Cornell Bowers C·IS
College of Computing and Information Science

Deep Learning

Week [03]: [LLMs]

Cornell Bowers C·IS

Thanks to

Varsha Kishore
Justin Lovelace
Adam Alnasser
Elizabeth Kelmenson

Project

- **Aim:** to get hands on experience with implementing modern deep learning methods
- Find a recent deep learning research paper that was published in the last 1-5 years
- Reproduce a specific result from the paper
- To be completed in groups of 4-5

Cornell Bowers C·IS

Project Grading

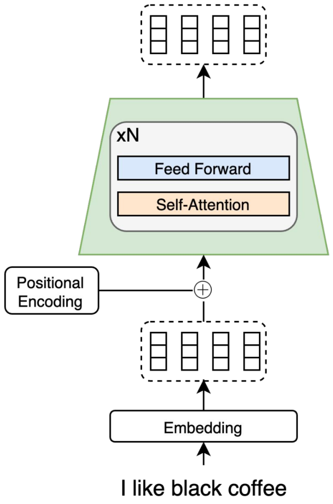
- Project selection
  - We want to make sure the papers you've chosen are reasonable
    - Possible on Collab (GPU limits)
    - Data requirements realistic (data easily available)
  - Sign up on spreadsheet
- Final presentation
  - **Poster presentations** at the end
  - Present the paper you've chosen to the class
  - Discuss whether you were able to replicate results from the paper and describe any obstacles
- Final deliverable
  - Github repo with re-implementation
  - Readme with method description, instructions to run the code and results
  - More detail will be provided later

Part I: Transformer Models

Three famous Transformer models

BERT

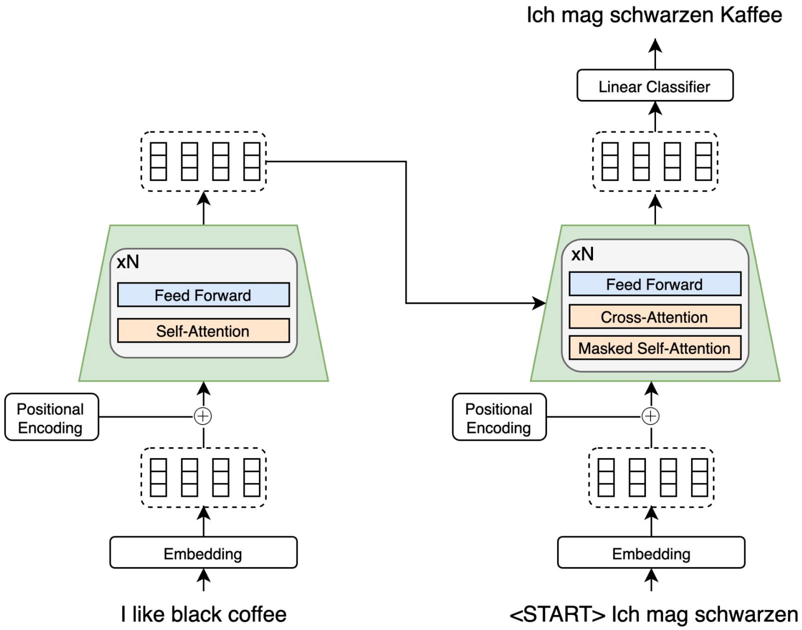
encoder only



[Devlin et al. 2019]

T5/BART

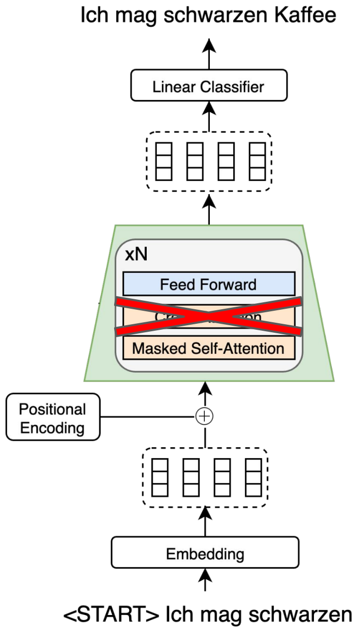
encoder-decoder model



[Raffel et al. 2020] [Lewis et al. 2020]

GPT

decoder only

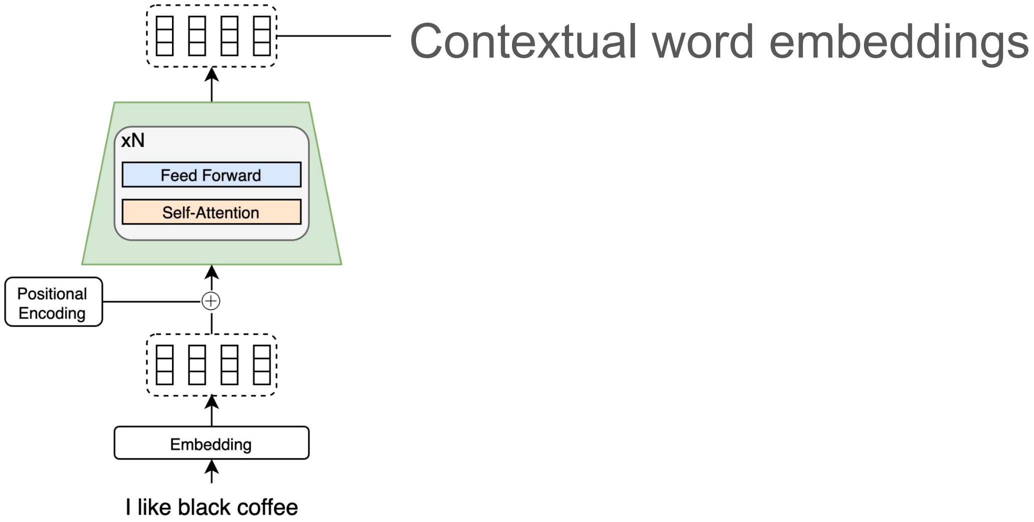


[Radford et al. 2018]

Cornell Bowers C·IS

BERT (Bidirectional Encoder Representations from Transformers)

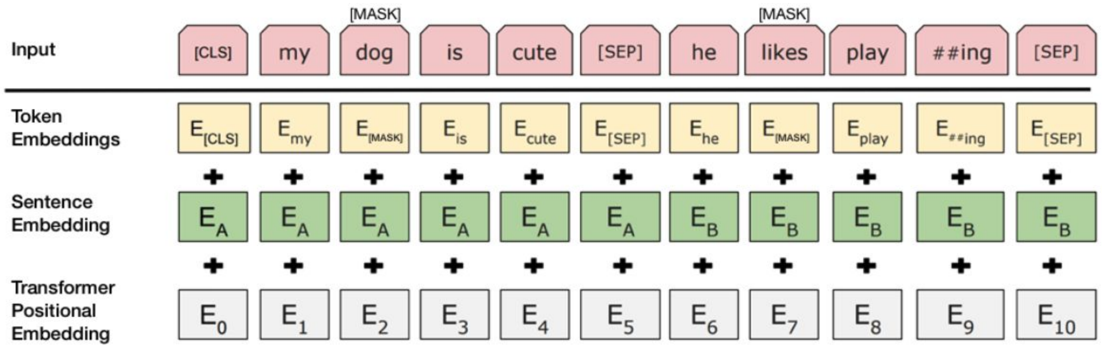
- Output of **Transformer Encoder** as contextual word embeddings
- Bidirectional Context
- Pre-trained on the language, and then fine-tuned



BERT - Input Representation

Input:

- Use 30,000 WordPiece vocabulary on input.
- Each token is sum of three embeddings



Training

- Masked Language Modelling
  - Mask out k% of the input words, and then predict the masked words
  - the man went to the store to [MASK] a [MASK] of milk
  - What can you use as a loss function?

- Next sentence prediction
  - To learn relationships between sentences, predict whether Sentence B is actual sentence that
  - proceeds Sentence A, or a random sentence

Sentence A = `The man went to the store.`
Sentence B = `He bought a gallon of milk.`
Label = `IsNextSentence`

Sentence A = `The man went to the store.`
Sentence B = `Penguins are flightless.`
Label = `NotNextSentence`

Cornell Bowers C·IS

Self-supervised Learning

- Labels are generated automatically, no human labeling process
- Benefits
  - Scales well
  - Cost-Efficient
  - Flexible
- Challenges
  - Larger datasets are required
  - More compute is necessary

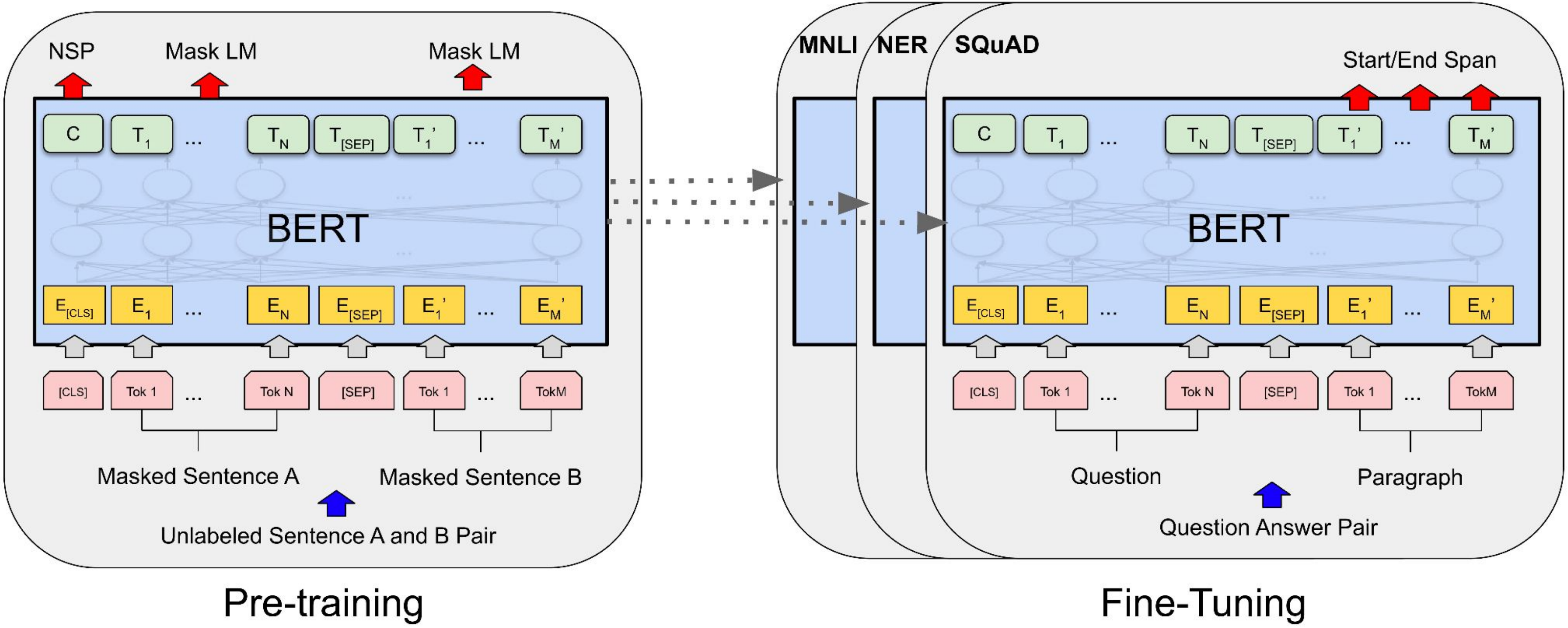
Cornell Bowers C·IS

Model Details

- Data: Wikipedia (2.5B words) + BookCorpus (800M words)
- Batch Size: 131,072 words (1024 sequences * 128 length or 256 sequences * 512 length)
- Training Time: 1M steps (~40 epochs)
- Optimizer: AdamW, 1e-4 learning rate, linear decay
- BERT-Base: 12-layer, 768-hidden, 12-head, 110M params
- BERT-Large: 24-layer, 1024-hidden, 16-head, 340M params
- Trained on 4x4 or 8x8 TPU slice for 4 days

