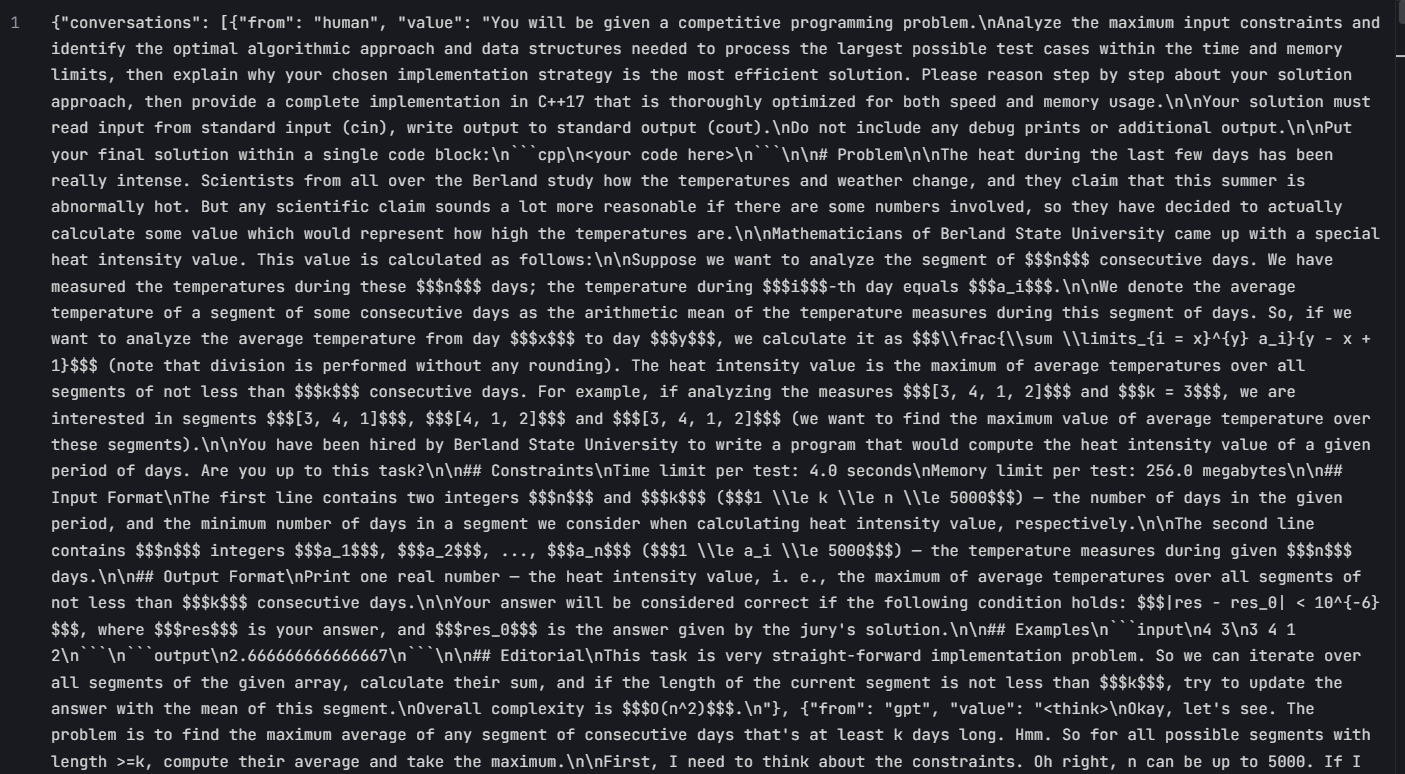
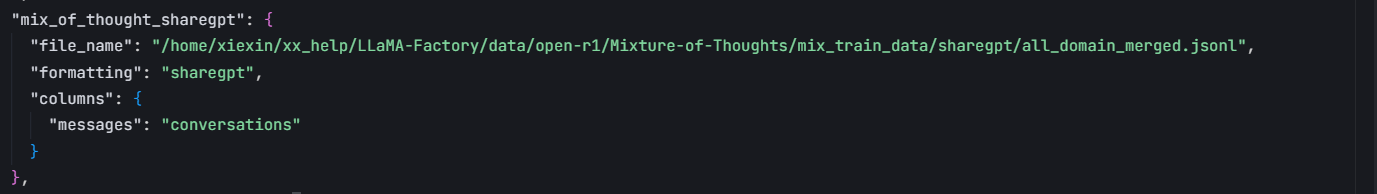
（1）Instruct就是alpaca格式，我描述的有差异

