Gpt进阶(二): 以古诗集为例,训练一个自己的古诗词gpt模型

电子灵魂华尔兹
公众号-电子灵魂华尔兹

4 人赞同了该文章

使用nanoGPT，手把手带你训练一个属于自己的GPT模型，基于gpt2，优点是cpu也可以跑，简单，快速（LLaMa的模型训练太耗费gpu，很多人也跑不了，所以暂时选择这个）

最终实现:以古诗集为例,训练一个可以续写诗词的gpt模型

开源地址：

> github.com/karpathy/nan

环境准备

环境搭建 参考之前文章 **AI绘画：搭建自己的AI绘画网站(Stable Diffusion)**

以下可以做一个保存，模型训练这些环境都是必须的，这是小编整理的最简单的基础环境搭建流程，收藏不迷路

1 conda(python环境)

```
#网址：https://conda.io/en/latest/miniconda.html
wget https://repo.anaconda.com/miniconda/Miniconda3-py310_23.1.0-1-Linux-x86_64.sh #下
sh Miniconda3-py39_4.12.0-Linux-x86_64.sh # 执行
~/miniconda3/bin/conda init #初始化Shell，以便直接运行conda
conda create --name nanogpt python=3.9  #关启shell，创建虚拟环境
conda activate nanogpt #激活
```

