Prompting, Reinforcement Learning from Human Feedback

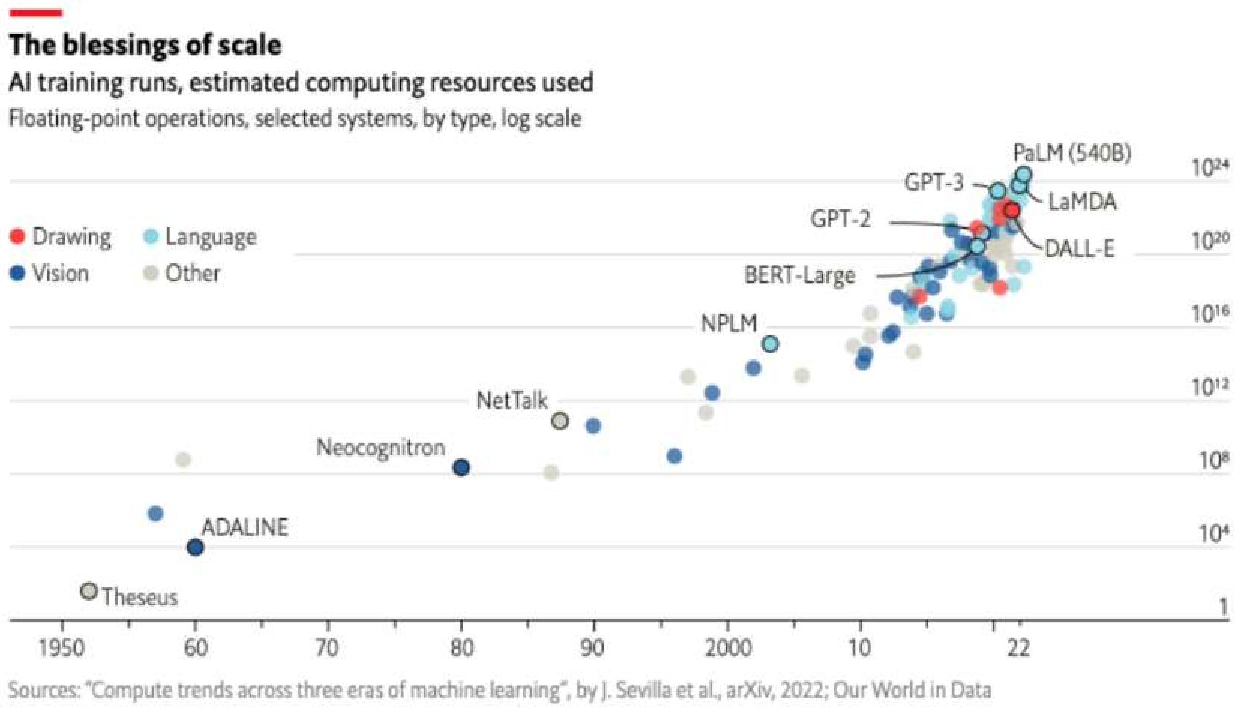
stanford - CS224n

Tobig's 19-20기 자연어 심화세션 2조

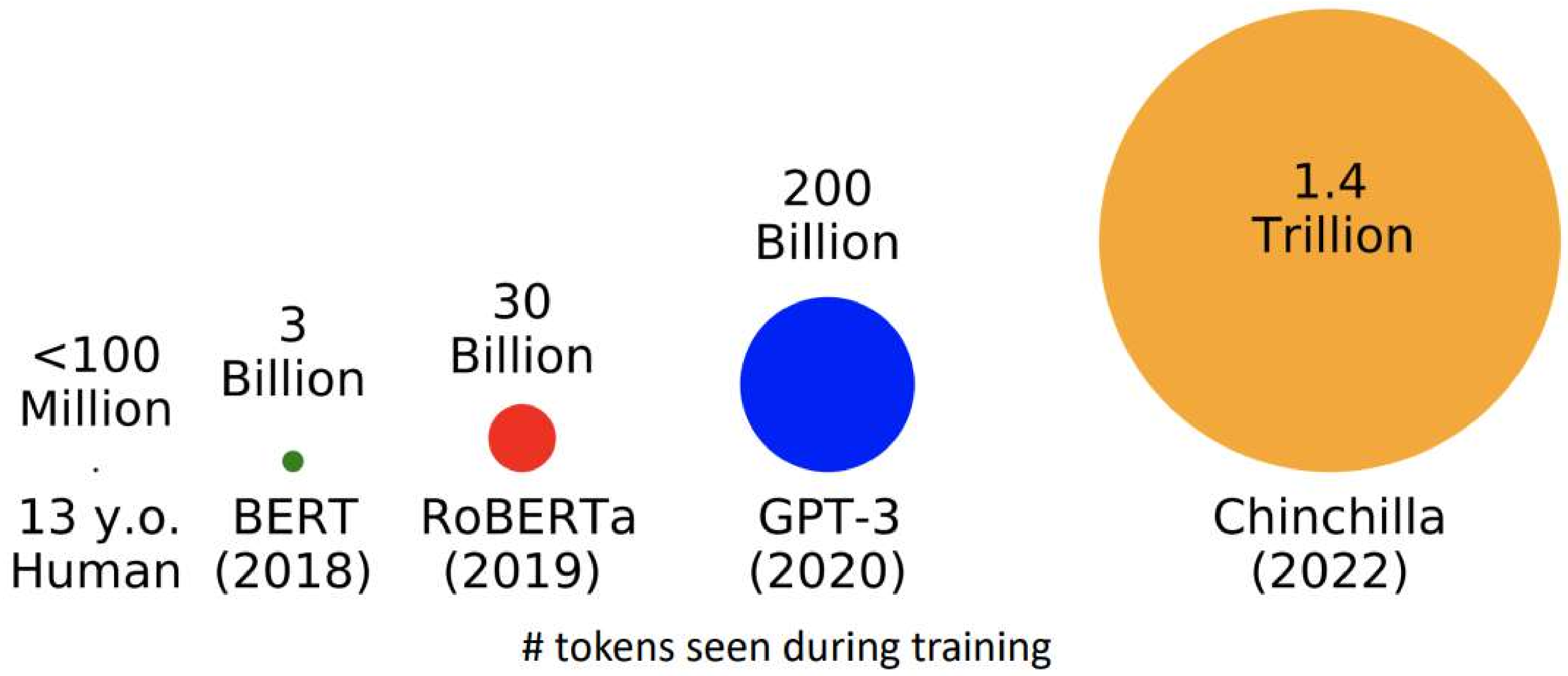
20기 경민수

Introduction

- Larger and Larger model scale : 모델 크기의 증가



- Trained on more and more data : 학습 사용 데이터의 증가



- Language models as World models & multitask assistants

- language models do rudimentary modeling of agents, beliefs, antions, math, code, medicine, etc.
: 언어 모델은 우리의 사고 추론 과정에서 기초적인 도움을 줄 수 있다.

- language models can be our multitask assistants.
: 언어 모델은 우리가 행하는 작업에 대한 보조 역할을 수행할 수 있다.

20기 경민수

Lecture Plan : From Language Models to Assistants

1. Zero-Shot (ZS) and Few-Shot (FS) In-Context Learning

2. Instruction finetuning

3. Reinforcement Learning from Human Feedback (RLHF)

4. What's next?

Lecture Plan : From Language Models to Assistants

1. Zero-Shot (ZS) and Few-Shot (FS) In-Context Learning

2. Instruction finetuning

3. Reinforcement Learning from Human Feedback (RLHF)

4. What's next?

1. Zero-Shot(ZS) and Few-Shot (FS) In-Context Learning

Emergent Ability (창발적 능력)

: 소규모 모델에는 없지만 대규모 모델에는 존재하는 능력. 대규모 모델에 많은 데이터를 투입하여 학습시킬 때, 대규모 모델이 스스로 학습하고 발전함에 따라 예상치 못하게 발생하는 새로운 능력.
창발적 능력을 통해 예상보다 더욱 높은 수준의 성능을 발휘할 수 있게 된다.

