

Puro-2B：全流程开源语言模型训练

用 \$5090 在 RTX5090 上挑战 Qwen2-1.5B

罗开荣 崔家瑞 殷尧瑞 陈晟祺 杨一鸣 高麟翔 王严墨涵 李彬夏 张铭哲 吕凯风* 陈文光*

清华大学 / 鹏城实验室

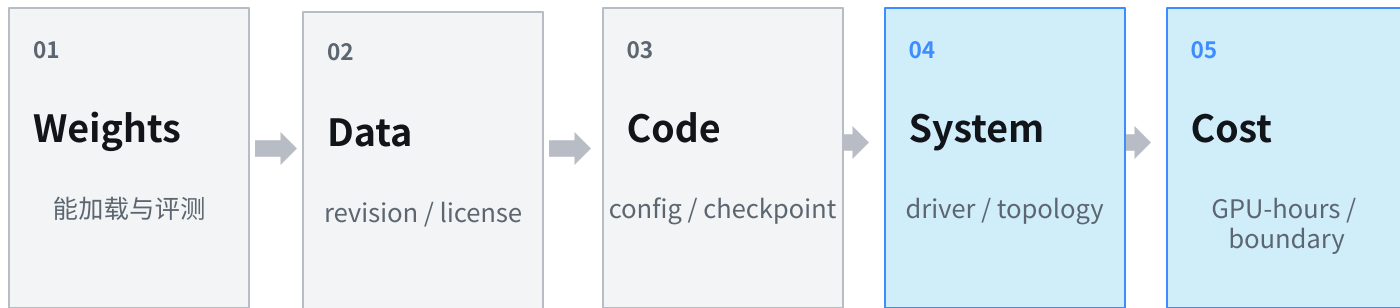
<https://hf.co/collections/thu-pacman/puro-2b>
<https://arxiv.org/abs/2608.27370>

从开源权重到开源配方



我们的目标：数据+系统+权重+成本估算全流程开放、可复现 → 取名 “**普罗**” / Puro

开源 != 可复现



权重能下载

≠ 训练能复现

普罗公开：模型、训练数据、训练代码、数据处理代码与多个版本 checkpoint。

小模型 != 低成本

\$1.50M

Llama-3.2-3B

reported H100 GPU-hours
开源权重

\$718.8K

SmolLM3-3B

reported count × time
开源配方

\$199.7K

OLMoE-1B/7B

reported count × time
开源配方

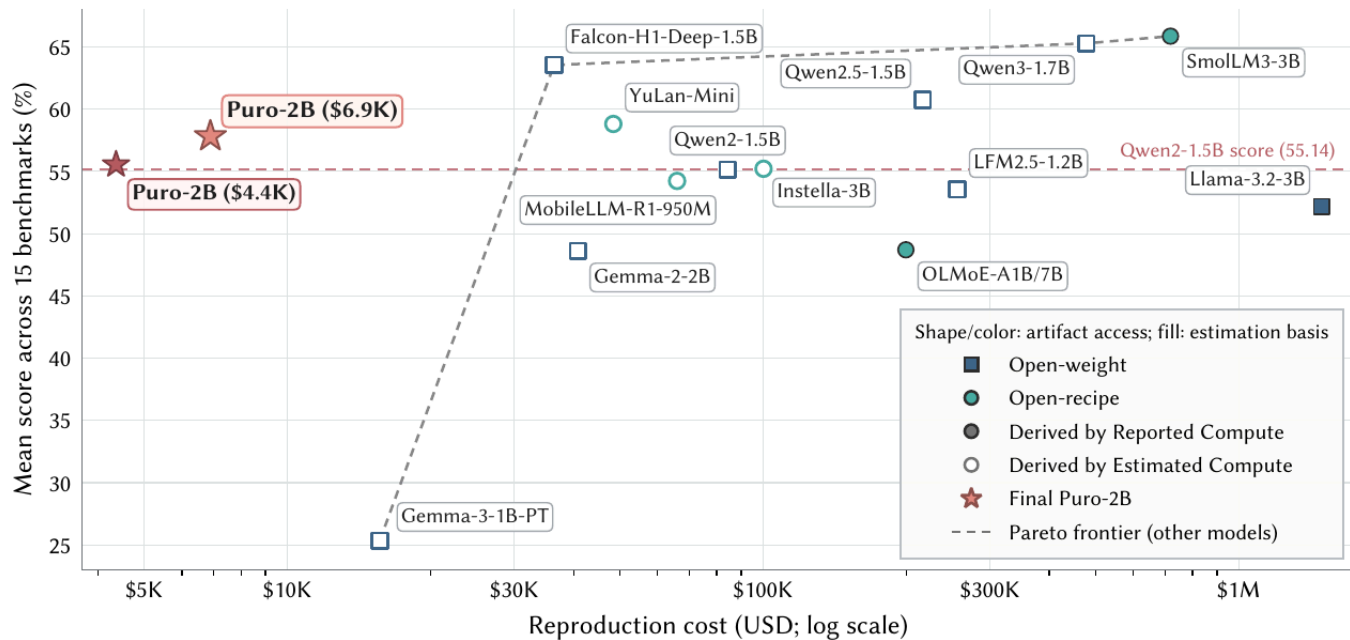
\$6.9K

Puro-2B (普罗-2B)

measured active-training GPU-
hours

还是太贵了?

