National University of Sciences and Technology

School of Electrical Engineering and Computer Science

Department of Computing

CS-405 Deep Learning

Fall 2023

Lab 11

Lab11_CS405_Deep_Learning

GPT-2 Fine tunning

Date: 30th Nov 2023

Submitted by:

NAME: Kamlesh Kumar
CMS: 348395
CLASS: BSCS-10C
link

Instructor: Bostan Khan/Dr Daud Abdullah

Deep Learning

Lab-11 Report

Problem statement:

In this task, we shall give a prompt to the model, and the model must learn how to generate the coherent story of the prompt. To solve this problem I have used the power of transfer learning, specifically fine-tuning the DistilGPT-2 model on the following dataset.

About Dataset

The WritingPrompts dataset is our treasure trove for this undertaking. It consists of two key components: "prompts" and "stories." These components are further divided into subsets for each stage of the experiment (train, validation, and test). The "prompts" provide the starting point for our stories, while the "stories" serve as the gold standard, offering exemplary completions to these prompts.

I have created **custom_dataset** class to each datapoint. Before loading the each dataset_point I have combined the prompt+stories, soo that we can give them at once to our language model (distilgpt). So the template looks like:

```
      prompt = """prompt :
      One morning, you wake up in a world where everyone can hear each
other's thoughts.
      story:
      """
```

Preprocessing:

In the preprocessing phase, I executed several critical tasks to ensure that our dataset is well-prepared for training. Firstly, I used the ‘sed; command to efficiently remove any extra brackets from the source files, that are irrelevant for our case.

Following that, I carried out a meticulous text cleaning process, replacing specific punctuation marks and spaces, which significantly enhanced the quality of our dataset. I made sure to apply these operations to both the source and target files, ensuring consistency throughout the dataset. Lastly, I processed the combined text to further refine its quality.

One important thing is that I am cleaning the punctuations from combined text when the model is taking the data for encoding. So I have called the **clear_punctuation** function in **custom_dataset** class.

Model Selection

In the model selection phase, I chose the "distilgpt2" model as our base model for fine-tuning. To facilitate text processing, I used the **GPT2Tokenizer** from the Hugging Face Transformers library, ensuring that it can handle padding efficiently.

Additionally, I loaded the **GPT2LMHeadModel**, setting it up with the pre-trained weights from "distilgpt2." This model selection process allows us to leverage the capabilities of the DistilGPT-2 architecture for our story generation task.

Training:

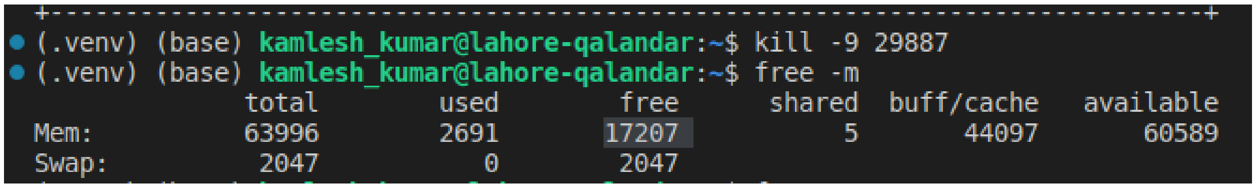
For the model training phase, I established essential hyperparameters to guide the fine-tuning process. The training_args object includes key settings and hyperparameters such as:

- Number of Training Epochs: 10
- Gradient Accumulation Steps: 4
- Learning Rate: 2e-3
- Learning Rate Scheduler Type: "cosine"
- Save Total Limit: 3
- Per-device Training Batch Size: 8
- Per-device Evaluation Batch Size: 8
- Evaluation Steps: 500
- Warmup Ratio: 0.05
- Logging Strategy: "steps"
- Logging Steps: 500

Then I used `Trainer from transformer` to provide the train and eval dataset and it started training.

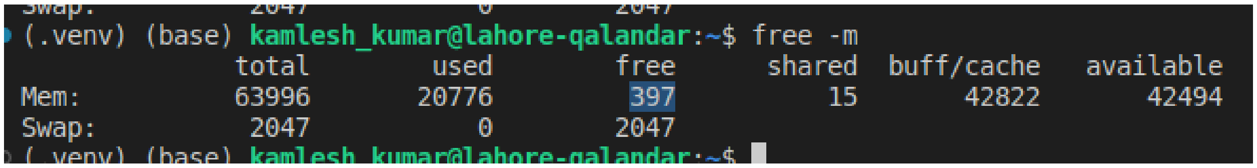
!! 1st trail to load the whole dataset into memory and then start the training.

Memory before loading the train and val dataset in **encoded form i.e. input ids and attention masks** : 17GB avaliable. In order to avoid the encoding during the training process for each batch.



```
● (.venv) (base) kamlesh_kumar@lahore-qalandar:~$ kill -9 29887
● (.venv) (base) kamlesh_kumar@lahore-qalandar:~$ free -m
               total        used        free      shared  buff/cache   available
 Mem:          63996        2691       17207           5       44097       60589
 Swap:          2047           0        2047
```

After 15 minutes of loading the data, memory remaining was 387 mbs only.

