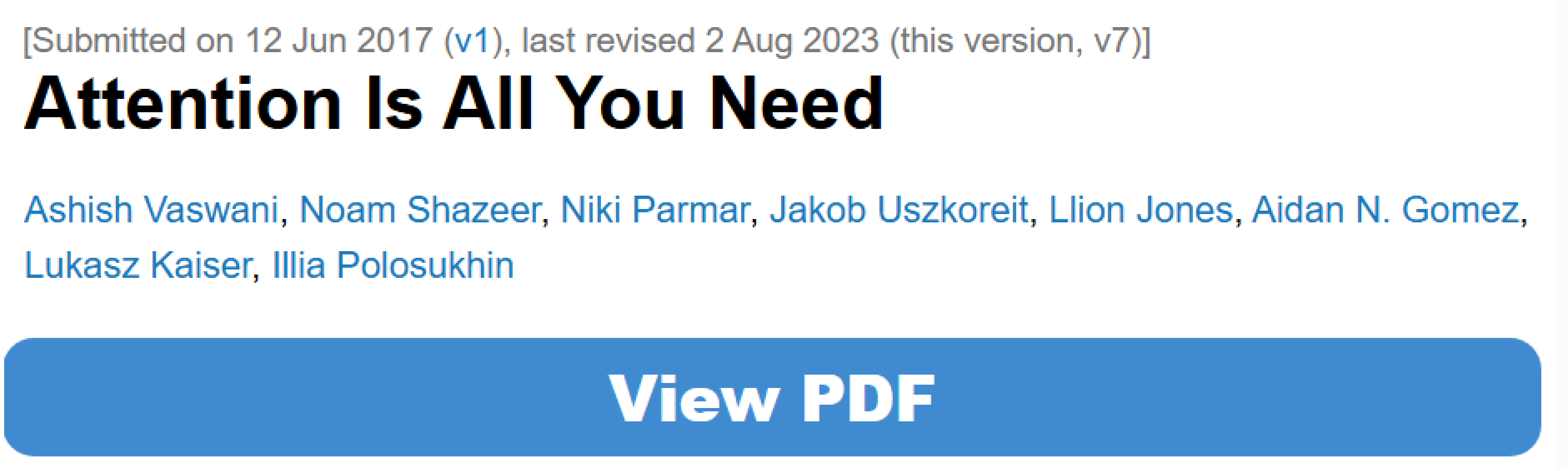
Large Language Models

Gautam Kamath

Last time...



[Submitted on 12 Jun 2017 (v1), last revised 2 Aug 2023 (this version, v7)]

Attention Is All You Need

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, Illia Polosukhin

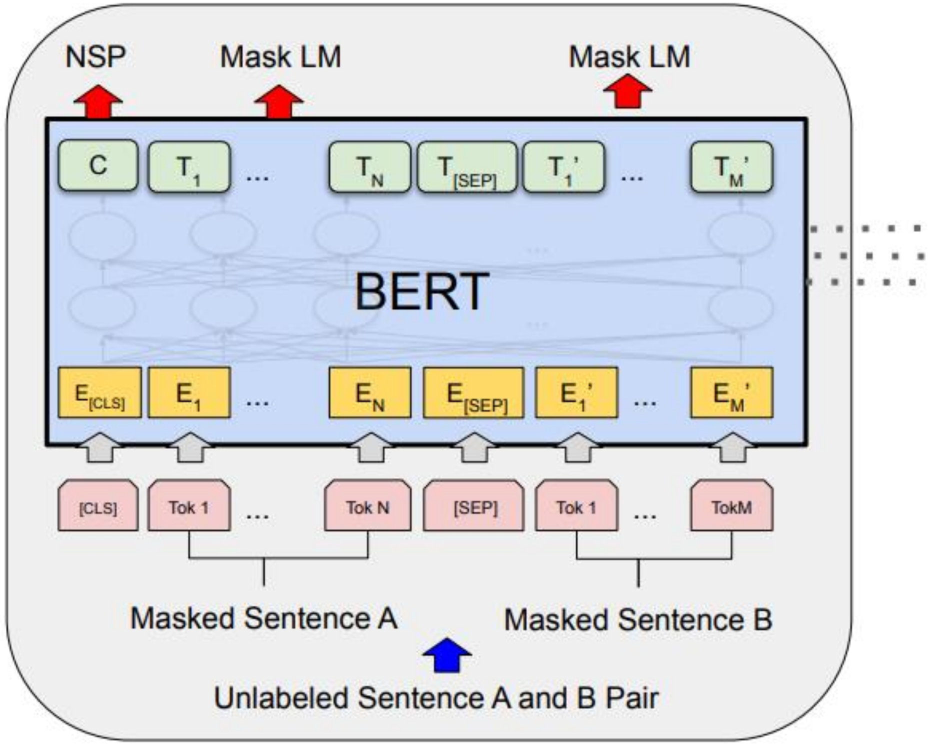
What has happened since June 2017?

General Language Understanding Evaluation (GLUE)

- Benchmark Natural Language Understanding
    - Read: classification tasks
- Collection of nine tasks
- E.g., Stanford Question Answering Dataset (SQuAD)
    - Text: At University of Waterloo, CS 480 is the course “Introduction to Machine Learning.” It is taught by Gautam Kamath.
    - Q: Who teaches CS 480?
    - A: 18, 19
        - Indices of start and end of answer “Gautam Kamath”

BERT (Bidirectional Encoder Representations from Transformers)

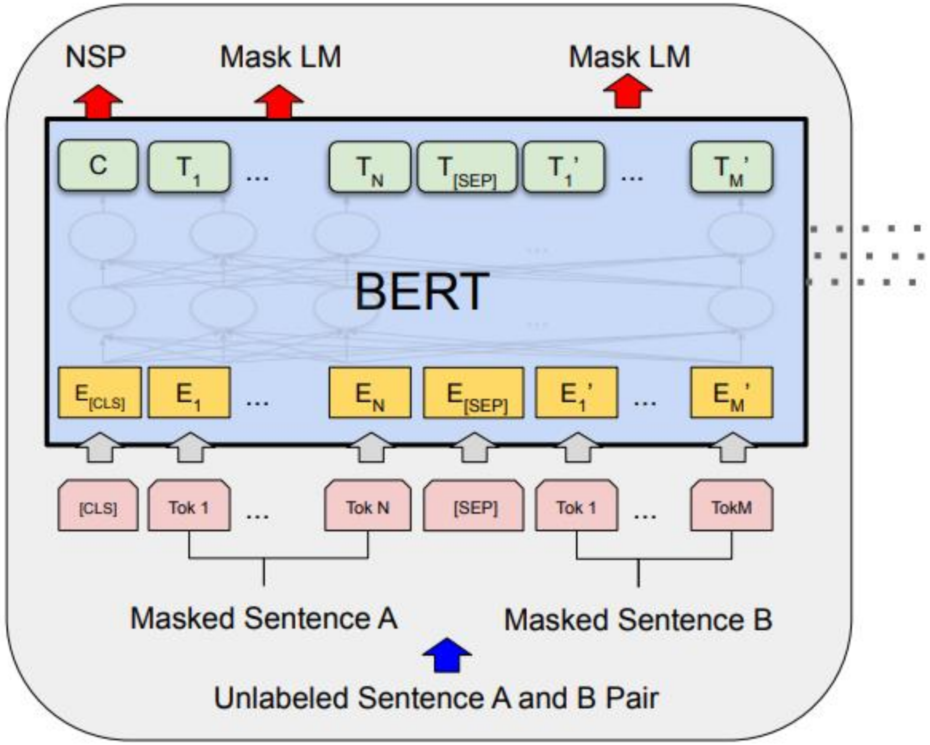
- Google, October 2018
- Two phases of training
  - Pre-training: self-supervised learning to "understand language"
  - Fine-tuning: supervised learning to specialize to task
- "Encoder-only" architecture
- [CLS] token
- [SEP] token
- 340M parameters



Pre-training

BERT Pre-training

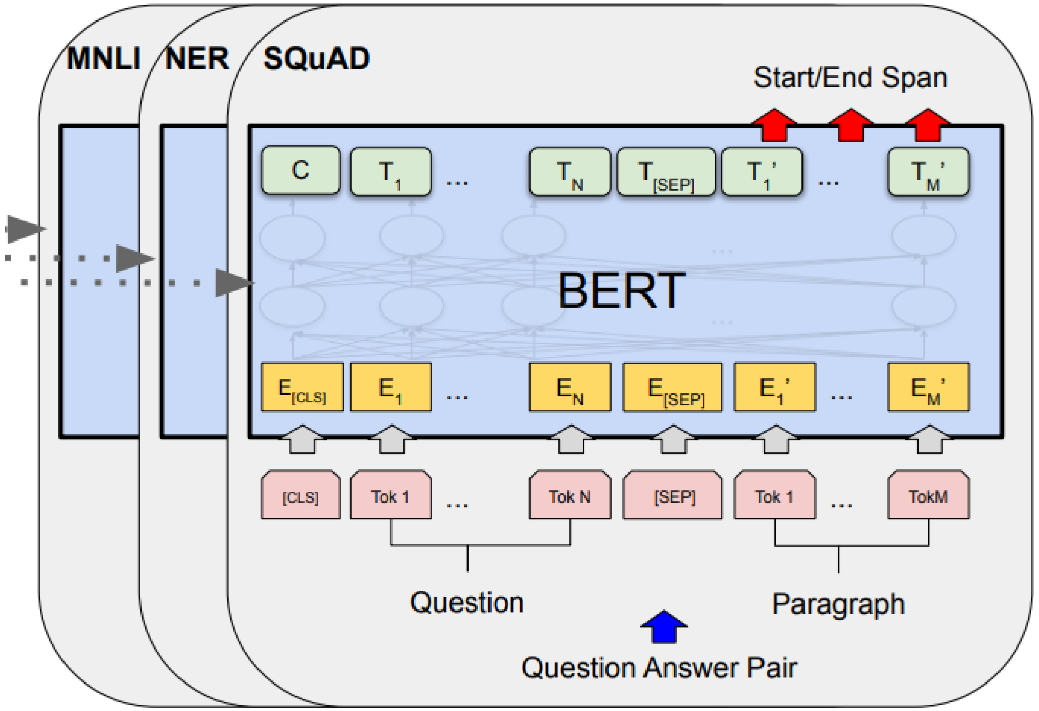
- BooksCorpus 🤔 and Wikipedia
- Two self-supervised tasks
- Masked Language Modeling
  - The algorithm minimizes the prediction error of [MASK].
    - Static Masking
- Next Sentence Prediction
  - [CLS] The algorithm minimizes the loss. [SEP] It converges after ten epochs.



