Week 4 Assignment

Generative AI

Saarland University – Winter Semester 2024/25

Martínez
7057573
cama00005@stud.uni-saarland.de

1 Exercise Assignment: E1

Given:

- $N = 67$ billion parameters
- $D = 4.1$ trillion training tokens
- Constants for loss estimation: $E = 1.69; A = 406.4; B = 410.7; \alpha = 0.34; \beta = 0.28$

The approximate compute cost C (measured in FLOPs) required to train the model for the given N and D can be approximated as [1]:

$$C = \text{FLOPs}(N, D) \approx 6ND$$

Thus,

$$C \approx 6 \times 67 \times 10^9 \times 4.1 \times 10^{12} = 1.65 \times 10^{24} \text{ FLOPs}$$

And the expected loss L is given by [2]:

$$L(N, D) \triangleq E + \frac{A}{N^\alpha} + \frac{B}{D^\beta}$$

Thus,

$$L = 1.69 + \frac{406.4}{(67 \times 10^9)^{0.34}} + \frac{410.7}{(4.1 \times 10^{12})^{0.28}} \approx 1.90$$

2 Exercise Assignment: E2

To improve the model's performance decreasing the previously computed loss $L = 1.90$ by 1%, we would need a new loss L' such that[1]:

$$L' = 0.99 \times L = 0.99 \times 1.895506113 \approx 1.876551052$$

Now, we can calculate the new amount of training tokens D' required to achieve this new loss L' assuming the number of parameters N remains the same:

$$L' = E + \frac{A}{N^\alpha} + \frac{B}{D'^\beta} \rightarrow D' = \left(\frac{B}{L' - E - \frac{A}{N^\alpha}}\right)^{\frac{1}{\beta}} = \left(\frac{410.7}{1.876551052 - 1.69 - \frac{406.4}{(67 \times 10^9)^{0.34}}}\right)^{\frac{1}{0.28}}$$
$$\approx 7.54 \text{ trillion tokens}$$

The new total number of FLOPs C' required for this training would then be[2]:

$$C' \approx 6ND' = 6 \times 67 \times 10^9 \times 7.54407114 \times 10^{12} = 3.03 \times 10^{24} \text{ FLOPs}$$

[1] For (E.1), the loss approximated to 2 decimal places is 1.90, but in order to get precise results for this exercise and the rest, the complete loss value was used, i.e., $L = 1.895506113$.

[2] Using the exact value for $D' = 7.54407114 \times 10^{12}$

Finally, with the following geometric series, we can estimate the number of additional training epochs x after the first epoch required to achieve the new loss L':

$$D' = 2 \cdot \left(1 - \frac{1}{2^{x+1}}\right) \cdot D \rightarrow x = \log_2 \left(\frac{1}{1 - \frac{D'}{2D}}\right) - 1 \tag{1}$$

Replacing the corresponding non-approximated values in (1), we get the additional number of epochs x:

$$x = \log_2 \left(\frac{1}{1 - \frac{7.54407114}{2 \times 4.1}}\right) - 1 \approx 2.64 \text{ epochs}$$

3 Exercise Assignment: E3

For compute-optimal training, we know that $C' \approx 6N'D$ regardless of epochs. This is the total amount of compute we have available (which might be used for more than 1 epoch, if necessary). Thus, if we fix C, we can calculate the number of epochs x required to train the model:

$$\begin{aligned} C' = 6N'Dx \rightarrow D = 6 \times 1.41N \times Dx \rightarrow x &= \frac{C'}{6 \times 1.41N \times D} \\ &= \frac{3.03271660 \times 10^{24}}{6 \times 1.41 \times 67 \times 10^9 \times 4.1 \times 10^{12}} \\ &\approx 1.30 \text{ epochs} \end{aligned}$$

Thus, the optimal number of epochs to train the model is approximately 1.30 epochs.

For the given scenario, we can see that the optimal scaling of the model size and number of epochs w.r.t. the compute cost is such that the model size should be increased while the number of epochs should be decreased. This is because the compute cost is fixed, and the model size and number of epochs are inversely proportional to each other. Thus, to achieve the best performance with the given compute cost after increasing the model size ($N' = 1.41N$), it makes sense that we decrease the number of epochs to 1.30 epochs, compared to the previous 2.64 epochs.

