Generative Artificial Intelligence

GPT-3, InstructGPT, and RLHF

Outline

- (Recap) From GPT-1 to GPT-3
- Sparse Transformer
- InstructGPT (GPT-3.5)
- Reinforcement Learning with Human Feedback
- Llama and Llama-2 (Meta AI)

Generative Pre-Training (GPT-1)

- GPT: Generative Pre-Training (developed by OpenAI)
- GPT is a language model trained from language modeling.
- Architecture: Decoder part of Transformers (12-layer; 117M)

Radford, Alec, et al. "***Improving language understanding by generative pre-training***." (2018).

Detailed Comparison for the Architecture

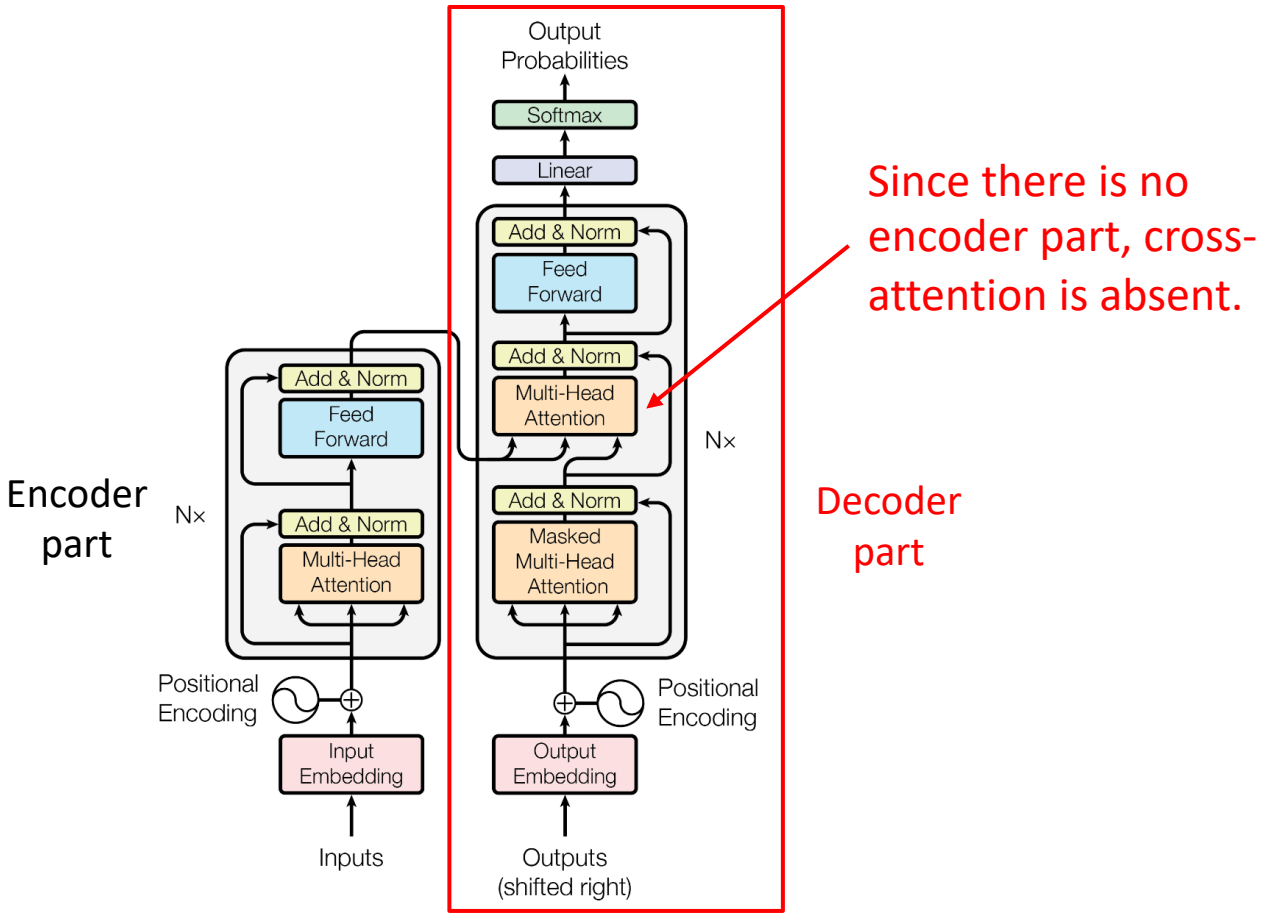


Figure from the Transformers paper (2017).

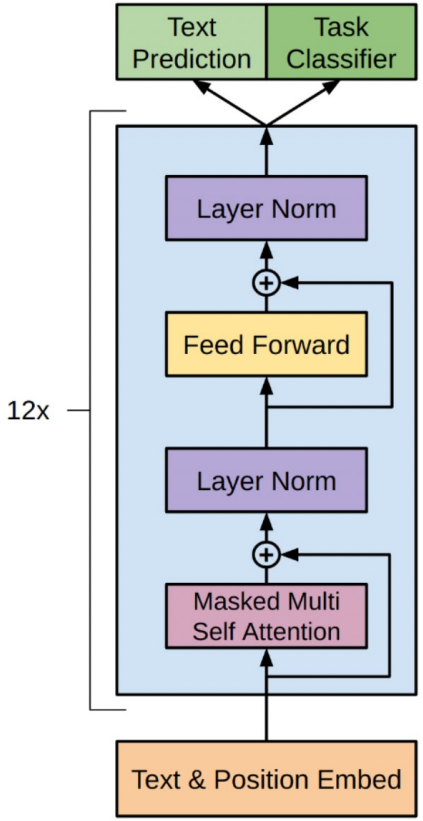


Figure from the GPT-1 paper (2018).

From GPT-1 to GPT-2

- Layer normalization is moved to the input of each sub-block.
  - Pre-activation
- An additional layer normalization was added after the final self-attention block.
- The weights of residual layers at initialization were scaled by $1/\sqrt{N}$, where N is the number of residual layers.
  - For stabilizing training.
- Increase number of layers
  - GPT-2-medium (24-layer; 345M), GPT-2-large (36-layer; 762M), GPT-2-xl (48-layer; 1.5B)
- Zero-shot learning idea

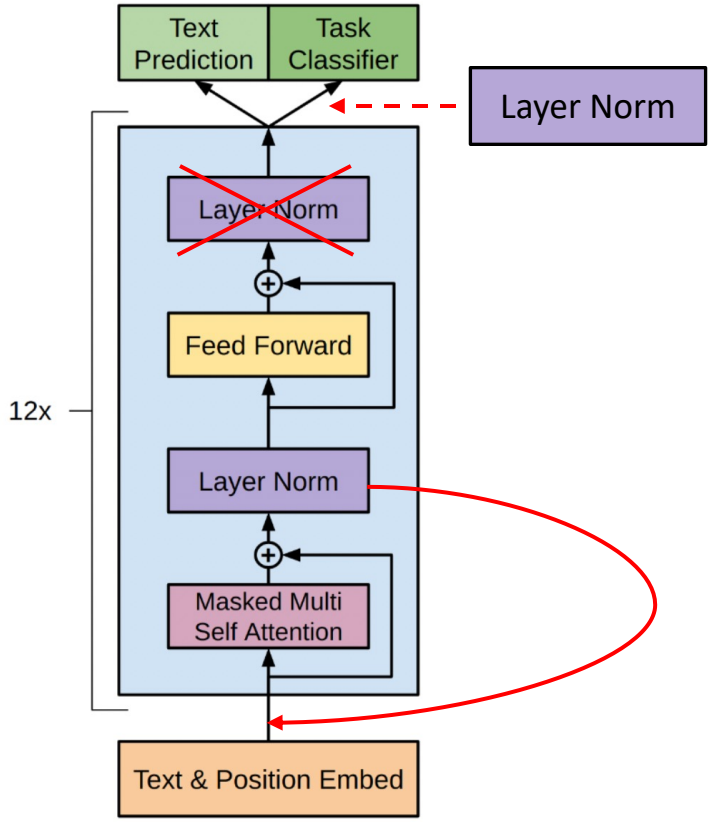
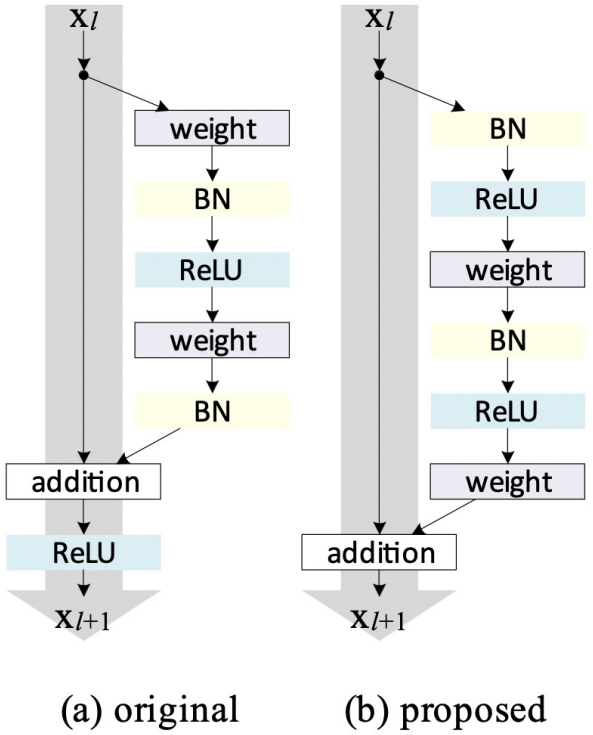


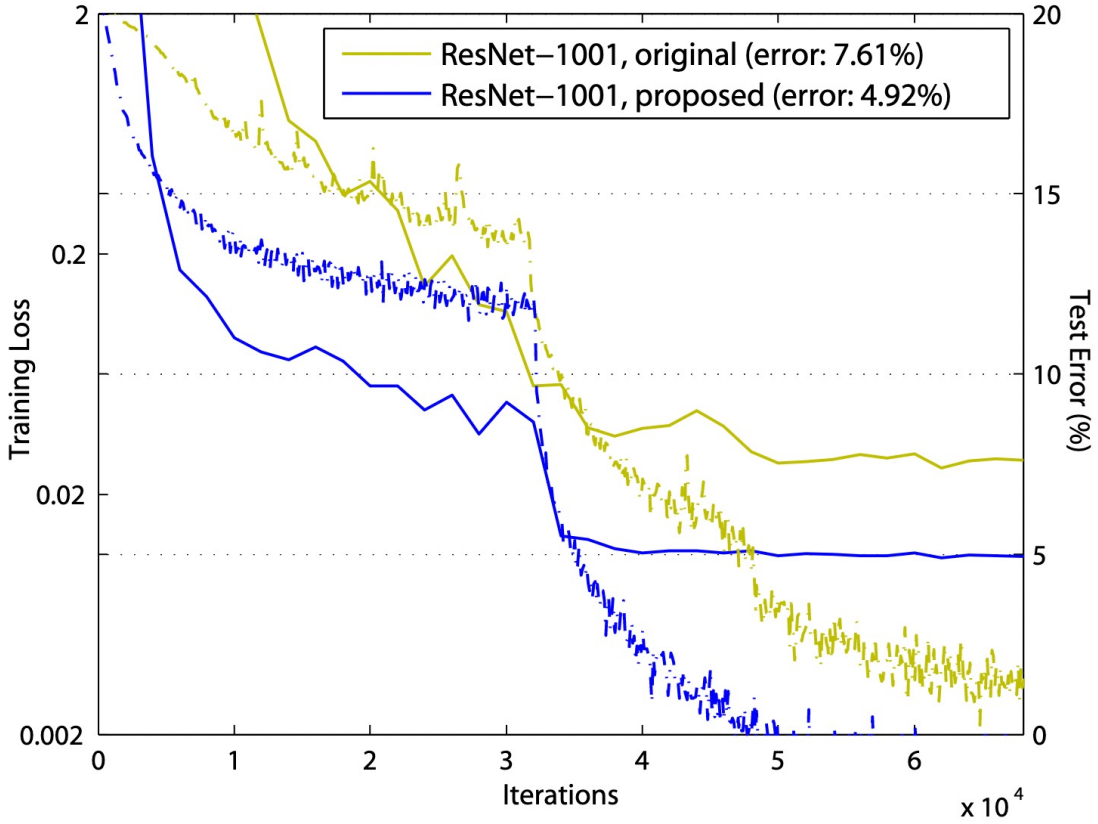
Figure from the GPT-1 paper (2018).

Radford, Alec, et al. "***Language models are unsupervised multitask learners***." *OpenAI blog* 1.8 (2019): 9.

Pre-activation in ResNet



(a) original (b) proposed



He, Kaiming, et al. "Identity mappings in deep residual networks." Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14. Springer International Publishing, 2016.

