中文古诗生成

该项目参考https://github.com/lansinuote/Chinese_Poetry_Generate

使用中文gpt2模型进行古诗文数据集的微调后，通过文本生成任务完成古诗生成。

使用方法为输入1-5个字作为古诗开头，选择生成“五言绝句”或者“七言绝句”，点击按钮生成古诗。

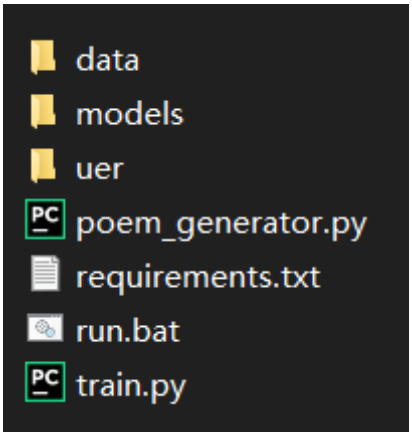
总体使用效果还可以，可以生成完整连贯的诗句。

文件构成及环境

环境：

torch~=1.10.2+cu113
pandas~=1.2.4
tqdm~=4.59.0
matplotlib~=3.4.3
transformers~=4.25.1
pyqt5~=5.15.4

文件构成：



Data文件夹为训练所使用的数据集。

models文件中储存了训练完成的模型。

uer中保存了gpt2的tokenizer，若想自己进行训练，请下载uer/gpt2-chinese-cluecorpussmall模型放入uer/gpt2-chinese-cluecorpussmall文件夹中。

poem_generator.py为项目的源码。

requirements.txt为需要的环境。

train.py为训练模型源码文件。

点击run.bat开始运行。

模型训练

本项目在uer/gpt2-chinese-cluecorpussmall模型上进行微调：https://huggingface.co/uer/gpt2-chinese-cluecorpussmall

本项目针对两种不同的训练集“唐.csv”和“chinese_poems.txt”使用了两种方法训练模型，但是两种方法都使用了AdamW优化器，并且设置学习率为5e-5。

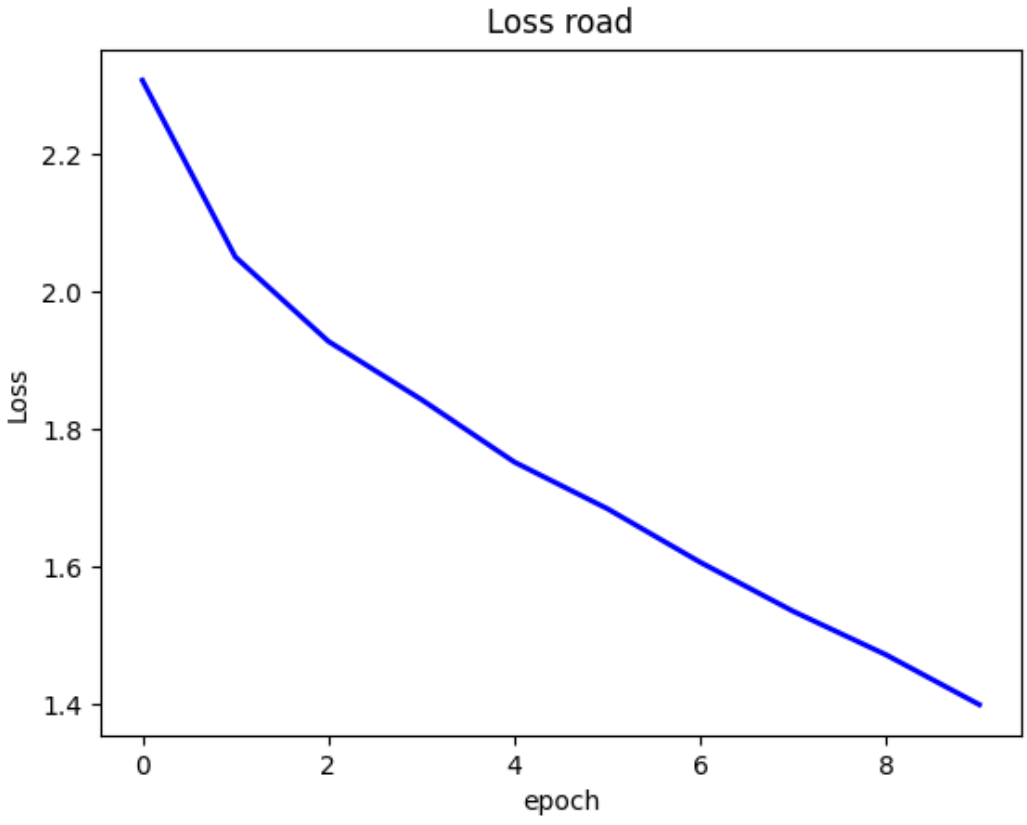
调度器作用是在训练中动态调整学习率，大学习率有助于模型快速收敛，小学习率有助于模型收敛到更高精度的局部极小值。本实验使用了抛弃预热阶段的线性调度器，即学习率一直以如下公式减小，其中γ是设定的常数：

