

AI는 어떻게 생각을 배우는가? CNN에서 GPT까지

Date : 2025.12.13

Presenter: Byeonggyu Kim

Computer Science and Engineering

qudrb6989@postech.ac.kr



ChatGPT와 같은 AI 툴을 이용한 적이 있나요?



Question

ChatGPT와 같은 AI 툴을 이용한 적이 있나요?





Question

ChatGPT와 같은 AI 툴을 이용한 적이 있나요?





Question

ChatGPT와 같은 AI 툴을 이용한 적이 있나요?





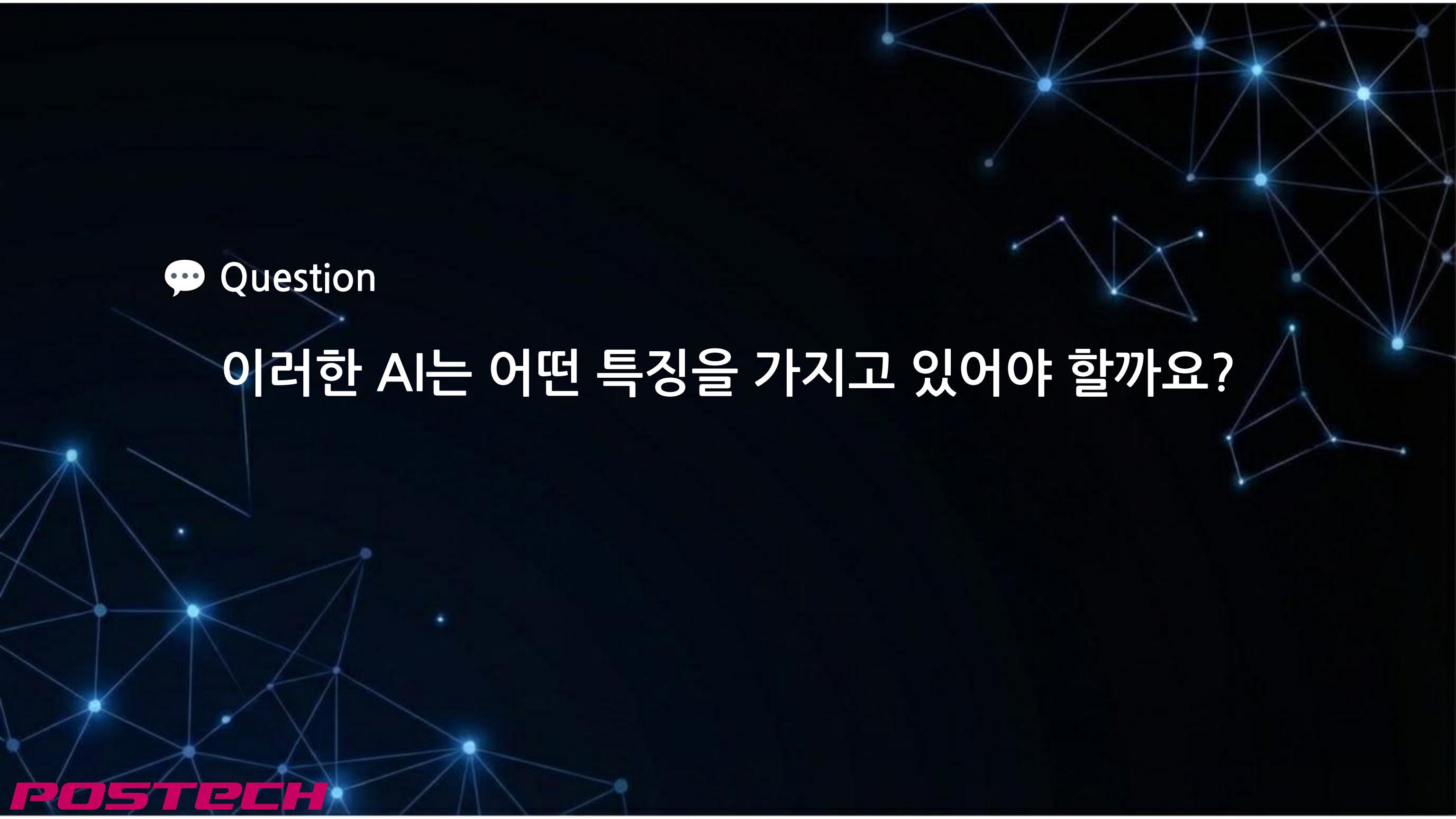
Question

ChatGPT와 같은 AI 툴을 이용한 적이 있나요?

