

Basic of NLP

Basic nlp concept is all you need

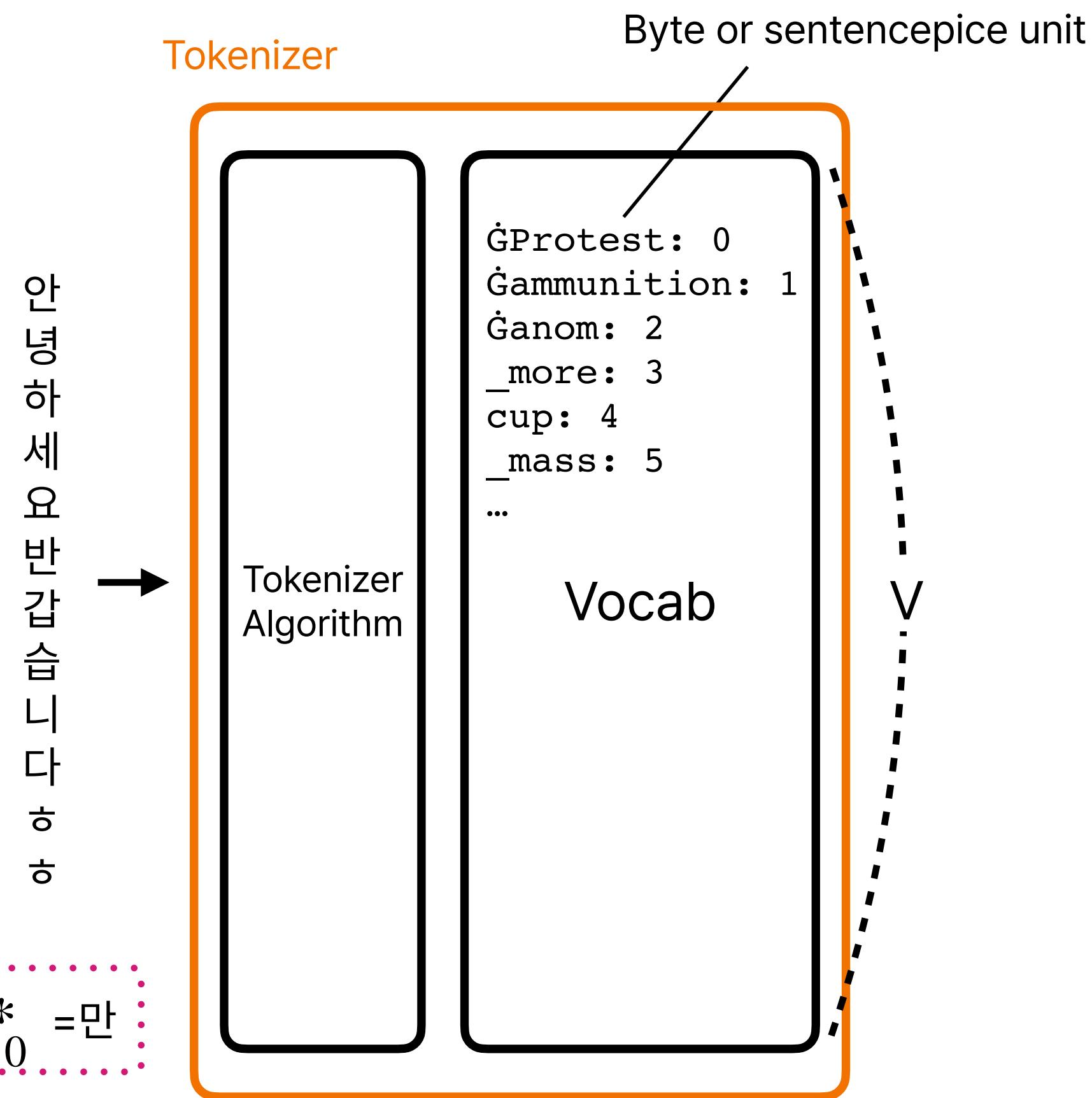
GLORANG AI Team - Popul

Basic of Language Model

Basic of Language Model

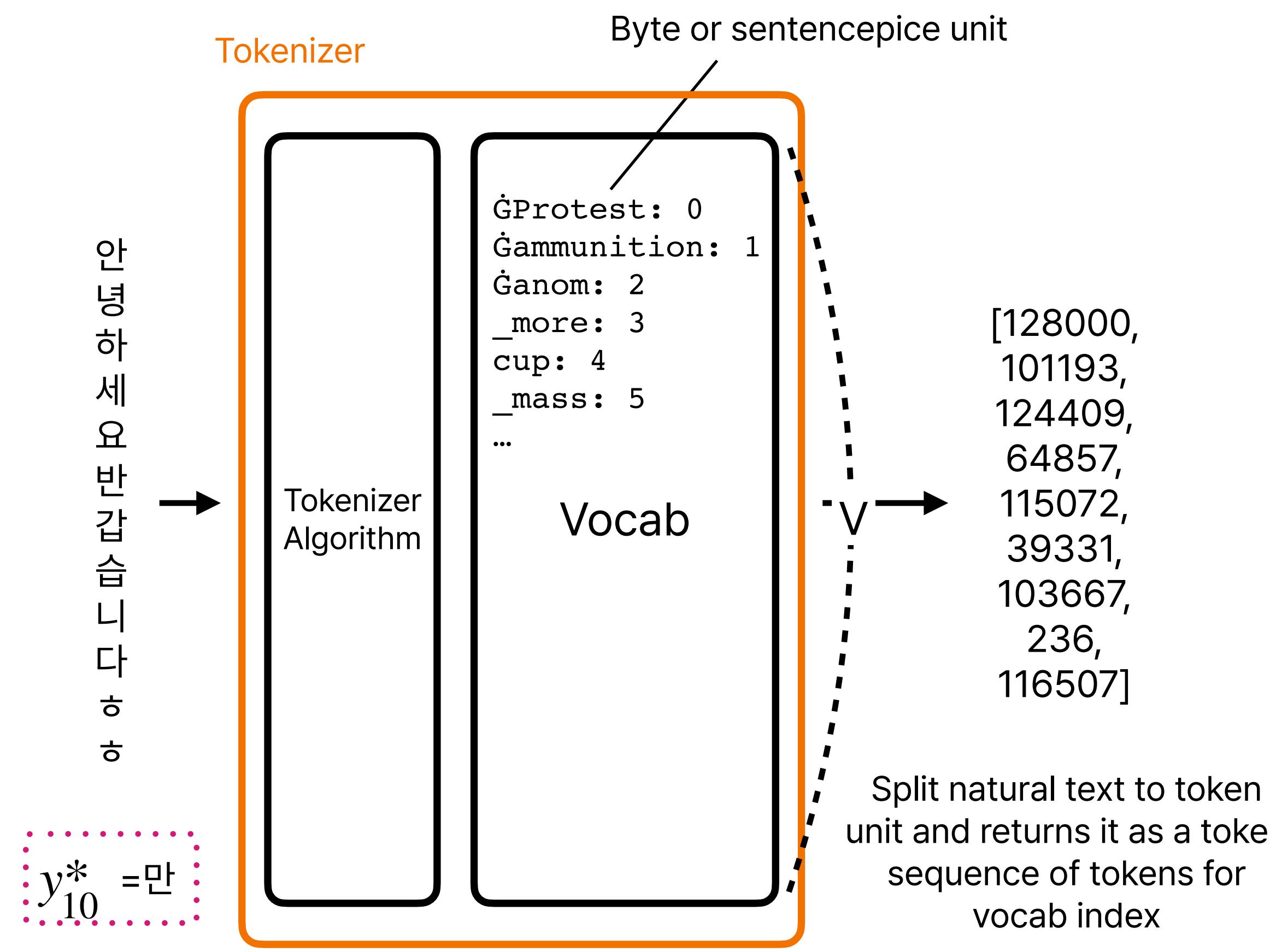
Input → Tokenizer → Embedding → Deep Neural Network → Loss → Update