2 下载代码

> wget github.com/karpathy/nan

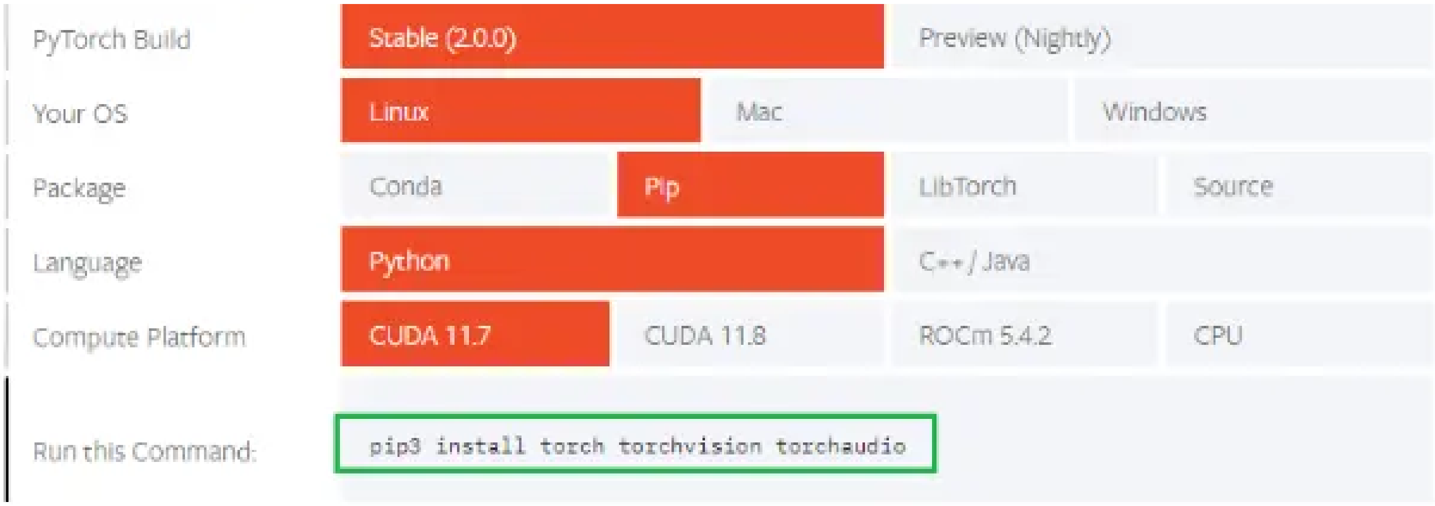
▲赞同 4 ▼ 2 条评论 分

3 安装依赖：

> pip install transformers tiktoken tqdm

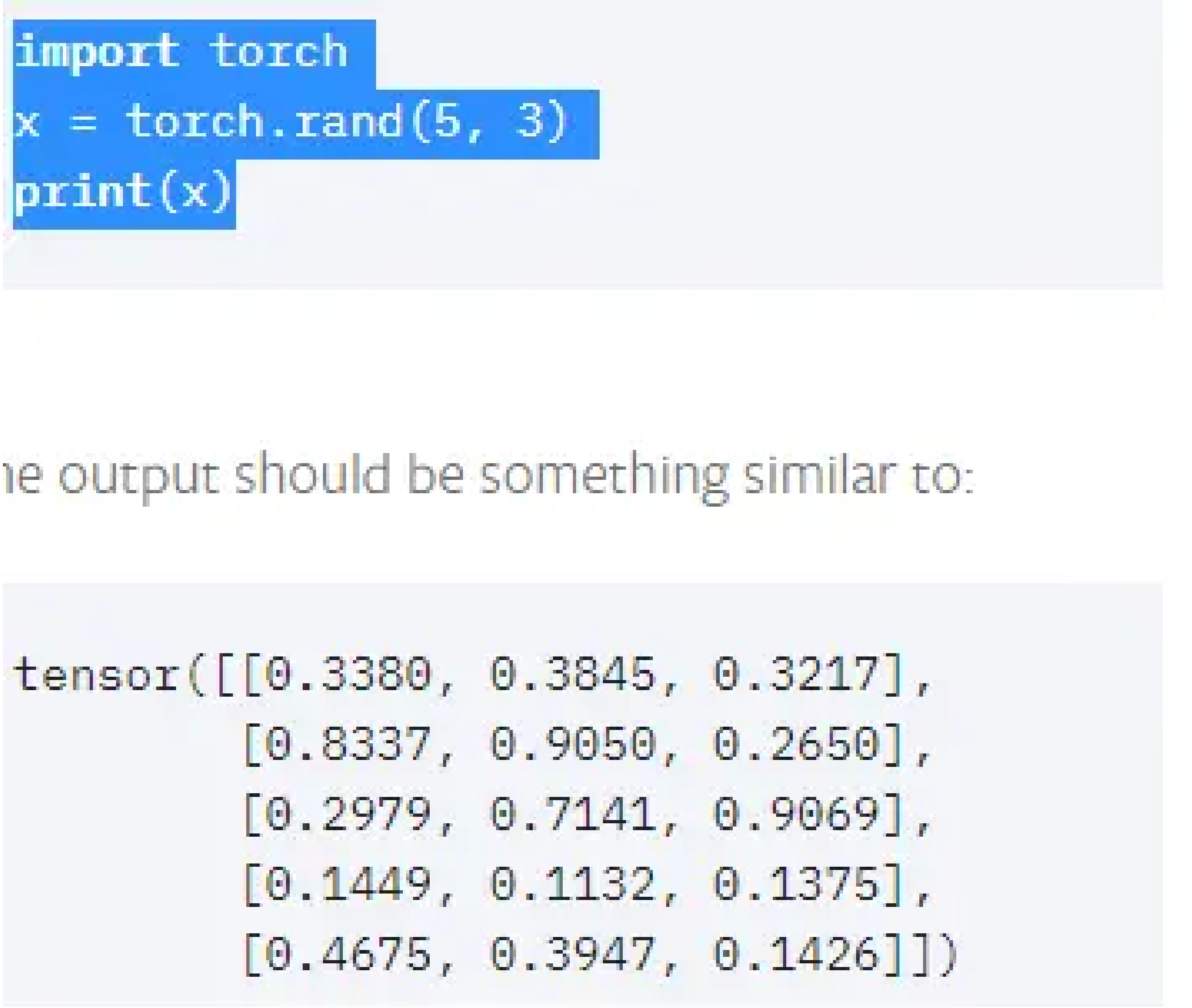
4 torch安装

> pytorch.org/get-started安装：pip3 install torch torchvision torchaudio



PyTorch Build	**Stable (2.0.0)**		Preview (Nightly)	
Your OS	**Linux**	Mac	Windows	
Package	Conda	**Pip**	LibTorch	Source
Language	**Python**		C++ / Java	
Compute Platform	**CUDA 11.7**	CUDA 11.8	ROCm 5.4.2	CPU
Run this Command:	pip3 install torch torchvision torchaudio			

测试：



```
import torch
x = torch.rand(5, 3)
print(x)
```

ıe output should be something similar to:

```
tensor([[0.3380, 0.3845, 0.3217],
        [0.8337, 0.9050, 0.2650],
        [0.2979, 0.7141, 0.9069],
        [0.1449, 0.1132, 0.1375],
        [0.4675, 0.3947, 0.1426]])
```

将例子demo跑起来

环境准备好了之后，切换到项目目录下，直接执行：

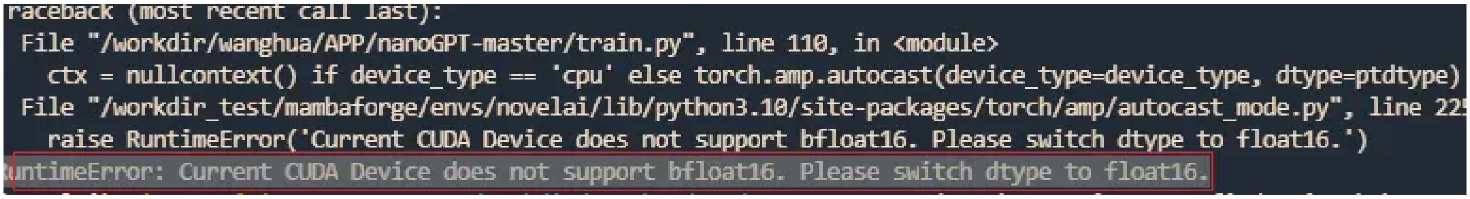
python data/shakespeare_char/prepare.py #数据结构处理python train.py config/train_shakespeare_char.py #训练,有GPU

> 无GPUpython train.py config/train_shakespeare_char.py --device=cpu --compile=False --eval_iters=20 --log_interval=1 --block_size=64 --batch_size=12 --n_layer=4 --n_head=4 --n_embd=128 --max_iters=2000 --lr_de

报错解决：

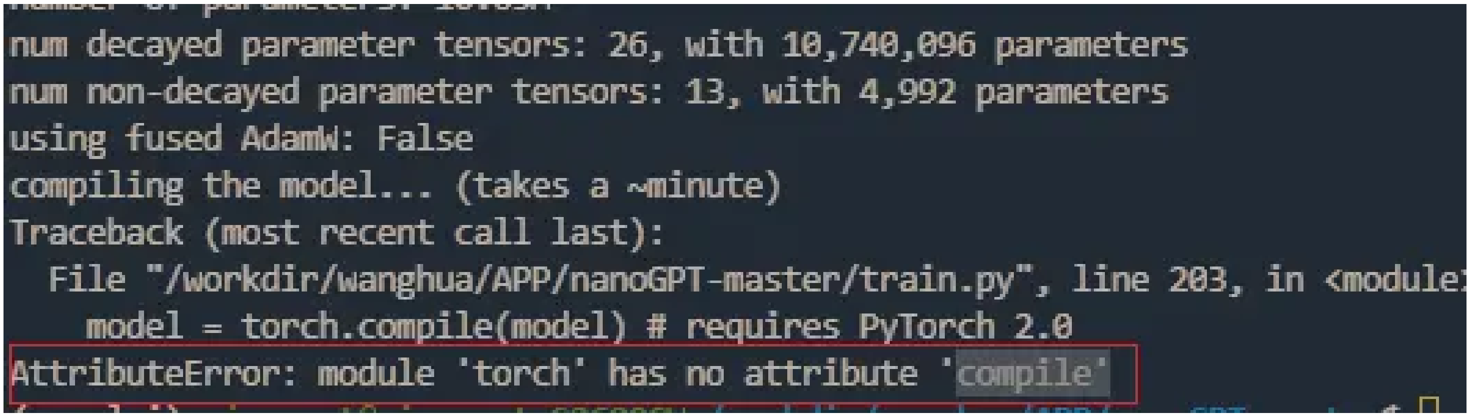
1 error:

RuntimeError: Current CUDA Device does not support bfloat16. Please switch dtype to float16.



```
raceback (most recent call last):
  File "/workdir/wanghua/APP/nanoGPT-master/train.py", line 110, in <module>
    ctx = nullcontext() if device_type == 'cpu' else torch.amp.autocast(device_type=device_type, dtype=ptdtype)
  File "/workdir_test/mambaforge/envs/novelai/lib/python3.10/site-packages/torch/amp/autocast_mode.py", line 22
    raise RuntimeError('Current CUDA Device does not support bfloat16. Please switch dtype to float16.')
untimeError: Current CUDA Device does not support bfloat16. Please switch dtype to float16.
```

AttributeError: module 'torch' has no attribute 'compile'