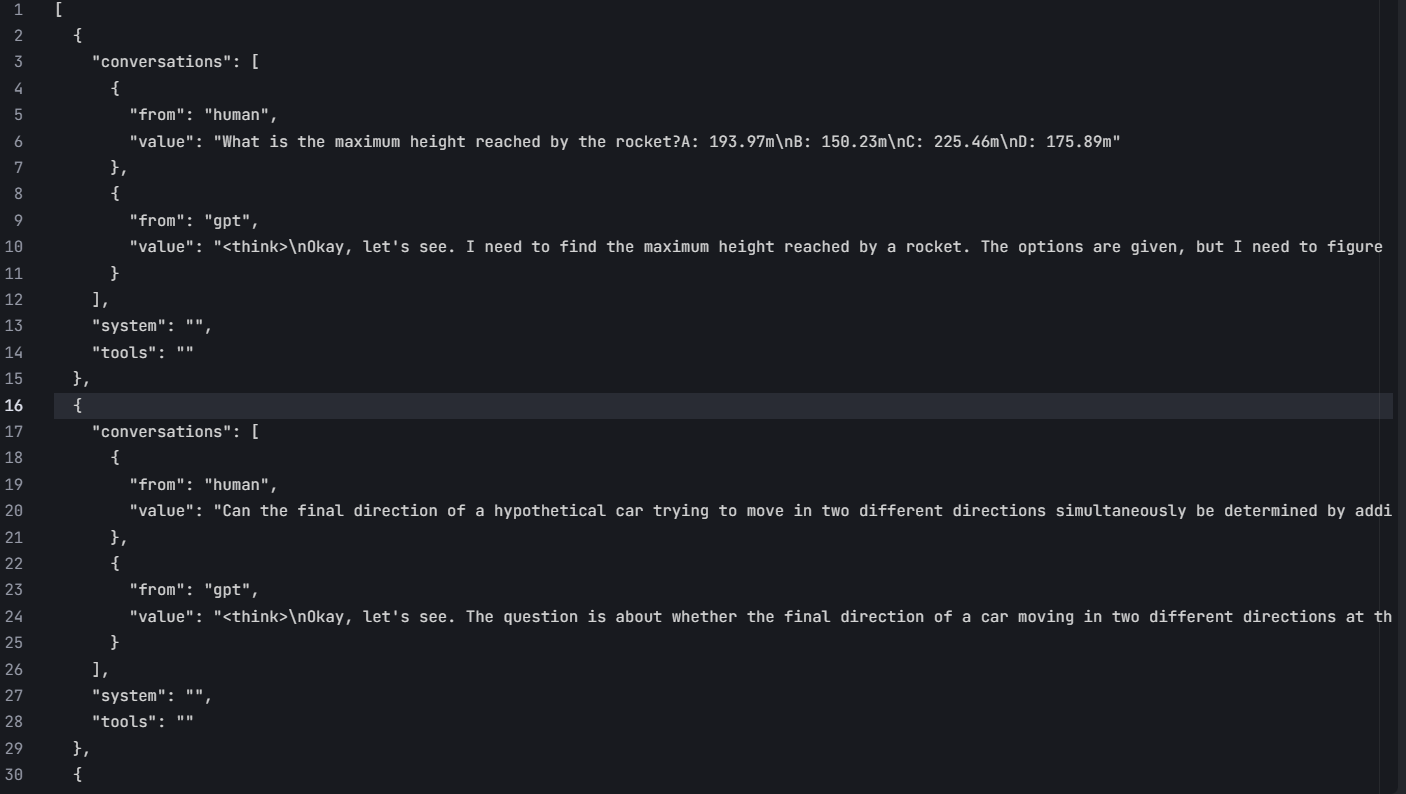
1. 正确的jsonl格式是指标准的jsonl数据，按行存储对象。原生的Llamafactory是能够支持处理json和jsonl的数据格式的，json和jsonl文件的区别在于如果开启了streaming使用流式处理，要使用jsonl格式的数据才能够正确的加载。sharegpt格式和alpaca格式的文件都能使用jsonl格式来加载