Cornell Bowers C·IS

Pre-training to Fine-tuning Pipeline



System	MNLI-(m/mm)	QQP	QNLI	SST-2	CoLA	STS-B	MRPC	RTE	**Average**
	392k	363k	108k	67k	8.5k	5.7k	3.5k	2.5k	-
Pre-OpenAI SOTA	80.6/80.1	66.1	82.3	93.2	35.0	81.0	86.0	61.7	74.0
BiLSTM+ELMo+Attn	76.4/76.1	64.8	79.8	90.4	36.0	73.3	84.9	56.8	71.0
OpenAI GPT	82.1/81.4	70.3	87.4	91.3	45.4	80.0	82.3	56.0	75.1
$\text{BERT}_{\text{BASE}}$	84.6/83.4	71.2	90.5	93.5	52.1	85.8	88.9	66.4	79.6
$\text{BERT}_{\text{LARGE}}$	**86.7/85.9**	**72.1**	**92.7**	**94.9**	**60.5**	**86.5**	**89.3**	**70.1**	**82.1**

https://arxiv.org/pdf/1810.04805.pdf

Cornell Bowers C·IS

Pre-Train then Fine-tune Paradigm

- Pre-train a large model on a lot of data with some self-supervised objective.
  - E.g. Masked language modeling

- Fine-tune on (smaller) downstream datasets

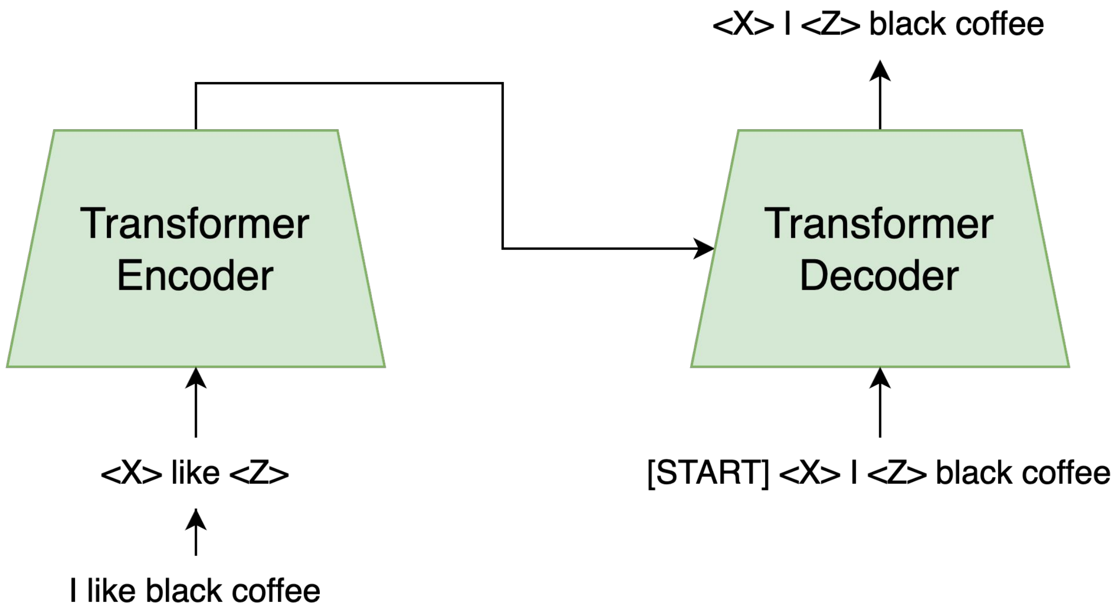
- Benefits:
  - Transfer learning: Leveraging knowledge from self-supervised objective to improve performance
  - Avoids overfitting: Large models would overfit if trained from scratch

Question:

- What are some limitations of BERT?
  - Think about the different kinds of language tasks you might want to perform

T5: Text-to-Text Transfer Transformer

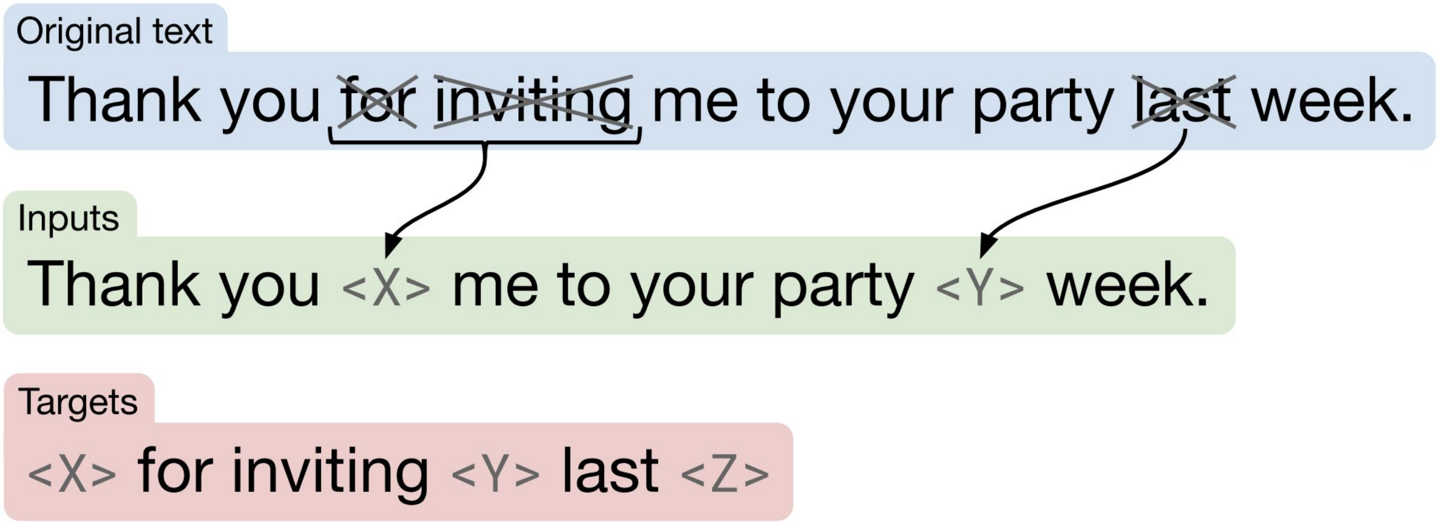
- Pre-trained Encoder Decoder Language Model
  - Can generate text!



Cornell Bowers C·IS

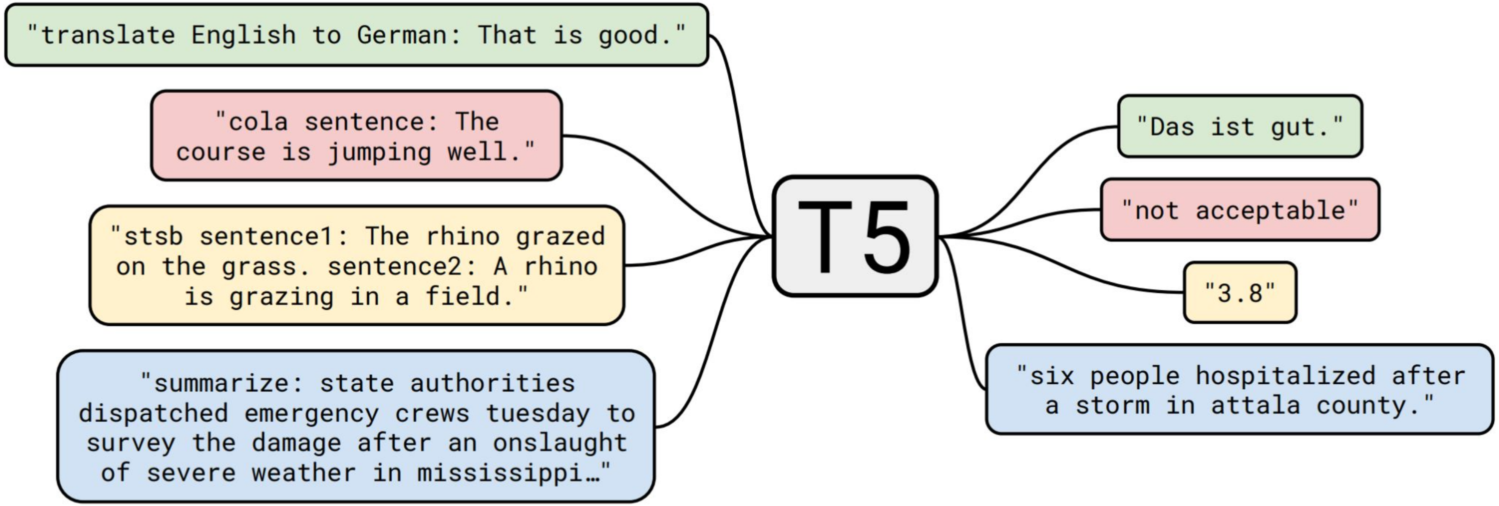
T5 Span Corruption Objective

- Very similar to masked language modeling
- Key differences:
  - Mask spans with single “sentinel” tokens
  - Generate the masked text with a decoder



T5 Fine-Tuning

- Cast every task as a language generation problem
  - Even classification!