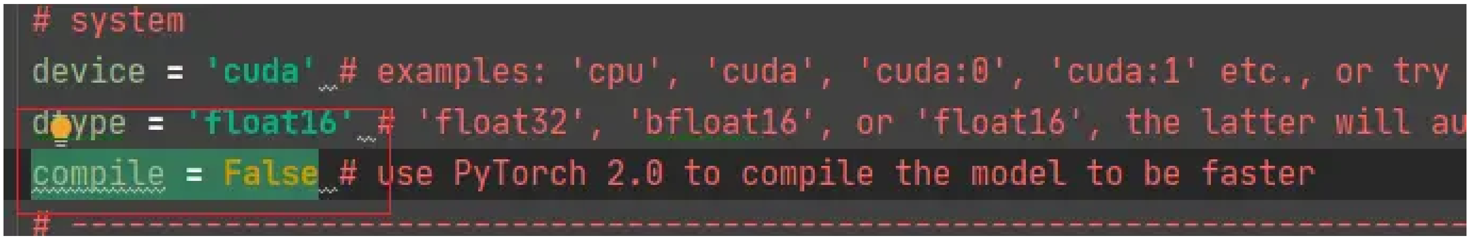
```
num decayed parameter tensors: 26, with 10,740,096 parameters
num non-decayed parameter tensors: 13, with 4,992 parameters
using fused AdamW: False
compiling the model... (takes a ~minute)
Traceback (most recent call last):
  File "/workdir/wanghua/APP/nanoGPT-master/train.py", line 203, in <module
    model = torch.compile(model) # requires PyTorch 2.0
AttributeError: module 'torch' has no attribute 'compile'
```

2 解决以上两个问题：

打开train.py文件，修改如下参数(73行附近）

dtype = 'float16'

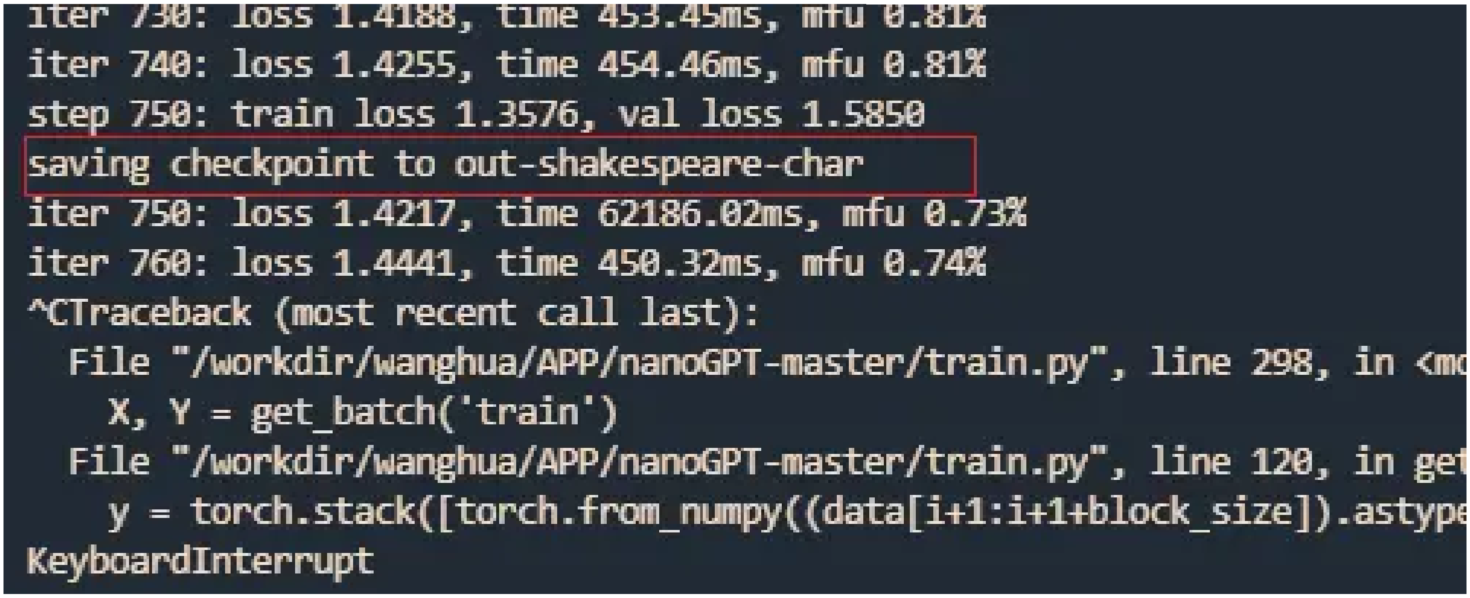
compile = False



```
# system
device = 'cuda' # examples: 'cpu', 'cuda', 'cuda:0', 'cuda:1' etc., or try
dtype = 'float16' # 'float32', 'bfloat16', or 'float16', the latter will au
compile = False # use PyTorch 2.0 to compile the model to be faster
# ----------------------------------------------------------------------------
```

