

# PR #2583 完整报告

radixark/miles

docs(diffusion): latest doc ported from miles diffusion 20260818-01:00

合并时间: 2026-08-18 16:44

原文链接: <http://prhub.com.cn/radixark/miles/pull/2583>

## 执行摘要

- 一句话: 主站新增 Diffusion 文档分区, 同步 miles\_diffusion 全套指南
- 推荐动作: 值得快速浏览而非精读: 这是一次高质量的文档体系扩展, 单页内容密度高。建议重点阅读 customization.md (了解 miles-diffusion 的 `--*-path` 扩展架构, 对要定制 diffusion 训练的用户是必读入口) 和 launch-script.md (理解批量大小算术与各 flag 分组)。对文档维护者而言, commit c346dee 揭示的 docs.json 格式所有权冲突是本次最值得记住的工程细节——编辑该文件前先确认 sync-example-docs hook 的约定。

## 功能与动机

PR body 明确了这是一次周期性文档移植: 从 radixark/miles\_diffusion 的 docs/ 目录 diff 到 Miles 的 docs/diffusion/, 并重写所有跨页链接为 Mintlify /diffusion/... 形式。提交 794977c 进一步说明动机是让主仓库内嵌的 Diffusion 文档与源仓库保持一致。此前 PR#2599 只是把用户指向独立仓库, 本次则直接在主文档站内建立完整分区, 降低用户跨仓库获取信息的成本。

## 实现拆解

实现过程按同步管道、内容落地、导航接入、格式修复、措辞收敛五步展开:

1. 同步管道与触发方式: PR body 描述了半自动移植流程——从 radixark/miles 的 main 创建分支, diff miles\_diffusion 的 docs/ 与 Miles 的 docs/diffusion/, 应用 diff 后把每个跨页链接改写为 Mintlify 的 /diffusion/... 形式并逐一验证, 最后以 draft PR 提交。此次为流程的第二次运行 (commit 4cacdf6 即 20260817-00:36 的移植, 最终 PR 标题为 20260818-01:00 版本)。
2. 内容落地 (docs/diffusion/ 下 30 个新页面): 按五组组织——Getting Started (installation、quick-start)、User Guide (cli-reference、concepts、customization、launch-script、recipe-verification、rewards)、Models (cosmos3、h3、ltx2、qwen-image、sd3、wan2-2)、Advanced (deterministic、dtype-control、lora、sde-backend、single-prompt-multi-gen、streaming-reward)、Developer (contributor-guide)。其中 customization.md (326 行) 系统性列出所有 `--*-path` 扩展钩子, launch-script.md (264 行) 拆解 ScriptArgs 机制和批量大小算术, cli-reference.md (388 行) 按子系统覆盖全部 flag, sd3.md (259 行) 给出首个已验证 recipe 的完整配置。

3. 导航接入 (docs/docs.json) : 在 Developer Guide 与 Resources 两个 Tab 之间插入 Diffusion Tab, pages 按五组 grouping 引用上述页面, 新增 52 行。这是本次唯一的结构性 config 变更。
4. 格式所有权修复 (commit c346dee) : 分支最初将 docs.json 重排为 2 空格缩进, 导致 sync-example-docs hook 每次运行都会重写该文件并报出 modification。维护者 Zhichenzzz 重新运行 hook 恢复 main 的 1 空格缩进, 使 docs.json 相对 main 的 diff 从 439 增 / 387 删 缩小到 52 增 / 0 删, 且导航内容结构不变。
5. 措辞收敛 (后续 7 个 commit) : 配合 Cursor agent 对 overview 页做简化与澄清——恢复 rollout determinism 归属、调整 LoRA 能力定位、保持与源仓库逐字节对齐。纯文档改动, 无测试、无 schema、无部署配套。

关键文件:

- docs/diffusion/user-guide/customization.md (模块 扩展架构; 类别 docs; 类型 documentation; 符号 generate\_rollout, CustomDataSource, init, get\_samples) : 系统性列出 miles-diffusion 的全部 `--*-path` 扩展钩子 (rollout、reward、filtering、training、logging、model family 六大类), 是理解其可扩展架构的核心文档, 信号强度最高。
- docs/diffusion/user-guide/launch-script.md (模块 启动脚本; 类别 docs; 类型 documentation; 符号 ScriptArgs, prepare, execute, main) : 解释 launch script 的三层结构 (ScriptArgs、prepare、execute、main) 与批量大小算术, 是理解 recipe 运行机制的入口。
- docs/diffusion/user-guide/cli-reference.md (模块 CLI 参考; 类别 docs; 类型 documentation) : train\_diffusion.py 全部 flag 的按子系统参考, 覆盖 cluster、training backend、optimizer、diffusion sampling 等, 是用户查询配置的最全索引。
- docs/docs.json (模块 文档导航; 类别 config; 类型 configuration) : 唯一的 config 变更文件, 新增 Diffusion Tab 将 30 个页面接入 Mintlify 导航, 并因 sync-example-docs 格式所有权产生一次修复提交。
- docs/diffusion/models/sd3/sd3.md (模块 模型指南; 类别 docs; 类型 documentation; 符号 SD3TrainPipelineConfig) : 首个经完整验证的 recipe 文档, 覆盖 SD3.5-Medium 的 Flow-GRPO 与 DiffusionNFT 双算法、LoRA 权重同步、fp16 精度注意与参考结果。
- docs/diffusion/developer/contributor-guide.md (模块 贡献指南; 类别 docs; 类型 documentation) : 说明 miles\_diffusion 的仓库布局、四层测试体系、CI 标签注册约定与 PR 规范, 对要为主仓库贡献 diffusion 相关代码的开发者是必读。
- docs/diffusion/getting-started/installation.md (模块 安装指南; 类别 docs; 类型 documentation) : 说明 Docker 推荐安装、CUDA 12.9 镜像约束、sglang main 跟踪策略与裸金属安装警告, 是环境搭建的第一入口。

关键符号: generate\_rollout, CustomDataSource, custom\_generate, ScriptArgs, prepare, execute, main, SD3TrainPipelineConfig, resolve\_diffusion\_model\_family, execute\_train

关键源码片段

<docs/diffusion/user-guide/customization.md>

系统性列出 miles-diffusion 的全部 `--path` 扩展钩子（rollout、reward、filtering、training、logging、model family 六大类），是理解其可扩展架构的核心文档，信号强度最高。

```
# miles-diffusion 的扩展点：几乎所有行为都能用 --path 钩子替换，
# 通过 miles.utils.misc.load_function 按点分路径加载。
# 下面是 Rollout 阶段最核心的数据源接口，替代默认的 RolloutDataSourceWithBuffer。
```

```
class CustomDataSource:
    def __init__(self, args):
        # 接收解析后的完整 args，可在此读取 --data-dir 等路径类配置
        ...

    def get_samples(self, num_samples) -> list[list[Sample]]:
        # 每次 rollout 前取样本，返回按 prompt 分组的样本列表
        ...

    def add_samples(self, samples) -> None:
        # 训练产出后回填带 reward 的样本，供后续去重 / 过滤策略使用
        ...

    def save(self, rollout_id) -> None:
        # 按 rollout_id 持久化 buffer，断点续跑时可恢复现场
        ...

    def load(self, rollout_id=None) -> None:
        ...

# 替换单个 microgroup 的生成逻辑，默认实现是 generate_microgroup。
async def custom_generate(
    args, microgroup: list[Sample], sampling_params: dict, *, evaluation: bool = False
) -> list[Sample]:
    # evaluation 是可选关键字参数：若你的签名省略它，调用方会自动跳过该 kwarg，
    # 因此自定义函数不必强制接收这个参数。
    ...
```

## docs/docs.json

唯一的 config 变更文件，新增 Diffusion Tab 将 30 个页面接入 Mintlify 导航，并因 sync-example-docs 格式所有权产生一次修复提交。

```
// 新增的 Diffusion 导航 Tab：挂在 Developer Guide 与 Resources 之间。
// 该文件由 sync-example-docs hook 持有格式所有权（缩进固定为 1 空格），
// 手工编辑后应重跑 hook 避免格式噪音。
{
  "tab": "Diffusion",
  "pages": [
    "diffusion/index",
    {
      "group": "Getting Started",
```

```

    "pages": [
      "diffusion/getting-started/installation",
      "diffusion/getting-started/quick-start"
    ]
  },
  {
    "group": "User Guide",
    "pages": [
      "diffusion/user-guide/cli-reference",
      "diffusion/user-guide/concepts",
      "diffusion/user-guide/customization",
      "diffusion/user-guide/launch-script",
      "diffusion/user-guide/recipe-verification",
      "diffusion/user-guide/rewards"
    ]
  },
  {
    "group": "Models",
    "pages": [
      "diffusion/models/cosmos/cosmos3",
      "diffusion/models/h3/h3",
      "diffusion/models/ltx/ltx2",
      "diffusion/models/qwen-image/qwen-image",
      "diffusion/models/sd3/sd3",
      "diffusion/models/wan/wan2-2"
    ]
  },
  {
    "group": "Advanced",
    "pages": [
      "diffusion/advanced/deterministic",
      "diffusion/advanced/dtype-control",
      "diffusion/advanced/lora",
      "diffusion/advanced/sde-backend",
      "diffusion/advanced/single-prompt-multi-gen",
      "diffusion/advanced/streaming-reward"
    ]
  },
  {
    "group": "Developer",
    "pages": [
      "diffusion/developer/contributor-guide"
    ]
  }
]
}