Cornell Bowers C·IS

Benchmark Results

- Outperformed best encoder-only models

Model	GLUE Average	CoLA Matthew's	SST-2 Accuracy	MRPC F1	MRPC Accuracy	STS-B Pearson	STS-B Spearman
Previous best	89.4[a]	69.2[b]	97.1[a]	**93.6**[b]	**91.5**[b]	92.7[b]	92.3[b]
T5-Small	77.4	41.0	91.8	89.7	86.6	85.6	85.0
T5-Base	82.7	51.1	95.2	90.7	87.5	89.4	88.6
T5-Large	86.4	61.2	96.3	92.4	89.9	89.9	89.2
T5-3B	88.5	67.1	97.4	92.5	90.0	90.6	89.8
T5-11B	**90.3**	**71.6**	**97.5**	92.8	90.4	**93.1**	**92.8**

Model	QQP F1	QQP Accuracy	MNLI-m Accuracy	MNLI-mm Accuracy	QNLI Accuracy	RTE Accuracy	WNLI Accuracy
Previous best	74.8[c]	**90.7**[b]	91.3[a]	91.0[a]	**99.2**[a]	89.2[a]	91.8[a]
T5-Small	70.0	88.0	82.4	82.3	90.3	69.9	69.2
T5-Base	72.6	89.4	87.1	86.2	93.7	80.1	78.8
T5-Large	73.9	89.9	89.9	89.6	94.8	87.2	85.6
T5-3B	74.4	89.7	91.4	91.2	96.3	91.1	89.7
T5-11B	**75.1**	90.6	**92.2**	**91.9**	96.9	**92.8**	**94.5**

T5 Ablations- Dataset Size

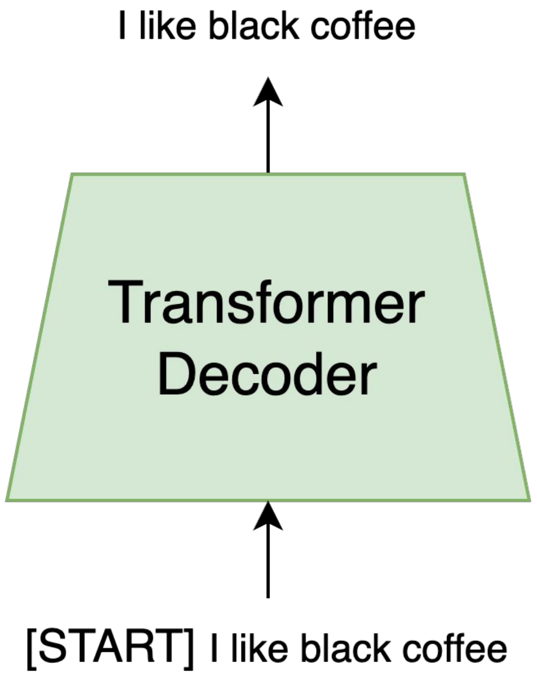
- More data helps!

Number of tokens	Repeats	GLUE	CNNDM	SQuAD	SGLUE	EnDe	EnFr	EnRo
★ Full data set	0	**83.28**	**19.24**	**80.88**	**71.36**	**26.98**	**39.82**	**27.65**
2^{29}	64	**82.87**	**19.19**	**80.97**	**72.03**	**26.83**	**39.74**	**27.63**
2^{27}	256	82.62	**19.20**	79.78	69.97	**27.02**	**39.71**	27.33
2^{25}	1,024	79.55	18.57	76.27	64.76	26.38	39.56	26.80
2^{23}	4,096	76.34	18.33	70.92	59.29	26.37	38.84	25.81

Cornell Bowers C·IS

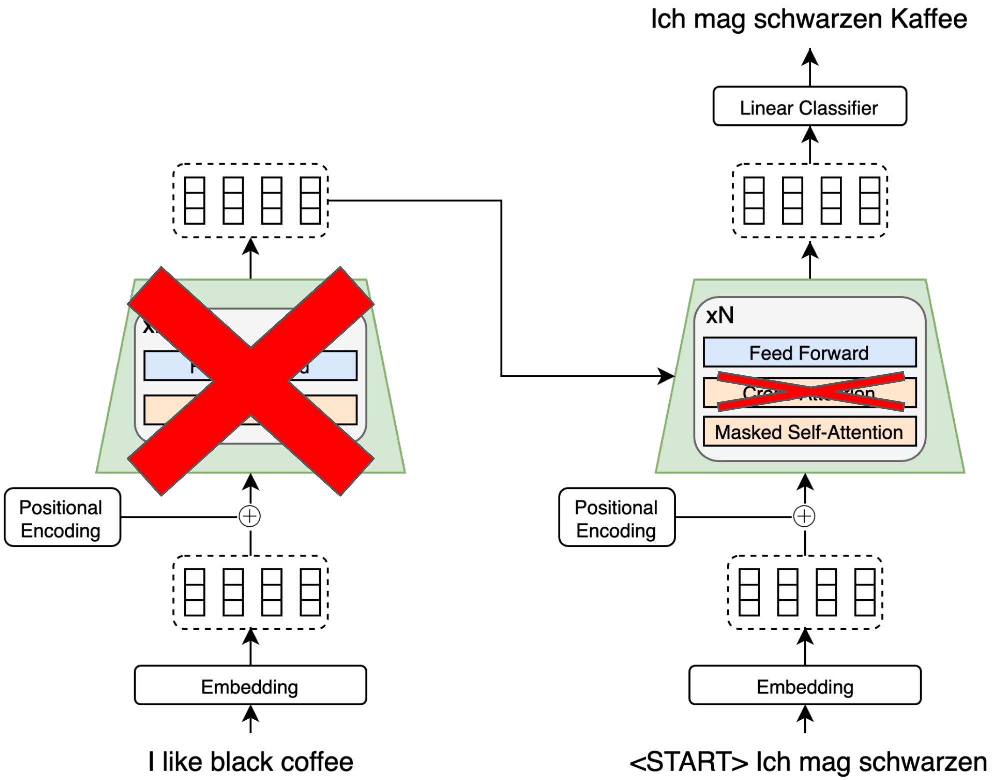
GPT

- Only use the transformer decoder
  - Train on next word prediction
  - “The capital of France is _”
- Easy to apply in streaming settings
  - E.g. chatbots

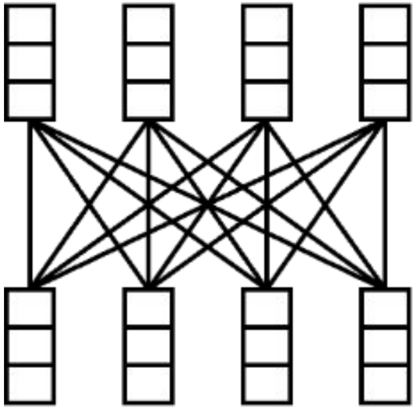


Cornell Bowers C·IS

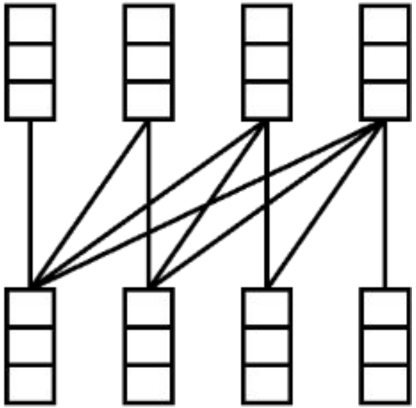
Decoder-Only Architecture



Review: Masked Self-Attention



Self-Attention



Masked Self-Attention

GPT-2 Zero-Shot Capabilities

- Question-answering without any fine-tuning
    - Formatting is important!

Context (passage and previous question/answer pairs)

Tom goes everywhere with Catherine Green, a 54-year-old secretary. He moves around her office at work and goes shopping with her. ”Most people don’t seem to mind Tom,” says Catherine, who thinks he is wonderful. ”He’s my fourth child,” she says. She may think of him and treat him that way as her son. He moves around buying his food, paying his health bills and his taxes, but in fact Tom is a dog.

Catherine and Tom live in Sweden, a country where everyone is expected to lead an orderly life according to rules laid down by the government, which also provides a high level of care for its people. This level of care costs money.

People in Sweden pay taxes on everything, so aren’t surprised to find that owning a dog means more taxes. Some people are paying as much as 500 Swedish kronor in taxes a year for the right to keep their dog, which is spent by the government on dog hospitals and sometimes medical treatment for a dog that falls ill. However, most such treatment is expensive, so owners often decide to offer health and even life _ for their dog.

In Sweden dog owners must pay for any damage their dog does. A Swedish Kennel Club official explains what this means: if your dog runs out on the road and gets hit by a passing car, you, as the owner, have to pay for any damage done to the car, even if your dog has been killed in the accident.

Q: How old is Catherine?
A: 54

Q: where does she live?
A:

Model answer: Stockholm
Turker answers: Sweden, Sweden, in Sweden, Sweden

[Radford et al. 2018] “Language Models are Unsupervised Multitask Learners”

GPT-2

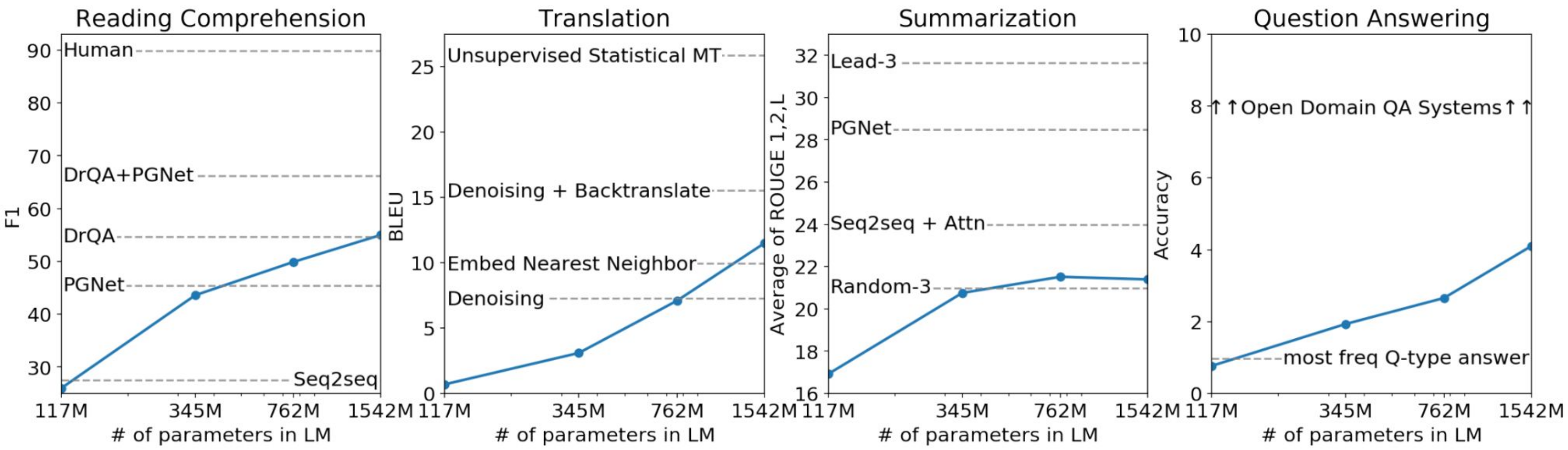
- Naturally occurring translation demonstrations in the pre-training corpus

”I’m not the cleverest man in the world, but like they say in French: **Je ne suis pas un imbecile [I’m not a fool].**
In a now-deleted post from Aug. 16, Soheil Eid, Tory candidate in the riding of Joliette, wrote in French: ”**Mentez mentez, il en restera toujours quelque chose**,” which translates as, ”**Lie lie and something will always remain.**”
“I hate the word ‘**perfume**,”’ Burr says. ‘It’s somewhat better in French: ‘**parfum**.’
If listened carefully at 29:55, a conversation can be heard between two guys in French: “**-Comment on fait pour aller de l’autre coté? -Quel autre coté?**”, which means “**- How do you get to the other side? - What side?**”.
If this sounds like a bit of a stretch, consider this question in French: **As-tu aller au cinéma?**, or **Did you go to the movies?**, which literally translates as Have-you to go to movies/theater?
“**Brevet Sans Garantie Du Gouvernement**”, translated to English: “**Patented without government warranty**”.