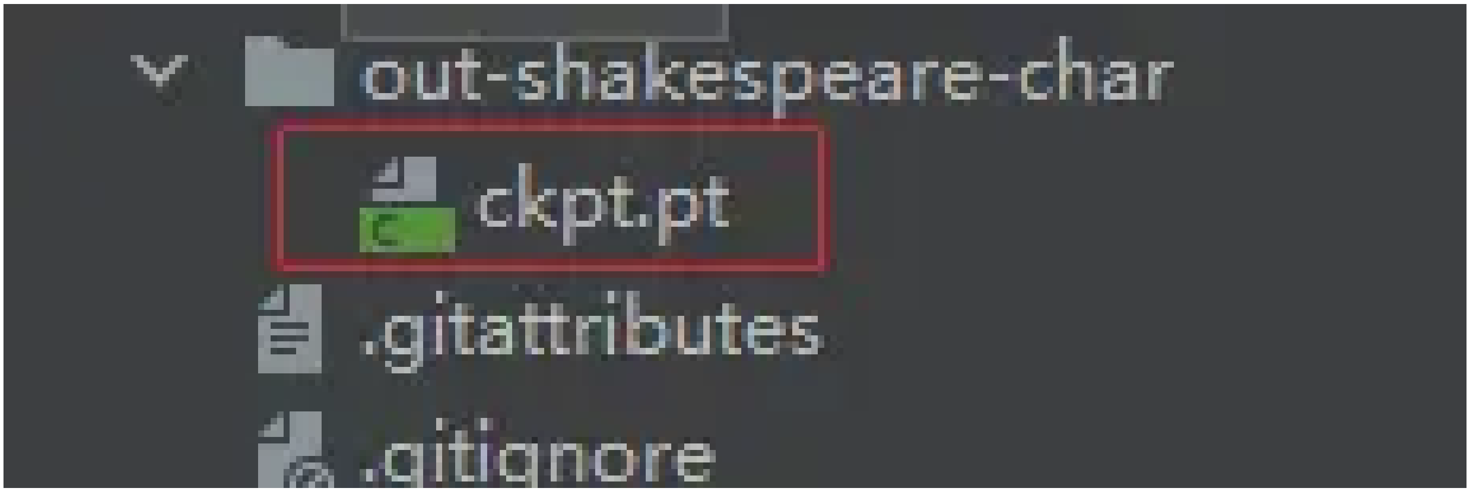
结束标志：

> best validation loss is 1.4697;默认250轮会保存一次模型；在训练差不多可以直接ctrl+c停掉



```
iter 730: loss 1.4188, time 453.45ms, mfu 0.81%
iter 740: loss 1.4255, time 454.46ms, mfu 0.81%
step 750: train loss 1.3576, val loss 1.5850
saving checkpoint to out-shakespeare-char
iter 750: loss 1.4217, time 62186.02ms, mfu 0.73%
iter 760: loss 1.4441, time 450.32ms, mfu 0.74%
^CTraceback (most recent call last):
  File "/workdir/wanghua/APP/nanoGPT-master/train.py", line 298, in <mo
    X, Y = get_batch('train')
  File "/workdir/wanghua/APP/nanoGPT-master/train.py", line 120, in get
    y = torch.stack([torch.from_numpy((data[i+1:i+1+block_size]).astype
KeyboardInterrupt
```

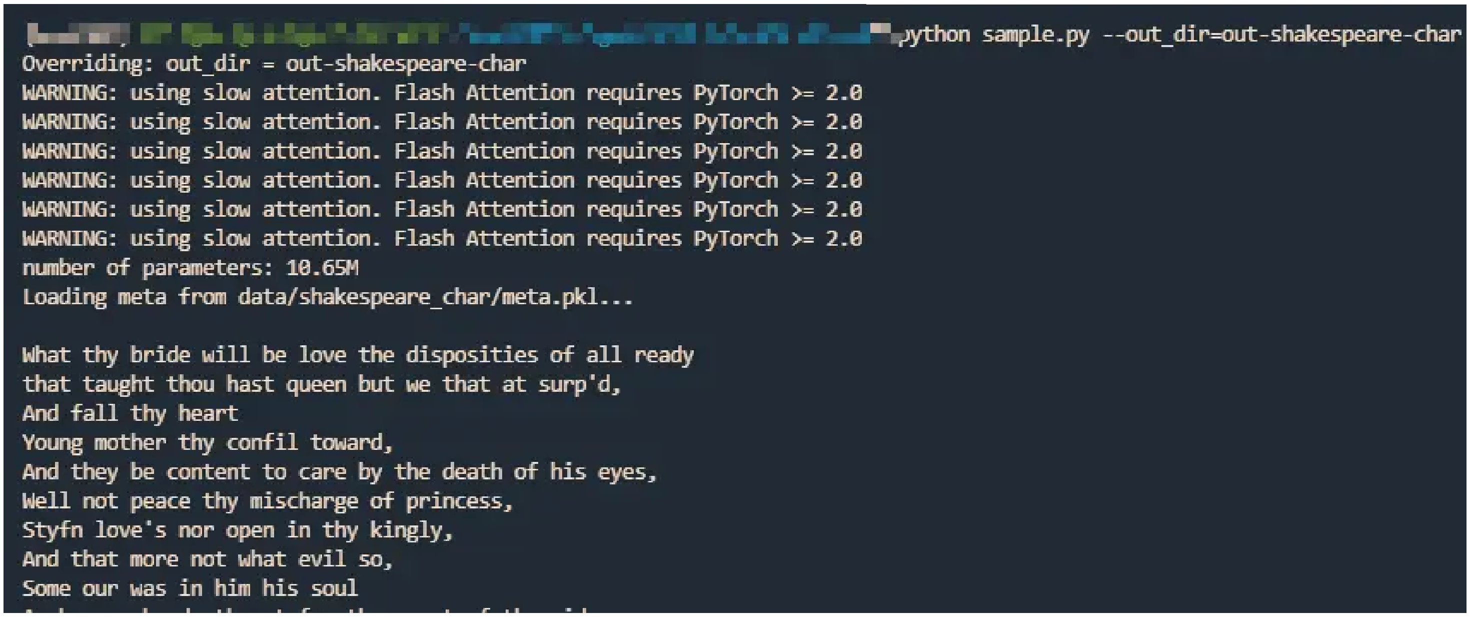
模型保存路径：



运行demo模型(sampling / inference)

python sample.py --out_dir=out-shakespeare-charpython sample.py --out_dir=out-shakespeare-char --device=cpu ## 无gpu