Pre-training

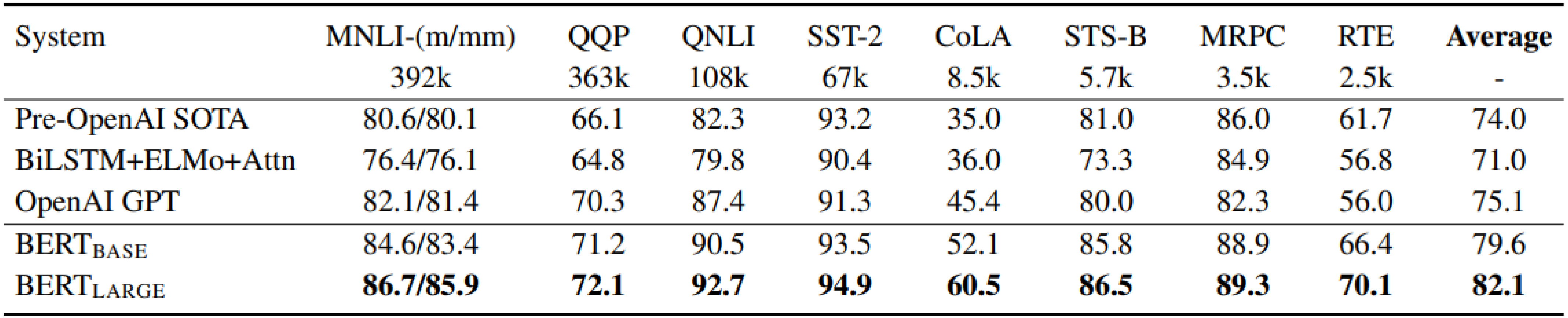
BERT Fine-Tuning

- Remove Mask LM and NSP head
- Attach new head for task you care about
- Continue training model weights on training set of task
  - E.g., SQuAD or other dataset



Fine-Tuning

BERT Results

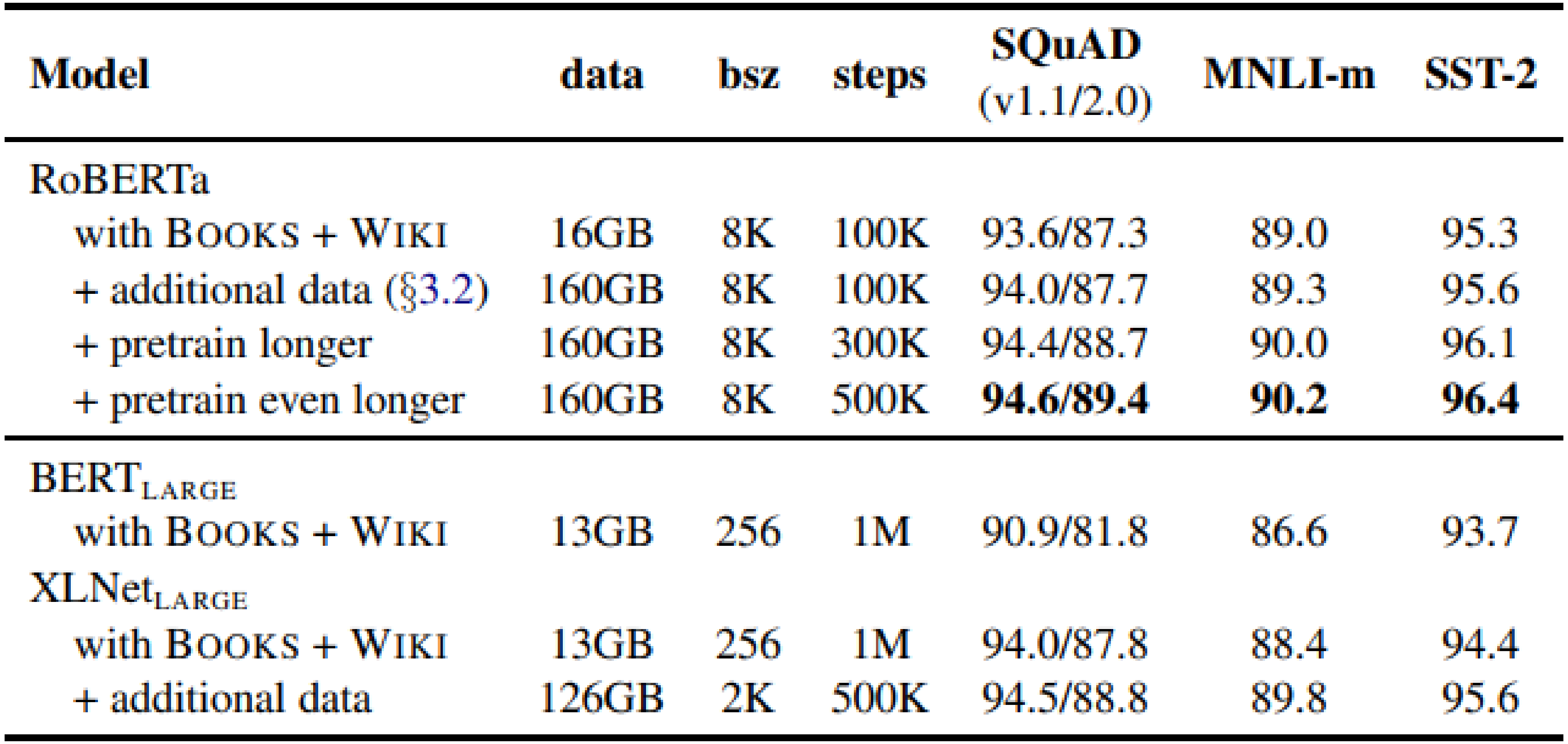


System	MNLI-(m/mm)	QQP	QNLI	SST-2	CoLA	STS-B	MRPC	RTE	**Average**
	392k	363k	108k	67k	8.5k	5.7k	3.5k	2.5k	-
Pre-OpenAI SOTA	80.6/80.1	66.1	82.3	93.2	35.0	81.0	86.0	61.7	74.0
BiLSTM+ELMo+Attn	76.4/76.1	64.8	79.8	90.4	36.0	73.3	84.9	56.8	71.0
OpenAI GPT	82.1/81.4	70.3	87.4	91.3	45.4	80.0	82.3	56.0	75.1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

RoBERTa (Robustly Optimized BERT)

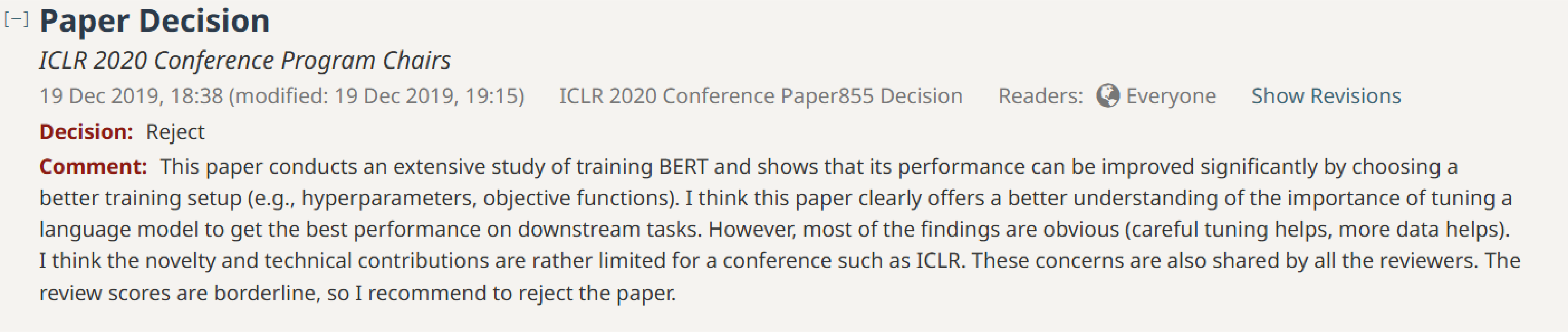
- Facebook and University of Washington, July 2019
- Remove NSP pre-training objective
- Dynamic masking
- Trained on 10x data (16 GB -> 160 GB)
    - Includes CC-News, OpenWebText, Stories (explain datasets)
- Longer training (~15x more tokens)
    - BERT: 1 million steps with batch size 256
    - RoBERTa: 500 thousand steps with batch size 8,000

RoBERTa Results



Model	data	bsz	steps	SQuAD (v1.1/2.0)	MNLI-m	SST-2
RoBERTa						
with BOOKS + WIKI	16GB	8K	100K	93.6/87.3	89.0	95.3
+ additional data (§3.2)	160GB	8K	100K	94.0/87.7	89.3	95.6
+ pretrain longer	160GB	8K	300K	94.4/88.7	90.0	96.1
+ pretrain even longer	160GB	8K	500K	**94.6/89.4**	**90.2**	**96.4**
BERT$_{\text{LARGE}}$						
with BOOKS + WIKI	13GB	256	1M	90.9/81.8	86.6	93.7
XLNet$_{\text{LARGE}}$						
with BOOKS + WIKI	13GB	256	1M	94.0/87.8	88.4	94.4
+ additional data	126GB	2K	500K	94.5/88.8	89.8	95.6

...RoBERTa Reviews



[−] **Paper Decision**

ICLR 2020 Conference Program Chairs

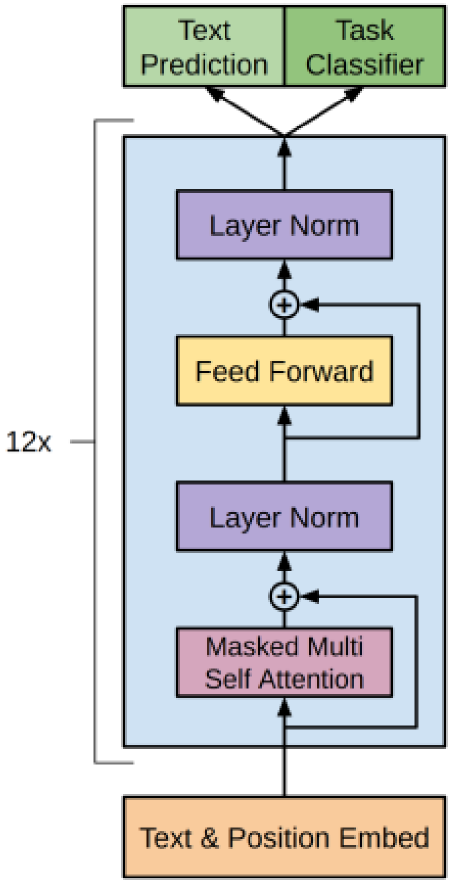
19 Dec 2019, 18:38 (modified: 19 Dec 2019, 19:15) ICLR 2020 Conference Paper855 Decision Readers: 🌐 Everyone Show Revisions

Decision: Reject

Comment: This paper conducts an extensive study of training BERT and shows that its performance can be improved significantly by choosing a better training setup (e.g., hyperparameters, objective functions). I think this paper clearly offers a better understanding of the importance of tuning a language model to get the best performance on downstream tasks. However, most of the findings are obvious (careful tuning helps, more data helps). I think the novelty and technical contributions are rather limited for a conference such as ICLR. These concerns are also shared by all the reviewers. The review scores are borderline, so I recommend to reject the paper.