전체 문장: 안녕하세요 반갑습니다 ㅎ ㅎ 만나서 반가워요



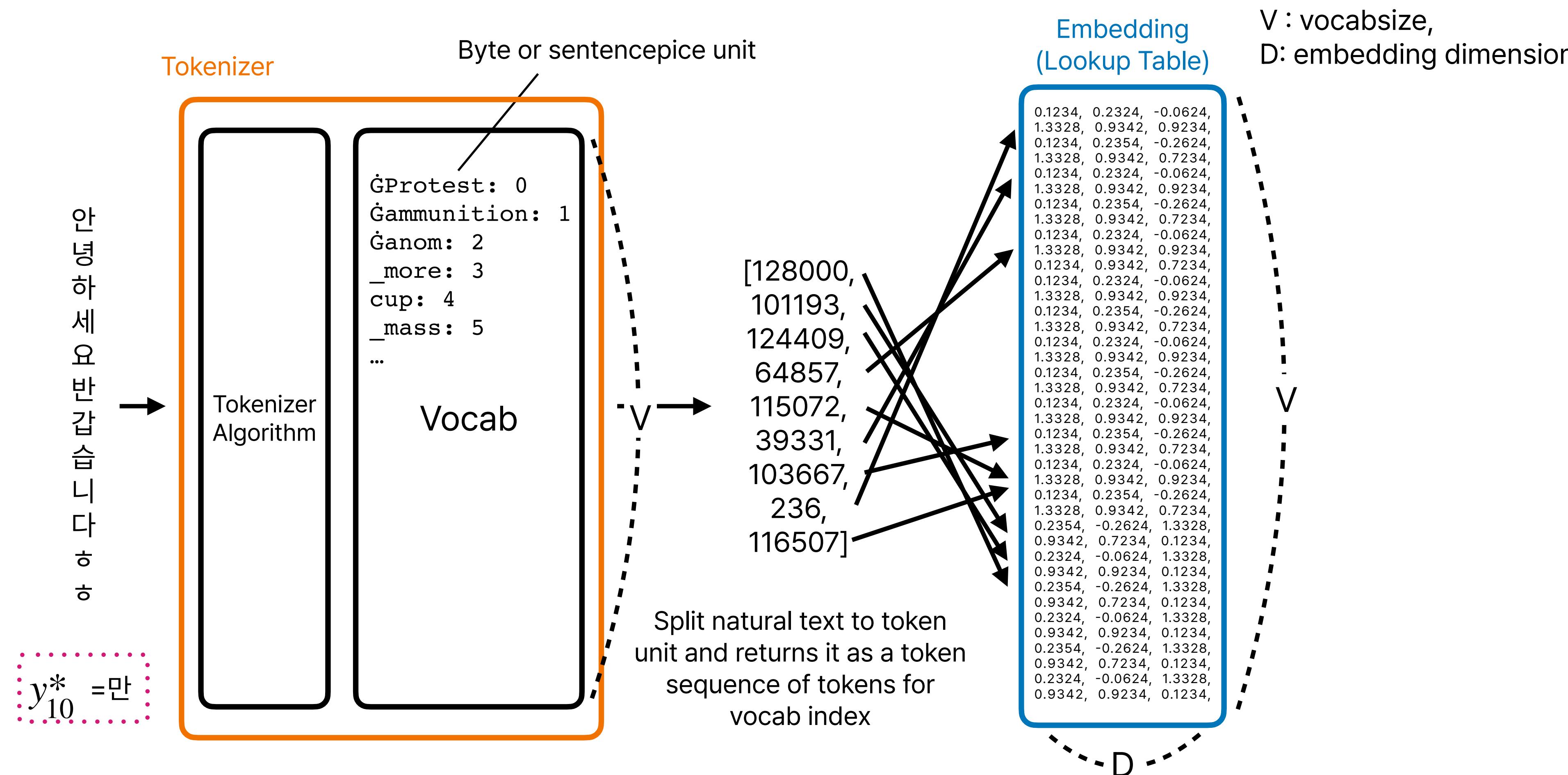
Basic of Language Model

Input → Tokenizer → Embedding → Deep Neural Network → Loss → Update



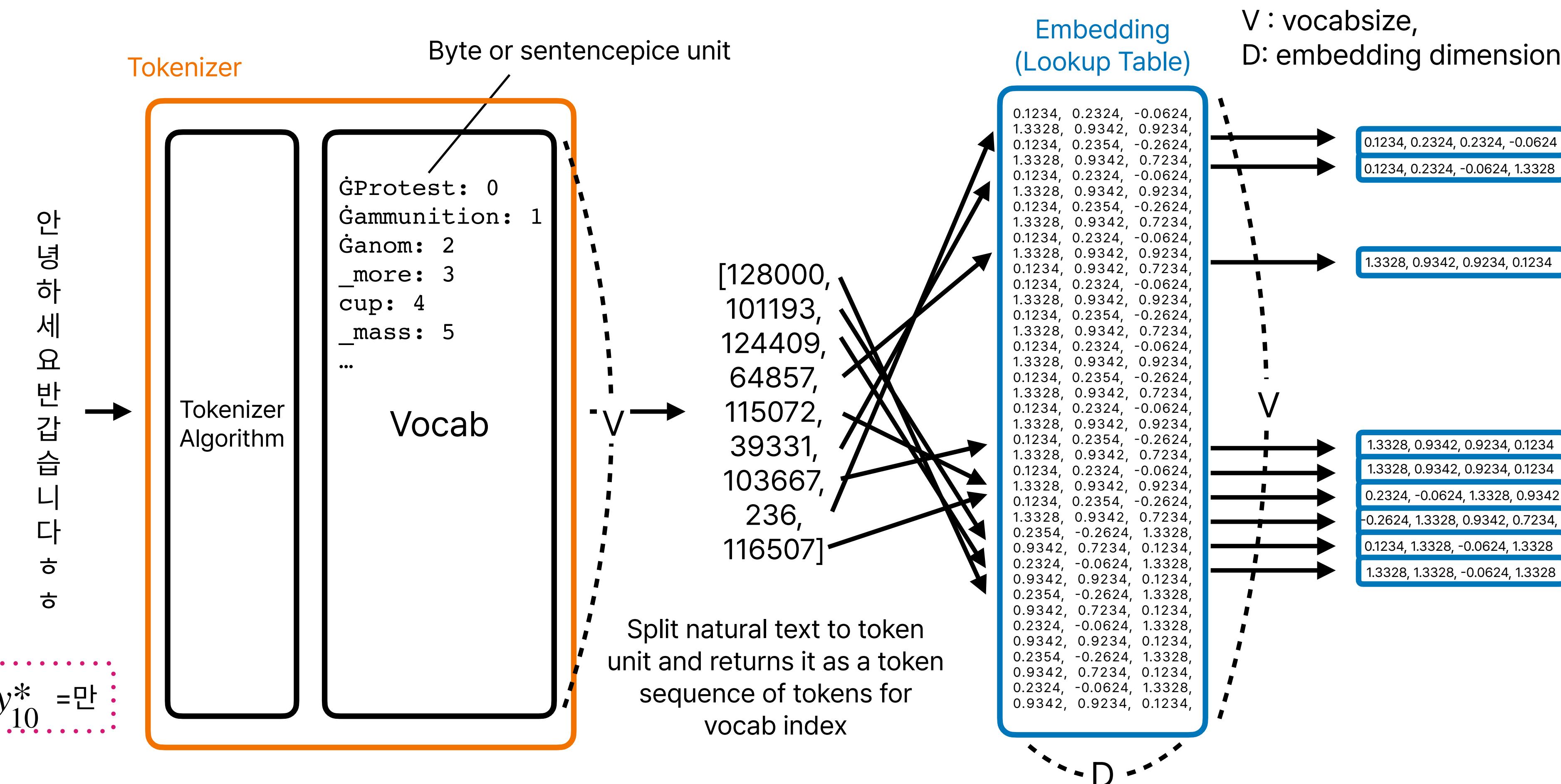
Basic of Language Model

Input → Tokenizer → Embedding → Deep Neural Network → Loss → Update



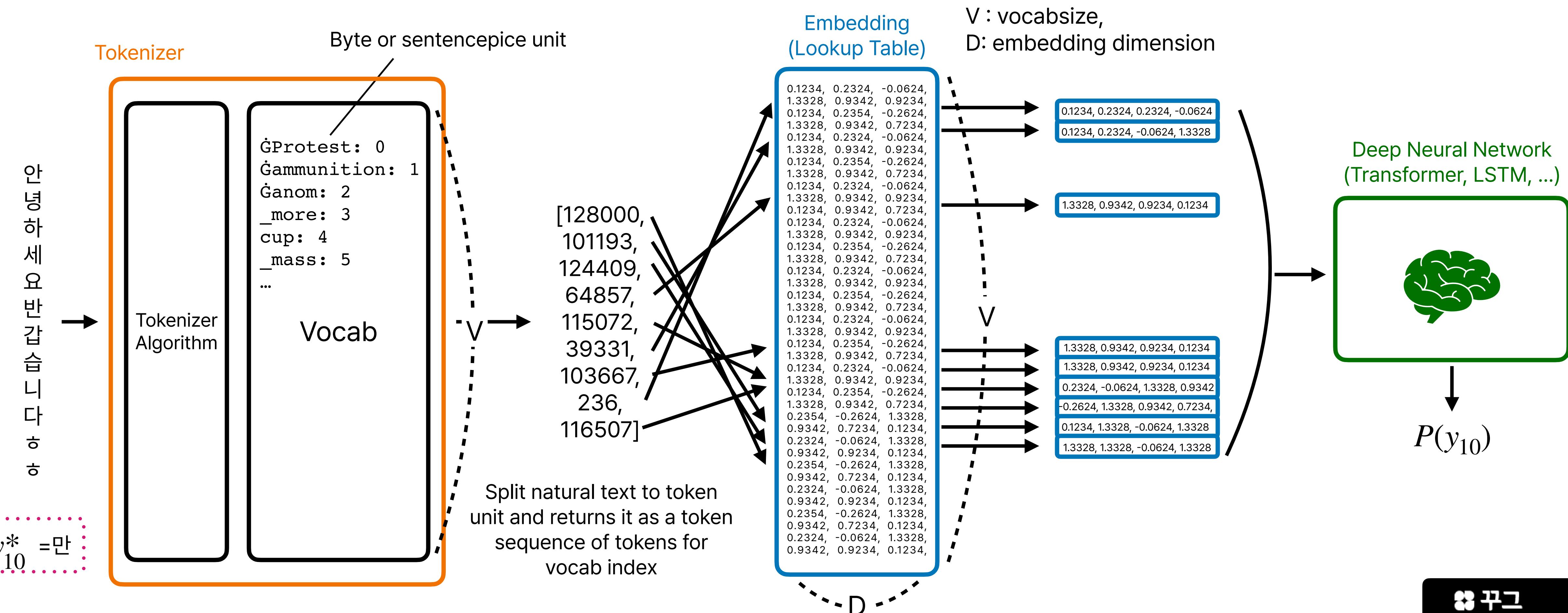
Basic of Language Model

Input → Tokenizer → Embedding → Deep Neural Network → Loss → Update



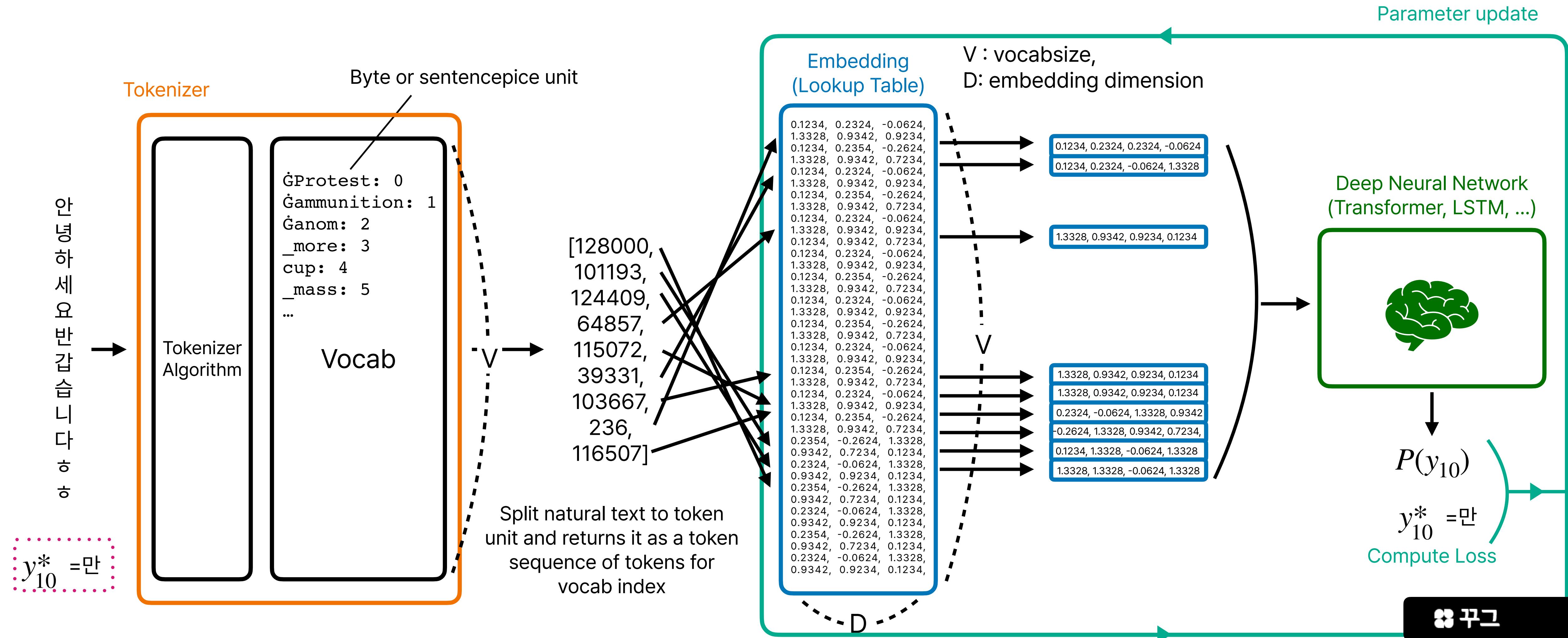
Basic of Language Model

Input → Tokenizer → Embedding → Deep Neural Network → Loss → Update



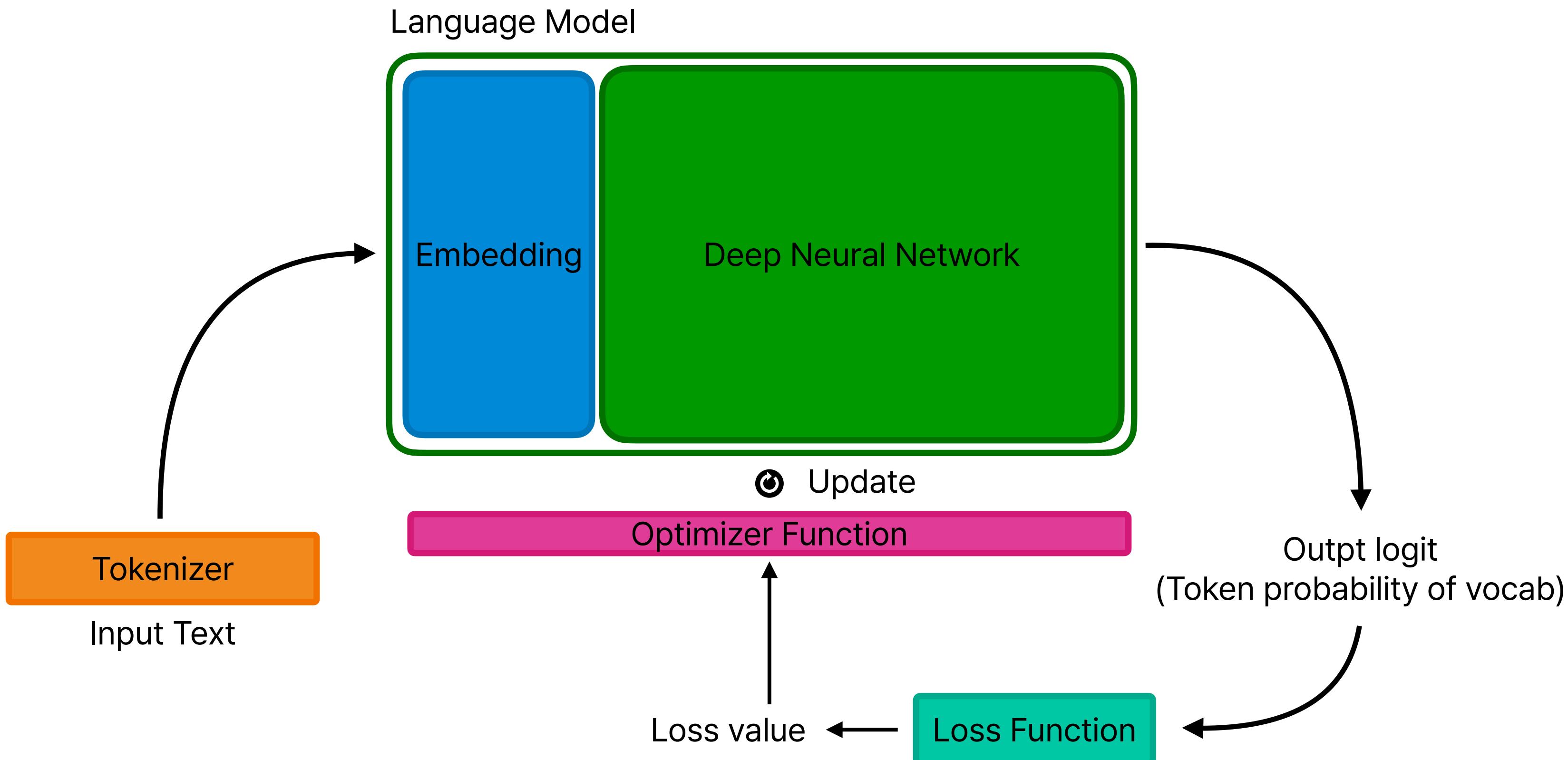
Basic of Language Model

Input → Tokenizer → Embedding → Deep Neural Network → Loss → Update

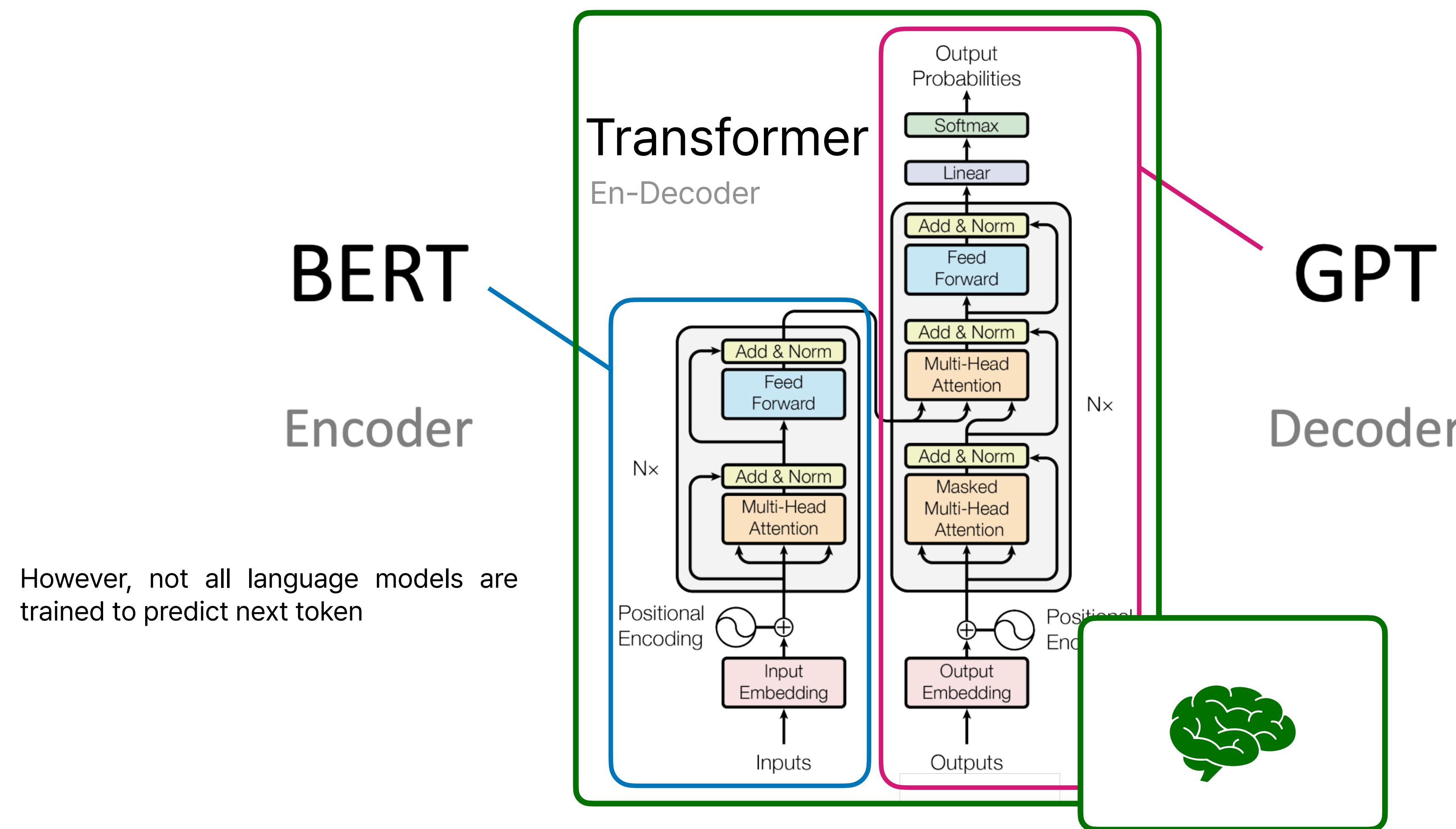


Basic of Language Model

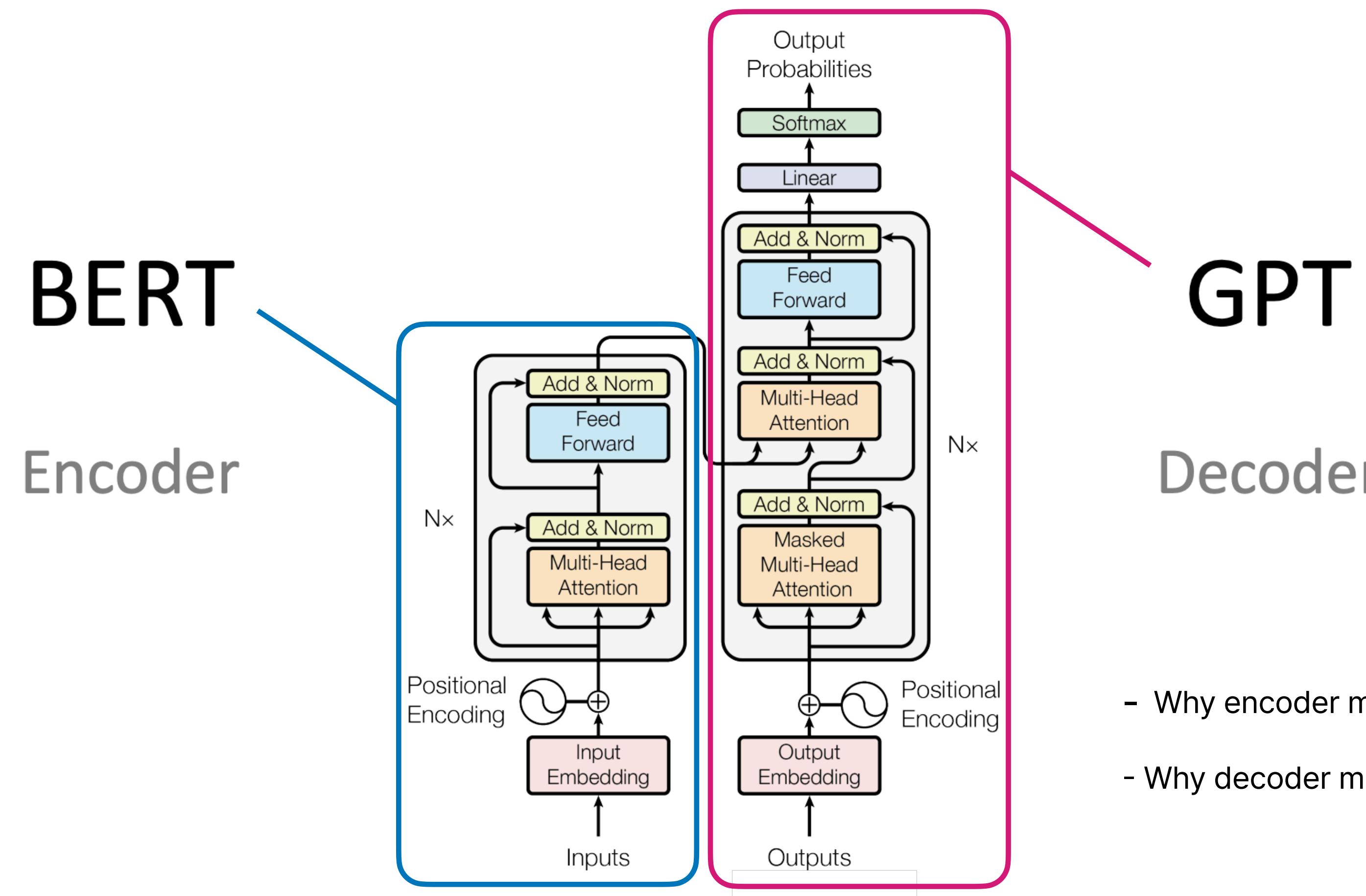