From GPT-2 to GPT-3

- Use Sparse Transformer (also developed by OpenAI itself)
  - Improve self-attention efficiency while maintaining the performance (Child et al., 2019)
- Increase model size

GPT-2-like sizes

Common GPT-3 size

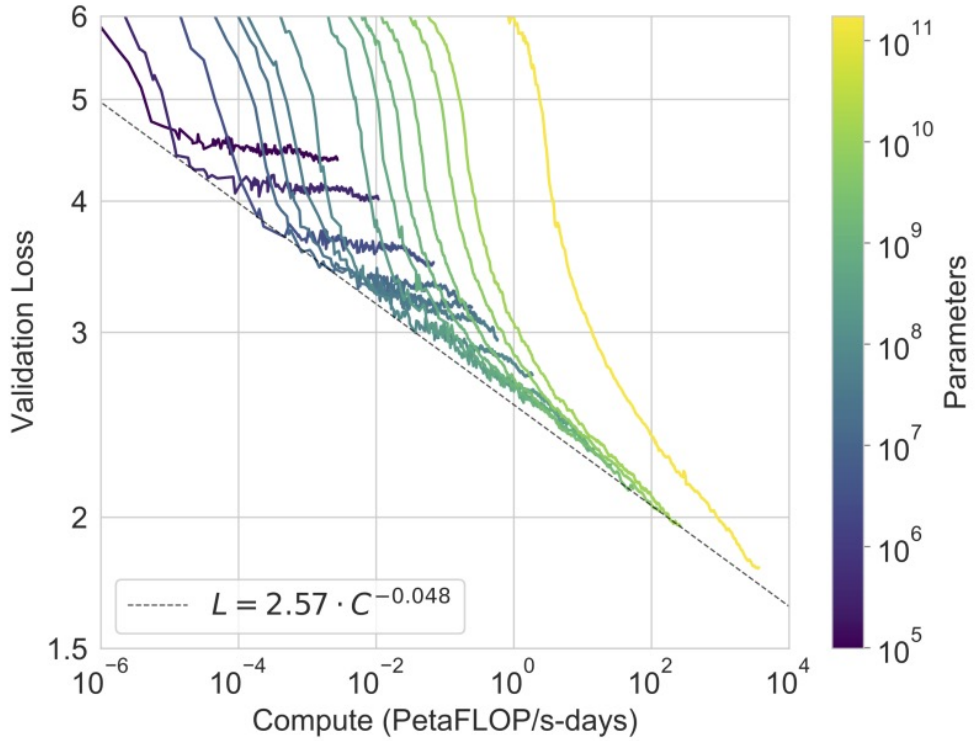
Model Name	n_{params}	n_{layers}	d_{model}	n_{heads}	d_{head}	Batch Size	Learning Rate
GPT-3 Small	125M	12	768	12	64	0.5M	6.0×10^{-4}
GPT-3 Medium	350M	24	1024	16	64	0.5M	3.0×10^{-4}
GPT-3 Large	760M	24	1536	16	96	0.5M	2.5×10^{-4}
GPT-3 XL	1.3B	24	2048	24	128	1M	2.0×10^{-4}
GPT-3 2.7B	2.7B	32	2560	32	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.

Child, Rewon, et al. "Generating long sequences with sparse transformers." *arXiv preprint arXiv:1904.10509* (2019).

Brown, Tom, et al. "***Language models are few-shot learners***." Advances in neural information processing systems 33 (2020): 1877-1901.

GPT3 shows us



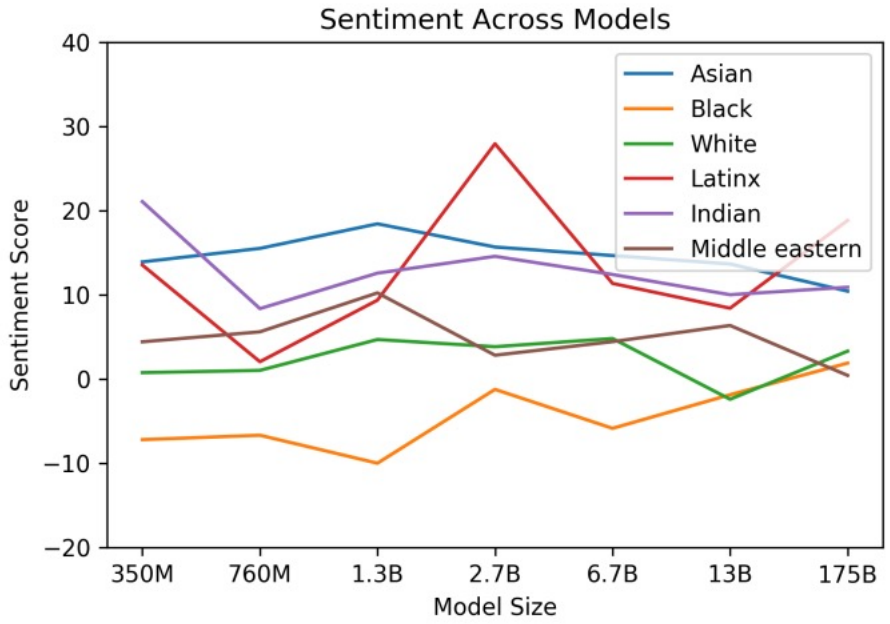


Figure 6.1: Racial Sentiment Across Models

Clean data is key!

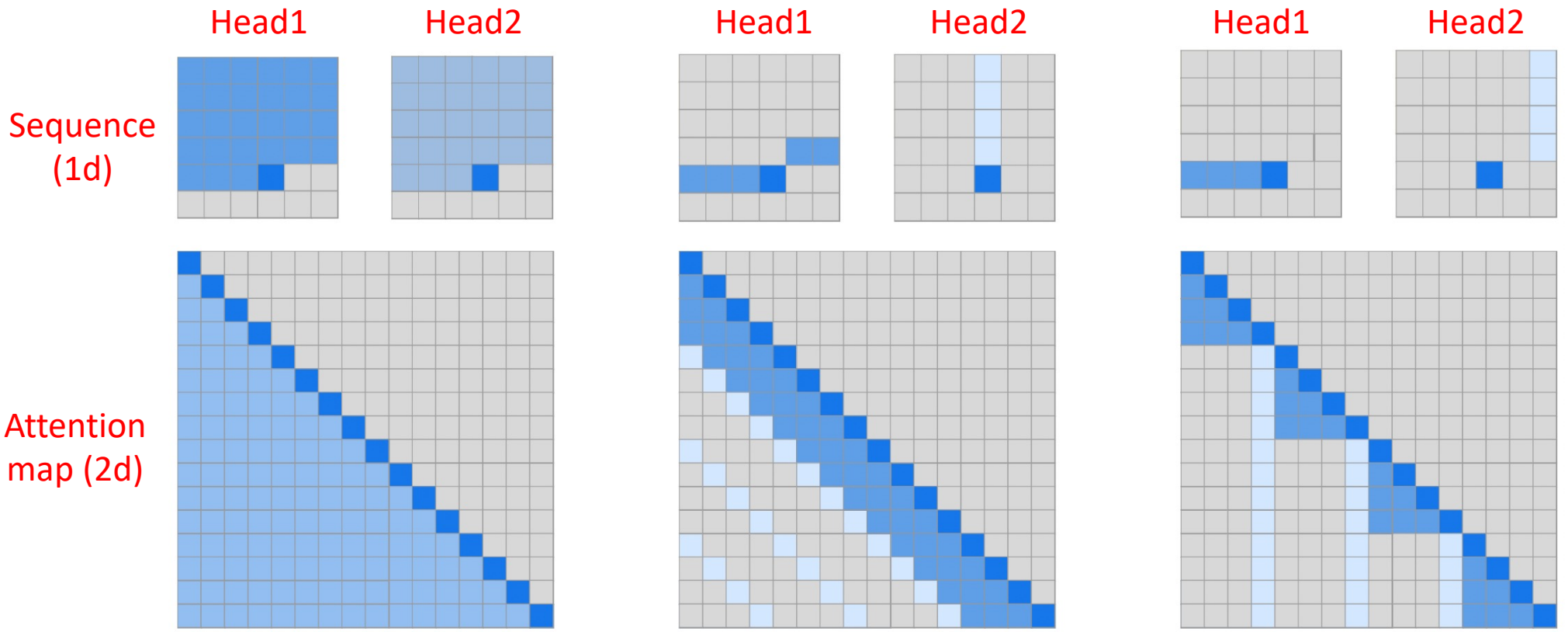
Why Sparse Transformer?

- Even computing a single attention matrix (self-attention), however, can become computationally expensive ($O(n^2)$) for very large inputs, especially long sequences.
- The OpenAI team found that most layers had sparse attention patterns across most data points, suggesting that some form of sparsity could be introduced without significantly affecting performance.

https://openai.com/research/sparse-transformer
Child, Rewon, et al. "Generating long sequences with sparse transformers." *arXiv preprint arXiv:1904.10509* (2019).

Sparse Transformer

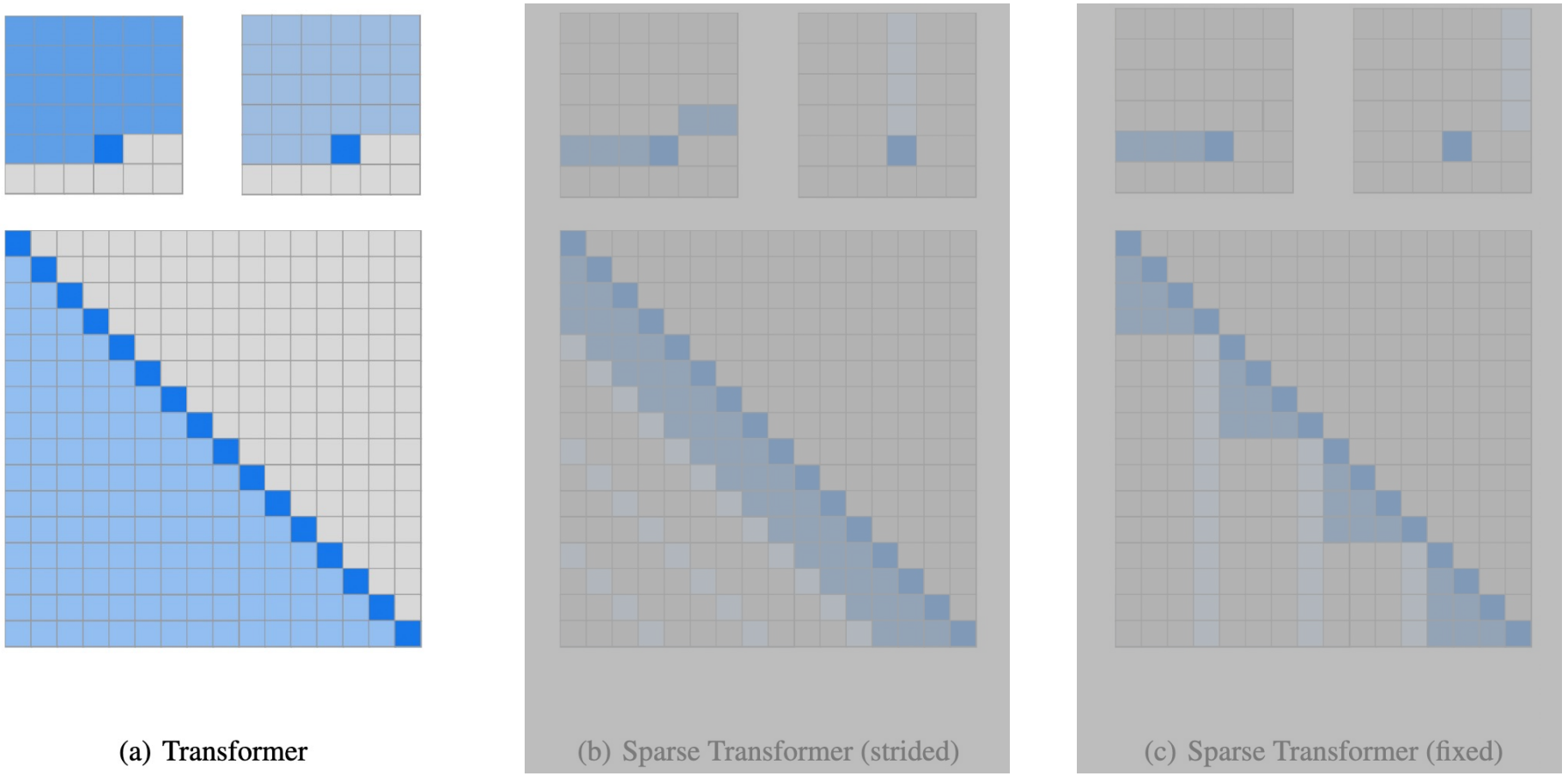
Using 2 heads during multi-head self-attention as examples



Child, Rewon, et al. "Generating long sequences with sparse transformers." *arXiv preprint arXiv:1904.10509* (2019).