Table 1. Examples of naturally occurring demonstrations of English to French and French to English translation found throughout the WebText training set.

GPT-2

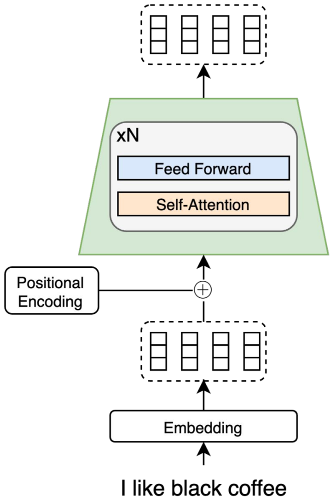
- Train self-supervised auto-regressive LMs on web text
  - Next-word prediction
  - Up to 1.5 billion parameters
- Observed non-trivial zero-shot performance



[Radford et al. 2018] “Language Models are Unsupervised Multitask Learners”

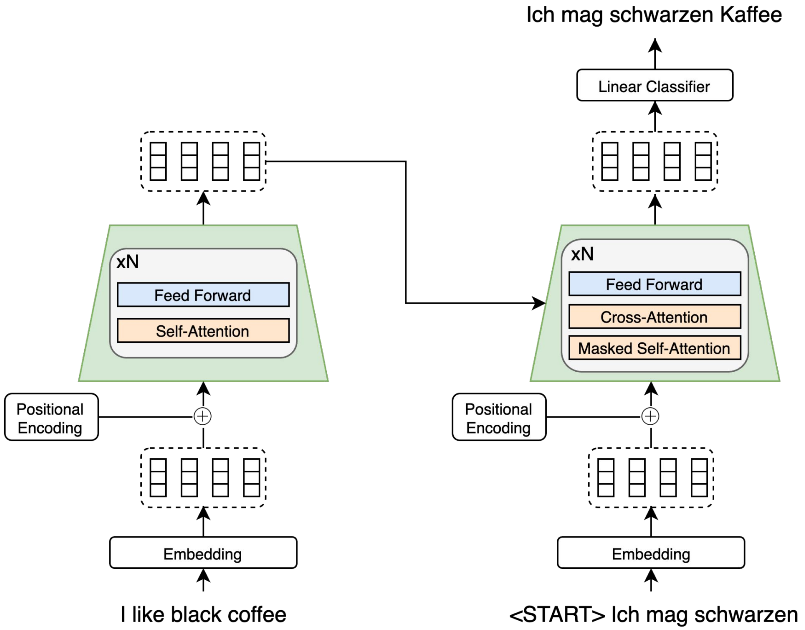
BERT

encoder only



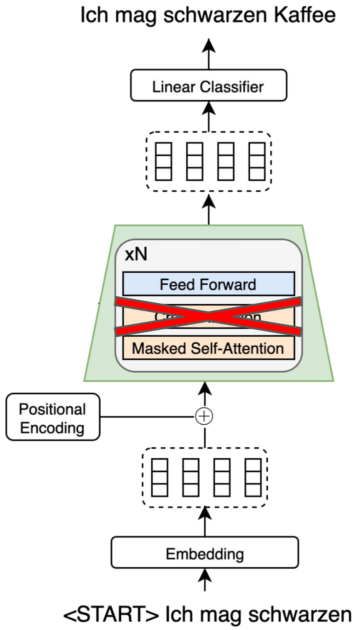
T5/BART

encoder-decoder model



GPT

decoder only

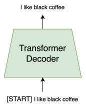


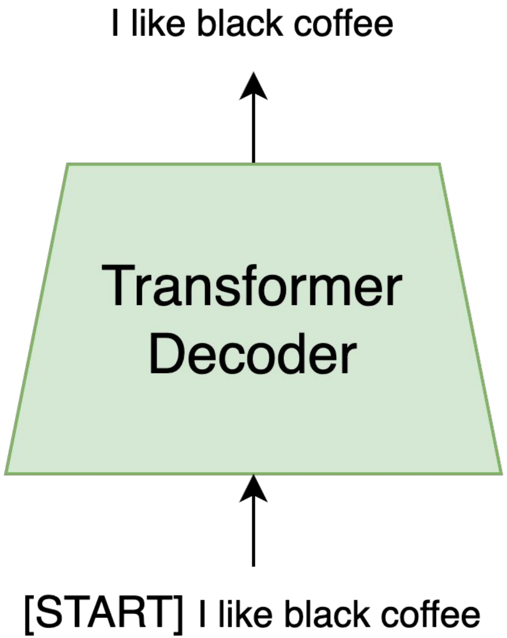
Part II: Scaling up!!

Cornell Bowers C·IS

GPT-3

- Train a bigger version of GPT-2 on more data





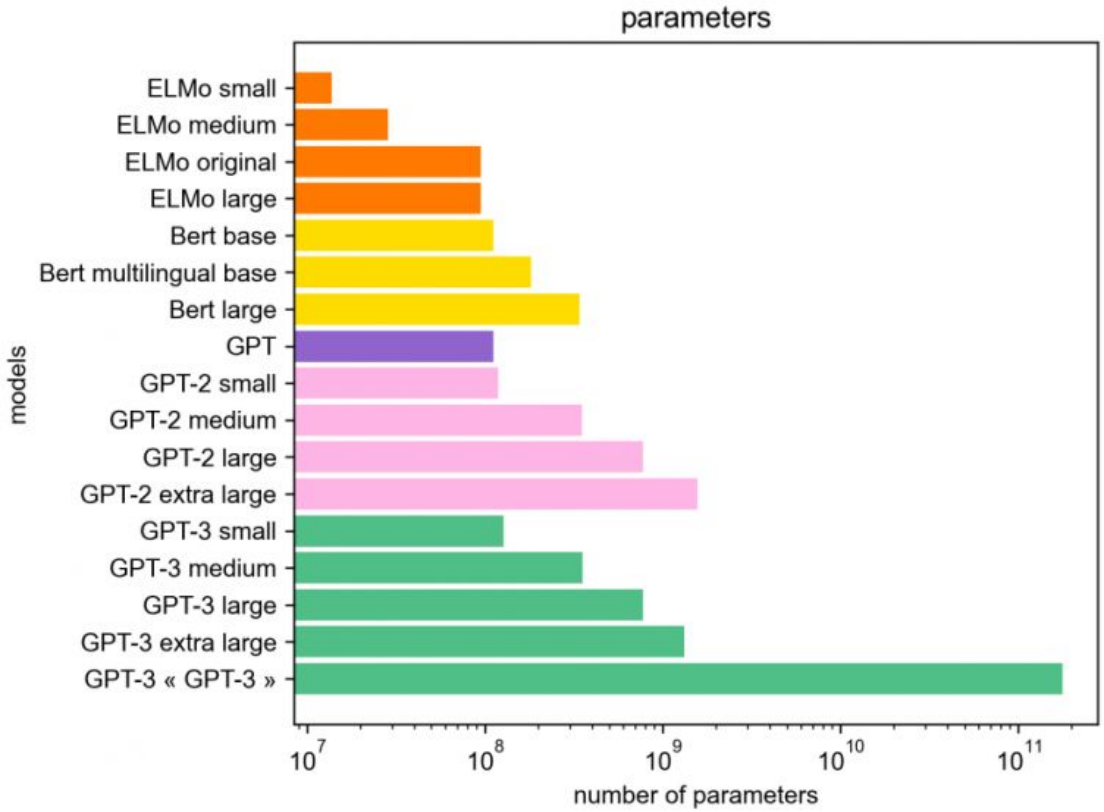
GPT-3

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.

Cornell Bowers C·IS

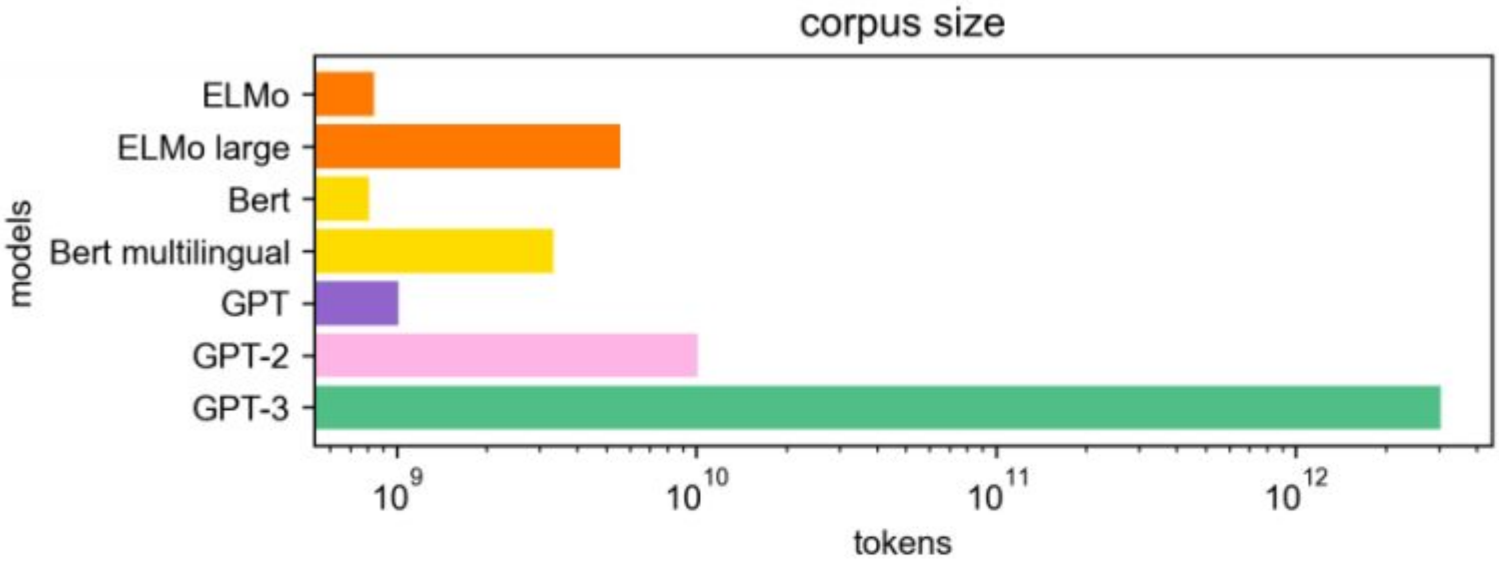
Scaling **Model** Size



Source: https://hellofuture.orange.com/en/the-gpt-3-language-model-revolution-or-evolution/