4 Exercise Assignment: E4

GPT-3 has 175 billion parameters, each with 16-bit (2 bytes) precision. This means that one single GPT-3 model would occupy in memory[3]:

$$175 \times 10^9 \text{ parameters} \times 2 \text{ bytes/parameters} = 3.50 \times 10^{11} \text{ bytes} \quad (\approx 325.96 \text{ GB})$$

On the other hand, 100 distinct tasks require finetuning 100 different models. This means that the total amount of memory required to store all these models is:

$$100 \times 3.50 \times 10^{11} \text{ bytes} = 3.50 \times 10^{13} \text{ bytes} \quad (\approx 31.83 \text{ TB})$$

5 Exercise Assignment: E5

If instead of full-finetuning, we use adapters to fine-tune only the query and value projection matrices in the self-attention module, we would need the following amount of memory per model:

$$\begin{aligned} 2 \times d_{\text{model}} \times d_{\text{model}} \times 96 \times 2 \text{ bytes} &= 2 \times 12{,}288 \times 12{,}288 \times 96 \times 2 \text{ bytes/parameters} \\ &\approx 5.80 \times 10^{10} \text{ bytes} \quad (54 \text{ GB}) \end{aligned}$$

Since we have 2 matrices (query and value matrices) of size $d_{\text{model}} \times d_{\text{model}}$ per layer, where $d_{\text{model}} = 12{,}288$ for GPT-3, and 96 layers.

Thus, the total amount of memory required to store all 100 of these models would be approximately:

$$100 \times 5.80 \times 10^{10} \text{ bytes} = 5.80 \times 10^{12} \text{ bytes} \quad (\approx 5.27 \text{ TB})$$

[3] 1 GB $= 1024^3$ bytes

6 Exercise Assignment: E6

If we apply low-rank adaptation (LoRA) with rank $r = 4$ to the query $Q \in \mathbb{R}^{d_{\text{model}} \times d_{\text{model}}}$ and value $V \in \mathbb{R}^{d_{\text{model}} \times d_{\text{model}}}$ projection matrices in each of the 96 layers, we need to define **for each** Q and V matrices a pair of matrices, $B \in \mathbb{R}^{d_{\text{model}} \times 4}$ and $A \in \mathbb{R}^{4 \times d_{\text{model}}}$, per layer. Thus, in this setting, we would need per model:

$$\underbrace{2}_{Q \text{ and } V} \times \underbrace{2}_{A \text{ and } B} \times \underbrace{d_{\text{model}} \times 4}_{\text{Size of both } A \text{ and } B} \times \underbrace{96}_{\text{Attention layers}} \times 2 \text{ bytes} = 2 \times 2 \times 12{,}288 \times 4 \times 96 \times 2 \text{ bytes}$$

$$\approx 3.77 \times 10^7 \text{ bytes} \quad (36 \text{ MB})$$

Hence, the total amount of memory required to store all 100 of these models would be approximately:

$$100 \times 3.77487360 \times 10^7 \text{ bytes} \approx 3.77 \times 10^9 \text{ bytes} \quad (\approx 3.51 \text{ GB})$$

7 Exercise Assignment: E7

Using a similar derivation as in (E.6) and restricting ourselves with the same amount of memory per model (previously calculated as $\approx 3.77 \times 10^7$ bytes), the value of rank r for LoRA if we adapt only the query $Q \in \mathbb{R}^{d_{\text{model}} \times d_{\text{model}}}$ projection matrix this time would have to be:

$$M = 2 \times d_{\text{model}} \times r \times 96 \times 2 \text{ bytes} \rightarrow r = \frac{M}{2 \times d_{\text{model}} \times 96 \times 2 \text{ bytes}} = \frac{3.77487360 \times 10^7}{2 \times 12{,}288 \times 96 \times 2} = 8$$

8 Individual Assignment: I3

Table 1 shows the qualitative comparison based on my own perspective, of the base pre-trained and supervised fine-tuned Phi-1.5, which helps us see how the quality varied between both of the models for each evaluation sample. From these results, we can conclude that the supervised fine-tuned model clearly outperforms the base pre-trained model.

Table 1: Comparison of the pretrained and supervised fine-tuned Phi-1.5 in terms of how many times out of the 3 generations are qualitatively good per evaluation sample (i.e., a `PREFIX`).

Model	`PREFIX`				
	1	2	3	4	5
Base pre-trained Phi-1.5	1/3	0/3	3/3	0/3	0/3
Supervised fine-tuned Phi-1.5	3/3	3/3	2/3	3/3	3/3

Moreover, as discussed in Week 3's assignment, Microsoft's Phi-1.5 is a custom model with 1.3B parameters, whose

> "*training involved a variety of data sources, including subsets of Python codes from The Stack v1.2, Q&A content from StackOverflow, competition code from code contests, and synthetic Python textbooks and exercises generated by gpt-3.5-turbo-0301*" [3]

Also, its model card further clarifies that

> "(...) *Phi-1.5 is best suited for prompts using the Q&A format, the chat format, and the code format. Note that Phi-1.5, being a base model, often produces irrelevant text following the main answer* (...)" [4]

Thus, it is understandable that we get completions such as the one in Figure 1, where the model deviates from the original prefix, since it is simply more suited for Q&A and code generation tasks given its training data. These kinds of hallucinations were very common in the completions made by the base pre-trained Phi-1.5 model.