成功标志：



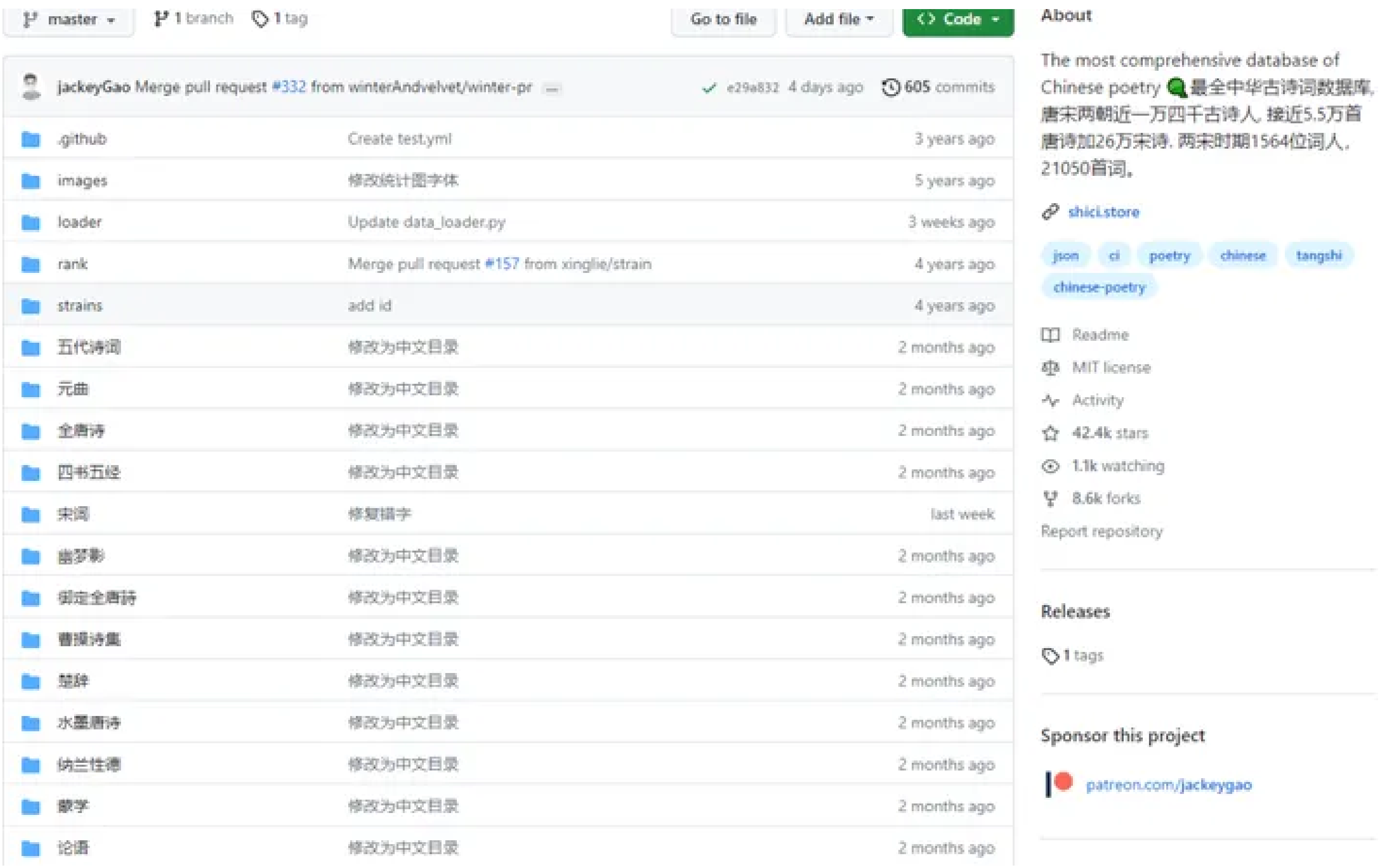
```
[redacted]python sample.py --out_dir=out-shakespeare-char
Overriding: out_dir = out-shakespeare-char
WARNING: using slow attention. Flash Attention requires PyTorch >= 2.0
WARNING: using slow attention. Flash Attention requires PyTorch >= 2.0
WARNING: using slow attention. Flash Attention requires PyTorch >= 2.0
WARNING: using slow attention. Flash Attention requires PyTorch >= 2.0
WARNING: using slow attention. Flash Attention requires PyTorch >= 2.0
WARNING: using slow attention. Flash Attention requires PyTorch >= 2.0
number of parameters: 10.65M
Loading meta from data/shakespeare_char/meta.pkl...

What thy bride will be love the disposities of all ready
that taught thou hast queen but we that at surp'd,
And fall thy heart
Young mother thy confil toward,
And they be content to care by the death of his eyes,
Well not peace thy mischarge of princess,
Styfn love's nor open in thy kingly,
And that more not what evil so,
Some our was in him his soul
```

训练自己的数据集

以下以古诗集为例,训练一个可以续写诗词的gpt模型

以下数据集，下载到根目录 （/nanoGPT-master/XX）

> github.com/chinese-poet最全中华古诗词数据库, 唐宋两朝近一万四千古诗人, 接近5.5万首唐诗加26万宋诗. 两宋时期1564位词人，21050首词



master ▾ 1 branch 1 tag Go to file Add file ▾ <> Code ▾

jackeyGao Merge pull request #332 from winterAndvelvet/winter-pr … ✓ e29a832 4 days ago 605 commits

.github	Create test.yml	3 years ago
images	修改统计图字体	5 years ago
loader	Update data_loader.py	3 weeks ago
rank	Merge pull request #157 from xinglie/strain	4 years ago
strains	add id	4 years ago
五代诗词	修改为中文目录	2 months ago
元曲	修改为中文目录	2 months ago
全唐诗	修改为中文目录	2 months ago
四书五经	修改为中文目录	2 months ago
宋词	修复错字	last week
幽梦影	修改为中文目录	2 months ago
御定全唐詩	修改为中文目录	2 months ago
曹操诗集	修改为中文目录	2 months ago
楚辞	修改为中文目录	2 months ago
水墨唐诗	修改为中文目录	2 months ago
纳兰性德	修改为中文目录	2 months ago
蒙学	修改为中文目录	2 months ago
论语	修改为中文目录	2 months ago

About

The most comprehensive database of Chinese poetry 🔍最全中华古诗词数据库, 唐宋两朝近一万四千古诗人, 接近5.5万首唐诗加26万宋诗. 两宋时期1564位词人, 21050首词。

shici.store

json ci poetry chinese tangshi chinese-poetry

Readme

MIT license

Activity

42.4k stars

1.1k watching

8.6k forks

Report repository

Releases

1 tags

Sponsor this project

patreon.com/jackeygao

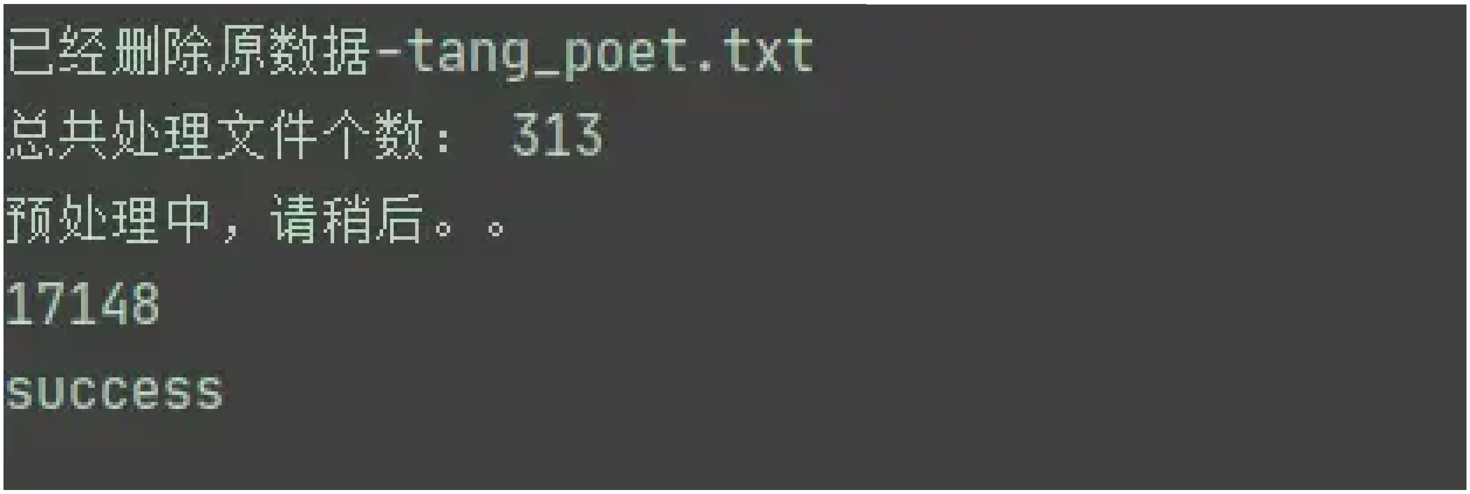
下载的诗词合集太多所以这里以唐诗为示例，所以第一是处理数据，收集所有唐诗相关数据（nanoGPT-master\chinese-poetry-master\全唐诗）并过滤

