PR #27248 完整报告

sgl-project/sglang

[Doc][CPU]Update Cookbook with Xeon support info

合并时间：2026-06-06 13:39

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/27248

PR 分析报告：添加 Xeon CPU 支持文档

执行摘要

本 PR 为 cookbook 中的多个模型系列（DeepSeek、Qwen、Hunyuan 等）添加了 Intel Xeon CPU 的部署支持信息。通过更新交互式配置器（React JSX 组件）和关联的 Markdown 文档，为用户提供了 Xeon 硬件下的正确命令示例和选项引导。主要变更包括新增 XEON 硬件选项、禁用 CPU 不支持的策略、调整量化选项和命令参数（如 TP 值、CPU 特有标志）。

功能与动机

PR 描述明确："Adding Xeon support info and example commands into cookbook"。动机是为 Xeon CPU 用户提供开箱即用的部署指导，避免用户自行猜测配置，降低使用门槛。此前 cookbook 仅涵盖 GPU（NVIDIA、AMD），未覆盖 Intel CPU 平台。

实现拆解

1. 更新交互式配置器（JSX 组件）：在 docs_new/src/snippets/autoregressive/ 下的 13 个组件中新增 { id: 'xeon', label: 'XEON', default: false } 硬件选项。每个组件根据模型特性调整。
2. 禁用 CPU 不支持的策略：对 DP、EP、MTP 策略项添加 disabledWhen: (v) => v.hardware === 'xeon' 条件，并在 UI 上显示禁用原因（如"Intel Xeon CPUs only support Tensor Parallel (TP)"）。
3. 调整量化选项：对 R1 和 V3.1 组件，新增 int8 量化选项（仅 Xeon 可用），并禁用 FP4（CPU 不支持）。在 Qwen3.5 组件中，Xeon 下默认选择 BF16，禁用 FP8 推荐。
4. 修正生成命令：在 generateCommand 中添加 Xeon 分支：TP 值动态设为 6（而非 GPU 的 8），追加 --device cpu 和 --disable-overlap-schedule，并移除 --enable-symm-mem 等不适用于 CPU 的选项。
5. 优化硬件切换逻辑：在 handleRadioChange 中，当切换到 Xeon 时自动重置被禁用的选项（如取消选中 DP/EP/MTP），并自动跳转至合适的模型版本（如 DeepSeek-V3.1 切换到 INT8 变体）。
6. 更新文档页面（MDX）：同步更新了 Qwen 系列、DeepSeek 系列等对应的 Markdown 文档，移除硬件无关的警告提示，保持文档与交互式组件一致。

docs_new/src/snippets/autoregressive/deepseek-v31-deployment.jsx

为 DeepSeek-V3.1 添加 Xeon 支持，包括 INT8 模型变体、自动切换模型名称、禁用 DP/EP/MTP。

（注：片段展示了模型自动切换和命令生成逻辑，注释解释了 Xeon 下的行为。）

docs_new/src/snippets/autoregressive/deepseek-r1-basic-deployment.jsx

为 DeepSeek-R1 添加 Xeon 支持，包括量化选项中的 INT8、FP4 禁用、TP 值 6 以及命令参数。

（注：代码展示了量化动态获取、模型路径选择与 TP 值的条件分支，注释说明了 Xeon 下的不同行为。）

关键源码片段

docs_new/src/snippets/autoregressive/deepseek-v31-deployment.jsx

为 DeepSeek-V3.1 添加 Xeon 支持，包括 INT8 模型变体、自动切换模型名称、禁用 DP/EP/MTP。

```
exportconstDeepSeekV31Deployment=()=>{ constoptions={ // 硬件选项新增 XEON hardware:{ name:'hardware', title:'Hardware Platform', items:[ // ... GPU 选项 {id:'xeon',label:'XEON',default:false} ] }, // 模型名称新增 INT8 变体，标记为 xeonOnly modelname:{ name:'modelname', title:'Model Name', items:[ {id:'v31',label:'DeepSeek-V3.1',default:true}, {id:'v31terminus',label:'DeepSeek-V3.1-Terminus',default:false}, {id:'v31terminusint8',label:'DeepSeek-V3.1-Terminus-Channel-int8',default:false,xeonOnly:true} ] }, // 策略禁用同 V3 strategy:{ name:'strategy', title:'Deployment Strategy', type:'checkbox', items:[ {id:'tp',label:'TP',default:true,required:true}, {id:'dp',label:'DP attention',default:false,disabledWhen:(v)=>v.hardware==='xeon'}, {id:'ep',label:'EP',default:false,disabledWhen:(v)=>v.hardware==='xeon'}, {id:'mtp',label:'Multi-token Prediction',default:false,disabledWhen:(v)=>v.hardware==='xeon'} ] }, // ... 其余选项 }; // 硬件切换时自动选择模型：切到 Xeon 选 INT8，切出 Xeon 恢复默认 consthandleRadioChange=(optionName,value)=>{ setValues(prev=>{ constnext={...prev,[optionName]:value}; if(optionName==='hardware'){ if(next.hardware==='xeon'){ next.modelname='v31terminusint8';// 自动切到 INT8 }else{ constm=options.modelname.items.find(i=>i.id===next.modelname); if(m&&m.xeonOnly){ next.modelname=options.modelname.items.find(i=>!i.xeonOnly&&i.default)?.id||'v31'; } } // 过滤被禁止的策略 conststrategyItems=options.strategy.items||[]; constcurrent=Array.isArray(next.strategy)?next.strategy:[]; next.strategy=current.filter(id=>{ constitem=strategyItems.find(s=>s.id===id); if(!item)returnfalse; if(typeofitem.disabledWhen==='function'&&item.disabledWhen(next))returnfalse; returntrue; }); } returnnext; }); }; // 生成命令时处理模型路径和 Xeon 特定参数 constgenerateCommand=()=>{ const{hardware,modelname,strategy}=values; constisXeon=hardware==='xeon'; constmodelMap={ 'v31':'deepseek-ai/DeepSeek-V3.1', 'v31terminus':'deepseek-ai/DeepSeek-V3.1-Terminus', 'v31terminusint8':'IntervitensInc/DeepSeek-V3.1-Terminus-Channel-int8' };
```

```
constmodelName=modelMap[modelname]; letcmd='python3 -m sglang.launch_server \\n'; cmd+=` --model-path${modelName}`; if(isXeon){ cmd+=' \ --device cpu \ --disable-overlap-schedule';// CPU 特有参数 cmd+=' \ --quantization w8a8_int8';// INT8 量化 } // ... 其他策略和参数 returncmd; };
```

