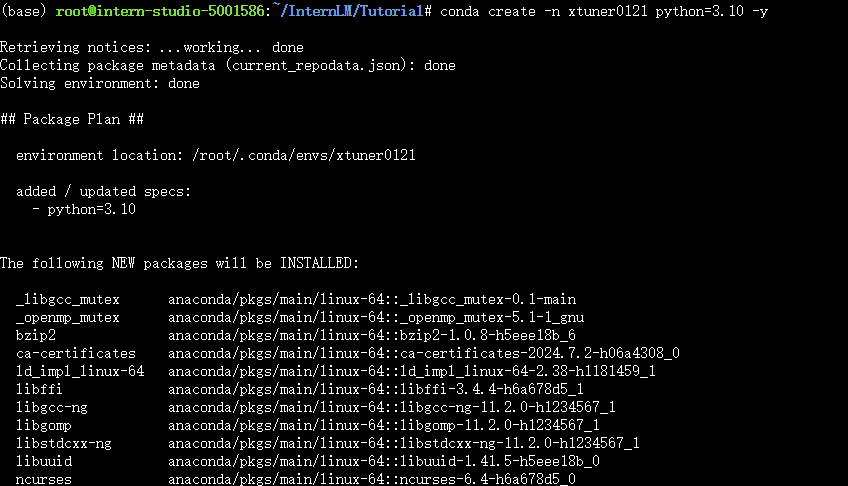
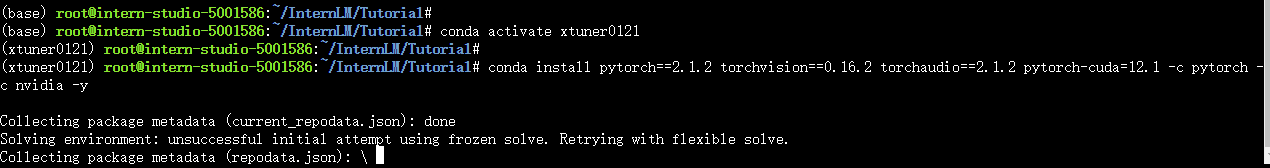
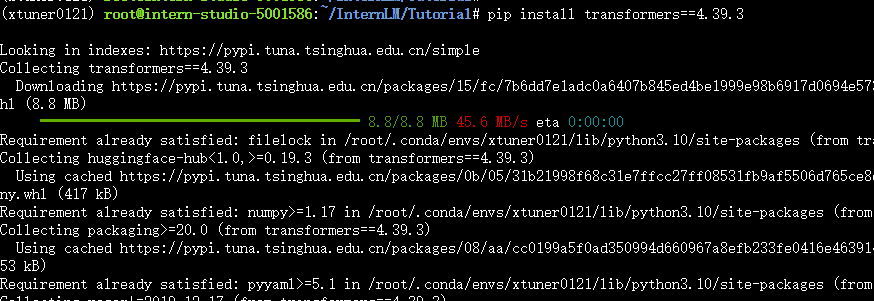
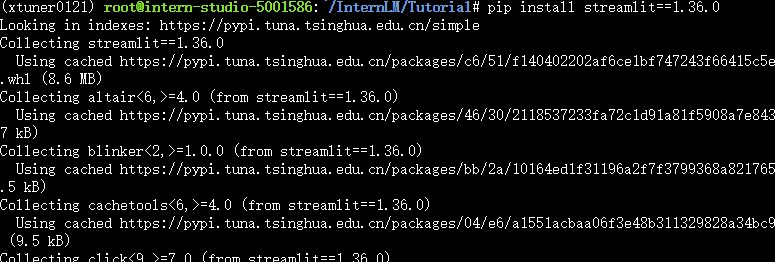


