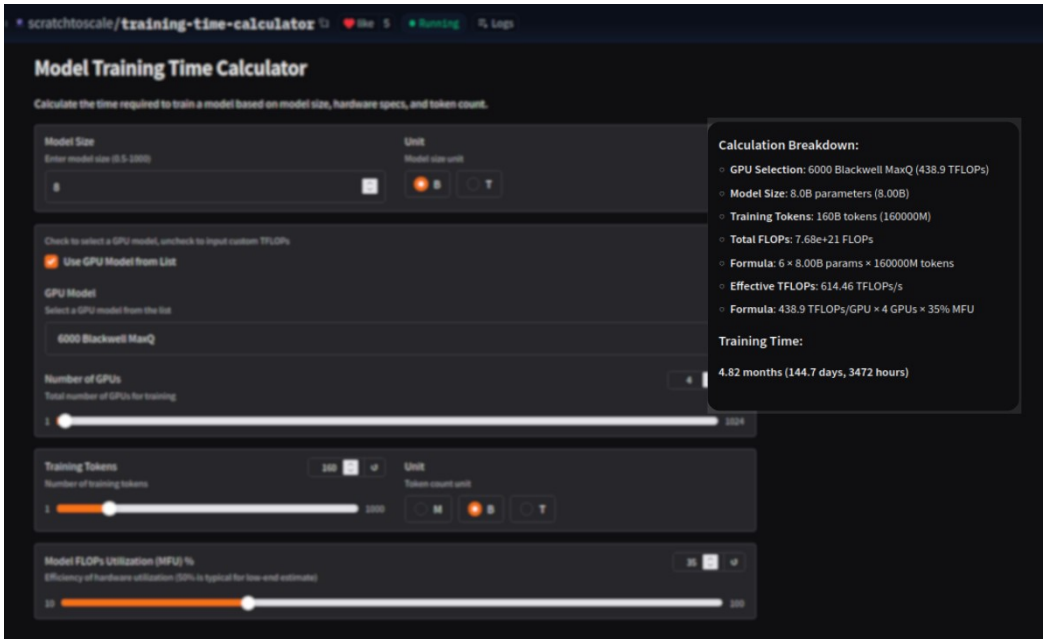
HOW FAST IS FAST ENOUGH?

The Depression Calculator™



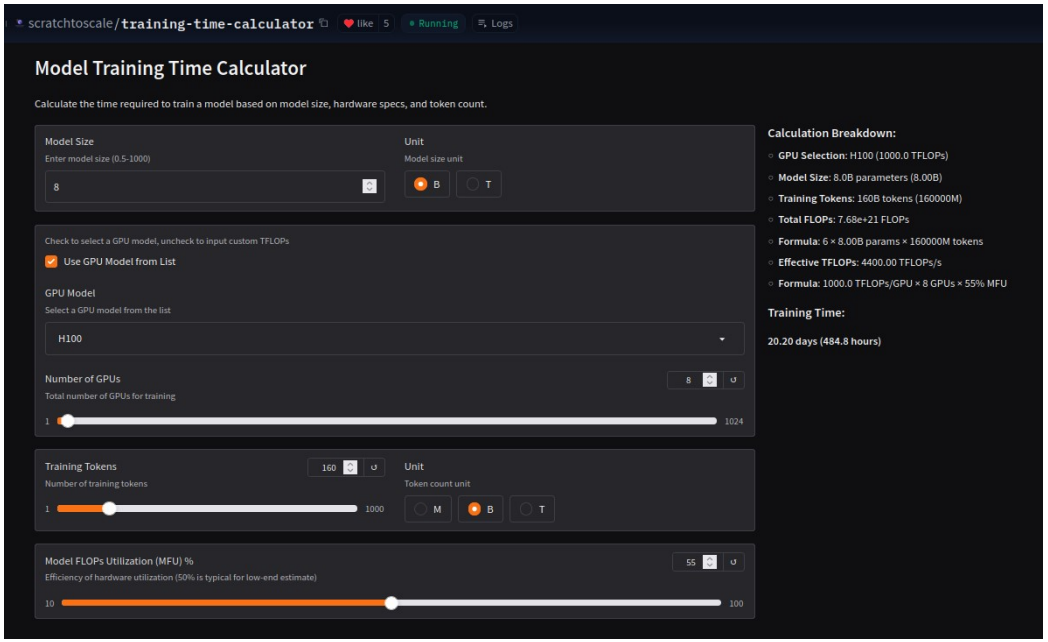
HOW FAST IS FAST ENOUGH?

The Depression Calculator™



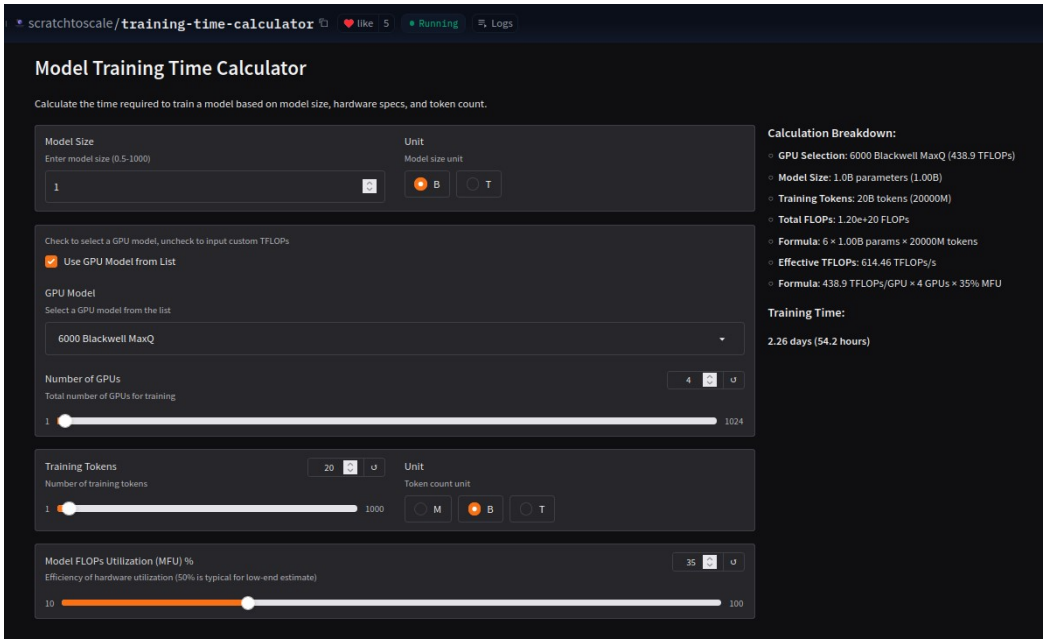
HOW FAST IS FAST ENOUGH?