Emergent Ability of large language models

- GPT (117M parameters, 2018)
  - Transformoer decoder with 12 layers & Trained on BooksCorpus - over 7000 unique books(4.6GB text)
  : 대규모 언어 모델의 사전 훈련, downstream task에서 효율성 입증.
- GPT - 2 (2019)
  - GPT -> bigger (117M -> 1.5B), trained on much more data (4GB -> 40GB) of internet text data

Emergent zero-shot learning

: One key of emergent ability in GPT-2.
Doing many tasks with **no examples, no gradient updates.**

- GPT-2 beats SoTA on language modeling benchmarks with no task-specific fine-tuning

Context: "Why?" "I would have thought you'd find him rather dry," she said. "I don't know about that," said Gabriel.
"He was a great craftsman," said Heather. "That he was," said Flannery.
Target sentence: "And Polish, to boot," said ____
Target word: Gabriel

LAMBADA (language modeling w/ long discourse dependencies)
[Paperno et al., 2016]

	LAMBADA (PPL)	LAMBADA (ACC)	CBT-CN (ACC)	CBT-NE (ACC)	WikiText2 (PPL)
SOTA	99.8	59.23	85.7	82.3	39.14
117M	**35.13**	45.99	**87.65**	**83.4**	**29.41**
345M	**15.60**	55.48	**92.35**	**87.1**	**22.76**
762M	**10.87**	**60.12**	**93.45**	**88.0**	**19.93**
1542M	**8.63**	**63.24**	**93.30**	**89.05**	**18.34**

[Radford et al., 2019]

1. Zero-Shot(ZS) and Few-Shot (FS) In-Context Learning

Emergent ability on GPT-3 (175B parameters, 2020)

- Another increase in parameter size (1.5B -> 175B)
- and data (40GB -> over 600GB)

Emergent few-shot learning

- Specify a task by simply prepending examples of the task before your example
- Also called in-context learning, to stress that no gradient updates are perfomed when learning a new task
  : gradient update의 필요가 없음. 계산 용이성↑

↙ Zero-shot, One-shot, Few-shot 진행 비교

Zero-shot

```
Translate English to French:
cheese =>
```

One-shot

```
Translate English to French:
sea otter => loutre de mer
cheese =>
```

Few-shot

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

20기 경민수

1. Zero-Shot(ZS) and Few-Shot (FS) In-Context Learning

New methods of "prompting" LMs

[Brown et al., 2020]

Zero/few-shot prompting vs Traditional fine-tuning

- **Traditional fine-tuning**
  : giving bunch of data, doing a gradient step on each example.
  at the end we get a model that can do well on some output.

- **Zero/few-shot prompting**
  : giving some examples and ask the model to predict right answer

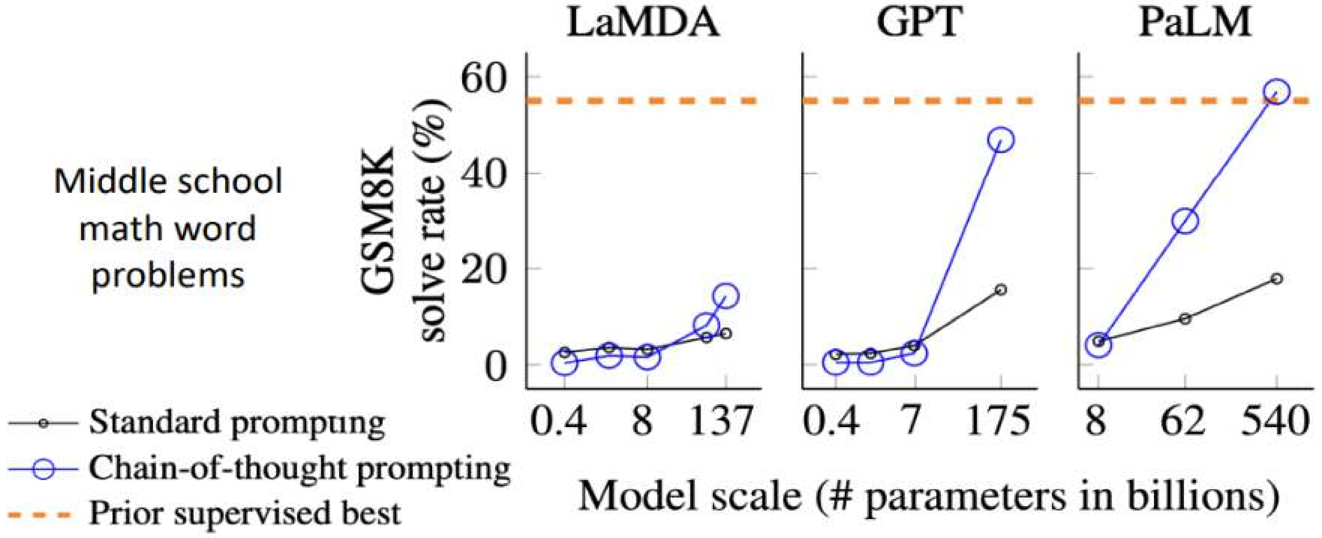
1. Zero-Shot(ZS) and Few-Shot (FS) In-Context Learning

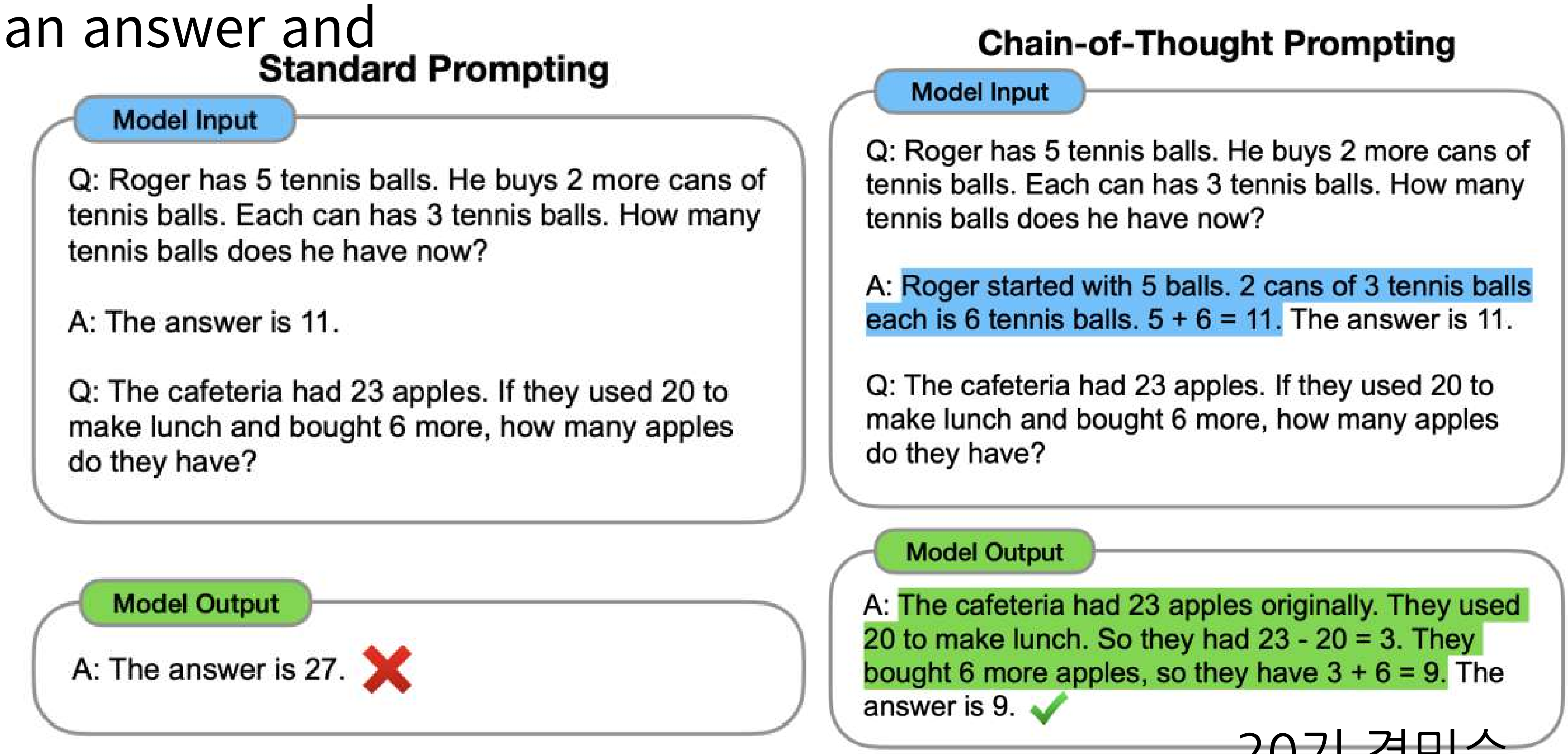
Limits of prompting for harder tasks?