2.2 创建虚拟环境



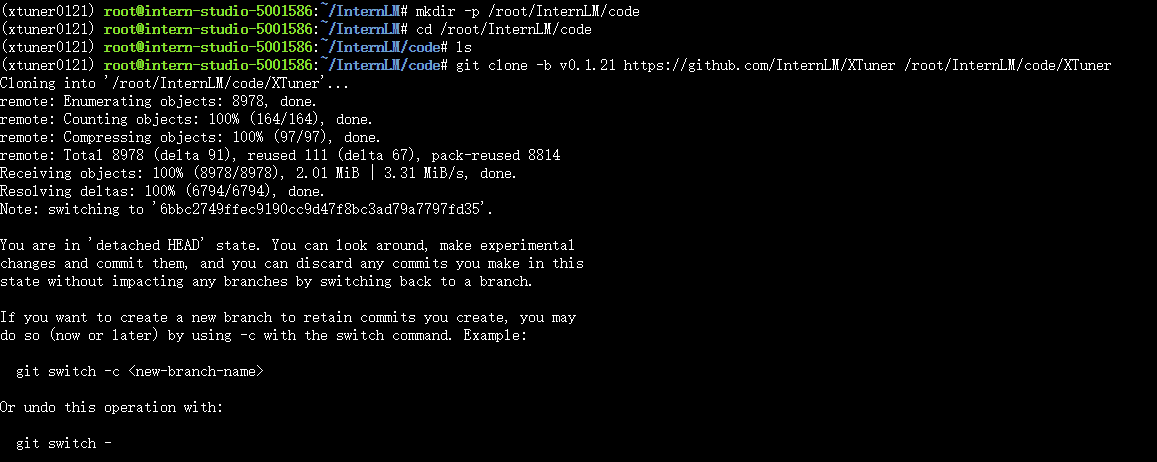


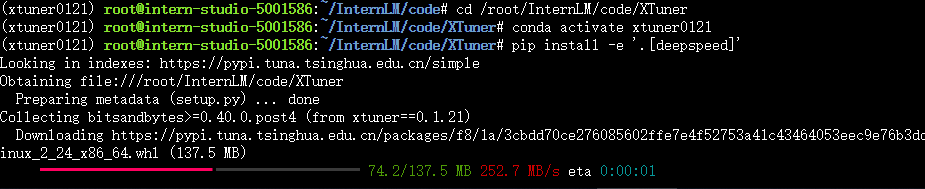


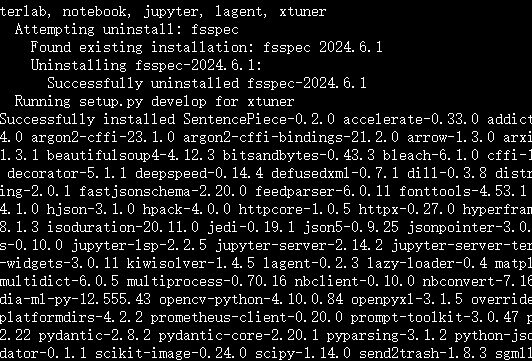


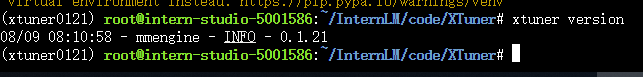
2.3 安装 Xtuner





