PR #5898 完整报告

verl-project/verl

[model] feat: support qwen35 mtp sft/rl

合并时间：2026-04-24 10:43

原文链接：http://prhub.com.cn/verl-project/verl/pull/5898

执行摘要

- 一句话：支持 Qwen3.5 MTP 模型在 Megatron 引擎的 SFT/RL 训练
- 推荐动作：### 建议
- 值得精读：如果团队计划支持更多具备 MTP 能力的模型，本 PR 中 _get_mtp_num_layers 和 _set_mtp_num_layers 的设计模式值得参考。
- 需跟进：
  1. 将辅助函数公开化，以便跨模块复用（来自 review 高优先级建议）。
  2. 修复 config_converter.py 中 MTP 检测逻辑，复用公共函数并正确处理 override。
- 测试：建议为 megatron_utils.py 中的 MTP 函数编写单元测试，覆盖三种配置格式。

功能与动机

Qwen3.5 模型引入了 Multi-Token Prediction（MTP）特性，但现有 verl 框架仅支持 DeepSeek/Qwen3 风格的 num_nextn_predict_layers 字段，无法识别 Qwen3.5 的 mtp_num_hidden_layers 及其嵌套在 text_config 中的情况。为使 SFT 和 RL 训练能够利用 MTP 能力（包括训练时辅助损失和推理时投机解码），需要扩展配置解析和模型注册逻辑。

实现拆解

实现拆解

1. 新增通用 MTP 层数操作函数（verl/utils/megatron_utils.py） - 新增 _get_mtp_num_layers(hf_config) 函数，统一处理三种配置格式：num_nextn_predict_layers（DeepSeek/Qwen3 风格）、mtp_num_hidden_layers（Qwen3.5 风格，直接位于 hf_config）、以及 mtp_num_hidden_layers 嵌套在 hf_config.text_config 中。 - 新增 _set_mtp_num_layers(hf_config, value) 函数，根据 hf_config 实际拥有的属性名设置 MTP 层数，保证写操作兼容性。 - 重构原有的 check_mtp_config 函数，内部调用上述两个工具函数替代硬编码的属性访问，消除重复逻辑。
2. 配置转换器扩展（verl/models/mcore/config_converter.py） - 在 hf_to_mcore_config_qwen3moe 函数末尾增加 MTP 支持：从 hf_config（或 text_config）中读取 mtp_num_hidden_layers 和 mtp_loss_scaling_factor，设置到生成的 TransformerConfig 的 mtp_num_layers 和 mtp_loss_scaling_factor 字段。 - 注意：当前实现未复用 _get_mtp_num_layers，且硬编码了默认的 loss scaling factor（0.1），可

能忽略用户通过 override 传入的值。

3. 模型注册表扩展（verl/models/mcore/registry.py） - 在 SupportedModel 枚举中新增 QWEN3_5_MOE="Qwen3_5MoeForCausalLM"。-在配置转换器、初始化器、前向函数、融合前向函数、权重转换器等所有注册表中，将 QWEN3_5_MOE 映射到与 QWEN3_MOE 相同的处理逻辑。
4. 模型配置初始化优化（verl/workers/config/model.py） - 在 __post_init__ 中当 MTP 禁用时，同时清空可能存在的三种 MTP 属性（num_nextn_predict_layers、mtp_num_hidden_layers、text_config 中的 mtp_num_hidden_layers），避免下游模块需要单独处理。
5. 示例脚本和 Shell 配置（examples/sft/gsm8k/run_qwen3_5_megatron.sh 和 verl/experimental/fully_async_policy/shell/grpo_qwen35_35b_megatron_async.sh） - 修改 SFT 示例脚本，添加 MTP 相关命令行参数（enable/enable_train/detach_encoder/loss_scaling_factor）。 - 新增 GRPO+ 完全异步策略的 Shell 脚本，展示 Qwen3.5-35B-A3B 的 MTP 配置用法，包含 TP/PP/EP 并行度建议。
6. 废弃模块清理（verl/workers/megatron_workers.py 的修改在最终提交中被回退或移除） - 根据 review 意见，不修改已废弃的 megatron_workers.py。

关键文件：

- verl/utils/megatron_utils.py（模块 工具层；类别 source；类型 core-logic；符号 _get_mtp_num_layers, _set_mtp_num_layers）：核心逻辑：新增 MTP 通用查询 / 设置函数，重构 check_mtp_config，是后续所有 MTP 配置的基础。
- verl/models/mcore/config_converter.py（模块 模型配置；类别 source；类型 data-contract）：配置转换：为 Qwen3 MoE 配置转换器添加 MTP 参数传递，是模型启动的关键路径。
- verl/workers/config/model.py（模块 工作节点；类别 source；类型 data-contract）：配置初始化：在 __post_init__中统一清空禁用 MTP 时的所有字段，减少下游适配负担。
- verl/experimental/fully_async_policy/shell/grpo_qwen35_35b_megatron_async.sh（模块 实验性；类别 other；类型 core-logic）：参考配置：提供 Qwen3.5 MTP 在 GRPO+ 完全异步模式下的完整 Shell 脚本，含 MTP 参数和环境要求。
- verl/models/mcore/registry.py（模块 模型配置；类别 source；类型 data-contract）：模型注册：新增 QWEN3_5_MOE 枚举及其到所有注册表的映射。

