Nano GPT on W. Schakespeare/ A. Pushkin datasets

Generatively Pretrained Transformer

- Core - Attention mechanism

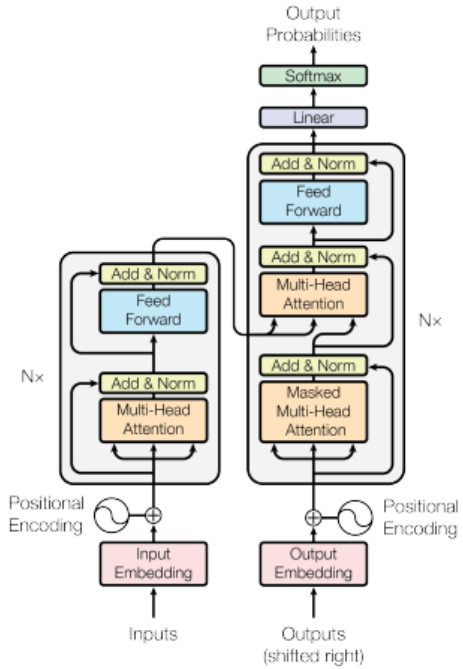
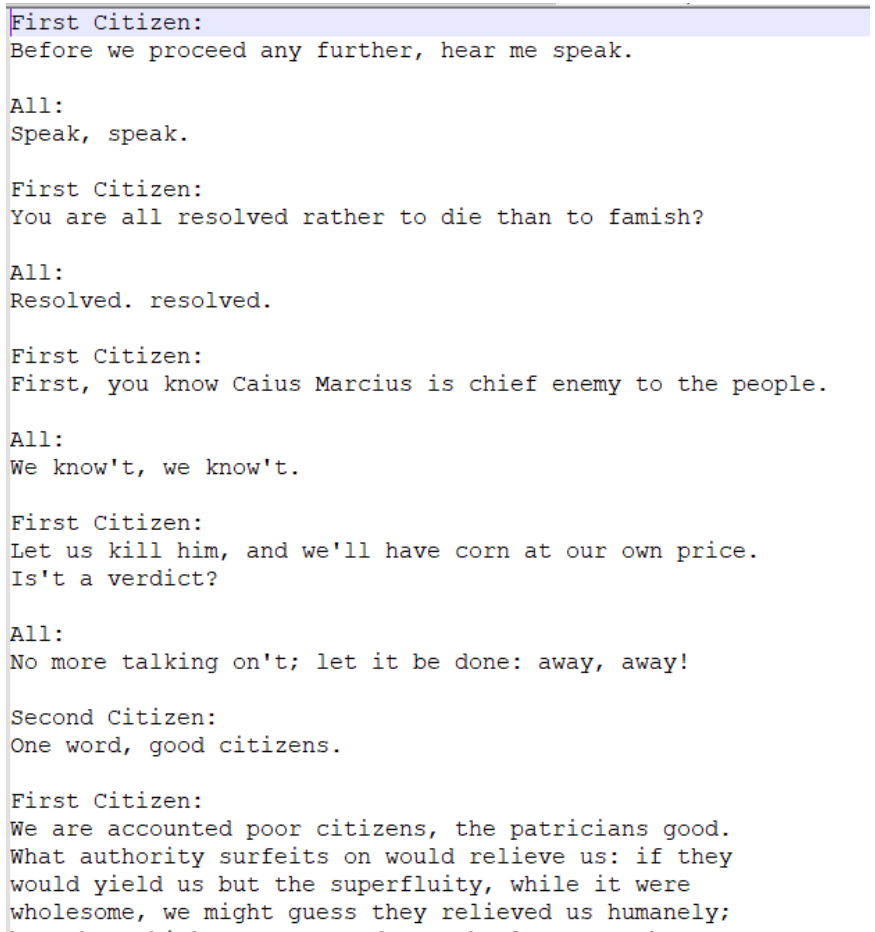


Figure 1: The Transformer - model architecture.