小模型训练的成本—性能 Pareto 前沿



\$6.9K

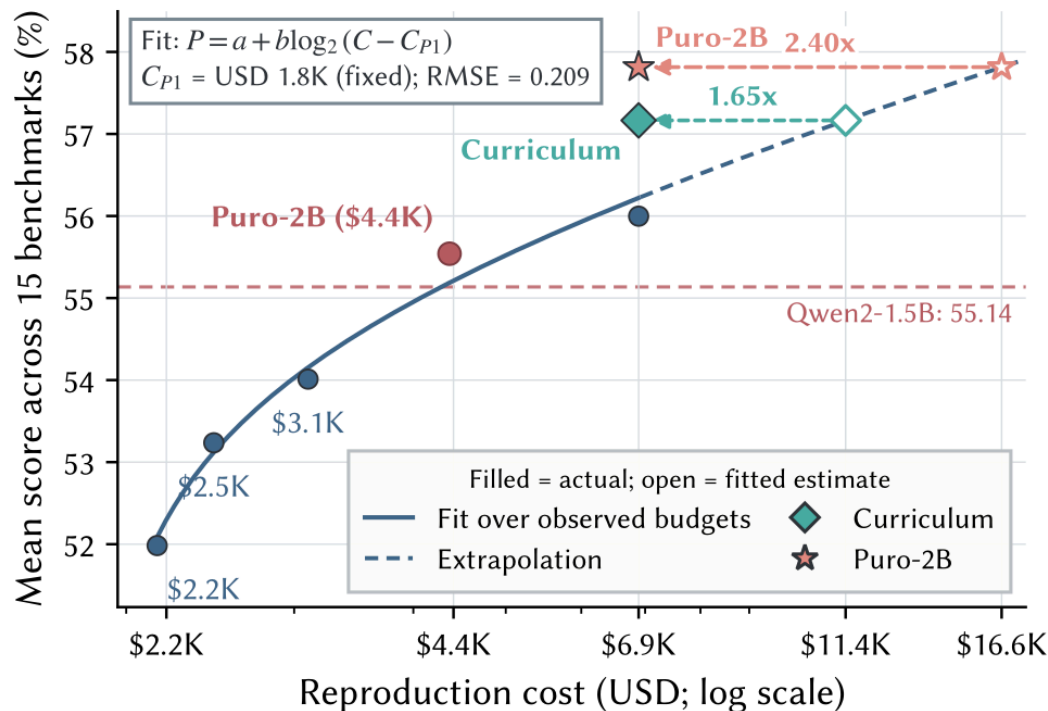
Puro-2B
Avg 57.81

\$4.4K

uniform
checkpoint
已超过 Qwen2-1.5B

横轴为报告口径下的 复现成本；纵轴为 15 个基准测试的非加权平均分。

Puro Cost Scaling Law

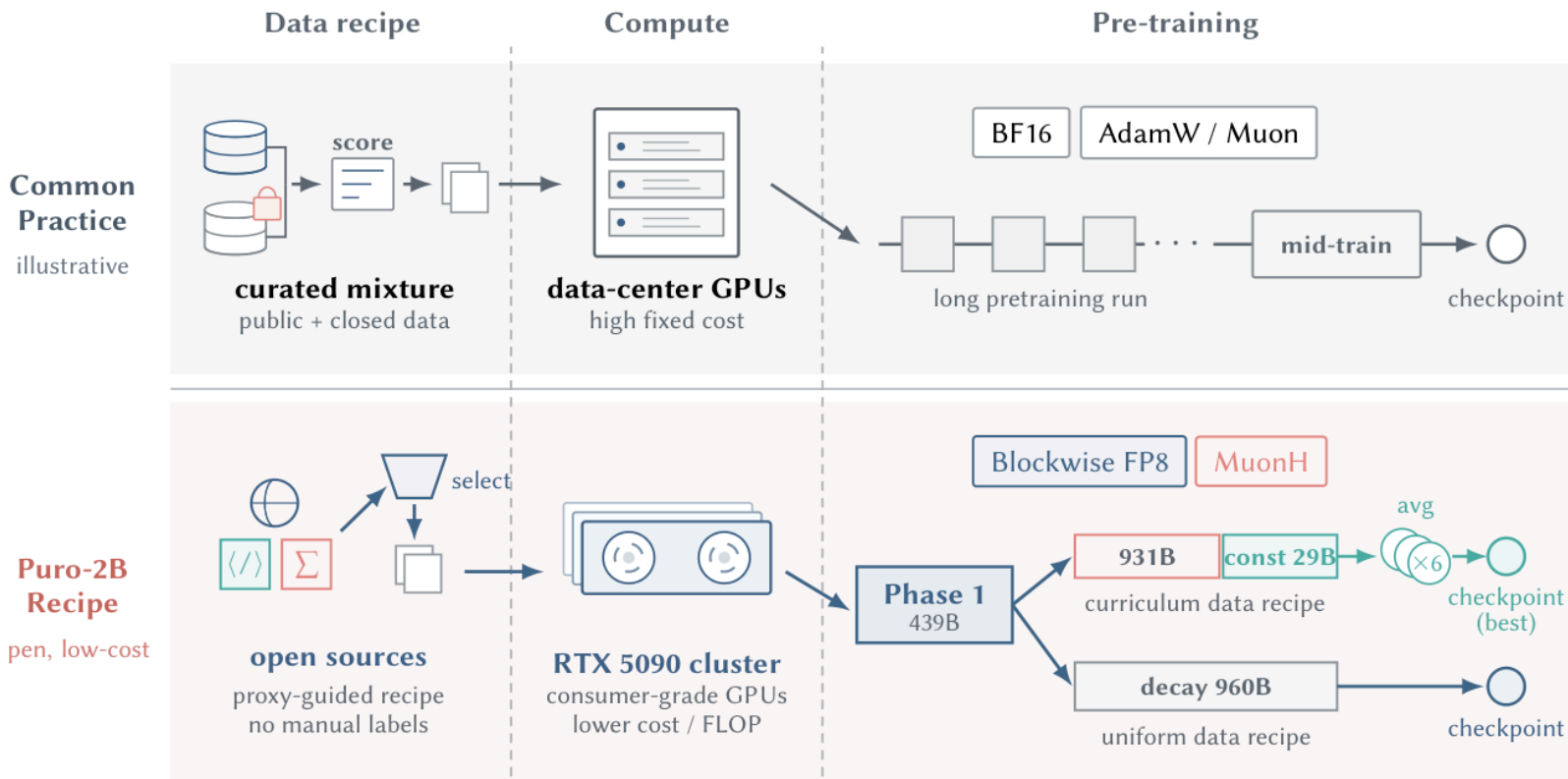


$$P = a + b \log_2(C - C_p)$$

前提: C_{p1} 固定为 \$1.84K: 只拟合 Phase 2 的增量成本

结论: 约 \$4.4K 的 uniform checkpoint 性能已越过 Qwen2-1.5B

