T2-1-2 服务能力优化赛题报告

参赛小组：加班仙人球

参赛者：肖虎诚

GitHub ID：CearX

最终提交：InfiniLM 单 PR，单卡 8B

1. 摘要

本项目面向单卡 8B 模型的多并发、长文本服务场景，优化 InfiniLM 的调度、请求生命周期和 OpenAI-compatible 服务热路径。最终 PR 仅包含 9 个 InfiniLM Python 实现与测试文件。

在 NVIDIA GeForce RTX 5090 32 GiB、TP=1、Qwen3-8B BF16、burst 请求条件下，最终实现完成官方 30 点 8B 矩阵，全部请求成功且输出长度严格命中。与同机、同模型、同客户端口径的初始服务基线在 27 个共同有效点比较，output throughput 与 total-token throughput 的几何平均提升均为 13.56%。高并发长输出关键点提升 67.44% 至 82.67%；小规模 c1/i32/o256 为 -0.62%，未出现明显下降。

2. 问题分析与方案

2.1 服务瓶颈

高并发和长输出同时出现时，服务性能不只取决于单个 CUDA kernel。本次定位的主要问题包括：

- 长 prompt 的整段 prefill 会阻塞同批 decode 请求；
- burst admission 若不预留 KV 空间，长输出容易在运行后期形成内存压力；
- 每 token 一次 Python queue/SSE 事件会放大高并发控制面开销；
- 非流式请求的轮询与重复 decode 会增加额外 CPU 开销；
- OpenAI 字段兼容和 length termination 需与性能优化同时保证。

2.2 Chunked Prefill

调度器将长 prompt 切分为有界 chunk，在 prefill 与 decode 之间创建更细的调度机会，降低长 prompt 对已在生成请求的头阻塞。中间 prompt chunk 只更新 KV 状态，不对外发送采样 token；最后一个 chunk 才进入正常生成流程。

2.3 KV Admission 安全水位

新请求入场时不再只检查当前 prompt 所需 block，而是保留可配置的 KV 安全水位。普通请求与长 prompt 使用不同阈值，以平衡 admission 并发度和长输出稳定性。该机制只延后新请求入场，不降低配置的最大 batch size。

2.4 服务热路径

- 新增 `stream_interval`：首 token 立即发送，之后按间隔合并 SSE 事件，不丢弃模型 token；
- 非流式请求使用 completion event 等待完成，最终文本只 decode 一次；
- 兼容 `max_tokens` 与 `max_completion_tokens`，补齐 usage 字段；
- 修正 LENGTH 终止路径，确保最后一个采样 token 不丢失。

3. 提交范围

3.1 代码差异

最终分支基于 InfiniLM 上游 `main`，仅包含一个 Conventional Commit：

```
perf(server): optimize concurrent long-output serving
```

差异为 9 files, +409/-45：

最终提交：`9d118ec5fcb643858b0b908154647d05d55d743d`。

类别	文件	用途
调度	`llm/scheduler.py`	chunked prefill 与 KV admission
请求	`llm/request.py`	completion event
引擎	`llm/llm.py`	SSE 合并与非流式快路径
服务	`server/inference_server.py`	OpenAI 兼容与非流式路径
正确性	`processors/basic_llm_processor.py`	chunk compute 与 length 终止
配置	3 个配置/运行文件	参数穿透
测试	`official_client_bench.py`	官方风格复现客户端

3.2 未纳入内容

• 70B 模型测试结果。

4. 评测方法

4.1 环境

• GPU：NVIDIA GeForce RTX 5090 32 GiB；

• 模型：Qwen3-8B BF16；

• 并行：TP=1，单卡；

• 接口：OpenAI `/v1/chat/completions`，streaming SSE；

• 负载：burst，`request_rate=inf`，`num_prompts=max_concurrency`；

• 生成：`ignore_eos=true`，要求输出长度精确命中；

• 复现 seed：`0`。

4.2 官方 8B 矩阵

并发	输入 **tokens**	输出 **tokens**
1, 4	32, 256, 4096	256, 1024, 4096
16, 64	32, 256	256, 1024, 4096