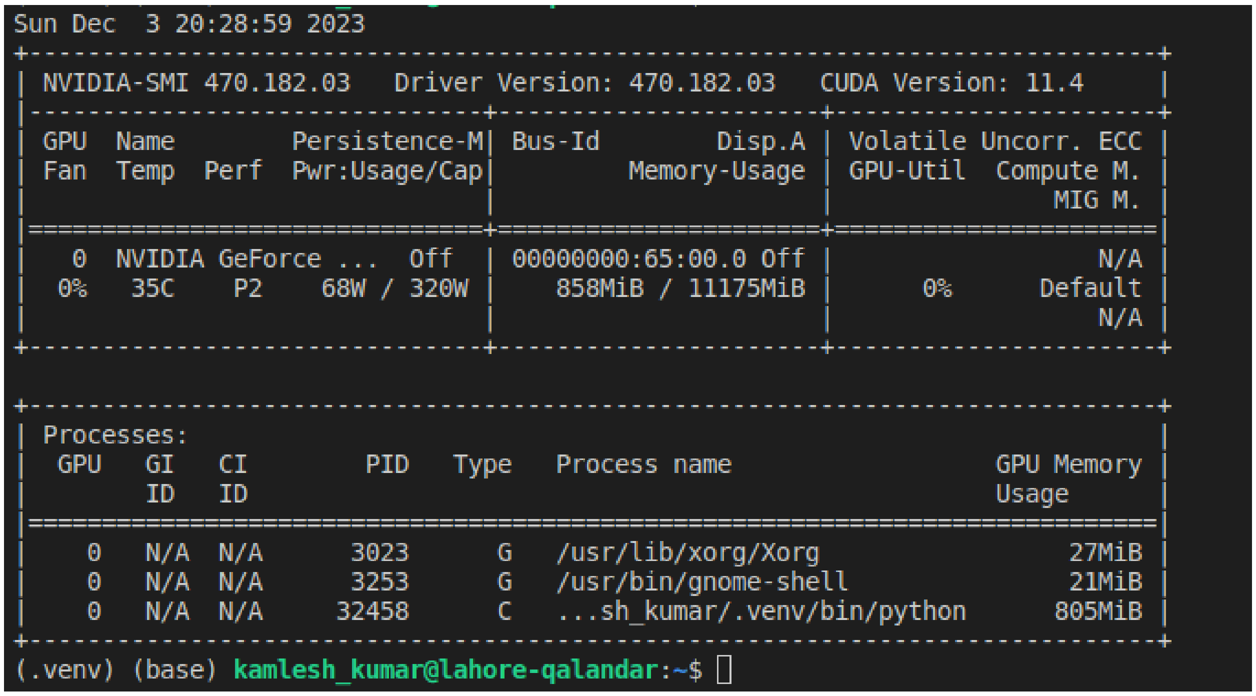


```
 (.venv) (base) kamlesh_kumar@lahore-qalandar:~$ free -m
               total        used        free      shared  buff/cache   available
 Mem:          63996       20776         397          15       42822       42494
 Swap:          2047           0        2047
 (.venv) (base) kamlesh_kumar@lahore-qalandar:~$
```

Conclusion: Even after waiting for 1 hour around 65 minutes It did not load the whole encoded data into memory.

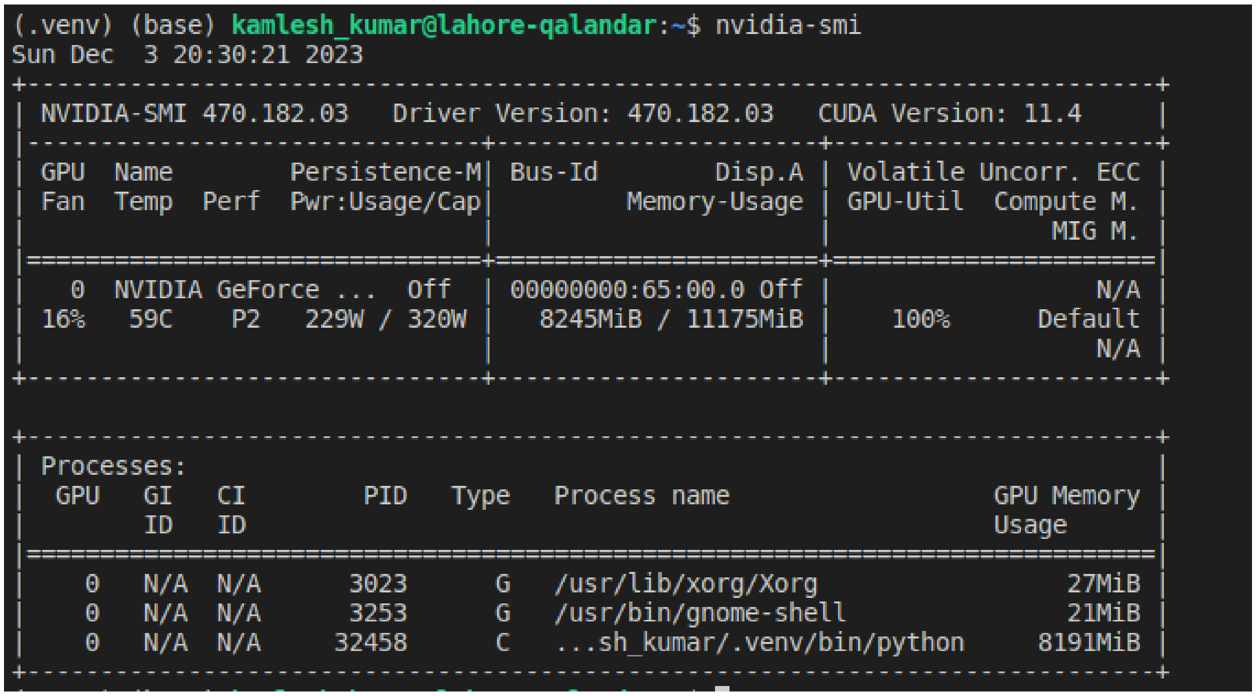
So now try the training on fly i.e. loading and encoding data while training.