- tasks involving richer, multi-step reasoning (ex. adding for larger digits) : seems to hard for even large LMs to learn throught prompting alone.
  (Humans struggle at thest tasks too!)
- solution : change the prompt! (프롬프트를 개선하자)

Chain-of-thought prompting

- demonstrate what kind of reasoning you want the model to complete
- in the prompt : not only put the question, but also put an answer and the kinds of reasoning steps that are required to arrive at the correct answer





[Wei et al., 2022; also see Nye et al., 2021]

20기 경민수

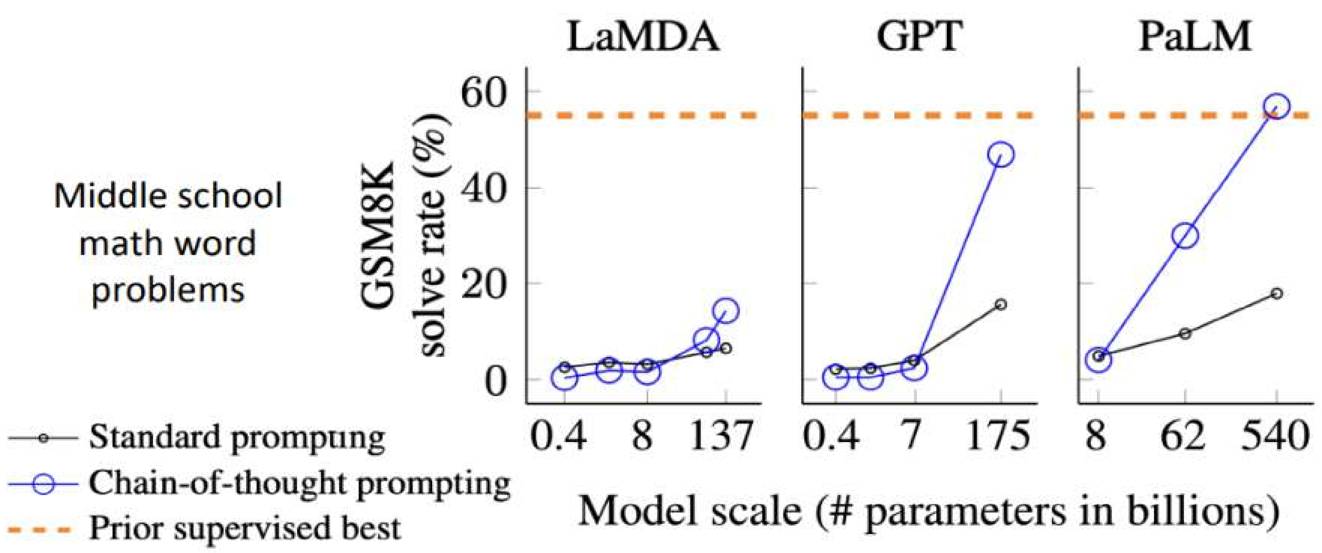
1. Zero-Shot(ZS) and Few-Shot (FS) In-Context Learning

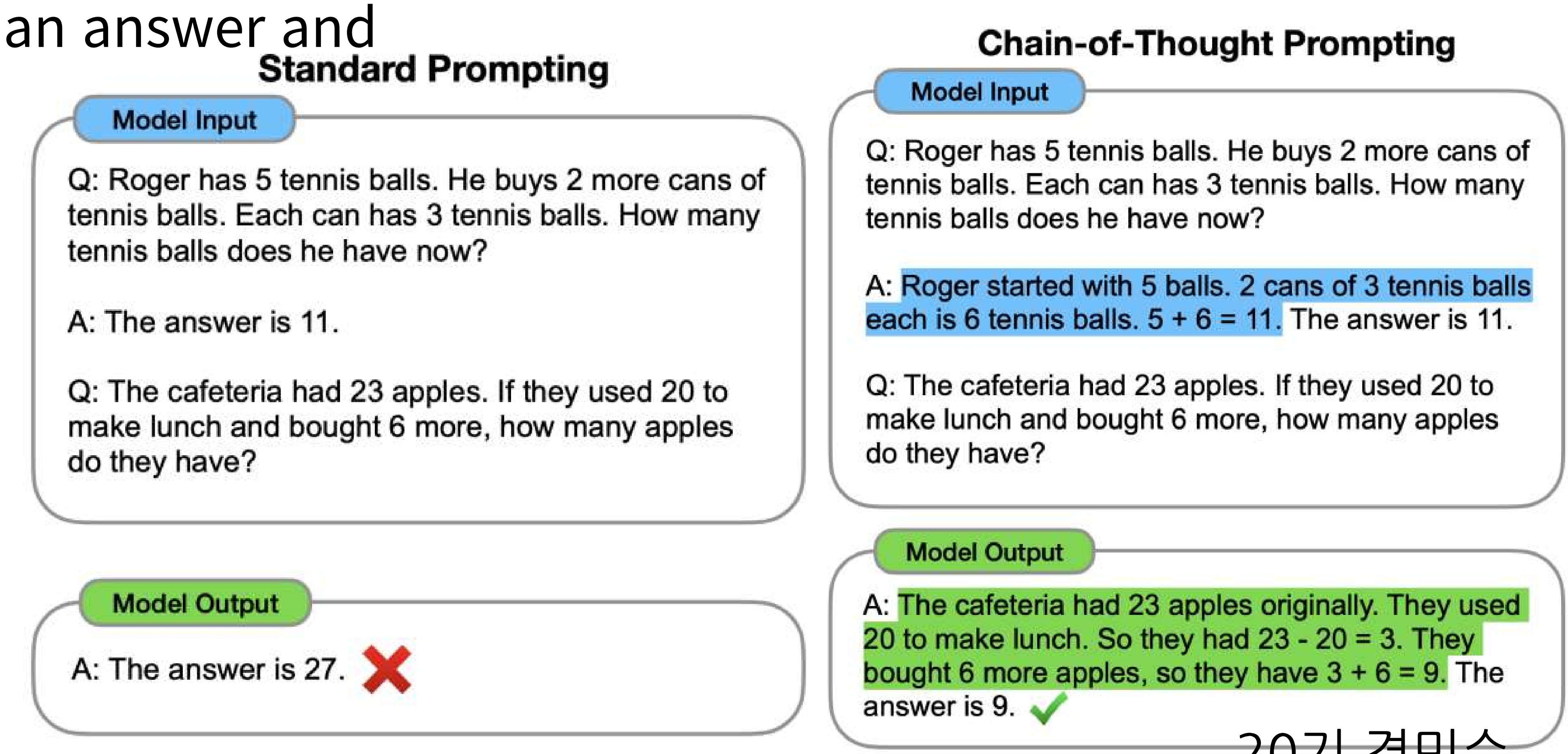
Limits of prompting for harder tasks?

- tasks involving richer, multi-step reasoning (ex. adding for larger digits) : seems to hard for even large LMs to learn throught prompting alone.
  (Humans struggle at thest tasks too!)
- solution : change the prompt!

