

XTuner大模型单卡低成本微调实战

官方文档 [tutorial/xtuner \(github.com\)](https://github.com/tutorail/xtuner)

视频讲解 [XTuner 大模型单卡低成本微调实战](#)

LLM 的下游应用中，增量预训练和指令跟随是经常会用到两种的微调模式

- 增量预训练微调

- 使用场景:让基座模型学习到一些新知识，如某个垂类领域的常识
- 训练数据:文章、书籍、代码等

- 指令跟随微调

- 使用场景:让模型学会对话模板，根据人类指令进行对话
- 训练数据:高质量的对话、问答数据

指令跟随微调

在实际对话时，通常会有三种角色

- **System**: 给定一些上下文信息，比如“你是一个安全的 AI 助手”
- **User**: 实际用户，会提出一些问题，比如“世界第一高峰是？”
- **Assistant**: 根据 User 的输入，结合 System 的上下文信息，做出回答，比如“珠穆朗玛峰”。



指令跟随微调

OpenMMLab bilibili

Input : 世界第一高峰是？

Output: 珠穆朗玛峰

不同于增量预训练微调，数据中会有 Input 和 Output 希望模型学会的是答案(Output)，而不是问题(Input) 训练时只会对答案(Output)部分计算 Loss

训练时，会和推理时保持一致，对数据添加相应的对话模板，以下为 InternLM 的训练数据和标签

data	<s>	< User >	世	界	第	一	高	峰	是	?	<eoh>	\n	< Bot >	珠	穆	朗	玛	峰	<eoq>	</s>
label														珠	穆	朗	玛	峰	<eoq>	</s>

就只需要简单理解这些原理即可

增量预训练微调

没有问答的形式，即system和user两个角色留空

增量预训练微调

OpenMMLab bilibili

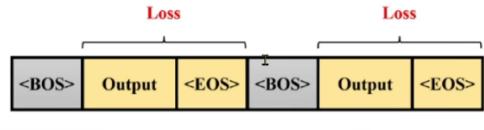
Output: 世界第一高峰是珠穆朗玛峰

为了让 LLM 知道什么时候开始一段话，什么时候结束一段话，实际训练时需要对数据添加起始符 (BOS) 和结束符 (EOS)；大多数的模型都是使用 <s> 作为起始符，</s> 作为结束符

<s>世界第一高峰是珠穆朗玛峰</s>

训练 LLM 时，为了让模型学会“世界第一高峰是珠穆朗玛峰”，并知道何时停止，对应的训练数据以及标签如下所示

```
"system": "",  
"input": "",  
"output": "I am an artificial intelligence.  
I can answer your question.  
<s>世界第一高峰是珠穆朗玛峰</s>  
The peak is Mount Qomolangma."}
```



data	<s>	世	界	第	一	高	峰	是	珠	穆	朗	玛	峰	</s>
label	世	界	第	一	高	峰	是	珠	穆	朗	玛	峰	</s>	

LoRA & QLoRA

QLoRA 是 LoRA 的一种改进

QLoRA 以4-bit量化的方式加载模型，同岁来讲就是不那么精确地来加载模型，可以极大减小显存开销

LoRA & QLoRA

LoRA: LOW-RANK ADAPTATION OF LARGE LANGUAGE MODELS

LLM 的参数量主要集中在模型中的 Linear，训练这些参数会耗费大量的显存

LoRA 通过在原本的 Linear 旁，新增一个支路，包含两个连续的小 Linear，新增的这个支路通常叫做 Adapter

Adapter 参数量远小于原本的 Linear，能大幅降低训练的显存消耗

微调原理

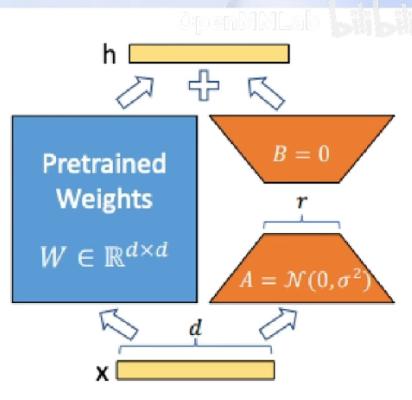
想象一下，你有一个超大的玩具，现在你想改造这个超大的玩具。但是，对整个玩具进行全面的改动会非常昂贵。