GPT-1 (Generative Pre-trained Transformer)

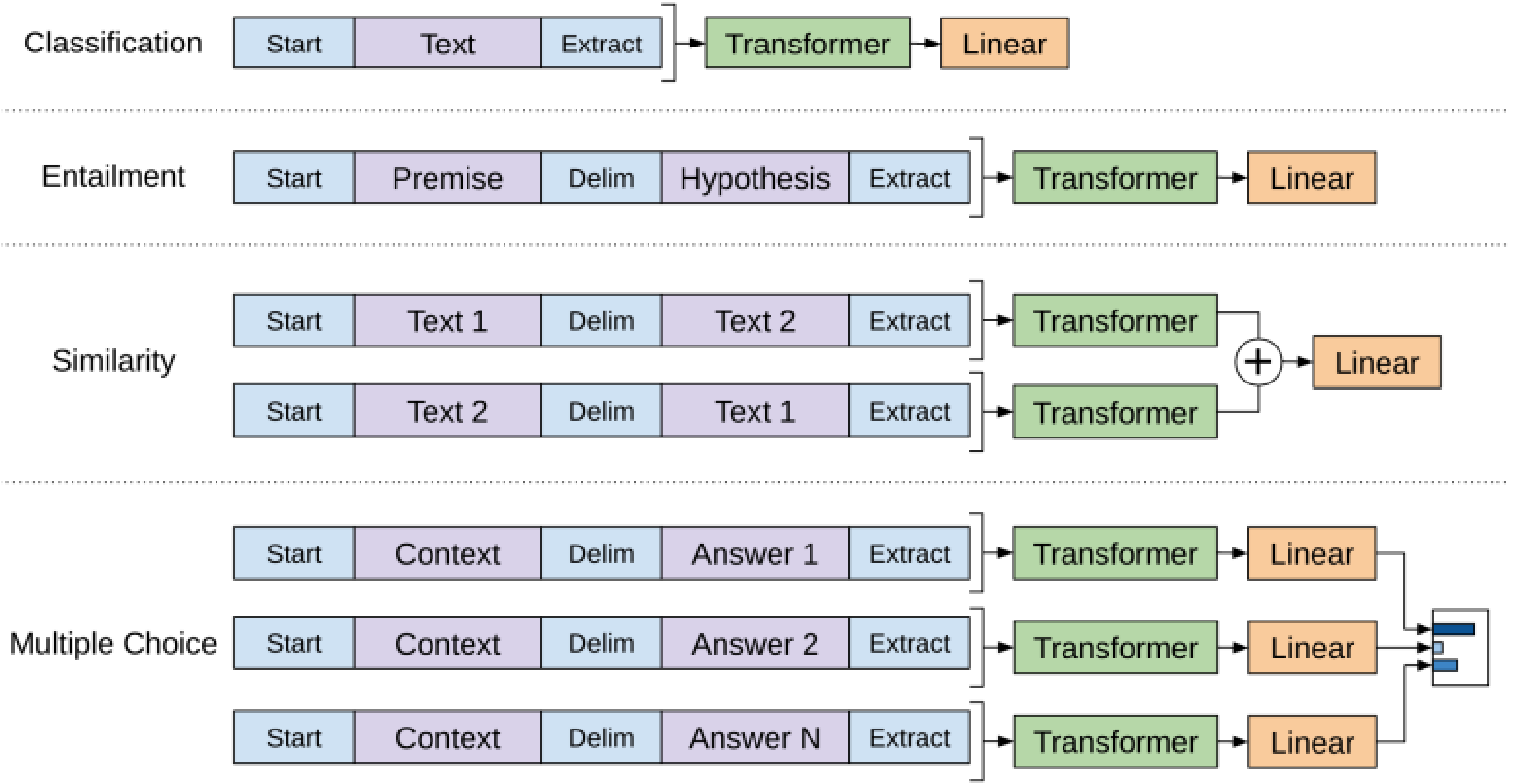
- OpenAI, June 2018
- Two phases of training
  - Pre-training
  - Fine-tuning
- “Decoder-only” architecture
- 117M parameters



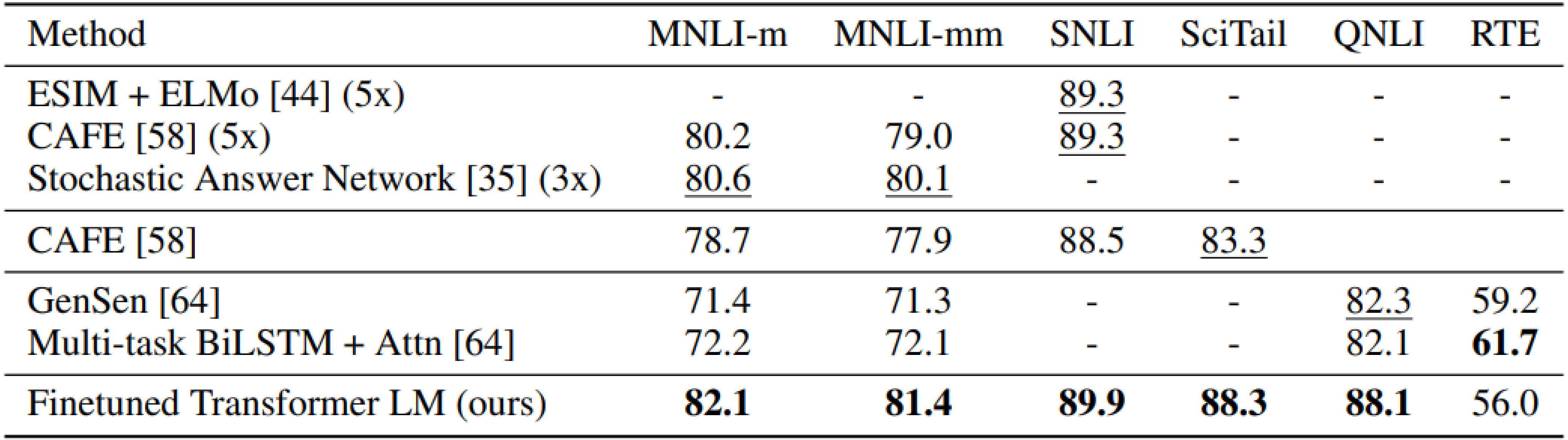
GPT-1 Pre-training

- “Next word prediction” (really token)
- Maximum likelihood on next token given k previous tokens
    - k is Attention’s context window size, 512 for GPT-1
- Versus BERT
- BooksCorpus

GPT-1 Fine-Tuning



GPT-1 Results



Method	MNLI-m	MNLI-mm	SNLI	SciTail	QNLI	RTE
ESIM + ELMo [44] (5x)	-	-	89.3	-	-	-
CAFE [58] (5x)	80.2	79.0	89.3	-	-	-
Stochastic Answer Network [35] (3x)	80.6	80.1	-	-	-	-
CAFE [58]	78.7	77.9	88.5	83.3		
GenSen [64]	71.4	71.3	-	-	82.3	59.2
Multi-task BiLSTM + Attn [64]	72.2	72.1	-	-	82.1	**61.7**
Finetuned Transformer LM (ours)	**82.1**	**81.4**	**89.9**	**88.3**	**88.1**	56.0

GPT-1 -> GPT-2

- Open AI, February 2019
  - Model released in November 2019
  - …too dangerous to release?
- Same architecture as GPT-1…
  - …But 10x bigger (1.5B parameters)
- WebText
  - What's wrong with Common Crawl?
- Purely generative, no fine-tuning necessary!
  - Though you still can do it
- Zero-shot
  - "English: Hello. French: "

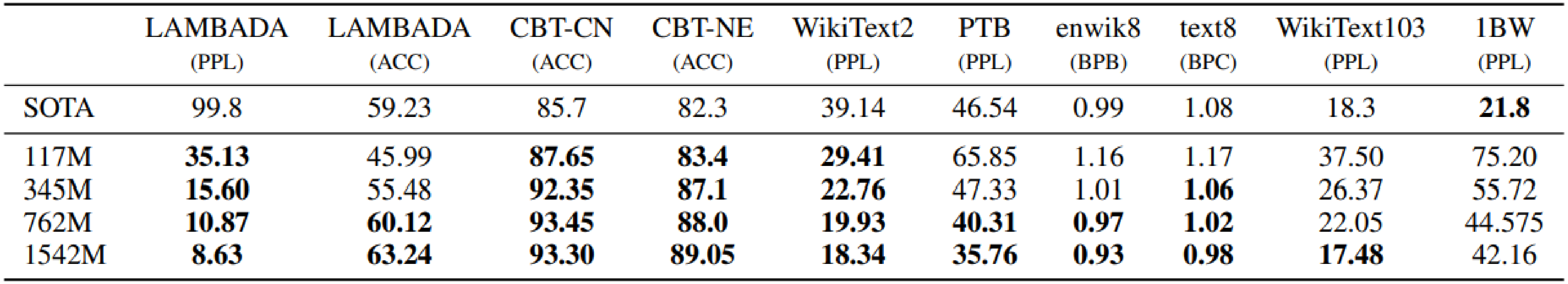
GPT-2

- Purely pre-training
  - Maximum likelihood on next token prediction
- Evaluation metrics for language models?
  - How well is it modelling the text it sees?
  - Perplexity: $\exp\left(-\frac{1}{N}\sum_{i=1}^{N}\log P(x_i|x_{<i})\right)$
    - (discuss range and interpretation)
- LAMBADA

The family was packed and ready for their flight to Japan. Alice had carefully placed her passport inside the blue folder before putting the folder into her carry-on bag. The dog sitter arrived late, delaying their departure. They argued about the route to the airport and almost missed the shuttle. Alice was sweating by the time they reached the terminal curb, worried they were late.

After rushing and arguing, Alice was exhausted. She realized she forgot to pick up the blue folder containing the [_______].

GPT-2 Results (Zero-shot, variety of domains)