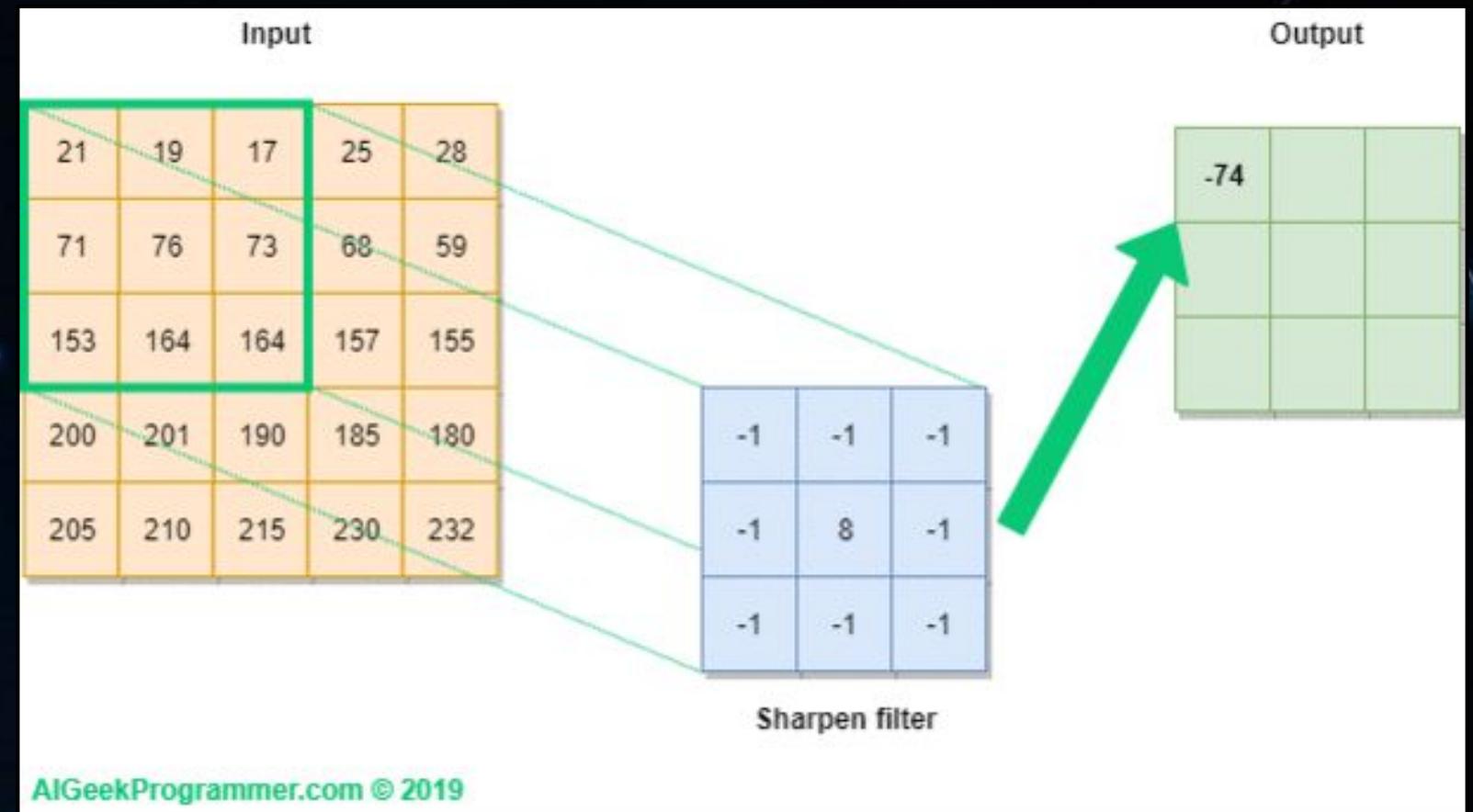




이러한 AI는 어떤 특징을 가지고 있어야 할까요?



CNN의 등장



1. 지역 패턴(부분 특징)을 찾아낸다

CNN은 이미지처럼 근처에 있는 정보만 보면서 모서리·선·무늬 같은 작은 패턴을 먼저 찾아냅니다.

2. 작은 특징을 큰 의미로 쌓아 올린다

여러 층을 지나면서 작은 패턴이 모여 눈·코처럼 큰 구조를 이해합니다.

CNN의 등장

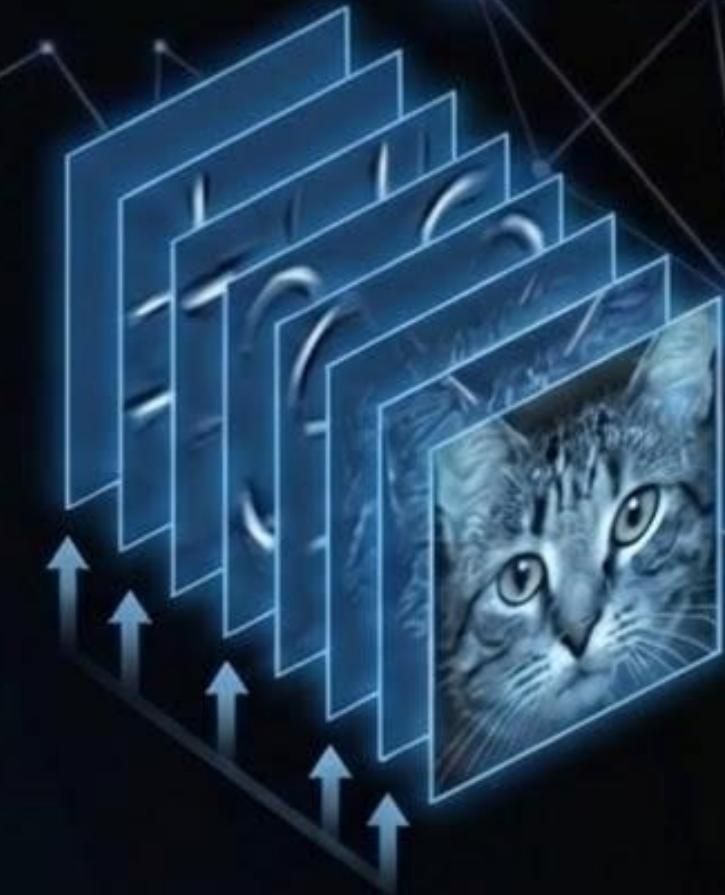


A 5x5 input image with a 3x3 red box highlighting a specific receptive field. The highlighted area contains the values: 21, 19, 17, 25, 71, 76, 73, 59, 153, 164, 157, 155, 200, 201, 185, 180.

21	19	17	28
71	76	73	59
153	164	157	155
200	201	185	180

A convolutional step diagram showing a 3x3 kernel applied to a 3x3 receptive field. The kernel values are: -1, -1, -1; -1, 8, -1; -1, -1, -1. The result of the multiplication and addition is -74.

-1	-1	-1
-1	8	-1
-1	-1	-1



1. 지역 패턴(부분 특징)을 찾아낸다.