The Depression Calculator™



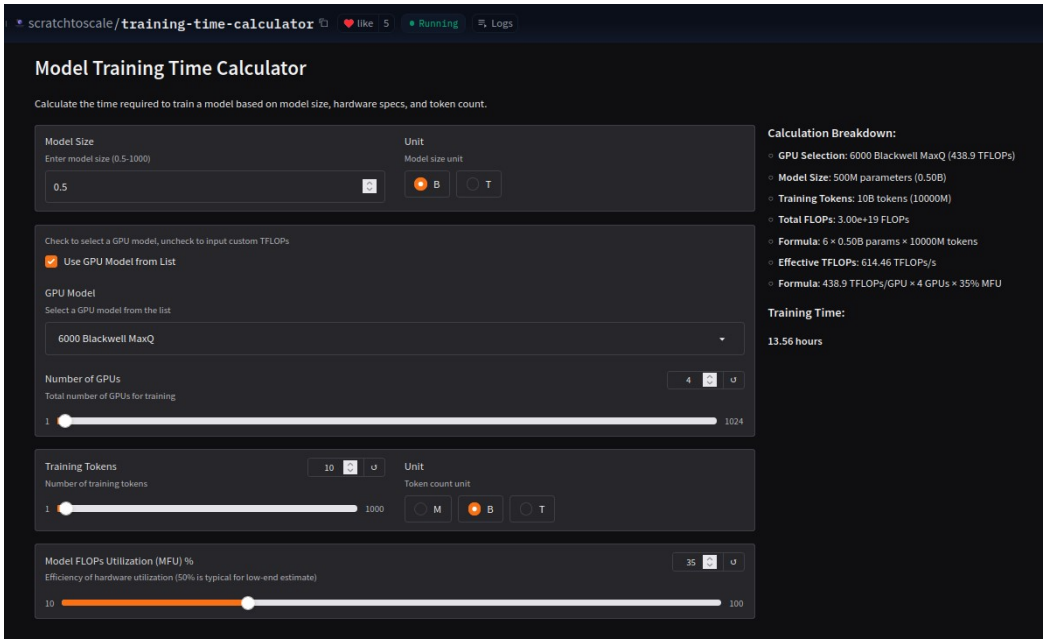
HOW FAST IS FAST ENOUGH?

The Depression Calculator™



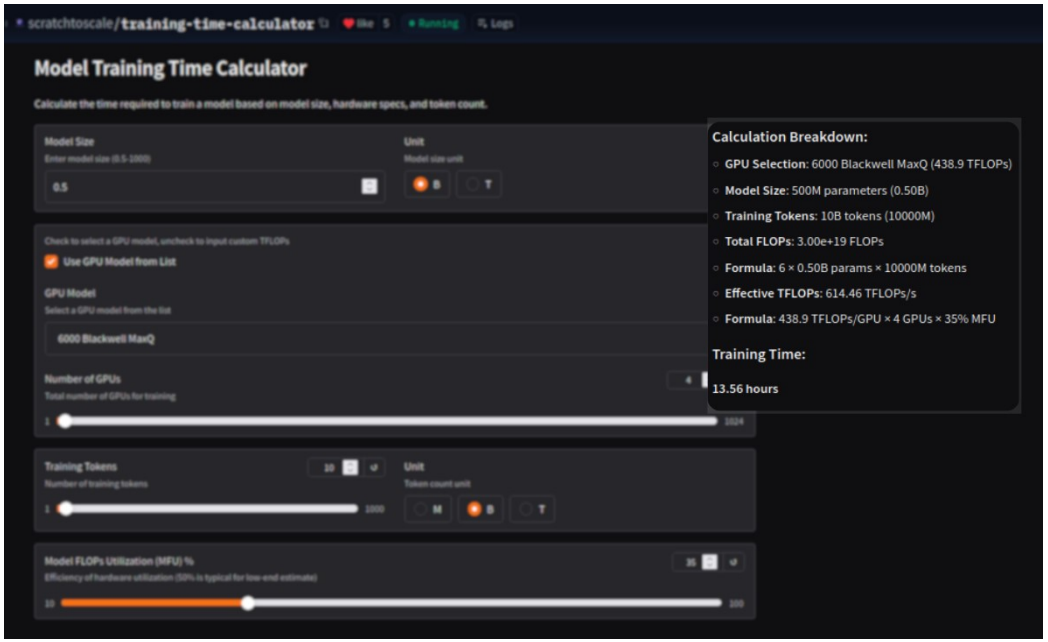
HOW FAST IS FAST ENOUGH?

The Depression Calculator™



HOW FAST IS FAST ENOUGH?

The Depression Calculator™



THE LONG ROAD TO 13 HOURS

THE MODEL

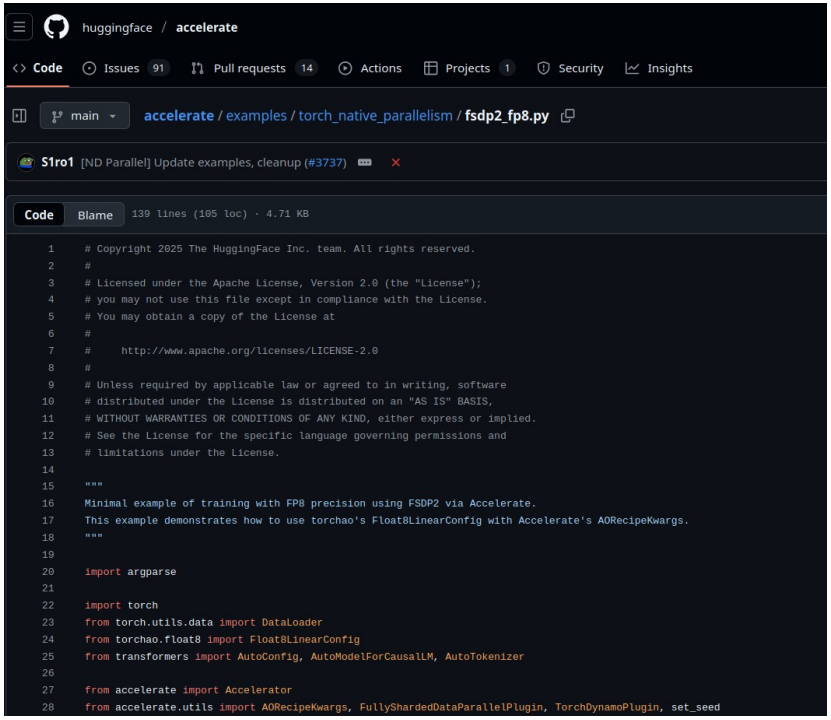
- Qwen3-MoE-500M-A330M
- *Most* of that 330M comes from embeddings
- At a vocab of only 32k it'd be considered Qwen3-MoE-500M-A120M

```yaml
model_type: qwen3_moe
architectures: ["Qwen3MoeForCausalLM"]

vocab_size: 151936
hidden_size: 896
num_hidden_layers: 12
num_attention_heads: 16
num_key_value_heads: 16
hidden_act: silu

# === MoE ===
num_experts: 16
num_experts_per_tok: 2
moe_intermediate_size: 512

# === Dense MLP baseline ===
intermediate_size: 4096
```

ESTABLISH A BASELINE ON THE SMALLEST NODE POSSIBLE

- Focus on saving $$$
- Focus on saving **time**
- Ensure that small variations work then scale

ESTABLISH A BASELINE ON THE SMALLEST NODE POSSIBLE



ESTABLISH A BASELINE ON THE SMALLEST NODE POSSIBLE

Raw BF16 using a core script:

- Batch size of 12
- Iterates through 200 batches
- Sequence length of 4096
- 9.8M tokens

Results:

- 44,942 tok/s
- Time to 10B tokens: **61.80 hours**

IS MAXIMUM BATCH SIZE WORTH IT?