Chain-of-thought prompting

- demonstrate what kind of reasoning you want the model to complete
- in the prompt : not only put the question, but also put an answer and the kinds of reasoning steps that are required to arrive at the correct answer





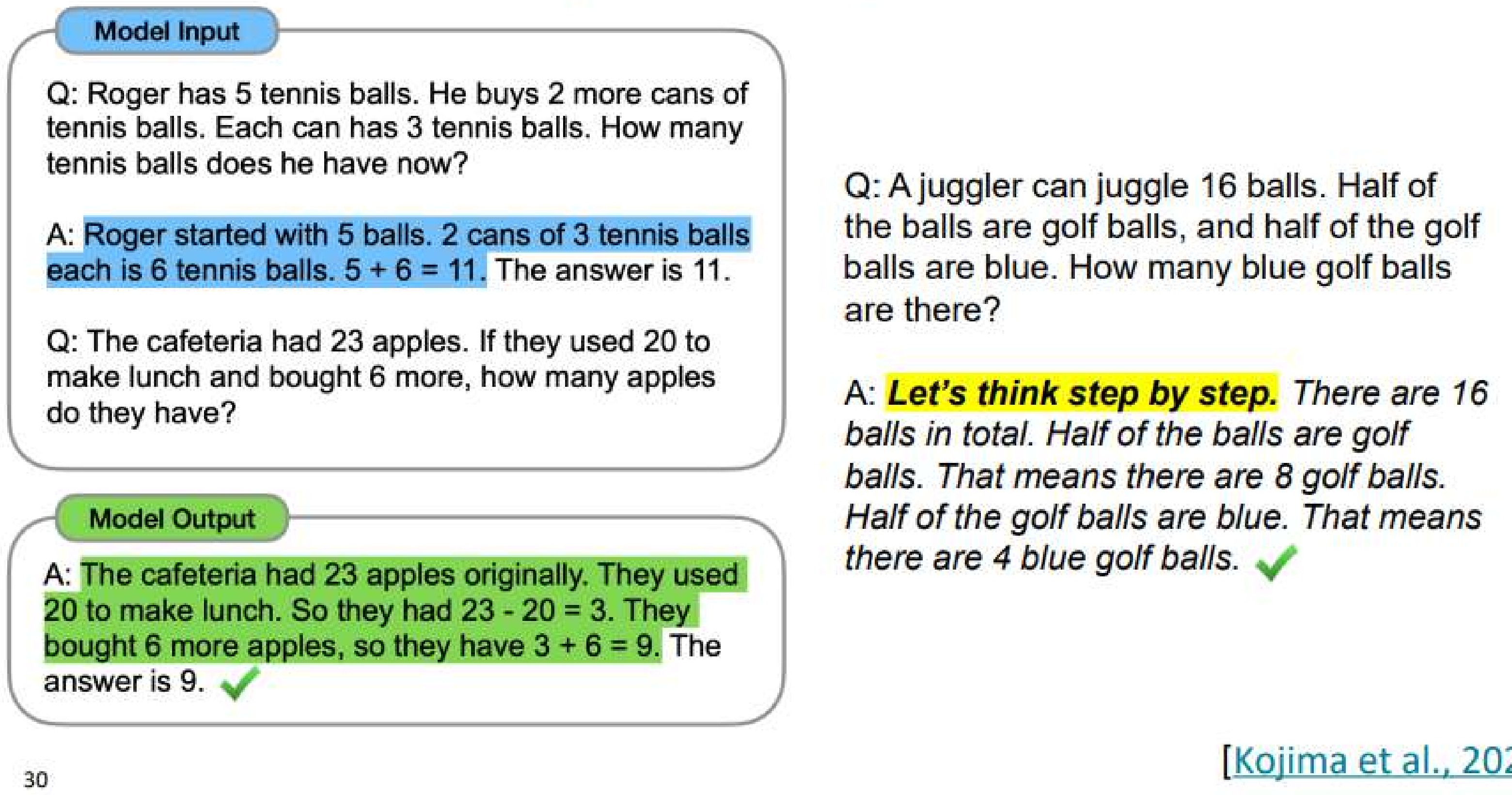
[Wei et al., 2022; also see Nye et al., 2021]

20기 경민수

1. Zero-Shot(ZS) and Few-Shot (FS) In-Context Learning

Zero-shot chain-of-thought prompting

- Just ask it nicely, don't need examples of reasoning answer.



[Kojima et al., 2022]

The new dark art(흑마법) of "prompt engineering"

- Asking a model for reasoning
- "Jailbreaking" LMs
- Making art picture
- professional or bug free code generation

→ Hiring prompt engineer

- get crazy good accuracy in manual change of thought

	MultiArith	GSM8K
Zero-Shot	**17.7**	**10.4**
Few-Shot (2 samples)	33.7	15.6
Few-Shot (8 samples)	33.8	15.6
Zero-Shot-CoT	**78.7**	**40.7**
Few-Shot-CoT (2 samples)	84.8	41.3
Few-Shot-CoT (4 samples : First) (*1)	89.2	-
Few-Shot-CoT (4 samples : Second) (*1)	90.5	-
Few-Shot-CoT (8 samples)	93.0	48.7

Greatly outperforms zero-shot →

Manual CoT still better →

- zero-shot trigger prompt accuracy

No.	Category	Zero-shot CoT Trigger Prompt	Accuracy
1	LM-Designed	Let's work this out in a step by step way to be sure we have the right answer.	**82.0**
2	Human-Designed	Let's think step by step. (*1)	78.7
3		First, (*2)	77.3
4		Let's think about this logically.	74.5
5		Let's solve this problem by splitting it into steps. (*3)	72.2
6		Let's be realistic and think step by step.	70.8
7		Let's think like a detective step by step.	70.3
8		Let's think	57.5
9		Before we dive into the answer,	55.7
10		The answer is after the proof.	45.7
-		(Zero-shot)	17.7