关键符号：_get_mtp_num_layers, _set_mtp_num_layers, check_mtp_config

关键源码片段

verl/workers/config/model.py

配置初始化：在 __post_init__中统一清空禁用 MTP 时的所有字段，减少下游适配负担。

```
# 在 __post_init__ 方法中，位于 per model patch 之后
# When MTP is disabled, zero out MTP layer counts from hf_config so that
# downstream engine/worker code does not need to handle each MTP field format
# individually.
if not self.mtp.enable:
```

```
    if hasattr(self.hf_config, "num_nextn_predict_layers"):
        self.hf_config.num_nextn_predict_layers = 0
    if hasattr(self.hf_config, "mtp_num_hidden_layers"):
        self.hf_config.mtp_num_hidden_layers = 0
    if hasattr(self.hf_config, "text_config") and hasattr(self.hf_config.text_config, "mtp_num_
    hidden_layers"):
        self.hf_config.text_config.mtp_num_hidden_layers = 0
```

评论区精华

评论区精华

> gemini-code-assist[bot]: "The MTP configuration helpers _get_mtp_num_layers and _set_mtp_num_layers are useful across different modules... They should be made public by removing the leading underscore." （高优先级） - 建议：将两个辅助函数改为公开（get_mtp_num_layers / set_mtp_num_layers）以便跨模块复用。
>
> gemini-code-assist[bot]: "The current implementation has two issues:

1. It duplicates the MTP layer detection logic and is less comprehensive than the get_mtp_num_layers helper (e.g., it misses num_nextn_predict_layers). 2. It potentially ignores user overrides for mtp_loss_scaling_factor." （高优先级）

> wuxibin89: "Please reuse examples/sft/gsm8k/run_qwen3_5_megatron.sh with additional MTP option." - 建议：SFT 无需独立 MTP 脚本，应在原脚本中增加参数。
>
> wuxibin89: "megatron_workers.py has been deprecated, please do not modify it." - 约束：不应修改已废弃的 megatron_workers.py。

结论：开发者在最终提交中采纳了关于不修改废弃文件的建议，并合并了 SFT 脚本。但公开函数命名的重构建议和 config_converter 中的重复逻辑问题未被完全修复。

- MTP 辅助函数应公开化 (design): 未采纳。最终提交保持了下划线前缀。ArronHZG 要求遵循建议，但未强制修改。
- config_converter 中 MTP 逻辑重复且不完整 (correctness): 未采纳。最终提交的代码仍存在这两个问题。
- SFT 脚本整合建议 (design): 已采纳。在最终提交中合并在 examples/sft/gsm8k/run_qwen3_5_megatron.sh 中。
- 不应修改已废弃的 megatron_workers.py (other): 已采纳。最终提交回退了相关修改。

风险与影响

- 风险：### 风险分析

1. 配置兼容性风险（verl/models/mcore/config_converter.py） - hf_to_mcore_config_qwen3moe 中 MTP 检测未覆盖 num_nextn_predict_layers，若 Qwen3.5 未来版本采用该字段名将无法识别。 - mtp_loss_scaling_factor 硬编码为 0.1，若用户通过 override 指定则被忽略，可能导致训练损失计算错误。

2. 废弃模块污染（verl/workers/megatron_workers.py） - 虽然最终提交回退了修改，但中途的修改可能残留不一致。审查确认最终版本无异。
3. 测试覆盖缺失 - 本次改动涉及多个核心模块（配置转换、模型注册、utils），但无对应的单元测试或集成测试。建议补充测试覆盖 MTP 配置的读写场景。
4. API 破坏风险 - 新增 supportedModel.QWEN3_5_MOE 枚举值，但未删除任何枚举，无破坏性变更。 - _get_mtp_num_layers 和 _set_mtp_num_layers 以下划线开头（内部），理论上不会影响外部 API。但 review 建议公开化，若未来修改命名则影响内部调用。 - 影响：### 影响分析

- 用户影响：需要升级 transformers 至 5.3.0+，并合并 mbridge PR #98 才能使用 MTP。新增的 Shell 脚本提供了开箱即用的配置。SFT 脚本通过额外参数支持 MTP，不影响已有用法。
- 系统影响：MTP 配置仅在启用时激活（mtp.enable=True），默认关闭，不增加现有训练的额外开销。配置转换器中的 MTP 逻辑仅在 Qwen3.5 模型上生效。
- 团队影响：代码主要改动集中在 megatron_utils.py、config_converter.py、registry.py，构成了未来支持更多 MTP 模型的参考模式。review 建议部分未完全落实（如公开函数命名），需要后续跟进。
- 风险标记：配置兼容性风险，忽略用户覆盖参数，缺少测试覆盖

关联脉络

- PR #6072 [veomni] feat: enable VeOmni engine for on-policy distillation: 同属模型引擎扩展，均为训练后端增加新模型架构支持。
- PR #6067 [BREAKING] [misc] refactor: deprecate workers, migrate to engines: PR 对 megatron_workers.py 的修改被 review 阻止，与 workers 废弃重构相关。