总计 30 个组合。基线与最终实现使用同一硬件、模型、请求口径和客户端。基线有 27 个共同有效点，几何平均只在这 27 个共同点上计算。

5. 评测结果

5.1 总体结果

• 最终 30/30 点完成；

• 所有最终请求的输出长度严格命中；

• 所有最终请求的 error 列表为空；

• 27 个共同有效点的 output throughput 几何平均：+13.56%；

• 27 个共同有效点的 total-token throughput 几何平均：+13.56%。

5.2 关键对比

场景	基线 output tok/s	最终 output tok/s	变化
c64/i32/o256	1443.47	3092.98	+114.27%
c64/i32/o1024	1005.39	1683.44	+67.44%
c64/i32/o4096	361.93	661.13	+82.67%
c16/i256/o4096	293.68	515.95	+75.69%
c1/i32/o256	92.34	91.77	-0.62%

高并发、长输出场景取得明显提升；单请求短输入点的 -0.62%
属于小幅波动，满足“小规模性能无明显下降”的目标。

5.3 分组结果

为避免只挑选最佳点，对 27 个共同有效点按并发数和输出长度分组计算 output-throughput 几何平均。

分组	共同点数	几何平均变化
C=1	9	-0.09%
C=4	9	+5.44%
C=16	6	+19.81%
C=64	3	+87.14%
Output=256	9	+7.50%
Output=1024	9	+5.20%
Output=4096	9	+29.50%

分组结果直接与赛题目标一致：优化在小规模场景基本持平，在高并发和长输出场景的收益最明显。

5.4 最终 30 点完整矩阵

C	Input	Output	Completed	Output tok/s	Total tok/s	Mean TTFT ms
1	32	256	1/1	91.77	103.25	21.72
1	32	1024	1/1	89.71	92.52	20.40
1	32	4096	1/1	80.13	80.76	20.46
1	256	256	1/1	90.43	180.86	21.68
1	256	1024	1/1	87.83	109.79	20.97
1	256	4096	1/1	78.87	83.80	24.75
1	4096	256	1/1	68.81	1169.85	99.24
1	4096	1024	1/1	68.11	340.53	94.27

C	Input	Output	Completed	Output tok/s	Total tok/s	Mean TTFT ms
1	4096	4096	1/1	62.84	125.68	94.34
4	32	256	4/4	343.03	385.90	37.33
4	32	1024	4/4	334.99	345.46	36.40
4	32	4096	4/4	297.52	299.84	39.67
4	256	256	4/4	328.53	657.05	90.31
4	256	1024	4/4	326.78	408.40	101.90
4	256	4096	4/4	292.31	310.31	104.58
4	4096	256	4/4	116.60	1975.24	3487.68
4	4096	1024	4/4	192.69	963.47	3731.67
4	4096	4096	4/4	215.27	428.94	3333.15
16	32	256	16/16	1250.78	1407.13	73.55
16	32	1024	16/16	1183.72	1218.97	79.22
16	32	4096	16/16	534.44	538.61	16987.97
16	256	256	16/16	1064.29	2121.04	335.62
16	256	1024	16/16	1098.77	1373.46	326.06
16	256	4096	16/16	515.95	548.20	26011.37
64	32	256	64/64	3092.98	3479.41	297.23
64	32	1024	64/64	1683.44	1734.74	6273.77
64	32	4096	64/64	661.13	666.29	149496.04
64	256	256	64/64	2145.20	4277.17	1312.21
64	256	1024	64/64	1573.19	1965.67	9675.59
64	256	4096	64/64	582.68	619.07	190070.41

`Completed` 列证明每个点的请求均完成；同时客户端校验每个请求的实际输出长度与目标一致，且 errors=0。

5.5 27 个共同点完整对比

C	Input	Output	Baseline	Final	Delta
1	32	256	92.34	91.77	-0.62%
1	32	1024	89.74	89.71	-0.03%
1	32	4096	80.13	80.13	-0.00%
1	256	256	90.69	90.43	-0.29%
1	256	1024	87.99	87.83	-0.18%
1	256	4096	79.08	78.87	-0.27%
1	4096	256	68.53	68.81	+0.42%
1	4096	1024	68.03	68.11	+0.11%