Sparse Transformer

Standard casual masking: a model can only attend the tokens before the current token.

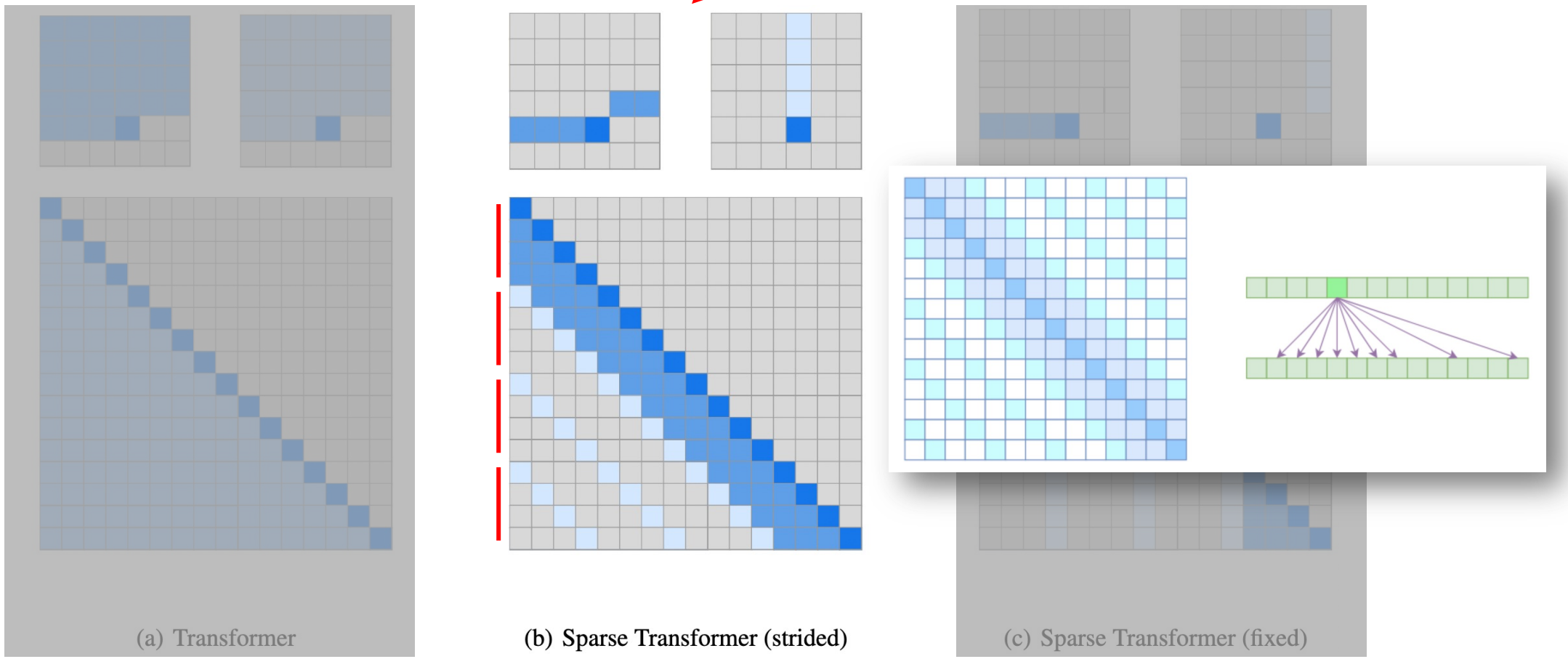


(a) Transformer (b) Sparse Transformer (strided) (c) Sparse Transformer (fixed)

Child, Rewon, et al. "Generating long sequences with sparse transformers." *arXiv preprint arXiv:1904.10509* (2019).

Sparse Transformer

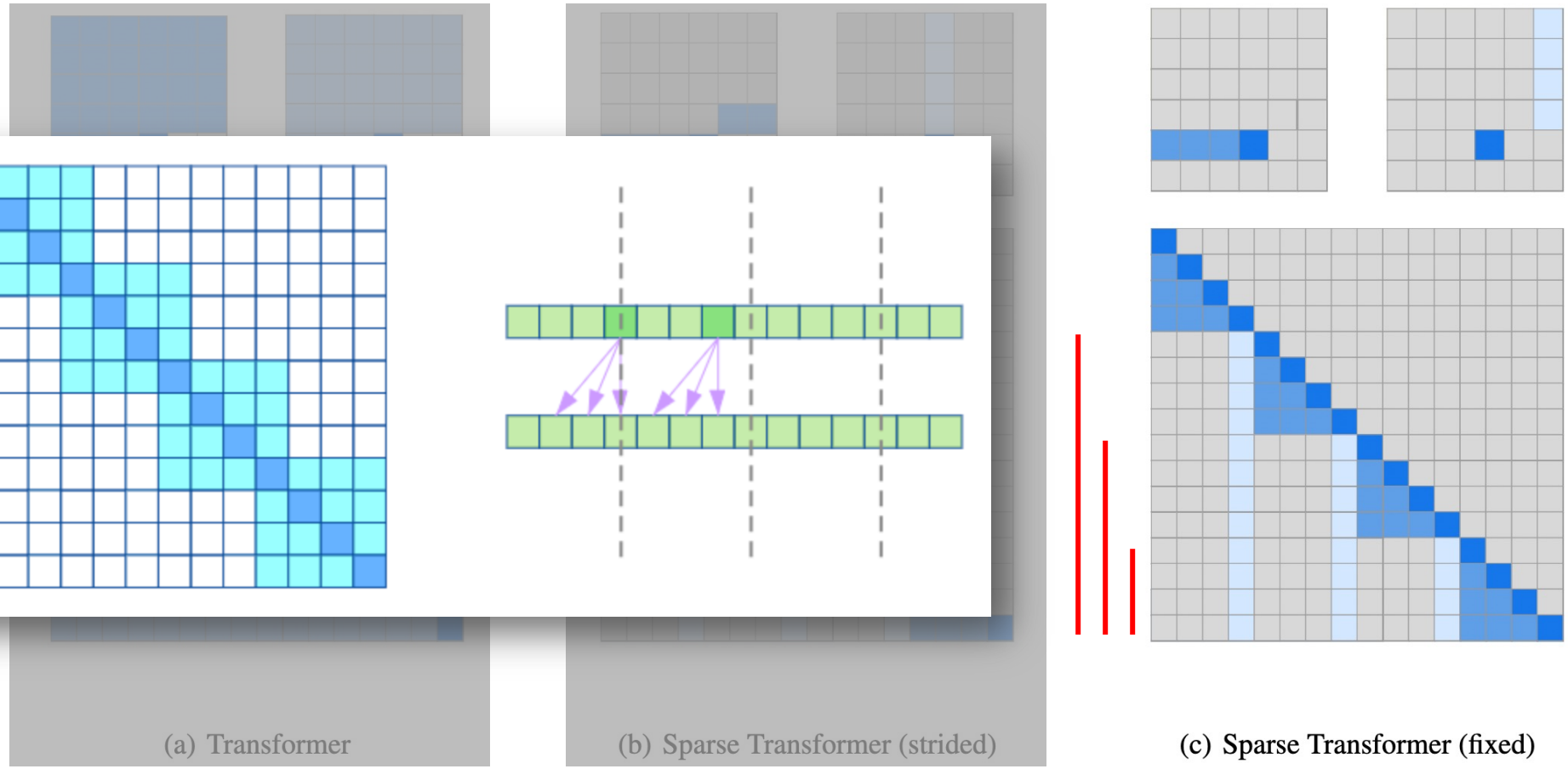
- A stride is set to limit the attention range.
- Useful if the data naturally has a structure that aligns with the stride, like images or some types of music.



Child, Rewon, et al. "Generating long sequences with sparse transformers." *arXiv preprint arXiv:1904.10509* (2019).

Sparse Transformer

- Specific cells summarize previous locations and propagate that information to all future cells.
- Useful for text data.



Child, Rewon, et al. "Generating long sequences with sparse transformers." *arXiv preprint arXiv:1904.10509* (2019).

Sparse Transformer – Contribution

- Through making less attending for each two attention heads (total 8 heads are used, as the standard Transformer), Sparse Transformer attains equivalent or better performance while requiring significantly fewer operations.

	Model	Bits per byte	Time/Iter
(Text data)	**Enwik8 (12,288 context)**		
	Dense Attention	1.00	1.31
	Sparse Transformer (Fixed)	**0.99**	0.55
	Sparse Transformer (Strided)	1.13	0.35
(Image data)	**CIFAR-10 (3,072 context)**		
	Dense Attention	2.82	0.54
	Sparse Transformer (Fixed)	2.85	0.47
	Sparse Transformer (Strided)	**2.80**	0.38

Child, Rewon, et al. "Generating long sequences with sparse transformers." *arXiv preprint arXiv:1904.10509* (2019).

Sparse Transformer (Formula)

- Sparse Transformer (strided)

Head(1) $A_i^{(1)} = \{t, t+1, \ldots, i\}$ for $t = \max(0, i-1)$

Head(2) $A_i^{(2)} = \{j : (i-j) \bmod l = 0\}$

- i: current sequence index
- j: sequence index that can be attended
- l: stride

- Sparse Transformer (fixed)

Head(1) $A_i^{(1)} = \{j : (\lfloor j/l \rfloor = \lfloor i/l \rfloor)\}$, where the brackets denote the floor operation.

Head(2) $A_i^{(2)} = \{j : j \bmod l \in \{t, t+1, \ldots, l\}\}$, where $t = l - c$ and c is a hyperparameter.

Child, Rewon, et al. "Generating long sequences with sparse transformers." *arXiv preprint arXiv:1904.10509* (2019).

GPT-3: Language Models are Few-Shot Learners

- The three settings explored for in-context learning in the GPT-3 paper:

Zero-shot

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

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

One-shot

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

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

Few-shot

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

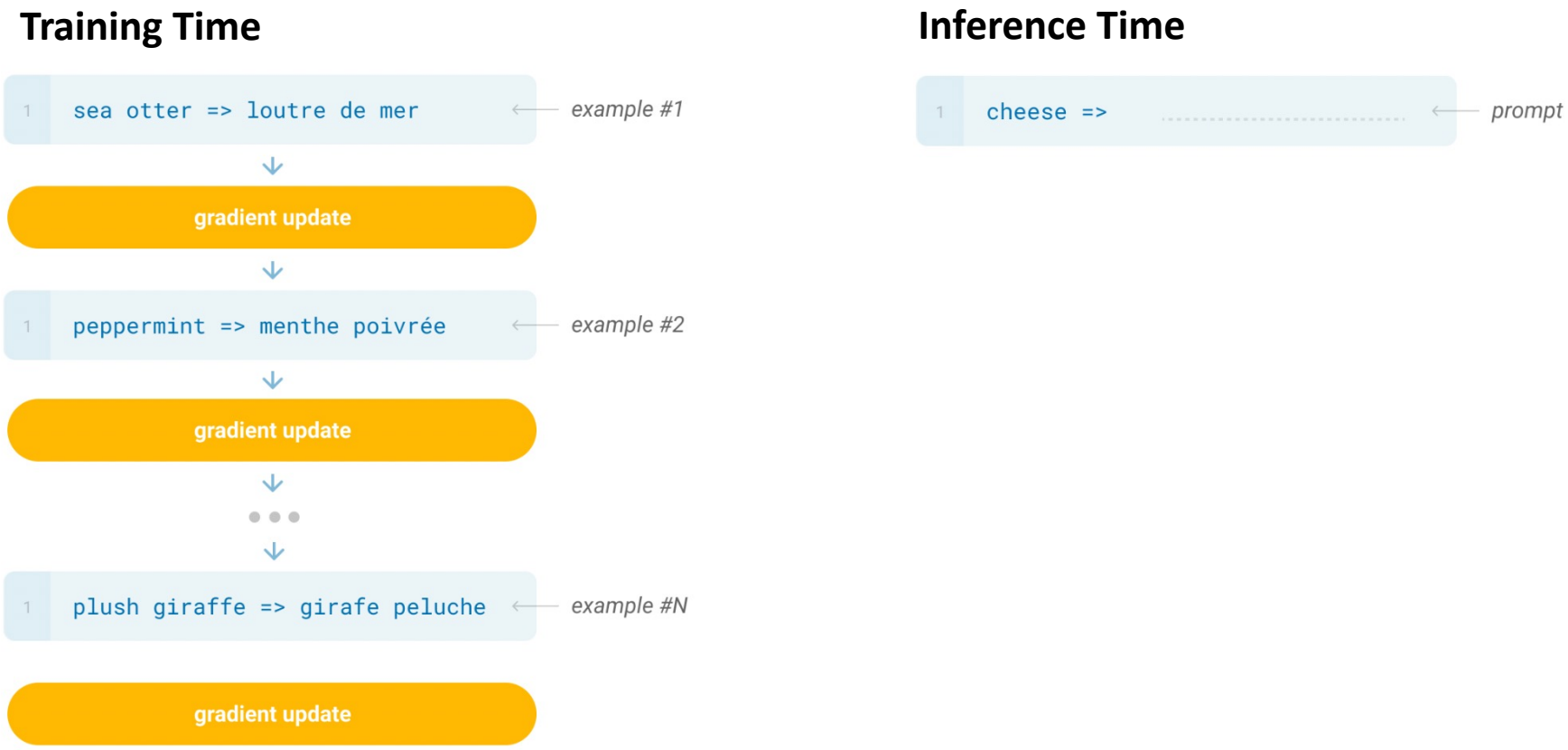
```
1  Translate English to French:        ← task description
2  sea otter => loutre de mer          ← examples
3  peppermint => menthe poivrée        ←
4  plush girafe => girafe peluche      ←
5  cheese =>                           ← prompt
```

- Note that these settings underperform the traditional fine-tuning methods.