* 因此，你找到了一种叫 LoRA 的方法：只对玩具中的某些零件进行改动，而不是对整个玩具进行全面改动。

* 而 QLoRA 是 LoRA 的一种改进：如果你手里只有一把生锈的螺丝刀，也能改造你的玩具。

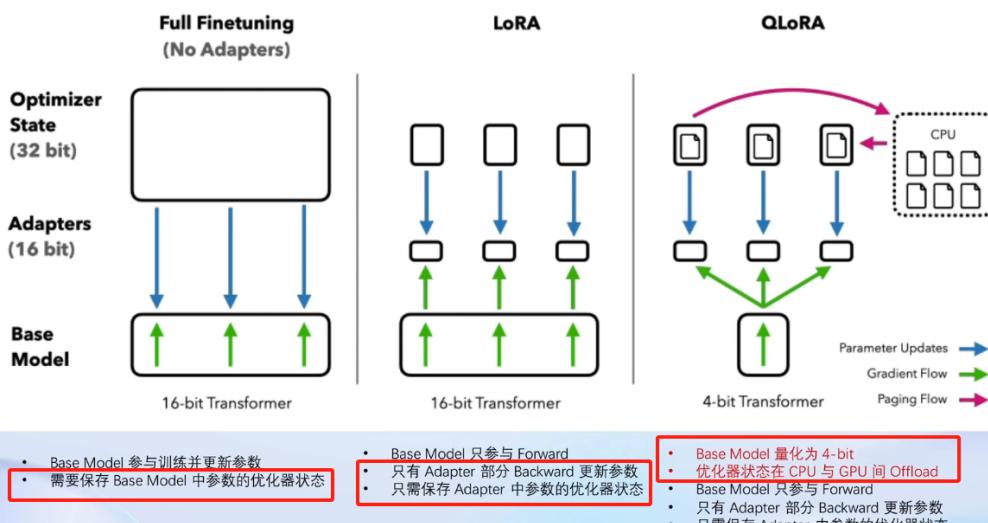
- Full: 😊 → 🚗
- LoRA: 😊 → 🚗
- QLoRA: 😊 → 🚗

对这个玩具中的某些零件进行改动



LoRA & QLoRA

OpenMMLab bili bili



X-Tuner

XTuner 简介

OpenMMLab 

功能亮点

适配多种生态

- 多种微调算法

多种微调策略与算法，覆盖各类 SFT 场景

- 适配多种开源生态

支持加载 HuggingFace、ModelScope 模型或数据集

- 自动优化加速

开发者无需关注复杂的显存优化与计算加速细节

适配多种硬件

- 训练方案覆盖 NVIDIA 20 系以上所有显卡

- 最低只需 8GB 显存即可微调 7B 模型

github 页面找到详细的内容和最新的动态

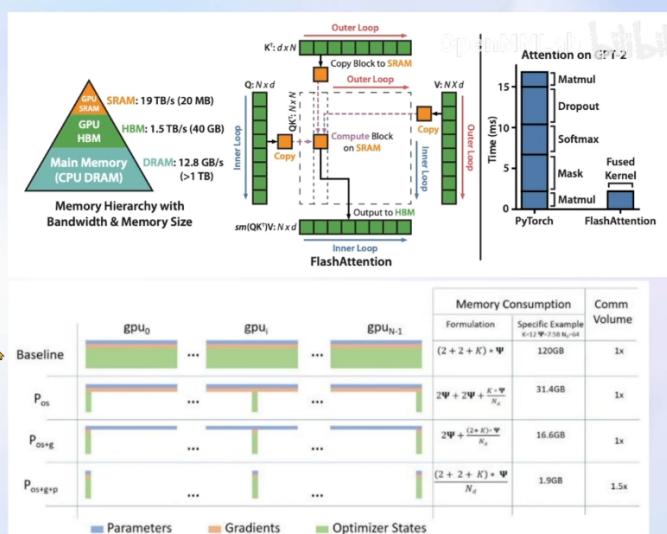


8GB 显存玩转 LLM

Flash Attention 和 DeepSpeed ZeRO 是 XTuner 最重要的两个优化技巧

Flash Attention

Flash Attention 将 Attention 计算并行化，避免了计算过程中 Attention Score NxN 的显存占用（训练过程中的 N 都比较大）



DeepSpeed ZeRO

ZeRO 优化，通过将训练过程中的参数、梯度和优化器状态切片保存，能够在多 GPU 训练时显著节省显存

除了将训练中间状态切片外，DeepSpeed 训练时使用 FP16 的权重，相较于 Pytorch 的 AMP 训练，在单 GPU 上也能大幅节省显存

这个deepspeed_zero的这个优化方法

	Memory Consumption	Comm Volume
Baseline	$(2 + 2 + K) \cdot \Psi$	1x
ZeRO 1	$2\Psi + 2\Psi + \frac{K \cdot \Psi}{N_d}$	1x
ZeRO 2	$2\Psi + \frac{(2+K) \cdot \Psi}{N_d}$	1x
ZeRO 3	$(2 + 2 + K) \cdot \Psi$ N_d	1.5x