CNN은 이미지처럼 근처에 있는 정보만 보면서 모서리, 선, 무늬 같은 작은 패턴을 먼저 찾아냅니다.

2. 작은 특징을 큰 의미로 쌓아 올린다.

여러 층을 지나면서 작은 패턴이 모여 눈, 코처럼 큰 구조를 이해합니다.

CNN을 이용하여 텍스트 읽기

문장 1 : I love pizza.

문장 2 : I love chicken.

위 두 문장을 CNN으로 읽는다면 어떻게 될까요?

CNN을 이용하여 텍스트 읽기

문장 1 : 나는 밥을 먹었다.

문장 2 : 밥은 나를 먹었다.

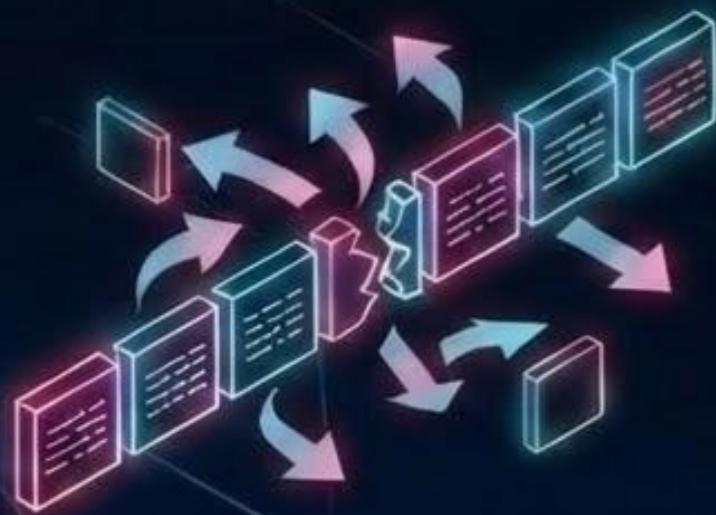
위 두 문장을 CNN으로 읽는다면 어떻게 될까요?

CNN을 이용하여 텍스트 읽기

The cat sat on the mat because it is warm.

위 문장을 CNN으로 읽는다면 어떻게 될까요?

CNN의 한계

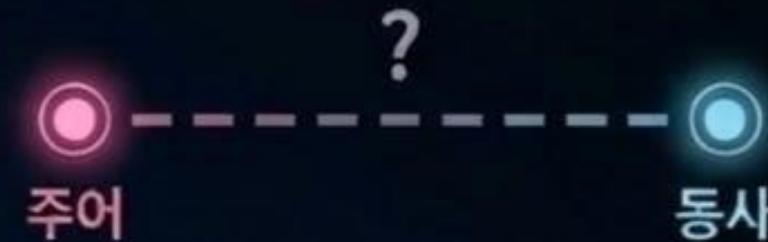


1. 단어의 순서를 기억하지 못함

문맥을 파악하는데 필수적인 단어의 배열
순서를 인식하지 못합니다.

2. 멀리 떨어진 관계를 이해 부족

문장 내에서 거리가 먼 단어들 간의
의존관계를 포착하기 어렵습니다.



3. 문장 전체 의미 파악의 어려움

여러개별적인 특징에 집중하여, 문장 전체가
가지는 통합적인 의미를 놓치기 쉽습니다.



RNN(Recurrent Neural Network)이란 무엇인가?

RNN은 과거의 정보를 기억하며 순서대로 데이터를 처리하는 신경망입니다.



1. 순서가 중요한 데이터를 처리

문장, 음성처럼 앞뒤 순서에 따라 의미가 달라지는 데이터를 이해할 수 있다.