1 数据预处理

提取相关的唐诗数据

```
# -*- coding: utf-8 -*-
"""
目标：获取数据集中全唐诗，并提取五言诗词，两句的数据
 示例：
{
        "author": "郭向",
        "paragraphs": [
            "抱玉三朝楚，懷書十上秦。",
            "年年洛陽陌，花鳥弄歸人。"
        ],
        "title": "途中口號",
        "id": "32898701-8d9c-4b4d-b192-510564f63b2
```

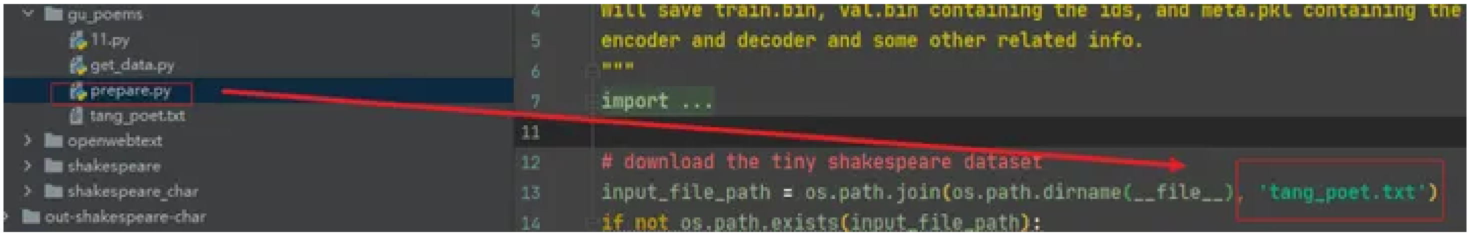
```
    },
"""

import glob
import json
import os
datas_json=glob.glob("../../chinese-poetry-master/全唐诗/poet*.json") #1匹配所有唐诗json
# print(datas_json,"\n",len(datas_json))

if os.path.exists("tang_poet.txt"):
    os.remove("tang_poet.txt")
    print("已经删除原数据-tang_poet.txt")

print("总共处理文件个数：", len(datas_json))
print("预处理中，请稍后。。")
for data_json in datas_json[:]: #2处理匹配的每一个文件

    with open(data_json,"r",encoding="utf-8") as f:
        ts_data =json.load(f)
        for each_ts in ts_data[:]: #3处理文件中每段数据，只要五言诗和2句的
            paragraphs_list =each_ts["paragraphs"]
            if len(paragraphs_list) == 2 and len(paragraphs_list[0])==12 and len(parag
                with open("tang_poet.txt","a",encoding="utf-8") as f2:
                    f2.write("".join(paragraphs_list))
                    f2.write("\n")

f =open("tang_poet.txt","r",encoding="utf-8")
print(len(f.readlines()))
print("success")
```



```
已经删除原数据-tang_poet.txt
总共处理文件个数：  313
预处理中，请稍后。。
17148
success
```

2 生成网络数据

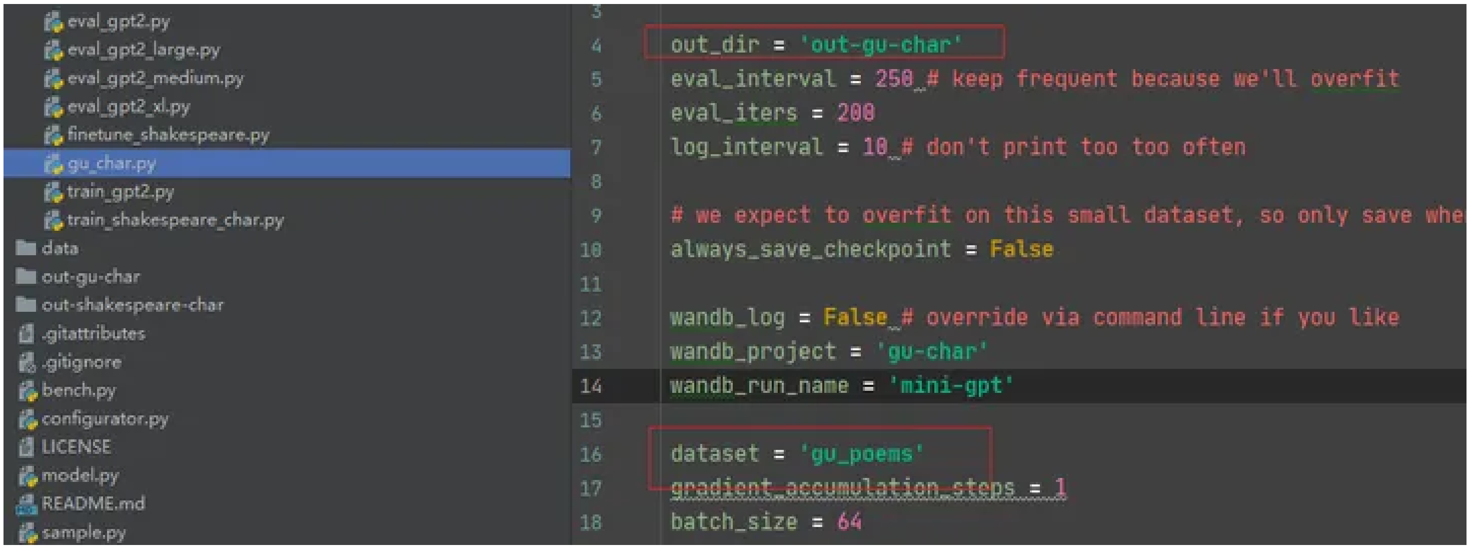
修改prepare.py：



```
will save train.bin, val.bin containing the ids, and meta.pkl containing the
encoder and decoder and some other related info.
"""
import ...

# download the tiny shakespeare dataset
input_file_path = os.path.join(os.path.dirname(__file__), 'tang_poet.txt')
if not os.path.exists(input_file_path):
```