实战环节：

1. 安装

```
conda create --name xtuner0.1.9 python=3.10 -y

conda activate xtuner0.1.9
cd ~
mkdir xtuner019 && cd xtuner019

# 无法访问github的用户请从 gitee 拉取：
git clone -b v0.1.9 https://gitee.com/InternLM/xtuner

cd xtuner
# 从源码安装 XTuner
pip install -e '.[all]'
```

```
# 创建一个微调 oasst1 数据集的工作路径, 进入  
mkdir ~/ft-oasst1 && cd ~/ft-oasst1
```

2. 微调

2.1 准备配置文件

```
# 列出所有开箱即用的配置  
xtuner list-cfg  
  
# 拷贝一个配置文件到当前目录, 选择internlm_chat_7b_qlora_oasst1_e3的配置文件  
cd ~/ft-oasst1  
xtuner copy-cfg internlm_chat_7b_qlora_oasst1_e3 .
```

2.2 模型 & 数据集下载

- internlm-chat-7b模型：直接复制平台已有模型，或者可以手动从ModelScope下载模型
- openassistant-guanaco(oasst1)数据集：由于huggingface网络问题，也直接复制平台提前下载的数据

```
cp -r /root/share/temp/model_repos/internlm-chat-7b ~/ft-oasst1/  
cp -r /root/share/temp/datasets/openassistant-guanaco .
```

2.3 修改配置文件

即修改配置文件internlm_chat_7b_qlora_oasst1_e3_copy.py中的**模型&数据集的本地路径**

核心超参：

参数名	解释
data_path	数据路径或HuggingFace仓库名
max_length	单条数据最大Token数，超过则截断
pack_to_max_length	是否将多条短数据拼接到max_length，提高GPU利用率
accumulative_counts	梯度累积，每多少次backward更新一次参数
evaluation_inputs	训练过程中，会根据给定的问题进行推理，便于观测训练状态
evaluation_freq	Evaluation的评测间隔iter数
.....

如果想把显卡的现存吃满，充分利用显卡资源，可以将**max_length**和**batch_size**这两个参数调大

2.4 开始微调

训练

```
# 单卡, 用刚才改好的config文件训练, 并用deepspeed加速
xtuner train ./internlm_chat_7b_qlora_oasst1_e3_copy.py --deepspeed
deepspeed_zero2
```

tmux (Terminal MULTipleXer) 终端复用器 [Tmux 使用教程](#)

```
apt update -y
apt install tmux -y
tmux new -s finetune # 新建名为<finetune>的tmux会话窗口, 按住ctrl+B, 再按D可返回原始终端
tmux attach -t finetune # 继续回到tmux虚拟窗口
```

将得到的 .pth 模型转换为 HuggingFace 模型, 即: 生成 Adapter 文件夹

```
mkdir hf
export MKL_SERVICE_FORCE_INTEL=1
# xtuner convert pth_to_hf ${CONFIG_NAME_OR_PATH} ${PTH_file_dir} ${SAVE_PATH}
xtuner convert pth_to_hf ./internlm_chat_7b_qlora_oasst1_e3_copy.py
./work_dirs/internlm_chat_7b_qlora_oasst1_e3_copy/epoch_1.pth ./hf
```

此时, hf 文件夹即为我们平时所理解的“LoRA 模型文件 (Adapter)”

生成的.safetensors文件 (以前是.bin文件) 即为微调过后的LoRA模型

3. 部署与测试

3.1 将 HuggingFace adapter 合并到大语言模型

```
xtuner convert merge ./internlm-chat-7b ./hf ./merged --max-shard-size 2GB
# xtuner convert merge \
#   ${NAME_OR_PATH_TO_LLM} \
#   ${NAME_OR_PATH_TO_ADAPTER} \
#   ${SAVE_PATH} \
#   --max-shard-size 2GB 分块保存
```

3.2 与合并后的模型对话

```
# 加载 Adapter 模型对话 (Float 16), 注意底座模型不一样, 对应的prompt-template就不同
xtuner chat ./merged --prompt-template internlm_chat
# 4 bit 量化加载
# xtuner chat ./merged --bits 4 --prompt-template internlm_chat
```

注: 也可以选择不融合 basemodel和adapter, 而直接使用语句

```
xtuner chat ./internlm-chat-7b --adapter internlm-7b-qlora-msagent-react
```