普罗训练 Pipeline 概览



为什么选择 RTX 5090?

参考云平台租赁价 (8/16)

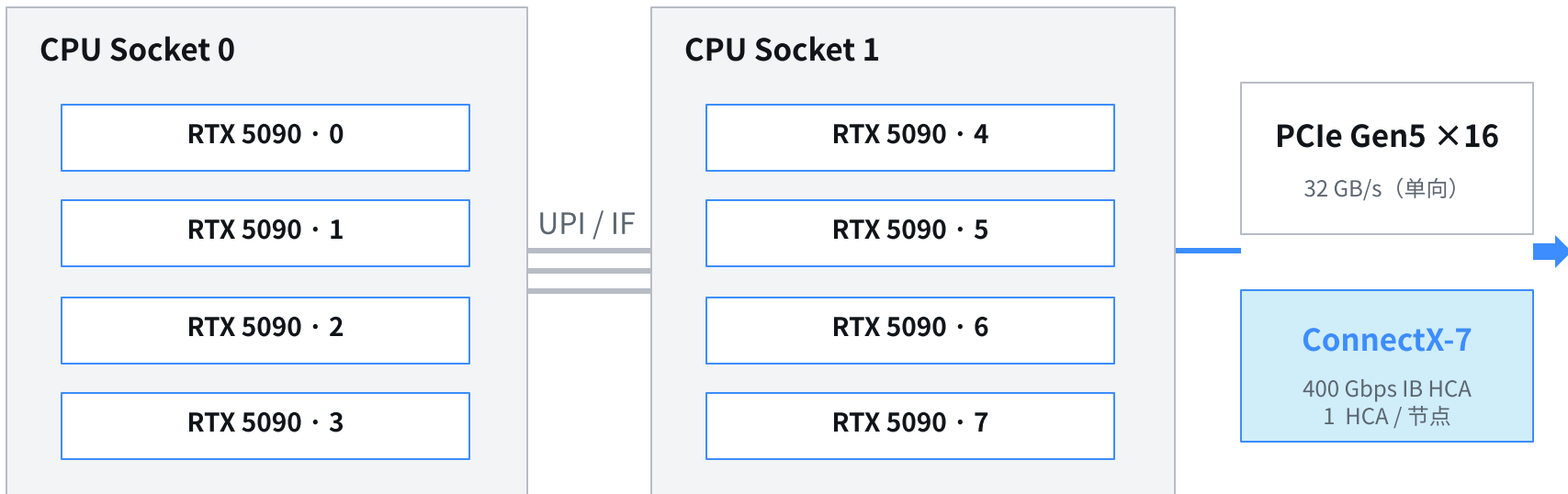
GPU	BF16 TFLOPS	FP8 TFLOPS	显存	双向 P2P 带宽	\$/GPU·h	BF16 EFLOP/\$
A100 SXM4	312	N/A	80 GB	600 GB/s	\$1.79	0.63
H200 SXM5	989.5	1979	141 GB	900 GB/s	\$4.00	0.89
RTX PRO 6000	503.8	1007.6	96 GB	128 GB/s	\$1.89	0.96
RTX 5090	209.5	419	32 GB	128 GB/s	\$0.31	2.43