On the other hand, the supervised fine-tuned model with Shakespeare's works, as shown in Figure 2, was able to generate more coherent completions that were more in line with the original `PREFIX`. This is the reason why we need to fine-tune models to specific tasks or domains to achieve better results.

9 Individual Assignment: I7

Table 2 shows the qualitative comparison based on my own perspective, of the base pre-trained, fully supervised fine-tuned, and LoRA supervised fine-tuned GPT-2 models, which helps us see how the quality varied between all of the models for each evaluation sample.

Table 2: Comparison of the base pre-trained, fully supervised fine-tuned and LoRA supervised fine-tuned GPT-2 models in terms of how many times out of the 3 summarizations are qualitatively good per evaluation sample (i.e., a `PREFIX`).

Model	`PREFIX`				
	1	2	3	4	5
Base pre-trained GPT-2	$1/3$	$0/3$	$0/3$	$0/3$	$0/3$
Fully supervised fine-tuned GPT-2	$3/3$	$3/3$	$3/3$	$3/3$	$3/3$
LoRA supervised fine-tuned GPT-2	$3/3$	$0/3$	$3/3$	$2/3$	$2/3$

From Table 2, we can rank the performance of the three models like this:

- $\Rightarrow$ **Fully supervised fine-tuned GPT-2**
  This provided the best results due to its fully supervised fine-tuning approach. More parameters to fit allowed it to better adapt to the task of *summarization*. This is illustrated by the summarizations in Figure 3, where all of them were good and correctly included the final question or problem the user posed in the original Reddit post. Nevertheless, it is not perfect and can still make typos, as seen in the word `"mange"` instead of `"manage"` in the first summarization.
- $\Rightarrow$ **LoRA supervised fine-tuned GPT-2**
  This model performed much better than the base pre-trained approach, but slightly lagged behind the fully supervised fine-tuned GPT-2. As seen on (E.6), normally we tune less parameters with LoRA, at the cost of some performance, but providing a huge reduction in memory requirements. This is illustrated by the summarizations in Figure 4, where $2/3$ were good, because one of them started repeating the same sentence over and over.
- $\Rightarrow$ **Base pre-trained GPT-2**
  This model showed the least performance as it was not fine-tuned on the specific task, relying only on its general pre-training. This model sometimes generated the same sentence over and over or produced almost the same output as the input. These problems are illustrated by the summarizations in Figure 5, where none of them were good.

10 Individual Assignment: I8

The qualitative ranking made previously aligns well with the `ROUGE-1` mean score outputted by each model:

- $\Rightarrow$ **Fully supervised fine-tuned GPT-2** (`ROUGE-1` mean score ≈ 0.29)
- $\Rightarrow$ **LoRA supervised fine-tuned GPT-2** (`ROUGE-1` mean score ≈ 0.20)
- $\Rightarrow$ **Base pre-trained GPT-2** (`ROUGE-1` mean score ≈ 0.098)

This metric is used to compare an automatically produced summary against a human-produced summary, where the higher score the better and if the score is 1 then it matches exactly the human reference summary. Therefore, as expected based on our qualitative review, the fully supervised fine-tuned GPT-2 model had the best score, followed by the LoRA supervised fine-tuned GPT-2 model, and finally the base pre-trained GPT-2 model.

Acknowledgements

This week's slides and listed references, as well as classmate Dhimitrios Duka & Kai Wittenmayer, with whom I discussed the exercises.

References

[1] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *CoRR*, abs/2001.08361, 2020.

[2] Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Jack W. Rae, Oriol Vinyals, and Laurent Sifre. Training compute-optimal large language models, 2022.

[3] Hugging Face. Microsoft phi-1 model card. `https://huggingface.co/microsoft/phi-1`. Accessed: 2024-11-06.

[4] Hugging Face. Microsoft phi-5 model card. `https://huggingface.co/microsoft/phi-1_5`. Accessed: 2024-11-06.

Appendix: Completions Examples

```
=== PREFIX 5 ===
Great lords, wise men ne’er sit and wail their loss,

====================================
=== COMPLETION 5-A ===

I seek a way to help my people through this strife.
But they are too afraid of the unknown, so I am alone."
This poem was written by John Milton in 1667 during his reign as Lord
Protector of England at that time.  It is an example of how poetry
can be used to express complex emotions and ideas about historical
events or themes like war and power struggles between different
rulers and kingdoms.

====================================
=== COMPLETION 5-B ===

The kings of the earth are weary from strife.
"""

def __init__(self):
super().__setattr__("_name", "Bard") # TODO: _title is not defined
for Bard class yet...  (TODO)
@property def name(self):  return self._name
@classmethod def

====================================
=== COMPLETION 5-C ===
For in the end they are all but fools.
A bed of roses may die down quickly if you don’t water it." -
Anonymous""" return self._fetch(query)[’data’][0]["title"] import
numpy as np from scipy import linalg # from sklearn.base import
BaseEstimator, Regressor
====================================
```