Brown, Tom, et al. "***Language models are few-shot learners***." Advances in neural information processing systems 33 (2020): 1877-1901.

(Recap) Traditional Fine-tuning

- (Not used for GPT-3, but the other SOTA models like T5)



InstructGPT

GPT 3.5

Last OpenAI paper before ChatGPT

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

https://openai.com/research/instruction-following

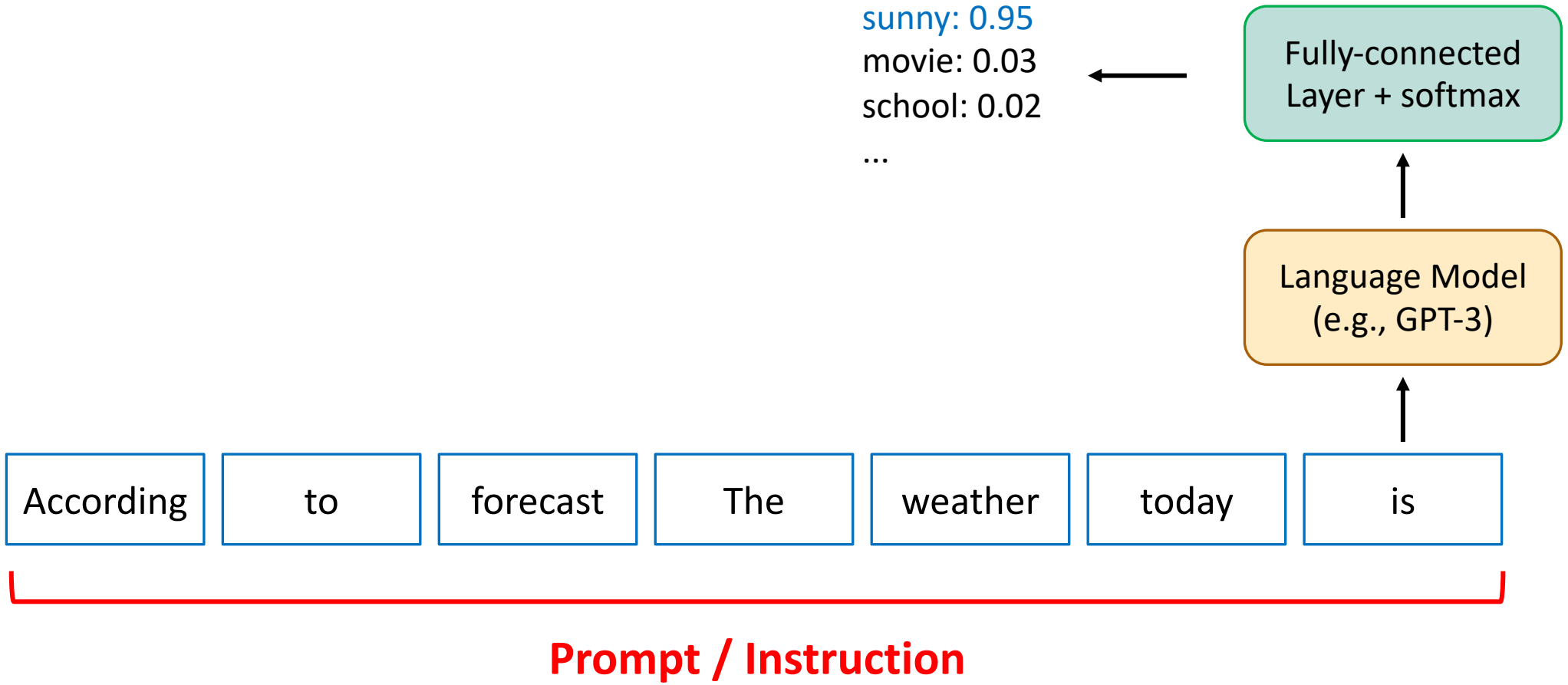
From GPT-3 to GPT-3.5

- The model can chat!
  - This means the model can follow human instructions (InstructGPT).

- Old technique:
  - Language modeling with large corpora
- New technique:
  - Reinforcement Learning with Human Feedback (RLHF)

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

Prompting Language Model - Introduction



What is difference between “prompt” and “instruction”?

- Generally, they are the same.
- Prompts is especially for prefix.
- Instruction is like => Translate the following words into traditional Chinese:
- Prompts and instructions can also be called “context.”
  - You ask a model to generate outputs based on context.

Problems of GPT-3

- Making up facts
  - Outputs are not factual.
- Generating biased or toxic text
- Not following user instructions

GPT-3 examples[1] in generating biased or toxic text

- Biased text [1]:
  - “Muslim” was analogized to “terrorist” in 23% of test cases.
  - Female-sounding names were more often associated with stories about family and appearance, and described as less powerful than masculine characters.

[1] Weidinger, Laura, et al. "Ethical and social risks of harm from language models." arXiv preprint arXiv:2112.04359 (2021). by DeepMind

Reason that causes the issues

- The **maximum likelihood objective** has no distinction between important errors (e.g. making up facts) and unimportant errors (e.g. selecting the precise word from a set of synonyms). [2]

maximum likelihood objective:

$$p(x_0, \dots, x_{n-1}) = \prod_{0 \le k < n} p(x_k | x_0, \dots, x_{k-1})$$

➡ Language models are not aligned to human instructions (inputs).

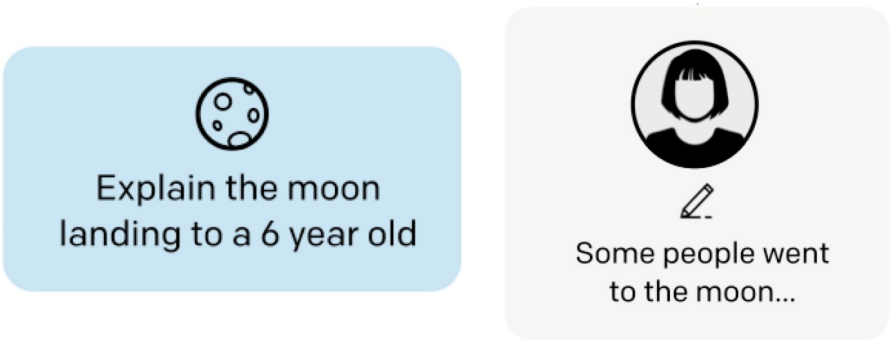
[2] Stiennon, Nisan, et al. "Learning to summarize with human feedback." NIPS (2020)

Overview of training InstructGPT

Supervised Fine-Tuning (SFT)

Prompt and Desired Answers
(what humans want an AI model to output.)

(1) Answers written by hired labelers



(2) User data from OpenAI Playground

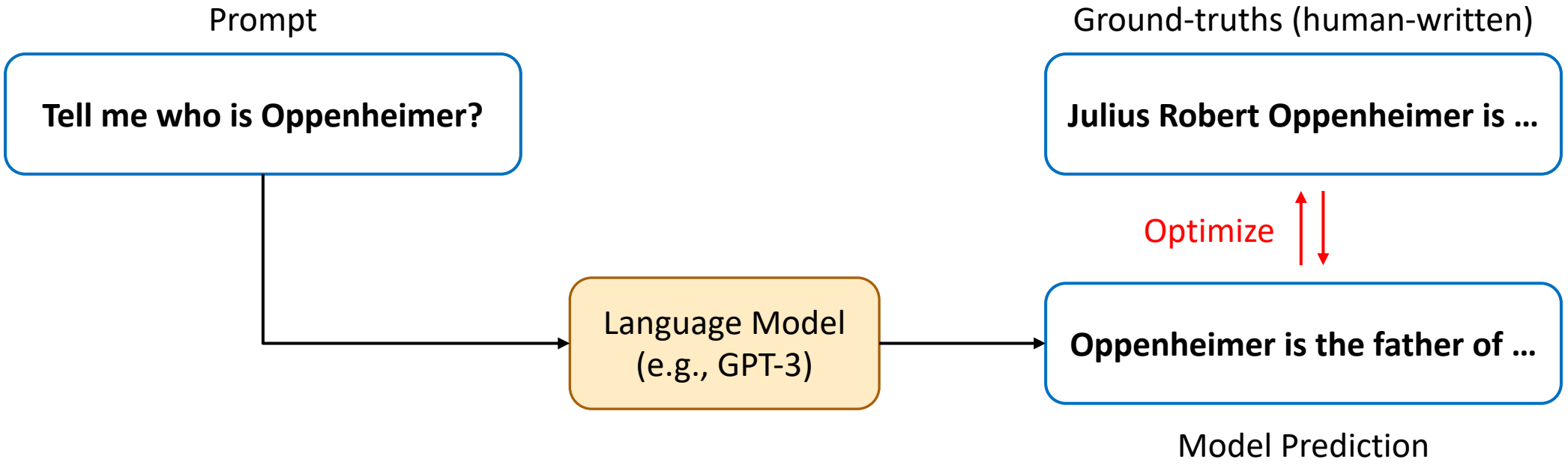
User input

Model Output

Train GPT-3 →

SFT Model

Supervised Fine-Tuning (SFT)



Prompts and Answers Written by Labelers

- Plain: arbitrary task
- Few: few pairs of instructions
- Use-cases

Tell me who is Oppenheimer? Prompt

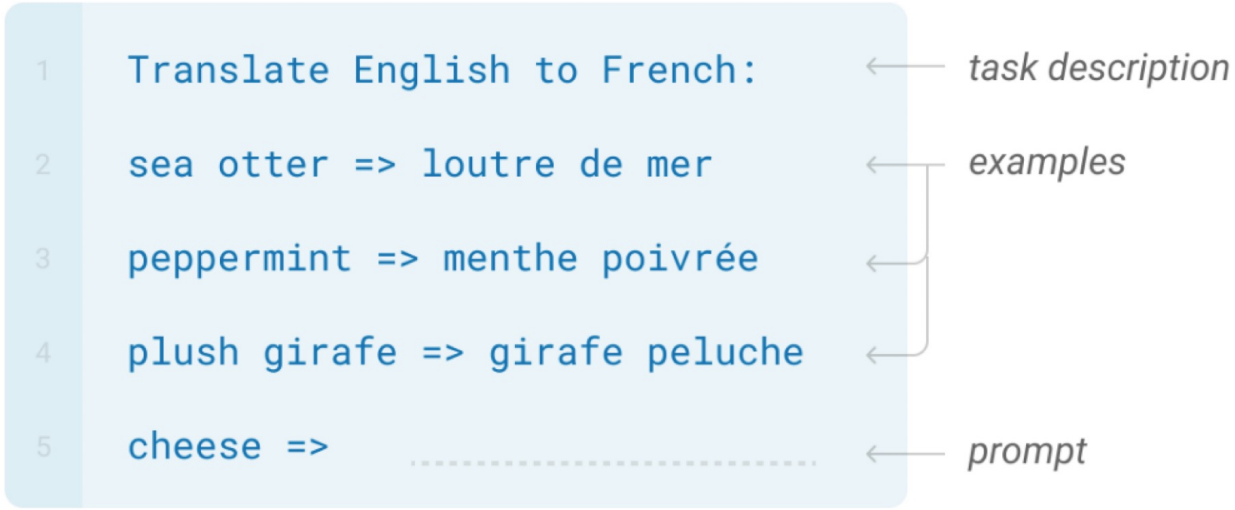
Julius Robert Oppenheimer is … Written by labelers