2.77×

BF16 peak compute / \$

峰值性能按稠密矩阵乘法 (FP32 精度累加) 计

RTX 5090 很便宜，但代价是什么？



与数据中心 GPU 相比，RTX 系列缺少必要的通信功能：

- PCIe P2P: **机内**通信完全绕过 CPU / 主存
- GPUDirect RDMA: **机间**通信绕过 CPU (依赖 P2P 而工作)

魔改开源驱动打开 P2P 支持



现有工作: <https://github.com/aikitoria/open-gpu-kernel-modules>
在补丁的基础上, 适配 Ubuntu / Debian 的 DKMS 驱动包

31.5 → 56

单向带宽 · GB/s

32 → 111

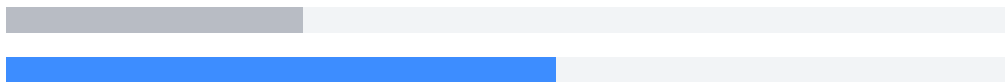
双向带宽 · GB/s

14.3 → 0.4

通信延迟 · μ s

NCCL AllReduce (busbw)

8 GPUs / 节点内



Before

14.75 GB/s

After

27.34 GB/s

4 GPUs / Socket 内



Before

16.33 GB/s

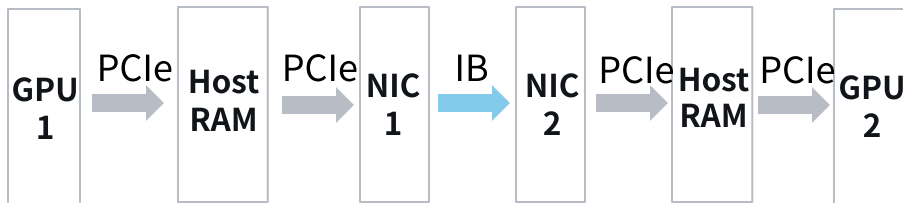
After

46.31 GB/s

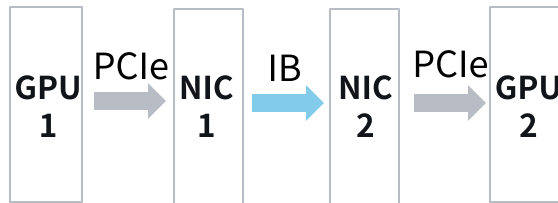
魔改 CUDA 打开 GPU DirectRDMA 支持



Before · staged through host memory



After · GDR Enabled



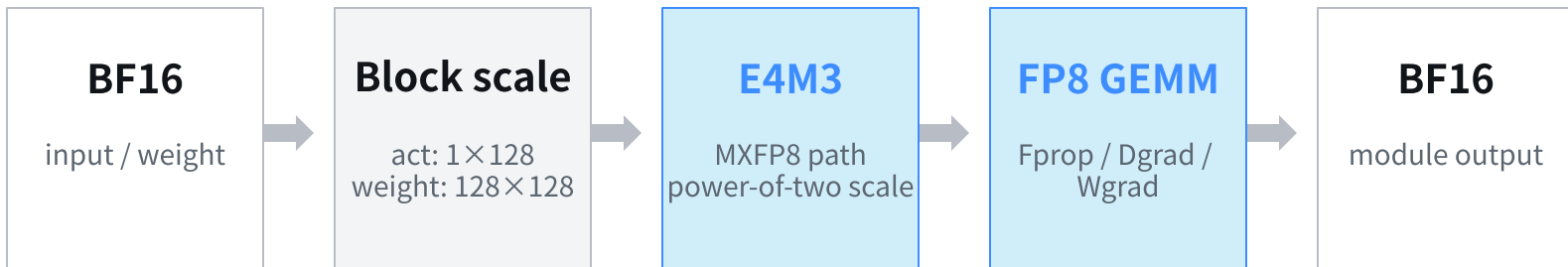
借助 Codex 调试、分析、打补丁
最终仅修改 libcuda.so 中的一个字节即可

8.87 → 19.93

24-GPU AllReduce · busbw (GB/s)