GPU Before training:



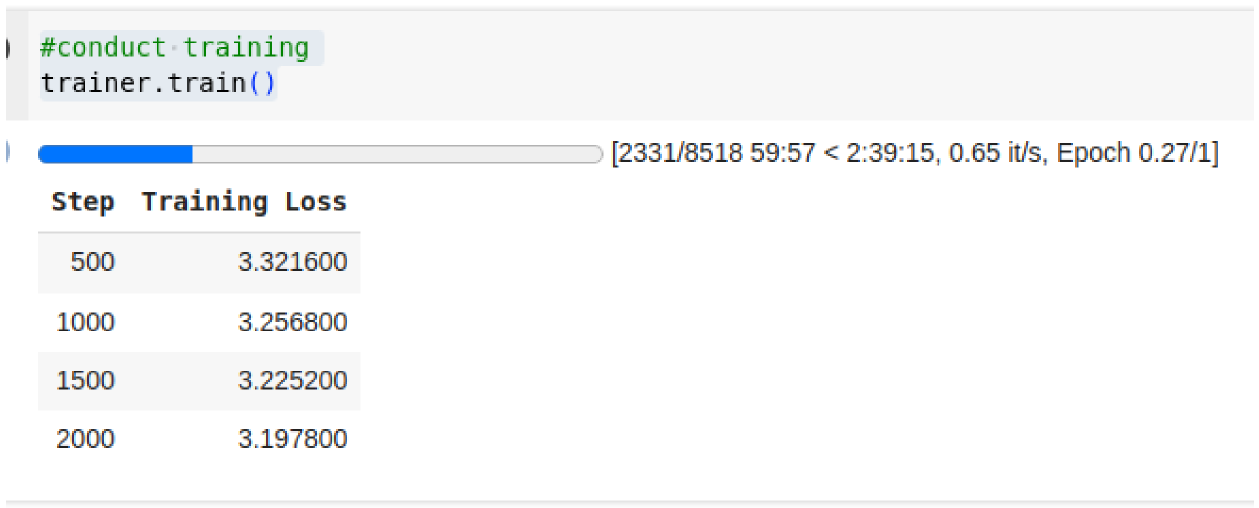
```
Sun Dec  3 20:28:59 2023
+-----------------------------------------------------------------------------+
| NVIDIA-SMI 470.182.03   Driver Version: 470.182.03   CUDA Version: 11.4     |
|-------------------------------+----------------------+----------------------+
| GPU  Name        Persistence-M| Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan  Temp  Perf  Pwr:Usage/Cap|         Memory-Usage | GPU-Util  Compute M. |
|                               |                      |               MIG M. |
|===============================+======================+======================|
|   0  NVIDIA GeForce ...  Off  | 00000000:65:00.0 Off |                  N/A |
|  0%   35C    P2    68W / 320W |    858MiB / 11175MiB |      0%      Default |
|                               |                      |                  N/A |
+-------------------------------+----------------------+----------------------+

+-----------------------------------------------------------------------------+
| Processes:                                                                  |
|  GPU   GI   CI        PID   Type   Process name                  GPU Memory |
|        ID   ID                                                   Usage      |
|=============================================================================|
|    0   N/A  N/A      3023      G   /usr/lib/xorg/Xorg                 27MiB |
|    0   N/A  N/A      3253      G   /usr/bin/gnome-shell               21MiB |
|    0   N/A  N/A     32458      C   ...sh_kumar/.venv/bin/python      805MiB |
+-----------------------------------------------------------------------------+
(.venv) (base) kamlesh_kumar@lahore-qalandar:~$
```

GPU during training: Around 8GB gpu is required.



```
(.venv) (base) kamlesh_kumar@lahore-qalandar:~$ nvidia-smi
Sun Dec  3 20:30:21 2023
+-----------------------------------------------------------------------------+
| NVIDIA-SMI 470.182.03   Driver Version: 470.182.03   CUDA Version: 11.4     |
|-------------------------------+----------------------+----------------------+
| GPU  Name        Persistence-M| Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan  Temp  Perf  Pwr:Usage/Cap|         Memory-Usage | GPU-Util  Compute M. |
|                               |                      |               MIG M. |
|===============================+======================+======================|
|   0  NVIDIA GeForce ...  Off  | 00000000:65:00.0 Off |                  N/A |
| 16%   59C    P2   229W / 320W |   8245MiB / 11175MiB |    100%      Default |
|                               |                      |                  N/A |
+-------------------------------+----------------------+----------------------+

+-----------------------------------------------------------------------------+
| Processes:                                                                  |
|  GPU   GI   CI        PID   Type   Process name                  GPU Memory |
|        ID   ID                                                   Usage      |
|=============================================================================|
|    0   N/A  N/A      3023      G   /usr/lib/xorg/Xorg                 27MiB |
|    0   N/A  N/A      3253      G   /usr/bin/gnome-shell               21MiB |
|    0   N/A  N/A     32458      C   ...sh_kumar/.venv/bin/python     8191MiB |
+-----------------------------------------------------------------------------+
```

Training got stuck in between, after running ¼ of the epoch, because the session with ssh get disconnected.