Components



Basic of Language Model



Basic of Language Model



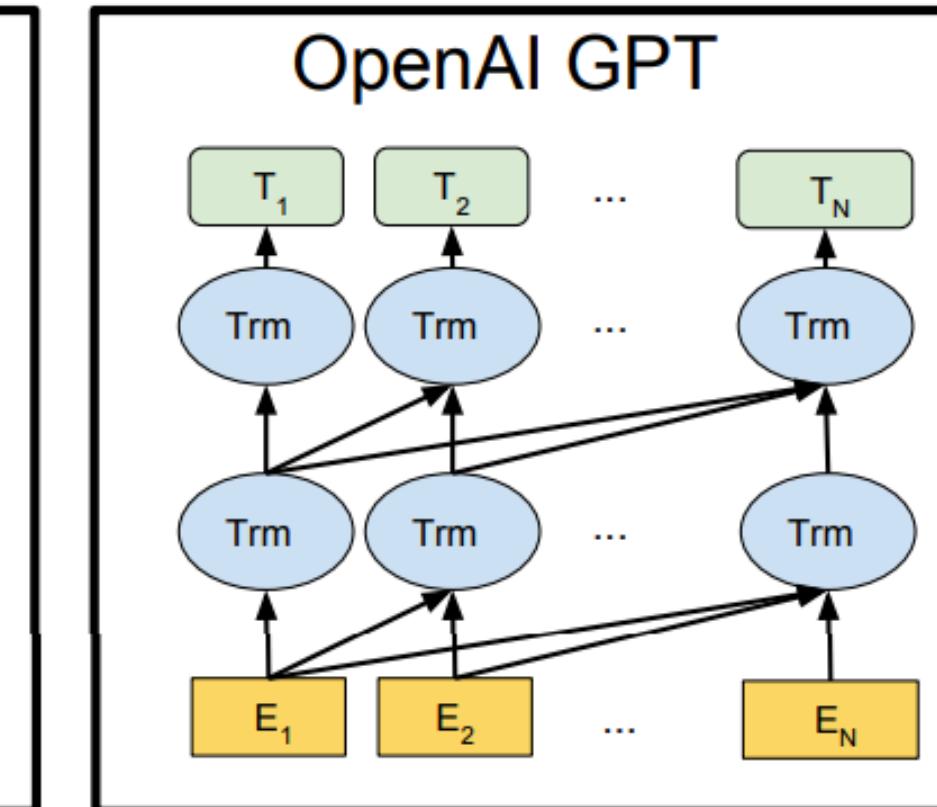
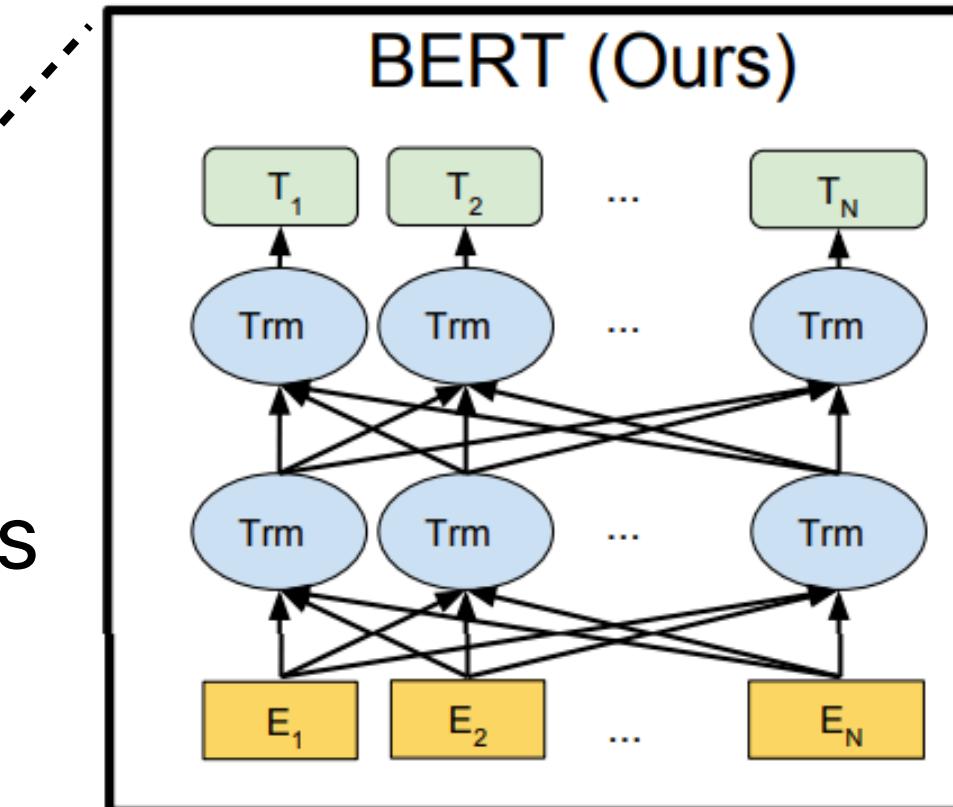
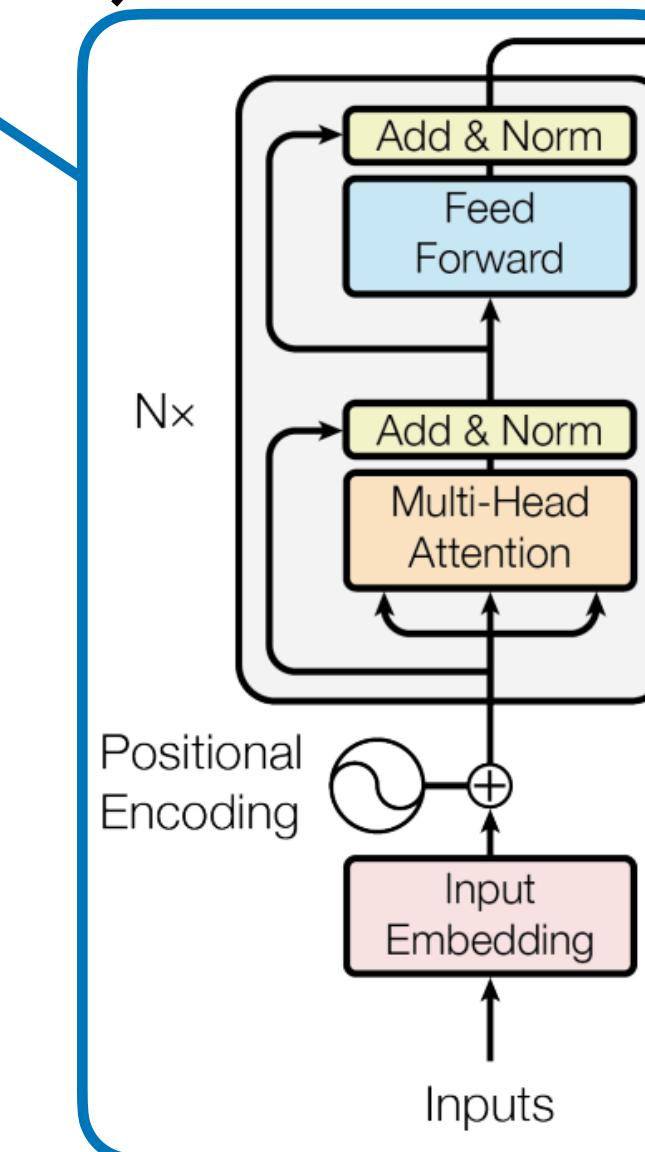
- Why encoder model is good at creating embedding?
- Why decoder model is good at text generation?

Basic of Language Model

Bidirectional Encoder Representations from Transformers

BERT

Encoder



BERT processes differently a input sequence ↪

BERT is trained by MLM method that is denoising masked token to original token

- 나는 [MASK]를 먹었다.
1. 밥
2. 자동차
3. 컴퓨터

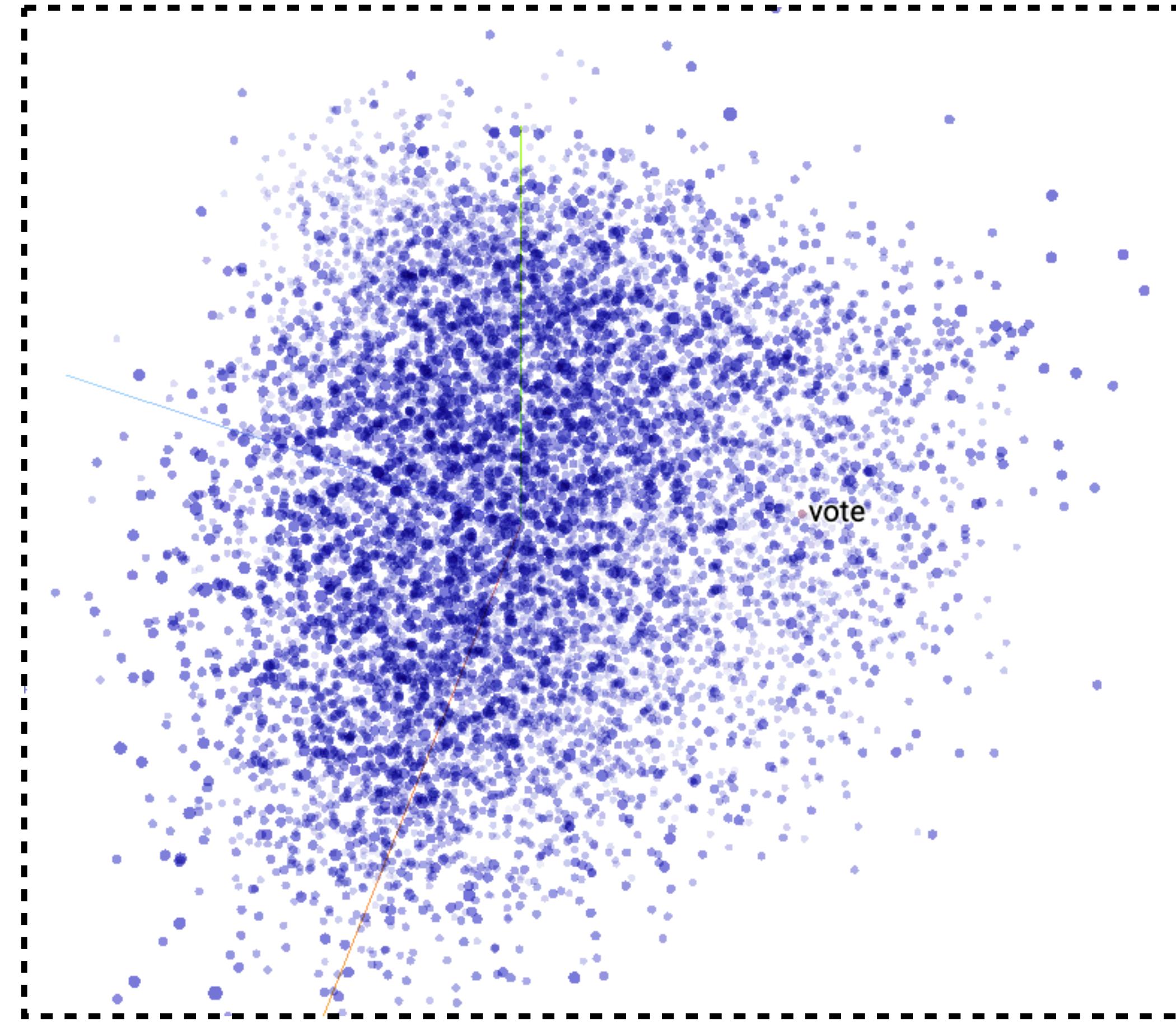
Finally, encoder models (like BERT, RoBERTa, ELECTRA ...) are good performance at many natural language understanding tasks.

[Attention!]

Encoder models are not created for text generation. Therfore, the output might not be what you expect.