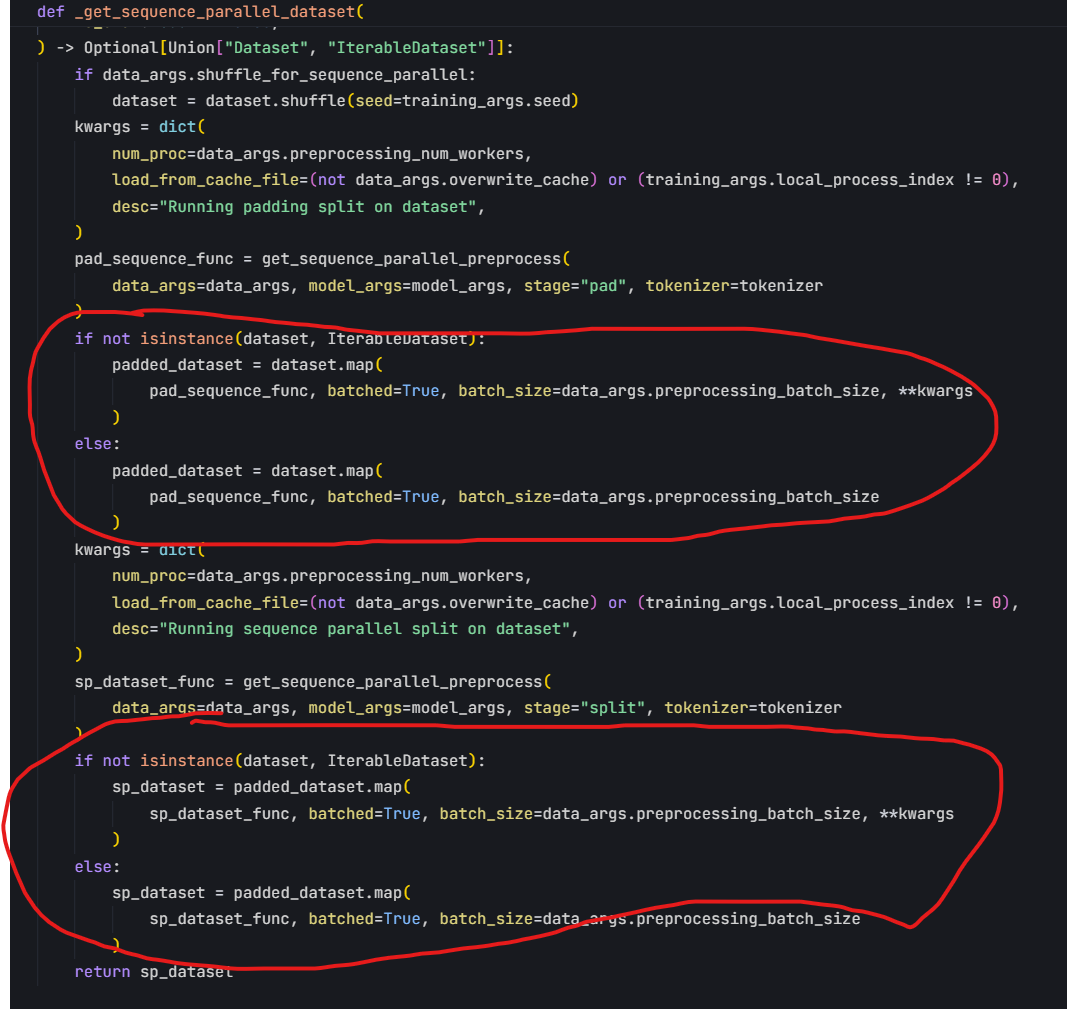
从json文件变成jsonl文件，只需要在datainfo中只需要把file\_name改变即可，不需要改变formating格式。