Figure 1: PREFIX 5 completion by the base pre-trained Microsoft’s Phi-1.5.

```
=== PREFIX 5 ===
Great lords, wise men ne’er sit and wail their loss, <complete>

====================================
=== COMPLETION 5-A ===
or we’ll be in the same place again.  But now tell me which of these
is a better man:  A fool who never makes you laugh or cry?  Or one
whose eyes are so deep that they can see your soul; And if he fails
to do it before his grave, then I would say him not as good at all
but as bad As an old horse-cart with no reins In this world

====================================
=== COMPLETION 5-B ===
though it’s as fair a thing As the world itself.  I’ll be there to
get you up; And when they see my face They will not speak a word of
sorrow With me in your company, that is so far behind The end of this
kingdom hath been known To his death by many names:  King John!  KING
JOHN. What are we?  Here comes Lord Walpole with news from London- ’

====================================
=== COMPLETION 5-C ===
for we have our own.  The world is a great sea; And the wind of it
cannot be stilled by man’s hand or eye.  We stand here to make thee
proud.  O Lord Earl of Gloucester!  Why do you not come?  I’ll send
my lady for thy company In this land that hath no end till sundown:
Let her see us before she leaves with me To bring some news
====================================
```

Figure 2: `PREFIX 5` completion by the supervised fine-tuned Phi-1.5.

```
====================================
=== SUMMARY 3-A ===
I work at a restaurant and I have to do dishes,prep food, and wipe
down every surface every night.  How can I mange this job?

====================================
=== SUMMARY 3-B ===
I have to do dishes, prep food, wipe down every surface and do all
the busboy jobs as well every night.  How can I mange this Job?

====================================
=== SUMMARY 3-C ===
I have to clean dishes, prepare food, and do all the jobs at a small
restaurant.  How can I mange this job
====================================
```

Figure 3: `POST 3` summarization by the fully supervised fine-tuned GPT-2.

```
====================================
=== SUMMARY 5-A ===
I need a solution to all the passwords I need to remember for every
website I use.  I'm not a programmer, I'm not a gamer, I'm not a
gamer.  I'm a developer.  I need an account for all of them.  I need
an account for all of them.  I need an account for all of them.  I
need an account for all of them.  I need an account for all of them.
I need an account for all of them.  I need an account for all of them.
I need an account for all of them.  I need an account for all of them.
I need an account for all of them.  I need an account for all of them.
I need an account for all of them.  I need an account for all of them.
I need an account for all of them.  I need an account for all of them.
I need an account

====================================
=== SUMMARY 5-B ===
I need to remember all the passwords for all the websites I use, and
I'm not sure how to do it.  What should I do

====================================
=== SUMMARY 5-C ===
I have a ton of usernames and passwords, and I need an account for
all of them.  I'm a developer, and I need an account for all of them.
====================================
```

Figure 4: `POST 5` summarization by the LoRA supervised fine-tuned.

```
====================================
=== SUMMARY 1-A ===
I am looking to cash out my 401k to make an emergency fund.  I am
26 years old male.  I have no savings to my name.  I have applied
for 7 positions that I feel confident match my skill set.  I have
an application out to about 7 other positions that I feel confident
match my skill set.  I have an application out to about 7 other
positions that I feel confident match my skill set.  I have an
application out to about 7 other positions that I feel confident
match my skill set.  I have an application out to about 7 other
positions that I feel confident match my skill set.  I have an
application out to about 7 other positions that I feel confident
match my skill set.  I have an application out to about 7 other
positions that I feel confident match my skill set.  I have an
application out to about 7 other positions that I feel confident
match my skill set.  I have

====================================
=== SUMMARY 1-B ===
I am looking to cash out my 401k to make an emergency fund.  Thanks
for reading!

====================================
=== SUMMARY 1-C ===
I am looking for advice on how to cash out my 401k to make an
emergency fund.  Thanks, -Ryan SUBREDDIT: r/personalfinance TITLE:
Decisions regarding a 401k Cash out POST: Hi r/personalfinance, I
have been looking for guidance on this issue, but do not have a
financial planner currently.  I am a 26 year old male looking to
leave my current job.  To bring you up to pace, I am an insurance
adjuster for a major insurance company in America.  I took a
promotion about 9-10 months ago that I am now regretting.  Without
getting into any details on why I am looking outside the company, I
have a financial dilemma that may not allow me to leave at this time.
I currently make about $46,700.  I currently have no savings to my
name due to some
====================================
```

Figure 5: POST 1 summarization by the base pre-trained GPT-2.