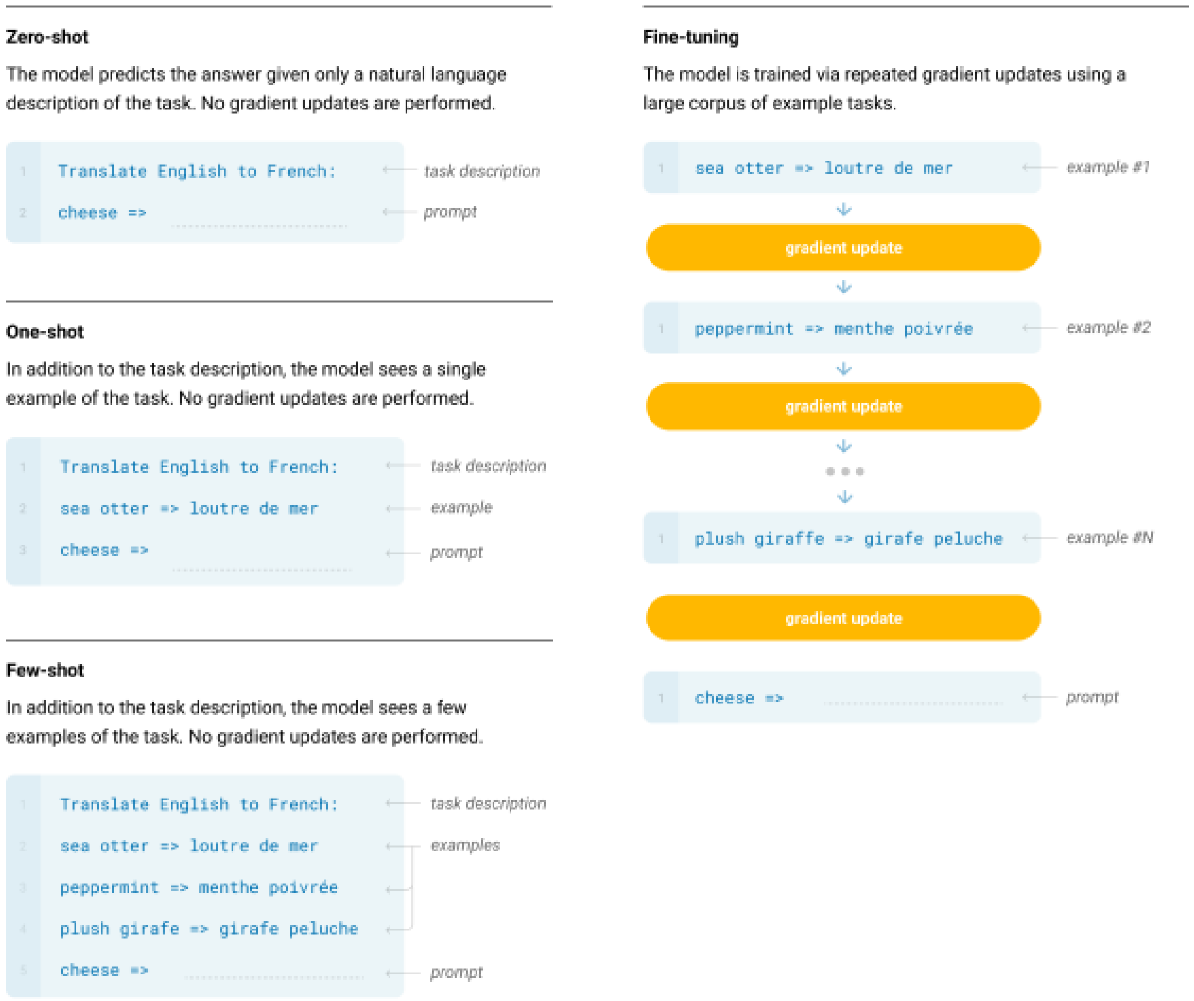


	LAMBADA (PPL)	LAMBADA (ACC)	CBT-CN (ACC)	CBT-NE (ACC)	WikiText2 (PPL)	PTB (PPL)	enwik8 (BPB)	text8 (BPC)	WikiText103 (PPL)	1BW (PPL)
SOTA	99.8	59.23	85.7	82.3	39.14	46.54	0.99	1.08	18.3	**21.8**
117M	**35.13**	45.99	**87.65**	**83.4**	**29.41**	65.85	1.16	1.17	37.50	75.20
345M	**15.60**	55.48	**92.35**	**87.1**	**22.76**	47.33	1.01	**1.06**	26.37	55.72
762M	**10.87**	**60.12**	**93.45**	**88.0**	**19.93**	**40.31**	**0.97**	**1.02**	22.05	44.575
1542M	**8.63**	**63.24**	**93.30**	**89.05**	**18.34**	**35.76**	**0.93**	**0.98**	**17.48**	42.16

GPT-2 -> GPT-3

- Open AI, May 2020
    - Model never released
    - Open? AI
- Same architecture as GPT-1…
    - …But **100x** bigger (175B parameters)
- WebText (40GB) to Filtered CommonCrawl (570GB), **10x** bigger
    - Plus WebText2, Books1, Books2, Wikipedia
- Demonstrates “in-context learning” for new tasks

GPT-3



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:
cheese =>
```

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:
sea otter => loutre de mer
cheese =>
```

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:
sea otter => loutre de mer
peppermint => menthe poivrée
plush girafe => girafe peluche
cheese =>
```

Fine-tuning

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