```

评论区精华

本 PR 无实质 review 讨论：claude[bot] 因 fork 来源自动跳过审查，Zhichenzzz 直接批准。最有价值的工程信息来自提交历史而非评论——commit c346dee 详细记录了 sync-example-docs hook 与 docs.json 格式所有权的冲突：该 hook 拥有 docs.json 的格式所有权（缩进固定为 1 空格），分支手工重排为 2 空格导致 hook 每次运行都报 diff，最终以重跑 hook 本身作为修复方式，并强调导航内容本就正确、只是缩进恢复。这为后续编辑 docs.json 的开发者提供了明确的工具约定。

- sync-example-docs hook 与 docs.json 格式所有权 (other): docs.json 的格式由 sync-example-docs 工具持有所有权，人工编排该文件时应直接以 hook 输出为准，避免引入格式噪音。

## 风险与影响

- 风险：纯文档变更，无运行时代码风险。主要风险集中在文档生命周期管理：
  1. 内容漂移风险（中）：docs/diffusion/ 的内容来自独立仓库 miles\_diffusion，若后续源仓库更新而主仓库未同步，会出现双源不一致。当前同步流程仍需人工触发，PR body 显示为周期性执行但未提及自动化调度。
  2. 死链风险（低）：文档中大量链接被重写为 /diffusion/... 形式，且页面间存在交叉引用（如 quick-start 指向 sd3、launch-script 指向 cli-reference），PR body 虽声明逐一验证，但源仓库新增页面时若未纳入同步范围会出现悬空链接。
  3. docs.json 格式冲突复发风险（低）：任何后续 PR 手工编辑 docs/docs.json 都可能再次触发 sync-example-docs hook 的格式改写，需以 hook 输出为准或直接由工具重写。
  4. 符号归属混淆风险（低）：文档中出现的 generate\_rollout、CustomDataSource、ScriptArgs 等符号均属于 miles\_diffusion 仓库，主仓库并无对应代码，用户若在主仓库搜索这些符号会落空，文档应在显眼处标注来源仓库。- 影响：影响范围集中在文档体系与团队协作流程：
- 对用户（高影响）：docs.miles 主站首次提供完整的 Diffusion 后训练文档分区，图像 / 视频生成模型的 Flow-GRPO 与 DiffusionNFT 训练有了一站式入口，覆盖从安装、quick-start 到模型 recipe、CLI 全量参考与扩展开发。此前用户需跳转独立仓库，现在可直接在主站检索。
- 对团队（中影响）：建立了主仓库与 miles\_diffusion 的文档同步管道，fork PR + 维护者修复格式的协作为后续定期移植沉淀了流程经验，sync-example-docs 的所有权约定成为团队共识。
- 对系统（无影响）：无任何 Python 源码、CI 或部署配置变更，验证门槛低。
- 风险标记：内容双源漂移风险，docs.json 格式所有权冲突，链接重写依赖人工验证，API 符号指向独立仓库

## 关联脉络

- PR #2599 docs: point at Miles-diffusion for diffusion post-training: 同一条产品线文档策略的演进：2599 仅将用户指向独立仓库，本 PR 则在主仓库 docs/diffusion/ 内建完整分区，是扩散后训练文档战略的落地延续。

- PR #2565 docs: fold Welcome into the User Guide tab and lead with it: 同为 docs/docs.json 的导航结构重构，本 PR 在该导航基础上新增 Diffusion Tab，两者共同塑造主文档站的最终信息架构。