20기 경민수

Lecture Plan : From Language Models to Assistants

1. Zero-Shot (ZS) and Few-Shot (FS) In-Context Learning

2. Instruction finetuning

3. Reinforcement Learning from Human Feedback (RLHF)

4. What's next?

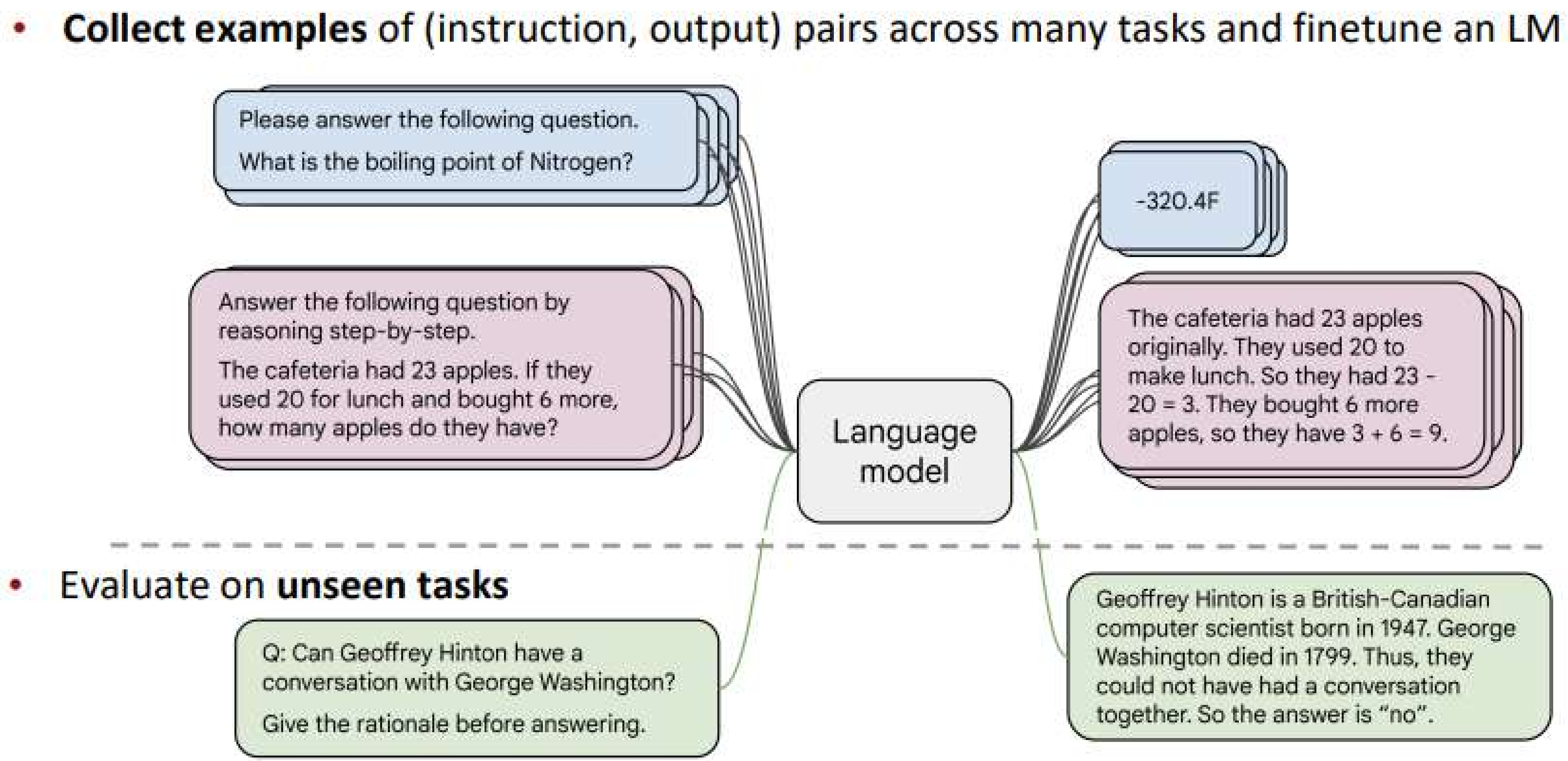
2. Instruction finetuning

Language modeling != assisting users

- language models are trained to predict the most likely continuation of tokens.
this is not the same as what we want language models to do.
- Language models are not aligned with user intent (언어 모델은 사용자의 목적에 맞게 설계되지 않음.)

∴ Finetuning to the rescue

Instruction finetuning



[FLAN-T5; Chung et al., 2022]

- 데이터 + 모델 스케일 크기가 finetuning의 키포인트.
- ex) the Super-NaturalInstrucitons dataset contains over 1.6K tasks, 3M+ examples
  - Classification, sequence tagging, rewriting, translation, QA ...

- Q : how do we evaluate such a model?

언어 모델을 평가하는 방법(benchmarking)