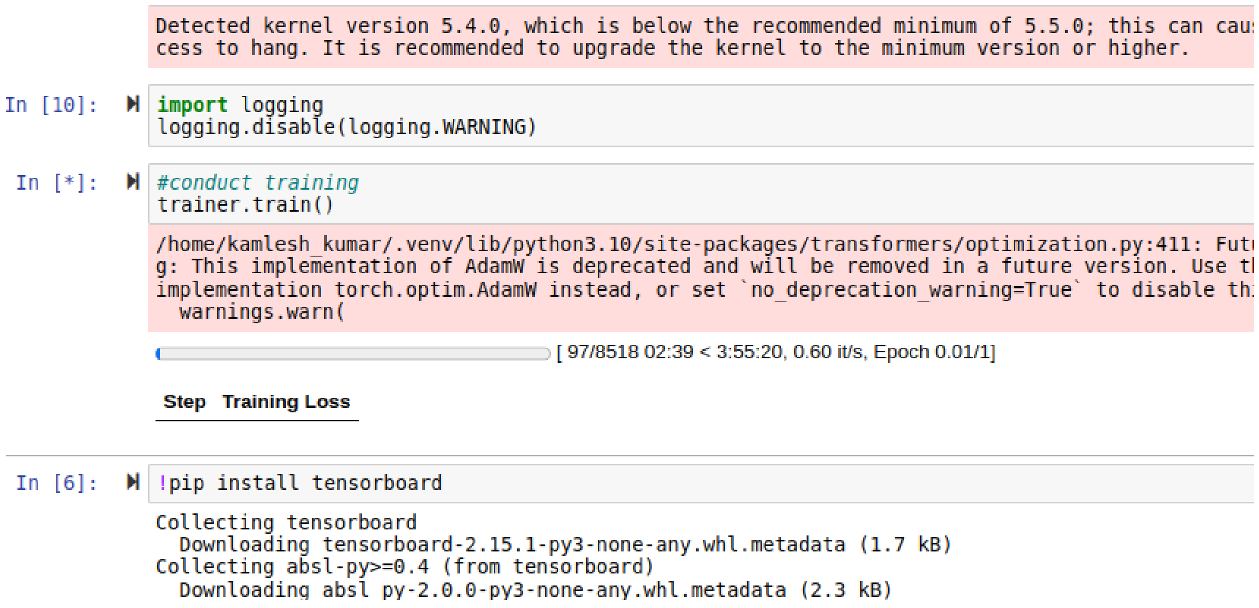
```
#conduct training
trainer.train()
```

[2331/8518 59:57 < 2:39:15, 0.65 it/s, Epoch 0.27/1]

Step	Training Loss
500	3.321600
1000	3.256800
1500	3.225200
2000	3.197800

To solve that I ran a “Jupyter notebook server” on the remote server and accessed the jupyter notebook using browser on my local machine with help of remote ssh tunnel. And started the training remotely and detached my terminal.

Following is the start of training on remote “Jupyter notebook server”.

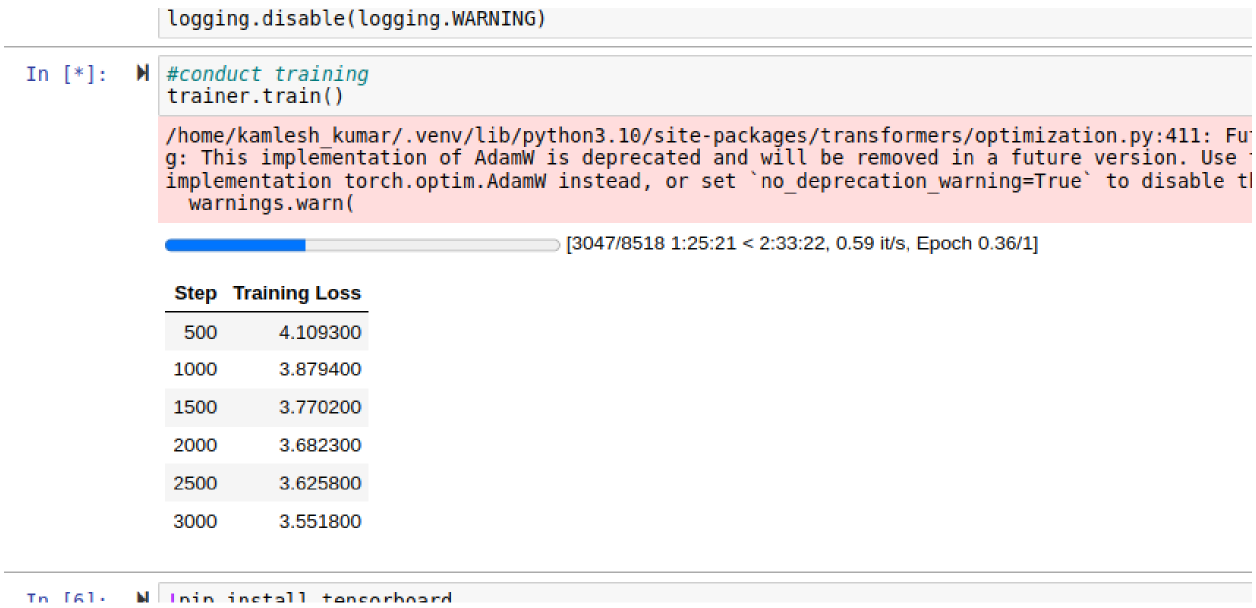


```
Detected kernel version 5.4.0, which is below the recommended minimum of 5.5.0; this can cau
cess to hang. It is recommended to upgrade the kernel to the minimum version or higher.

In [10]:  import logging
          logging.disable(logging.WARNING)

In [*]:   #conduct training
          trainer.train()

/home/kamlesh_kumar/.venv/lib/python3.10/site-packages/transformers/optimization.py:411: Futu
g: This implementation of AdamW is deprecated and will be removed in a future version. Use th
implementation torch.optim.AdamW instead, or set `no_deprecation_warning=True` to disable thi
  warnings.warn(
```

[97/8518 02:39 < 3:55:20, 0.60 it/s, Epoch 0.01/1]