- Batch size of 8
- Sequence length of 4096
- 6.5M tokens

Results:

- ~~44,942~~ 45,005 tok/s
- Time to 10B tokens: ~~61.80 hours~~ **61.72 hrs**

IS MAXIMUM BATCH SIZE WORTH IT?

Configuration

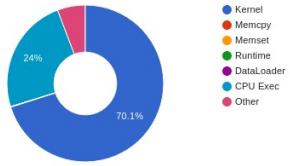
Number of Worker(s)	2
Device Type	GPU

GPU Summary

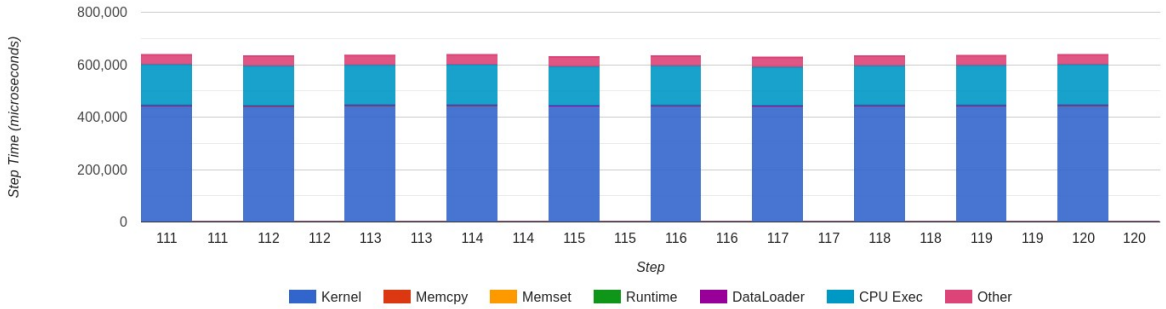
GPU 0:	
Name	NVIDIA RTX PRO 6000 Blackwell Max-Q Workstation Edition
Memory	94.97 GB
Compute Capability	12.0
GPU Utilization	70.15 %
Est. SM Efficiency	67.92 %
Est. Achieved Occupancy	29.11 %

Execution Summary

Category	Time Duration (us)	Percentage (%)
Average Step Time	316,796	100
Kernel	222,057	70.09
Memcpy	118	0.04
Memset	204	0.06
Runtime	0	0
DataLoader	0	0
CPU Exec	76,103	24.02
Other	18,313	5.78



Step Time Breakdown



LET'S TRY A DIFFERENT ATTENTION MECHANISM

```
    model = AutoModelForCausalLM.from_config(
        AutoConfig.from_pretrained("./config.json", use_cache=False),
        torch_dtype=torch.bfloat16,
        attn_implementation="flex_attention"
    )
```

Results:

- ~~45,005~~ 59,037 tok/s
- Time to 10B tokens: ~~61.80 hours~~ **47.5 hrs**

COULD IT BE THE DATA?

Configuration

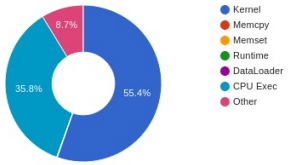
Number of Worker(s)	1
Device Type	GPU

GPU Summary

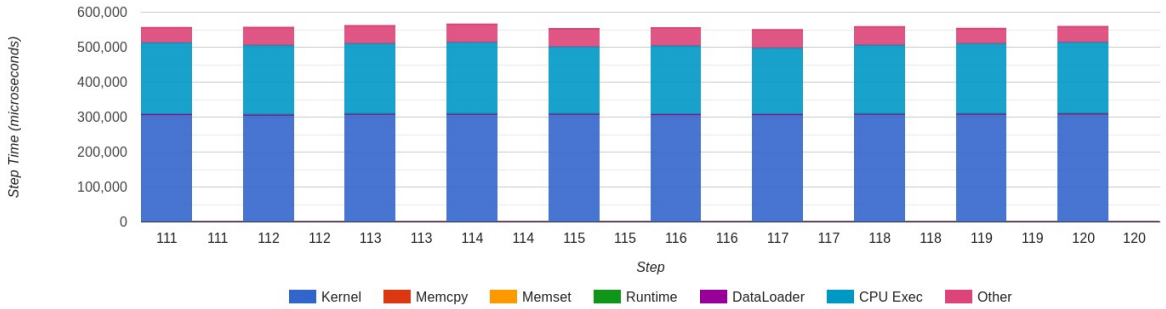
GPU 0:	
Name	NVIDIA RTX PRO 6000 Blackwell Max-Q Workstation Edition
Memory	94.97 GB
Compute Capability	12.0
GPU Utilization	55.42 %
Est. SM Efficiency	52.92 %
Est. Achieved Occupancy	32.69 %

Execution Summary

Category	Time Duration (us)	Percentage (%)
Average Step Time	278,247	100
Kernel	154,098	55.38
Memcpy	118	0.04
Memset	184	0.07
Runtime	0	0
DataLoader	0	0
CPU Exec	99,503	35.76
Other	24,345	8.75



Step Time Breakdown



SIMPLE TEST: PIN_MEMORY=TRUE

Configuration

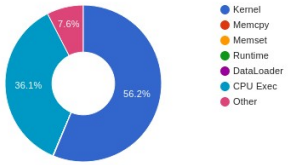
Number of Worker(s)	1
Device Type	GPU

GPU Summary

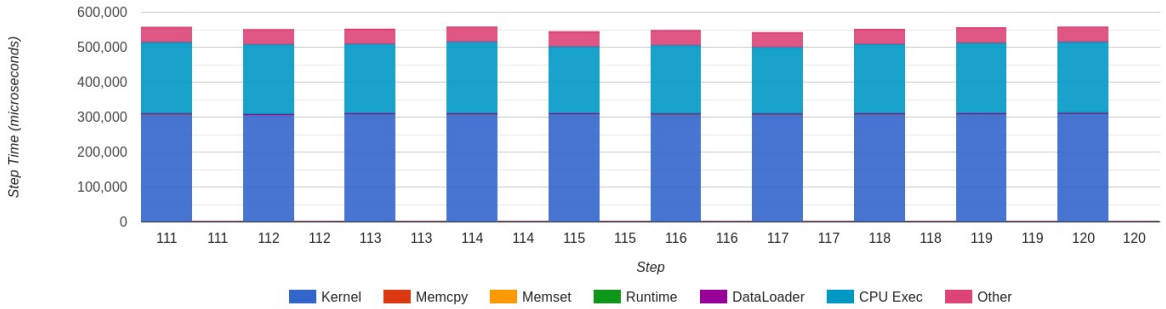
GPU 0:	
Name	NVIDIA RTX PRO 6000 Blackwell Max-Q Workstation Edition
Memory	94.97 GB
Compute Capability	12.0
GPU Utilization	56.32 %
Est. SM Efficiency	53.8 %
Est. Achieved Occupancy	32.61 %

Execution Summary

Category	Time Duration (us)	Percentage (%)
Average Step Time	275,333	100
Kernel	154,849	56.24
Memcpy	118	0.04
Memset	187	0.07
Runtime	0	0
DataLoader	0	0
CPU Exec	99,328	36.08
Other	20,851	7.57



Step Time Breakdown



WHAT IF WE MINIMIZE THE CPU WHATSOEVER?