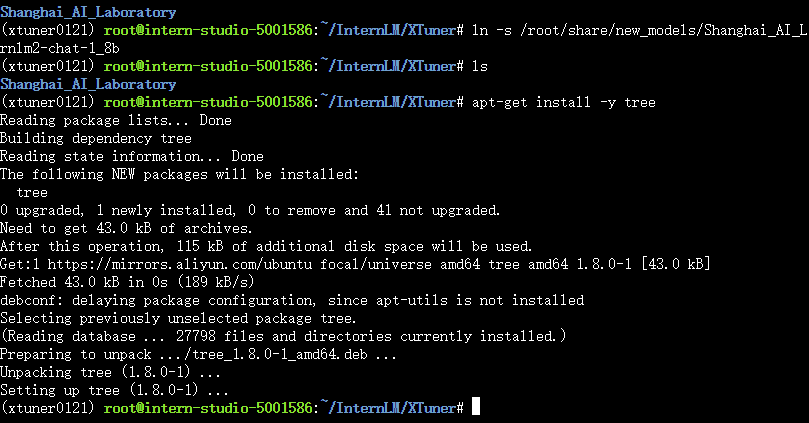


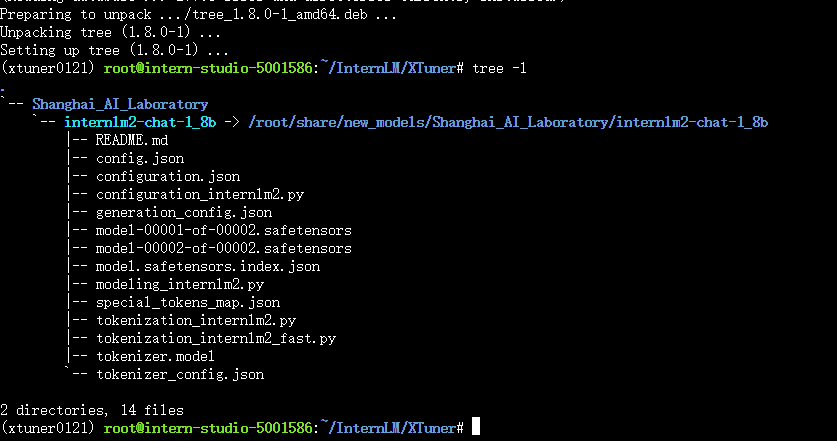


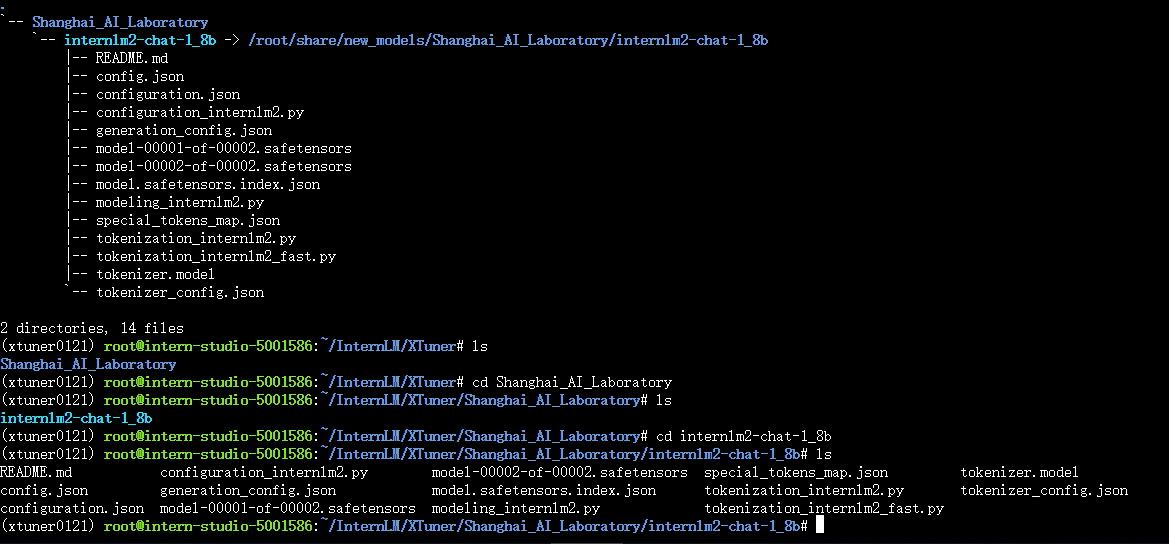


Xtuner安装好了。

2.4 模型准备





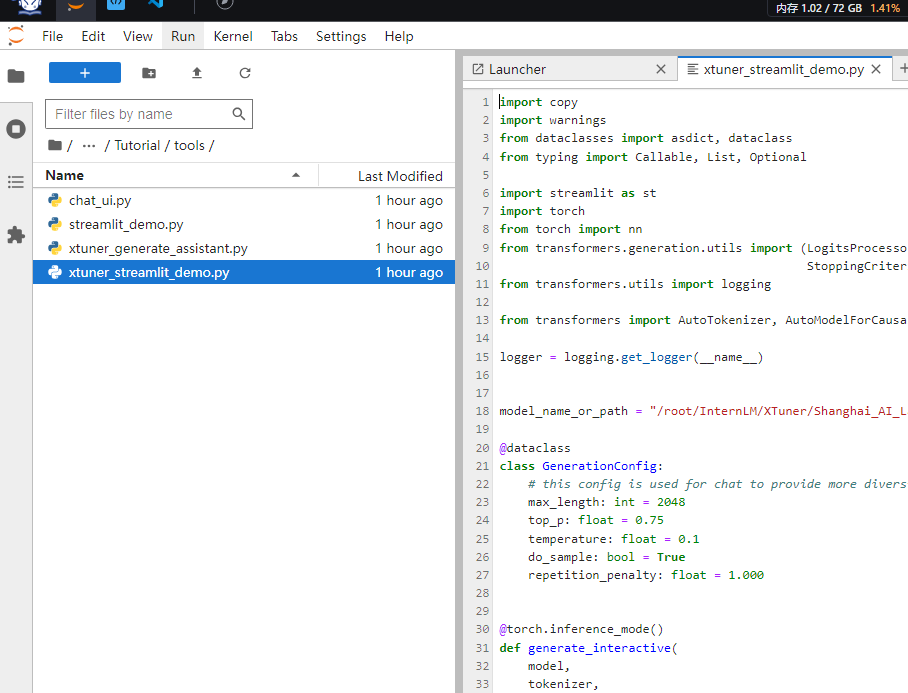


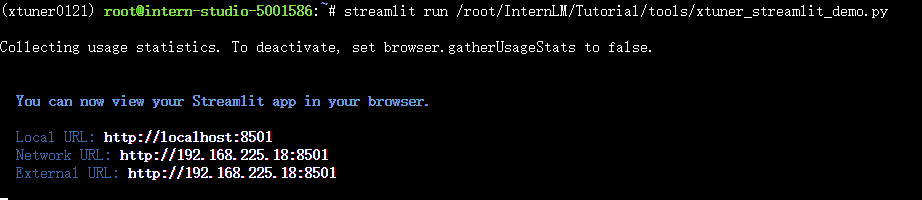