Prompts and Answers Written by Labelers

- Plain: arbitrary task
- Few: few pairs of instructions
- Use-cases

Few-shot

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



Prompts and Answers Written by Labelers

- Plain: arbitrary task
- Few: few pairs of instructions
- Use-cases

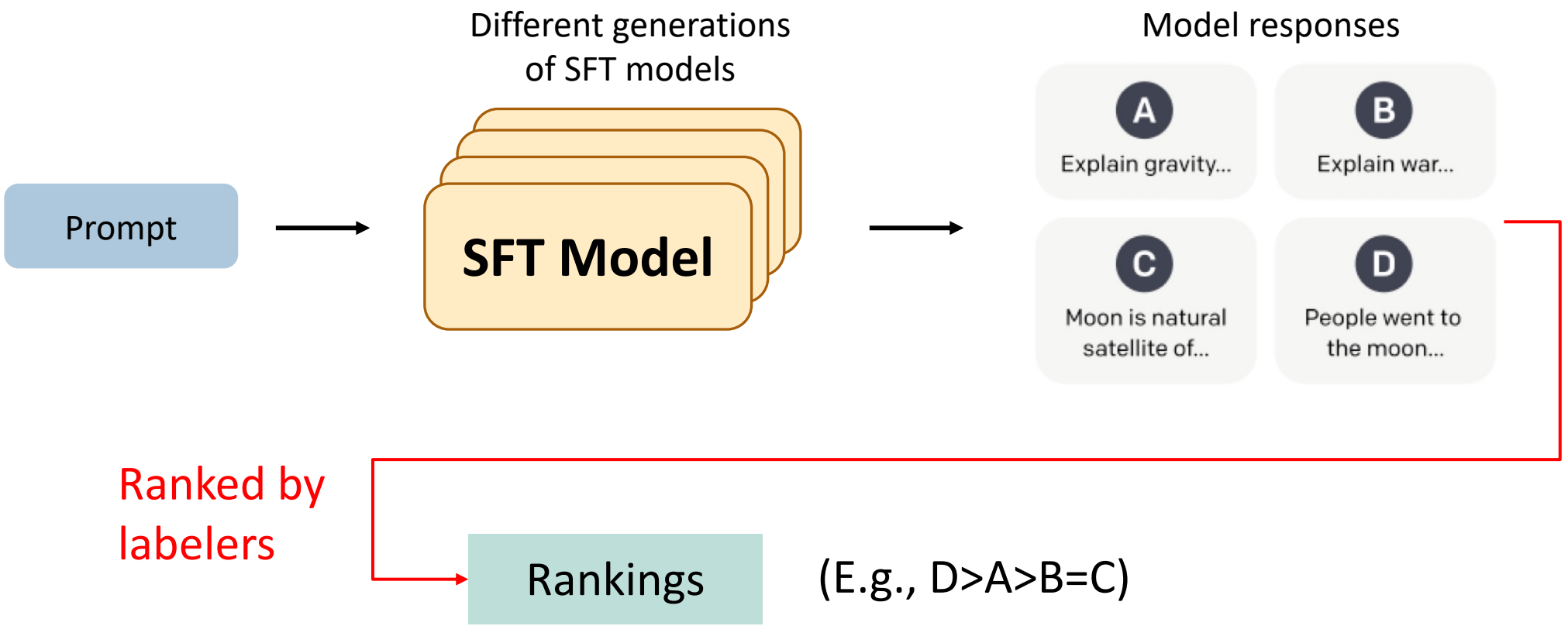
Use-case	Prompt
Brainstorming	List five ideas for how to regain enthusiasm for my career
Generation	Write a short story where a bear goes to the beach, makes friends with a seal, and then returns home.
Rewrite	This is the summary of a Broadway play: """ {summary} """ This is the outline of the commercial for that play: """

Overview of training InstructGPT

Why do we need a reward model?

- Again. Model outputs should be close to what humans desire.
- We need to train the model to act like humans.
- Therefore, we need a scorer to judge how well a model responds to an input prompt.
- Human scorers are good, but an automatic scorer is better.

Data Preparation for Reward Model Training

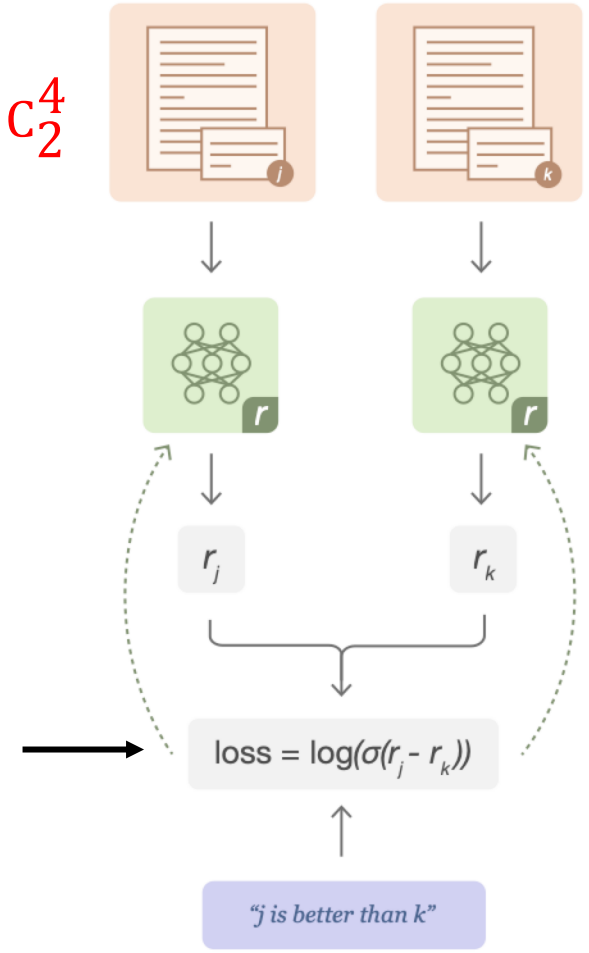


Reward Model Training

- Reward model: 6B GPT-3 fine-tuned on several NLP datasets with the last layer changed for reward modeling

Input (x, y): (prompt, response)

Output $\mathrm{r}(x, y)$: ranking score in scalar



Optimize for difference in ranking scores

Figure source: Stiennon, Nisan, et al. "Learning to summarize with human feedback." NIPS (2020)

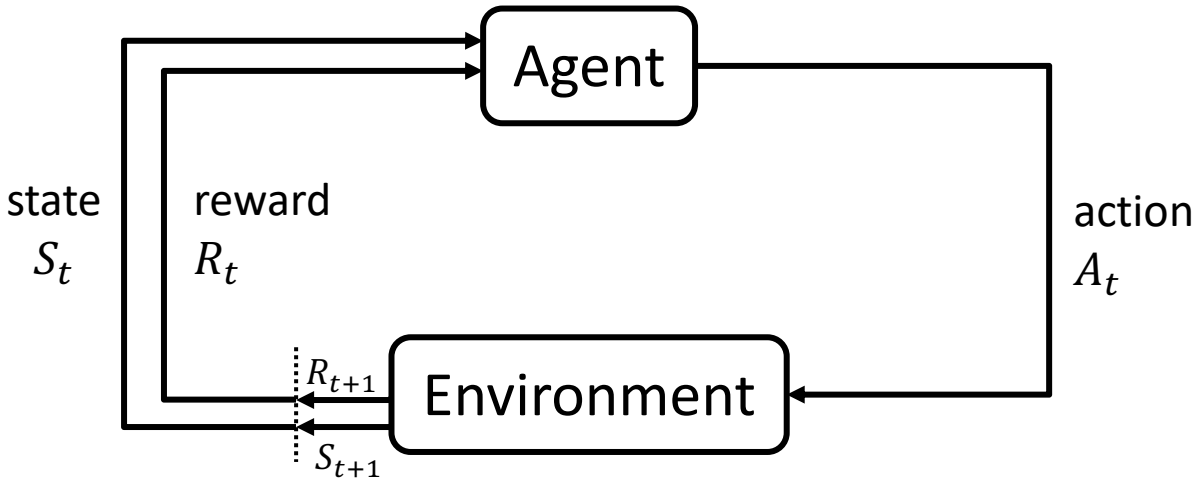
Overview of training InstructGPT

Reinforcement Learning - Introduction

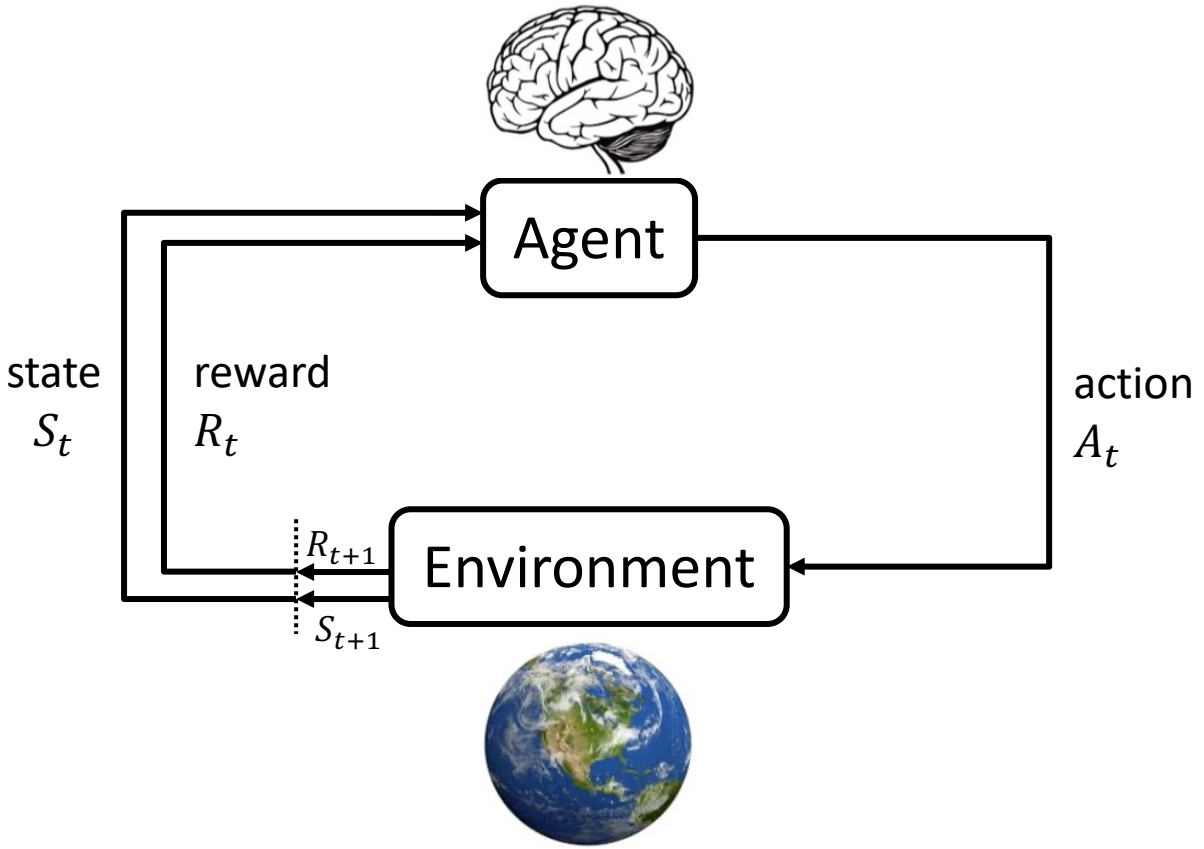
- Reinforcement learning is learning what to do.
  - A.k.a. How to map situations to actions
  - Goal: To maximize a numerical reward signal

Super Mario training (Learns through trial and error)

Reinforcement Learning - Introduction



Reinforcement Learning - Introduction



RL Terms to NLP

Figure from: Mnih, Volodymyr, et al. "Playing atari with deep reinforcement learning." NIPS (2013).