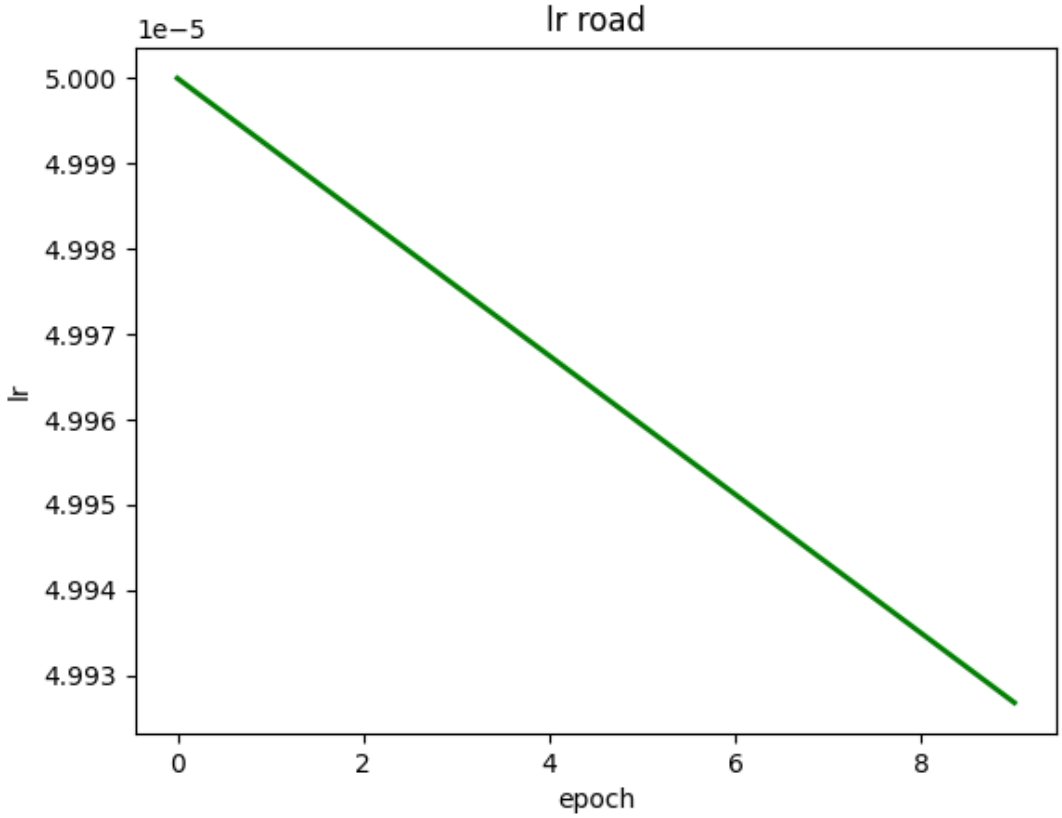
$$lr^{new} = lr * \gamma$$

model2

在使用唐.csv训练集训练时，因为训练集训练本身较小，为了保证每个数据都能对模型产生同等地位的影响，只在每一个epoch（训练完一次全部数据）之后使用线性调度器降低学习率。

