

# math-redpajama数据dolma work flow

文档涉及代码路径：

/mnt/geogpt-gpfs/llm-course/home/xzhh/OLMo/data/

作者：许增辉

---

## step0 收集数据

将数据按照dolma格式，和规定的文件夹命名方式整理到一起。

### 0.1 数据采样

使用get\_unzip\_datas.py,从已有的数据里面抽样出所需的数据。定义，源路径，保存路径，最多文件数，所需文件大小。注意脚本使用时，不要输入超过源路径文件大小，bug偷懒还没改，不然会陷入循环。

### 0.2 数据格式对齐

使用add\_id2.py, 将数据格式对齐到如下dolma规定格式。该函数添加id和source。输入参数详见代码

### 0.3 文件格式对齐

将add\_id后的数据按照dolma要求的文件夹命名方式，放到一个documents文件夹下。attribute文件会再tagger的时候自动生成

---

## step1 tagger标签

运行 run\_taggers.sh

参数和详见官方文档.

最后会在documents统计文件夹下生成attribute文件夹，保存各个对应文件的打标分数和标签。此步不会对源数据文件进行处理。

一些坑：

选取tagger是要注意根据数据实际情况选取不要随意选（可参考dolma论文中各数据的处理方式）；

注意，命令之间要有\，不要有#注释行，不然会无法识别命令关键词。

---

## step2 dedupe 去重

运行 run\_dedupe.sh

参数配置：dedupe.json文件

最后会在attribute文件夹下生成一个dedupe对应的文件（可以简单理解为dedupe和tagger一样，都是给数据打个标签）

---

## step3 mixer

运行 run\_mixer.sh

参数配置: math\_dolma\_mixer.yaml

此步会基于前面attribute里面的标签，更具yaml中配置的过滤规则，对源数据进行筛选，最后将数据输出到目标文件夹。

tip:并行参数建议不要超过128，太大会报错。

---

## step4 tokenizer

运行run\_token.sh

参数配置tokenizer.yaml 文件

此步会输出numpy文件用于后面的olmo训练。

get\_all\_numpy\_path.py脚本，输入numpy所在位置文件夹的绝对路径后，输出olmo训练要配置数据路径，直接复制即可。一些注意: eos\_token\_id要和token实际情况设置。同时和olmo配置训练yaml也保持一致。

---

## step5 pre\_train

torchrun -nproc\_per\_node=8 scripts/train.py configs/official/OLMo-1B.yaml

---

## step6 sft数据处理

**整理tulu数据，**

运行python scripts/prepare\_tulu\_data.py -output\_dir tulu/ -t tokenizers/allenai\_eleuther-ai-gpt-neox-20b-ppi-special.json , , 其中参数规范一下，'-out\_dir','-eos','-pad','j'根据实际情况修改。最后生成id和mask的numpy文件

**整理stackmathQA 问答数据**

运行python scripts/prepare\_sft\_stack\_qa\_data.py -output\_dir stachmathqa1600/ -t tokenizers/allenai\_eleuther-ai-gpt-neox-20b-ppi-special.json 其他基本配置同上。代码默认是stackmathqa1600的数据，如要其他的，须在代码内修改。最后同样生成id和mask的numpy文件

## step7 sft\_train

使用训练pre\_train的yaml文件。注意：模型的参数不要变；学习率建议调小10倍；epcoh:建议3-5次左右以下的参数如果感觉写的太长，也可以加入yaml文件中

```
torchrun -nproc_per_node=8 scripts/train.py configs/official/Guass_One-0.7B-v8.yaml \
-data.paths=[/mnt/geogpt-gpfs/llm-course/home/xzhh/OLMo/stackmathqa800/input_ids.npy] \
-data.label_mask_paths=[/mnt/geogpt-gpfs/llm-course/home/xzhh/OLMo/stackmathqa800/label_mask.npy] \
```

```
-load_path=/mnt/geogpt-gpfs/llm-  
course/home/xzhh/OLMo/save/20240324_0.7B_Guass_One_8/step13851-unsharded \  
-run_name=Gauss-One_0.7B_8_sft  
-save_folder=save/20240324_0.7B_Guass_One_8/20240327sft  
-reset_trainer_state  
#注意：使用=和[], 不能缺
```

多个sft数据集

```
torchrun --nproc_per_node=8 scripts/train.py configs/official/Guass_One-0.7B-v8.yaml \  
  --data.paths=[/mnt/geogpt-gpfs/llm-course/home/xzhh/OLMo/tulu/input_ids.npy,/mnt/geogpt-  
gpfs/llm-course/home/xzhh/OLMo/alpaca-gpt4/input_ids.npy,/mnt/geogpt-gpfs/llm-  
course/home/xzhh/OLMo/gsm8k/input_ids.npy] \  
  --data.label_mask_paths=[/mnt/geogpt-gpfs/llm-  
course/home/xzhh/OLMo/tulu/label_mask.npy,/mnt/geogpt-gpfs/llm-course/home/xzhh/OLMo/alpaca-  
gpt4/label_mask.npy,/mnt/geogpt-gpfs/llm-course/home/xzhh/OLMo/gsm8k/label_mask.npy] \  
  --load_path=/mnt/geogpt-gpfs/llm-  
course/home/xzhh/OLMo/save/20240324_0.7B_Guass_One_8/step13851-unsharded \  
  --run_name=Gauss-One_0.7B_8_sft_tul_gsm8k_alpaca  
  --save_folder=save/20240324_0.7B_Guass_One_8/sft_tul_gsm8k_alpaca  
  --reset_trainer_state
```

#注意：=后面的空格，‘，’后面的空格，都不要