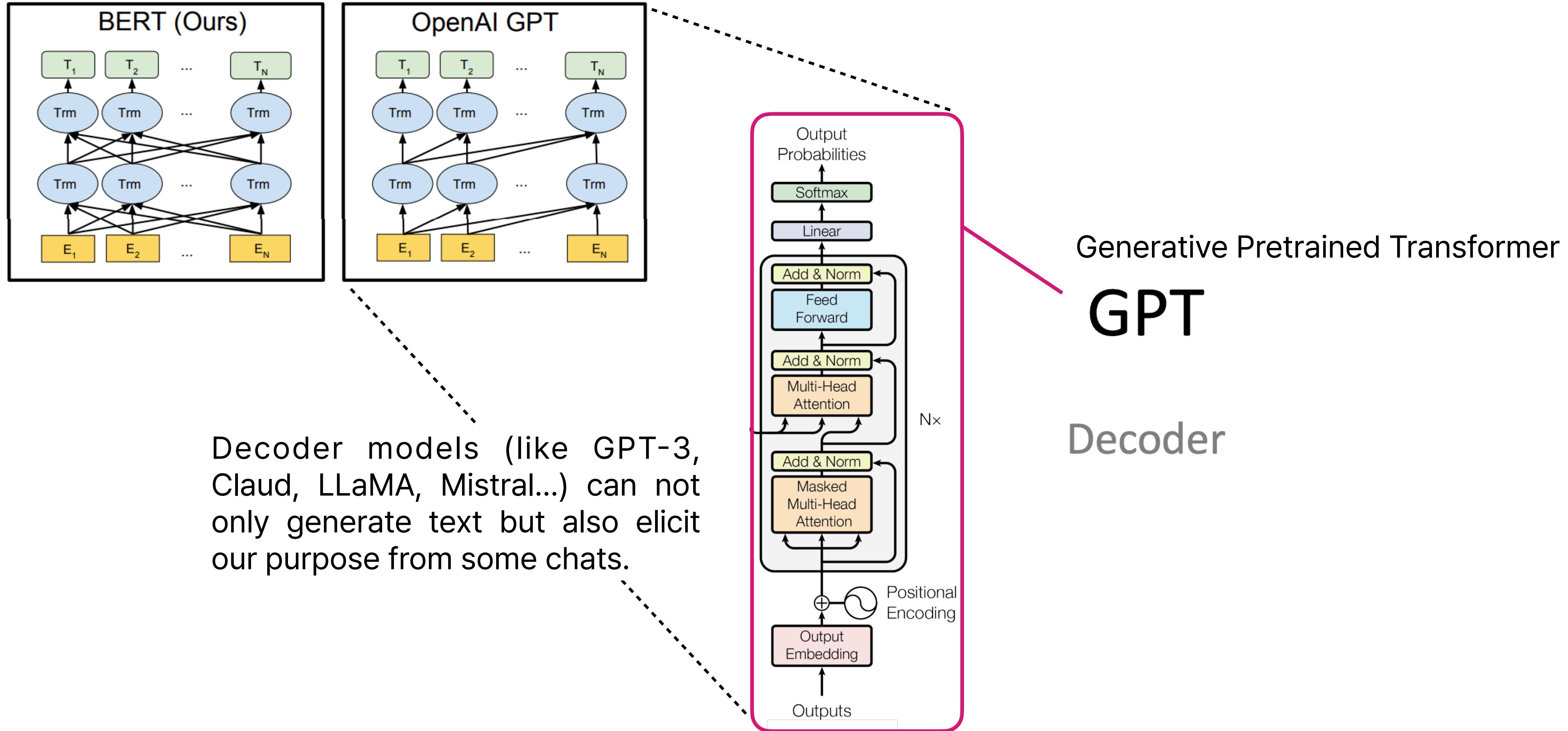
Basic of Language Model

Word embedding visualization



<https://projector.tensorflow.org/>

Basic of Language Model

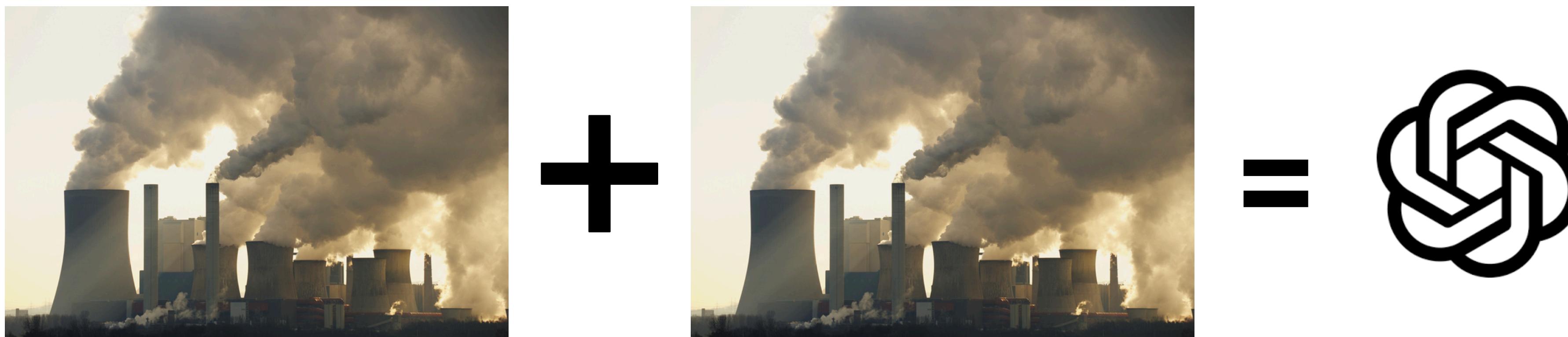


Basic of Language Model

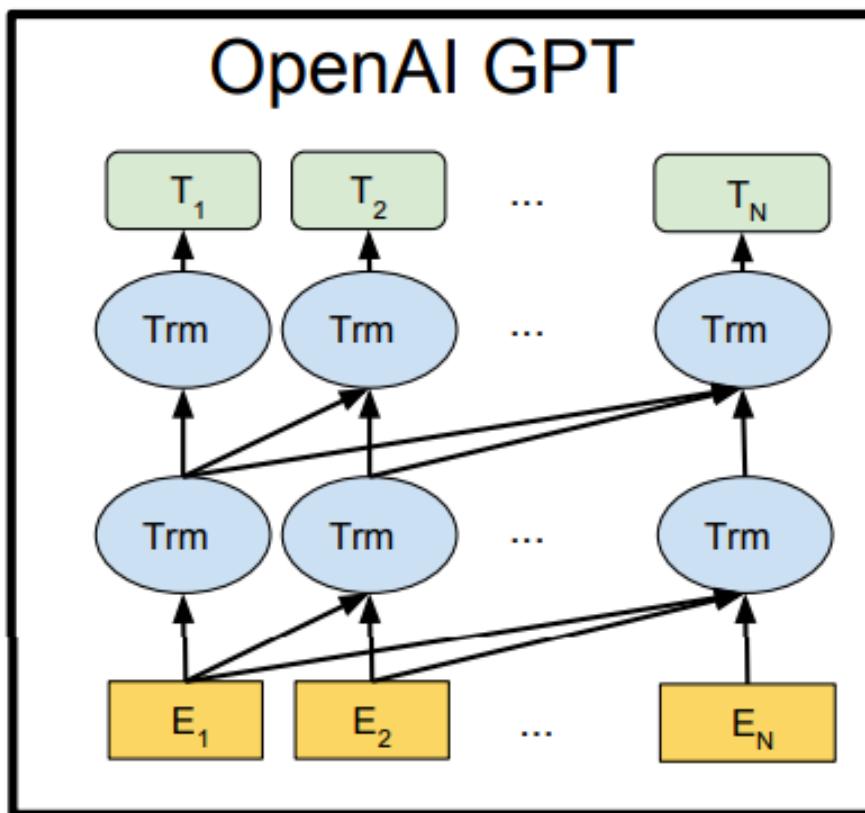
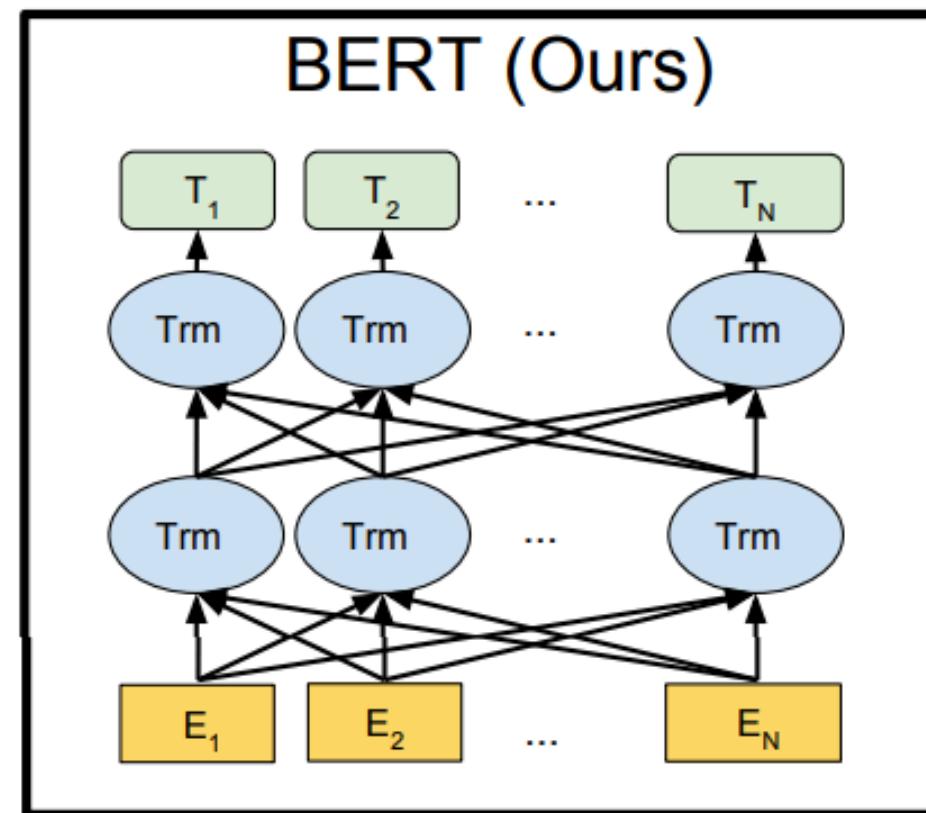
AI programs	power consumption
Bing	7,200 MWh
ChatGPT-3	1,248 MWh
Jasper, YouChat	1,248 MWh each
Google bard	312 MWh
Socratic	50 - 300 MWh

4000 x

Average monthly electricity usage for a household: 300KWh

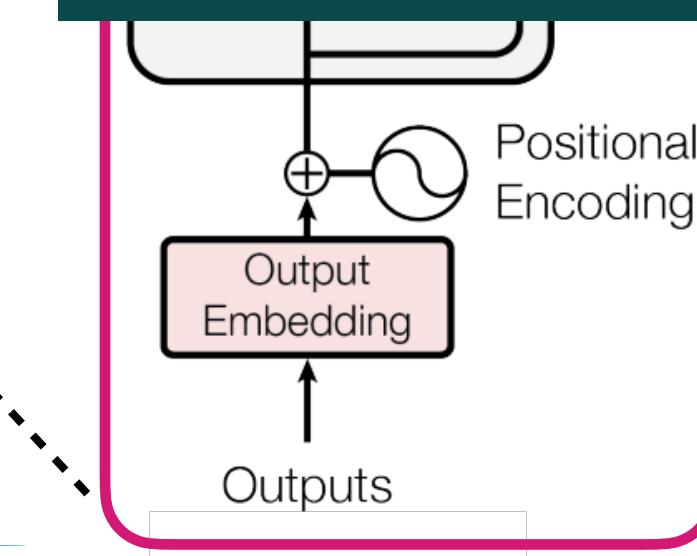
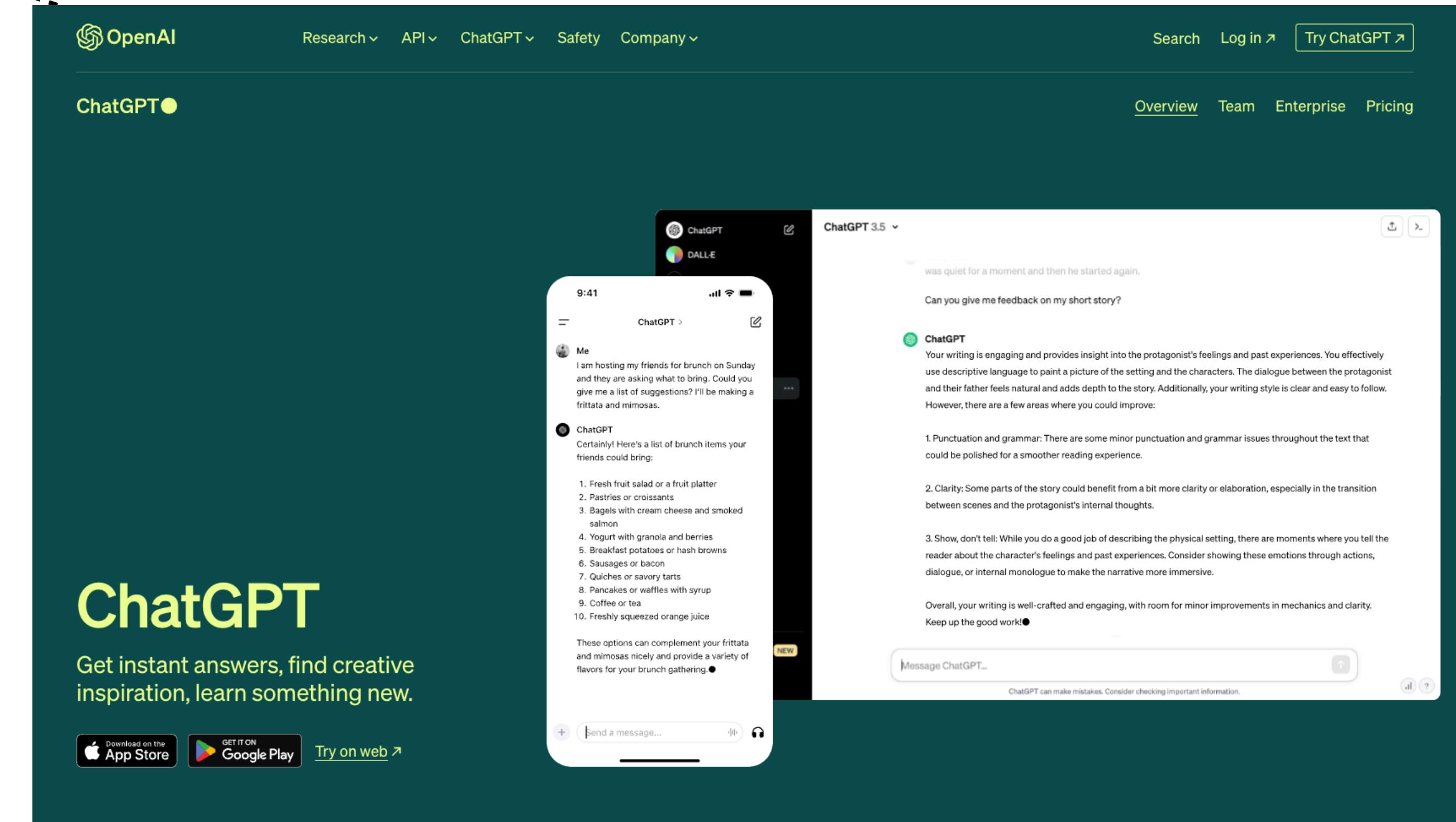


Basic of Language Model



Decoder models (like GPT-3, Claud, LLaMA, Mistral...) can not only generate text but also elicit our purpose from some chats.

Alignment Training



Basic of Language Model

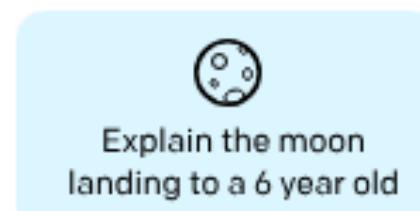
Alignment Training

RLHF

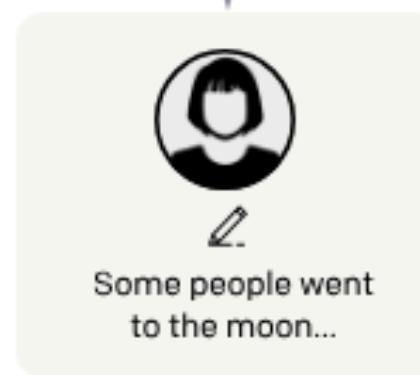
Step 1

Collect demonstration data, and train a supervised policy.

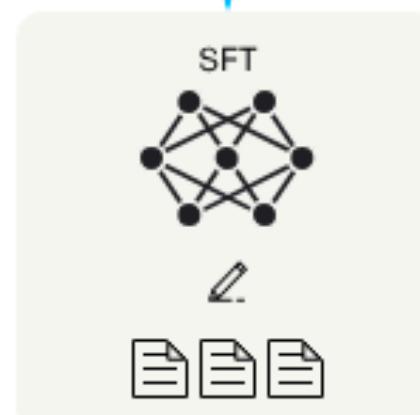
A prompt is sampled from our prompt dataset.



A labeler demonstrates the desired output behavior.



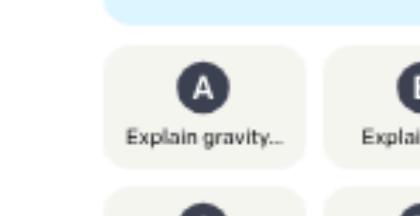
This data is used to fine-tune GPT-3 with supervised learning.



Step 2

Collect comparison data, and train a reward model.

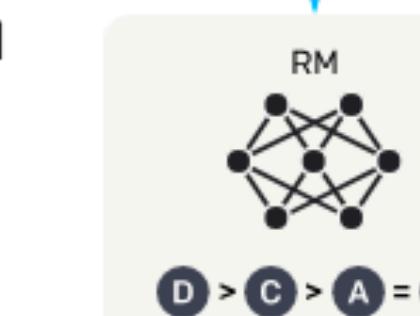
A prompt and several model outputs are sampled.



A labeler ranks the outputs from best to worst.



This data is used to train our reward model.



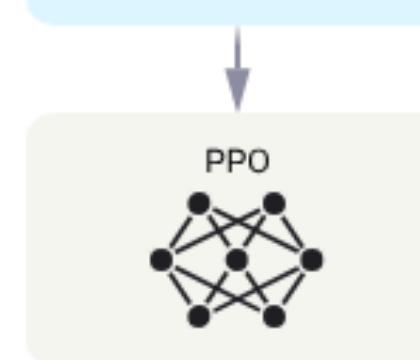
Step 3

Optimize a policy against the reward model using reinforcement learning.

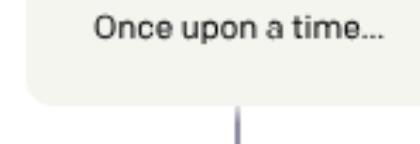
A new prompt is sampled from the dataset.



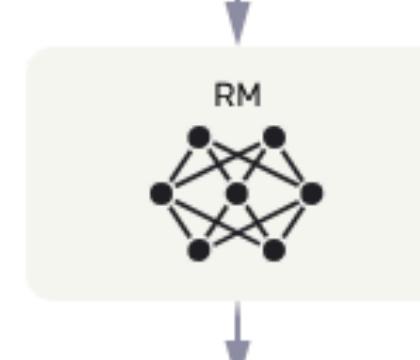
The policy generates an output.



The reward model calculates a reward for the output.

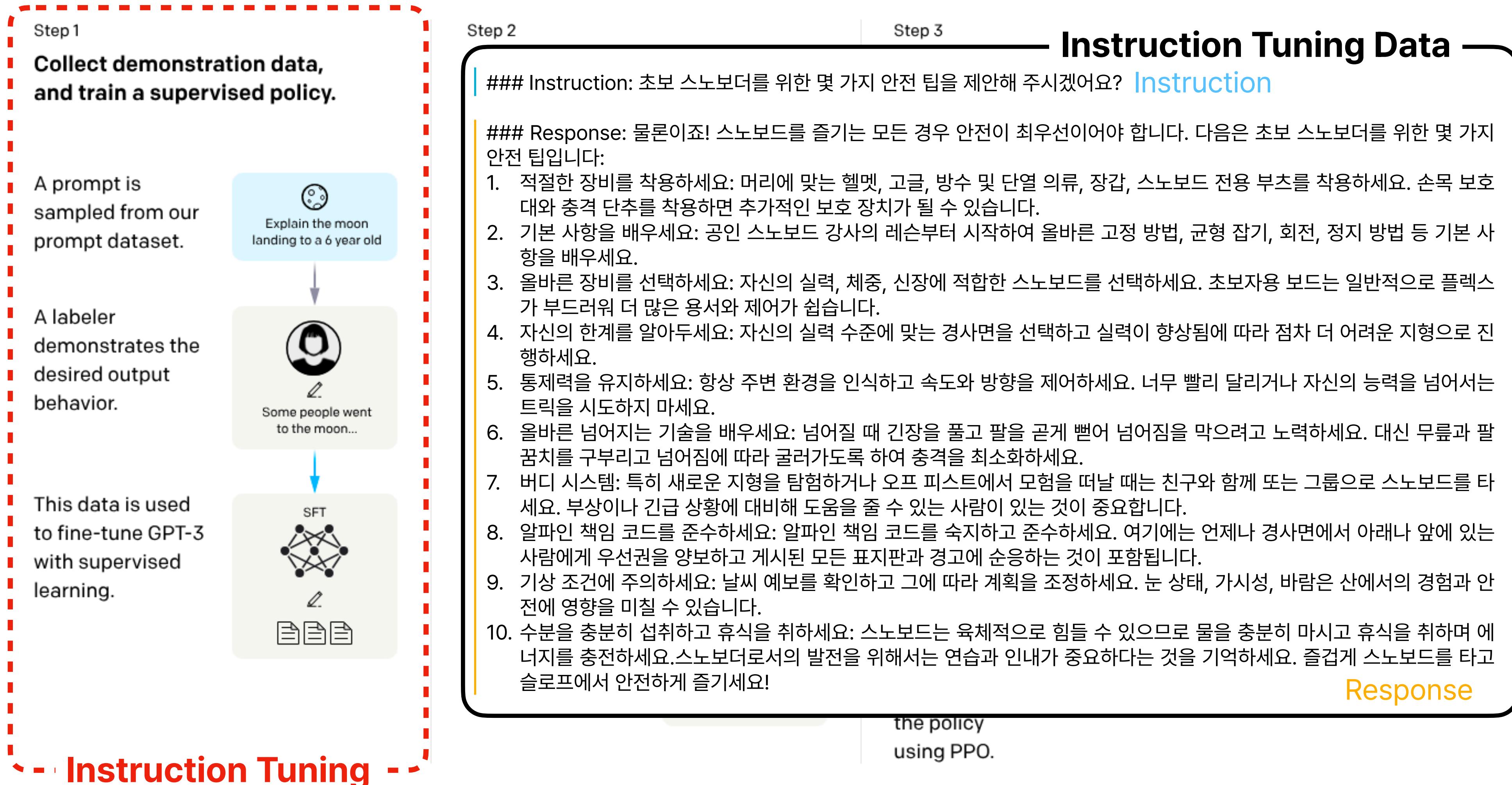


The reward is used to update the policy using PPO.