3 servers · 400 Gbps InfiniBand

原生低精度训练：类 DeepSeek 的 FP8 配方



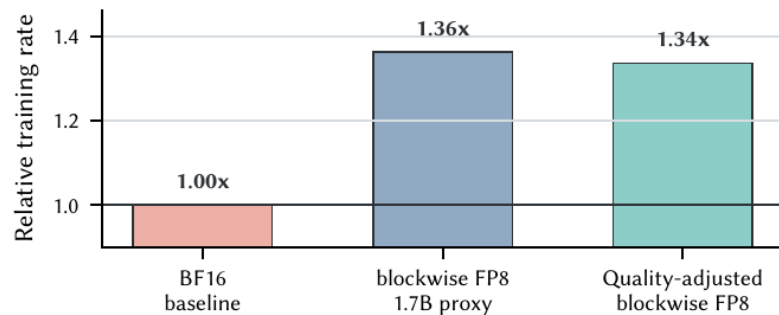
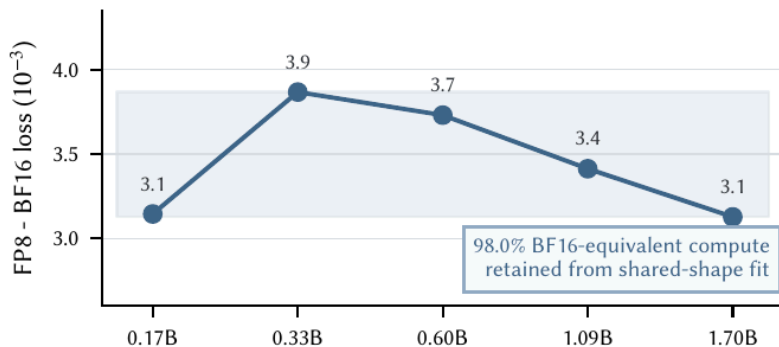
保留 BF16

Embedding · RMSNorm / residual · FlashAttention/SDPA · LM head · pipeline activations

保留 FP32

gradient accumulation / reduction · master weights · optimizer states · MuonH state

原生 FP8 的实际收益



98.0%

BF16-等价计算量

1.36×

吞吐加速比

1.34×

净加速比

0.0031–0.0039

validation-loss 差异

并行训练策略选择

策略	主要通信	使用	原因
DP	ReduceScatter / AllGather	是	通信集中在 optimizer 周期
PP	stage boundary hidden states	是	把 PP group 放在更近的 GPU
TP	每层频繁 collective	否	PCIe-only 环境代价过高

rank ordering: PP-DP, 让 PP 组尽量留在 socket 内。

MBS = 2: 取 FP8 kernel roofline 最优, 并充分利用显存。

LM head stage 少放 Transformer layers; optimizer state 用 memory-aware allocation。

模型小、词表大时, LM head 的 BF16 计算会接近多个 FP8 Transformer layer。

因此 PP 不能平均切层, 需要按理论计算量与内存压力综合考虑。

从 24 卡到 96 卡

Phase	Tokens	GPUs	TP / PP / DP	PP layout	GBS / MBS	NIC / Gbps	TFLOP/s/G PU
Phase 1	439B	24	1 / 2 / 12	(18 10)	1536 / 2	400	238
Phase 2*	961B	96	1 / 4 / 24	(9 9 9 1)	1536 / 2	200	192

73%

Phase 1 等价 MFU

[18 | 10]

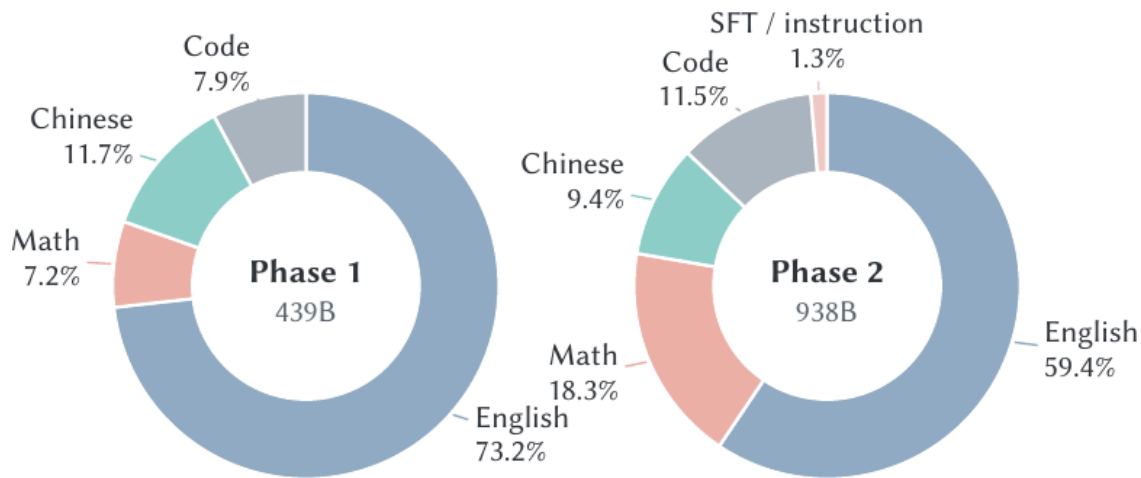
一阶段 PP 负载均衡

[9 | 9 | 9 | 1]