Configuration

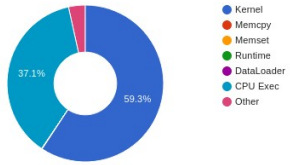
Number of Worker(s)	1
Device Type	GPU

GPU Summary

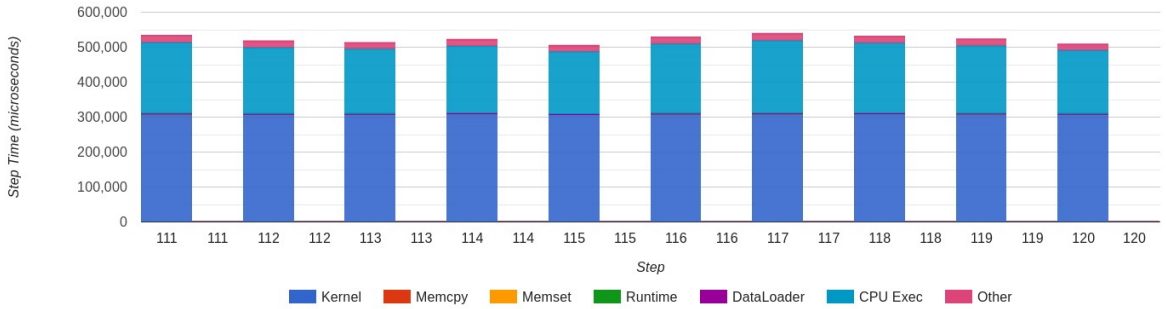
GPU 0:	
Name	NVIDIA RTX PRO 6000 Blackwell Max-Q Workstation Edition
Memory	94.97 GB
Compute Capability	12.0
GPU Utilization	59.16 %
Est. SM Efficiency	56.52 %
Est. Achieved Occupancy	32.55 %

Execution Summary

Category	Time Duration (us)	Percentage (%)
Average Step Time	260,801	100
Kernel	154,603	59.28
Memcpy	117	0.04
Memset	184	0.07
Runtime	0	0
DataLoader	0	0
CPU Exec	96,865	37.14
Other	9,031	3.46



Step Time Breakdown



WHAT IF WE MINIMIZE THE CPU WHATSOEVER?

Pin memory:

- ~~59,037~~ 59,207 tok/s
- Time to 10B tokens: ~~47.5 hours~~ **46.91 hrs**

WHAT IF WE MINIMIZE THE CPU WHATSOEVER?

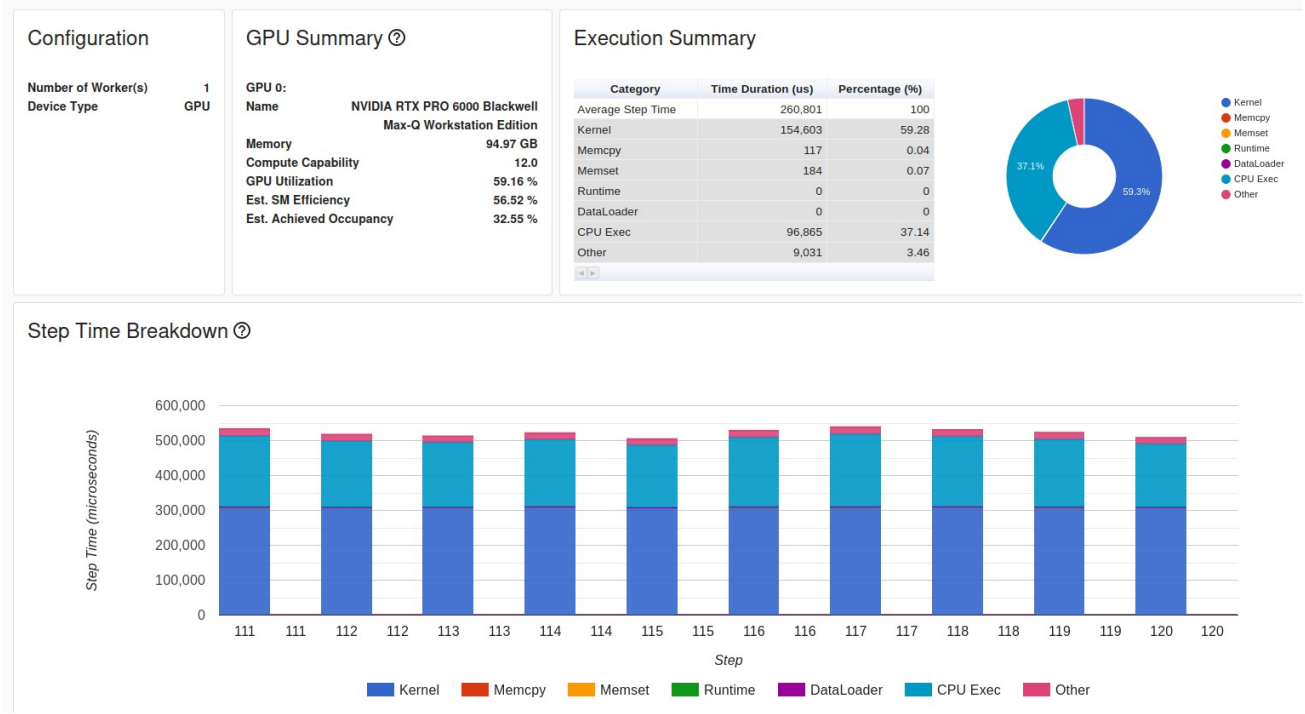
Pin memory:

- ~~59,037~~ 59,207 tok/s
- Time to 10B tokens: ~~47.5 hours~~ **46.91 hrs**

Pre-tokenize:

- ~~59,037~~ 62,840 tok/s
- Time to 10B tokens: ~~46.91 hours~~ **44.2 hrs**

LET'S PLAY WITH THE DATA MORE: CUDA BUFFERS (BEFORE)



