

13. π_0 模型

13.1 简介

π_0 (Pi-Zero) 是由 Physical Intelligence 团队开发的视觉-语言-动作 (Vision-Language-Action, VLA) 模型, 专为通用机器人控制设计。它结合了大规模预训练和基于流匹配 (Flow Matching) 的动作生成方法, 使机器人能够在多种形态下执行灵巧的操控任务。

与传统的机器人策略不同, π_0 使用流匹配来生成平滑的、实时的动作轨迹 (50Hz), 使其在现实世界中具有高度的效率、精度和适应性。流匹配算法最初用于连续正态流 (CNF) 与扩散模型中的高质量采样, 在 π_0 中, 去噪过程也是从随机噪声开始, 逐步生成一系列合理的电机动作。

13.2 显卡要求

要运行本仓库中的模型, 你需要配备至少满足以下规格的 NVIDIA GPU。这些估计假设使用单个 GPU, 但你也可以通过模型并行的方式, 利用多个 GPU 来降低单个 GPU 的内存需求, 具体可在训练配置中设置 `FSDP_DEVICES`。请注意, 当前的训练脚本尚未支持多节点训练。

模式	显存下限	参考显卡
Inference	> 8 GB	RTX 3090/4090
Fine-Tuning (LoRA)	> 22.5 GB	RTX 3090/4090
Fine-Tuning (Full)	> 70 GB	A100 (80GB) / H100

13.3 环境配置

联系售后获取适配的 `openpi` 压缩包, 解压后终端进入文件夹目录:

```
cd openpi
```

安装基本依赖:

```

sudo apt install python3-venv clang
python3 -m pip install --user pipx
pipx install uv -i http://mirrors.aliyun.com/pypi/simple
# 建议配置终端代理，否则网络问题容易导致安装失败
# export {HTTP_PROXY,HTTPS_PROXY,ALL_PROXY,http_proxy,https_proxy,all_proxy}
# =http://127.0.0.1:7890
# 注意由于uv会对某些包进行编译，因此不要使用conda环境
GIT_LFS_SKIP_SMUDGE=1 uv sync
GIT_LFS_SKIP_SMUDGE=1 uv pip install -e .

```

安装AIRBOT Play Python SDK（终端必须位于 `openpi` 路径下）：

```
uv pip install /path/to/your/airbot_py-5.1.4-py3-none-any.whl
```

安装推理依赖（数据采集程序及使用说明可联系售后获取）：

```

sudo apt-get install -y libturbojpeg gcc python3-dev v4l-utils
uv pip install -e /path/to/your/data-collection/package"[all]" -i https://
pypi.mirrors.ustc.edu.cn/simple

```

13.4 微调

13.4.1 准备数据

将采集的数据文件夹 `data` 转移到 `openpi` 根目录下。每个任务采集的数据一般会保存在单独一个以任务名命名的文件夹中，而在数据转换程序总允许同一个任务的不同的数据保存在不同的文件夹中，便于支持不同采集地点、不同采集时间的数据区分。假设任务名为 `1-1-example`，在 `openpi` 目录下创建 下目录结构：

```
mkdir -p data/1-1-example/station0
```

然后将采集的数据文件夹中的全部 `.mcap` 文件移动到 `station0` 文件夹中：

```
cp path/to/your/data/1-1-example/*.mcap data/1-1-example/station0
```

13.4.2 配置训练参数

示例配置文件位于 `examples/airbot`，包括 `config_1-1_example.py`（单臂任务）、`config_ptk_example.py`（双臂任务）、`config_mmk_example.py` 等，复制对应的配置文件到数据文件夹中并重命名为 `config.py`，例 复制到 `data/1-1-example/config.py`，根据数据情况修改其中的参数，其中 `TASK_NAME` 需与数据中的任务名一致，`FOLDERS` 需与数据文件夹一致。其他参数说明可参考注释。

13.4.3 将 AIRBOT Mcap 数据格式转换为 LeRobot 数据格式

指定要转换的数据文件夹路径进行转换, 例 前述 1-1-example :

```
uv run examples/airbot/convert_mcap_data_to_lerobot.py --data-dir data/1-1-example
```

运行完成后, 转换后的数据将保存到 下目录中:

```
~/.cache/huggingface/lerobot/
```

注意, `config.py` 中的 `TASK_NAME` 与采集数据时使用的不同时, 数据转换将无法成功。

13.4.4 计算数据统计学信息

训练时数据归一化需要用到数据的统计学信息, 执行 下命令进行计算:

```
CUDA_VISIBLE_DEVICES=0 uv run examples/airbot/compute_norm_stats.py --config-path data/1-1-example/config.py
```

生成的统计信息会保存在 `assets` 目录中。

13.4.5 模型训练

执行 下命令进行模型训练:

```
XLA_PYTHON_CLIENT_MEM_FRACTION=0.9 uv run examples/airbot/airbot_train.py --config-path data/1-1-example/
```

其中:- `XLA_PYTHON_CLIENT_MEM_FRACTION=0.9`: 配置 JAX 使用最多 90% 的GPU显存(默认只用 75%)- `--config-path`: 配置文件路径或所在目录

日志将实时打印到终端, 并保存在 `checkpoints` 目录中。程序执行后会提示使用 `wandb` 实时监控训练过程, 根据提示注册登录即可(注意选择3不可视化会报错;网站登陆可能需要科学上网)。

13.5 推理

按 下命令启动两条机械臂(确保完成了机械臂的绑定, 可参考数据采集文档):

```
airbot_server -i can_left -p 50051  
airbot_server -i can_right -p 50053
```

对于单臂任务, 只执行其中一个命令即可。

运行推理脚本(请确保没有其他程序占用相机, 例 数据采集相关程序等; 确保终端没有设置代理, 否则可能无法连接到机械臂, 可通过 `unset HTTP_PROXY HTTPS_PROXY ALL_PROXY http_proxy`

https_proxy all_proxy 命令清除代理)：

- 单臂任务：

```
uv run examples/airbot/airbot_inference_sync.py policy-config:local-policy-config \
  --policy-config.config-path data/1-1-example \
  --policy-config.checkpoint-dir checkpoints/1-1-example/9000 \
  --robot-config.robot_groups "" \
  --robot-config.robot_ports 50051 \
  --robot-config.camera-index 2 4 \
  --reset-action 0 0 0 0 0 0 0
```

- 双臂任务：

```
uv run examples/airbot/airbot_inference_sync.py policy-config:local-policy-config \
  --policy-config.config-path data/ptk_example \
  --policy-config.checkpoint-dir checkpoints/ptk_example/9000 \
  --robot-config.camera-index 2 4 6 \
  --reset-action 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

其中：

- robot_ports：机器人的端口号，根据实际启动的情况修改
- checkpoint-dir：权重文件路径，根据实际路径修改
- camera-index：相机的序号，顺序需按照：环境相机、左臂相机、右臂相机
- reset-action：机器人的初始位置，根据实际数据采集时的初始位置指定

等待推理脚本启动，启动完成后终端提示 Press 'Enter' to continue or 'q' and 'Enter' to quit...，然后按回车开始推理。

推理结束后，可按 q 然后按回车键退出。