PR #33593 完整报告

sgl-project/sglang

[RL] Expose top-p-only sampling masks

合并时间：2026-08-15 07:40

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/33593

执行摘要

- 一句话：支持 top-p-only 采样掩码并新增 chat 接口暴露
- 推荐动作：值得精读。它展示了如何以最小改动（3 处源码 + 3 处测试）扩展一个既有原语的请求面，并通过拆分后续 PR 控制性能风险。关注 sampling_mask_max_top_k 对批处理的影响，以及 #34037 的后续落地。

功能与动机

现有请求准入要求有限 top_k，排除了常见的 RL rollout 配置 top_k=-1 与 top_p<1，尽管 top-p 截断定义了可重放的采样支持；同时 /v1/chat/completions 未暴露该原语，而 Miles 等 RL 工作流同时使用 /generate 与 /v1/chat/completions（PR body 原话）。

实现拆解

1. 放宽 /generate 准入

在 python/sglang/srt/managers/scheduler.py 的 handle_generate_request 中，将原有的 req.sampling_params.top_k == TOP_K_ALL 拒绝条件改为计算 uses_top_k_or_top_p_truncation（top_k != TOP_K_ALL 或 top_p < 1.0），并在不满足时以新错误消息拒绝。这样 top_k=-1 配合 top_p<1 的配置可进入掩码返回链路，同时完整词汇表请求仍被拦截。

2. 新增 chat 请求 schema 字段

在 python/sglang/srt/entrypoints/openai/protocol.py 的 ChatCompletionRequest 中新增 return_sampling_mask: bool = False，默认关闭，避免改变现有请求行为。

3. 聊天请求验证与转发

在 python/sglang/srt/entrypoints/openai/serving_chat.py 中，_validate_request 增加约束：return_sampling_mask 必须伴随 return_meta_info=true，否则返回错误；_convert_to_internal_request 将 request.return_sampling_mask 转发到 GenerateReqInput.return_sampling_mask，复用既有采样掩码输出链路。

4. 测试配套

更新 test/registered/sampling/test_sampling_mask.py，新增 /generate 与 /v1/chat/completions 的 top-p-only 掩码端到端用例（_TOP_P_SMALL=1e-5 保证严格截断）

，并新增完整词汇表拒绝用例；更新 test_serving_chat.py 与 test_protocol.py 覆盖 schema 默认值、验证与转换。GPU 端到端用例注册到 CI，本地协议与 serving-chat 单元套件 135 项通过。

关键文件：

- python/sglang/srt/managers/scheduler.py（模块 调度器；类别 source；类型 core-logic；符号 handle_generate_request）：核心准入逻辑变更：将采样掩码请求从仅限有限 top_k 扩展为 top_k 有限或 top_p<1，是本次功能的关键开关。
- test/registered/sampling/test_sampling_mask.py（模块 采样掩码；类别 test；类型 test-coverage；符号 test_generate_returns_top_p_only_sampling_mask, test_chat_completions_returns_top_p_only_sampling_mask, test_generate_rejects_full_vocabulary_sampling_mask）：端到端覆盖 top-p-only /generate 与 /v1/chat/completions 的掩码返回及完整词汇表拒绝，是功能正确性的主要验证。
- python/sglang/srt/entrypoints/openai/serving_chat.py（模块 聊天服务；类别 source；类型 core-logic；符号 _validate_request, _convert_to_internal_request）：聊天端点的验证与转发逻辑，使 return_sampling_mask 在 /v1/chat/completions 生效。
- python/sglang/srt/entrypoints/openai/protocol.py（模块 协议；类别 source；类型 configuration；符号 ChatCompletionRequest）：新增 chat 请求 schema 字段 return_sampling_mask。
- test/registered/unit/entrypoints/openai/test_serving_chat.py（模块 聊天服务；类别 test；类型 test-coverage；符号 test_validate_request_rejects_sampling_mask_without_meta_info, test_convert_to_internal_request_single）：验证聊天请求的验证与转换行为。
- test/registered/unit/entrypoints/openai/test_protocol.py（模块 协议；类别 test；类型 test-coverage；符号 test_basic_chat_completion_request）：验证 schema 默认值。

关键符号：handle_generate_request, _validate_request, _convert_to_internal_request, test_generate_returns_top_p_only_sampling_mask, test_chat_completions_returns_top_p_only_sampling_mask, test_generate_rejects_full_vocabulary_sampling_mask, test_validate_request_rejects_sampling_mask_without_meta_info