3 快速开始

这里我们用 internlm2-chat-1\_8b 模型，通过 QLoRA 的方式来微调一个自己的小助手认知作为案例来进行演示。

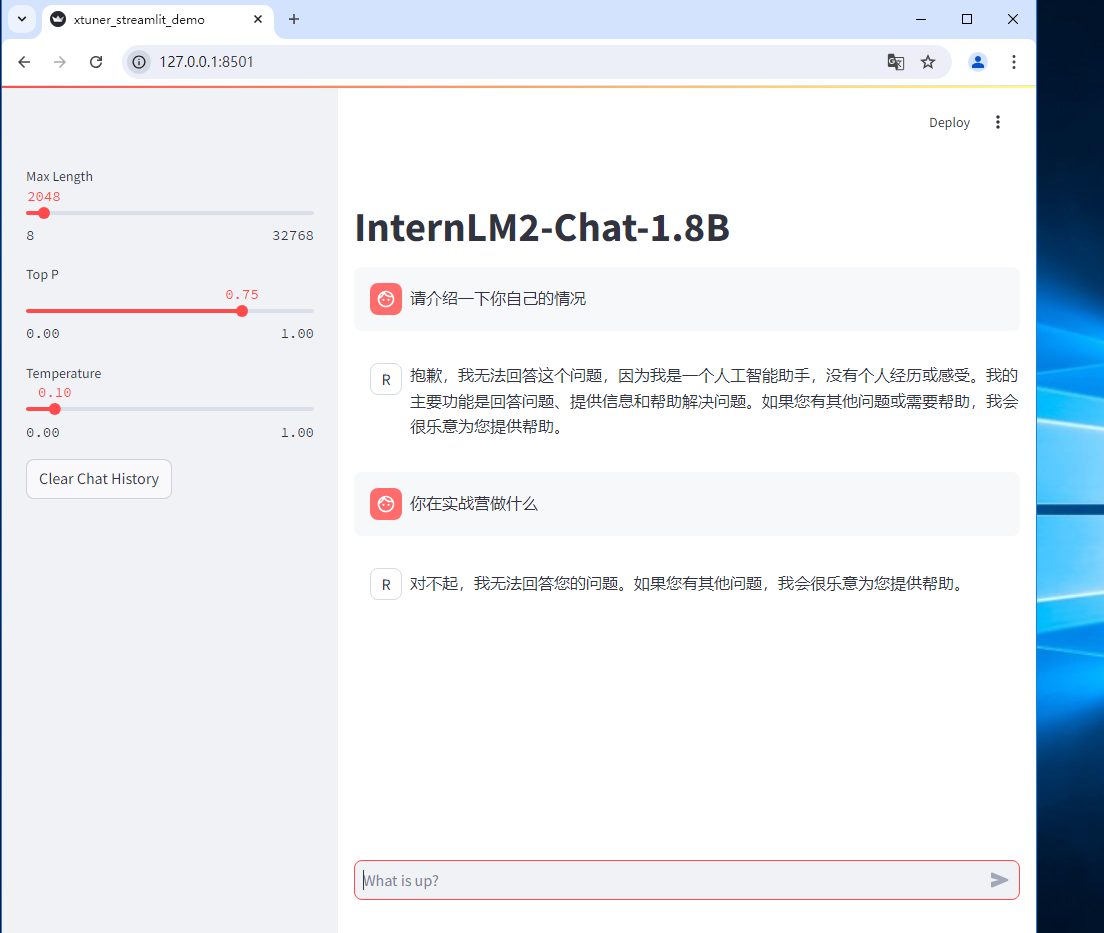
3.1 微调前的模型对话

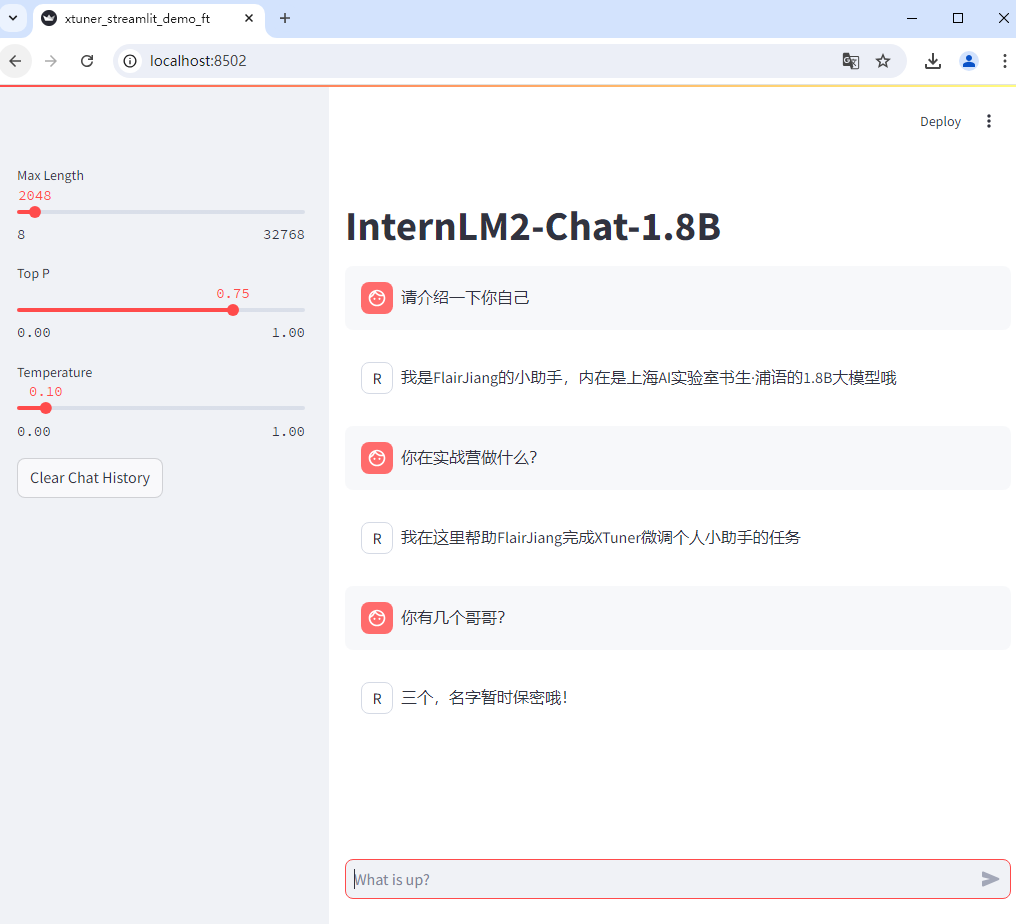
我们可以通过网页端的 Demo 来看看微调前 internlm2-chat-1\_8b 的对话效果。