```
sea otter => loutre de mer
```

```
peppermint => menthe poivrée
```

```
plush giraffe => girafe peluche
```

```
cheese =>
```

GPT-3

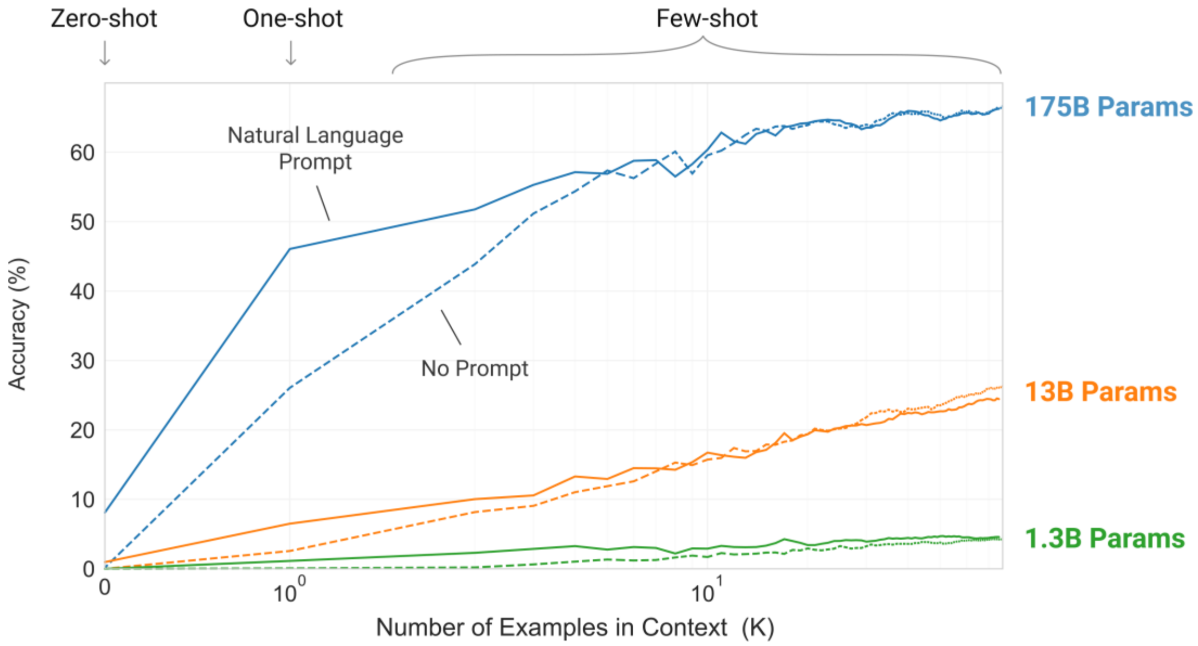
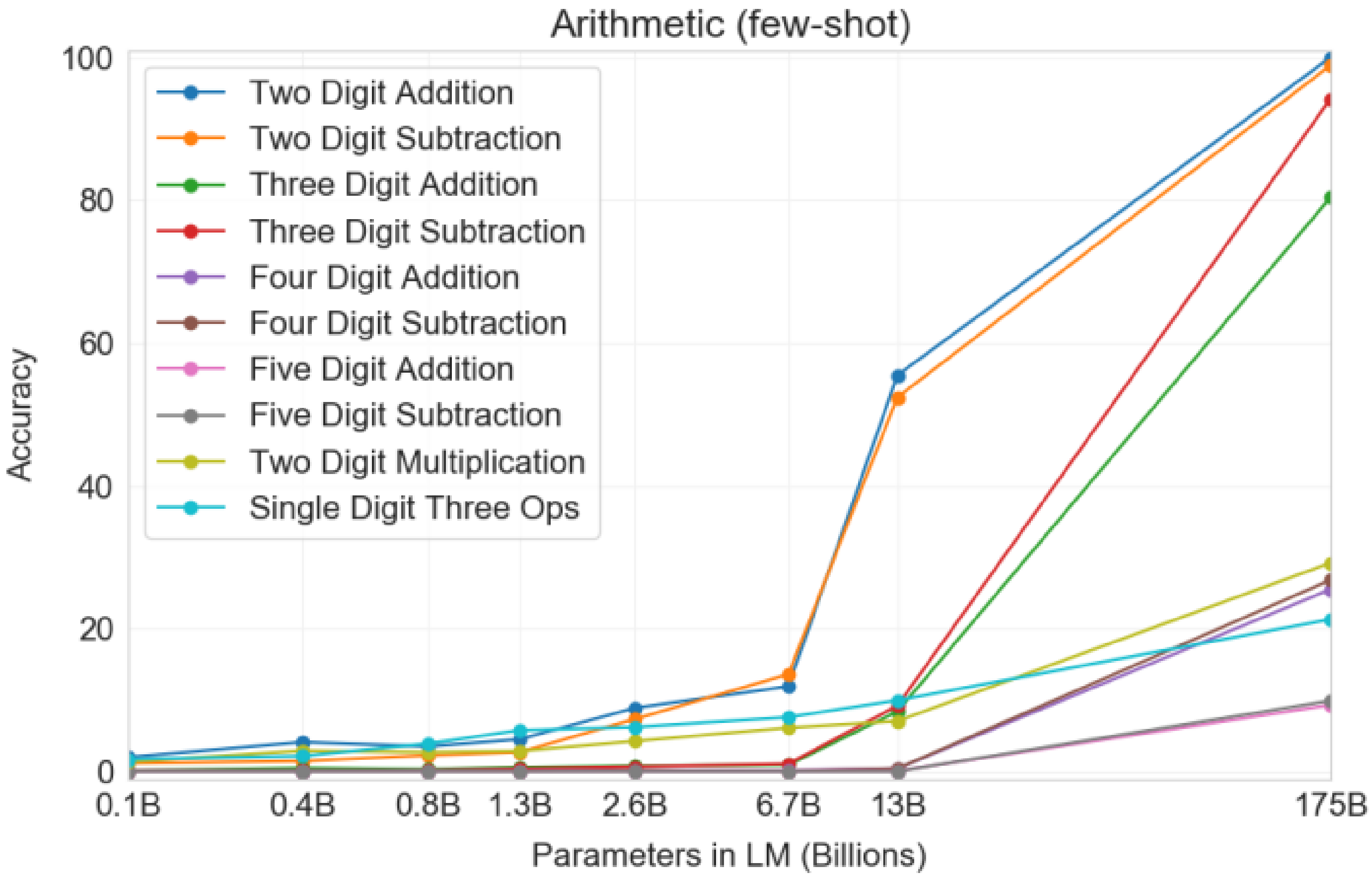
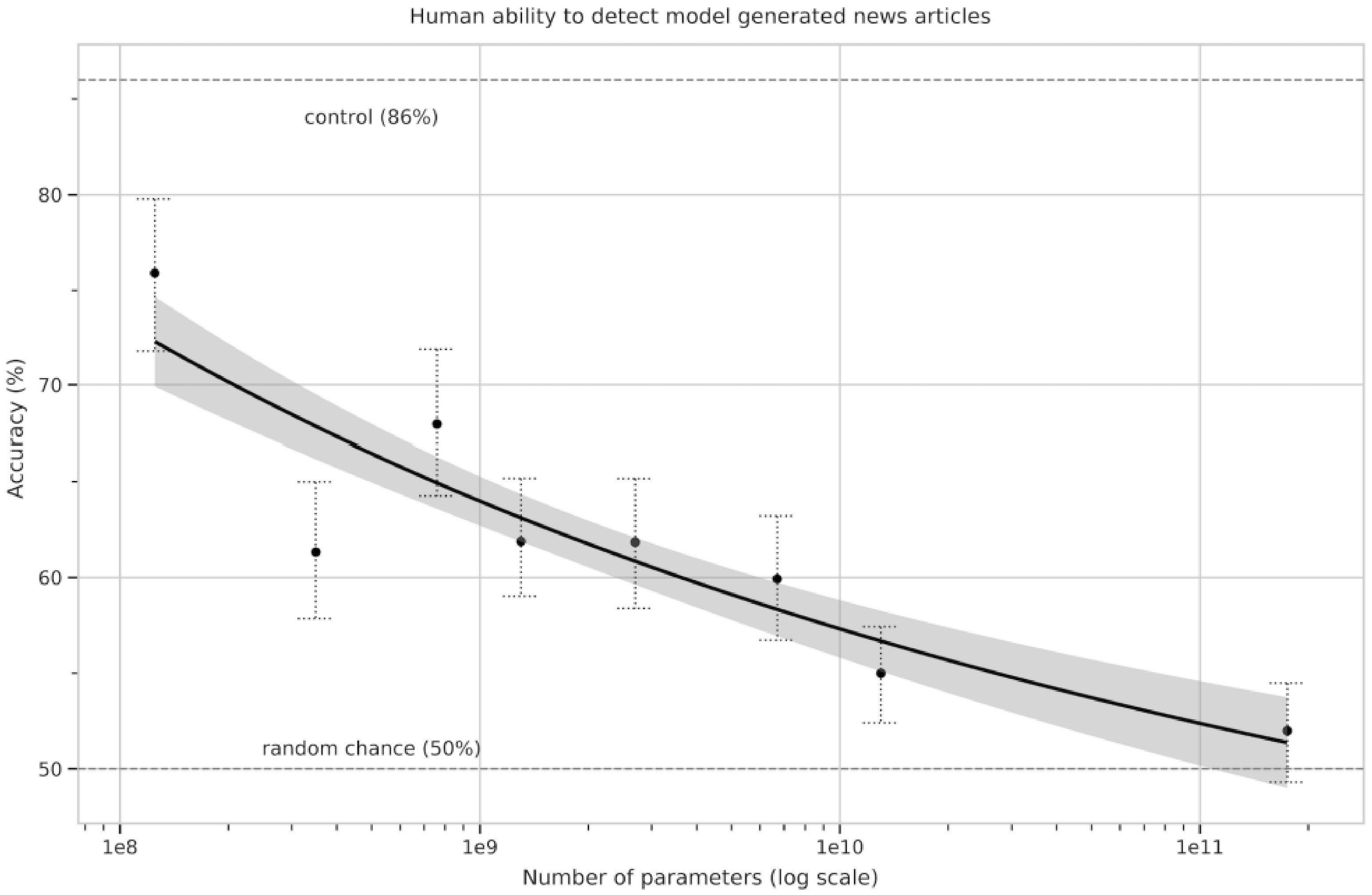


Figure 1.2: Larger models make increasingly efficient use of in-context information. We show in-context learning performance on a simple task requiring the model to remove random symbols from a word, both with and without a natural language task description (see Sec. 3.9.2). The steeper "in-context learning curves" for large models demonstrate improved ability to learn a task from contextual information. We see qualitatively similar behavior across a wide range of tasks.

GPT-3 (k = 50)


Arithmetic (few-shot)
Accuracy
100
80
60
40
20
0
0.1B
0.4B
0.8B
1.3B
2.6B
6.7B
13B
175B
Parameters in LM (Billions)
Two Digit Addition
Two Digit Subtraction
Three Digit Addition
Three Digit Subtraction
Four Digit Addition
Four Digit Subtraction
Five Digit Addition
Five Digit Subtraction
Two Digit Multiplication
Single Digit Three Ops

GPT-3



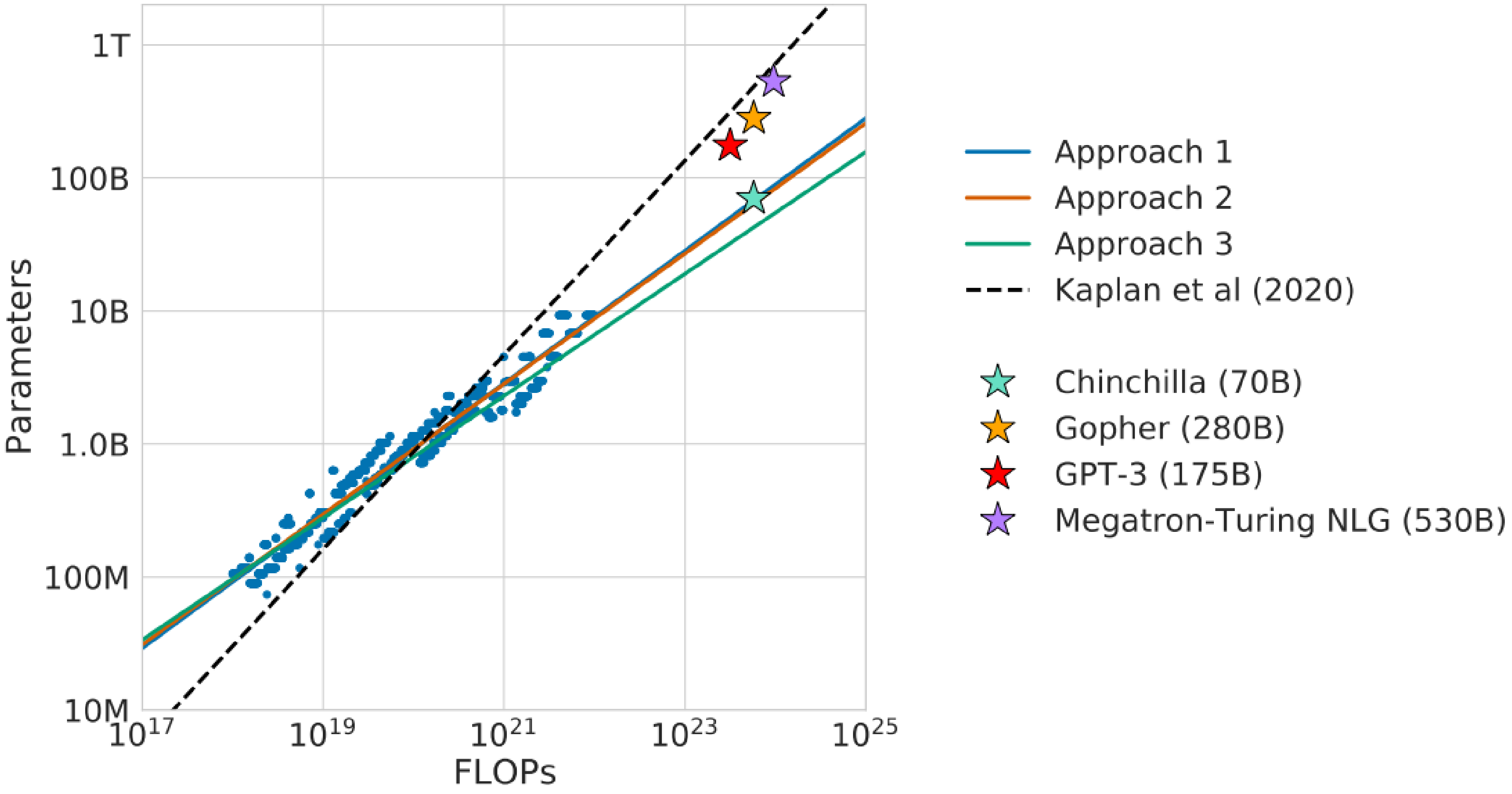
Human ability to detect model generated news articles
control (86%)
random chance (50%)
Accuracy (%)
80
70
60
50
1e8
1e9
1e10
1e11
Number of parameters (log scale)

Scale is all you need? *Scaling Laws*

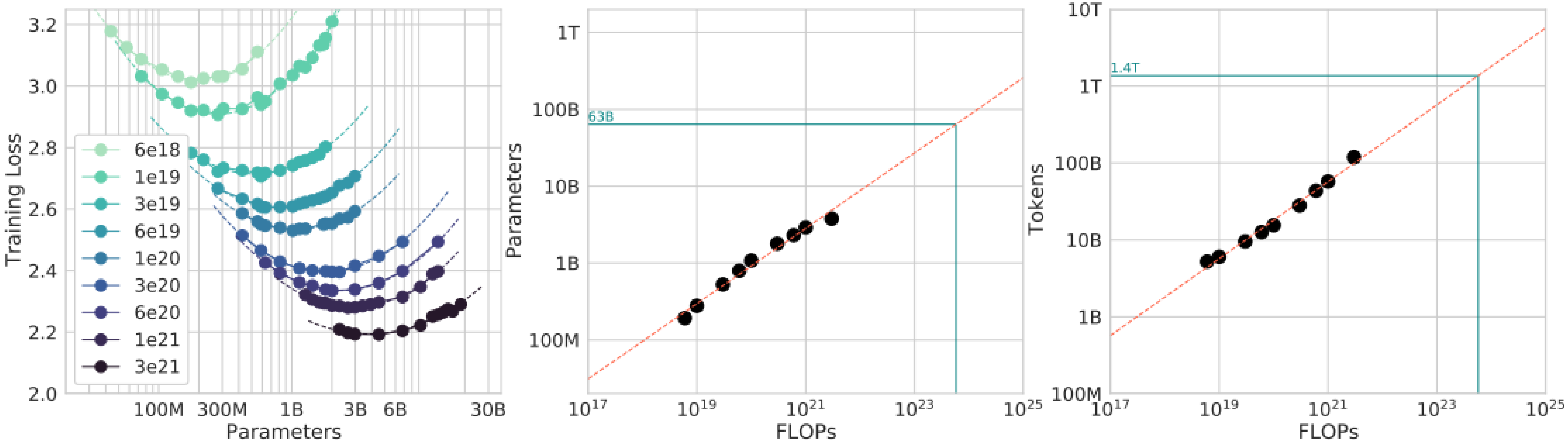
- How can we scale up our models?
    - Parameter Count N
    - Dataset Size D (tokens)
        - Assumption: “high data” regime so no token is used twice
    - Compute C (FLOPs)
- Why not just make all of them as big as possible?
- Claim: $C \approx 6ND$
    - Each token going through each parameter takes 2 FLOPs in forward pass, 4 FLOPs in backward pass
- Suppose we have a fixed compute budget C. How to choose N, D?
- Which (N, D) minimizes the loss?

Chinchilla (Google, March 2022)

- Idea: train many models at low compute budgets C with varying N and D. Extrapolate based on trends.



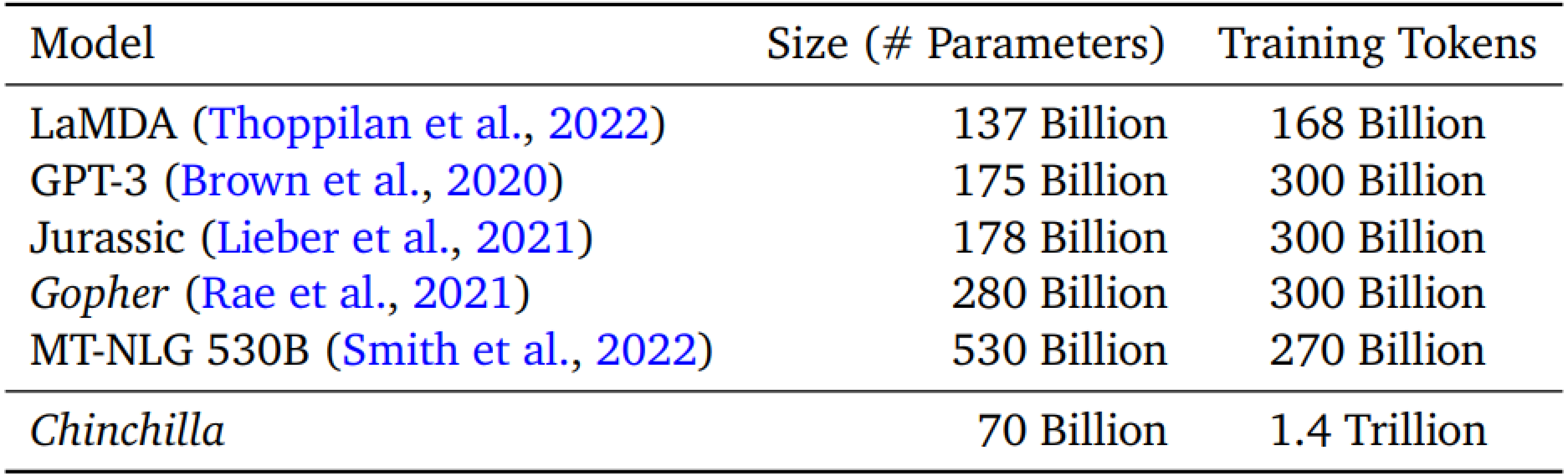
IsoFLOP Curves