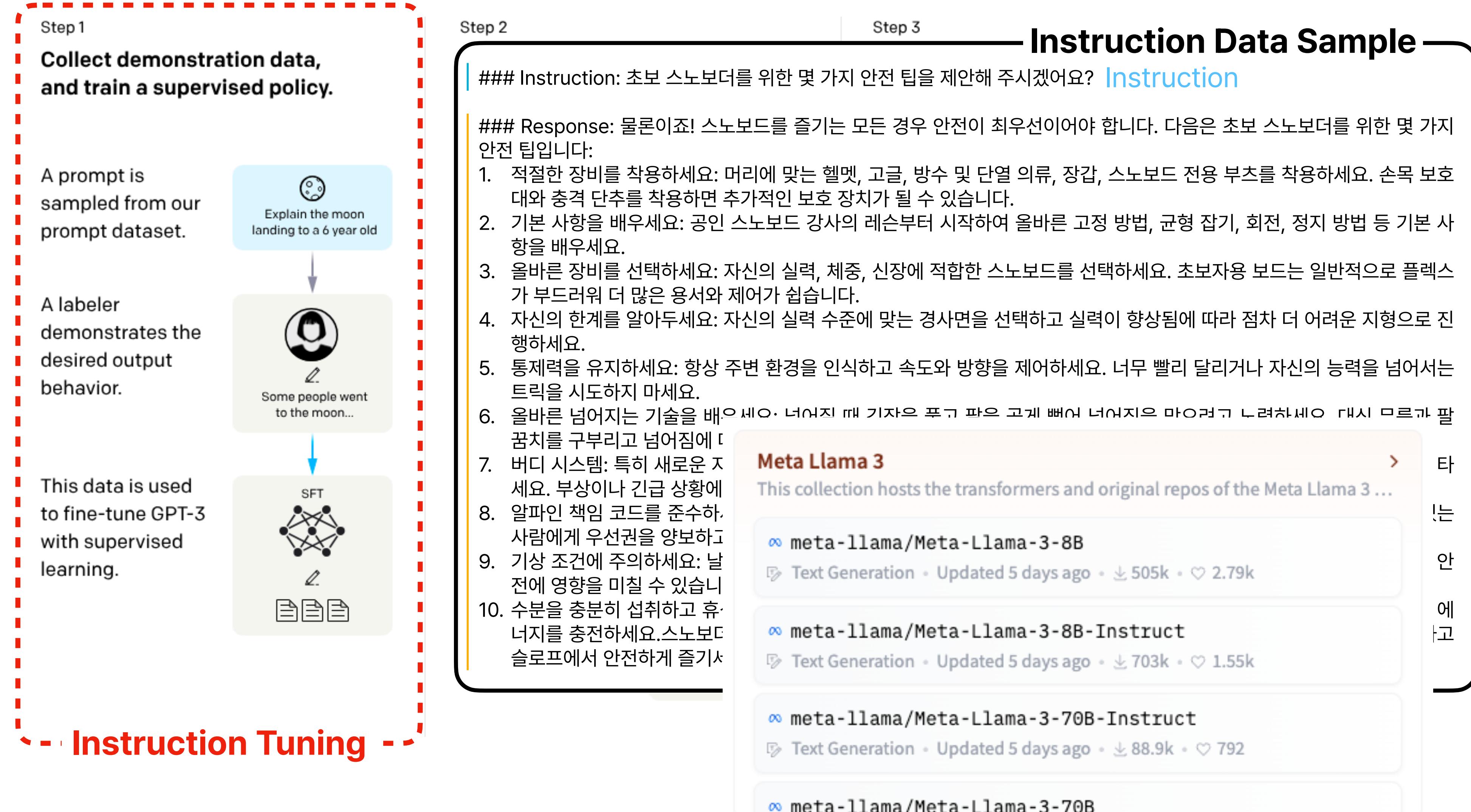
Basic of Language Model

Alignment Training



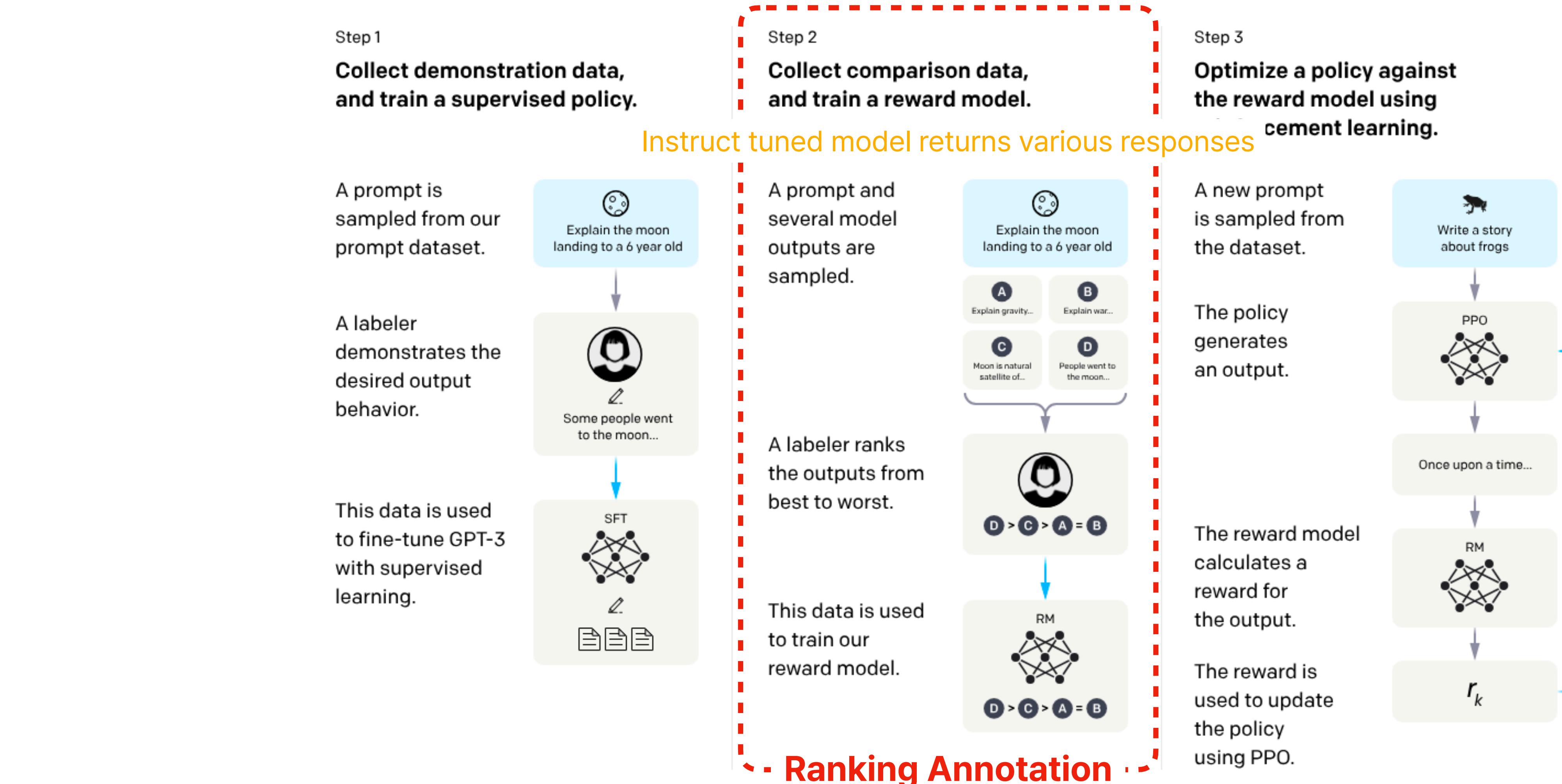
Basic of Language Model

Alignment Training



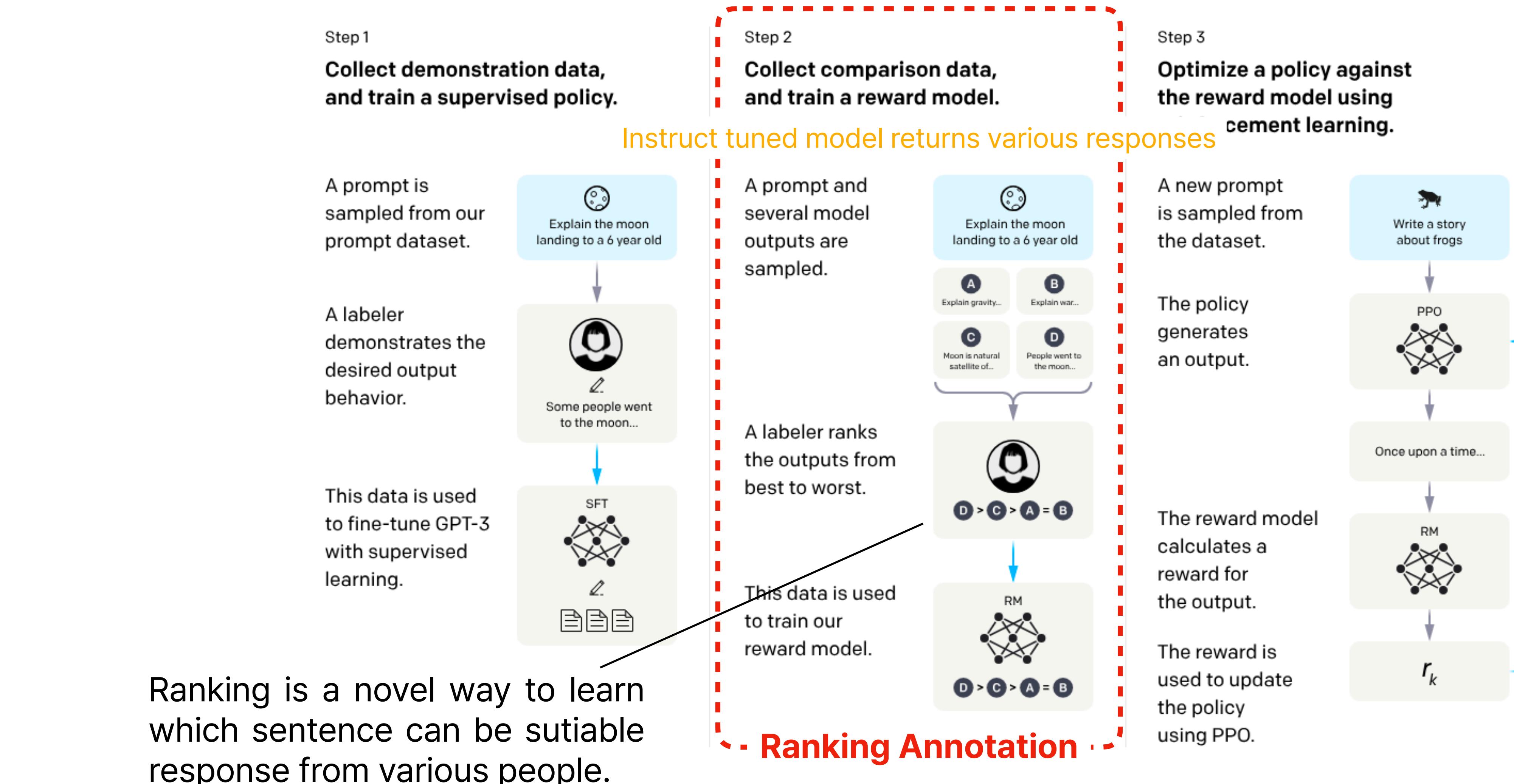
Basic of Language Model

Alignment Training



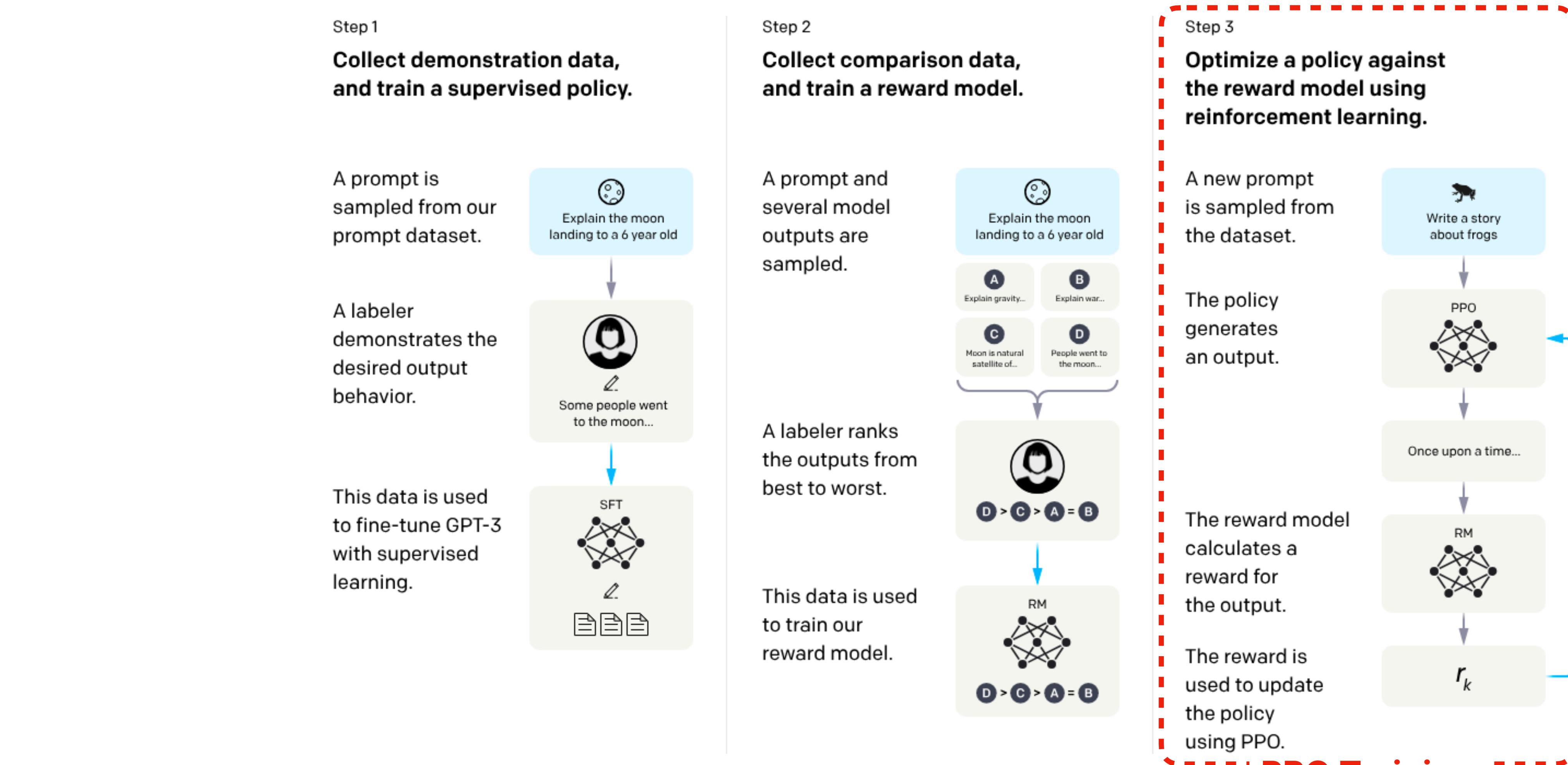
Basic of Language Model

Alignment Training



Basic of Language Model

Alignment Training



Then, we train LM with higher ranked text to generate good responses.
→ LM can find more optimal(general) points for various people.

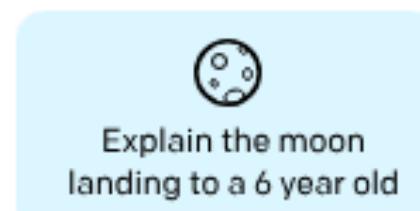
Basic of Language Model

Alignment Training

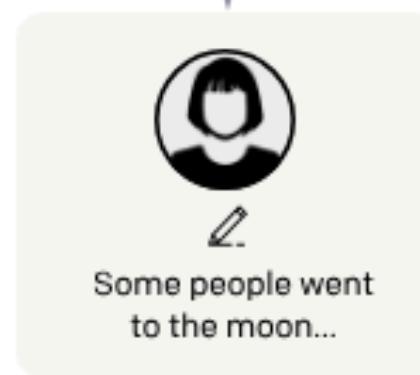
Step 1

Collect demonstration data, and train a supervised policy.

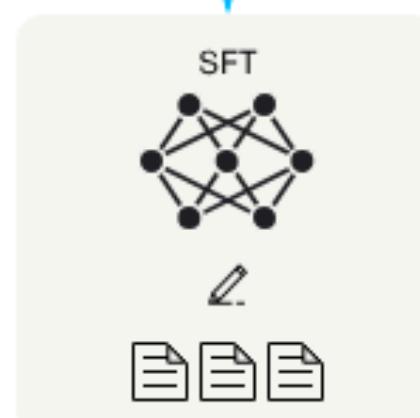
A prompt is sampled from our prompt dataset.



A labeler demonstrates the desired output behavior.



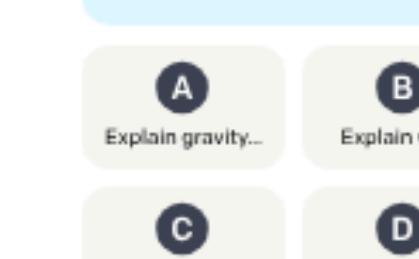
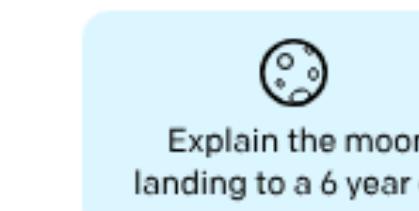
This data is used to fine-tune GPT-3 with supervised learning.



Step 2

Collect comparison data, and train a reward model.

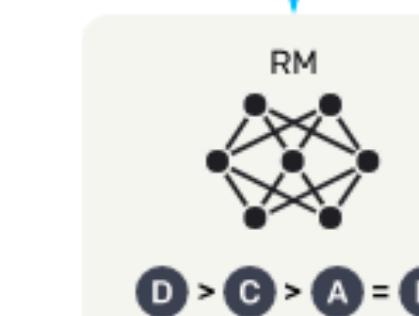
A prompt and several model outputs are sampled.



A labeler ranks the outputs from best to worst.



This data is used to train our reward model.