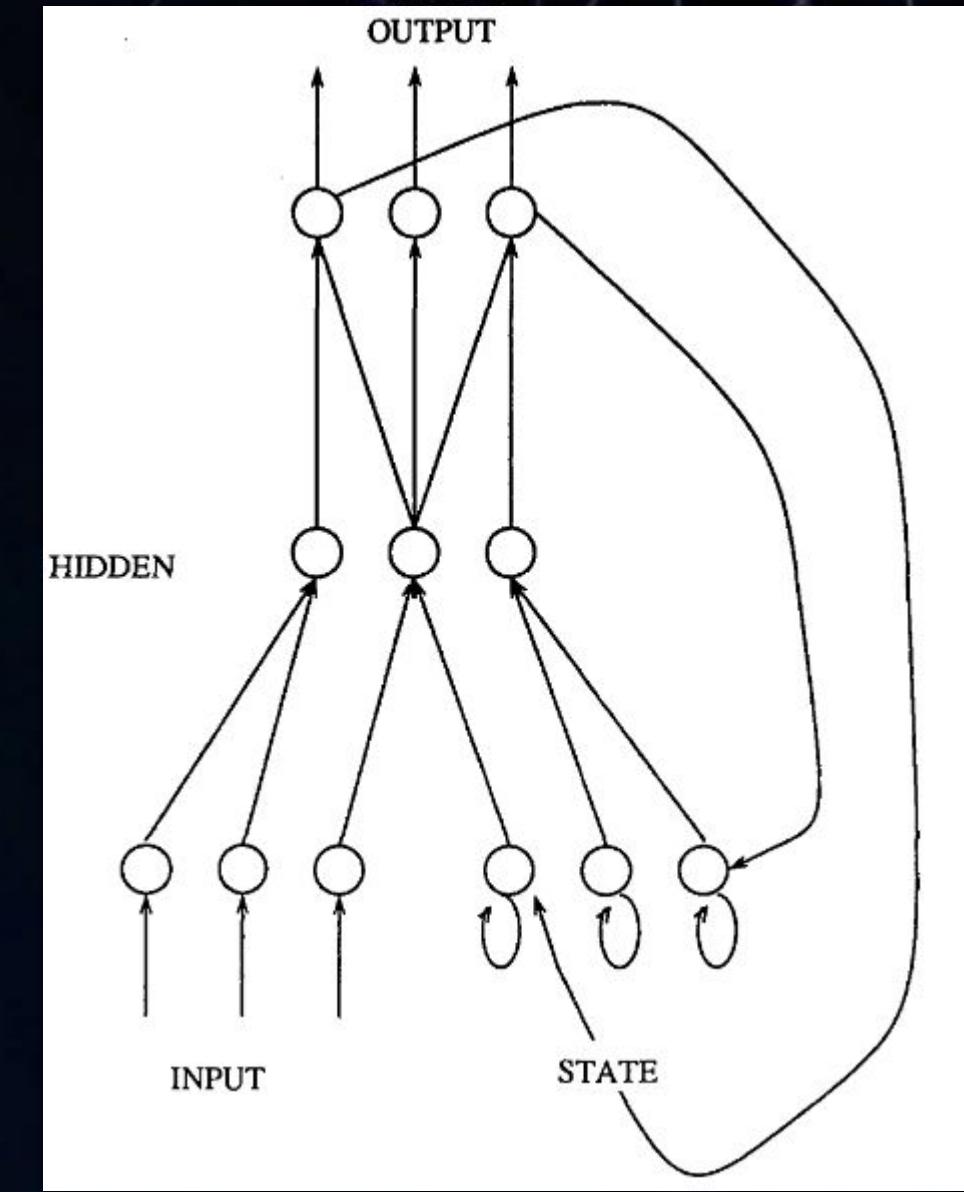
2. 과거 정보를 활용하여 현재를 해석

현재 입력뿐 아니라 이전 hidden state를 참고해 문맥을 파악한다.



3. 동일한 구조가 시간에 따라 반복되어 문맥을 형성

긴 시퀀스도 같은 모델을 반복 적용하여 전체 흐름을 이해할 수 있다.