Cornell Bowers C·IS

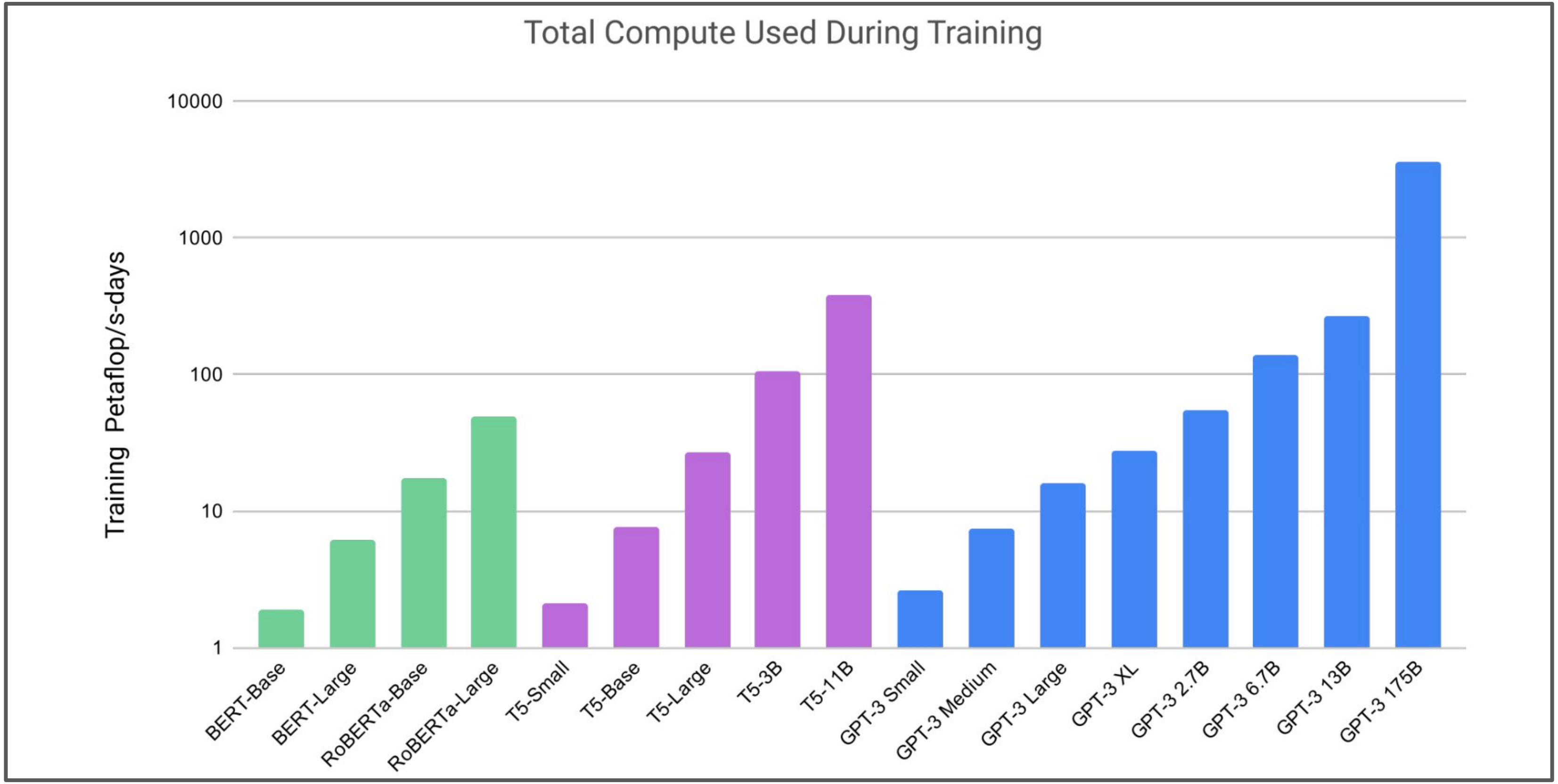
Scaling Training **Data**



Source: https://hellofuture.orange.com/en/the-gpt-3-language-model-revolution-or-evolution/

Cornell Bowers C·IS

Scaling Training **Compute**



Cornell Bowers C·IS

Discuss

- How many parameters in a single feedforward layer of GPT-3
  - An MLP with:
    - Input dimension of 12,228
    - One hidden layer with dimension 4*12,228
    - Output dimension of 12,228

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.

Discuss

- How many parameters in a single feedforward layer of GPT-3
  - An MLP with:
    - Input dimension of 12,228
    - One hidden layer with dimension 4*12,228
    - Output dimension of 12,228
- 12,228*4*12,228 + 12,228*4*12,228 = 1.2 billion!
  - 1 billion parameter MLPs!
  - BERT has 340M parameters!

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.

Cornell Bowers C·IS

Scaling Laws

[J. Kaplan et al. (2020)]

$$C = C_0 N D$$

$$L = \frac{A}{N^\alpha} + \frac{B}{D^\beta} + L_0$$

- Performance improves predictably with increased compute, data, and parameters
  - Can actually fit power laws!
  - Predict performance before training!

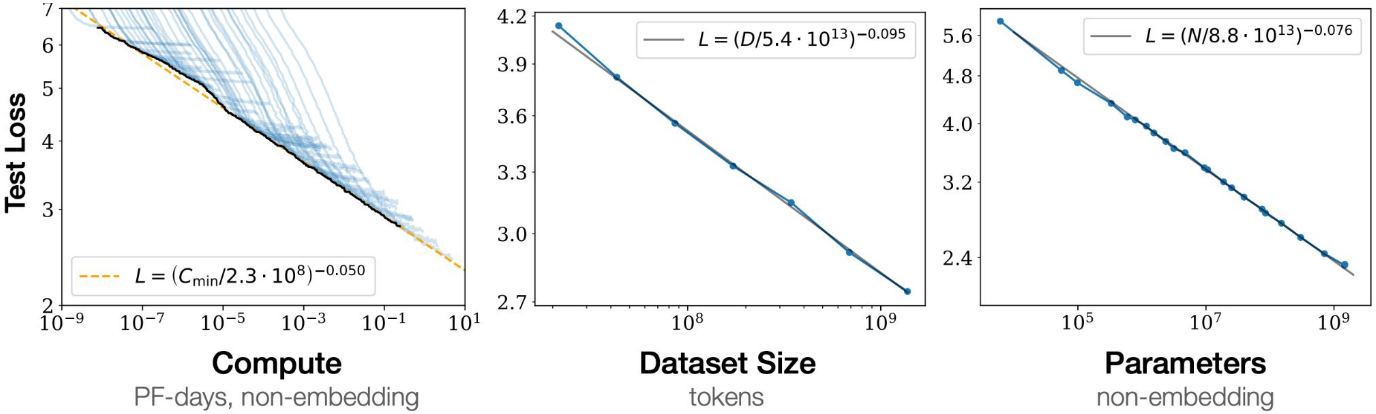


Figure 1 Language modeling performance improves smoothly as we increase the model size, datasetset size, and amount of compute[2] used for training. For optimal performance all three factors must be scaled up in tandem. Empirical performance has a power-law relationship with each individual factor when not bottlenecked by the other two.

Cornell Bowers C·IS

GPT-3 Evaluation

Zero-shot

The model predicts the answer given only a natural language description of the task. No gradient updates are performed.

```
Translate English to French:     ← task description
cheese =>                        ← prompt
```

One-shot

In addition to the task description, the model sees a single example of the task. No gradient updates are performed.

```
Translate English to French:     ← task description
sea otter => loutre de mer       ← example
cheese =>                        ← prompt
```

Few-shot

In addition to the task description, the model sees a few examples of the task. No gradient updates are performed.

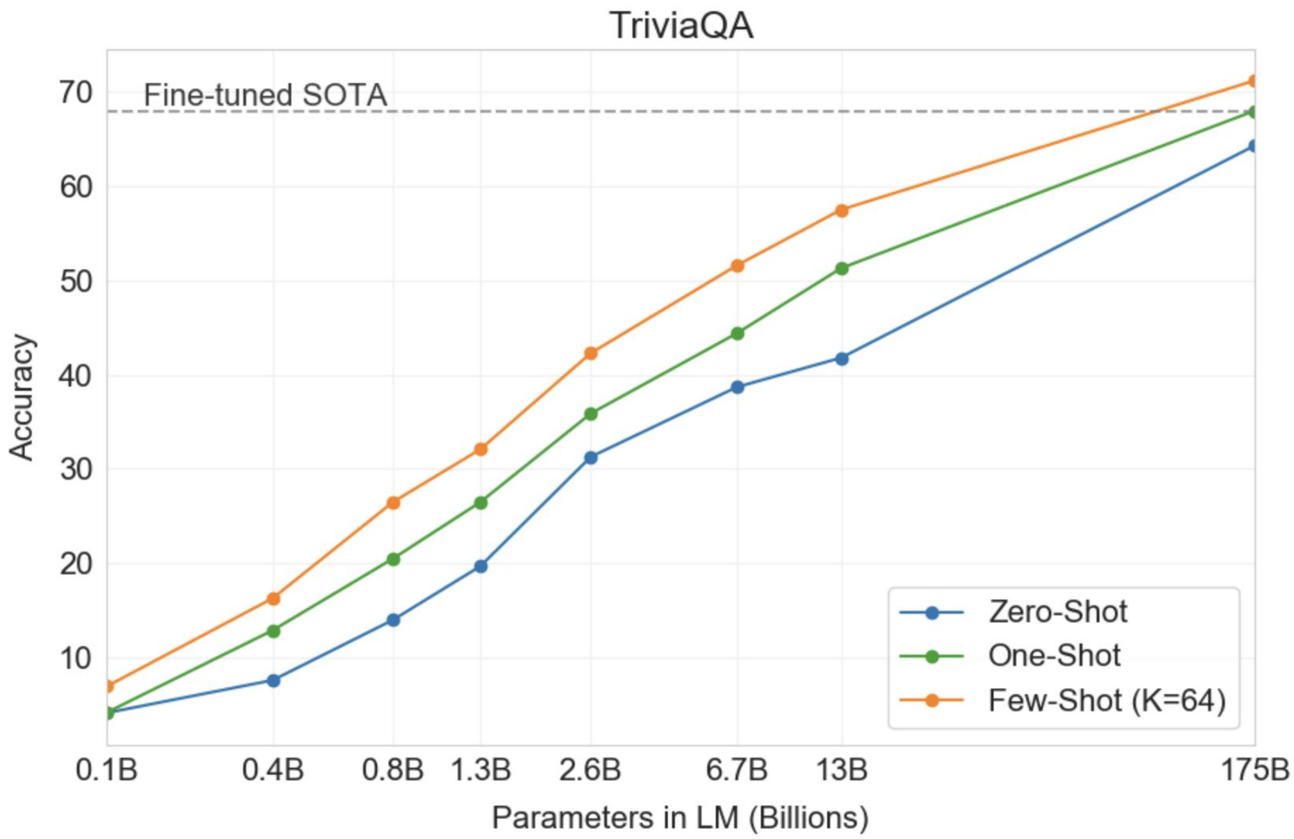
```
Translate English to French:     ← task description
sea otter => loutre de mer       ← examples
peppermint => menthe poivrée     ←
plush girafe => girafe peluche   ←
cheese =>                        ← prompt
```

Fine-tuning

The model is trained via repeated gradient updates using a large corpus of example tasks.