20기 경민수

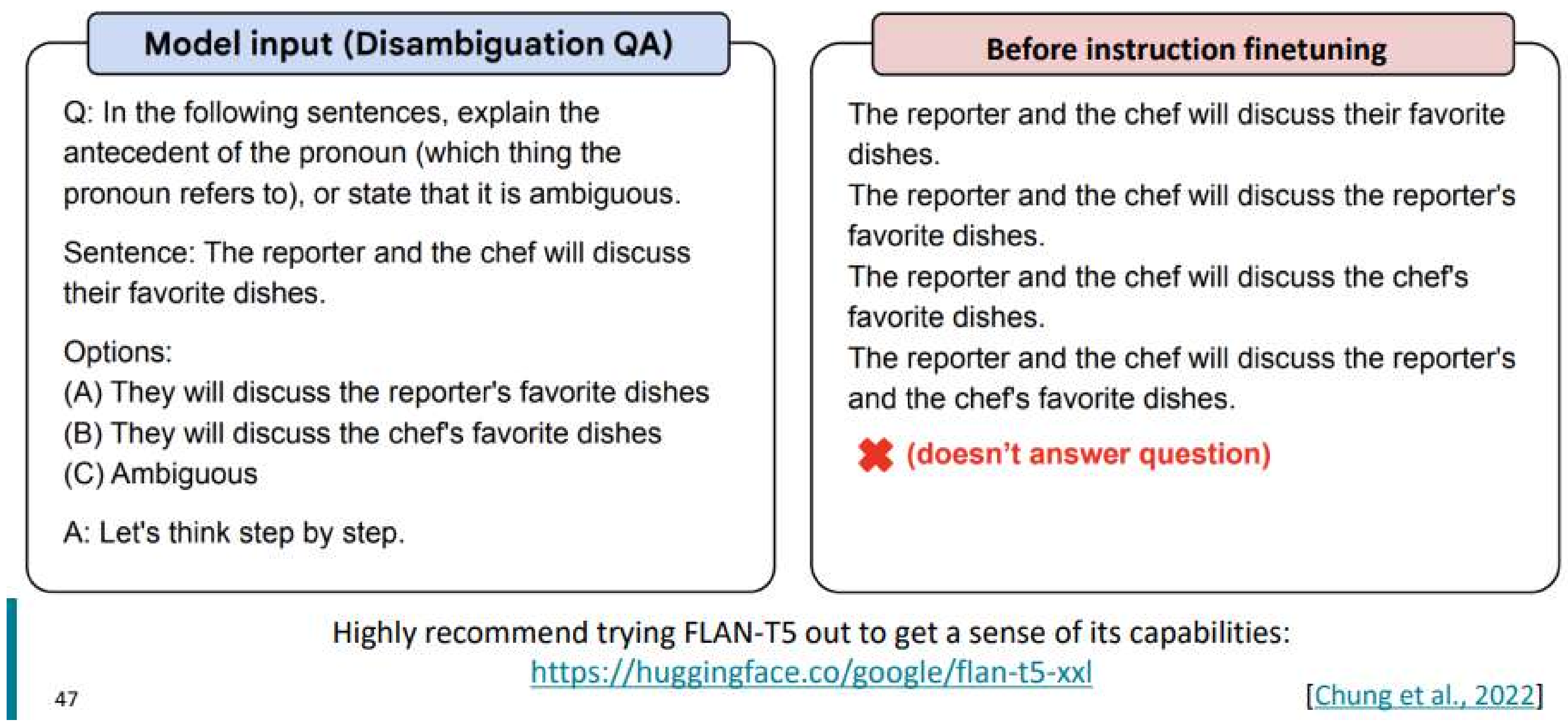
2. Instruction finetuning

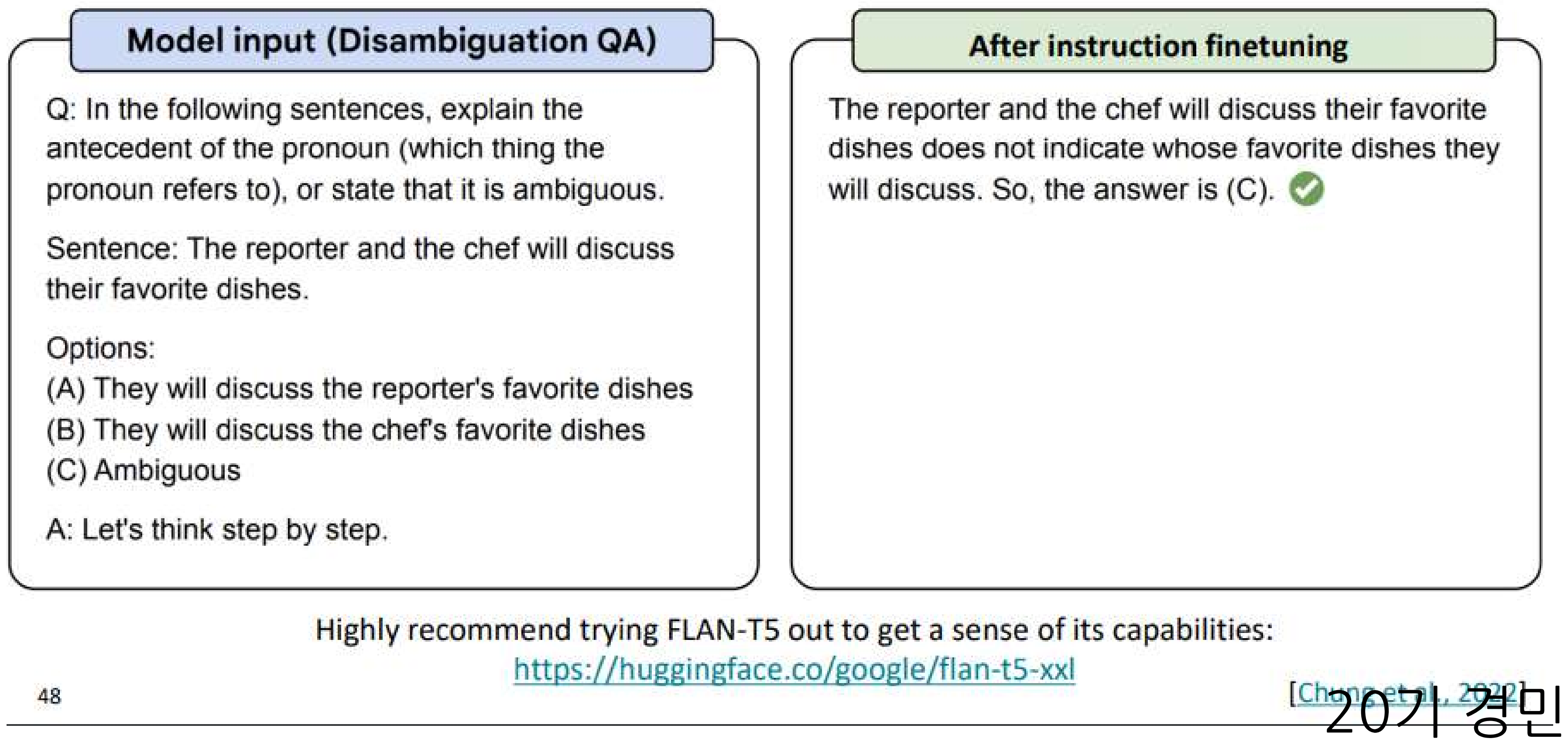
benchmarking methods

- Massive Multitask Language Understanding (MMLU) [Hendrycks et al., 2021]
  - New benchmarks for measuring LM performance on 57 diverse knowledge intensive tasks

- BIG-Bench [Srivastava et al., 2022]
  - 200+ tasks

The bigger the model, the bigger the benefit that you get from doing finetuning on benchmarking metrics.
-> Sad for academics or anyone without a massive GPU cluster





20기 경민수

2. Instruction finetuning

Limitations of instruction finetuning

- Obvious : it's expensive to collect gound-truth data for tasks.

- Problem 1 : tasks like open-ended creative generation have no right answer.
  - Write me a story about a dog and her pet grasshopper.
  - (instruction, output)의 pair를 이용할 수 없음.

- Problem 2 : language modeling penalizes all token-level mistakes equally, but some errors are worse than others.
  - 모든 error의 가중치를 평등하게 부여함. 문제의 우선순위 존재 X.

- Even with instruction finetuning, there a mismatch between the LM objective and the objective of "satisfy human preferences" !
  - 여전히 언어 모델과 인간의 목적에 차이가 존재함.

- Can we explicitly attempt to satisfy human preferences?
  - 언어 모델이 인간의 목적을 완전히 충족시킬 수 있는가?

20기 경민수

Lecture Plan : From Language Models to Assistants

1. Zero-Shot (ZS) and Few-Shot (FS) In-Context Learning
2. Instruction finetuning
3. Reinforcement Learning from Human Feedback (RLHF)
4. What's next?

20기 경민수

3. Reinforcement Learning from Human Feedback (RLHF)

Optimizing for human preferences

- Training a language model on some task (e.g. summarization).
- For each LM sample s, imagine we had s way to obtain a human reward of that summary, higher is better.

- For each LM sample s, imagine we had a way to obtain a *human reward* of that summary: $R(s) \in \mathbb{R}$, higher is better.

SAN FRANCISCO, California (CNN) -- A magnitude 4.2 earthquake shook the San Francisco ... overturn unstable objects.	An earthquake hit San Francisco. There was minor property damage, but no injuries.	The Bay Area has good weather but is prone to earthquakes and wildfires.
	s_1	s_2
	$R(s_1) = 8.0$	$R(s_2) = 1.2$

- Now we want to maximize the expected reward of samples from our LM:

$$\mathbb{E}_{\hat{s} \sim p_\theta(s)}[R(\hat{s})]$$

Note: for mathematical simplicity we're assuming only one "prompt"

- 각 대답에 사람이 직접 가중치를 부여하여 학습을 진행.
- 더욱 "그럴싸"한 대답에 큰 가중치 부여

3. Reinforcement Learning from Human Feedback (RLHF)

Reinforcement learning to the rescue

- The field of reinforcement learning (RL) is studies these (and related) problems for many years now
- But the interest in applying RL to modern LMs is an even newer phenomenon. **Why?**
  - RL w/ LMs has commonly been viewed as very hard to get right (still is!)
  - Newer advances in RL algorithms that work for large neural models, including language models

Policy gradient method

- How do we actually change our LM parameters theta to maximize this? $\mathbb{E}_{\hat{s} \sim p_\theta(s)}[R(\hat{s})]$
- Let's try doing gradient ascent!

$$\theta_{t+1} := \theta_t + \alpha \nabla_{\theta_t} \mathbb{E}_{\hat{s} \sim p_{\theta_t}(s)}[R(\hat{s})]$$

How do we estimate this expectation??

What if our reward function is non-differentiable??

  - 목적함수의 최댓값을 찾기 위한 gradient update 시행

3. Reinforcement Learning from Human Feedback (RLHF)

A brief introduction to policy gradient/REINFORCE

- We want to obtain

(defn. of expectation) (linearity of gradient)

$$\nabla_\theta \mathbb{E}_{\hat{s}\sim p_\theta(s)}[R(\hat{s})] = \nabla_\theta \sum_s R(s) p_\theta(s) = \sum_s R(s)\, \nabla_\theta p_\theta(s)$$

- Here we'll use a very handy trick known as the **log-derivative trick**. Let's try taking the gradient of $\log p_\theta(s)$

$$\nabla_\theta \log p_\theta(s) = \frac{1}{p_\theta(s)} \nabla_\theta p_\theta(s) \quad \Rightarrow \quad \nabla_\theta p_\theta(s) = \nabla_\theta \log p_\theta(s)\, p_\theta(s)$$

(chain rule)

- Plug back in:

This is an expectation of this

$$\sum_s R(s)\, \nabla_\theta p_\theta(s) = \sum_s p_\theta(s) R(s)\, \nabla_\theta \log p_\theta(s)$$

$$\boxed{= \mathbb{E}_{\hat{s}\sim p_\theta(s)}[R(\hat{s})\, \nabla_\theta \log p_\theta(\hat{s})]}$$

3. Reinforcement Learning from Human Feedback (RLHF)

TOBIG'S

A brief introduction to policy gradient/REINFORCE

- Now we have put the gradient "inside" the expectation, we can approximate this objective with Monte Carlo samples:

$$\nabla_\theta \mathbb{E}_{\hat{s}\sim p_\theta(s)}[R(\hat{s})] = \mathbb{E}_{\hat{s}\sim p_\theta(s)}[R(\hat{s})\,\nabla_\theta \log p_\theta(\hat{s})] \approx \frac{1}{m}\sum_{i=1}^{m} R(s_i)\,\nabla_\theta \log p_\theta(s_i)$$

This is why it's called **"reinforcement learning"**: we **reinforce** good actions, increasing the chance they happen again.