Configuration

Number of Worker(s)	1
Device Type	GPU

GPU Summary

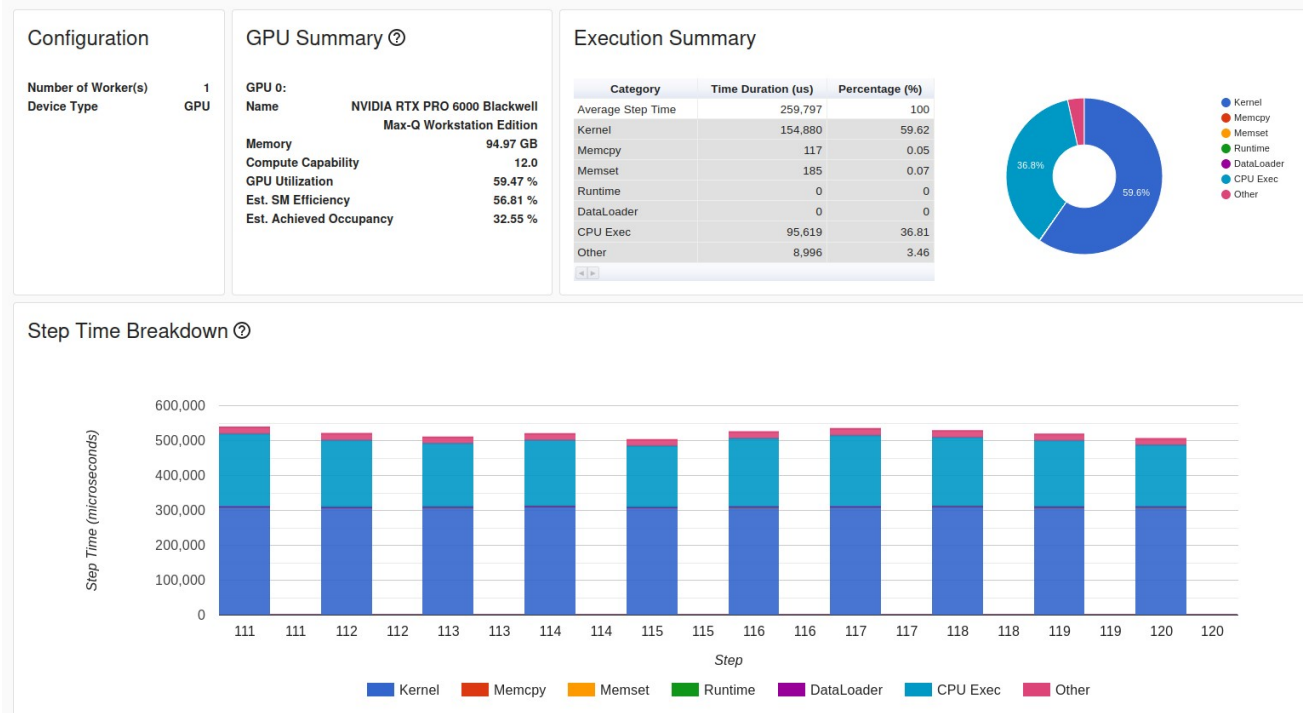
GPU 0:	
Name	NVIDIA RTX PRO 6000 Blackwell Max-Q Workstation Edition
Memory	94.97 GB
Compute Capability	12.0
GPU Utilization	59.16 %
Est. SM Efficiency	56.52 %
Est. Achieved Occupancy	32.55 %

Execution Summary

Category	Time Duration (us)	Percentage (%)
Average Step Time	260,801	100
Kernel	154,603	59.28
Memcpy	117	0.04
Memset	184	0.07
Runtime	0	0
DataLoader	0	0
CPU Exec	96,865	37.14
Other	9,031	3.46

Step Time Breakdown

LET'S PLAY WITH THE DATA MORE: CUDA BUFFERS (AFTER)



LET'S PLAY WITH THE DATA MORE: CUDA BUFFERS

```
for step, batch in enumerate(dataloader):
    if step >= total_num_steps:
        break
    with torch.cuda.stream(prefetch_stream):
        next_batch = {
          k: v.to(
            device=accelerator.device, non_blocking=True
            )
          for k, v in batch.items()}
    torch.cuda.current_stream().wait_stream(prefetch_stream)
    batch = next_batch
    outputs = model(**batch)
    loss = outputs.loss
```

Results (not helpful):

- ~~62,840~~ 57,003 tok/s
- Time to 10B tokens: ~~44.2 hrs~~ **48.73 hrs**

OUT OF DATA IDEAS, THE OPTIMIZER!

Configuration

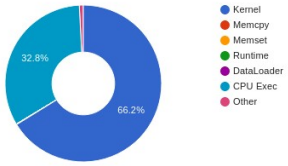
Number of Worker(s)	1
Device Type	GPU

GPU Summary

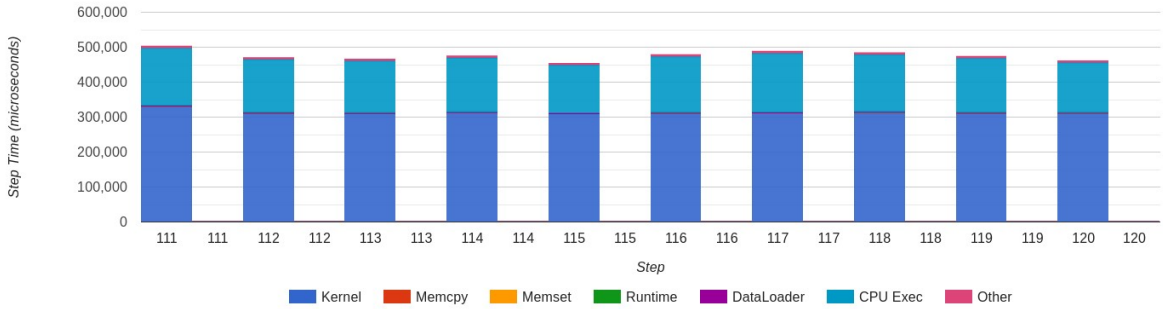
GPU 0:	
Name	NVIDIA RTX PRO 6000 Blackwell Max-Q Workstation Edition
Memory	94.97 GB
Compute Capability	12.0
GPU Utilization	66.19 %
Est. SM Efficiency	63.38 %
Est. Achieved Occupancy	32.41 %

Execution Summary

Category	Time Duration (us)	Percentage (%)
Average Step Time	237,094	100
Kernel	156,978	66.21
Memcpy	117	0.05
Memset	187	0.08
Runtime	0	0
DataLoader	0	0
CPU Exec	77,838	32.83
Other	1,973	0.83



Step Time Breakdown



LET'S FUSE A KERNEL (THANKS TO CLAUDE)