错误的格式是指我之前用的是json数据的格式，用多行来存储一个json对象，只是文件命名的后缀是jsonl。Json对象中system和tools可有可无，如果要添加这两个属性，在data\_info中columns中添加这两个属性即可。

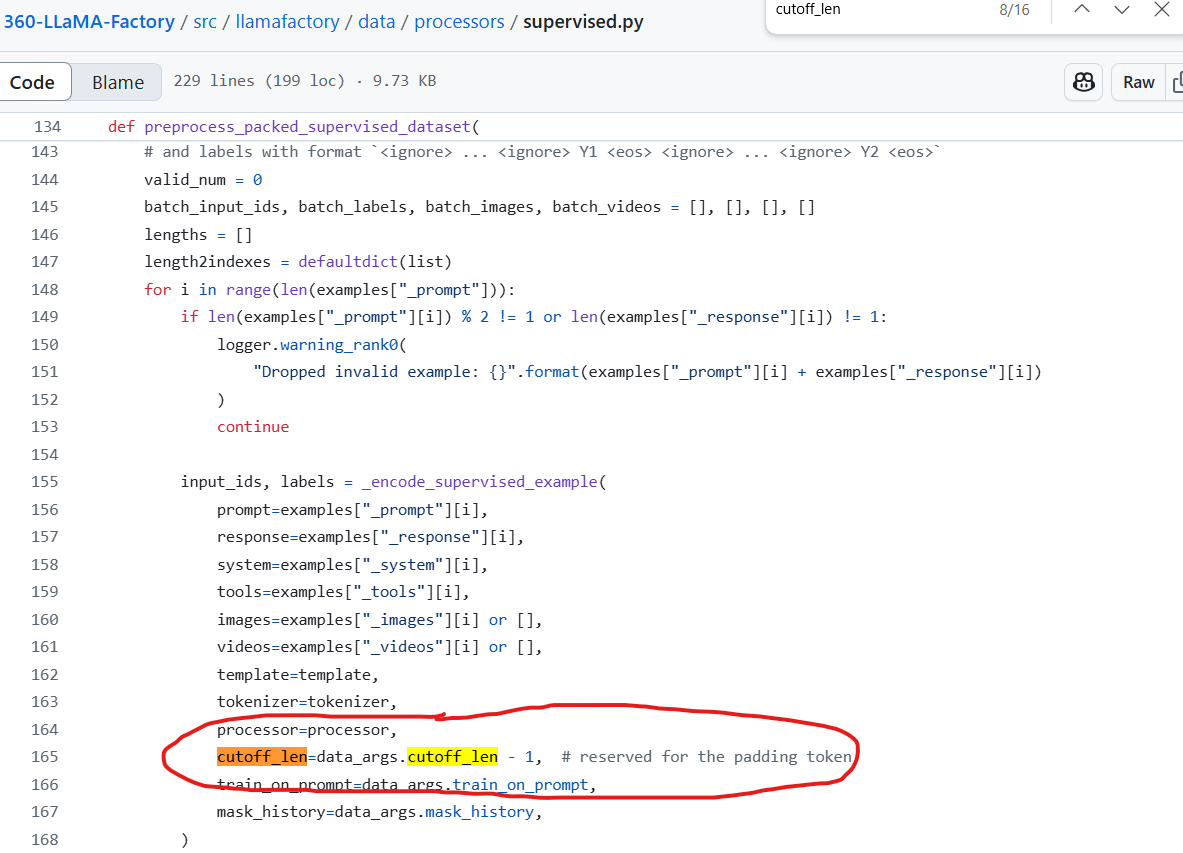


1. 改loader只是interableDataset加载数据上的问题，360-llamafactory没考虑到开启streaming设置时使用interableDataset的加载的问题。会对streaming开启时加载的数据有影响。我依据终端报错信息，更改了LLaMA-Factory/src/llamafactory/data/loader.py文件中的函数\_get\_sequence\_parallel\_dataset，其他src目录下的内容没有做任何改动。