二阶段 PP 负载均衡

* 感谢衍复投资提供第二阶段训练的计算资源

训练数据概览



Phase 1 • 439B

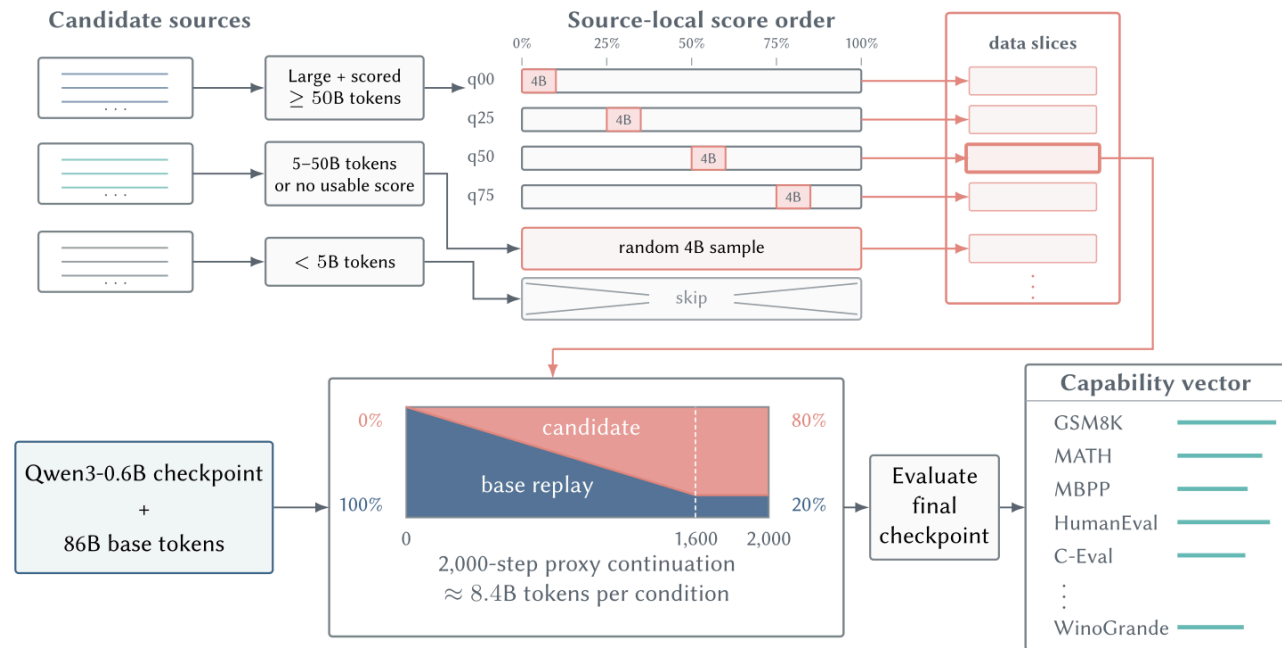
English 73.2%
Math 7.2%
Chinese 11.7%
Code 7.9%