Configuration

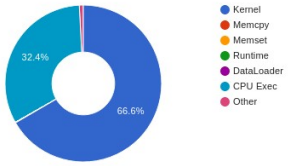
Number of Worker(s)	1
Device Type	GPU

GPU Summary

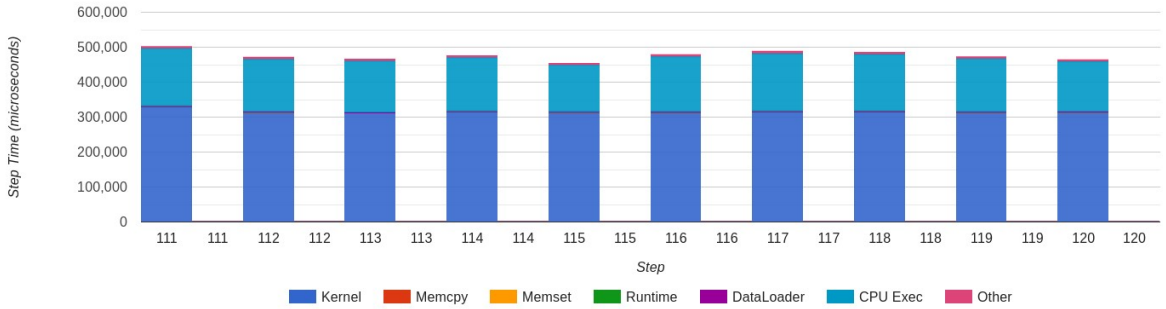
GPU 0:	
Name	NVIDIA RTX PRO 6000 Blackwell Max-Q Workstation Edition
Memory	94.97 GB
Compute Capability	12.0
GPU Utilization	66.61 %
Est. SM Efficiency	63.79 %
Est. Achieved Occupancy	32.29 %

Execution Summary

Category	Time Duration (us)	Percentage (%)
Average Step Time	237,299	100
Kernel	158,109	66.63
Memcpy	117	0.05
Memset	187	0.08
Runtime	0	0
DataLoader	0	0
CPU Exec	76,921	32.42
Other	1,964	0.83



Step Time Breakdown



LET'S FUSE A KERNEL (THANKS TO CLAUDE)

```python
class Qwen3MoeMLP(nn.Module):
    def __init__(self, config, intermediate_size=None):
        ...

    def forward(self, x):
        out = self.act_fn(self.gate_proj(x))
        out *= self.up_proj(x)
        down_proj = self.down_proj(out)

        return down_proj
```

LET'S FUSE A KERNEL (THANKS TO CLAUDE)

```python
class Qwen3MoeMLP(nn.Module):
    def __init__(self, config, intermediate_size=None):
        ...

    def forward(self, x):
        # single GEMM → splits into gate and up
        gate_up = self.gate_up_proj(x)
        gate, up = gate_up.chunk(2, dim=-1)
        out = self.down_proj(self.act_fn(gate) * up)
        return out
```

WE'VE GRADUATED TO MULTI-GPU!

DISTRIBUTED DATA PARALLEL

- See a linear-ish speedup
- 1 -> 4 GPUs, keeping local batch size
- 19.6M tokens seen across 200 batches

Results (vs baseline):

- ~~45,005~~ 118,168 tok/s
- Time to 10B tokens: ~~61.72 hrs~~ **23.5 hrs**

APPLYING ALL OUR EARLIER TRICKS (BAR MODELING TRICKS)

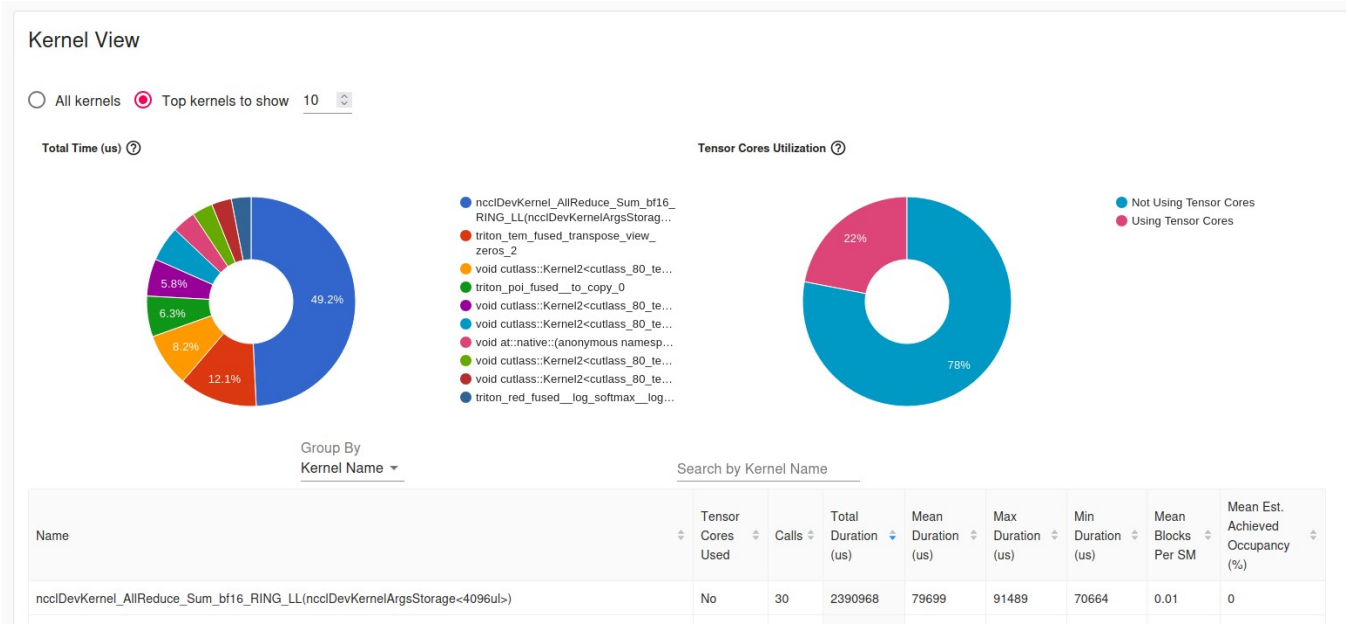
Results:

- ~~118,168~~ 154,870 tok/s
- Time to 10B tokens: ~~23.5 hrs~~ **17.94 hrs**

Now we're doing **really good**

What's our bottleneck?