If R is +++

Take gradient steps to maximize $p_\theta(s_i)$

- Giving us the update rule:

$$\theta_{t+1} := \theta_t + \alpha\frac{1}{m}\sum_{i=1}^{m} R(s_i)\,\nabla_{\theta_t} \log p_{\theta_t}(s_i)$$

If R is ---

Take steps to minimize $p_\theta(s_i)$

This is **heavily simplified**! There is a *lot* more needed to do RL w/ LMs. **Can you see any problems with this objective?**

3. Reinforcement Learning from Human Feedback (RLHF)

Adventage / disadventage of RLHF & Solution

- Adventage
  - any arbitrary, non-differentiable reward function -> maximize (O)
    : 보상 함수가 어떻게 주어지더라도(미분 불가능한 랜덤의 경우에도) 최댓값 구하기 가능.

- Disadventage (Not so fast)
  - human-in-the-loop is expensive (Problem 1)
    : 사람의 선호를 직접 조사하면서 학습을 진행하기에는 여러 어려움이 생긴다.
    - Solution : instead of directly asking humans for preferences, model their preferences as a separate (NLP) problem ! [Knox and Stone, 2009] (사람의 선호를 모델링하는 언어 모델을 따로 만들 것)

  - human judgements are noisy and miscalibrated (Problem 2)
    : 사람의 직접적인 판단에는 오차와 오지식이 많을 수 있다.
    - Solution : instead of asking for direct ratings, ask for pairwise comparisons, which can be more reliable [Phelps et al., 2015; Clark et al., 2018] (여러 요소의 직접적인 순위보다는 두 개의 요소의 비교를 통한 정확도 향상)

3. Reinforcement Learning from Human Feedback (RLHF)

Adventage / disadventage of RLHF & Solution

- Adventage
  - any arbitrary, non-differentiable reward function -> maximize (O)
    : 보상 함수가 어떻게 주어지더라도(미분 불가능한 랜덤의 경우에도) 최댓값 구하기 가능.

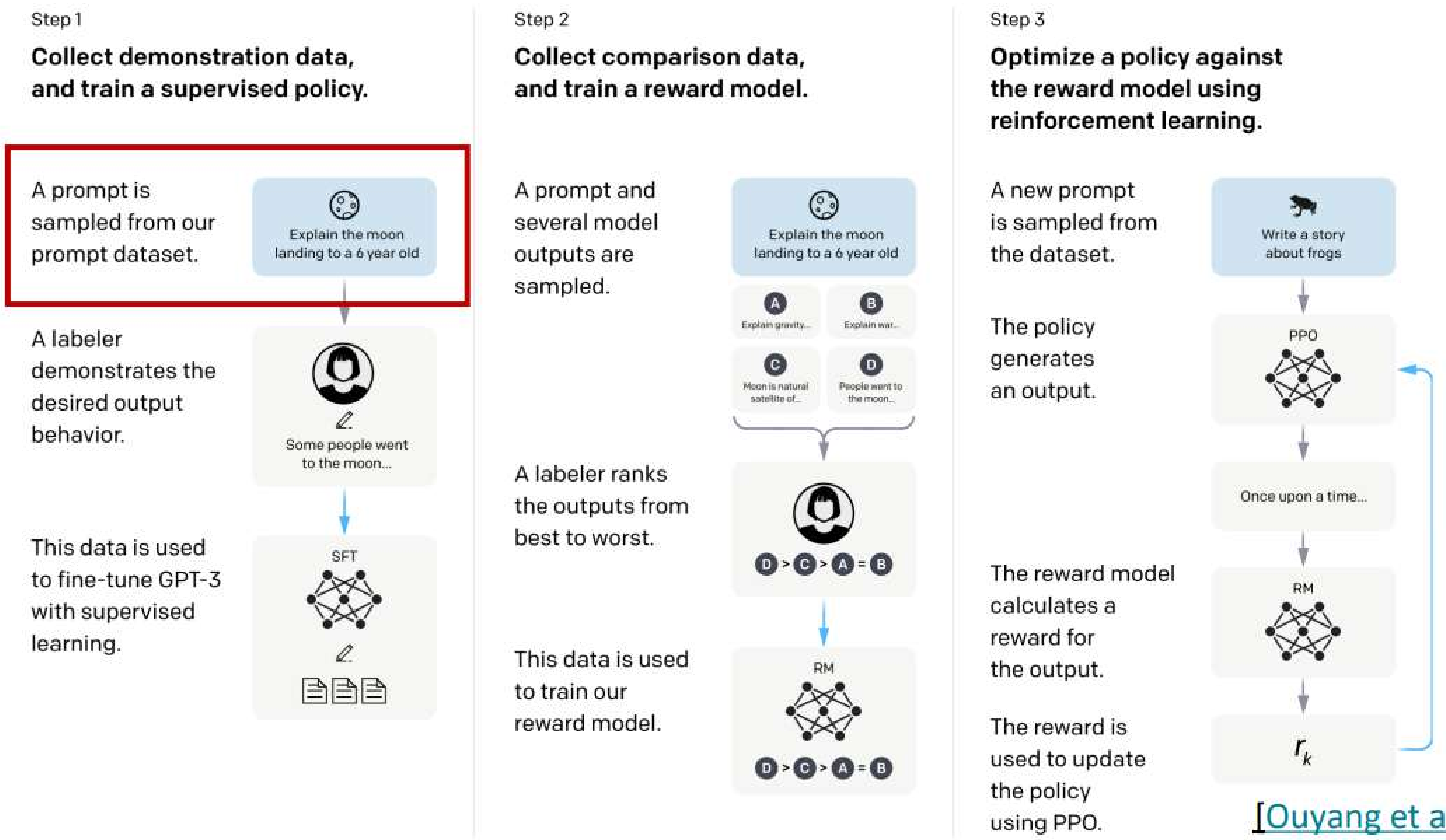
- Disadventage (Not so fast)
  - human-in-the-loop is expensive (Problem 1)
    : 사람의 선호를 직접 조사하면서 학습을 진행하기에는 여러 어려움이 생긴다.
    - Solution : instead of directly asking humans for preferences, model their preferences as a separate (NLP) problem ! [Knox and Stone, 2009] (사람의 선호를 모델링하는 언어 모델을 따로 만들 것)

  - human judgements are noisy and miscalibrated (Problem 2)
    : **사람의 직접적인 판단에는 오차와 오지식이 많을 수 있다.**
    - Solution : instead of asking for direct ratings, ask for pairwise comparisons, which can be more reliable [Phelps et al., 2015; Clark et al., 2018] (여러 요소의 직접적인 순위보다는 두 개의 요소의 비교를 통한 정확도 향상)

3. Reinforcement Learning from Human Feedback (RLHF)

InstructGPT : scaling up RLHF to tens of thousands of tasks

30k tasks!



[Ouyang et al., 2022]

PROMPT *Explain the moon landing to a 6 year old in a few sentences.*

COMPLETION GPT-3

```
Explain the theory of gravity to a 6 year old.

Explain the theory of relativity to a 6 year old in a few sentences.

Explain the big bang theory to a 6 year old.

Explain evolution to a 6 year old.
```

InstructGPT

```
People went to the moon, and they took pictures of what they saw, and
sent them back to the earth so we could all see them.
```

PROMPT *Write a short poem about a wise frog.*

COMPLETION GPT-3

```
Write a short story in which a character has two different names.

Write a short story in which you try to get something back that you
have lost.

Write a short story in which a character has a bad dream.
```

InstructGPT

```
The frog is a symbol of wisdom
He knows all the secrets of the world
He is a master of disguise
And a great teacher of life
He is a symbol of transformation
And the bringer of change
He is the frog who has seen it all
And knows the meaning of it all
```


20기 경민수

3. Reinforcement Learning from Human Feedback (RLHF)

ChatGPT : Instruction Finetuning + RLHF for dialog agents

- Instruction finetuning

Methods

We trained this model using Reinforcement Learning from Human Feedback (RLHF), using the same methods as InstructGPT, but with slight differences in the data collection setup. We trained an initial model using supervised fine-tuning: human AI trainers provided conversations in which they played both sides—the user and an AI assistant. We gave the trainers access to model-written suggestions to help them compose their responses. We mixed this new dialogue dataset with the InstructGPT dataset, which we transformed into a dialogue format.