Step	Training Loss

```
In [6]:   !pip install tensorboard

Collecting tensorboard
  Downloading tensorboard-2.15.1-py3-none-any.whl.metadata (1.7 kB)
Collecting absl-py>=0.4 (from tensorboard)
  Downloading absl_py-2.0.0-py3-none-any.whl.metadata (2.3 kB)
```



```
logging.disable(logging.WARNING)
```

```
In [*]:  #conduct training
         trainer.train()
```

```
/home/kamlesh_kumar/.venv/lib/python3.10/site-packages/transformers/optimization.py:411: Fu
g: This implementation of AdamW is deprecated and will be removed in a future version. Use
implementation torch.optim.AdamW instead, or set `no_deprecation_warning=True` to disable t
  warnings.warn(
```

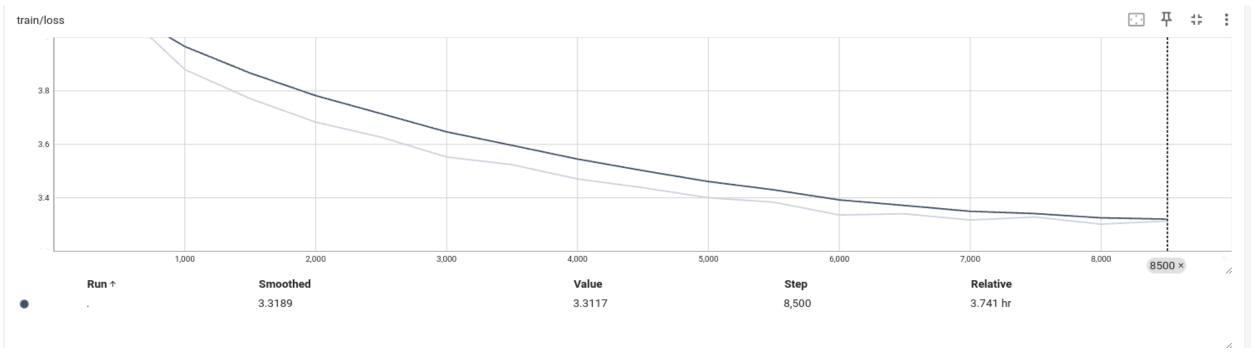
```
[3047/8518 1:25:21 < 2:33:22, 0.59 it/s, Epoch 0.36/1]
```

Step	Training Loss
500	4.109300
1000	3.879400
1500	3.770200
2000	3.682300
2500	3.625800
3000	3.551800

```
In [6]:  !pip install tensorboard
```

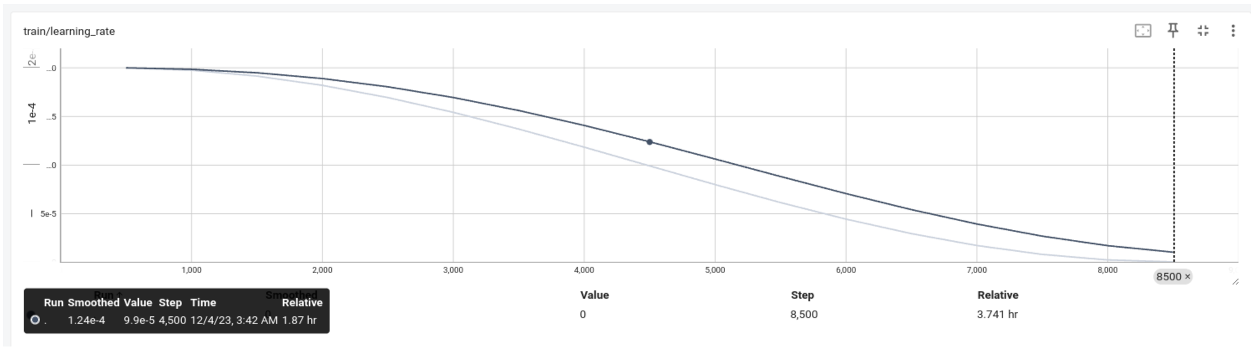
Evaluation while training with whole dataset and 1 epoch:

Train/test loss:



	Run ↑	Smoothed	Value	Step	Relative
●	.	3.3189	3.3117	8,500	3.741 hr

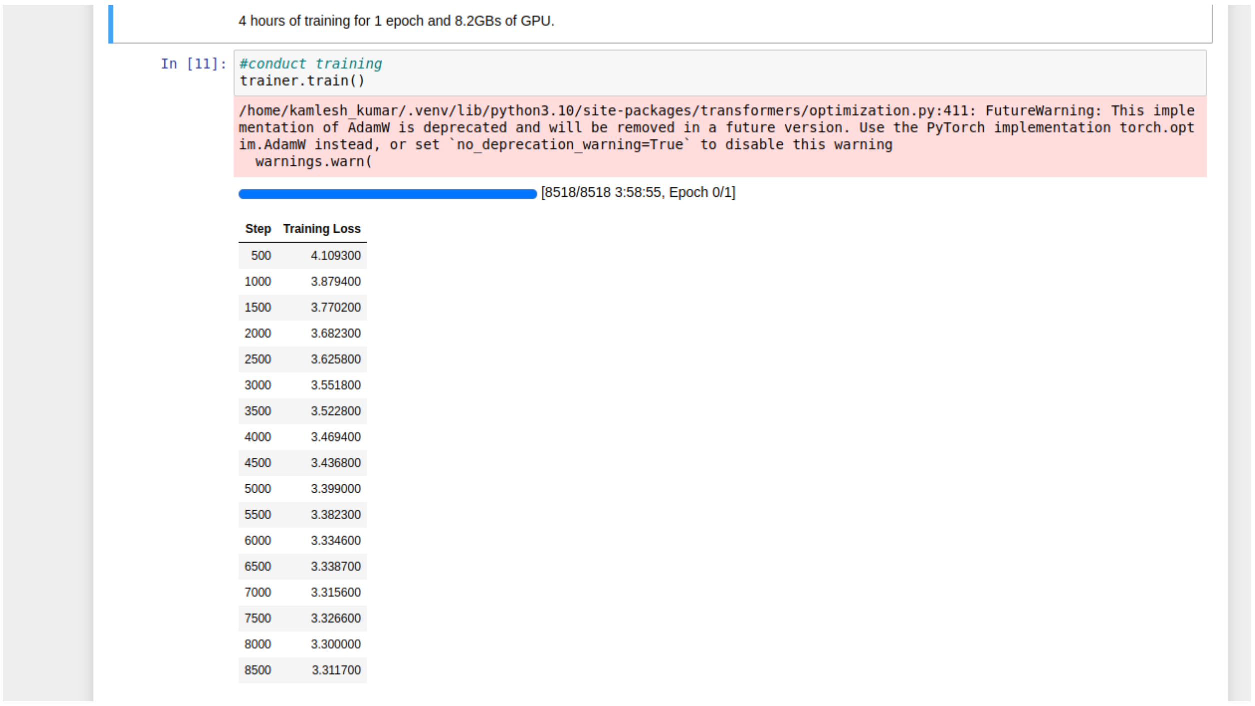
Train/learning_rate: Learning rate scheduler in action.



Run	Smoothed	Value	Step	Time	Relative
.	1.24e-4	9.9e-5	4,500	12/4/23, 3:42 AM	1.87 hr

Value	Step	Relative
0	8,500	3.741 hr

Complete training with all dataset and 1 epoch.



4 hours of training for 1 epoch and 8.2GBs of GPU.

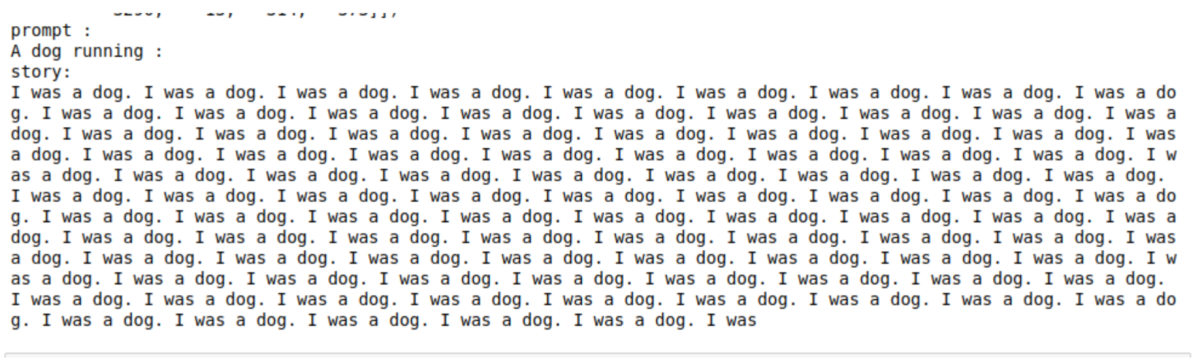
```
In [11]: #conduct training
         trainer.train()
```

```
/home/kamlesh_kumar/.venv/lib/python3.10/site-packages/transformers/optimization.py:411: FutureWarning: This imple
mentation of AdamW is deprecated and will be removed in a future version. Use the PyTorch implementation torch.opt
im.AdamW instead, or set `no_deprecation_warning=True` to disable this warning
  warnings.warn(
```

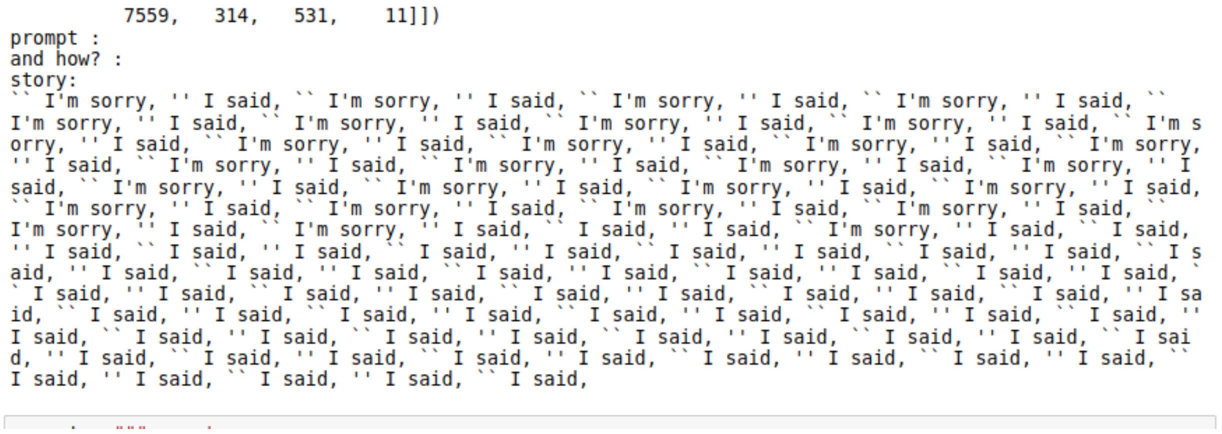
[8518/8518 3:58:55, Epoch 0/1]