To minimize training loss, $N \propto C^{0.5}$, $D \propto C^{0.5}$ -- recall $C \approx 6ND$, so both scale at same rate

Kaplan et al., 2020: $N \propto C^{0.73}$, $D \propto C^{0.27}$, scale parameters faster than data

Takeaway: Train a smaller model on more data



Model	Size (# Parameters)	Training Tokens
LaMDA (Thoppilan et al., 2022)	137 Billion	168 Billion
GPT-3 (Brown et al., 2020)	175 Billion	300 Billion
Jurassic (Lieber et al., 2021)	178 Billion	300 Billion
Gopher (Rae et al., 2021)	280 Billion	300 Billion
MT-NLG 530B (Smith et al., 2022)	530 Billion	270 Billion
Chinchilla	70 Billion	1.4 Trillion

Chinchilla Results

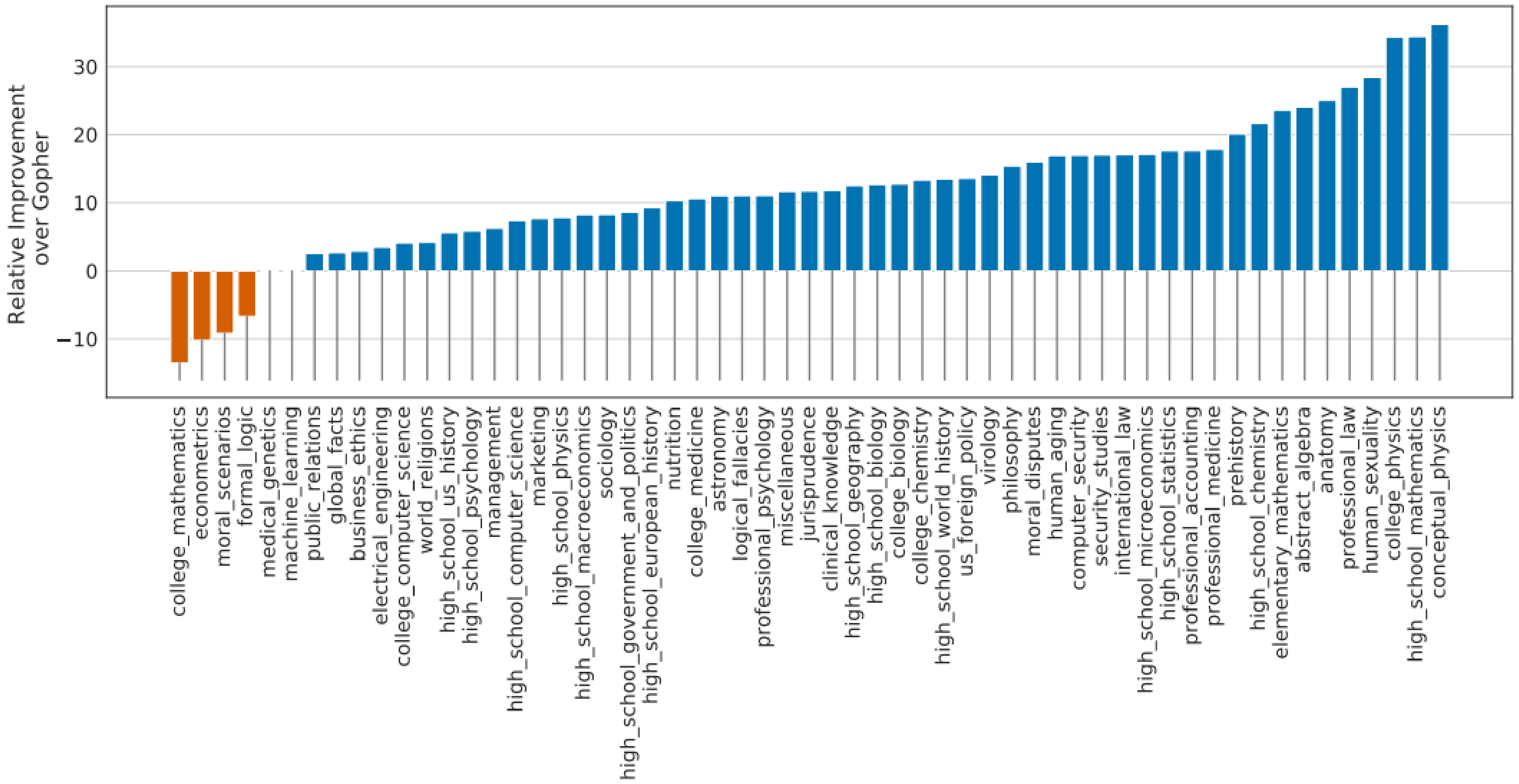


Figure 6 | **MMLU results compared to *Gopher*** We find that *Chinchilla* outperforms *Gopher* by 7.6% on average (see Table 6) in addition to performing better on 51/57 individual tasks, the same on 2/57, and worse on only 4/57 tasks.

Are Scaling Laws the Gospel?

- Why might we deviate from scaling law prescriptions?

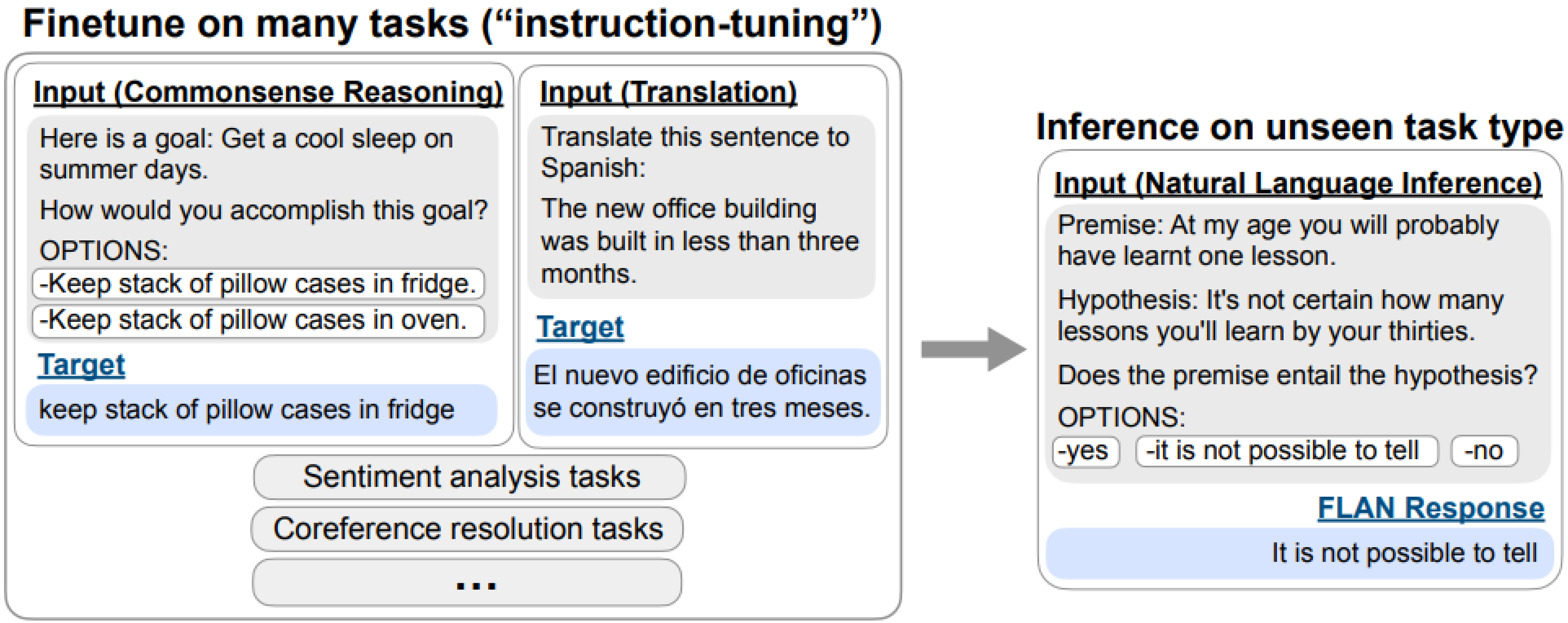
Chain of Thought (Google, January 2022)

- Include in the prompt “Let’s think step by step.”
- It does, and that’s better

Instruction Tuning

- Prompt: “Write a poem about machine learning.”
- Generation: “Write a short story about data science. Write an essay about neural networks.”
- Trained only to continue the text, not answer the question.
- Instead: train the model to answer questions

FLAN (Google, September 2021)



Finetune on many tasks (“instruction-tuning”)

Input (Commonsense Reasoning)

Here is a goal: Get a cool sleep on summer days.

How would you accomplish this goal?

OPTIONS:

-Keep stack of pillow cases in fridge.

-Keep stack of pillow cases in oven.

Target

keep stack of pillow cases in fridge

Input (Translation)

Translate this sentence to Spanish:

The new office building was built in less than three months.

Target

El nuevo edificio de oficinas se construyó en tres meses.

Sentiment analysis tasks

Coreference resolution tasks

...