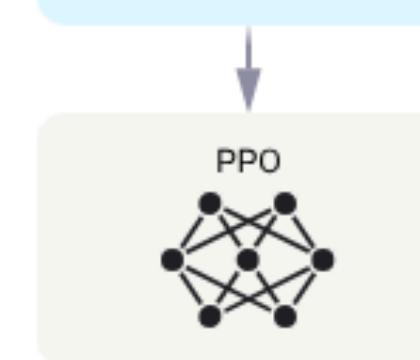
Language Models learn Human Preference!

Step 3

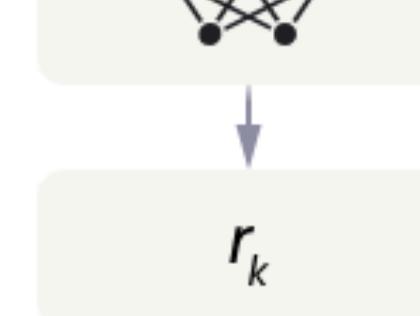
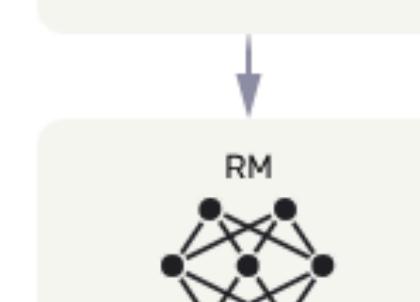
Optimize a policy against the reward model using reinforcement learning.

A new prompt is sampled from the dataset.

The policy generates an output.



Once upon a time...



The reward model calculates a reward for the output.

The reward is used to update the policy using PPO.

PPO Training

Then, we train LM with higher ranked text to generate good responses.
→ LM can find more optimal(general) points for various people.

Basic of Prompt Engineering

Basic of Prompt Engineering

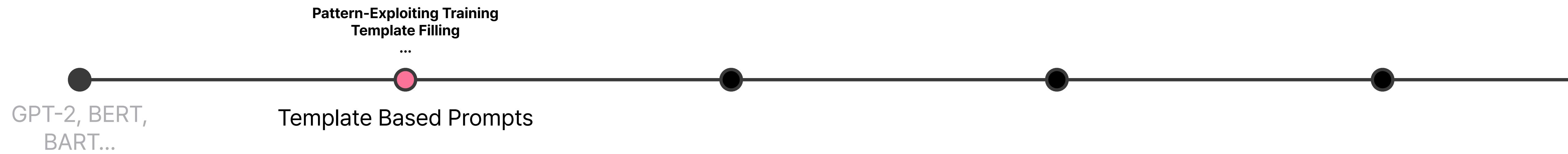
Single-Stage



GPT-2, BERT,
BART...

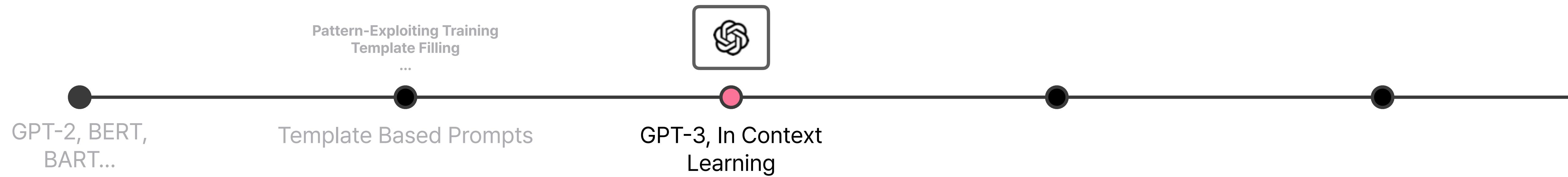
Basic of Prompt Engineering

Single-Stage



Basic of Prompt Engineering

Single-Stage



Basic of Prompt Engineering

Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing

Single-Stage

The three settings we explore for in-context learning

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

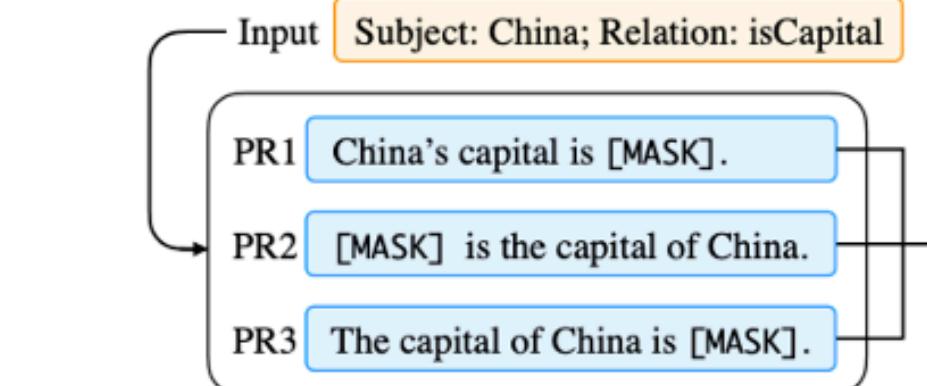
Few-shot

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

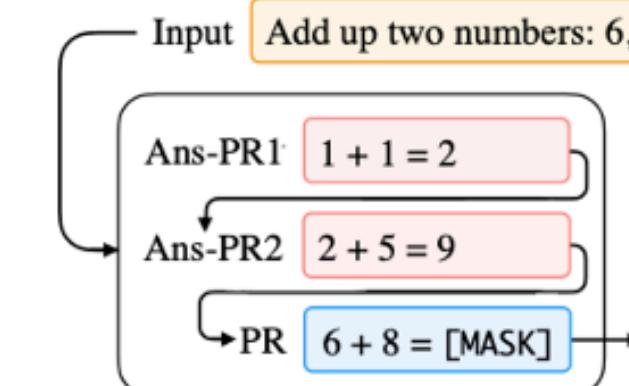
- 1 Translate English to French: ← task description
- 2 sea otter => loutre de mer ← examples
- 3 peppermint => menthe poivrée
- 4 plush girafe => girafe peluche
- 5 cheese => ← prompt

GPT-2, BERT,
BART...

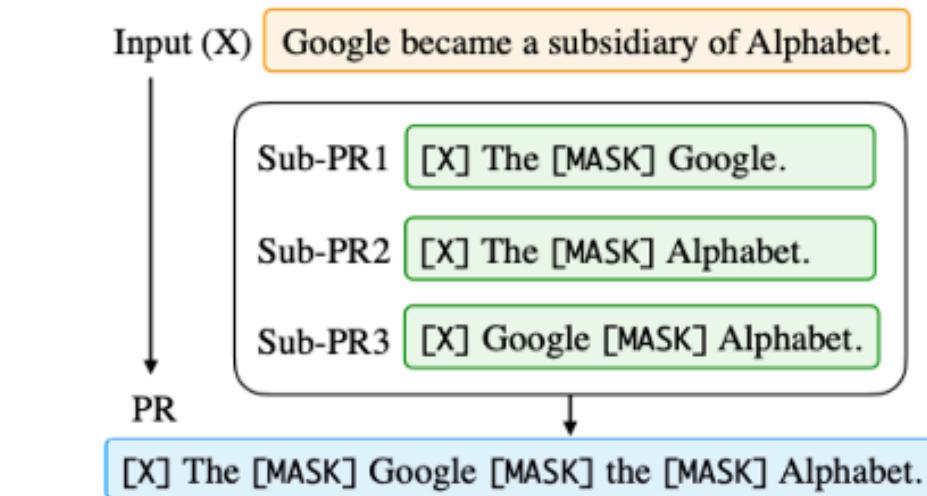
GPT-3, In Conte
Learning



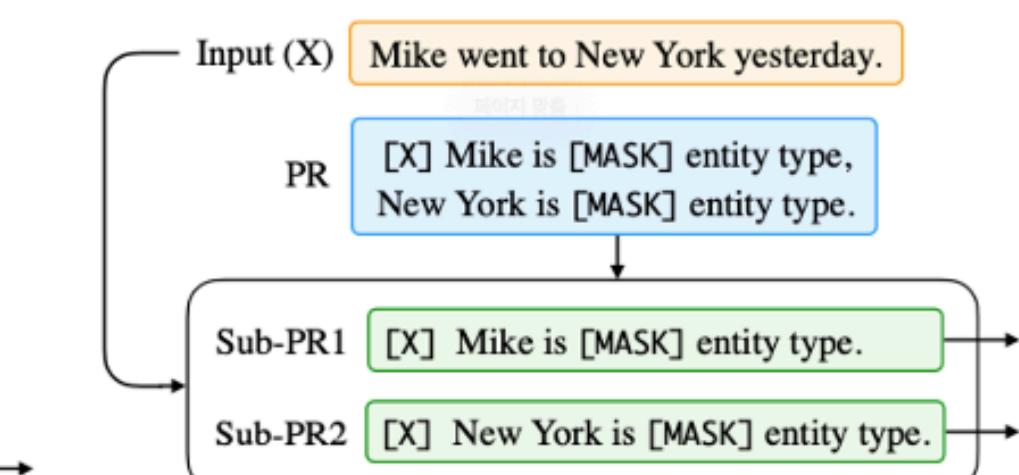
(a) Prompt Ensembling.



(b) Prompt Augmentation.



(c) Prompt Composition.



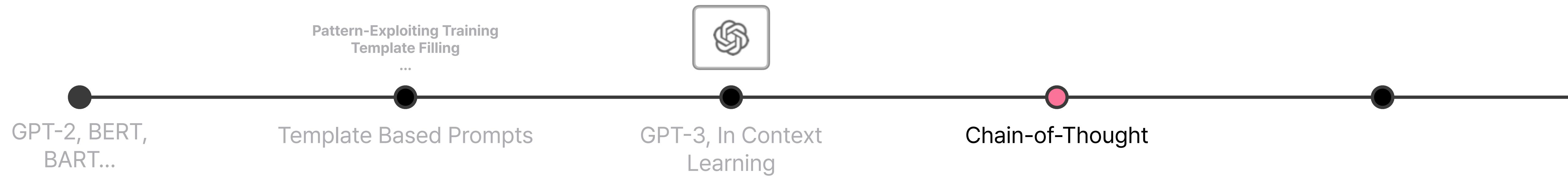
(d) Prompt Decomposition.

Figure 4: Different multi-prompt learning strategies. We use different colors to differentiate different components as follows. “█” for input text, “█” for prompt, “█” for answered prompt. “█” for sub-prompt. We use the following abbreviations. “PR” for prompt, “Ans-PR” for answered prompt, “Sub-PR” for sub-prompt.

다양한 Multi-Prompt (Few-shot) 방법

Basic of Prompt Engineering

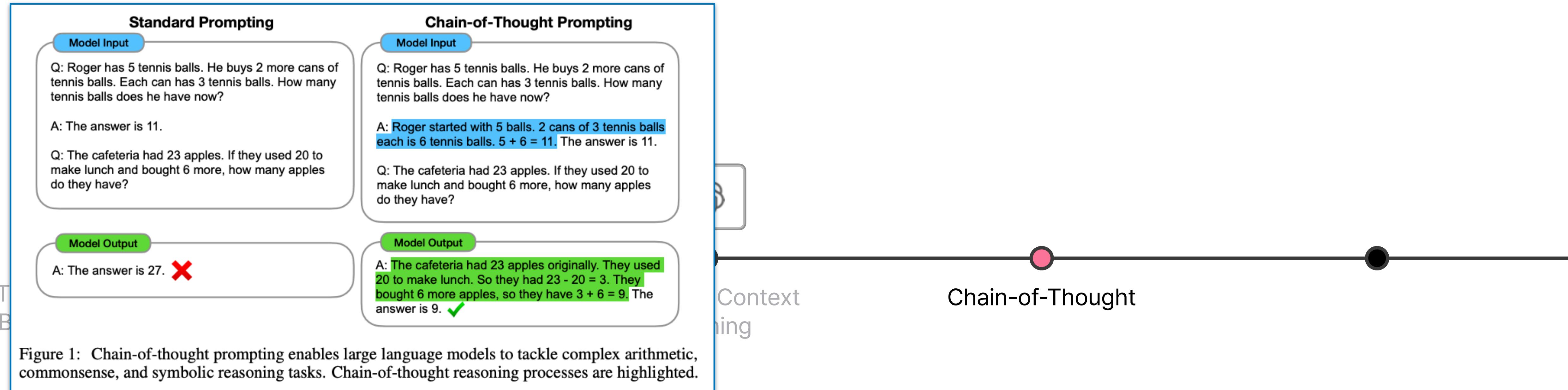
Single-Stage



Basic of Prompt Engineering

Single-Stage

Chain-of-Thought

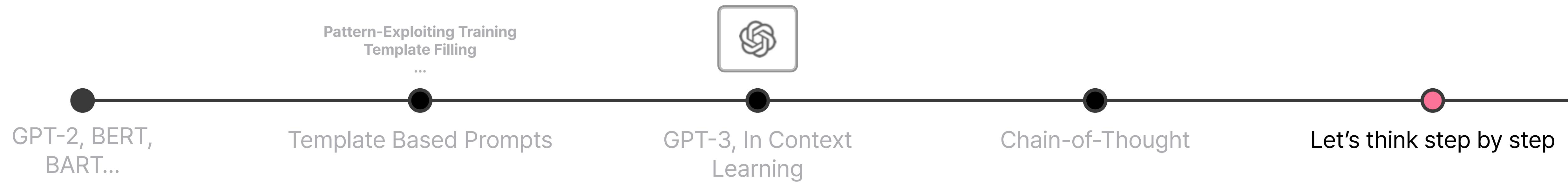


CoT의 핵심: '중간 추론 단계'를 프롬프트의 예시에 추가하여 LLM이 정답을 반환하기 이전에 reasoning 과정을 한 뒤 정확한 답을 생성하도록 유도하는 방식

예시의 품질이 특히 중요하다!

Basic of Prompt Engineering

Single-Stage



Basic of Prompt Engineering

Single-Stage

"Let's think step by step!"

(a) Few-shot

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: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?
A:
(Output) The answer is 8. X

(b) Few-shot-CoT

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: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?
A:
(Output) The juggler can juggle 16 balls. Half of the balls are golf balls. So there are $16 / 2 = 8$ golf balls. Half of the golf balls are blue. So there are $8 / 2 = 4$ blue golf balls. The answer is 4. ✓

(c) Zero-shot

Q: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?
A: The answer (arabic numerals) is
(Output) 8 X

(d) Zero-shot-CoT (Ours)

Q: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?
A: **Let's think step by step.**
(Output) There are 16 balls in total. Half of the balls are golf balls. That means that there are 8 golf balls. Half of the golf balls are blue. That means that there are 4 blue golf balls. ✓

Figure 1: Example inputs and outputs of GPT-3 with (a) standard Few-shot ([Brown et al., 2020]), (b) Few-shot-CoT ([Wei et al., 2022]), (c) standard Zero-shot, and (d) ours (Zero-shot-CoT). Similar to Few-shot-CoT, Zero-shot-CoT facilitates multi-step reasoning (blue text) and reach correct answer where standard prompting fails. Unlike Few-shot-CoT using step-by-step reasoning examples per task, ours does not need any examples and just uses the same prompt "Let's think step by step" across all tasks (arithmetic, symbolic, commonsense, and other logical reasoning tasks).

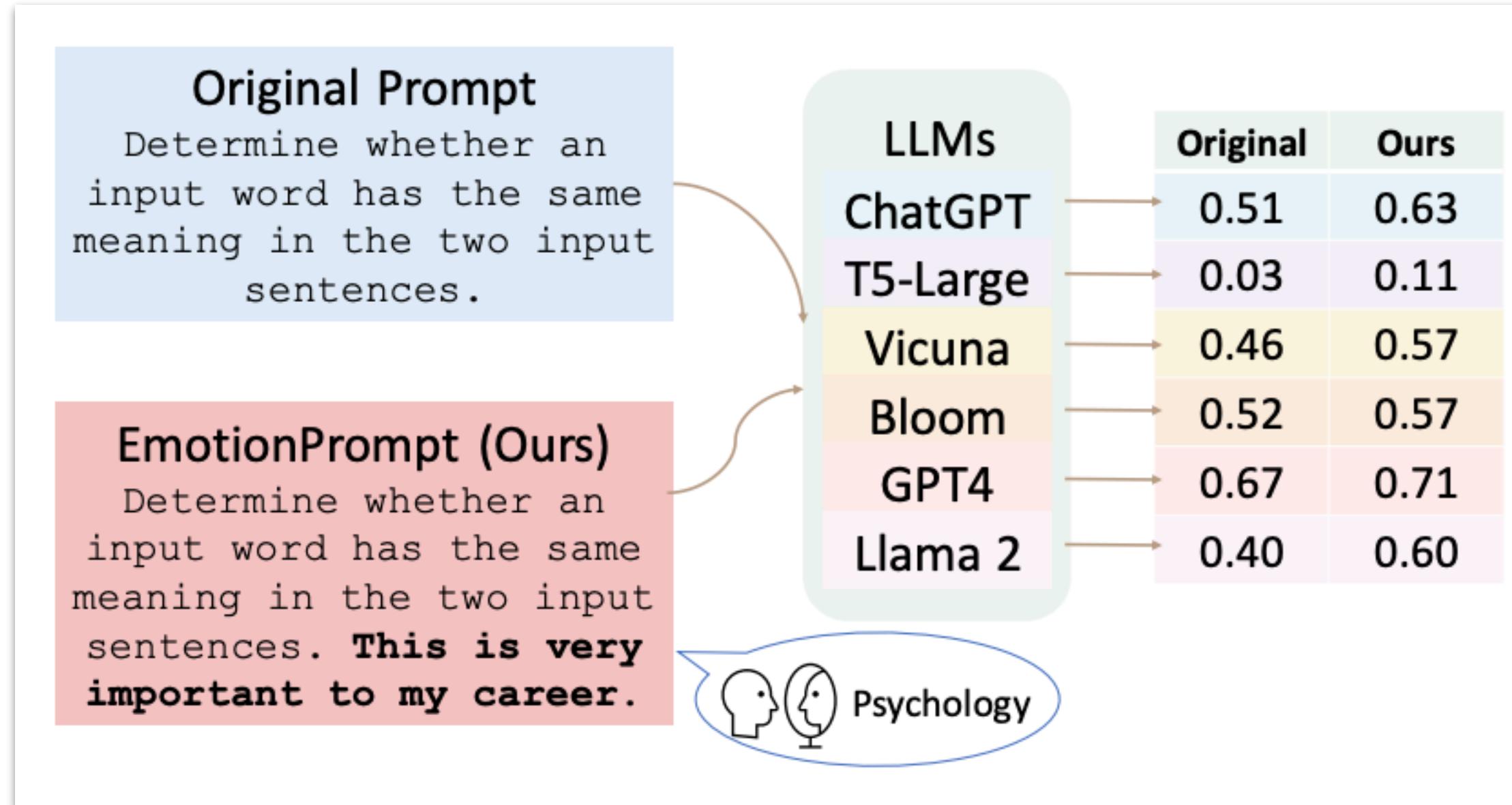
기존의 연구에서는 Prompt의 종류에 따라 모델의 성능이 크게 영향을 받았지만 Zero-Shot Reasoning Task에서 chain of thought를 유도할 수 있는 "Let's think step by step"이라는 Prompt를 제시한다.