关键源码片段

python/sglang/srt/managers/scheduler.py

核心准入逻辑变更：将采样掩码请求从仅限有限 top_k 扩展为 top_k 有限或 top_p<1，是本次功能的关键开关。

```
# 准入判断：只有 top_k 有限或 top_p < 1 时才允许返回采样掩码。
# top_p < 1 不是严格的大小边界，但这是准入层的可接受近似；
# 行级过滤与硬上限在 #34037 中统一加固。
uses_top_k_or_top_p_truncation = (
    req.sampling_params.top_k != TOP_K_ALL or req.sampling_params.top_p < 1.0
)
if req.return_sampling_mask and not uses_top_k_or_top_p_truncation:
```

```
    # 完整词汇表掩码会放大元数据，直接拒绝请求
    error_msg = (
        "return_sampling_mask cannot return the full vocabulary; set "
        "top_p < 1 or a finite top_k."
    )
    req.set_finish_with_abort(error_msg)
    self.init_req_max_new_tokens(req)
    self._add_request_to_queue(req)
    return
```

test/registered/sampling/test_sampling_mask.py

端到端覆盖 top-p-only /generate 与 /v1/chat/completions 的掩码返回及完整词汇表拒绝，是功能正确性的主要验证。

```
  def test_chat_completions_returns_top_p_only_sampling_mask(self):
    # 通过 /v1/chat/completions 验证 top-p-only 采样掩码返回
    response = requests.post(
        self.base_url + "/v1/chat/completions",
        json={
            "model": self.model,
            "messages": [{"role": "user", "content": "Name a capital city."}],
            "temperature": 1.0,
            "top_p": _TOP_P_SMALL, # 1e-5 保证严格截断
            "max_tokens": _MAX_NEW_TOKENS,
            "ignore_eos": True,
            "return_sampling_mask": True,
            "return_meta_info": True, # return_sampling_mask 的前置要求
            "return_token_ids": True,
        },
        timeout=60,
    )
    self.assertEqual(response.status_code, 200, response.text)

    choice = response.json()["choices"][0]
    output_ids = choice["token_ids"]
    meta_info = choice["meta_info"]
    sampling_masks = meta_info["output_token_sampling_mask"]
    sampling_logprobs = meta_info["output_token_sampling_logprobs"]

    # mask 与 token 一一对齐，且每个采样 token 都在其 mask 中
    self.assertEqual(len(output_ids), _MAX_NEW_TOKENS)
    self.assertEqual(len(sampling_masks), len(output_ids))
    self.assertEqual(len(sampling_logprobs), len(output_ids))
    for output_id, sampling_mask in zip(output_ids, sampling_masks):
        self.assertIn(output_id, sampling_mask)
```

评论区精华

JustinTong0323 在 scheduler.py 的评审中指出准入放宽的性能隐患：

> Allowing TOP_K_ALL here makes any top-p-only mask request set sampling_mask_max_top_k to TOP_K_ALL; Sampler._compute_sampling_mask_from_probs then full-sorts every row in the decode batch and materializes all kept IDs on CPU before filtering return_sampling_masks, so one opt-in request can impose full-vocabulary work on unrelated co-batched requests.

并补充 top_p < 1 不是大小边界（平坦 logits 可保留约 top_p * vocab_size，接近 1 的值可能 float32 舍入为 1.0）。作者 nanjiangwill 回应：

> the performance concern is valid...But that is a general issue with the primitive from #27408...I split the row filtering and size bound into #34037

最终 JustinTong0323 批准，确认本 PR 的准入扩展与 #34037 的通用加固方向一致。

- top-p-only 掩码的批处理性能与大小上限 (performance): nanjiangwill 认可性能问题，但认为这是 #27408 原语的一般问题，已将行过滤与大小上限拆分到 #34037；本 PR 的准入条件是必要且独立的合并单元。JustinTong0323 最终批准。

风险与影响

- 风险：
  - 性能风险（scheduler.py）：放宽准入后，top-p-only 请求可能产生接近完整词汇表的 support，Sampler._compute_sampling_mask_from_probs 会对整个 decode 批次逐行全排序并物化到 CPU，单个 opt-in 请求可能拖累同批次无关请求。
  - 正确性风险（scheduler.py）：top_p < 1 不是严格大小边界，平坦 logits 下 support 约 top_p * vocab_size，接近 1 的 top_p 可能在 float32 top_ps 张量中舍入为 1.0，从而返回完整词汇表。
  - 兼容性风险（serving_chat.py）：新增 return_sampling_mask 必须配合 return_meta_info=true 的校验，用户若未设置会得到 400 错误，属 API 行为变化，需文档同步。
  - 范围风险：/v1/completions 未支持，端点行为不一致，用户可能误用。
- 影响：
  - 用户：Miles 等 RL 工作流可在 top_k=-1, top_p<1 配置下通过 /generate 与 /v1/chat/completions 获取可重放的采样掩码与 logprobs，补齐了 RL rollout 的 replay 支持。
  - 系统：采样掩码准入从有限 top_k 扩展到 top-p 截断，可能增加部分请求的元数据计算与传输开销，但默认关闭（False），且复用既有输出链路。
  - 团队：明确了采样掩码原语（#27408）与策略层准入的职责边界，为 #34037 的行级过滤与大小上限加固提供了独立合并的前置。
  - 风险标记：性能风险：全批次全排序，准入逻辑变更，潜在完整词汇表掩码，缺少硬大小上限，新增 API 校验约束

关联脉络

- PR #27408 Add sampling mask primitive to SGLang: 引入采样掩码原语，本 PR 在其基础上扩展准入条件与聊天端点暴露。
- PR #34037 Harden sampling-mask reconstruction and support size: PR body 明确将行级过滤与硬大小上限拆分为独立工作，本 PR 依赖其后续落地。