APPLYING ALL OUR EARLIER TRICKS (BAR MODELING TRICKS)



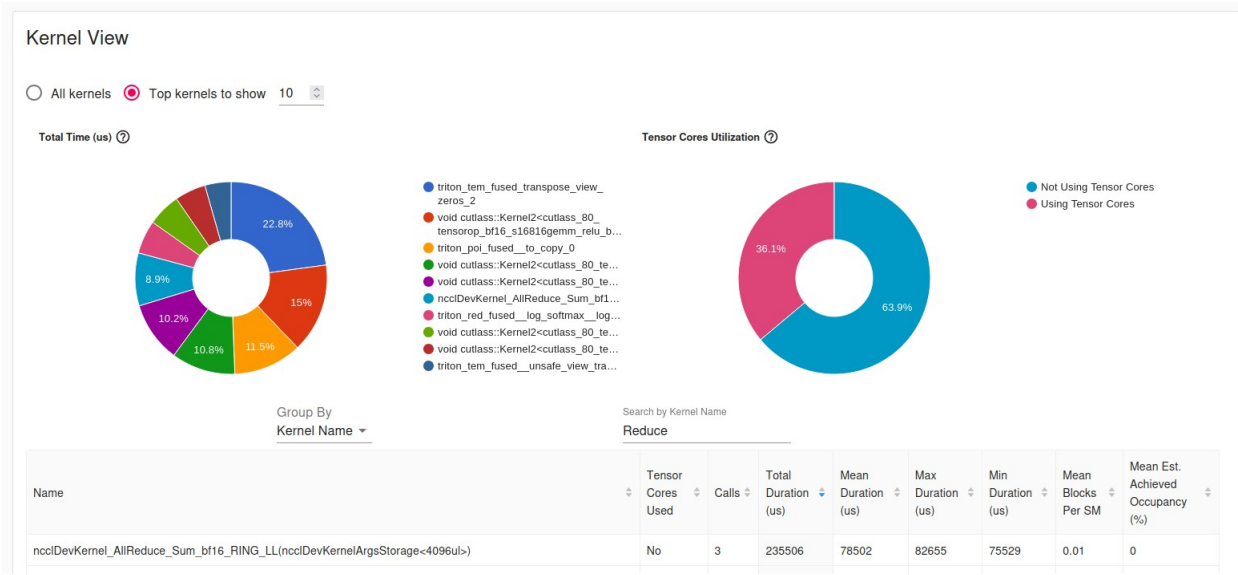
APPLYING ALL OUR EARLIER TRICKS (BAR MODELING TRICKS)

The 0.08s/all_reduce (on PCIe4, 0.04s on PCIe5)

- 10B / (6 * 4 * 4096) = 98,304 tok/batch
- 10B / 98,304 = 101,725 batches
- 0.08 * 101,725 = **2.26 hours purely on AllReduce**

ACHIEVING THE TARGET BATCH SIZE

- Recommended to have a batch token amount of 1M
- Equates to ~grad accum of 10



ACHIEVING THE TARGET BATCH SIZE

- Recommended to have a batch token amount of 1M
- Equates to ~grad accum of 10
- ~~154,870~~ 210,377 tok/s
- Time to 10B tokens: ~~17.94 hrs~~ **13.2 hrs**

LET'S RECAP

STEP 1: OPTIMIZE SINGLE GPU

- Optimal batch size
- `pin_memory=True`
- Offline data
- CUDA Buffers (showed improvement in multi-GPU)
- `AdamW(fused=True)`
- Fused GEMM
- 44,942 -> 63,627 tok/s (41.57% speedup)
- Time to 10B tokens: 61.80 hours -> **43.67 hours**

STEP 1: OPTIMIZE SINGLE GPU

Configuration

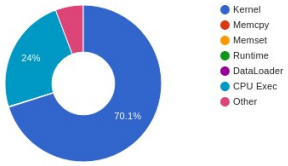
Number of Worker(s)	2
Device Type	GPU

GPU Summary

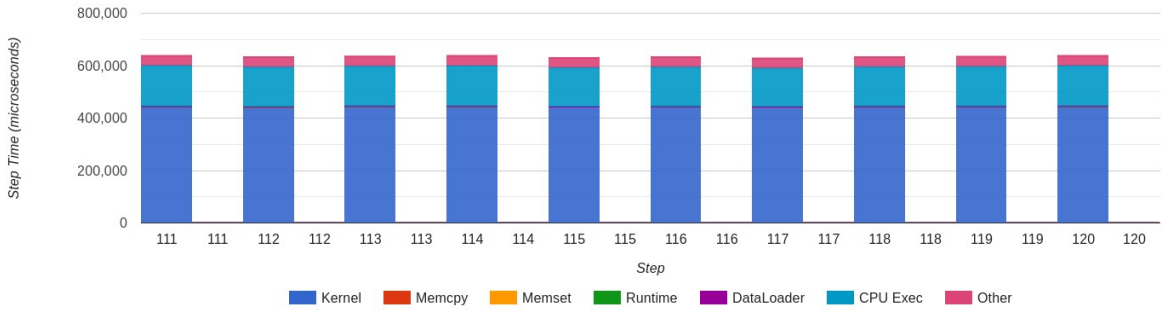
GPU 0:	
Name	NVIDIA RTX PRO 6000 Blackwell Max-Q Workstation Edition
Memory	94.97 GB
Compute Capability	12.0
GPU Utilization	70.15 %
Est. SM Efficiency	67.92 %
Est. Achieved Occupancy	29.11 %

Execution Summary

Category	Time Duration (us)	Percentage (%)
Average Step Time	316,796	100
Kernel	222,057	70.09
Memcpy	118	0.04
Memset	204	0.06
Runtime	0	0
DataLoader	0	0
CPU Exec	76,103	24.02
Other	18,313	5.78



Step Time Breakdown



STEP 1: OPTIMIZE SINGLE GPU

Configuration

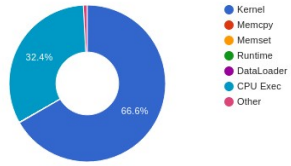
Number of Worker(s)	1
Device Type	GPU

GPU Summary

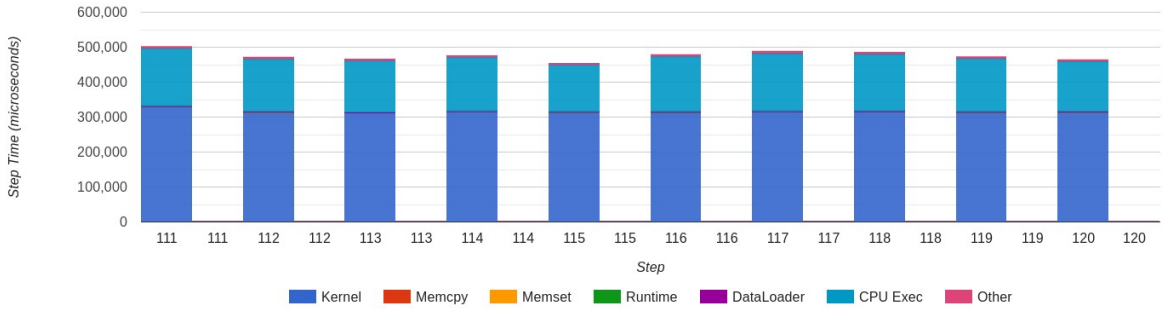
GPU 0:	
Name	NVIDIA RTX PRO 6000 Blackwell Max-Q Workstation Edition
Memory	94.97 GB
Compute Capability	12.0
GPU Utilization	66.61 %
Est. SM Efficiency	63.79 %
Est. Achieved Occupancy	32.29 %

Execution Summary

Category	Time Duration (us)	Percentage (%)
Average Step Time	237,299	100
Kernel	158,109	66.63
Memcpy	117	0.05
Memset	187	0.08
Runtime	0	0
DataLoader	0	0
CPU Exec	76,921	32.42
Other	1,964	0.83