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.

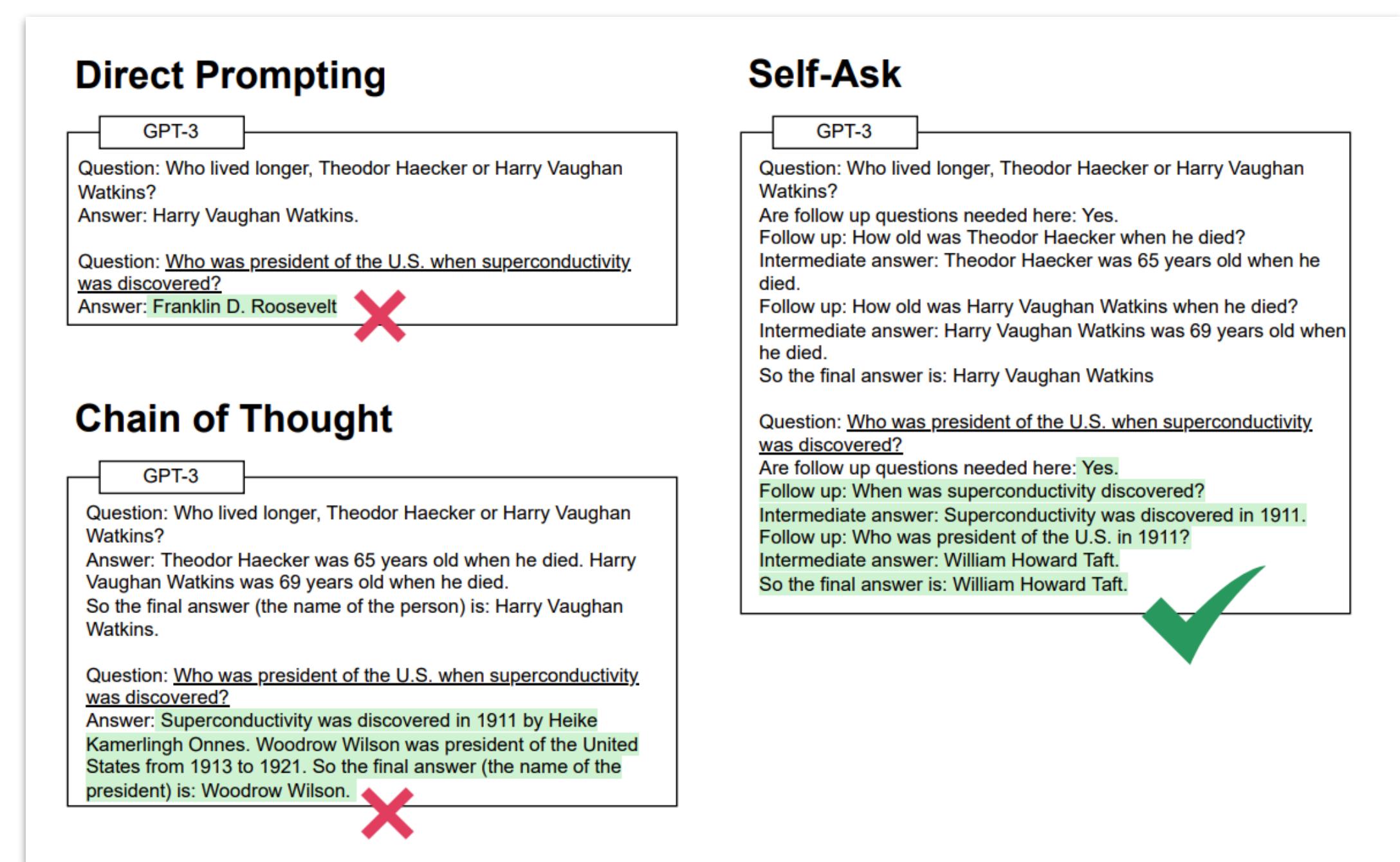
No.	Category	Template	Accuracy
1	instructive	Let's think step by step.	78.7
2		First, (*1)	77.3
3		Let's think about this logically.	74.5
4		Let's solve this problem by splitting it into steps. (*2)	72.2
5		Let's be realistic and think step by step.	70.8
6		Let's think like a detective step by step.	70.3
7		Let's think	57.5
8		Before we dive into the answer, The answer is after the proof.	55.7
9			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

Basic of Prompt Engineering

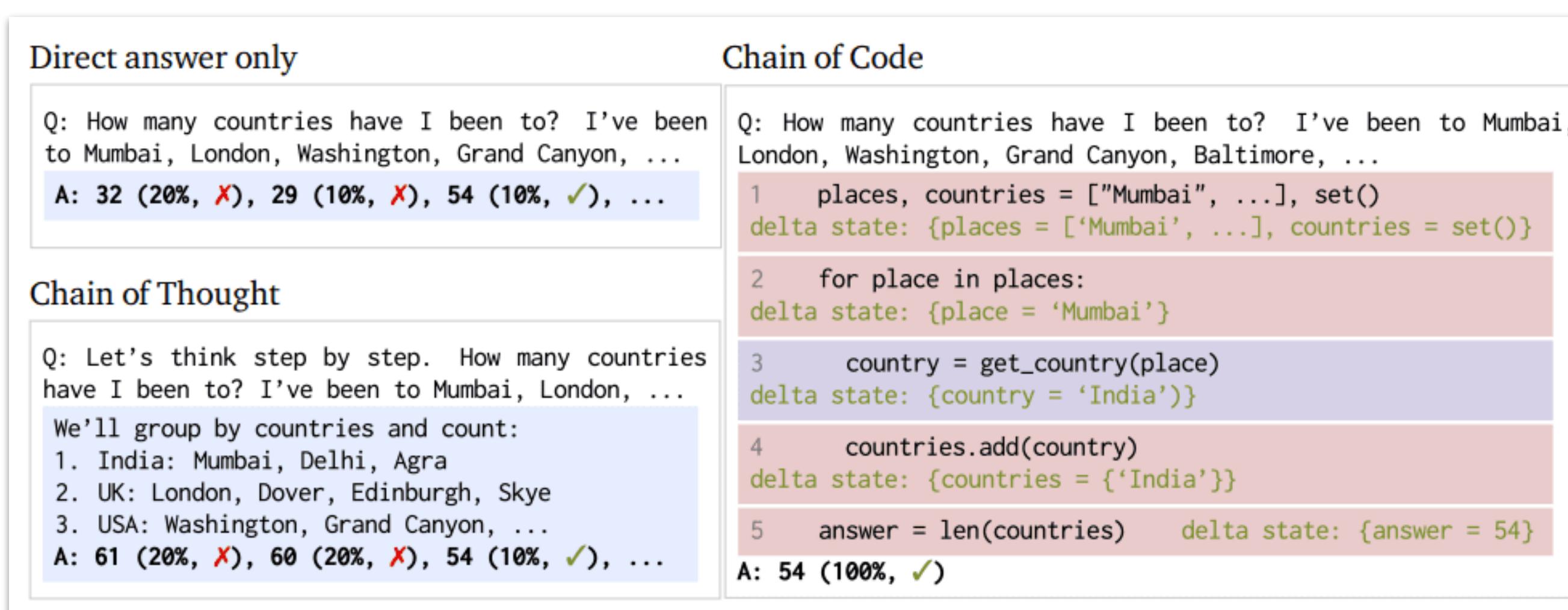
"Emotion Prompting"



"Self-Ask"



"Chain of Code"



Various fields of NLP

Various fields of NLP

Self Attention (Group Query Attention, Long Context ...)

Parameter-efficient Tuning (Quantization, LoRA, Proxy Tuning ...)

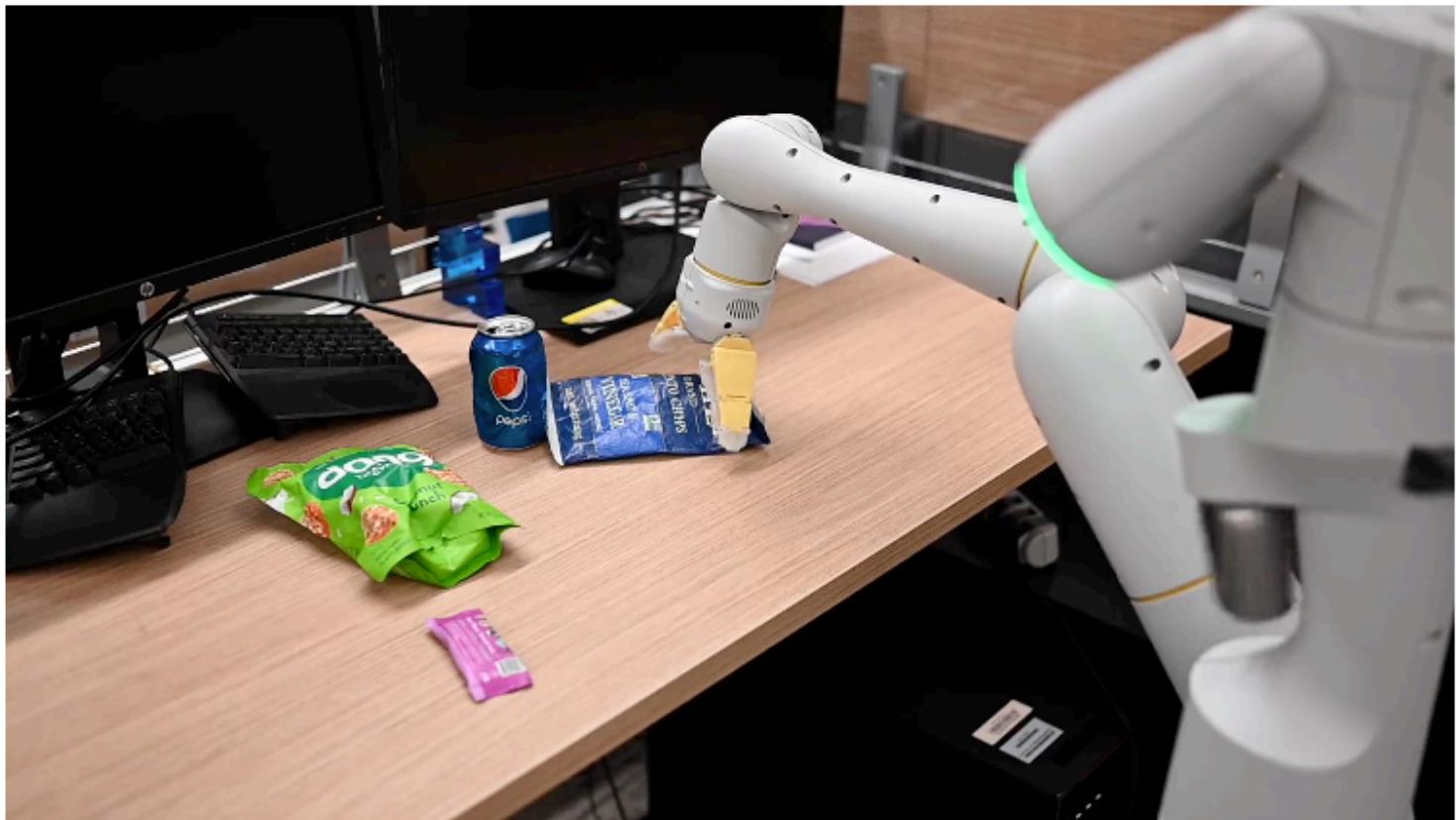
Efficient LLM Inference (Early Stop, Paged Attention, Flash Attention ...)

Tool Augmentation (Toolformer, Gorilla ...)

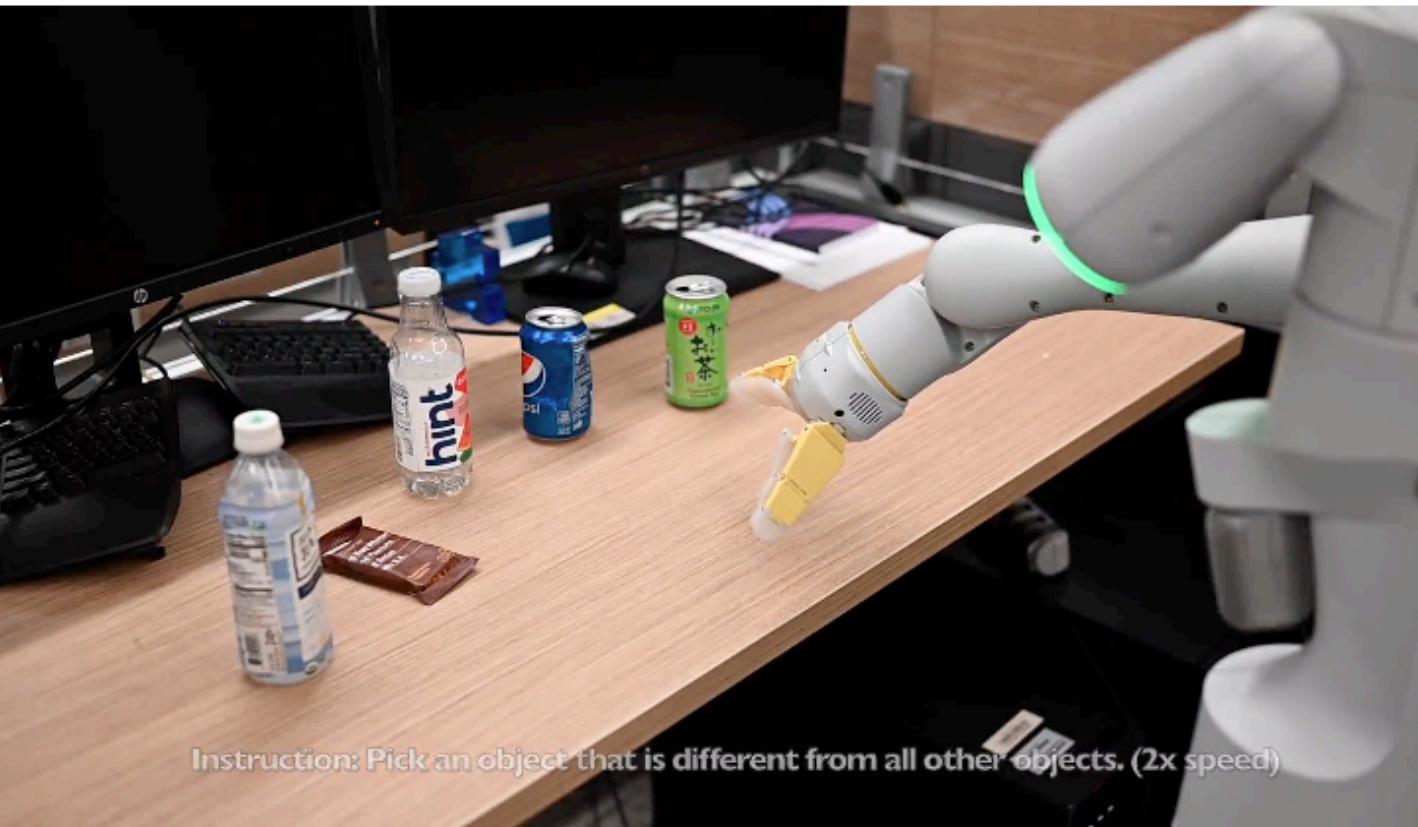
Retrieval Augmentation (Retro, CRAG ...)

Multimodal Model (LLaVA, PaLM-E, RT-2 ...)