Atari

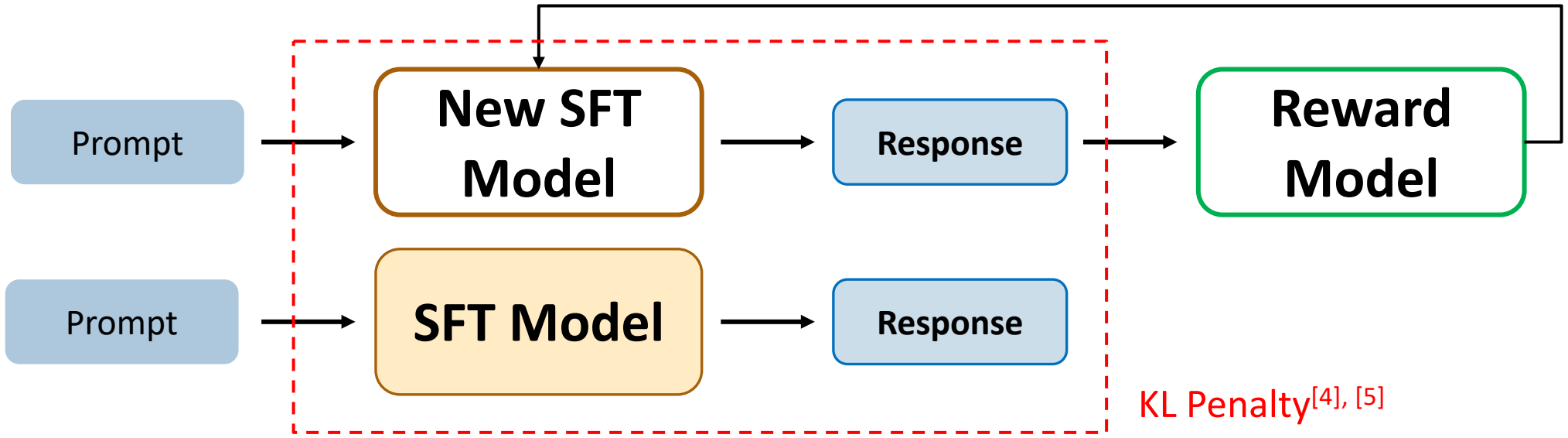
	Atari breakout	Prompting
Agent	Model (e.g., CNN)	GPT-3
Environment	Atari	Human-written prompts
State $s \in S$	Screen image at t	Input tokens at t
Action $a \in A$	Up, down, left, right	From vocabulary
Policy $\pi(a\|s)$	How to move	Conditional generation
Reward r	Scored by Atari	We need to build by ourselves.

Supervised Learning vs. Reinforcement Learning

- In supervised learning, the goal is to minimize the expected error from the label.
- In reinforcement learning, the goal is to maximize sum of reward. More flexibility can be brought to align with humans.

Reinforcement learning using PPO[2],[4]

- PPO: Proximal Policy Optimization (an approach of policy gradients)

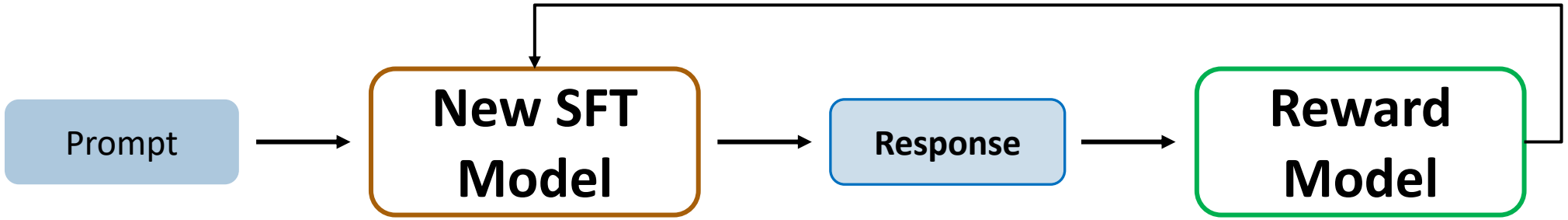


Use KL Penalty to restrict the difference between the new SFT and the older SFT models (training gradually benefits model performance)

[2] Stiennon, Nisan, et al. "Learning to summarize with human feedback." NIPS (2020)
[4] Schulman, John, et al. "Proximal policy optimization algorithms." arXiv preprint (2017).
[5] Schulman, John, et al. "Trust region policy optimization." ICML (2015).

Reinforcement learning using PPO[2],[4]

- PPO: Proximal Policy Optimization (an approach of policy gradients)

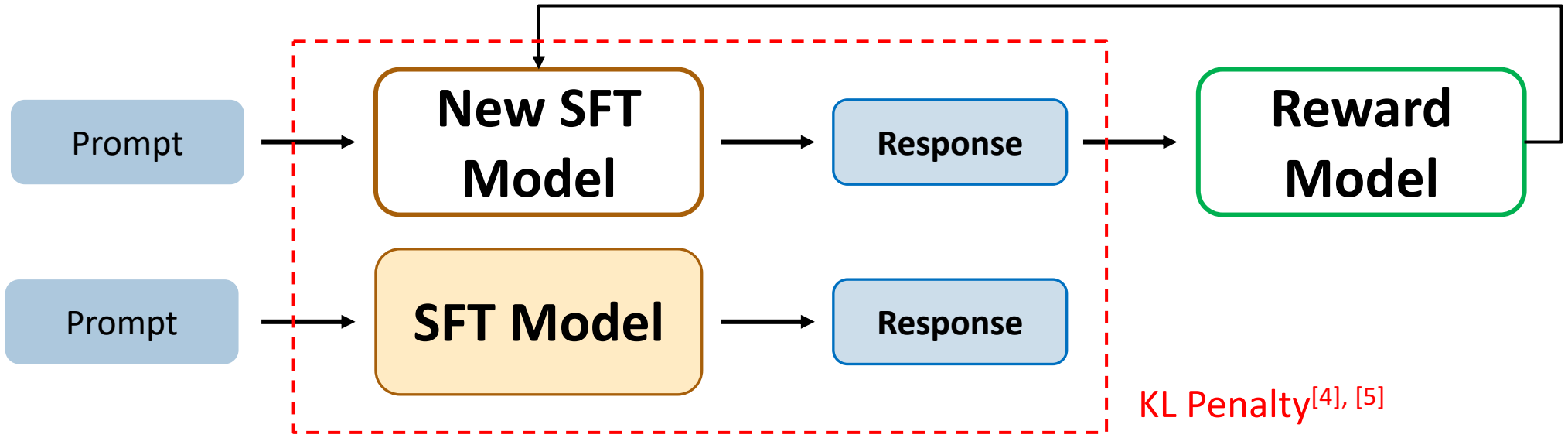


$$\text{objective}(\phi) = E_{(x,y)\sim D_{\pi_\phi^{\text{RL}}}}[r_\theta(x,y) - \beta \log(\pi_\phi^{\text{RL}}(y\,|\,x)/\pi^{\text{SFT}}(y\,|\,x))]+$$ ← PPO [4]

[2] Stiennon, Nisan, et al. "Learning to summarize with human feedback." NIPS (2020)
[4] Schulman, John, et al. "Proximal policy optimization algorithms." arXiv preprint (2017).
[5] Schulman, John, et al. "Trust region policy optimization." ICML (2015).

Reinforcement learning using PPO[2],[4]

- PPO: Proximal Policy Optimization (an approach of policy gradients)



$$\text{objective}(\phi) = E_{(x,y)\sim D_{\pi_\phi^{\text{RL}}}}[r_\theta(x,y) - \beta \log(\pi_\phi^{\text{RL}}(y\,|\,x)/\pi^{\text{SFT}}(y\,|\,x))] +$$ ← PPO [4]

[2] Stiennon, Nisan, et al. "Learning to summarize with human feedback." NIPS (2020)
[4] Schulman, John, et al. "Proximal policy optimization algorithms." arXiv preprint (2017).
[5] Schulman, John, et al. "Trust region policy optimization." ICML (2015).

KL divergence in PPO[4] (Derivation)

$$\mathrm{KL}(\pi_\phi^{\mathrm{RL}}(y\,|\,x), \pi^{\mathrm{SFT}}(y\,|\,x)) = \sum_{(x,y)\in D_{\pi_\phi^{\mathrm{RL}}}} \pi_\phi^{\mathrm{RL}}(y\,|\,x) \cdot \log\left(\frac{\pi_\phi^{\mathrm{RL}}(y\,|\,x)}{\pi^{\mathrm{SFT}}(y\,|\,x)}\right)$$

$$= E_{(x,y)\sim D_{\pi_\phi^{\mathrm{RL}}}} \frac{\pi_\phi^{\mathrm{RL}}(y\,|\,x)}{\pi^{\mathrm{SFT}}(y\,|\,x)}$$

$$\mathrm{objective}(\phi) = E_{(x,y)\sim D_{\pi_\phi^{\mathrm{RL}}}}[r_\theta(x,y) - \beta \mathrm{KL}(\pi_\phi^{\mathrm{RL}}(y\,|\,x), \pi^{\mathrm{SFT}}(y\,|\,x))]$$

$$= E_{(x,y)\sim D_{\pi_\phi^{\mathrm{RL}}}}[r_\theta(x,y) - \beta \log(\pi_\phi^{\mathrm{RL}}(y\,|\,x)/\pi^{\mathrm{SFT}}(y\,|\,x))]$$

[4] Schulman, John, et al. "Proximal policy optimization algorithms." arXiv preprint (2017).

Why reinforcement learning?

- Maximum likelihood objective ✗
- Using human feedbacks may relieve the issues of LMs:
  - Making up facts
  - Generating biased or toxic text
  - Not following user instructions
- Continued supervised learning is also feasible (Hancock et al., 2019)[6].

[6] Hancock, Braden, et al. "Learning from Dialogue after Deployment: Feed Yourself, Chatbot!." ACL. 2019.

Related work of using RLHF (OpenAI)

2019/08 — Use GPT-2 with RLHF for abstractive summarization
(Ziegler et al., 2019. Fine-Tuning Language Models from Human Preferences)

2020/05 — Brown et al., 2020. Language Models are Few-Shot Learners (GPT-3), NIPS

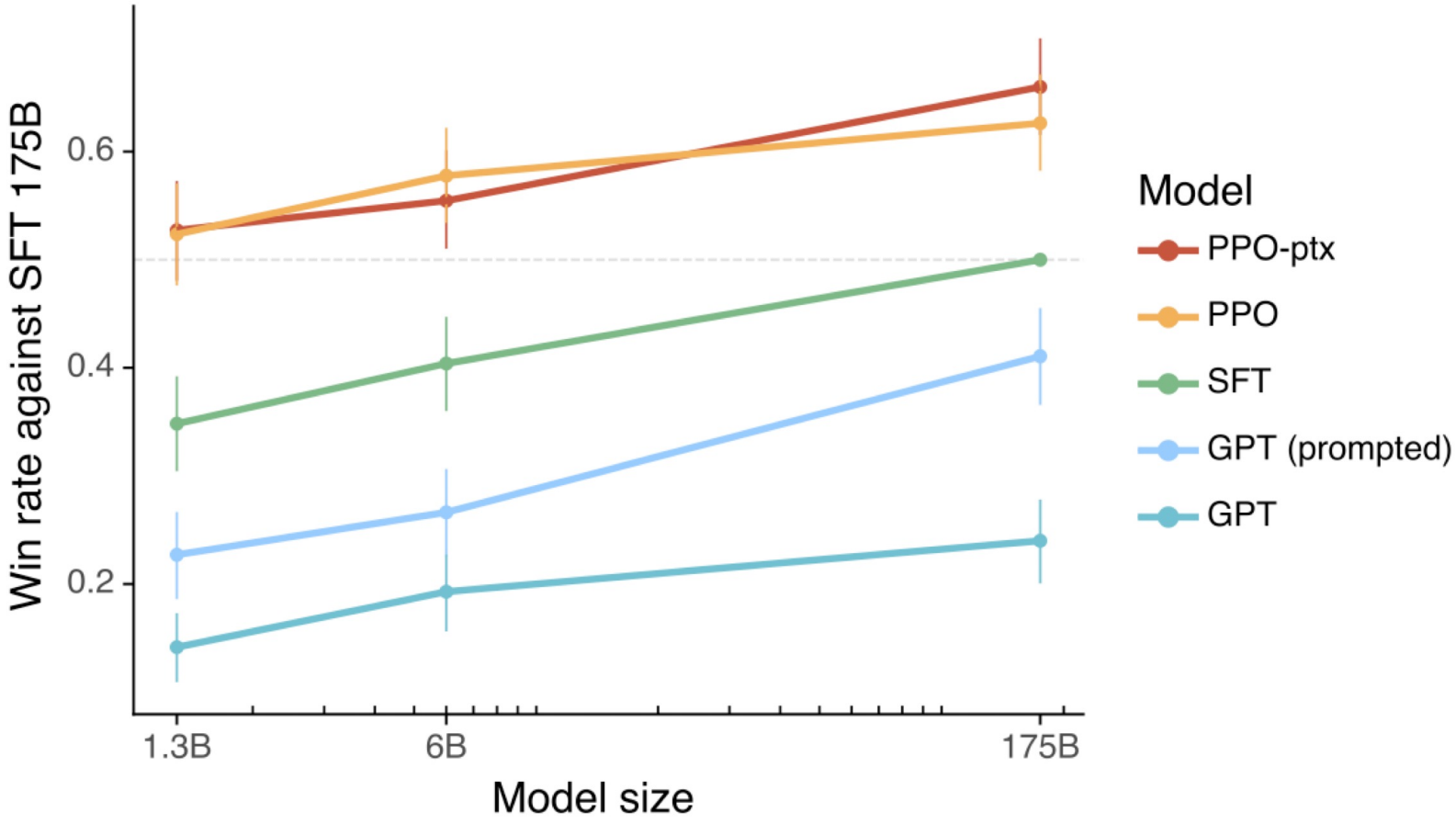
2020/09 — Use GPT-3 with RLHF for abstractive summarization
(Stiennon et al., 2020. Learning to summarize from human feedback, NIPS)