Step	Training Loss
500	4.109300
1000	3.879400
1500	3.770200
2000	3.682300
2500	3.625800
3000	3.551800
3500	3.522800
4000	3.469400
4500	3.436800
5000	3.399000
5500	3.382300
6000	3.334600
6500	3.338700
7000	3.315600
7500	3.326600
8000	3.300000
8500	3.311700

Output Generation:



```
prompt :
A dog running :
story:
I was a dog. I was a dog. I was a dog. I was a dog. I was a dog. I was a dog. I was a dog. I was a dog. I was a do
g. I was a dog. I was a dog. I was a dog. I was a dog. I was a dog. I was a dog. I was a dog. I was a dog. I was a
dog. I was a dog. I was a dog. I was a dog. I was a dog. I was a dog. I was a dog. I was a dog. I was a dog. I was
a dog. I was a dog. I was a dog. I was a dog. I was a dog. I was a dog. I was a dog. I was a dog. I was a dog. I w
as a dog. I was a dog. I was a dog. I was a dog. I was a dog. I was a dog. I was a dog. I was a dog. I was a dog.
I was a dog. I was a dog. I was a dog. I was a dog. I was a dog. I was a dog. I was a dog. I was a dog. I was a do
g. I was a dog. I was a dog. I was a dog. I was a dog. I was a dog. I was a dog. I was a dog. I was a dog. I was a
dog. I was a dog. I was a dog. I was a dog. I was a dog. I was a dog. I was a dog. I was a dog. I was a dog. I was
a dog. I was a dog. I was a dog. I was a dog. I was a dog. I was a dog. I was a dog. I was a dog. I was a dog. I w
as a dog. I was a dog. I was a dog. I was a dog. I was a dog. I was a dog. I was a dog. I was a dog. I was a dog.
I was a dog. I was a dog. I was a dog. I was a dog. I was a dog. I was a dog. I was a dog. I was a dog. I was a do
g. I was a dog. I was a dog. I was a dog. I was a dog. I was a dog. I was
```



```
         7559,    314,    531,     11]])
prompt :
and how? :
story:
`` I'm sorry, '' I said, `` I'm sorry, '' I said, `` I'm sorry, '' I said, `` I'm sorry, '' I said, ``
I'm sorry, '' I said, `` I'm sorry, '' I said, `` I'm sorry, '' I said, `` I'm sorry, '' I said, `` I'm s
orry, '' I said, `` I'm sorry, '' I said, `` I'm sorry, '' I said, `` I'm sorry, '' I said, `` I'm sorry,
'' I said, `` I'm sorry, '' I said, `` I'm sorry, '' I said, `` I'm sorry, '' I said, `` I'm sorry, '' I
said, `` I'm sorry, '' I said, `` I'm sorry, '' I said, `` I'm sorry, '' I said, `` I'm sorry, '' I said,
`` I'm sorry, '' I said, `` I'm sorry, '' I said, `` I'm sorry, '' I said, `` I'm sorry, '' I said, ``
I'm sorry, '' I said, `` I'm sorry, '' I said, `` I said, '' I said, `` I'm sorry, '' I said, `` I said,
'' I said, `` I said, '' I said, `` I said, '' I said, `` I said, '' I said, `` I said, '' I said, `` I s
aid, '' I said, `` I said, '' I said, `` I said, '' I said, `` I said, '' I said, `` I said, '' I said, `
` I said, '' I said, `` I said, '' I said, `` I said, '' I said, `` I said, '' I said, `` I said, '' I sa
id, `` I said, '' I said, `` I said, '' I said, `` I said, '' I said, `` I said, '' I said, `` I said, ''
I said, `` I said, '' I said, `` I said, '' I said, `` I said, '' I said, `` I said, '' I said, `` I sai
d, '' I said, `` I said, '' I said, `` I said, '' I said, `` I said, '' I said, `` I said, '' I said, ``
I said, '' I said, `` I said, '' I said, `` I said,
```

This issue wasn’t with the training but way I was sampling the model. So I solved this problem in next approach of sampling the words.

2. Training 10 epoches with 30000 examples from the dataset.

4 hours of training for 1 epoch and 8.2GBs of GPU.

```
In [ ]: #conduct training
        trainer.train()
```

```
/home/kamlesh_kumar/.venv/lib/python3.10/site-packages/transformers/optimization.py:411: FutureWarning
his implementation of AdamW is deprecated and will be removed in a future version. Use the PyTorch imp
entation torch.optim.AdamW instead, or set `no_deprecation_warning=True` to disable this warning
  warnings.warn(
```

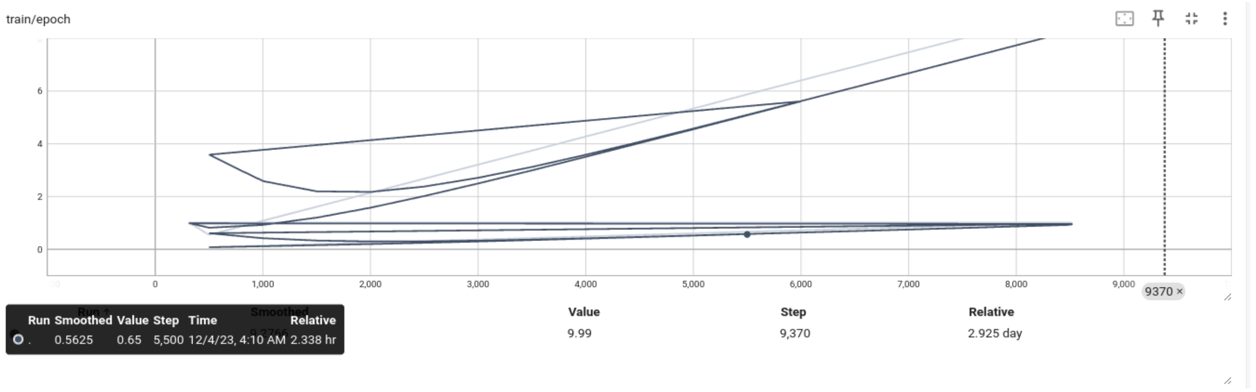
```
[ 103/9370 02:28 < 3:46:24, 0.68 it/s, Epoch 0.11/10]
```