C	Input	Output	Baseline	Final	Delta
1	4096	4096	62.81	62.84	+0.04%
4	32	256	348.04	343.03	-1.44%
4	32	1024	337.20	334.99	-0.65%
4	32	4096	300.38	297.52	-0.95%
4	256	256	332.65	328.53	-1.24%
4	256	1024	331.30	326.78	-1.36%
4	256	4096	293.85	292.31	-0.52%
4	4096	256	121.06	116.60	-3.69%
4	4096	1024	197.11	192.69	-2.24%
4	4096	4096	118.23	215.27	+82.08%
16	32	256	1273.52	1250.78	-1.79%
16	32	1024	1193.67	1183.72	-0.83%
16	32	4096	299.78	534.44	+78.28%
16	256	256	1090.00	1064.29	-2.36%
16	256	1024	1106.45	1098.77	-0.69%
16	256	4096	293.68	515.95	+75.69%
64	32	256	1443.47	3092.98	+114.27%
64	32	1024	1005.39	1683.44	+67.44%
64	32	4096	361.93	661.13	+82.67%

对比表保留所有正负结果，未删除负收益点。其中 C=1 分组几何平均仅 -0.09%；C=4 的两个负收益较大点为长输入与中短输出组合，但 C4/i4096/o4096 提升 82.08%。

6. 正确性与风险控制

6.1 正确性门槛

- chunked prefill 中间段不向客户端泄漏采样 token；
- `stream_interval` 只合并事件，不丢 token；
- LENGTH 终止保留最后一个 token；
- 非流式请求使用 completion event，完成后一次性 decode；
- 官方矩阵通过“请求成功 + 输出长度精确 + errors=0”三重检查。

6.2 兼容性

新增性能控制均通过配置或环境变量接入，不修改模型权重、采样算法或对外 API 路径。未使用的配置保持原有路径。

7. 复现说明

7.1 服务端

```
export INFINILM_ENABLE_CHUNKED_PREFILL=1
export INFINILM_KV_SAFETY_BLOCKS=8
export INFINILM_LONG_PROMPT_KV_SAFETY_BLOCKS=64

python python/infinilm/server/inference_server.py \
  --device nvidia \
  --model /path/to/qwen3-8b \
  --port 8102 \
  --tp 1 \
  --max-new-tokens 4096 \
  --num-blocks 384 \
  --block-size 128 \
  --max-batch-size 64 \
  --stream-interval 16 \
  --enable-graph \
  --enable-paged-attn \
  --attn flash-attn \
  --ignore-eos
```

7.2 客户端

```
python scripts/competition/official_client_bench.py \
  --url http://127.0.0.1:8102/v1/chat/completions \
  --model Qwen3-8B \
  --tokenizer /path/to/qwen3-8b \
  --num-prompts 64 \
  --max-concurrency 64 \
  --random-input-len 32 \
  --random-output-len 4096 \
  --seed 0
```

实际模型路径由评测环境提供，不写入仓库。完整 30 点结果和 27 点基线对比已分别列于本报告 5.4 和 5.5，不依赖外部文件才能阅读。

8. 参考与 AI 披露

最终实现参考 vLLM/SGLang 公开的 continuous batching、chunked prefill、paged KV cache 与 admission-control 设计思路，但未引入这些项目的源码文件。完整来源、使用边界与生成式 AI 辅助范围见附件 `REFERENCE.md`。

本项目使用 OpenAI Codex 协助代码阅读、候选方案实现、测试、日志汇总与文档整理。

9. 结论

最终提交通过 chunked prefill、KV admission 安全水位、SSE 合并和非流式完成路径优化，针对单卡 8B 的多并发、长输出场景提升服务吞吐。官方 30 点矩阵全部完成，27 个共同有效点的几何平均提升为 13.56%，关键高并发长输出点提升 75.69% 至 82.67%，同时小规模点仅 -0.62%。