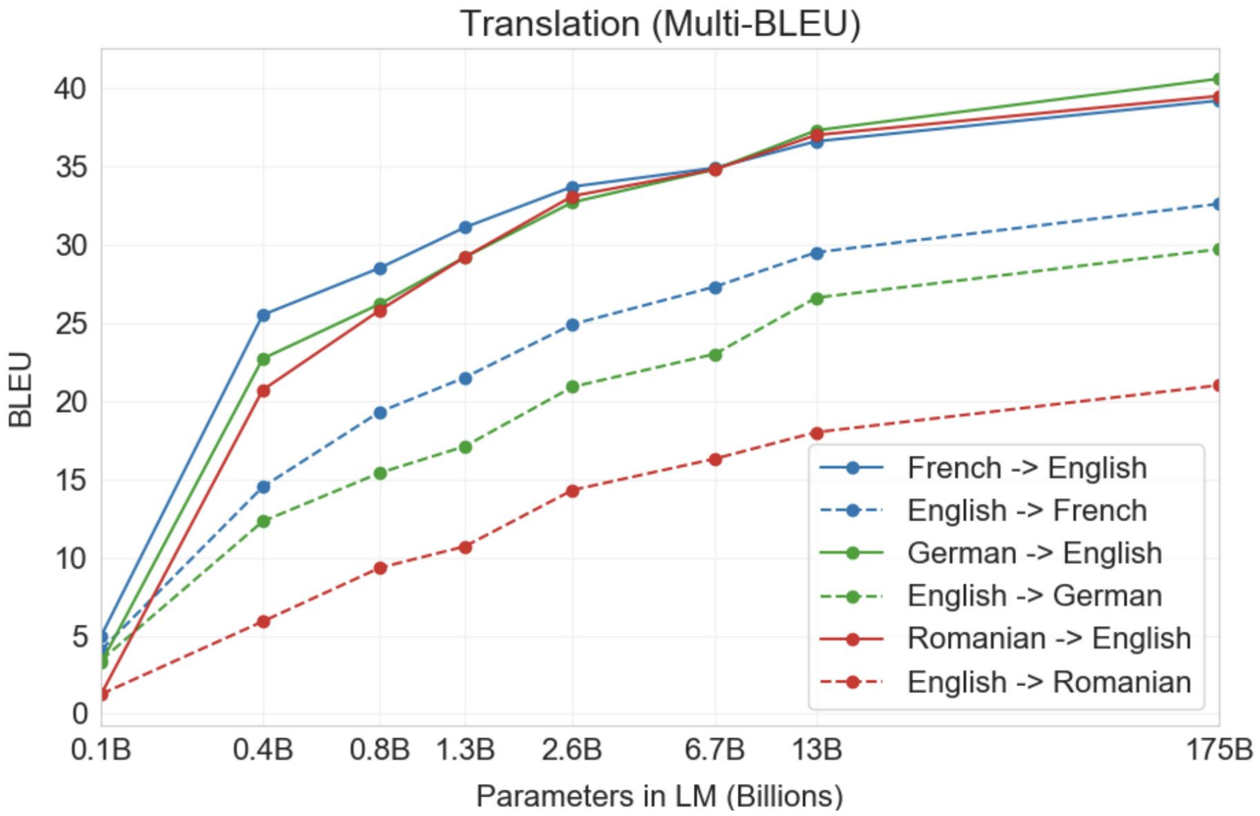
```
sea otter => loutre de mer       ← example #1
        ↓
gradient update
        ↓
peppermint => menthe poivrée     ← example #2
        ↓
gradient update
        ↓
       ...
        ↓
plush giraffe => girafe peluche  ← example #N

gradient update

cheese =>                        ← prompt
```

GPT-3 Evaluation

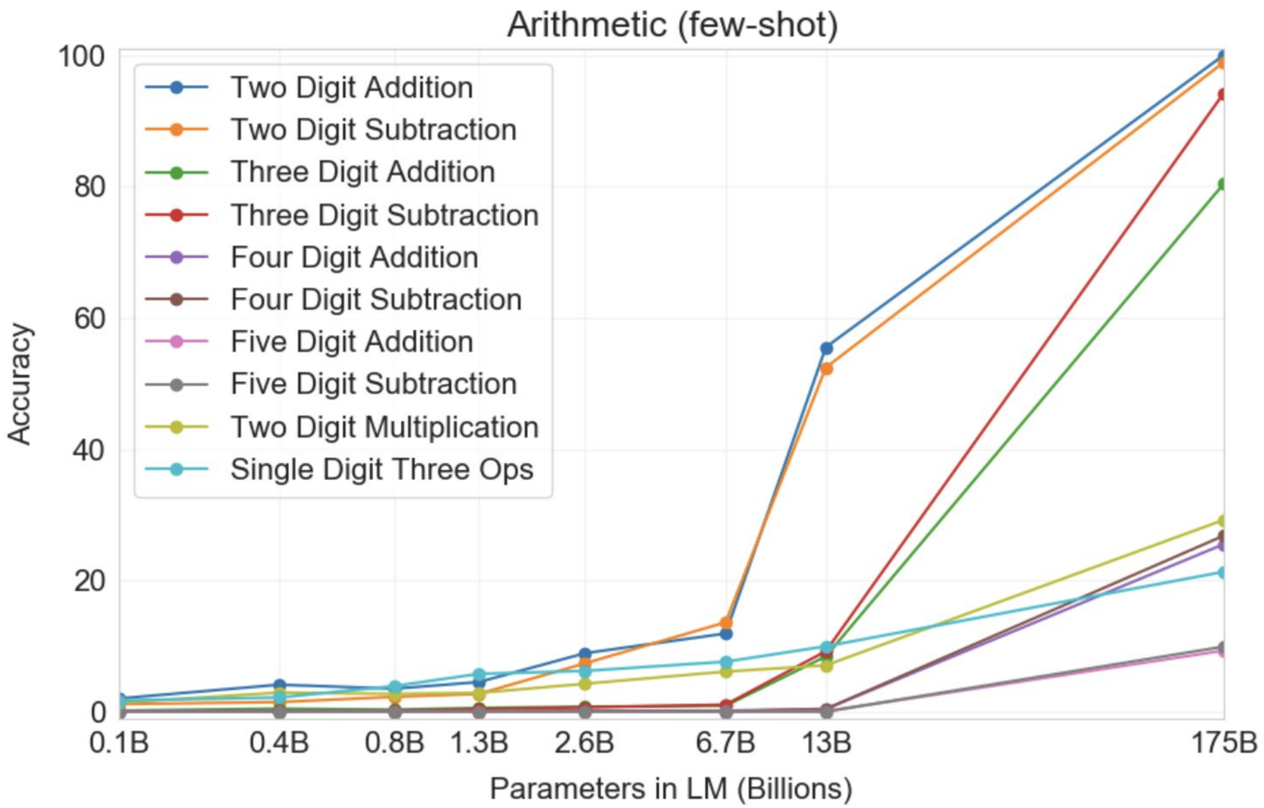


Cornell Bowers C·IS

GPT-3 Evaluation



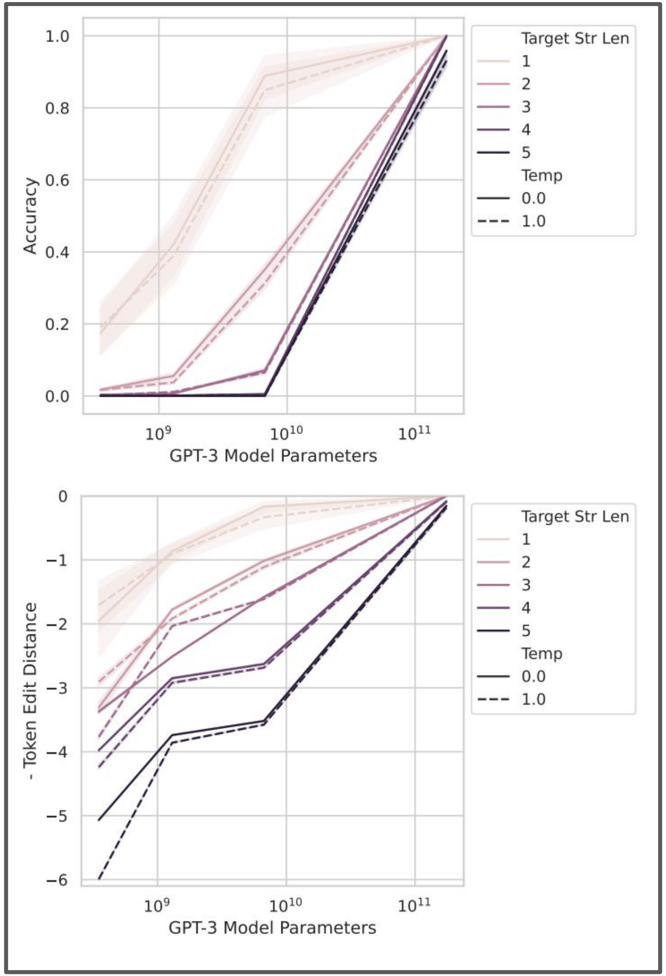
Emergent Capabilities



Cornell Bowers C·IS

Is Emergence a Mirage?

- Look at four digit addition under different metrics
  - Emergence can be an artefact of the evaluation metric
- Addition capabilities improve smoothly when using a more granular metric
  - Exact match accuracy -> Token edit distance



Schaeffer, Rylan, Brando Miranda, and Sanmi Koyejo. "Are emergent abilities of large language models a mirage?." Advances in Neural Information Processing Systems 36 (2024).

Cornell Bowers C·IS

Very Versatile

Explaining a Joke

I will explain these jokes:
(1) The problem with kleptomaniacs is that they always take things literally.
Explanation: This joke is wordplay. Someone who "takes things literally" is someone who doesn't fully understand social cues and context, which is a negative trait. But the definition of kleptomania is someone who literally takes things.
(2) Always borrow money from a pessimist. They'll never expect it back.
Explanation: Most people expect you to pay them back when you borrow money, however a pessimist is someone who always assumes the worst, so if you borrow money from them, they will expect that you won't pay them back anyways.

Input: I was supposed to start writing the paper at 5:00 PM. But then I started playing with this cool new language model for 10 minutes. 10 minutes later, it's suddenly 9:30 PM!
Model Output: This joke is about how time flies when you're having fun. The person was supposed to start writing a paper at 5:00 PM, but they started playing with a language model instead. They thought they were only playing with it for 10 minutes, but it was actually 4.5 hours.

Input: I think I understand why me and my partner are such a good match. I get excited when I smell food and become super alert when the doorbell rings. My girlfriend Shelley becomes ecstatic when she sees a person who she just saw 5 hours ago.
Model Output: This joke is about dogs. Dogs get excited when they smell food and when the doorbell rings. The girlfriend is excited to see someone she just saw 5 hours ago, which is a trait of dogs.

Chowdhery, Aakanksha, et al. "Palm: Scaling language modeling with pathways." Journal of Machine Learning Research 24.240 (2023): 1-113.

Cornell Bowers C·IS

Part III: Post-Training

Any problems with auto-regressive LMs?

Prompt:
What is the purpose of the list C in the code below?

```
def binomial_coefficient(n, r):
     C = [0 for i in range(r + 1)];
     C[0] = 1;
     for i in range(1, n + 1):
          j = min(i, r);
          while j > 0:
               C[j] += C[j - 1];
               j -= 1;
     return C[r]
```

Any problems with auto-regressive LMs?