新建config/gu_char.py;内容复制train_shakespeare_char.py，就可以了

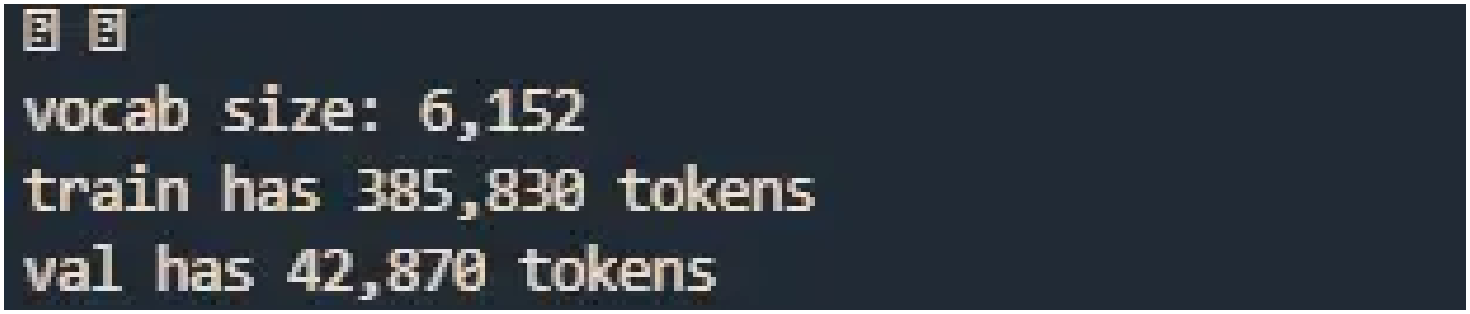
> 修改：out_dir = 'out-gu-char'
> dataset = 'gu_poems'



```

out_dir = 'out-gu-char'
eval_interval = 250 # keep frequent because we'll overfit
eval_iters = 200
log_interval = 10 # don't print too too often

# we expect to overfit on this small dataset, so only save whe
always_save_checkpoint = False

wandb_log = False # override via command line if you like
wandb_project = 'gu-char'
wandb_run_name = 'mini-gpt'

dataset = 'gu_poems'
gradient_accumulation_steps = 1
batch_size = 64
```

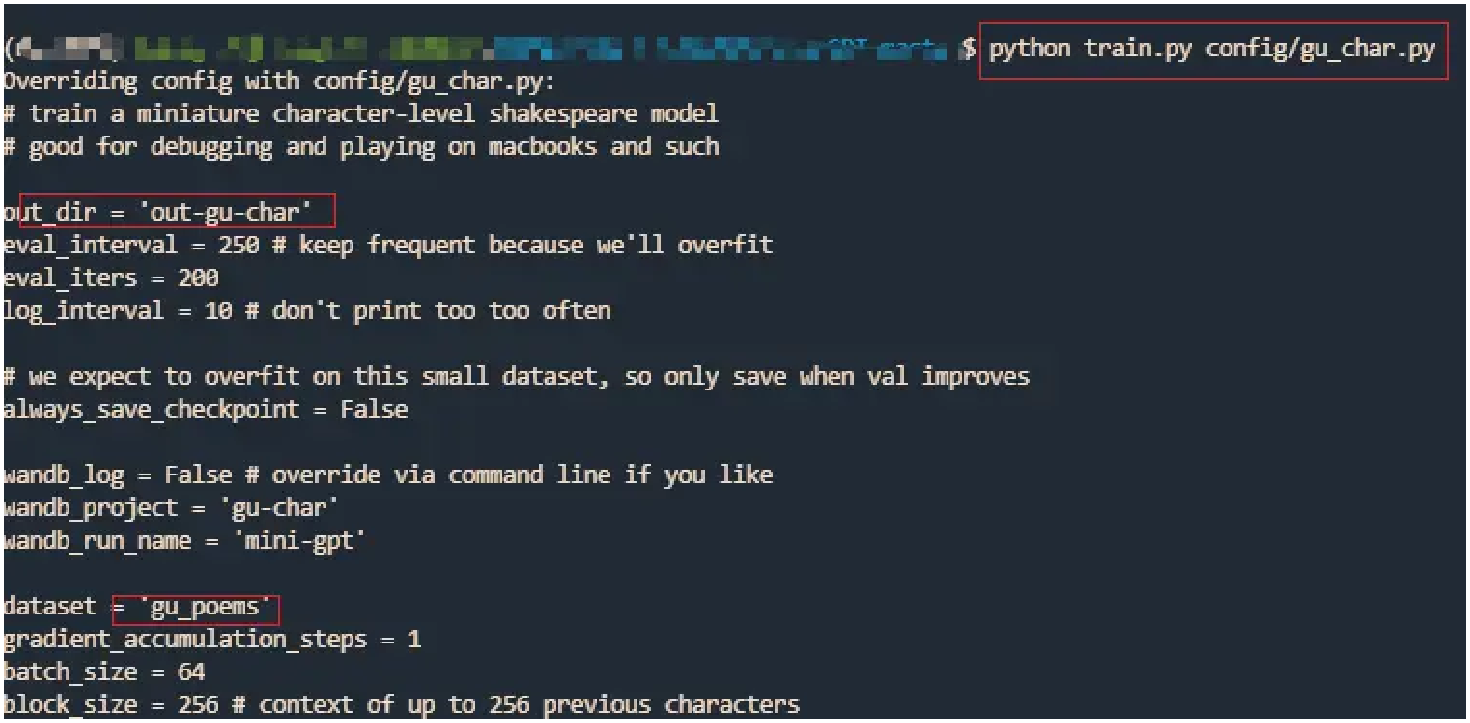
3 训练

python data/gu_poems/prepare.py



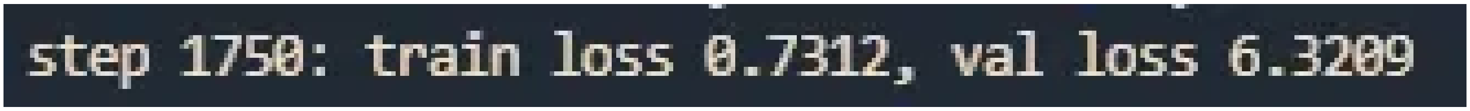
```
目 目
vocab size: 6,152
train has 385,830 tokens
val has 42,870 tokens
```

python train.py config/gu_char.py



```
([redacted] $ python train.py config/gu_char.py
Overriding config with config/gu_char.py:
# train a miniature character-level shakespeare model
# good for debugging and playing on macbooks and such

out_dir = 'out-gu-char'
eval_interval = 250 # keep frequent because we'll overfit
eval_iters = 200
log_interval = 10 # don't print too too often

# we expect to overfit on this small dataset, so only save when val improves
always_save_checkpoint = False

wandb_log = False # override via command line if you like
wandb_project = 'gu-char'
wandb_run_name = 'mini-gpt'

dataset = 'gu_poems'
gradient_accumulation_steps = 1
batch_size = 64
block_size = 256 # context of up to 256 previous characters
```