一共训练了10个epoch，以下为loss和学习率的变化，可以观察到loss在稳定降低，说明训练有效：

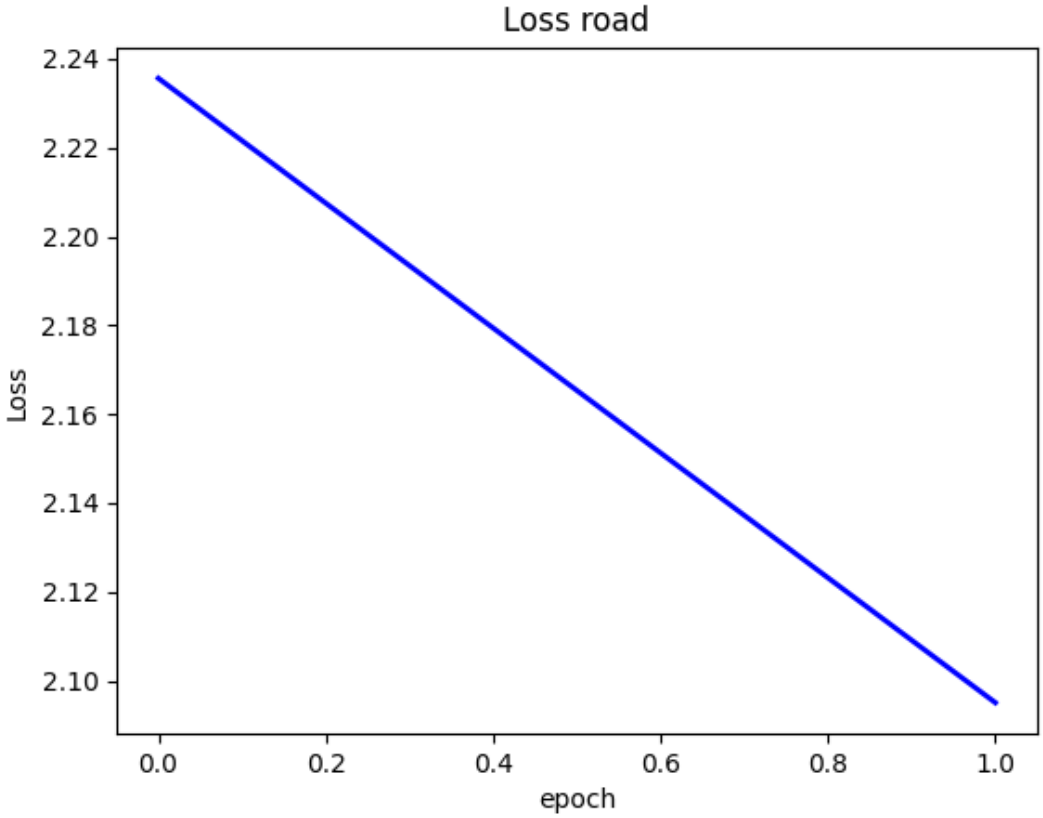




model1

因为model1的使用的训练集chinese_poems.txt数据量很大，所以样本地位有稍微的不均匀也是可以接受的，并且因为训练epoch数量少，所以每一个step（训练一条数据）便使用线性调度器降低学习率，一共训练了2个epoch。

以下为loss变化，可以观察到loss在稳定降低，说明训练有效：



模型效果

两个模型所训练的总step是差不多的，并且二者都能流畅地生成诗句。但是由于model1所使用的训练集除了古诗还包含其他的不规律文章，model2的唐诗训练数据格律严格，微调的epoch多，反复强调了训练数据的格律，所以model2在格律的表现上更好，比如2，4句的押韵。


古诗生成器
请输入1-5个字，并选择生成类型，点击
按钮生成故事
可以使用如下测试文本：
五言绝句
model1
秋去冬来
生成古诗
秋 去 冬 来 无，
秋 来 秋 来 无。
一 年 长 有 几，
万 事 不 关 心.


古诗生成器
请输入1-5个字，并选择生成类型，点击
按钮生成故事
可以使用如下测试文本：
五言绝句
model2
秋去冬来
生成古诗
秋 去 冬 来 至 ，
寒 云 半 碛 空 。
雁 惊 沙 碛 下 ，
马 踏 雪 山 中 。


古诗生成器
请输入1-5个字，并选择生成类型，点击
按钮生成故事
可以使用如下测试文本：
五言绝句
model2
秋去冬来
生成古诗
秋 去 冬 来 渐 ，
月 明 三 四 轮 。
病 多 行 爱 睡 ，
老 少 卧 知 春 。