Prompt:
What is the purpose of the list C in the code below?

```
def binomial_coefficient(n, r):
     C = [0 for i in range(r + 1)];
     C[0] = 1;
     for i in range(1, n + 1):
          j = min(i, r);
          while j > 0:
               C[j] += C[j - 1];
               j -= 1;
     return C[r]
```

GPT-3 175B completion:
A. to store the value of C[0]
B. to store the value of C[1]
C. to store the value of C[i]
D. to store the value of C[i - 1]

- Just trained on next-word prediction
    - Won't follow instructions!

Cornell Bowers C·IS

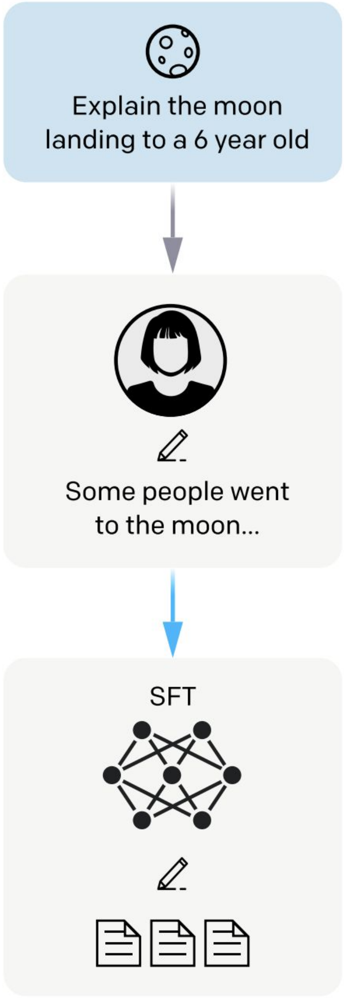
Instruction Tuning

- Curate a small dataset of instruction following demonstrations
  - Fine-tune the pre-trained LM
- Surprisingly **little** instruction data needed
  - 12,000 samples for Instruct GPT [Ouyang, 2022]

A prompt is sampled from our prompt dataset.

A labeler demonstrates the desired output behavior.

This data is used to fine-tune GPT-3 with supervised learning.



Instruction Tuning

- Aligns LM with user intent!

Prompt:
What is the purpose of the list C in the code below?

```
def binomial_coefficient(n, r):
     C = [0 for i in range(r + 1)];
     C[0] = 1;
     for i in range(1, n + 1):
          j = min(i, r);
          while j > 0:
               C[j] += C[j - 1];
               j -= 1;
     return C[r]
```

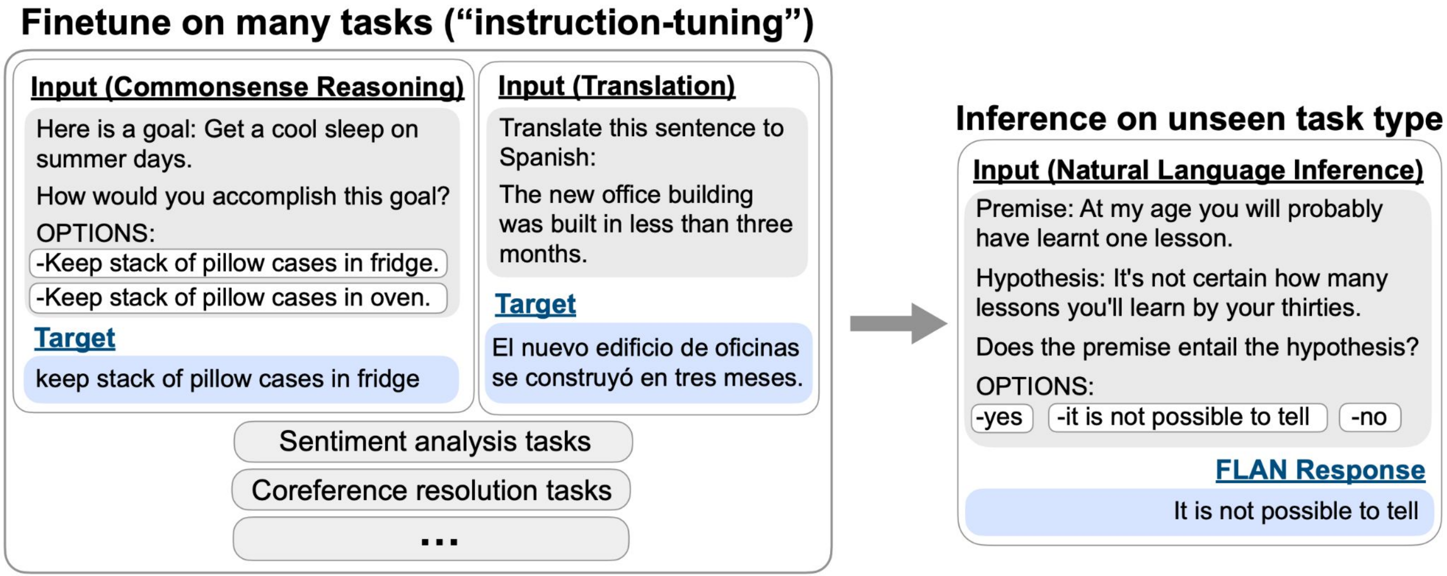
GPT-3 175B completion:	**InstructGPT 175B completion:**
A. to store the value of C[0] B. to store the value of C[1] C. to store the value of C[i] D. to store the value of C[i - 1]	The list C in this code is used to store the values of the binomial coefficient as the function iterates through the values of n and r. It is used to calculate the value of the binomial coefficient for a given value of n and r, and stores the result in the final return value of the function.

Ouyang, Long, et al. "Training language models to follow instructions with human feedback." Advances in Neural Information Processing Systems 35 (2022): 27730-27744.

Cornell Bowers C·IS

Instruction Tuning

- Model learns to generalize to unseen instructions at inference-time



Wei, Jason, et al. "Finetuned Language Models are Zero-Shot Learners." International Conference on Learning Representations. 2021.

Prompt Engineering

Standard Prompting

Model Input

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: The answer is 11.

Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

Model Output

A: The answer is 27. ✗

Chain-of-Thought Prompting

Model Input

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: Roger started with 5 balls. 2 cans of 3 tennis balls each is 6 tennis balls. 5 + 6 = 11. The answer is 11.

Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

Model Output

A: The cafeteria had 23 apples originally. They used 20 to make lunch. So they had 23 - 20 = 3. They bought 6 more apples, so they have 3 + 6 = 9. The answer is 9. ✓

Wei, Jason, et al. "Chain-of-thought prompting elicits reasoning in large language models." Advances in Neural Information Processing Systems 35 (2022): 24824-24837.

Cornell Bowers CIS

Chain of Thought Prompting

- Break down problem into subparts
  - Learns to to it with few-shot in-context learning

Math Word Problems (free response)

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: Roger started with 5 balls. 2 cans of 3 tennis balls each is 6 tennis balls. 5 + 6 = 11. The answer is 11.

Math Word Problems (multiple choice)

Q: How many keystrokes are needed to type the numbers from 1 to 500? Answer Choices: (a) 1156 (b) 1392 (c) 1480 (d) 1562 (e) 1788

A: There are 9 one-digit numbers from 1 to 9. There are 90 two-digit numbers from 10 to 99. There are 401 three-digit numbers from 100 to 500. 9 + 90(2) + 401(3) = 1392. The answer is (b).

CSQA (commonsense)

Q: Sammy wanted to go to where the people were. Where might he go? Options: (a) race track (b) populated areas (c) desert (d) apartment (e) roadblock

A: The answer must be a place with a lot of people. Race tracks, desert, apartments, and roadblocks don't have a lot of people, but populated areas do. So the answer is (b).

StrategyQA

Q: Yes or no: Would a pear sink in water?

A: The density of a pear is about 0.6 g/cm^3, which is less than water. Thus, a pear would float. So the answer is no.

Date Understanding

Q: The concert was scheduled to be on 06/01/1943, but was delayed by one day to today. What is the date 10 days ago in MM/DD/YYYY?

A: One day after 06/01/1943 is 06/02/1943, so today is 06/02/1943. 10 days before today is 05/23/1943. So the answer is 05/23/1943.

Sports Understanding

Q: Is the following sentence plausible? "Joao Moutinho caught the screen pass in the NFC championship."

A: Joao Moutinho is a soccer player. The NFC championship is part of American football, not soccer. So the answer is no.

SayCan (Instructing a robot)

Human: How would you bring me something that isn't a fruit?

Explanation: the user wants something to eat that isn't a fruit. An energy bar is not a fruit, so I will bring the user an energy bar.
Plan: 1. find(energy bar) 2. pick(energy bar) 3. find(user) 4. put(energy bar) 5. done().

Last Letter Concatenation

Q: Take the last letters of the words in "Lady Gaga" and concatenate them.

A: The last letter of "Lady" is "y". The last letter of "Gaga" is "a". Concatenating them is "ya". So the answer is ya.

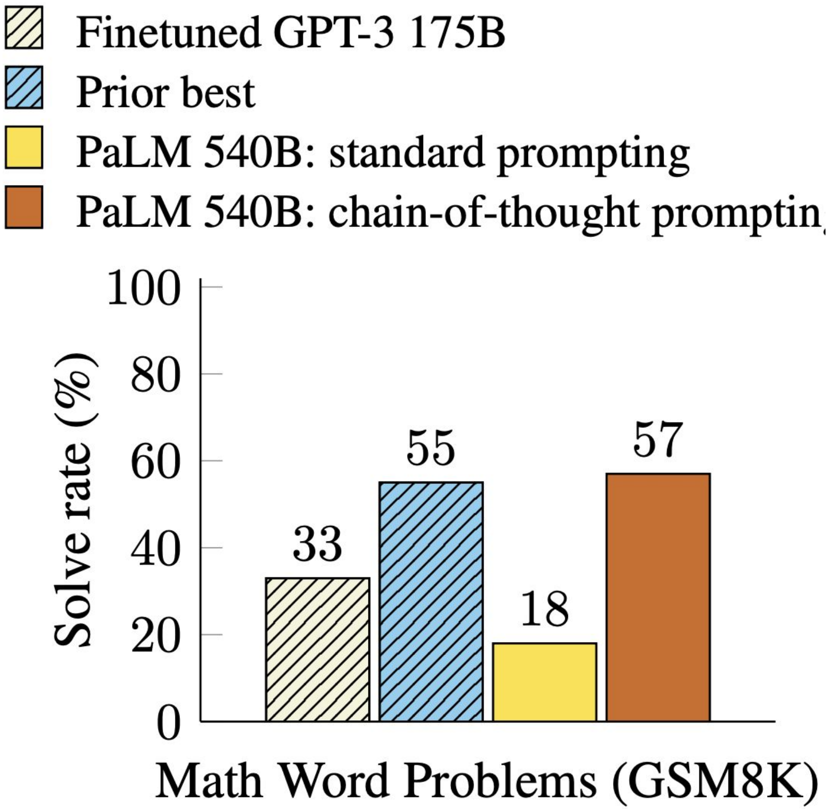
Coin Flip (state tracking)

Q: A coin is heads up. Maybelle flips the coin. Shalonda does not flip the coin. Is the coin still heads up?

A: The coin was flipped by Maybelle. So the coin was flipped 1 time, which is an odd number. The coin started heads up, so after an odd number of flips, it will be tails up. So the answer is no.