Step	Training Loss

Saving Model

Evaluation of this training using tensorboard:

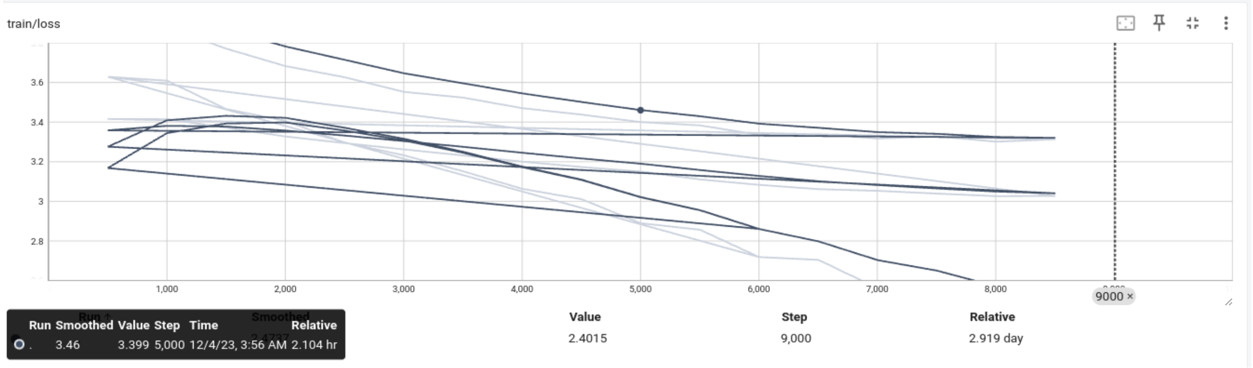
train/epoch:



Run	Smoothed	Value	Step	Time	Relative
.	0.5625	0.65	5,500	12/4/23, 4:10 AM	2.338 hr

Value	Step	Relative
9.99	9,370	2.925 day

train/loss:



Run	Smoothed	Value	Step	Time	Relative
.	3.46	3.399	5,000	12/4/23, 3:56 AM	2.104 hr

Value	Step	Relative
2.4015	9,000	2.919 day

!!Now we have two trained model!!

Outputs with different sampling method: i.e. introducing **do_sample = True** and **top_k = 0**

```
    sample_output = model.generate(
        **model_inputs,
        max_new_tokens=300,
        do_sample=True,
        top_k=0
    )
```

Example Evaluation for both training methods:

Prompt: You've finally managed to discover the secret to immortality. Suddenly, Death appears before you, hands you a business card, and says, `` When you realize living forever sucks, call this number, I've got a job offer for you. ''

Story 1: With model run for all 273K examples but 1 epoch only.

`` Hello? Hello? '' <newline> <newline> Jenny whispered she could sense his name coming from nowhere. A thin, sad looking dude in what appeared to be a complete punk charade was playing on him whilst he secured a massive candy bar with his initials. All around him were carnivals. In other words, Kim Kardashian and Veronica Reynolds at the ends of the job in Canada. Every one of them was right there. Half done. He had actually been invited back to a popular house party. A home party for a few years. One that made him happy enough to not be killed. He had done his best to regain his consciousness and own this carnival. This didn't make sense. <newline> <new

Story 2: Generation with model run with 30,000 examples for 10 epoches

`` you really aren't expecting this, are you? '' Death asked me quietly, not really caring if my answer was actually going to snowball.

`` I guess that means you're fucking *here*. '' I replied, a bit sheepish as I was trying to rationalize.Things like that usually happened when someone died, or some mundane at that moment suddenly different, in an instant or two. I was right, I was done. I had actually been offered the opportunity.

`` Yes. Well, I have actually got this deal for you, and as such you must understand I've lost quite a bit of your previous choice. Great deal. '' He started to shiver, and for a second I was filled with dread. I can't even pin down on what I thought of his choice. `` Am I either going to die like Romeo or not there are... '' I began hoarsely for a second, the painful memory was reliving the moment and acceptance almost overwhelming me in the process. Everyone seemed to me, a few acquaintances who had been laying about their profits in cancer rooms and drug dealing, others with debilitating cancer tormented souls who, at this very moment, feared they would find a way to do it. `` So I'll grant this to you for, like me or not, death? '' I replied sternly.

He looked at me disinterestedly and turned away, I considered all this for a

Observation:

For this example Prompt : You've finally managed to discover the secret to immortality. Suddenly, Death appears before you...," Story 2 is more relevant. It directly addresses the concept of immortality, the appearance of Death, and a job offer, which are key elements described in the prompt.

So the second model with 10 epochs seems to work the best. The reason can be that in this approach model has went through each of the 30K examples 10 times, which can help model better learn the context compared to the model with 273K examples where model has seen large dataset only once.

!!10 interesting Stories generated are attached in the notebook!!.

References:

https://huggingface.co/blog/how-to-generate

https://www.kaggle.com/code/ratthachat/writingprompts-gpt2-lm-fine-tune/notebook

https://www.kaggle.com/datasets/ratthachat/writing-prompts/data

THE END