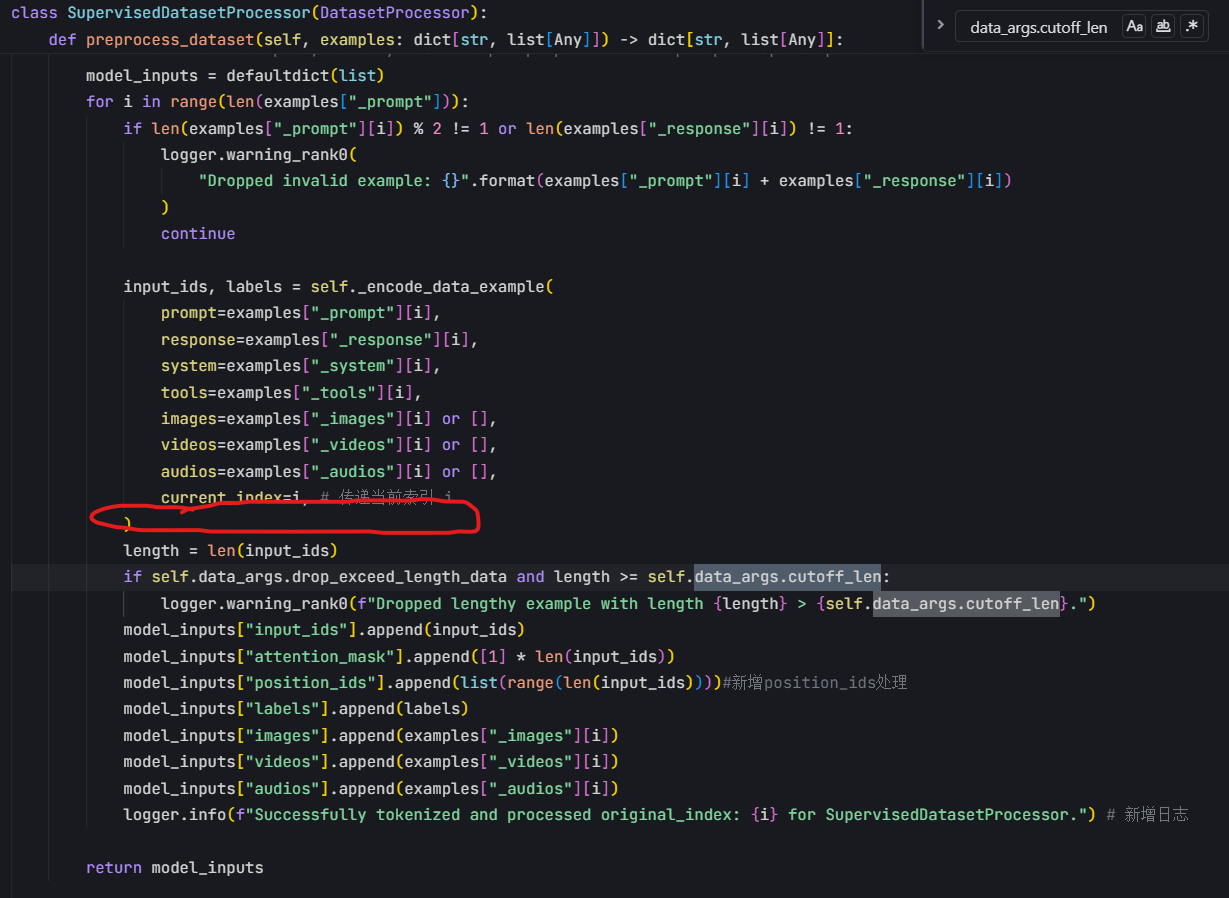
local\_process\_index表示的是进程索引号，这个参数llamafactory和360-llamafactory都没有做任何改动，是transformer包自带的参数。例如，如果一台机器上跑了4个GPU，每个GPU一个进程，那么这台机器上的 local\_process\_index 会是 0, 1, 2, 3。num\_proc和load\_from\_cache\_file只会影响数据加载的速度，不影响训练过程。