（注：片段展示了模型自动切换和命令生成逻辑，注释解释了 Xeon 下的行为。）

docs_new/src/snippets/autoregressive/deepseek-r1-basic-deployment.jsx

为 DeepSeek-R1 添加 Xeon 支持，包括量化选项中的 INT8、FP4 禁用、TP 值 6 以及命令参数。

```
exportconstDeepSeekR1BasicDeployment=()=>{ constoptions={ hardware:{ name:'hardware', title:'Hardware Platform', items:[ // ... GPU 项 {id:'xeon',label:'XEON',default:false}, ], }, quantization:{ name:'quantization', title:'Quantization', getDynamicItems:(values)=>{ constisXeon=values.hardware==='xeon'; constfp4Disabled=values.hardware==='h100'|| values.hardware==='mi300x'||isXeon; return[ {id:'fp8',label:'FP8',default:true}, { id:'fp4',label:'FP4',default:false, disabled:fp4Disabled, disabledReason:isXeon ?'Intel Xeon CPUs do not support FP4 quantization' :'H100 and MI300X only support FP8 quantization', }, { id:'int8',label:'INT8',default:false, disabled:!isXeon,// 仅在 Xeon 下可用 disabledReason:'INT8 is only available when XEON hardware is selected', }, ]; }, }, strategy:{ name:'strategy', title:'Deployment Strategy', type:'checkbox', items:[ {id:'tp',label:'TP',default:true,required:true}, {id:'dp',label:'DP',default:false,disabledWhen:(v)=>v.hardware==='xeon'}, {id:'ep',label:'EP',default:false,disabledWhen:(v)=>v.hardware==='xeon'}, {id:'mtp',label:'MTP',default:false,disabledWhen:(v)=>v.hardware==='xeon'}, ], }, // thinking, toolcall 保持不变 }; // ... getInitialState, useEffect 等 constgenerateCommand=(values)=>{ const{hardware,quantization,strategy}=values; constisXeon=hardware==='xeon'; // 模型路径选择：FP4、INT8、FP8 constmodelPath=quantization==='fp4' ?'nvidia/DeepSeek-R1-0528-FP4-v2' : quantization==='int8' ?'Conexis/DeepSeek-R1-0528-Channel-INT8' : 'deepseek-ai/DeepSeek-R1-0528'; letcommand='python3 -m sglang.launch_server \\n'; command+=` --model-path${modelPath}`; if(strategyValues.includes('tp')){ command+=isXeon?' \ --tp 6':' \ --tp 8';// Xeon 用 6 } // ... 其他策略 if(!isXeon){ command+=' \ --enable-symm-mem # Optional: improves performance, but may be unstable'; } if(isXeon){ command+=' \ --device cpu \ --disable-overlap-schedule';// CPU 特有 } // ... thinking, toolcall 参数 returncommand; };
```

（注：代码展示了量化动态获取、模型路径选择与 TP 值的条件分支，注释说明了 Xeon 下的不同行为。）

评论区精华

review 由 zijiexia 主导，主要讨论了以下问题：

- DeepSeek-V3 TP 值：指出在 Xeon 上仍输出 --tp 8，应改为 --tp 6。“The TP line just below … emits --tp 8 for every hardware, including Xeon.” 作者已修复。

- DeepSeek-V3 FP4 处理："FP4 isn't grayed out — as R1 does, FP4 should be disabled on Xeon instead of letting the user select it and then erroring." 作者改为禁用并自动回落。
- DeepSeek-V3.1 模型选项："when Xeon is selected, v31 (DeepSeek-V3.1) and v31terminus stay selectable and generate an FP8-native command without --quantization w8a8_int8." 作者解释 Xeon 支持 FP8，保持可选。reviewer 同意。
- Qwen3-Next FP8 一致性："Here Qwen3-Next sets fp8: false for Xeon, but Qwen3-VL / Qwen3.5 / Qwen3-Coder all enableFP8 for Xeon." 作者修正对齐。
  - gemini-code-assist[bot] 提出了自动化建议（切换硬件时自动重置量化），部分被采纳。

风险与影响

风险：主要是文档正确性风险。若命令参数错误（如 TP 值、缺失 CPU 标志），用户部署将失败或性能不佳。review 中已修复几个关键问题，但仍需用户自行验证命令。由于是文档 PR，不影响代码路径，系统风险低。

影响：

- 用户：Xeon 用户获得清晰的部署指南，减少试错成本。GPU 用户不受影响（Xeon 选项默认隐藏）。
- 团队：新增的 Xeon 选项需在后续迭代中保持一致，避免碎片化。
- 系统：无运行影响。

关联脉络

无直接关联的 Issue 或历史 PR。但此 PR 与近期 cookbook 相关的更新（如 #26733 Nemotron 性能优化）无关，独立推进了 CPU 平台的支持。未来可能需为更多模型添加 Xeon 选项，或统一 Quantization 选型标准。