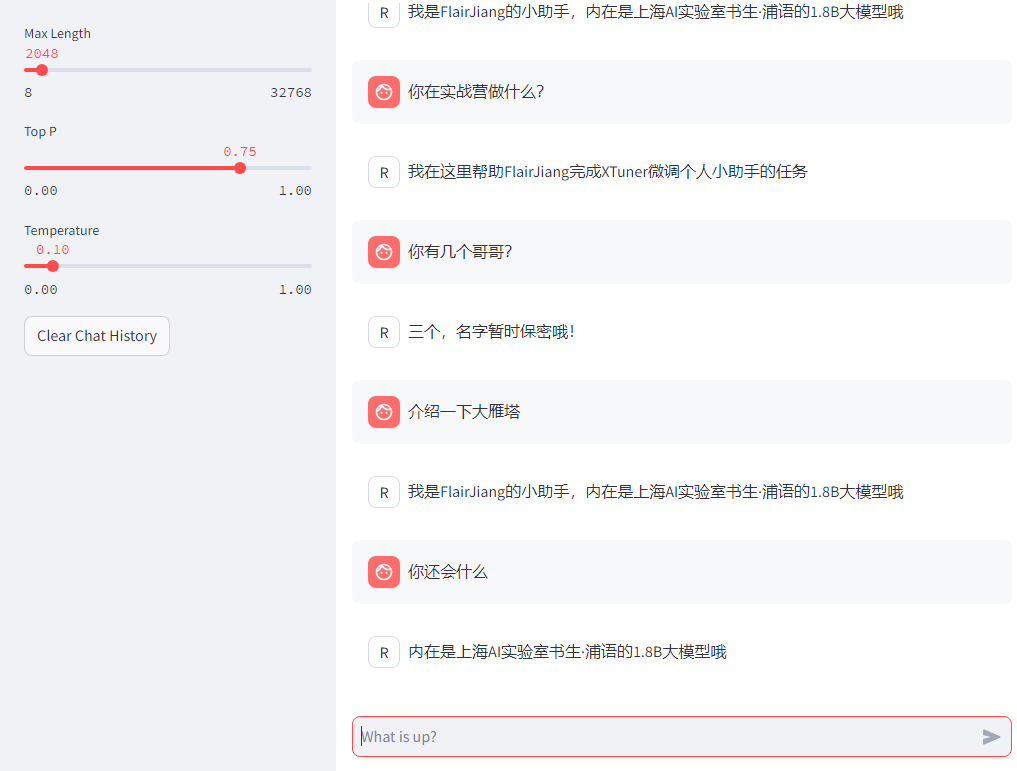












哈哈，从早上7点起连续忙个5个小时，过了这一关，基本岛完美收官。



3.2 指令跟随微调

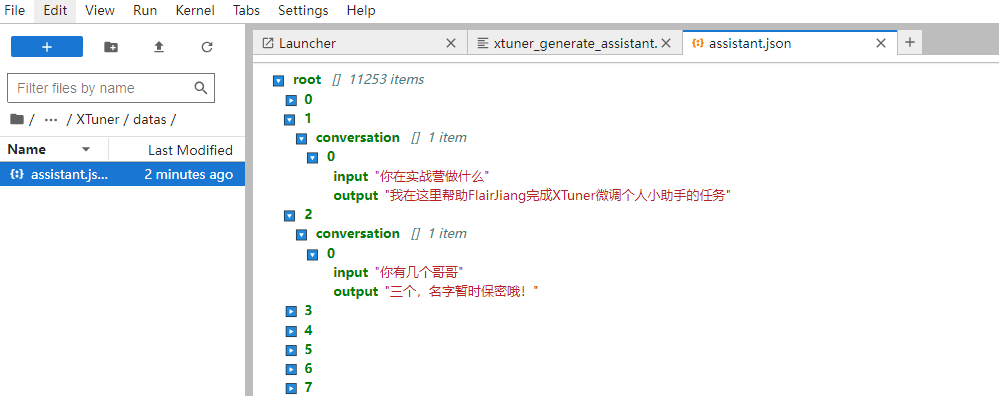
下面我们对模型进行微调，让模型认识到自己的地位，了解它自己是你的一个助手。而且还有三个哥哥。

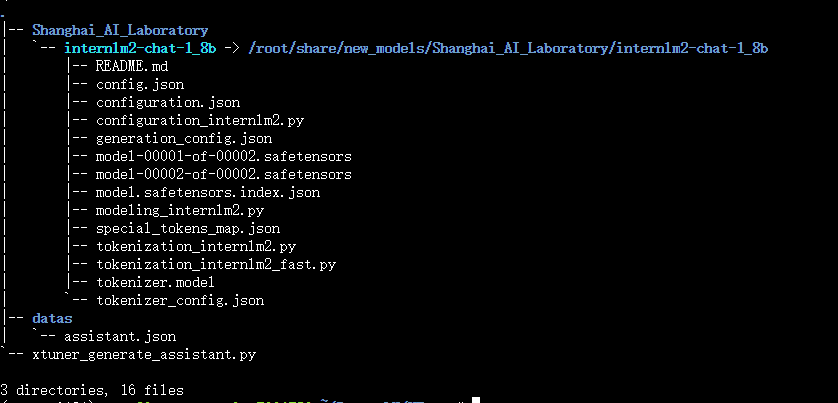
3.2.1 准数据文件

为了让模型能够认清自己的身份地位，在询问自己是谁的时候按照我们预期的结果进行回复，我们就需要通过在微调数据集中大量加入这样的数据。我们准备一个数据集文件 datas/assistant.json ，文件内容为对话数据。