1. 如果是使用了packing参数，cutoff\_len要在原基础上加1，是360-llamafactory的代码逻辑，不加1会有对应的终端报错提示，这个很容易发现。（这个错误忽略，应该是0.9.2版本的llamafactory与0.9.1的版本差异引起的）

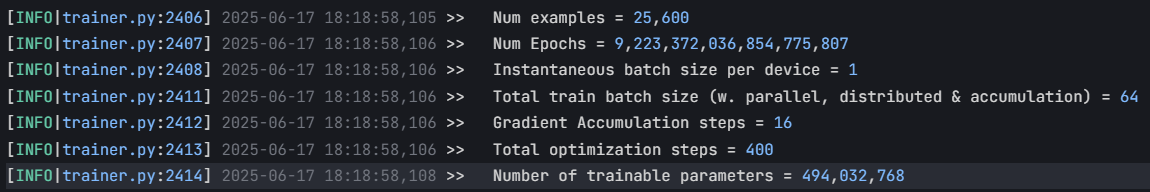
0.9.1 llamafactory对应代码：

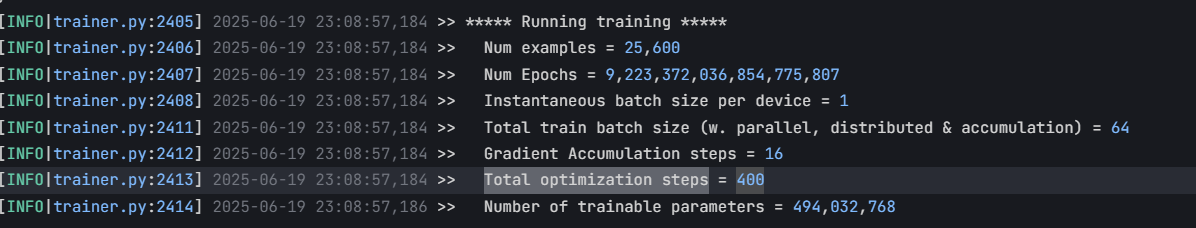


我适配0.9.2 llamafactory后的对应代码，没有重新传递cutoff\_len-1到data\_args.cutoff\_len



1. 验证了分别使用Instruct和sharegpt格式的4K数据进行模型训练时的显存占用，完全相同。是在处理完数据，模型刚进入训练时检查的显存占用
2. 上传的新文件ParquetConvertInstruct.py用于转换mix-of-thought为alpaca或者sharegpt文件，MergeJson用于合并多个json（jsonl）为一个json(jsonl)文件。
3. 在streaming同时开启的情况下，前者与后者分别是packing不开启与开启的训练参数对比，完全一致。Num Epoch是2^63-1，可以忽略。





1. 在开启streaming后max\_step会覆盖num\_train\_epochs参数以及max\_samples

的设置，max\_step用来表示最终的训练步数。实际使用的样本数Num examples=max\_step \* gpu\_nums \* per\_device\_batch\_size \* gradient\_accumulation\_steps=500\*4\*1\*16=32000。所以如果我们打算所有样本训练N个epoch,max\_step应该以最终样本数349317\*N倒推来计算得到max\_step。

packing后相比于packing前，不需要调整max\_step。红框中的内容是开始打包后的调试信息，这表明打包的步骤是在统计完Num examples后执行的，LLaMA-Factory 的打包机制会自动处理后续的样本合并和数量减少，而不会影响训练流程的正确性，不需要手动调整样本数