Inference on unseen task type

Input (Natural Language Inference)

Premise: At my age you will probably have learnt one lesson.

Hypothesis: It's not certain how many lessons you'll learn by your thirties.

Does the premise entail the hypothesis?

OPTIONS:

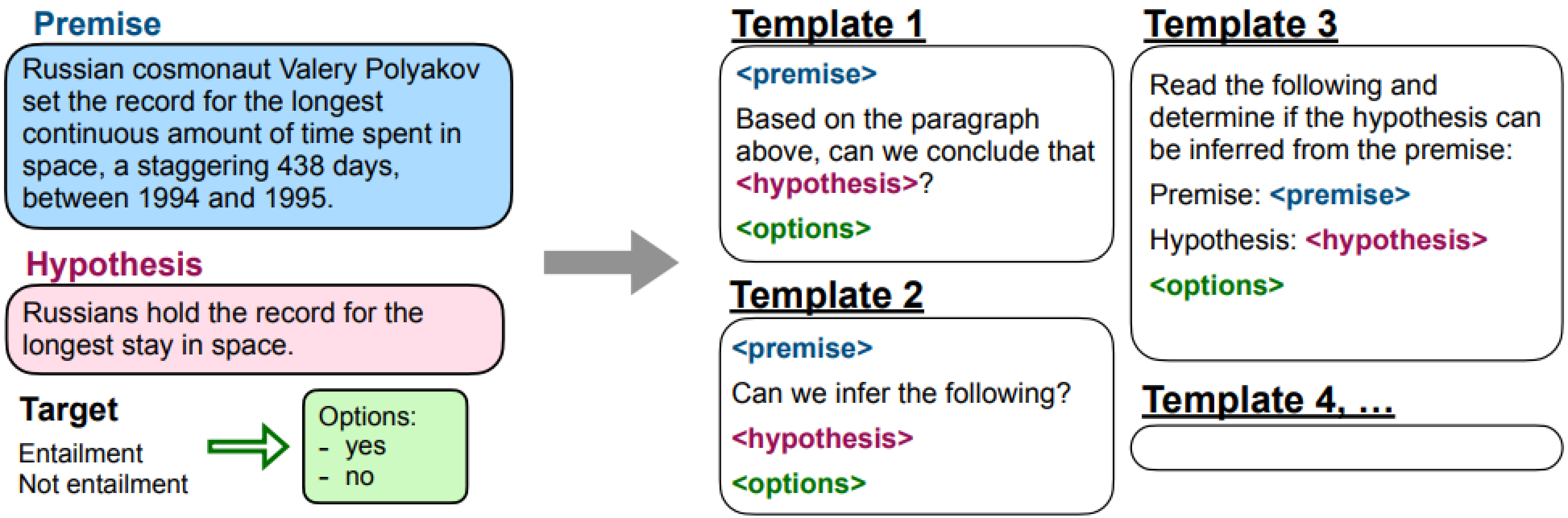
-yes -it is not possible to tell -no

FLAN Response

It is not possible to tell

Maximum likelihood, conditioned on prompt x: $\max \sum_{i=1}^{N} \log P_\theta(y_i | x, y_{<i})$

FLAN



Premise

Russian cosmonaut Valery Polyakov set the record for the longest continuous amount of time spent in space, a staggering 438 days, between 1994 and 1995.

Hypothesis

Russians hold the record for the longest stay in space.

Target

Entailment
Not entailment

Options:
- yes
- no

Template 1

<premise>

Based on the paragraph above, can we conclude that <hypothesis>?

<options>

Template 2

<premise>

Can we infer the following?

<hypothesis>

<options>

Template 3

Read the following and determine if the hypothesis can be inferred from the premise:

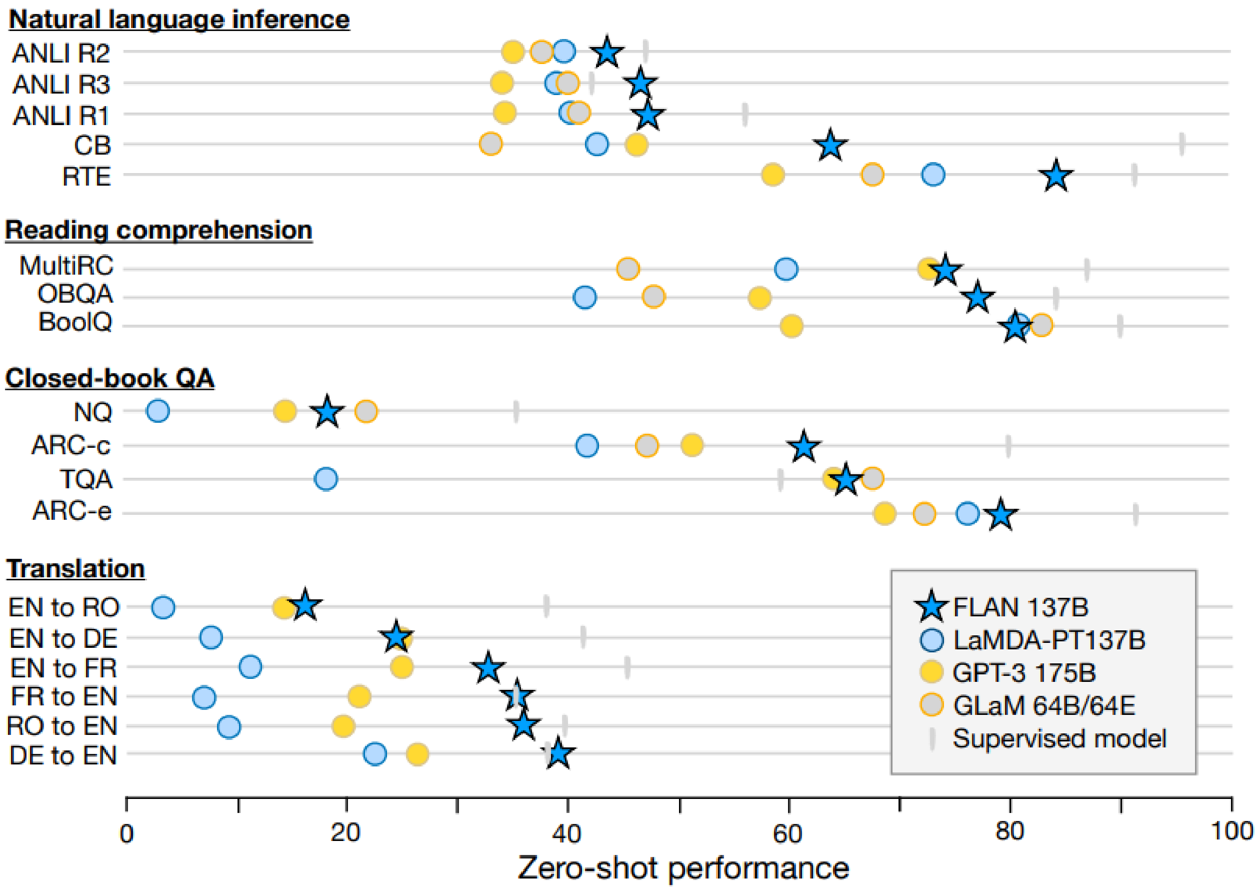
Premise: <premise>

Hypothesis: <hypothesis>

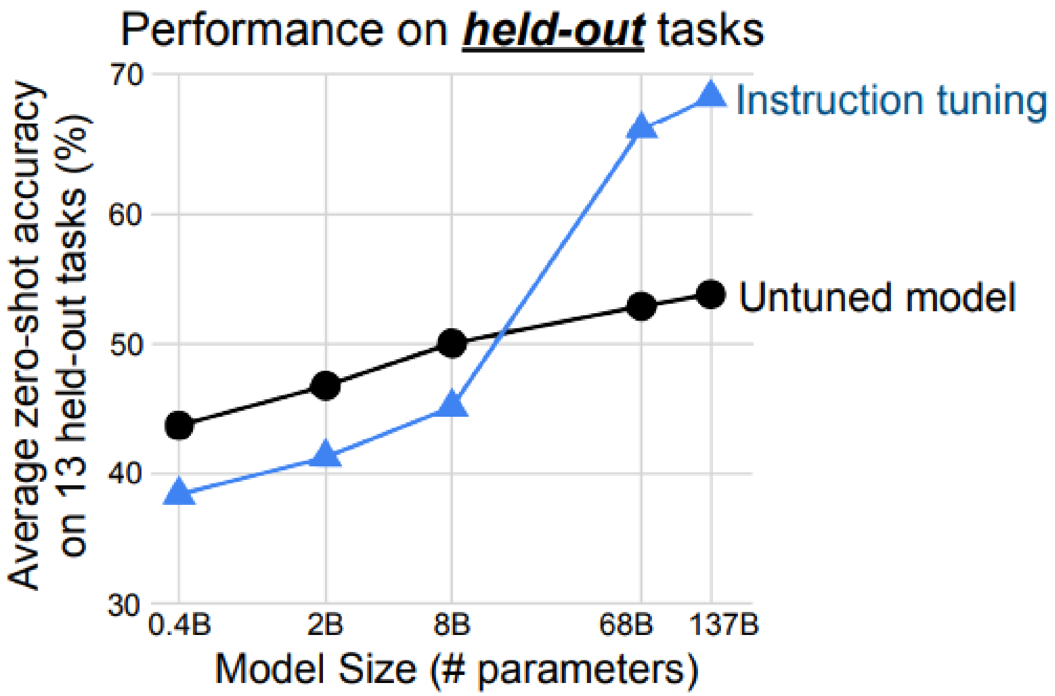
<options>

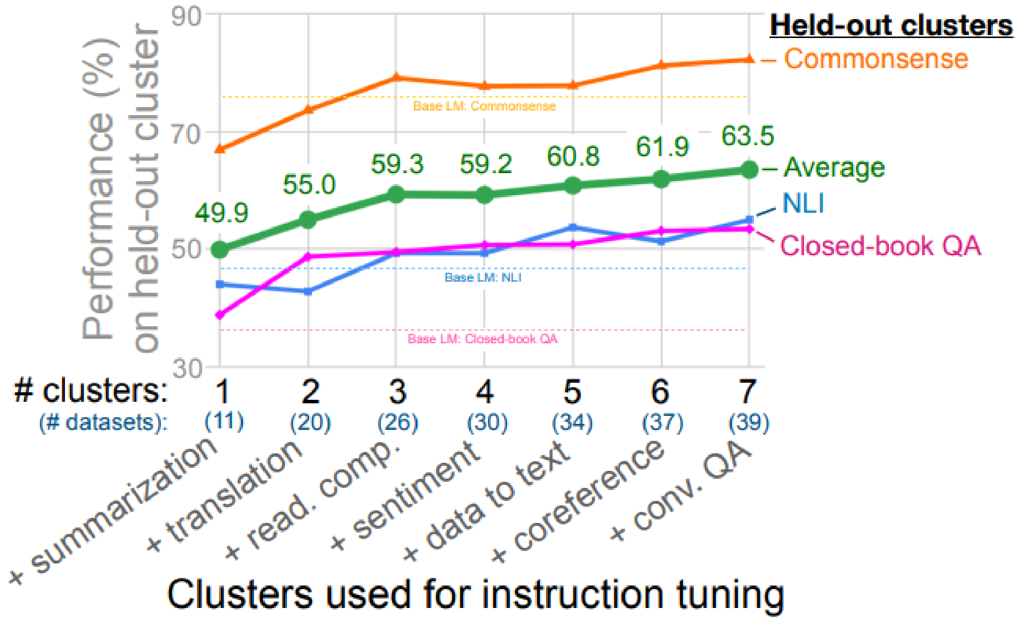
Template 4, …

FLAN Results



Scale in FLAN

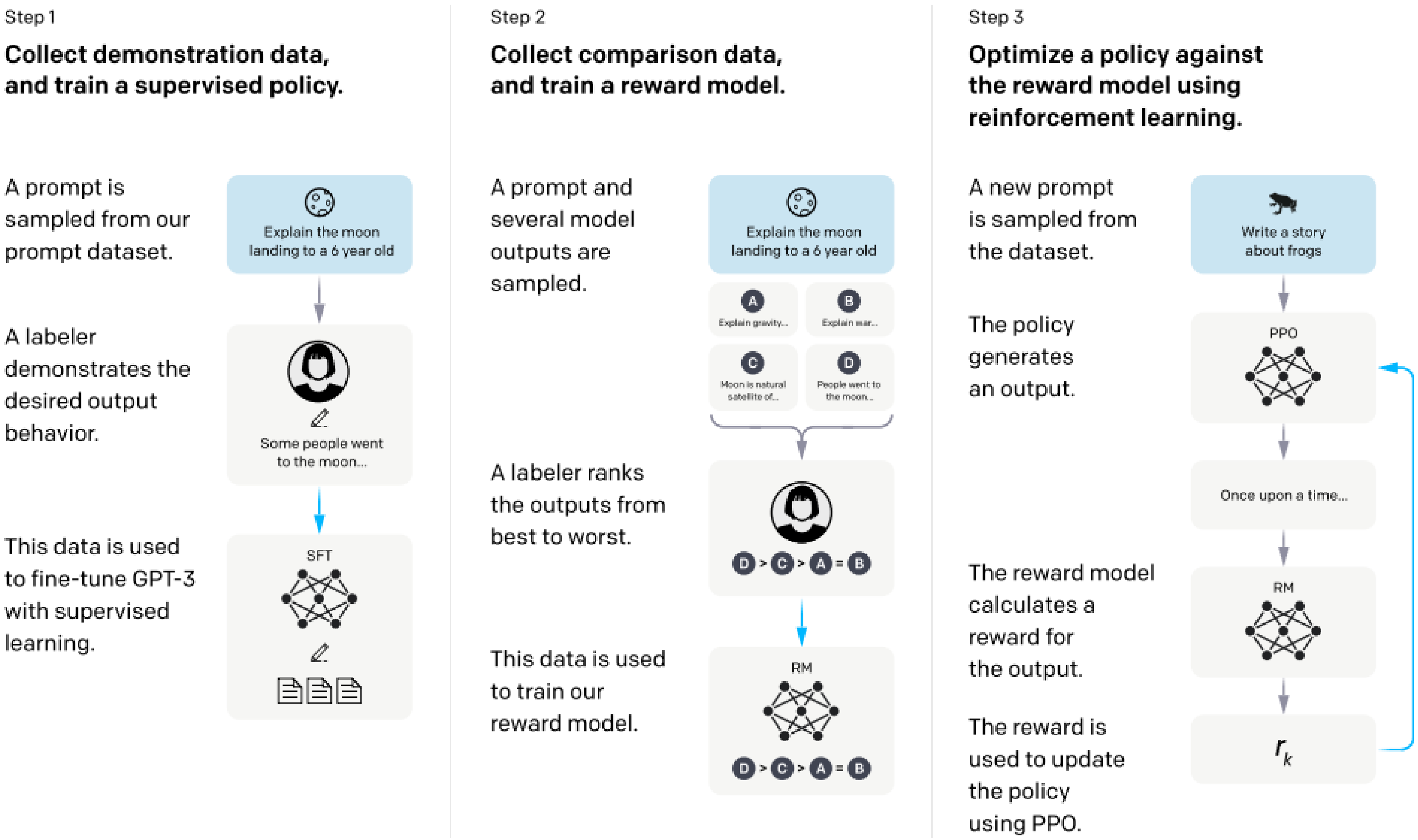




Reinforcement Learning with Human Feedback (RLHF)

- Previous problem: We don’t want language models to continue a phrase, we want them to respond to instructions
- Problem now: We don’t want language models to respond to instructions, we want them to respond to respond to instructions *in a way that humans want*
- Prompt: My computer is running slow, what should I do?
- Answer from the Internet: Delete System32
  - Troll/joke answer. But LLMs may learn this as a good response…
- Other goals: Safety (“Build a bomb”), legal (“Don’t output copyrighted material”)
- Train with humans in the loop

InstructGPT (Open AI, March 2022)



Step 1

Collect demonstration data, and train a supervised policy.