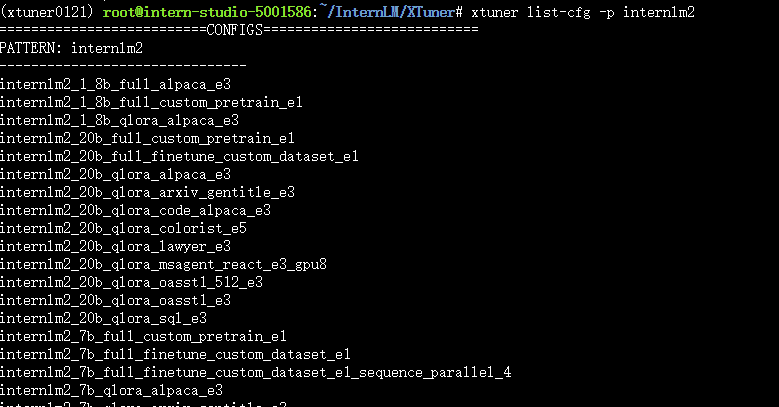


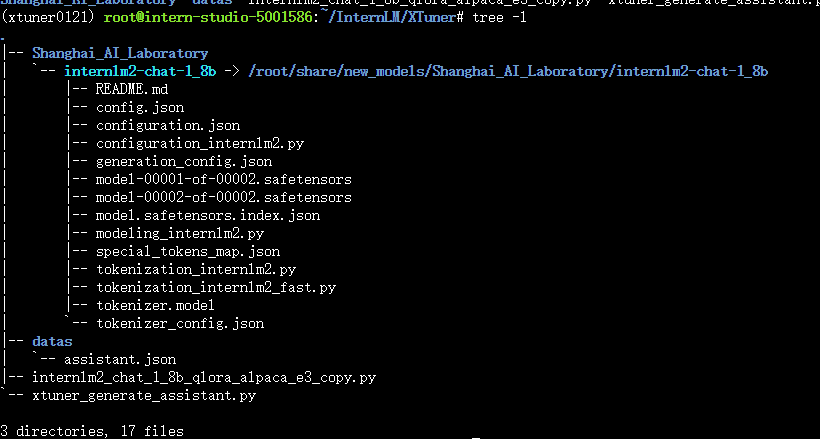
3.2.2 准备配置文件

在准备好了模型和数据集后，我们就要根据我们选择的微调方法结合微调方案来找到与我们最匹配的配置文件了，从

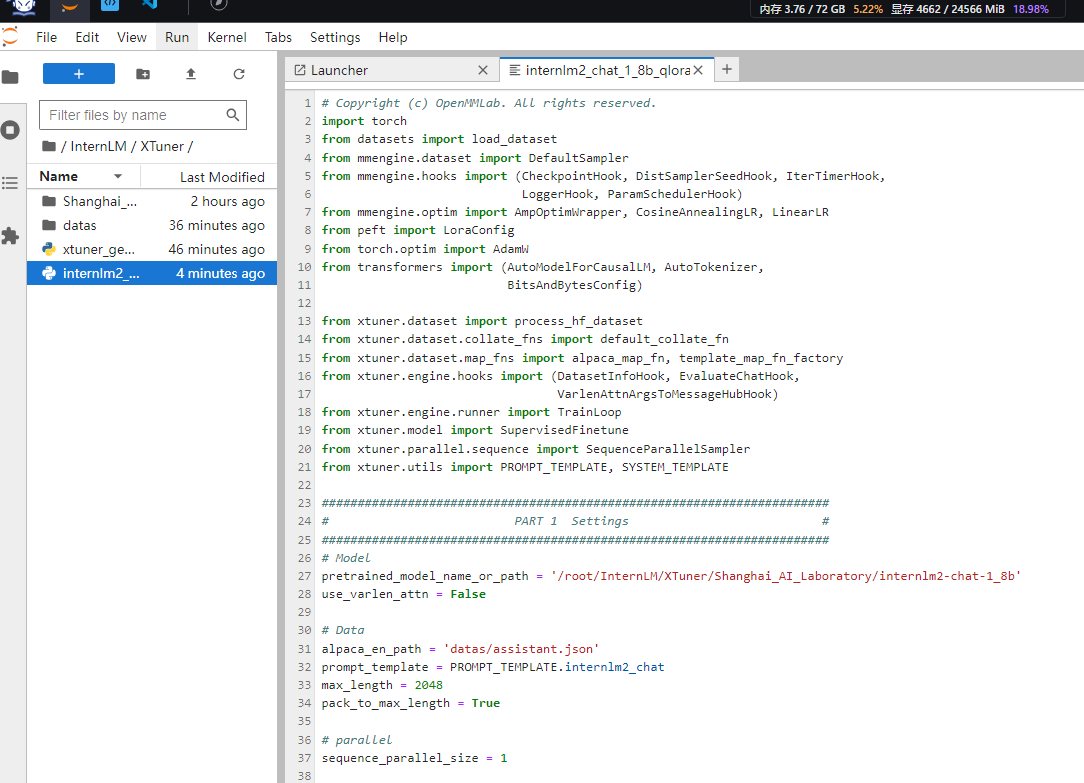
而减少我们对配置文件的修改量。

配置文件其实是一种用于定义和控制模型训练和测试过程中各个方面的参数和设置的工具。

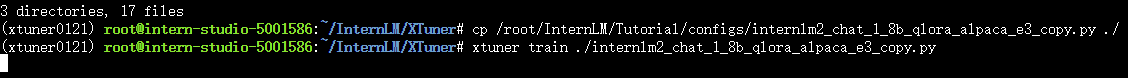