3.3 微调效果比较

加载模型时，选择加载 `internlm-chat-7b` 或者微调后的 `merged` 来比较两者效果

```
xtuner chat ./merged --prompt-template internlm_chat  
xtuner chat ./internlm-chat-7b --prompt-template internlm_chat
```

`xtuner chat` 的启动参数

启动参数	干哈滴
--prompt-template	指定对话模板
--system	指定SYSTEM文本
--system-template	指定SYSTEM模板
--bits	LLM位数
--bot-name	bot名称
--with-plugins	指定要使用的插件
--no-streamer	是否启用流式传输
--lagent	是否使用lagent
--command-stop-word	命令停止词
--answer-stop-word	回答停止词
--offload-folder	存放模型权重的文件夹（或者已经卸载模型权重的文件夹）
--max-new-tokens	生成文本中允许的最大 <code>token</code> 数量
--temperature	温度值
--top-k	保留用于顶k筛选的最高概率词汇标记数
--top-p	如果设置为小于1的浮点数，仅保留概率相加高于 <code>top_p</code> 的最小一组最有可能的标记
--seed	用于可重现文本生成的随机种子

4. 自定义微调

以 [Medication QA](#) 数据集为例，将其往 医学问答 领域对齐

问题	答案
What are ketorolac eye drops? (什么是酮咯酸滴眼液?)	Ophthalmic ketorolac is used to treat itchy eyes caused by allergies. It also is used to treat swelling and redness (inflammation) that can occur after cataract surgery. Ketorolac is in a class of medications called nonsteroidal anti-inflammatory drugs (NSAIDs). It works by stopping the release of substances that cause allergy symptoms and inflammation.

4.1 数据准备

原格式: (.xlsx)

问题	药物类型	问题类型	回答	主题	URL
aaa	bbb	ccc	ddd	eee	fff

4.1.1 将原格式数据转为XTuner的.jsonl数据格式

ChatGPT 生成的 python 代码见本仓库的 [xlsx2jsonl.py](#) 有空再跑一遍

```
[{
  "conversation": [
    {
      "system": "xxx",
      "input": "xxx",
      "output": "xxx"
    }
  ]
},
{
  "conversation": [
    {
      "system": "xxx",
      "input": "xxx",
      "output": "xxx"
    }
  ]
}]
```

4.2 开始自定义微调

```
mkdir ~/ft-medqa && cd ~/ft-medqa
cp -r ~/ft-oasst1/internal-chat-7b . # 基座模型
git clone https://github.com/InternLM/tutorial # 开发机用不了, 只能upload
cp ~/tutorial/xtuner/MedQA2019-structured-train.jsonl .
```

4.2.1 准备配置文件

```
# 复制配置文件到当前目录
xtuner copy-cfg internlm_chat_7b_qlora_oasst1_e3 .
# 改个文件名, 对应medqa数据集
mv internlm_chat_7b_qlora_oasst1_e3_copy.py
internlm_chat_7b_qlora_medqa2019_e3.py
```

修改配置文件.py的内容

```
# 修改import部分
- from xtuner.dataset.map_fns import oasst1_map_fn, template_map_fn_factory
+ from xtuner.dataset.map_fns import template_map_fn_factory

# 修改模型为本地路径
- pretrained_model_name_or_path = 'internlm/internlm-chat-7b'
+ pretrained_model_name_or_path = './internlm-chat-7b'

# 修改训练数据为 MedQA2019-structured-train.jsonl 路径
- data_path = 'timdettmers/openassistant-guanaco'
+ data_path = 'MedQA2019-structured-train.jsonl'

# 修改 train_dataset 对象
train_dataset = dict(
    type=process_hf_dataset,
-   dataset=dict(type=load_dataset, path=data_path),
+   dataset=dict(type=load_dataset, path='json',
data_files=dict(train=data_path)),
    tokenizer=tokenizer,
    max_length=max_length,
-   dataset_map_fn=alpaca_map_fn,
+   dataset_map_fn=None,
    template_map_fn=dict(
        type=template_map_fn_factory, template=prompt_template),
    remove_unused_columns=True,
    shuffle_before_pack=True,
    pack_to_max_length=pack_to_max_length)
```

4.2.2 启动!

先老样子 tmux new -s medqa

```
xtuner train internlm_chat_7b_qlora_medqa2019_e3.py --deepspeed deepspeed_zero2
```

4.2.3 .pth转.safetensors

跟2.4节一样

4.2.4 部署与测试

和第3节一样