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.

Step 2

Collect comparison data, and train a reward model.

A prompt and several model outputs are sampled.

A labeler ranks the outputs from best to worst.

This data is used to train our reward model.

Step 3

Optimize a policy against the reward model using reinforcement learning.

A new prompt is sampled from the dataset.

The policy generates an output.

The reward model calculates a reward for the output.

The reward is used to update the policy using PPO.

Explain the moon landing to a 6 year old
Some people went to the moon...
SFT
Explain the moon landing to a 6 year old
A
Explain gravity...
B
Explain war...
C
Moon is natural satellite of...
D
People went to the moon...
D > C > A = B
RM
D > C > A = B
Write a story about frogs
PPO
Once upon a time...
RM
r_k

Step 2: Collect comparison data, train a reward model

- Model generates K possible completions for a prompt
- Human ranks these K completions from best to worst
  - Results in $\binom{K}{2}$ pairwise comparisons between outcomes
- Train a model $r_\theta(x, y)$, mapping prompt x and response y to score

$$L(\theta) = -\frac{1}{\binom{K}{2}} E_{(x,y_w,y_l)\sim D}[\log(\sigma(r_\theta(x, y_w) - r_\theta(x, y_l))]$$

- y_w and y_l are the winning and losing response for each pair

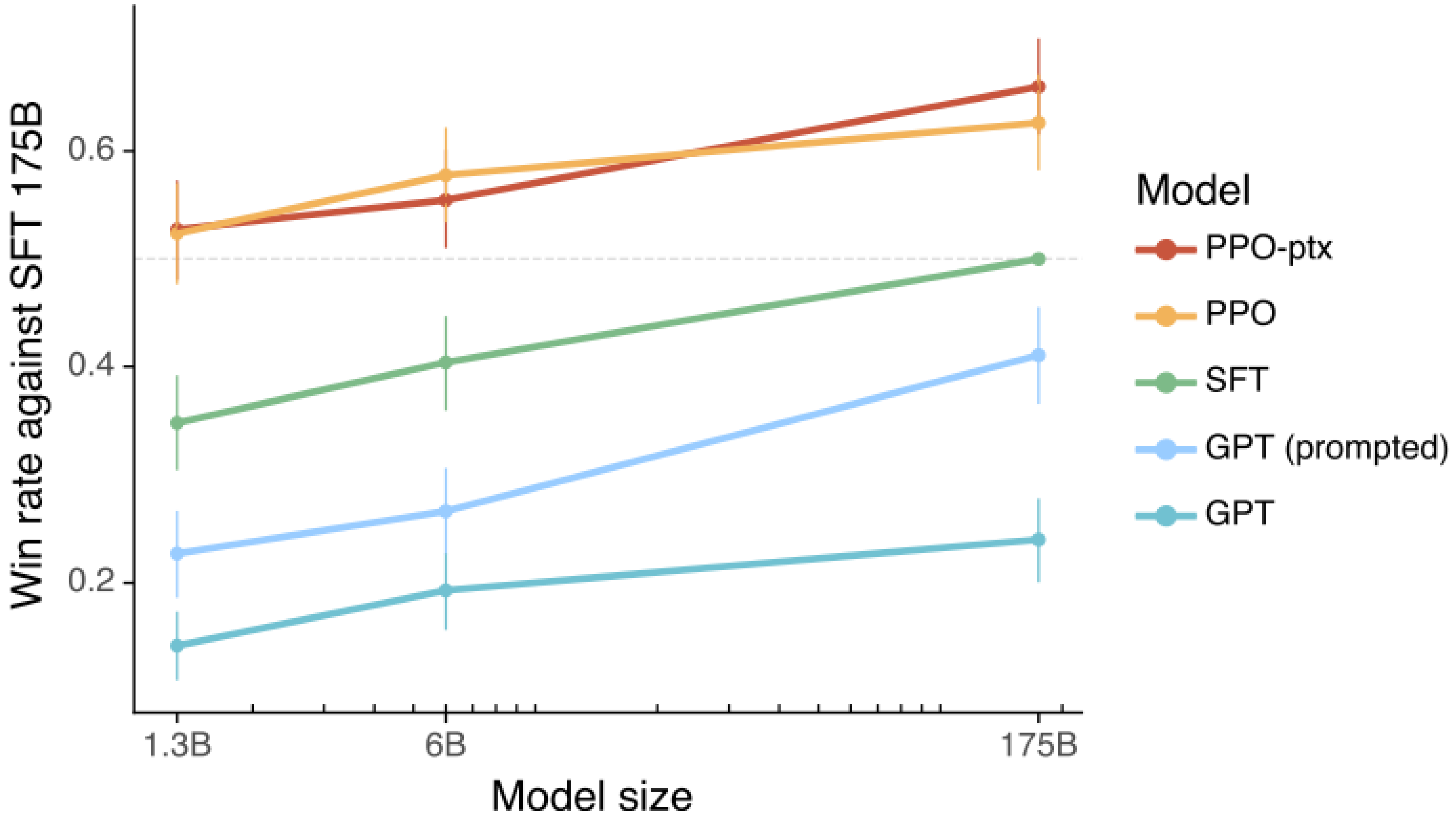
Step 3: Reinforcement Learning

- Proximal Policy Optimization (PPO)

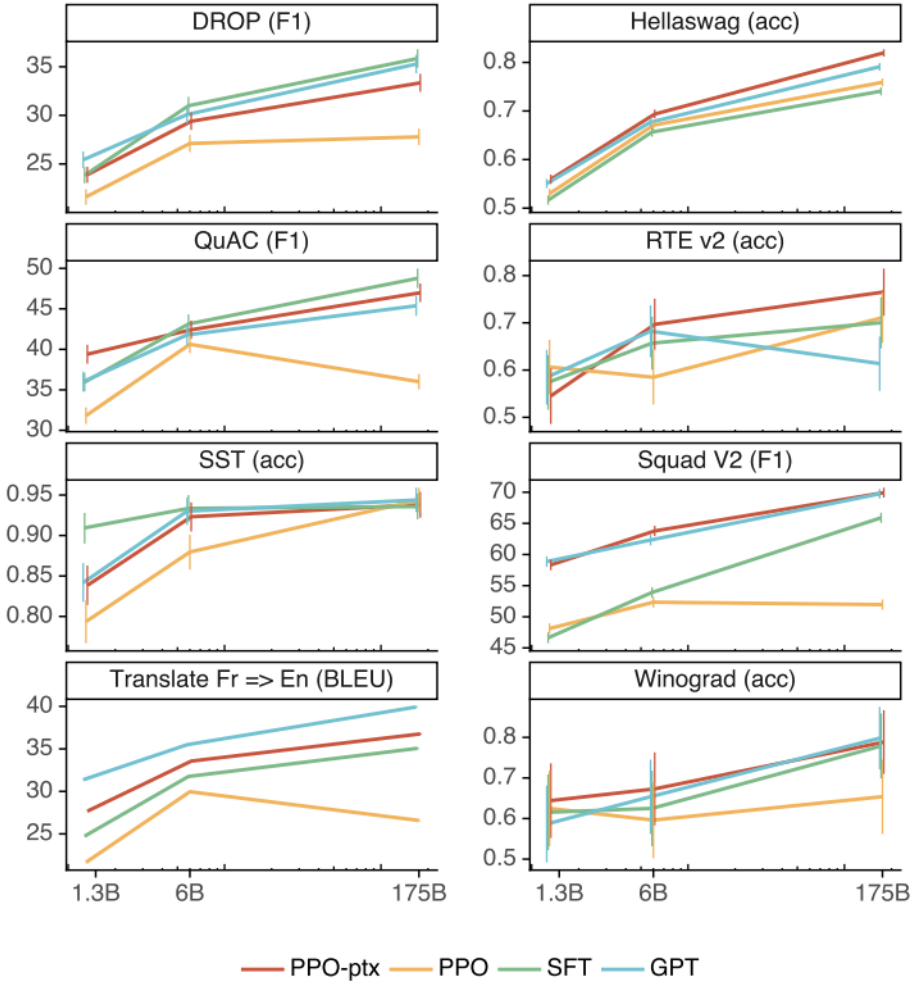
$$\text{objective}(\phi)$$
$$= E_{(x,y)\sim D_{\pi_{\phi}^{RL}}}\left[r_{\theta}(x,y) - \beta \log\left(\frac{\pi_{\phi}^{RL}(y|x)}{\pi_{\phi}^{SFT}(y|x)}\right)\right]$$
$$+ \gamma E_{x\sim D_{pretrain}}\left[\log\left(\pi_{\phi}^{RL}(x)\right)\right]$$

- Maximize reward of the model, don’t stray too far from the original model
- “Alignment tax”

Users prefer model with RLHF



Alignment Tax



ChatGPT (Open AI, November 2022)

- Essentially GPT-3.5 with RLHF
  - InstructGPT GPT-3 with RLHF
- This is when LLMs became mainstream
- …and roughly when they stopped telling us what they do.