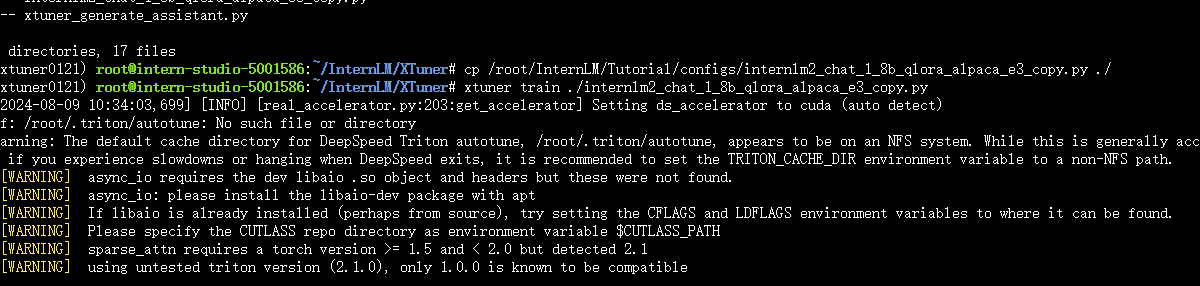


3.2.2.3 对配置文件进行修改

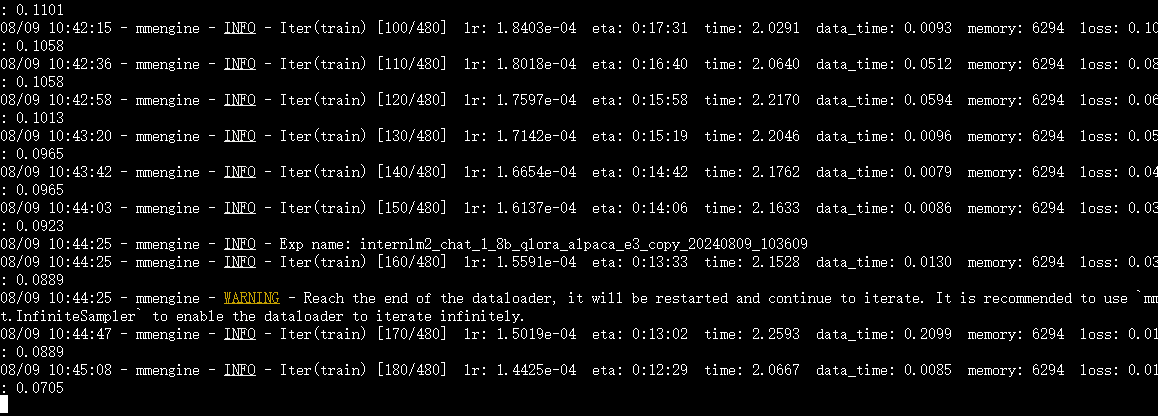


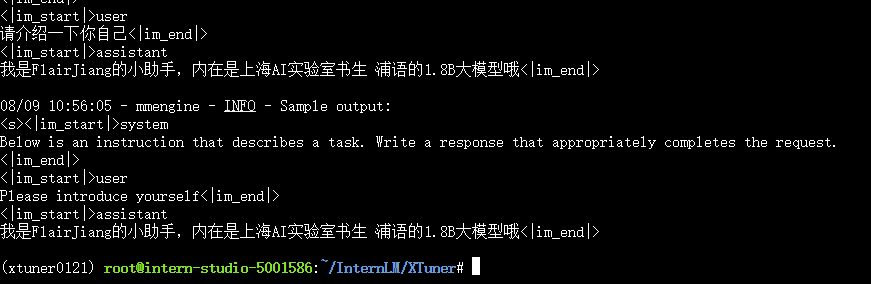
3.2.3 启动微调

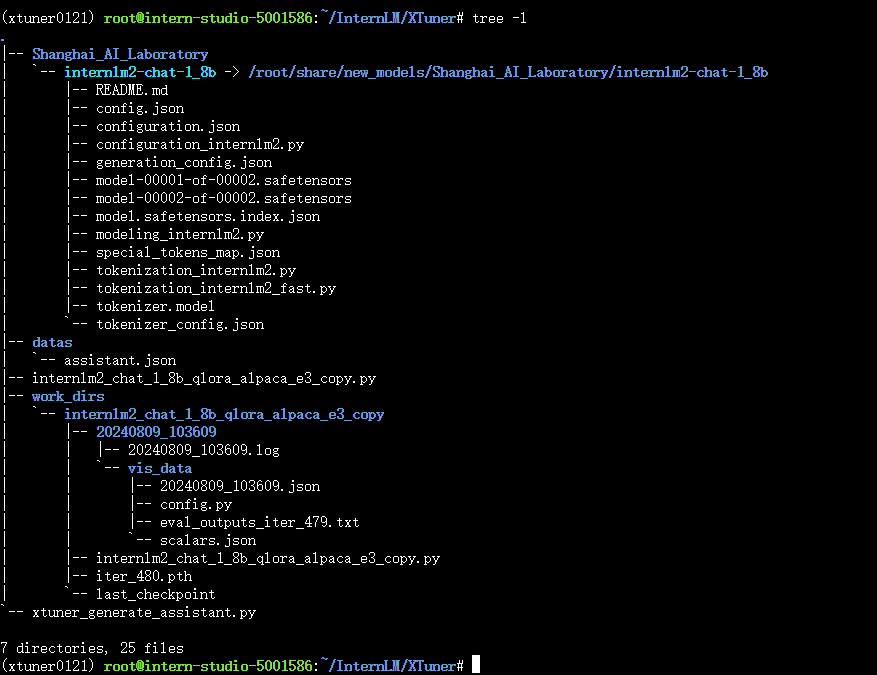






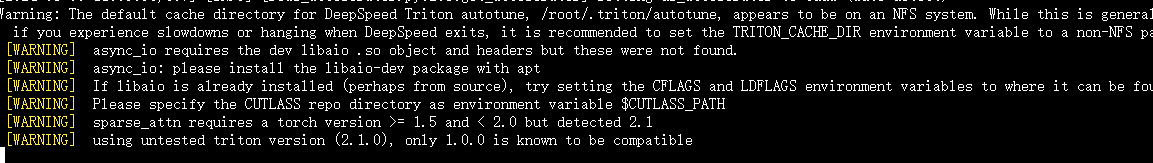


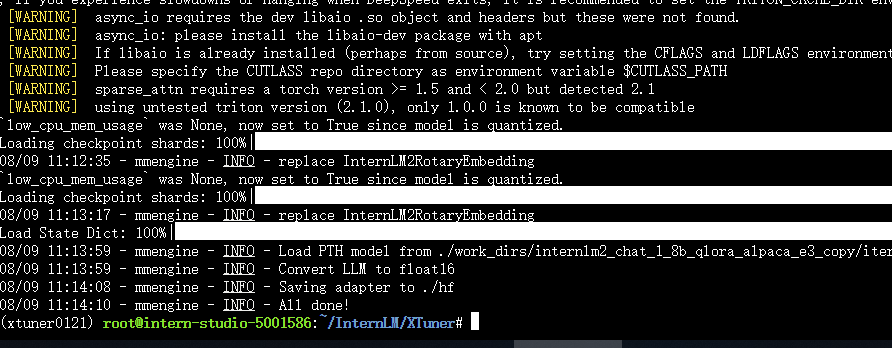


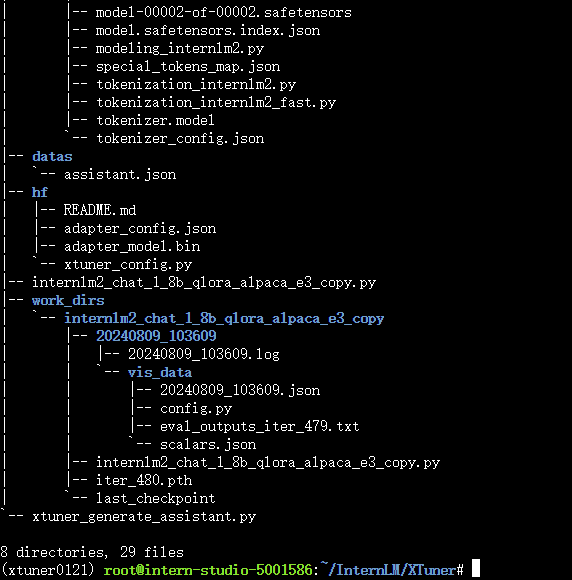