Train transformer-based (character level)language model

- tiny_shakespheare



```
First Citizen:
Before we proceed any further, hear me speak.

All:
Speak, speak.

First Citizen:
You are all resolved rather to die than to famish?

All:
Resolved. resolved.

First Citizen:
First, you know Caius Marcius is chief enemy to the people.

All:
We know't, we know't.

First Citizen:
Let us kill him, and we'll have corn at our own price.
Is't a verdict?

All:
No more talking on't; let it be done: away, away!

Second Citizen:
One word, good citizens.

First Citizen:
We are accounted poor citizens, the patricians good.
What authority surfeits on would relieve us: if they
would yield us but the superfluity, while it were
wholesome, we might guess they relieved us humanely;
```

- tiny_pushkin

```
Воспоминания в Царском Селе

Навис покров угрюмой нощи
На своде дремлющих небес;
В безмолвной тишине почили дол и рощи,
В седом тумане дальний лес;
Чуть слышится ручей, бегущий в сень дубравы,
Чуть дышит ветерок, уснувший на листах,
И тихая луна, как лебедь величавый,
Плывет в сребристых облаках.

С холмов кремнистых водопады
Стекают бисерной рекой,
Там в тихом озере плескаются наяды
Его ленивою волной;
А там в безмолвии огромные чертоги,
На своды опершись, несутся к облакам.
Не здесь ли мирны дни вели земные боги?
Не се ль Минервы росской [1] храм?

Не се ль Элизиум полнощный,
Прекрасный Царскосельский сад,
Где, льва [2] сразив, почил орел России мощный
На лоне мира и отрад?
Промчались навсегда те времена златые.
Когда под скипетром великия жены
Венчалась славою счастливая Россия,
Цветя под кровом тишины!
```

install

Dependencies:

- pytorch <3
- numpy <3
- `pip install transformers` for huggingface transformers <3 (to load GPT-2 checkpoints)
- `pip install datasets` for huggingface datasets <3 (if you want to download + preprocess OpenWebText)
- `pip install tiktoken` for OpenAI's fast BPE code <3
- `pip install wandb` for optional logging <3
- `pip install tqdm`

Chat GPT

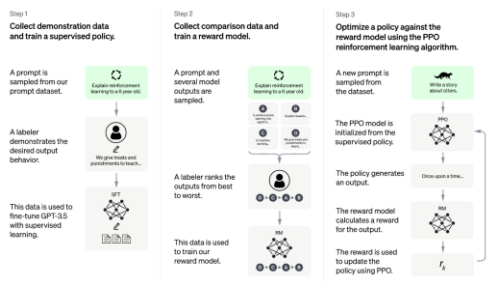
- Pretraining stage

OpenWeb DataSet

Model Name	n_{params}	n_{layers}	d_{model}	n_{heads}	d_{head}	Batch Size	Learning Rate
GPT-3 Small	125M	12	768	12	64	0.5M	6.0×10^{-4}
GPT-3 Medium	350M	24	1024	16	64	0.5M	3.0×10^{-4}
GPT-3 Large	760M	24	1536	16	96	0.5M	2.5×10^{-4}
GPT-3 XL	1.3B	24	2048	24	128	1M	2.0×10^{-4}
GPT-3 2.7B	2.7B	32	2560	32	80	1M	1.6×10^{-4}
GPT-3 6.7B	6.7B	32	4096	32	128	2M	1.2×10^{-4}
GPT-3 13B	13.0B	40	5140	40	128	2M	1.0×10^{-4}
GPT-3 175B or "GPT-3"	175.0B	96	12288	96	128	3.2M	0.6×10^{-4}

Table 2.1: Sizes, architectures, and learning hyper-parameters (batch size in tokens and learning rate) of the models which we trained. All models were trained for a total of 300 billion tokens.

- Optimisation stage



Nano GPT

- Pretraining stage

Architecture near identical

~ 10M parameters,

~ 1M characters (~300 000 subwords)

- Optimisation stage

no