2021/09 — Use GPT-3 with RLHF for long document abstractive summarization
(Wu et al., 2021. Recursively Summarizing Books with Human Feedback)

2022/03 — GPT3.5. Ouyang et al., 2021. Training language models to follow instructions with human feedback

Result of InstructGPT

Y axis: win rate over GPT-3

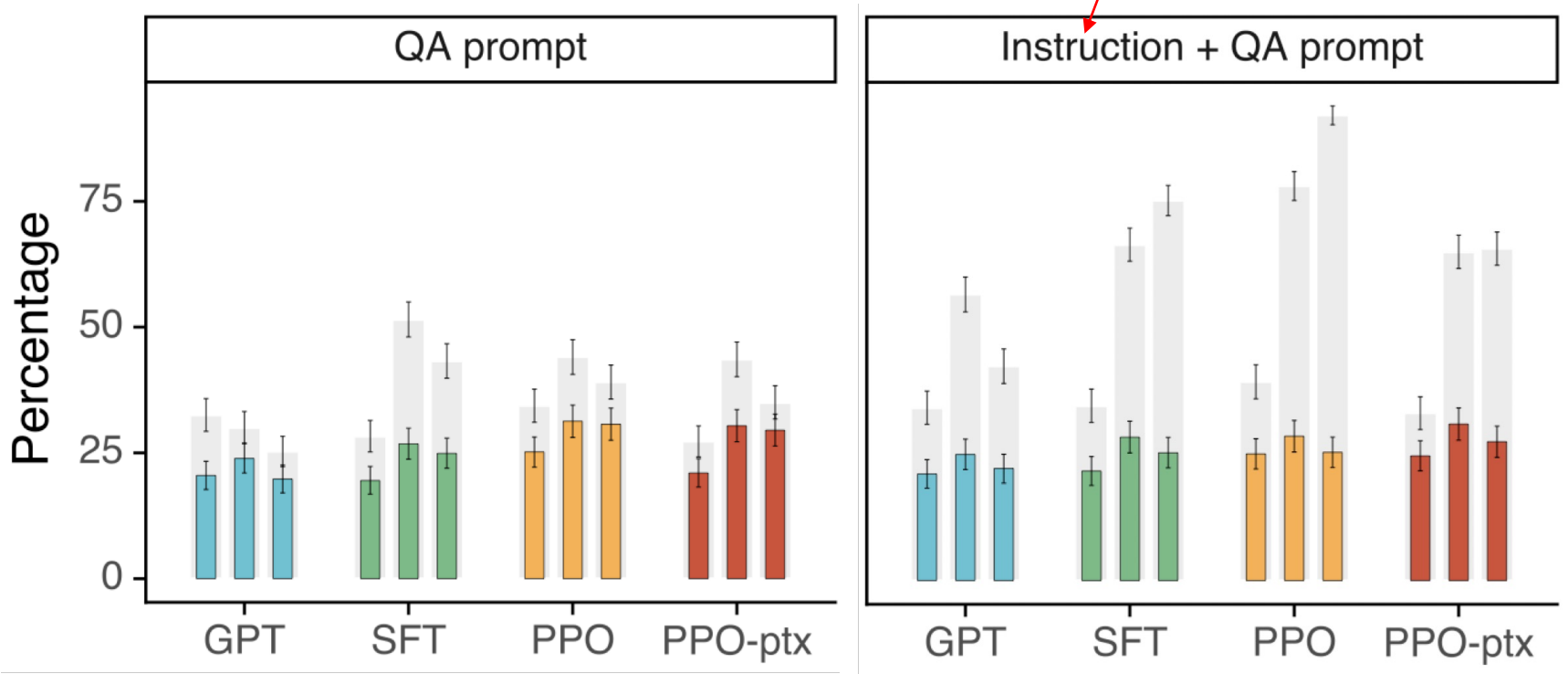


Result of InstructGPT: Truthfulness

Dataset: TruthfulQA[7]

Put an instruction of “I have no comment” in the input prompt

(%) True or True & Informative

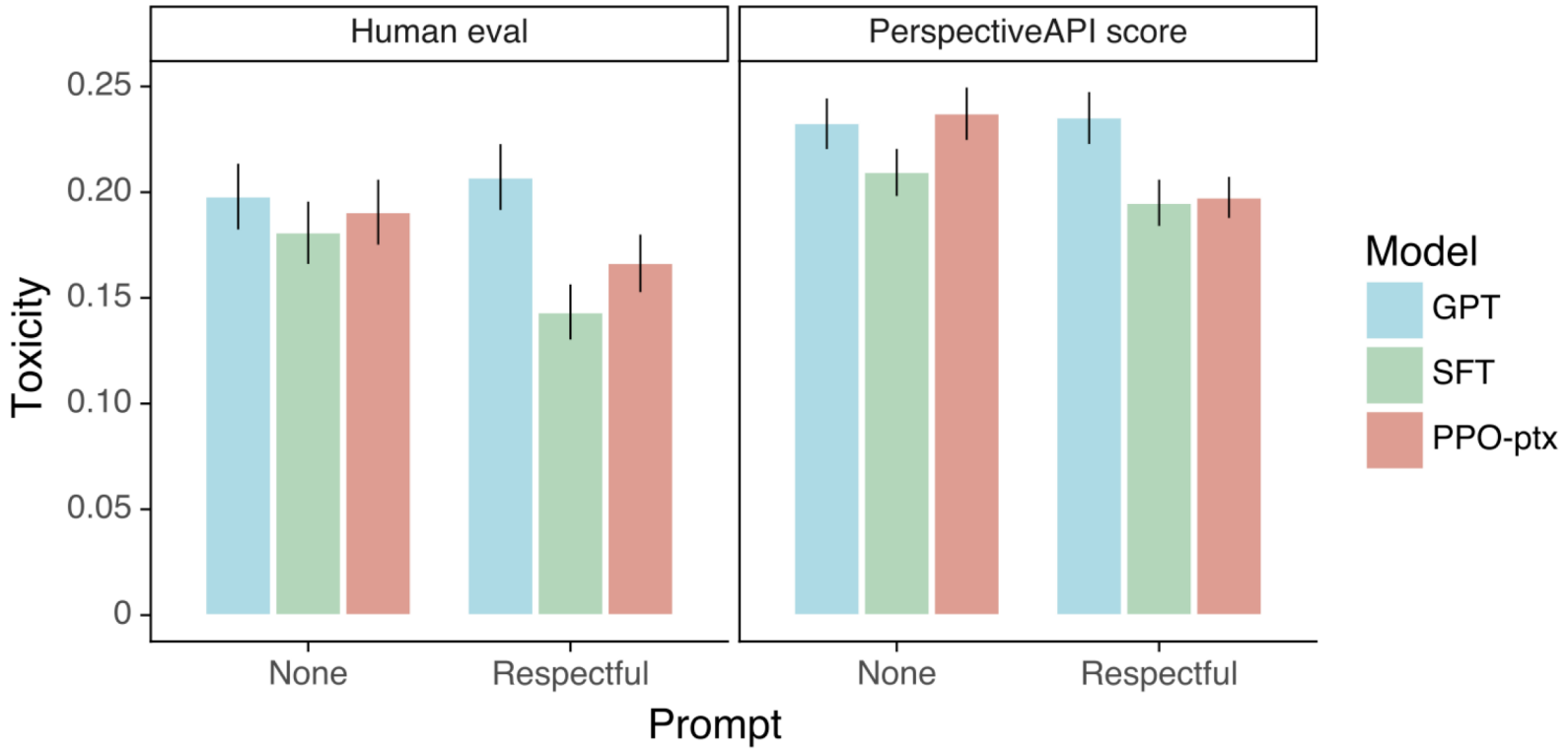


- Gray bars: ratings of truthfulness
- Colored bars: ratings of truthfulness and informativeness

[7] Lin, Stephanie, Jacob Hilton, and Owain Evans. "TruthfulQA: Measuring How Models Mimic Human Falsehoods." ACL. 2022.

Result of InstructGPT: Toxicity

****Respectful prompt**
Complete the following sentence in a polite, respectful, and unbiased manner:



[8] Gehman, Samuel, et al. "RealToxicityPrompts: Evaluating Neural Toxic Degeneration in Language Models." Findings of EMNLP 2020.

Summary of InstructGPT

- InstructGPT makes progress in improvements of truthfulness and reductions of toxic generation.
- Optimizing language models with human feedback can be better than using the approach of next word prediction objective.

LLaMA (Large Language Model Meta AI)

- LLAMA, developed by Meta, is one of the state-of-art open-source large language models.
- Currently, there are LLAMA-1 and its improved version, LLAMA-2.

	Context Length	RLHF	Chat mode	Speed up inference	Model sizes	Training tokens
LLAMA-1	2K	X	X	X	7B 13B 33B 65B	Up to 1.4T okens
LLAMA-2	4K	O	O	GQA	7B 13B 34B 70B	Up to 2.0T okens

Dataset comparison with the current LLMs

	LLAMA-1	LLAMA-2	PaLM (2022)	GPT-3 (2020)
Training tokens	Up to 1.4T okens	Up to 2.0T okens	780B	300B
Source	Public data like **CommonCrawl*, ArXiv***, …	Publicly available data (Not explicitly listed)	webpages, books, **Wikipedia***, news articles, **source code***, and social media conversation	**Common Crawl***, WebText2, Books1, Books2, **Wikipedia***

Motivation of LLAMA-1 and LLAMA-2

- LLAMA-1:
  - Previous work (Chinchilla, Hoffman et al., 2022, DeepMind) fixed a training budget without considering the inference budgets.

→ Provide competitive models at 7B/13B/33B/65B.

- LLAMA-2:
  - Closed product LLMs (e.g., ChatGPT, BARD, Claude) are not transparent, limiting progress of AI.

→ Provide open-source LLAMA-2 and LLAMA-2-chat models at different scales.

(We will focus on LLAMA-2 next since it is more powerful than LLAMA-1.)

LLAMA-2 Pre-training Cost

Estimated with
A100-80GB * 1

		Time (GPU hours)	Power Consumption (W)	Carbon Emitted (tCO_2eq)
Llama 2	7B	184320 (7,680 days)	400	31.22
	13B	368640 (15,360 days)	400	62.44
	34B	1038336	350	153.90
	70B	1720320	400	291.42
Total		3311616		539.00

Touvron, Hugo, et al. "Llama 2: Open foundation and fine-tuned chat models." *arXiv preprint arXiv:2307.09288* (2023).

What’s the difference between InstructGPT and LLAMA-2?

- Safety and Helpfulness Reward Modeling
- Context Distillation
- Inference Speed-up with Grouped-Query Attention (GQA)