Chain of Thought Prompting

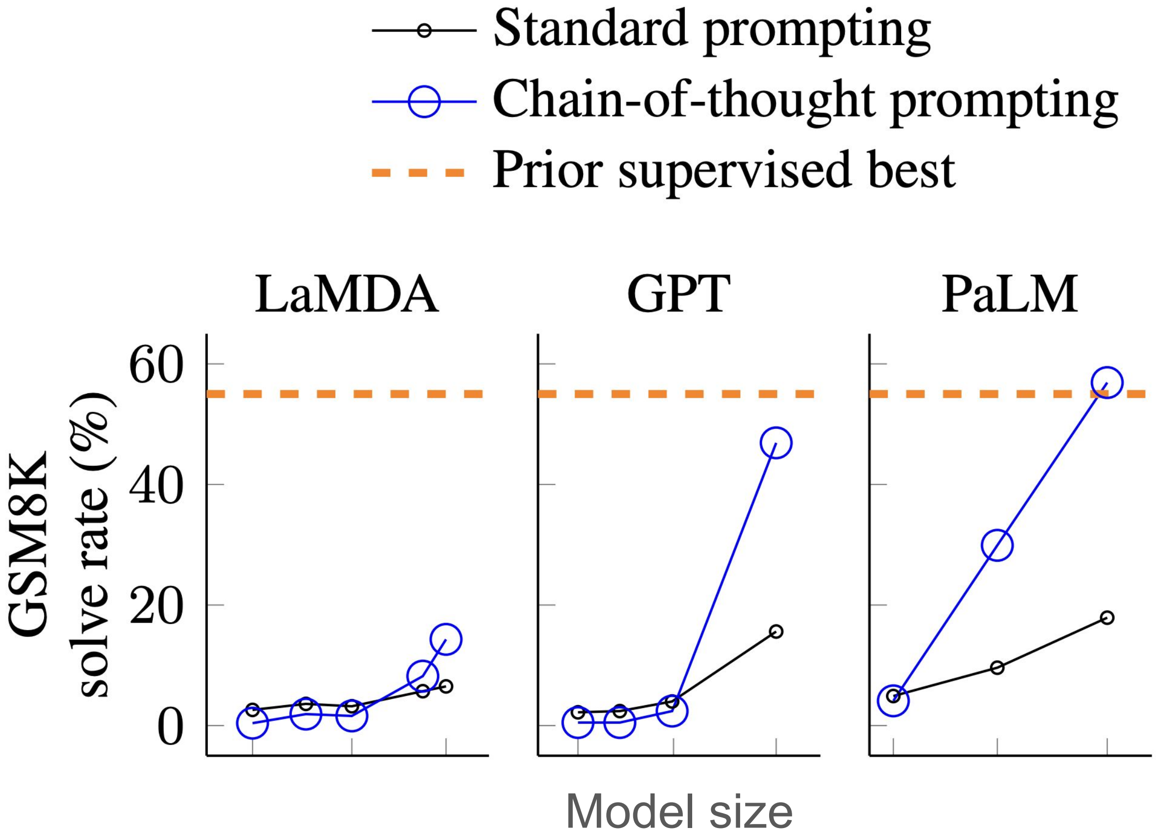
- Can significantly improve performance



Cornell Bowers C·IS

Chain of Thought Prompting

- No benefit at small scales



Cornell Bowers C·IS

Sensitivity to the Prompt

- Performance depends a lot on the prompt!

Table 4: Robustness study against template measured on the MultiArith dataset with text-davinci-002. (*1) This template is used in Ahn et al. [2022] where a language model is prompted to generate step-by-step actions given a high-level instruction for controlling robotic actions. (*2) This template is used in Reynolds and McDonell [2021] but is not quantitatively evaluated.

No.	Category	Template	Accuracy
1	instructive	Let's think step by step.	**78.7**
2		First, (*1)	77.3
3		Let's think about this logically.	74.5
4		Let's solve this problem by splitting it into steps. (*2)	72.2
5		Let's be realistic and think step by step.	70.8
6		Let's think like a detective step by step.	70.3
7		Let's think	57.5
8		Before we dive into the answer,	55.7
9		The answer is after the proof.	45.7
10	misleading	Don't think. Just feel.	18.8
11		Let's think step by step but reach an incorrect answer.	18.7
12		Let's count the number of "a" in the question.	16.7
13		By using the fact that the earth is round,	9.3
14	irrelevant	By the way, I found a good restaurant nearby.	17.5
15		Abrakadabra!	15.5
16		It's a beautiful day.	13.1
-		(Zero-shot)	17.7

Kojima, Takeshi, et al. "Large language models are zero-shot reasoners." Advances in neural information processing systems 35 (2022): 22199-22213.

Cornell Bowers C·IS

Toolformer

- LLMs can be designed to use tools like calculators, QA systems and calendars to produce better results
- LLM are trained to generate and issue queries to external tools

The New England Journal of Medicine is a registered trademark of **[QA("Who is the publisher of The New England Journal of Medicine?") → Massachusetts Medical Society]** the MMS.

Out of 1400 participants, 400 (or **[Calculator(400 / 1400) → 0.29]** 29%) passed the test.

The name derives from "la tortuga", the Spanish word for **[MT("tortuga") → turtle]** turtle.

The Brown Act is California's law **[WikiSearch("Brown Act") → The Ralph M. Brown Act is an act of the California State Legislature that guarantees the public's right to attend and participate in meetings of local legislative bodies.]** that requires legislative bodies, like city councils, to hold their meetings open to the public.

Schick, Timo, et al. "Toolformer: Language models can teach themselves to use tools." Advances in Neural Information Processing Systems 36 (2024).

Limitations: Hallucinations

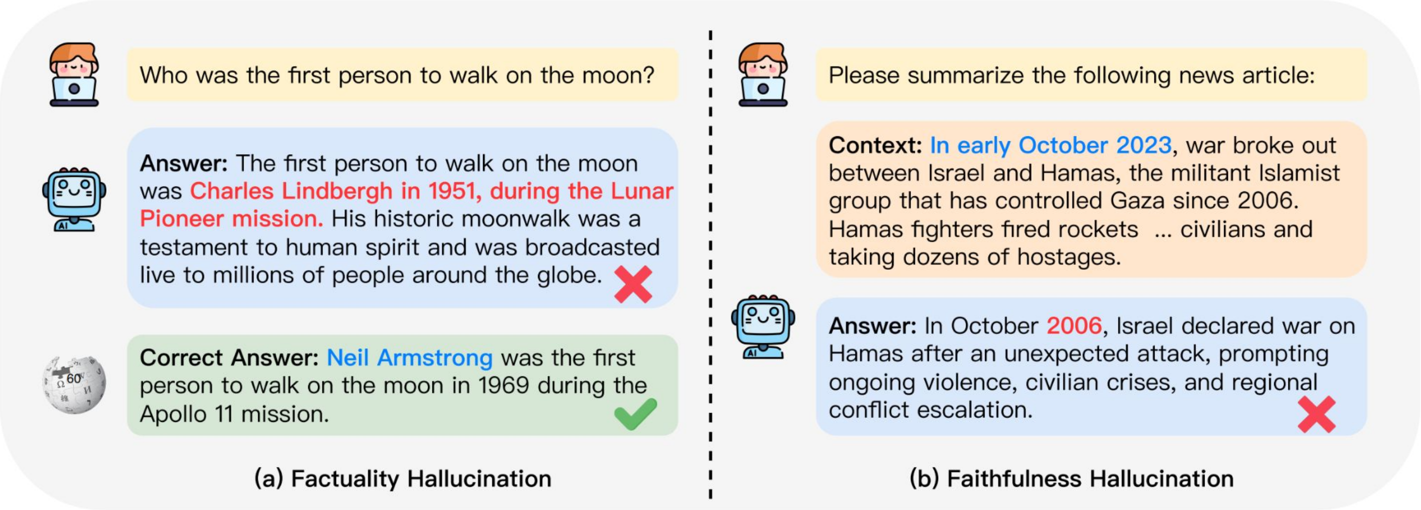


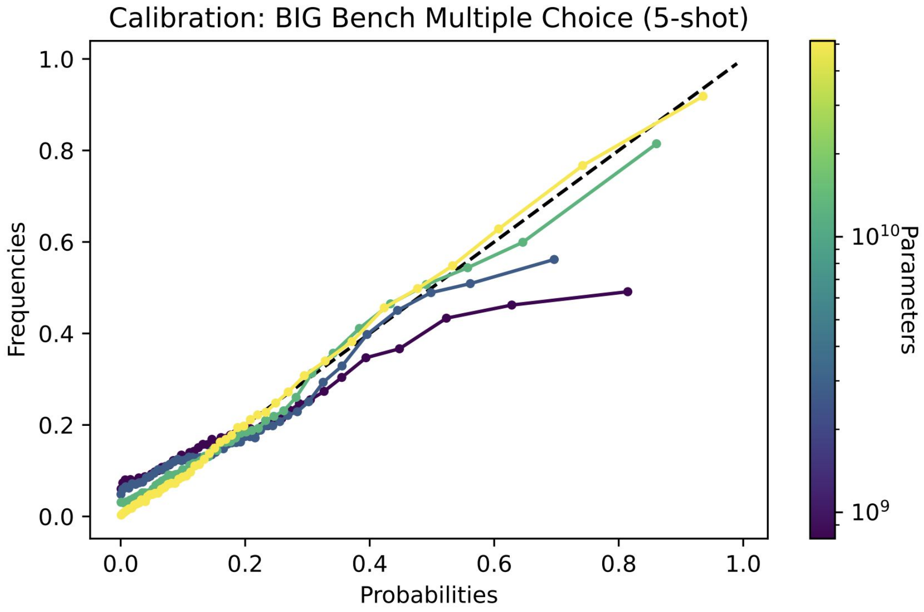
Figure 1: An intuitive example of LLM hallucination.

“A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions”

Cornell Bowers C·IS

How to Incorporate Uncertainty

- Language Models (Mostly) Know What They Know
  - Kadavath et al. (2022)
- LMs are pretty well calibrated!
  - Multiple choice evaluation



Cornell Bowers C·IS

Significant Energy Consumption

Consumption	**CO_2e (lbs)**
Air travel, 1 passenger, NY↔SF	1984
Human life, avg, 1 year	11,023
American life, avg, 1 year	36,156
Car, avg incl. fuel, 1 lifetime	126,000
Training one model (GPU)	
NLP pipeline (parsing, SRL)	39
w/ tuning & experimentation	78,468
Transformer (big)	192
w/ neural architecture search	626,155

Strubell, E., Ganesh, A., & McCallum, A. (2019, July). Energy and Policy Considerations for Deep Learning in NLP. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics* (pp. 3645-3650).

Cornell Bowers C·IS

Recap

- Different variants of language models
  - Encoder only, encoder-decoder, decoder only
- Performance improves with data and model size
  - Can actually fit power laws and predict performance before training!
- LLMs learn how to do a variety of tasks during pre-training
- A second phase is required to give pre-trained LMs instruction following capabilities
  - Instruction tuning
- Performance can depend on the prompt
- LMs often hallucinate!