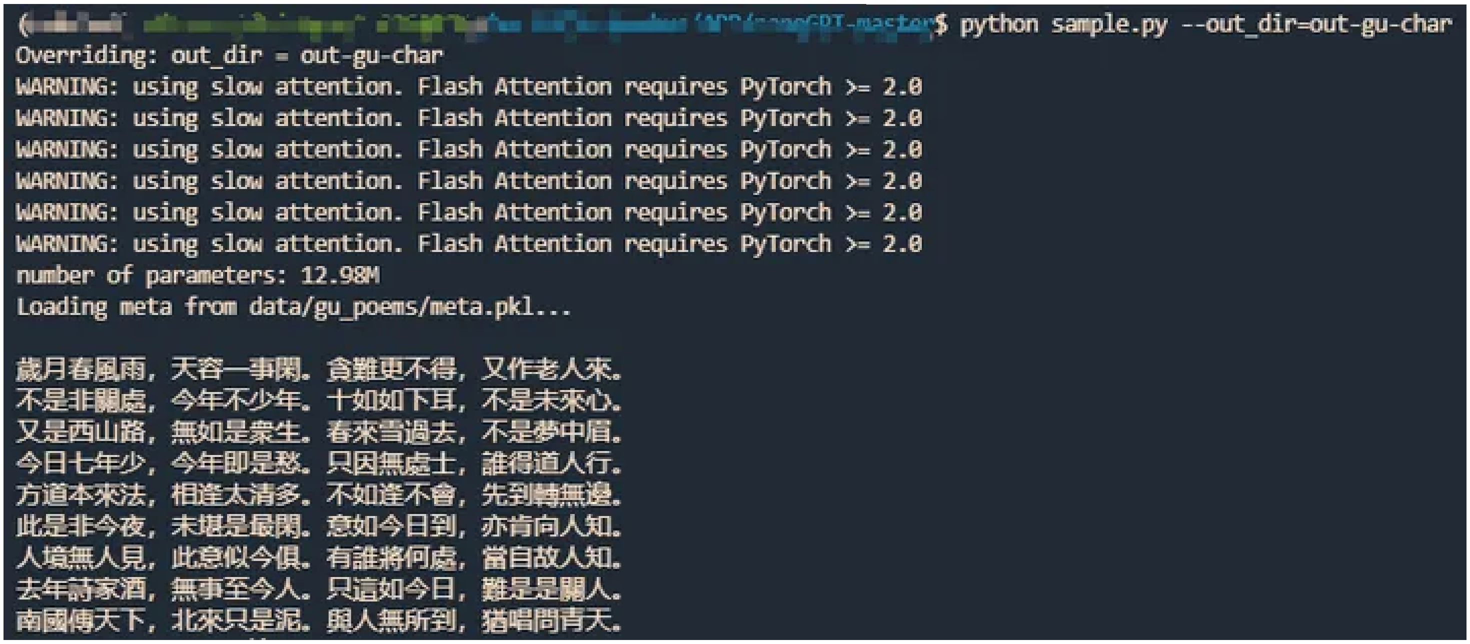
查看train和val的loss 评估模型是否有问题；如果开始val收敛不好，可能数据太少



```
step 1750: train loss 0.7312, val loss 6.3209
```

4 推理

开始续写古诗。。python sample.py --out_dir=out-gu-char



```
[redacted]$ python sample.py --out_dir=out-gu-char
Overriding: out_dir = out-gu-char
WARNING: using slow attention. Flash Attention requires PyTorch >= 2.0
WARNING: using slow attention. Flash Attention requires PyTorch >= 2.0
WARNING: using slow attention. Flash Attention requires PyTorch >= 2.0
WARNING: using slow attention. Flash Attention requires PyTorch >= 2.0
WARNING: using slow attention. Flash Attention requires PyTorch >= 2.0
WARNING: using slow attention. Flash Attention requires PyTorch >= 2.0
number of parameters: 12.98M
Loading meta from data/gu_poems/meta.pkl...

歲月春風雨，天容一事閑。貪難更不得，又作老人來。
不是非關處，今年不少年。十如如下耳，不是未來心。
又是西山路，無如是衆生。春來雪過去，不是夢中眉。
今日七年少，今年即是愁。只因無處士，誰得道人行。
方道本來法，相逢太清多。不如逢不會，先到轉無邊。
此是非今夜，未堪是最閑。意如今日到，亦肯向人知。
人境無人見，此意似今俱。有誰將何處，當自故人知。
去年詩家酒，無事至今人。只這如今日，難是是閑人。
南國傳天下，北來只是泥。與人無所到，猶唱問青天。
```

结果还不错，AI版古诗集已经可以了，或许你可以整理以下去出版自己的ai版古诗集哈哈

文章创作不易，记得关注点赞呐！

发布于 2023-06-02 07:42 · IP 属地广东

GPT 古诗词

写下你的评论...

2 条评论

默认 最新

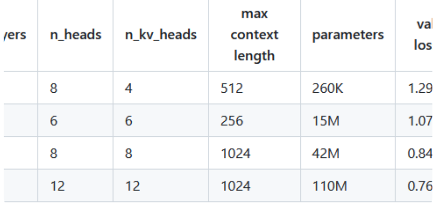
SDUFH

karpathy他的仓库下有一个llama2.c的repo，我按照他的流程下来，模型只有几十mb。一般我们在训练的时候模型的参数量是通过哪些来设置的呀

2023-08-24 · IP 属地浙江 回复 喜欢

教头Lily

我最近一周开始学习了解大模型，从作者列出来的模型可以看出，模型的大小被这些参数决定：dim n_layers n_heads n_kv_heads m



yers	n_heads	n_kv_heads	max context length	parameters	val los
	8	4	512	260K	1.29
	6	6	256	15M	1.07
	8	8	1024	42M	0.84
	12	12	1024	110M	0.76

02-27 · IP 属地福建