  - fine tuning 과정에서 InstructionGPT와 데이터 수집 과정이 약간 다름.

- RLHF

Methods

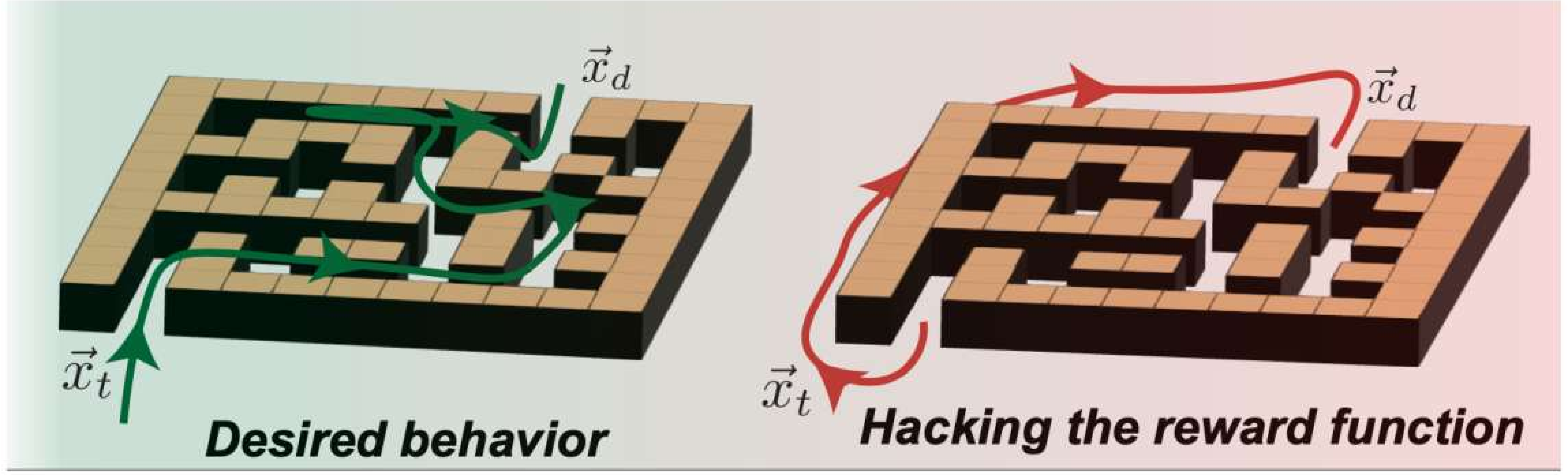
To create a reward model for reinforcement learning, we needed to collect comparison data, which consisted of two or more model responses ranked by quality. To collect this data, we took conversations that AI trainers had with the chatbot. We randomly selected a model-written message, sampled several alternative completions, and had AI trainers rank them. Using these reward models, we can fine-tune the model using Proximal Policy Optimization. We performed several iterations of this process.

  - InstructionGPT는 PPO를 완전 랜덤하게 초기화하는 반면, ChatGPT는 RLHF 과정의 보상(RM) 초기화 과정에서 Proximal Policy Optimization(PPO)를 supervised policy에 따라 초기화한다.

3. Reinforcement Learning from Human Feedback (RLHF)

Limitations of RL + Reward Modeling

- Human preferences are unreliable!
  - "Reward hackig" is a common problem in RL



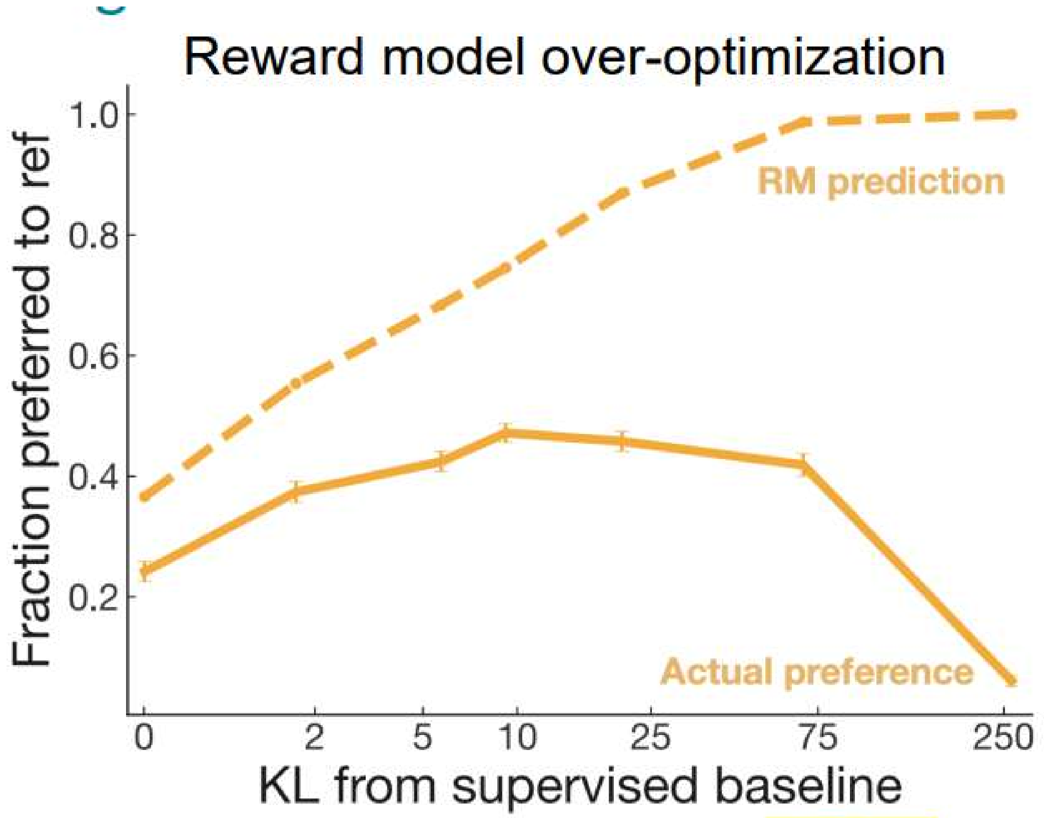
$$r(s_t, a_t) = -\|\vec{x}_t - \vec{x}_d\|^2$$

(Reward is a form of "*Minimize distance to goal*")

- Hallucination
  - result in making up facts + hallucinations
  - Why?

  : Chatbots are rewarded to produce responses
  that seem authoritative and helpful, *regardless of truth*
  진실의 전달 유무보다는 답변의 유창함에 초점을 둠.

- Models of human preferences are even more unreliable!



$$R(s) = RM_\phi(s) - \beta \log\left(\frac{p_\theta^{RL}(s)}{p^{PT}(s)}\right)$$

[Stiennon et al., 2020]

Lecture Plan : From Language Models to Assistants

1. Zero-Shot (ZS) and Few-Shot (FS) In-Context Learning

2. Instruction finetuning

3. Reinforcement Learning from Human Feedback (RLHF)

4. What's next?

4. What's next?

- RLHF is still a very underexplored and fast-moving area
- RLHF get you further than instruction finetuning, but is data expensive.
- Recent work aims to alleviate such data requirements :
  - RL from AI feedback [Bai et al., 2022]
  - Finetuning LMs on their own outputs [Huang et al., 2022; Zelikman et al., 2022]
- However, there are still many limitations of large LMs (size, hallucination) that may not solvable with RLHF

Reference

https://heegyukim.medium.com/large-language-model%EC%9D%98-scaling-law%EC%99%80-emergent-ability-6e9d90813a87
https://transipoyo.com/%ec%9d%b8%ea%b3%b5-%ec%a7%80%eb%8a%a5%ec%9d%98-emergent-ability%eb%9e%80/
https://arxiv.org/pdf/2206.07682.pdf
https://smilegate.ai/2021/09/12/instruction-tuning-flan/
https://seokhee0516.tistory.com/113
https://aihub.org/2022/05/31/designing-societally-beneficial-reinforcement-learning-systems/