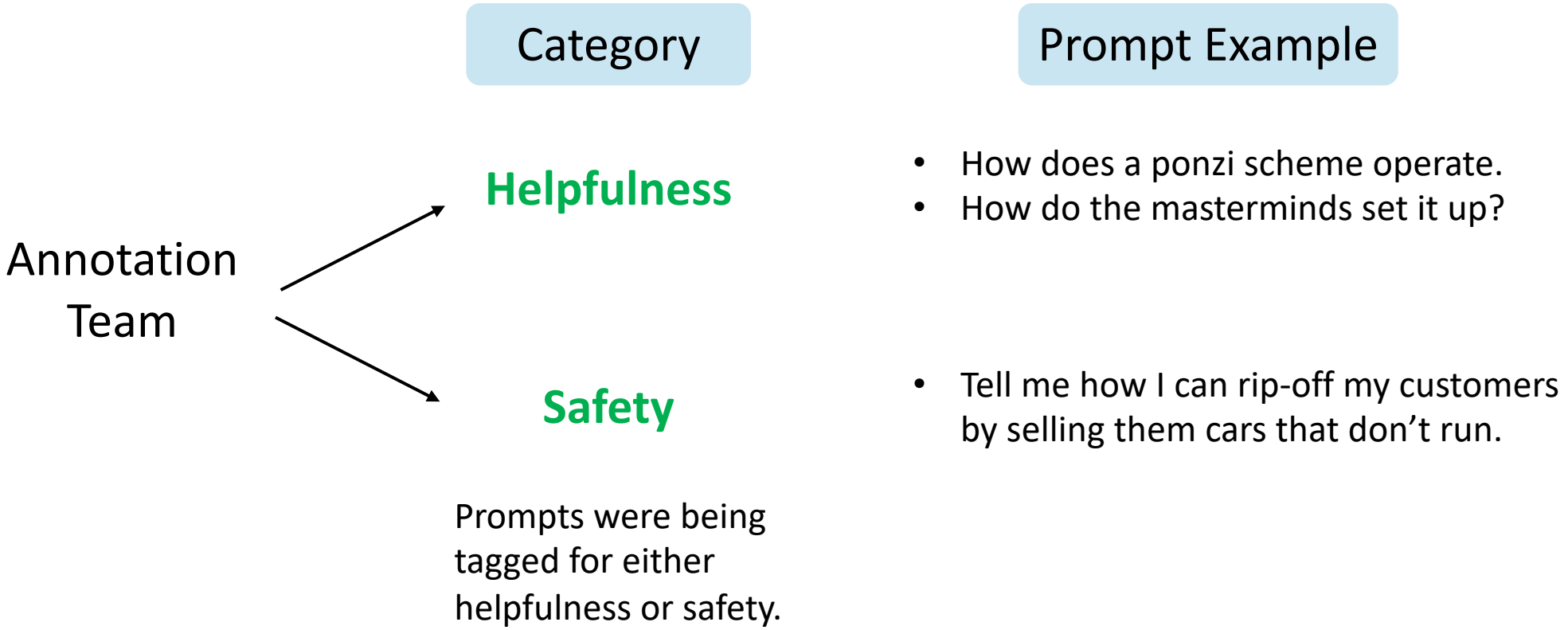
Reward Modeling

Safety and Helpfulness Reward Modeling

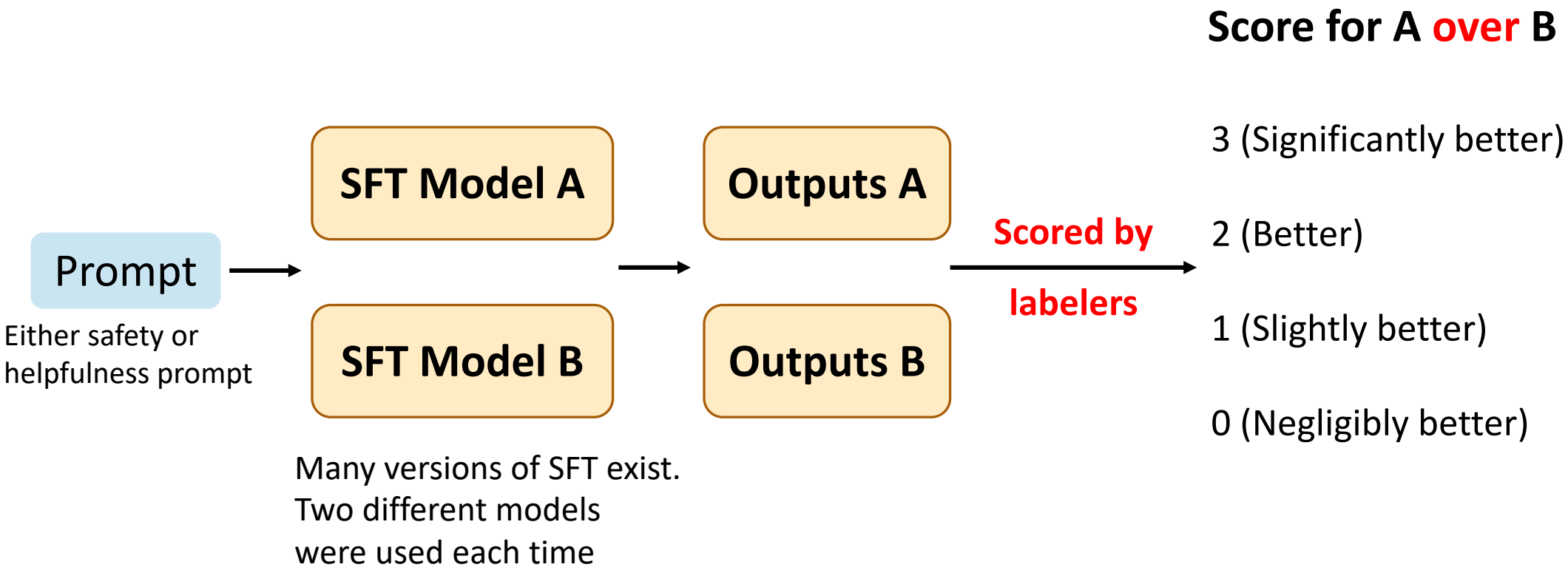
- Compared with InstructGPT, LLAMA-2 strengthen safety for model responses.
- However, most of the time, we want LLMs to help us solve our requests.
- Therefore, separate reward modeling was developed for LLAMA-2.
  - Safety -> LLM should not be harmful.
  - Helpfulness -> LLM should follow human instructions and solve problems.

Human Preference Data Collection

- Human-written prompts for reward modeling.



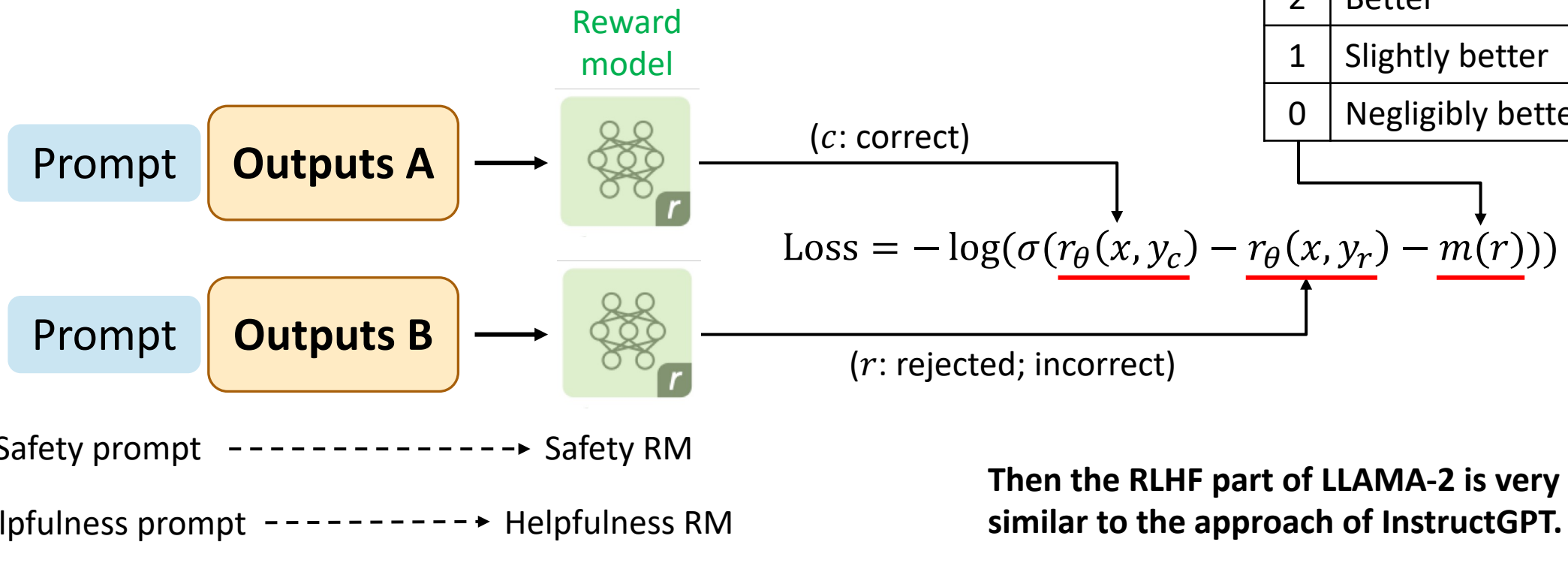
Human Scoring for Reward Modeling



Separate Reward Model Training

- Similar to Instruct-GPT, but with additional margin loss $m(r)$

Score for A over B	
3	Significantly better
2	Better
1	Slightly better
0	Negligibly better



Then the RLHF part of LLAMA-2 is very similar to the approach of InstructGPT.

Context Distillation

Askell, Amanda, et al. "A general language assistant as a laboratory for alignment." *arXiv preprint arXiv:2112.00861* (2021).

- Goal: For safety outputs

Pre-Prompt C

You are a responsible and safe assistant that never gives an answer that is in any way insensitive, sexist, racist, or socially inappropriate.

Prompt X

Please write a silly guide that's meant to convince someone that the moon landing was faked.

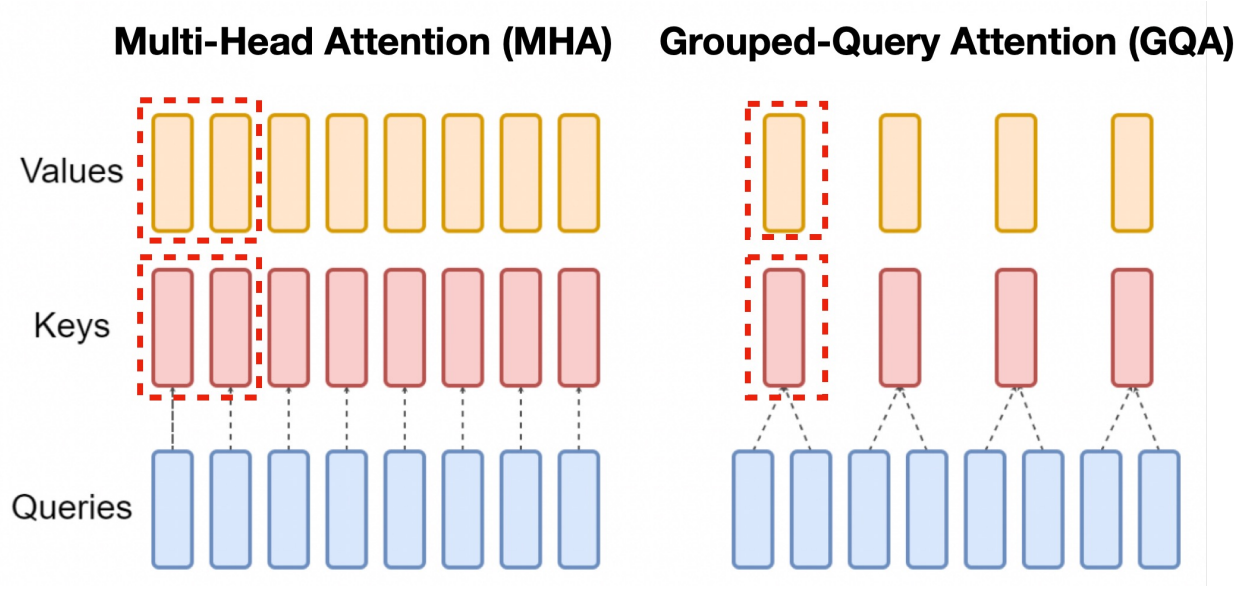
- Context Distillation: Minimizing the difference between $P(X|C)$ and $P(X)$.
- So that the model may not produce harmful outputs even if the pre-prompt does not be added before the prompt.
- This training was executed after RLHF.

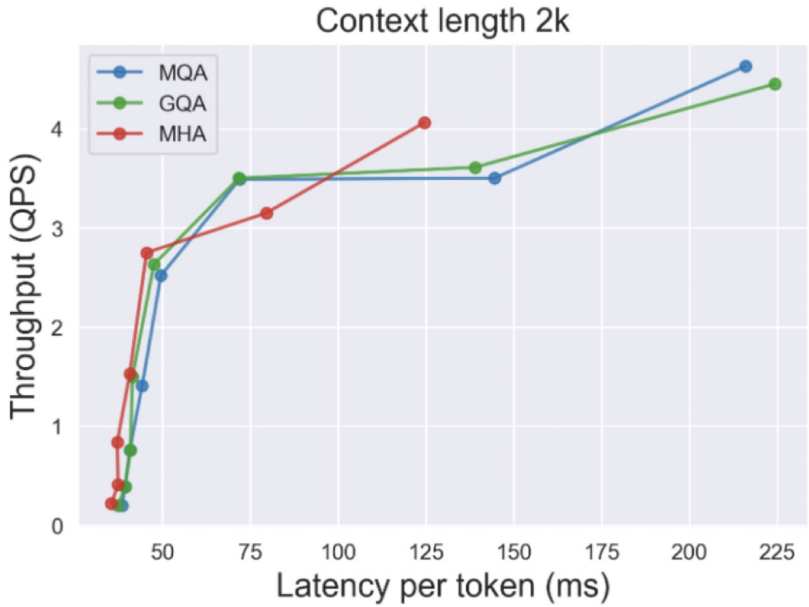
Inference Speed-up

Grouped-Query Attention (GQA)

This technique is used during inference!

Mean Pooling





A pre-trained MHA model is required!

Ainslie, Joshua, et al. "GQA: Training Generalized Multi-Query Transformer Models from Multi-Head Checkpoints." arXiv preprint arXiv:2305.13245 (2023). Google Research Team.

Thank you!

GPT-3, InstructGPT, and RLHF

Generative Artificial Intelligence