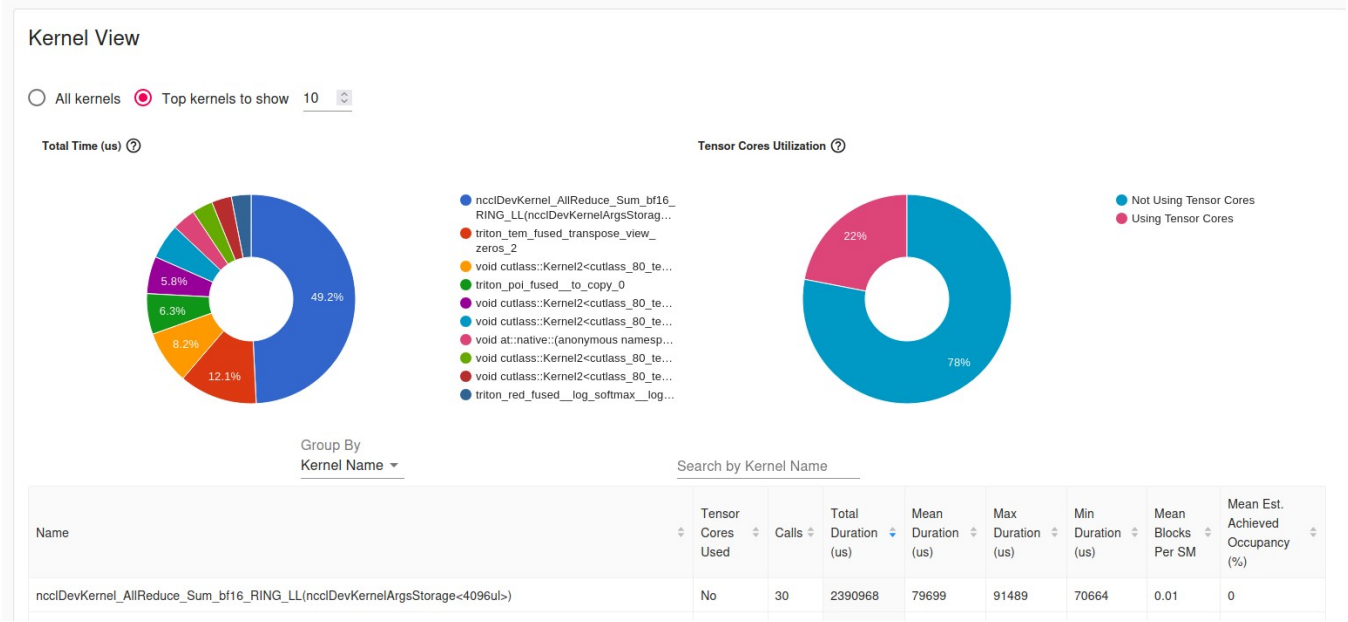
Step Time Breakdown



STEP 2: OPTIMIZE MULTI-GPU

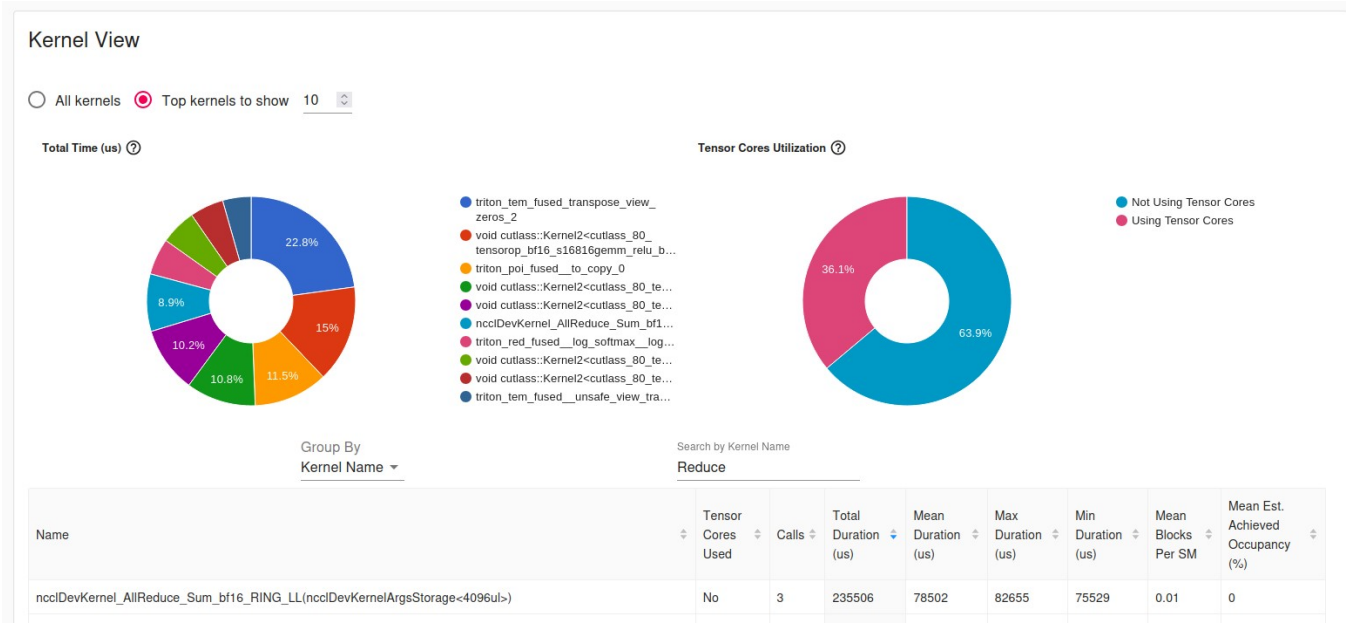
- Verify single GPU results align at scale
- Minimize comms as much as possible
- Saturate ~1M tok/batch using gradient accumulation
- 63,627 -> 154,870 -> 210,377 tok/s (230% | 35.85%)
- 43.67 -> 17.94 -> **13.2 hours**

STEP 2: OPTIMIZE MULTI-GPU



Name	Tensor Cores Used	Calls	Total Duration (us)	Mean Duration (us)	Max Duration (us)	Min Duration (us)	Mean Blocks Per SM	Mean Est. Achieved Occupancy (%)
ncclDevKernel_AllReduce_Sum_bf16_RING_LL(ncclDevKernelArgsStorage<4096ul>)	No	30	2390968	79699	91489	70664	0.01	0

STEP 2: OPTIMIZE MULTI-GPU



Name	Tensor Cores Used	Calls	Total Duration (us)	Mean Duration (us)	Max Duration (us)	Min Duration (us)	Mean Blocks Per SM	Mean Est. Achieved Occupancy (%)
ncclDevKernel_AllReduce_Sum_bf16_RING_LL(ncclDevKernelArgsStorage<4096ul>)	No	3	235506	78502	82655	75529	0.01	0

THANKS FOR COMING!