Phase 2 • 938B

Math ↑ 18.3%
Code ↑ 11.5%
SFT / instruction 1.3%

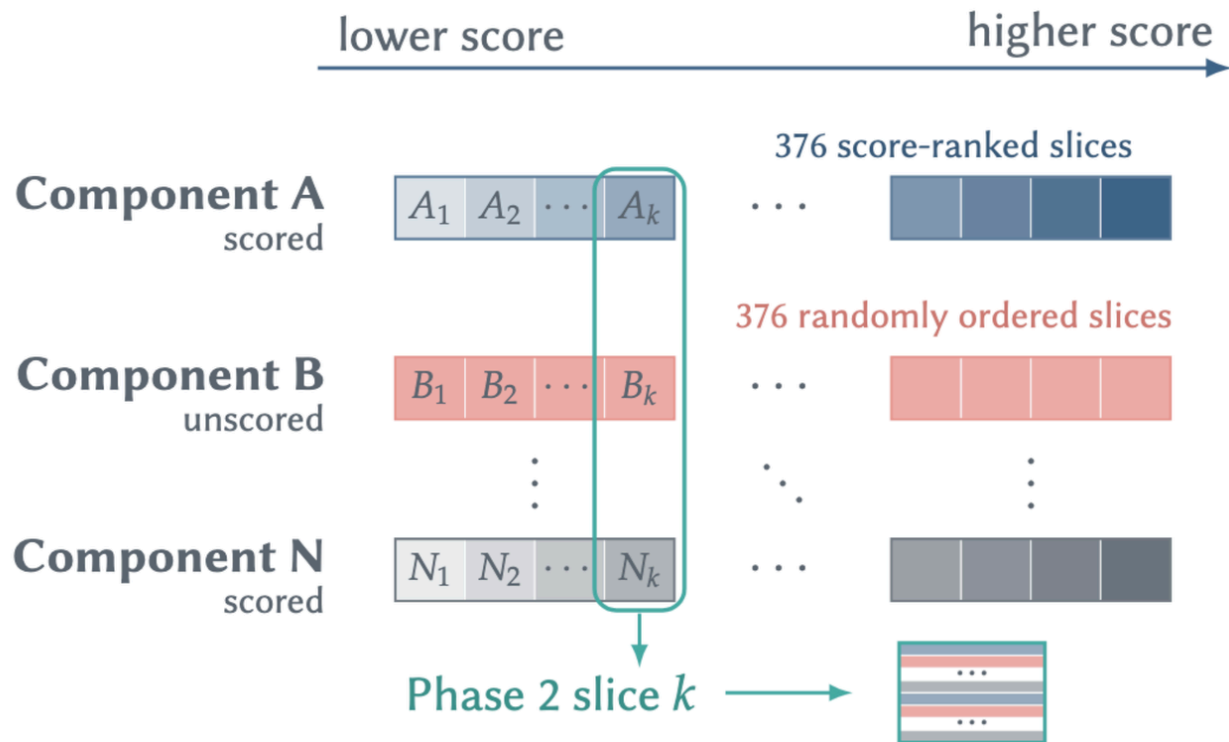
Phase 1 保持较广覆盖；Phase 2 更偏向数学与编码能力

训练数据选择：Proxy Benchmarking



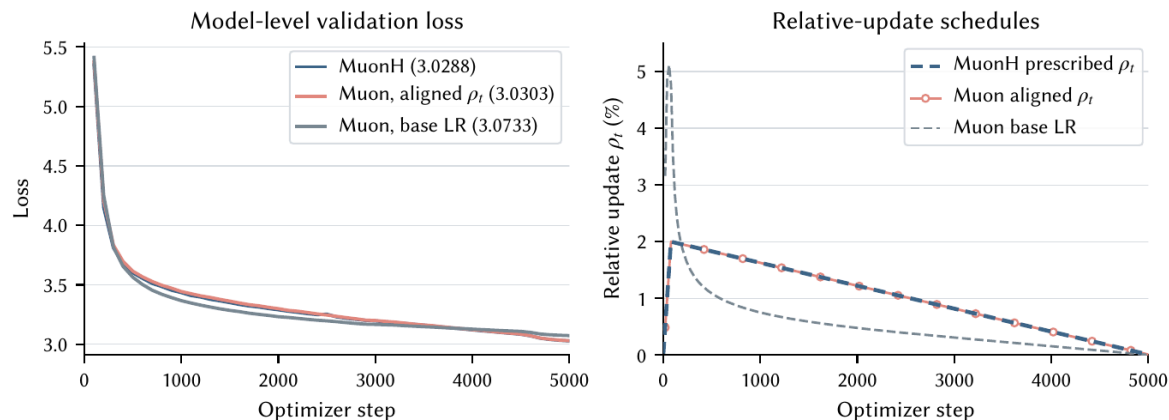
- 评价数据集绝对能力
- 决定数据的配比

训练数据排序：课程学习 (Curriculum)



