PR #23980 完整报告

sgl-project/sglang

docs(cookbook): add H200 (FP4) deployment option for DeepSeek-V4

合并时间：2026-04-29 10:38

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/23980

执行摘要

本 PR 为 DeepSeek-V4 部署 cookbook 的命令生成器新增了 H200 (FP4) 硬件选项，允许用户直接在 Hopper GPU 上运行原始 FP4 检查点（通过 Marlin MoE 运行器）。变更同时禁用了不兼容的 cp/pd-disagg 部署模式，并在切换硬件时自动回退 recipe 为 low-latency，更新了文档描述以清晰区分 FP4 与 FP8 路径。

功能与动机

此前 DeepSeek-V4 的 H200 部署仅支持转换为 FP8 的检查点，本次 PR 扩展了部署选项，使用户能够直接在 H200 上使用 Hugging Face 上的原始 FP4 检查点，利用 Marlin 内核在运行时将 expert 权重从 FP4 即时解量化为 FP16。这提供了更多灵活性，特别是 DeepSeek-V4-Pro 模型现在可以单节点 8 GPU 部署，而 FP8 版本需要双节点 16 GPU。

实现拆解

1. 新增硬件选项：在 docs_new/src/snippets/autoregressive/deepseek-v4-deployment.jsx 的 options.hardware.items 中添加 { id: 'h200-fp4', label: 'H200 (FP4)', default: false }。
2. 定义不兼容集合：创建 H200_FP4_UNSUPPORTED_RECIPES 常量，包含 cp（上下文并行）和 pd-disagg（预填充 - 解码分离）两种 recipe，这些功能在 Marlin 路径上不可用。
3. 动态过滤 recipe 选项：重写 resolveItems 函数，增加 vals 参数。当 hardware === 'h200-fp4' 且 option.name === 'recipe' 时，将不兼容的 recipe 标记为 disabled 并附带原因展示给用户。
4. 硬件切换时自动回退：修改 handleRadioChange，当用户从其他硬件切换到 h200-fp4 且当前选中的 recipe 不兼容时，自动将 recipe 重置为 low-latency，避免界面处于无效状态。
5. 扩展命令映射表：在 COMMANDS 对象中添加 h200-fp4lsmall 和 h200-fp4lbig 条目，指定 Hugging Face 模型路径和 TP 配置（Flash: TP=4, Pro: TP=8，均为单节点），并在允许的 recipe 集合和命令构建分支中增加对应逻辑。
6. 同步文档更新：修改 docs_new/cookbook/autoregressive/DeepSeek/DeepSeek-V4.mdx 的描述、硬件表格和 Hopper 说明节，明确区分 FP4 与 FP8 两种选项。

docs_new/src/snippets/autoregressive/deepseek-v4-deployment.jsx

核心交互组件，新增 h200-fp4 硬件选项，并实现了不兼容 recipe 的禁用和自动回退逻辑。

```
// 硬件选项列表（部分），新增 h200-fp4 项
```

```
const options = {
  hardware: {
    name: "hardware",
    title: "Hardware Platform",
    items: [
      { id: "b200", label: "B200 (FP4)", default: true },
      { id: "b300", label: "B300 (FP4)", default: false },
      { id: "gb200", label: "GB200 (FP4)", default: false },
      { id: "gb300", label: "GB300 (FP4)", default: false },
      { id: "h200", label: "H200 (FP8)", default: false },
      { id: "h200-fp4", label: "H200 (FP4)", default: false }, // 新增
    ],
  },
  // ... modelSize, recipe, reasoningParser, toolcall 保持不变
};

// 不支持的 recipe 集合（cp 和 pd-disagg）
const H200_FP4_UNSUPPORTED_RECIPES = new Set(["cp", "pd-disagg"]);

// 根据当前硬件动态计算可见的 recipe 列表，禁用不支持的选项
const resolveItems = (option, vals) => {
  if (option.name === "recipe" && vals && vals.hardware === "h200-fp4") {
    // 对 H200 (FP4) 路径禁用不支持的 recipe，并附带原因提示
    return option.items.map((it) =>
      H200_FP4_UNSUPPORTED_RECIPES.has(it.id)
        ? { ...it, disabled: true, disabledReason: "Not supported on H200 (FP4)" }
        : it
    );
  }
  return option.items;
};

// 切换硬件时自动回退 recipe 到 low-latency
const handleRadioChange = (optionName, value) => {
  setValues((prev) => {
    const next = { ...prev, [optionName]: value };
    // 切换到 h200-fp4 且当前 recipe 不受支持时，回退到默认低延迟模式
    if (
      optionName === "hardware" &&
      value === "h200-fp4" &&
      H200_FP4_UNSUPPORTED_RECIPES.has(next.recipe)
    ) {
      next.recipe = "low-latency";
    }
    return next;
  });
};
```

评论区精华

无实质性审核讨论。PR 由作者 Fridge003 独立迭代完成，5 次提交逐步完善了不兼容列表、MTP 参数调整和无效标志清理。Mintlify 自动生成了文档预览部署。

风险与影响

- 前端状态一致性：resolveItems 新增 vals 参数后，若组件其他部分直接引用原始 options 而非通过该函数，可能导致禁用选项无效。当前代码中仅在 getInitialState 和（推测）选项渲染时使用，但从 patch 无法确认所有调用点均已适配。
- 缺少自动化测试：该交互逻辑没有对应前端或集成测试，后续重构可能引入回归。
- 文档误导风险：表格中新旧选项并存，用户可能混淆 FP4 与 FP8 的功能差异（如 PD-Disagg 仅在 FP8 路径可用）。
- 影响范围：仅限于 DeepSeek-V4 部署文档页面，无后端或 API 变更。

关联脉络

本 PR 与 #23943（Add single-node H200 DeepSeek-V4-Pro low-latency recipe）紧密关联，后者首次为 H200 FP8 路径添加低延迟部署方案，本 PR 进一步补充了 FP4 原始检查点路径。两者结合为用户提供了在 H200 上运行 DeepSeek-V4 的完整部署选项矩阵。