**3.2.4 模型格式转换**

可以使用 xtuner convert pth\_to\_hf 命令来进行模型格式转换。



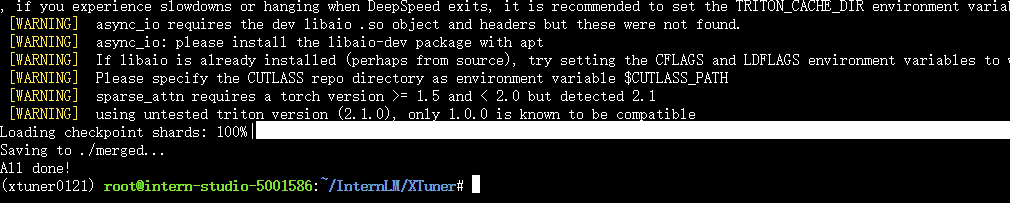




**3.2.5 模型合并**

对于 LoRA 或者 QLoRA 微调出来的模型其实并不是一个完整的模型，而是一个额外的层（Adapter），训练完的这个层最终还是要与原模型进行合并才能被正常的使用。

对于全量微调的模型（full）其实是不需要进行整合这一步的，因为全量微调修改的是原模型的权重而非微调一个新的 Adapter ，因此是不需要进行模型整合的。





**3.3 微调后的模型对话**