- 数据集内独立排序，并按相对比例对齐
- 决定向模型提供数据的顺序（课程）
- 可配合学习率调整

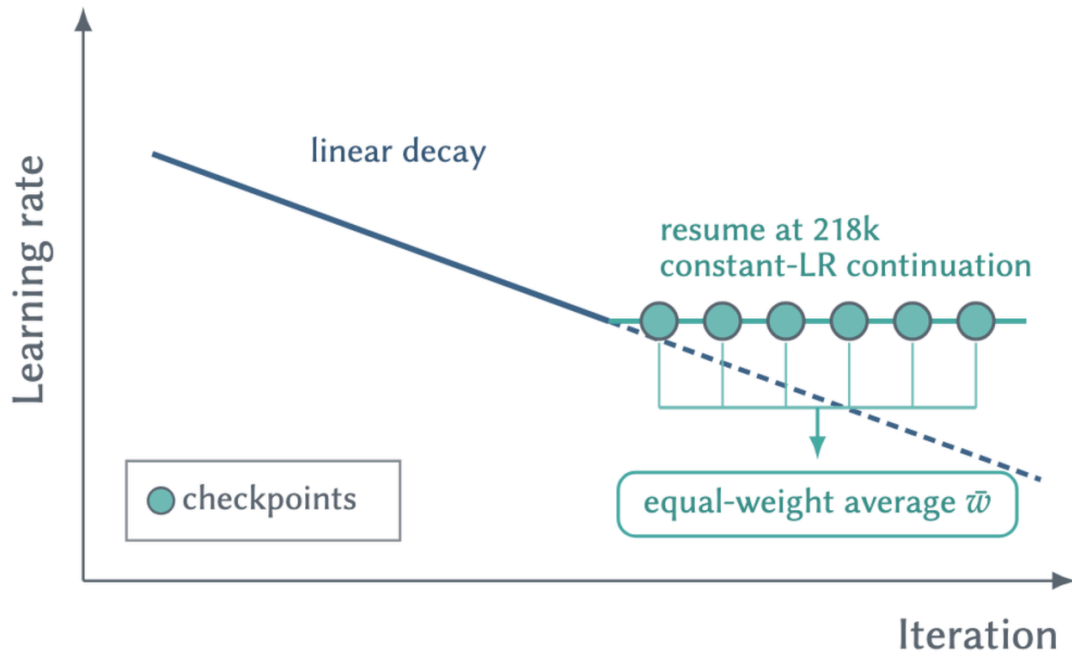
混合优化器：MuonH + AdamW



- Hyperball 优化器将梯度控制在球面上，限制了更新大小
- 提升数值稳定性，但也可能带来不可恢复的下溢
- 不同的层需要不同优化策略

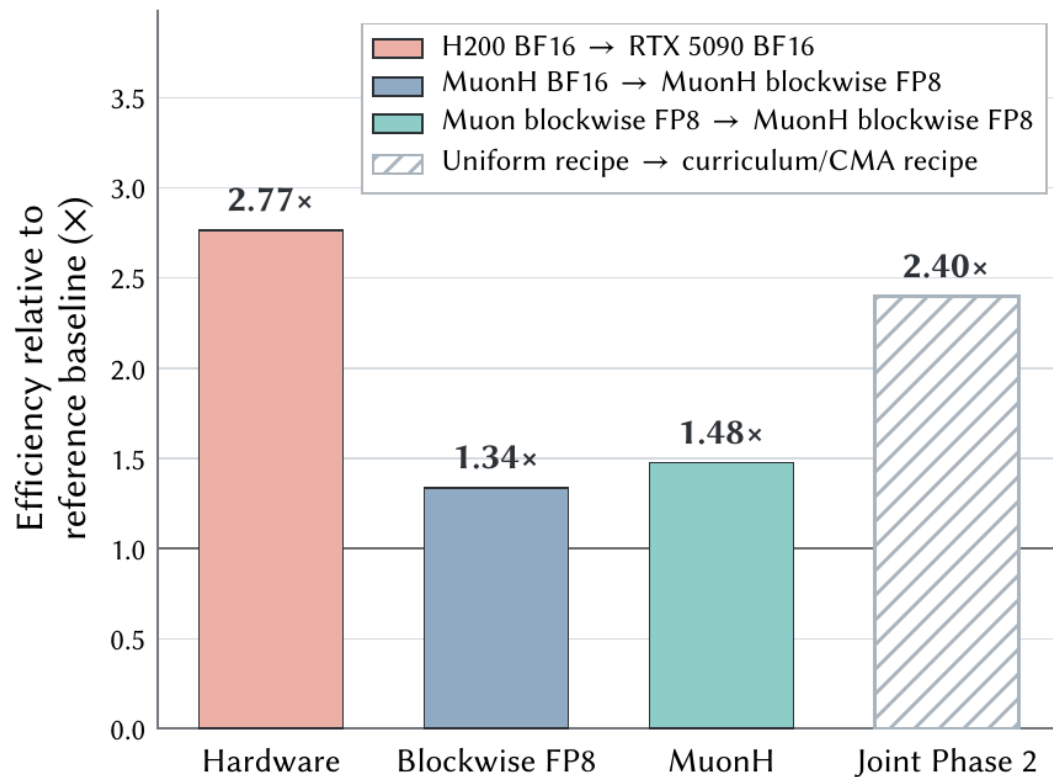
参数归属	优化器	Weight decay	学习率
attention / MLP matrices	MuonH	0	$10 \times$ base schedule
embedding / norm / LM head / others	AdamW	0.1	base schedule

最终模型：CMA 权重平均



- 末端低学习率低→浪费数据
- 高学习率→随机噪声降低质量
- 方案：多次训练，取平均

回顾：成本降低来源拆分



Puro-2B is Now Fully Open-Source!

罗开荣 崔家瑞 殷尧瑞 陈晟祺 杨一鸣 高麟翔 王严墨涵 李枨夏 张铭哲 吕凯风* 陈文光*

- 集合: [hf.co/thu-pacman/Puro-2B](https://huggingface.co/thu-pacman/Puro-2B)
- 模型: [hf.co/thu-pacman/Puro-2B-Base](https://huggingface.co/thu-pacman/Puro-2B-Base) 或 ModelScope 同名
- 数据: [hf.co/thu-pacman/Puro-2B](https://huggingface.co/thu-pacman/Puro-2B) 或 ModelScope 同名
- 代码: github.com/thu-pacman/Puro-Megatron
- 报告: <https://arxiv.org/abs/2608.27370>