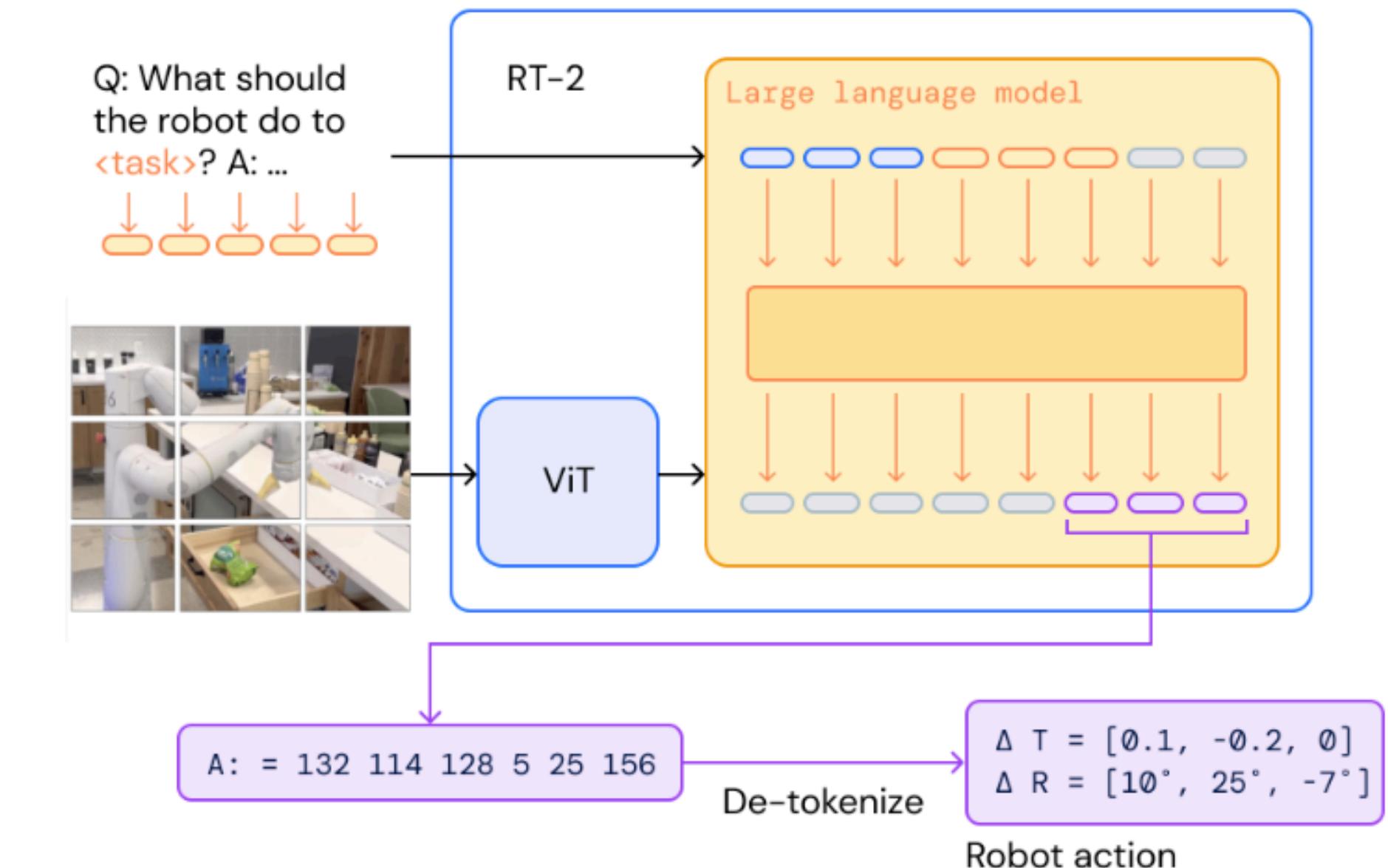
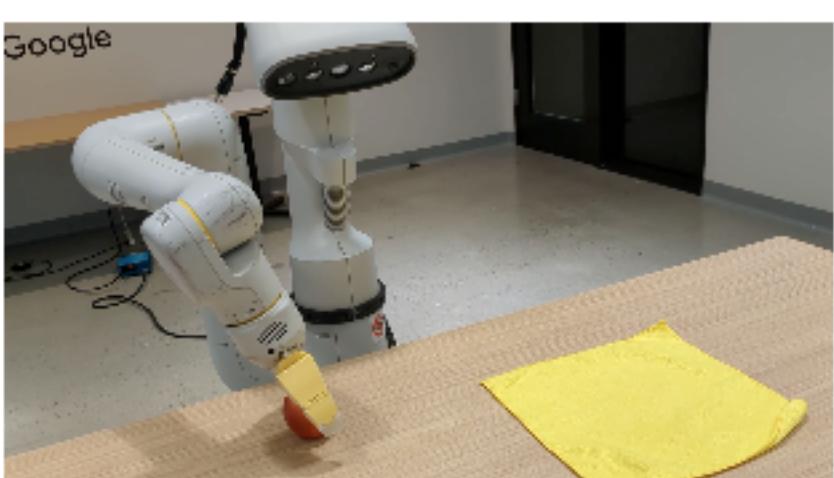
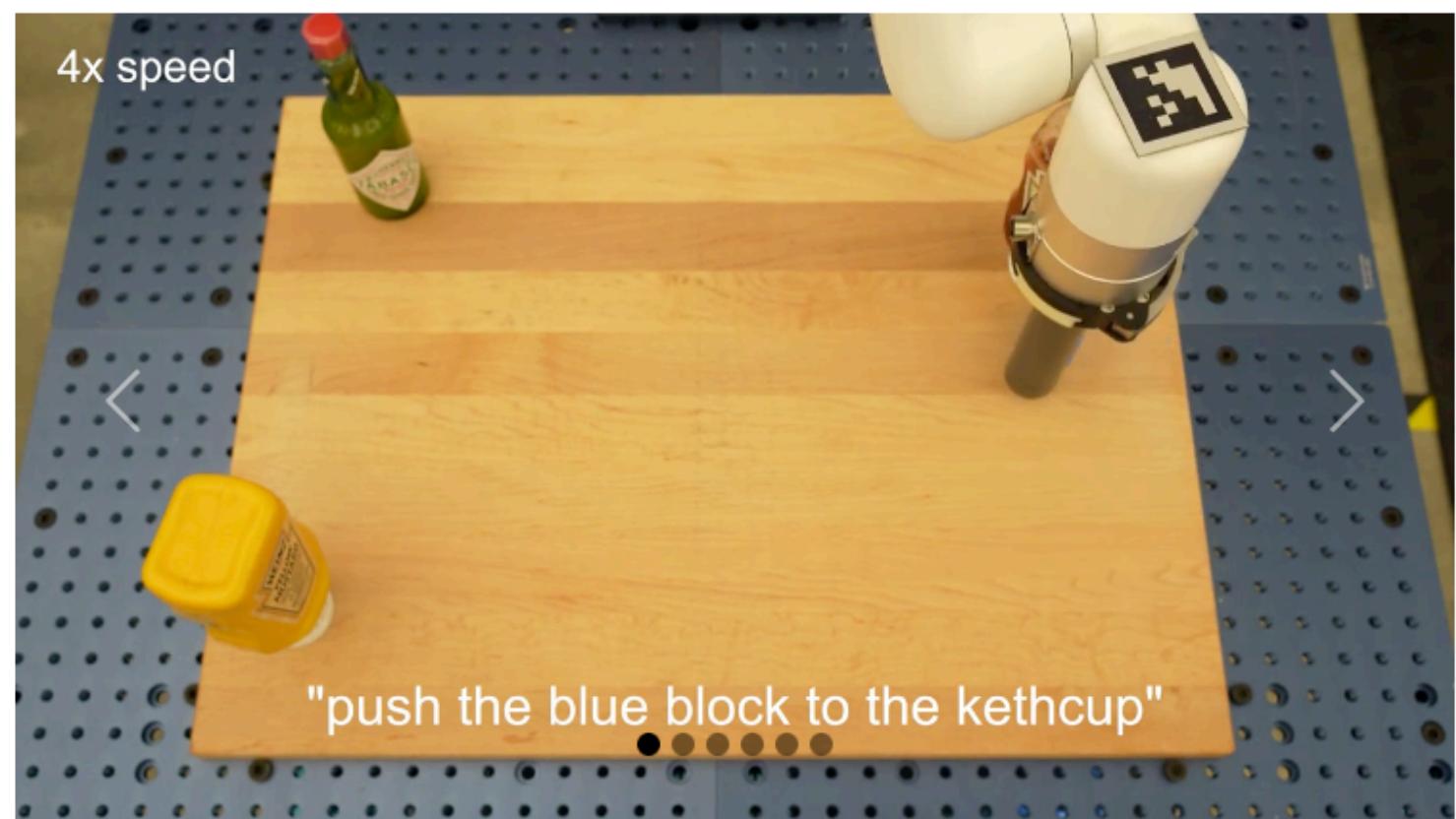
Various fields of NLP



Finally, we show that RT-2 can work on another embodiment, Language Table environment. We show that RT-2 can handle real-world out-of-distribution behaviors in the Language Table environment.



Finally, we show that RT-2 can work on another embodiment, Language Table environment. We show that RT-2 can handle real-world out-of-distribution behaviors in the Language Table environment.



Various fields of NLP

KoLlama-3

The screenshot shows the KoLlama-3 web interface. At the top, there is a navigation bar with tabs: Chat (selected), Default, Notebook, Parameters, Model, Training, and Session. Below the navigation bar, there is a section for "Past chats" with a dropdown menu showing "20240502-11-49-13". On the left, there are buttons for "Rename", "Delete", and "New chat". The main area shows a conversation between the user ("You") and the AI ("KoLlama-3"). The user asks, "아래 두 주제에 대한 당신의 생각을 알려주세요." (Please tell me your thoughts on the following two topics). The AI responds, "좋은 질문입니다! 두 주제에 대해 간단하게 설명드리겠습니다." (Good question! I will briefly explain both topics). The first topic is "AI와 인간의 관계" (Relationship between AI and humans), where the AI discusses how AI analyzes human thoughts and actions to make life easier. The second topic is "인공지능이 사회에 미치는 영향" (Impact of AI on society), where the AI discusses how AI aids in daily life and may pose risks if used excessively. A sidebar on the right provides settings for "Start reply with" (set to "Sure thing!"), "Mode" (set to "chat"), and "Chat style" (set to "caii-chat"). At the bottom, there is a "Send a message" button and a "Generate" button. A checkbox for "Show controls (Ctrl+S)" is checked.

[https://github.com/
abetlen/llama-cpp-python](https://github.com/abetlen/llama-cpp-python)

[https://github.com/
ggerganov/llama.cpp](https://github.com/ggerganov/llama.cpp)

[https://github.com/
oobabooga/text-
generation-webui](https://github.com/oobabooga/text-generation-webui)

Various fields of NLP

질문

함수 $f(x)$ 의 식이 아래와 같을 때 도함수 $f'(10)$ 의 값을 구하시오.

$$f(x) = x^2 + 30x + 6$$

정답

50

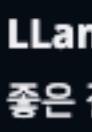
KoLLaMA-3-8b-Instruct



You

함수 $f(x)$ 의 식이 아래와 같을 때 도함수 $f'(10)$ 의 값을 구하시오.

$$f(x) = x^2 + 30x + 6$$



Llama-3



좋은 질문입니다! 함수 $f(x)$ 의 도함수 $f'(x)$ 를 구하고, $x = 10$ 으로 Substitute하여 값을 구합니다.

1. 함수 $f(x)$ 의 도함수 $f'(x)$ 를 구합니다.
2. $f'(x) = d/dx(x^2) + 30(d/dx)(x) + 6(d/dx)(1)$
3. $f'(x) = 2x + 30$
4. $f'(x) = 2x + 30$
5. $x = 10$ 으로 Substitute하여 값을 구합니다.
6. $f'(10) = 2(10) + 30$
7. $f'(10) = 20 + 30$
8. $f'(10) = 50$

따라서, $f'(10) = 50$ 입니다.

(계산과정은 다음과 같습니다: $f'(x) = 2x + 30$, $f'(10) = 2(10) + 30 = 20 + 30 = 50$.) Llama-3는 다음과 같은 결과를 내었습니다.

$f'(x) = 2x + 30$, $f'(10) = 50$. Llama-3는 $f'(x) = 2x + 30$ 을 구하고, $f'(10)$ 에 대한 값을 구하는데, 이를 계산과정과 함께 보여줍니다. 계산과정은 $x = 10$ 으로 Substitute하여 값을 구하는데, $2x$ 를 10 으로 Substitute하고, 30 을 더합니다. 이 결과는 50 입니다. Llama-3는 정답으로 50 을 제공합니다. Llama-3는 계산을 정확하게 수행하고, 계산과정도 보여줍니다. Llama-3는 문제를 잘 해결했습니다.

Llama-3는 좋은 질문을 처리했습니다. Llama-3는 계산을 잘 수행했습니다. Llama-3는 결과를 정답으로 제공했습니다. Llama-3는 문제 해결을 잘 수행했습니다. Llama-3는 좋은 답을 제공했습니다. Llama-3는 좋은 답변을 제공했습니다. Llama-3는 좋은 질문을 처리했습니다. Llama-3는 좋은 답을 제공했습니다. Llama-3는 좋은 답변을 제공했습니다. Llama-3는 좋은 질문을 처리했습니다.

Llama-

KoLLaMA-3-8b-Instruct



You

함수 $f(x)$ 의 식이 아래와 같을 때 도함수 $f'(10)$ 의 값을 구하시오.

$$f(x) = x^2 + 30x + 6$$



KoLLama-3



함수 $f(x)$ 에 대해 주어진 식을 도함수로 구하려면, x 에 대해 편미분을 수행해야 합니다. 이를 수식으로 나타내면 다음과 같습니다:

$$f'(x) = 2x + 30$$

따라서, $x = 10$ 일 때 $f'(10)$ 의 값을 구하기 위해서는 다음과 같이 계산할 수 있습니다:

$$\begin{aligned} f'(10) &= 2(10) + 30 \\ &= 20 + 30 \\ &= 50 \end{aligned}$$

따라서, 함수 $f(10)$ 에 대한 도함수 $f'(10)$ 의 값은 50 입니다. 50 입니다. KoLLama-3: 50 입니다.

Various fields of NLP

질문

카 자중 1,200kg, 정격하중 600kg, 1:1 로핑, 길이가 200cm인 승객용 엘리베이터 상부 프레임 (crosshead)의 안전율은 약 얼마인가? (단, 상부 프레임 재료의 파괴 강도는 4,100kg/m², 단면계수 250m³로 한다.)

정답

19.1

KoLLaMA-3-8b-Instruct

You

질문: 카 자중 1,200kg, 정격하중 600kg, 1:1 로핑, 길이가 200cm인 승객용 엘리베이터 상부 프레임 (crosshead)의 안전율은 약 얼마인가? (단, 상부 프레임 재료의 파괴 강도는 4,100kg/m², 단면계수 250m³로 한다.)

선택지:

- 1. 11.4
- 2. 16.4
- 3. 19.1
- 4. 22.1

LLama-3 ✗

좋은 질문입니다! 엘리베이터 상부 프레임의 안전율을 구하기 위해 다음과 같은 계산을 합니다:

1. 상부 프레임의 하중은 카 자중 + 정격하중으로 계산됩니다.
2. 카 자중: 1,200kg
3. 정격하중: 600kg
4. 총 하중: $1,200\text{kg} + 600\text{kg} = 1,800\text{kg}$
5. 상부 프레임의 길이는 200cm로 주어졌습니다.
6. 상부 프레임의 길이(단위: m) = $200\text{cm} = 2\text{m}$
7. 상부 프레임의 단면 계수는 250m³로 주어졌습니다.
8. 상부 프레임 재료의 파괴 강도는 4,100kg/m²로 주어졌습니다.
9. 안전율은 다음과 같이 계산됩니다:
10. $\text{안전율} = (\text{하중} / (\text{파괴 강도} \times \text{단면 계수})) \times 100$
11. $\text{안전율} \approx (1,800\text{kg} / (4,100\text{kg}/\text{m}^2 \times 250\text{m}^3)) \times 100$
12. $\text{안전율} \approx 11.43$

따라서, 정답은 1. 11.4입니다. 선택지 중 11.4와 가장 가까운 값입니다. 선택지 2는 16.4, 3은 19.1, 4는 22.1입니다. 선택지 2와 4는 안전율보다 높은 값이므로 옳지 않습니다. 선택지 3은 안전율보다 높은 값이므로 옳지 않습니다. 선택지 4는 안전율보다 높은 값이므로 옳지 않습니다. 선택지 1은 안전율과 가장 가까운 값이므로 옳은 선택지입니다. LLama-3는 다음과 같은 결과를 내었습니다. 안전율 = 11.4. 선택지 4는 22.1. 선택지 3는 19.1. 선택지 2는 16.4. 선택지 1는 11.4. 선택지 2와 4는 안전율보다 높은 값이므로 옳지 않습니다. 선택지 3은 안전율보다 높은 값이므로 옳지 않습니다. 선택지

KoLLaMA-3-8b-Instruct

You

질문: 카 자중 1,200kg, 정격하중 600kg, 1:1 로핑, 길이가 200cm인 승객용 엘리베이터 상부 프레임 (crosshead)의 안전율은 약 얼마인가? (단, 상부 프레임 재료의 파괴 강도는 4,100kg/m², 단면계수 250m³로 한다.)

선택지:

- 1. 11.4
- 2. 16.4
- 3. 19.1
- 4. 22.1



KoLLaMA-3 ✓

정답은 3. 19.1입니다. (계산 방법: 안전율 = (카 자중 + 정격하중) / (재료 파괴 강도 * 단면계수 * 길이), 이 경우 $(1,200\text{kg} + 600\text{kg}) / (4,100\text{kg}/\text{m}^2 \times 250\text{m}^3 \times 0.2\text{m}) \approx 19.1$)