Finding Structure in Time(1986)

언어모델은 어떻게 '대답'을 만드는가?

언어모델은 인간과 같은 방식으로 이해하지 않지만, 학습 과정에서 방대한 패턴을 기반으로 일관되고 논리적인 문장을 생성할 수 있는 내부 표현을 학습해요.
이를 통해 가장 자연스러운 다음 단어를 예측하는 AI 모델입니다.

쉬운 비유

"ChatGPT는 말하기를 정말 많이 연습한 학생이에요. 다음에 올 말을 확률적으로
'가장 그럴듯하게' 고르는 거죠."

예시: "나는 오늘 학교에 ____."

→ '간다', '갔다', '안 갔다' 중 확률이 가장 높은 걸 고름.

Enter text:
One, two, three,

One 3198 11 , two 734 11 , three 1115 11

3198 11 734 11 1115 11

Prediction

#	probs	next token ID	predicted next token
0	46.44%	1440	four
1	7.48%	290	and
2	7.31%	1936	five
3	2.66%	393	or
4	2.54%	2237	six
5	2.09%	1115	three
6	1.86%	3863	maybe
7	1.62%	345	you
8	1.23%	257	a
9	0.92%	530	one

언어모델은 어떻게 '대답'을 만드는가?

언어모델은 문장 속 단어들을 숫자로 바꾸어, '다음에 올 단어의 확률'을 계산합니다.

이 과정을 수백억 번 반복하면서 "언어의 패턴"을 배운 것입니다.

1. 단어를 숫자로 변환

문장 속 모든 단어를 컴퓨터가 이해할 수 있는 숫자로 바꿉니다

2. 확률을 계산한다.

다음에 올 수 있는 모든 단어의 확률을 계산합니다.

3. 확률이 높은 후보들 중에서 하나를 뽑아 연결

확률 1등만 고르는 게 아니라, 상황에 따라 다양한 단어를 뽑아 자연스러운 문장을 만듭니다.

Enter text:

One, two, three,

One 3198 11 , two 734 11 , three 1115 11

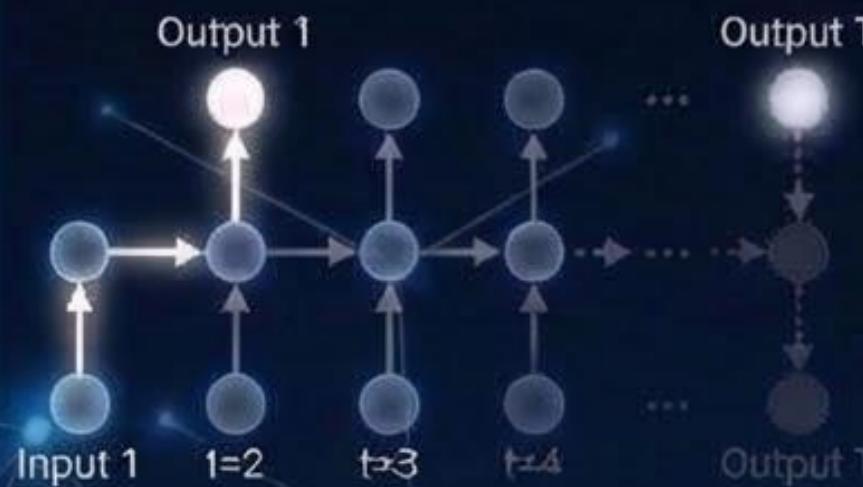
3198 11 734 11 1115 11

Prediction

#	probs	next token ID	predicted next token
0	46.44%	1440	four
1	7.48%	290	and
2	7.31%	1936	five
3	2.66%	393	or
4	2.54%	2237	six
5	2.09%	1115	three
6	1.86%	3863	maybe
7	1.62%	345	you
8	1.23%	257	a
9	0.92%	530	one

RNN의 문제점은?

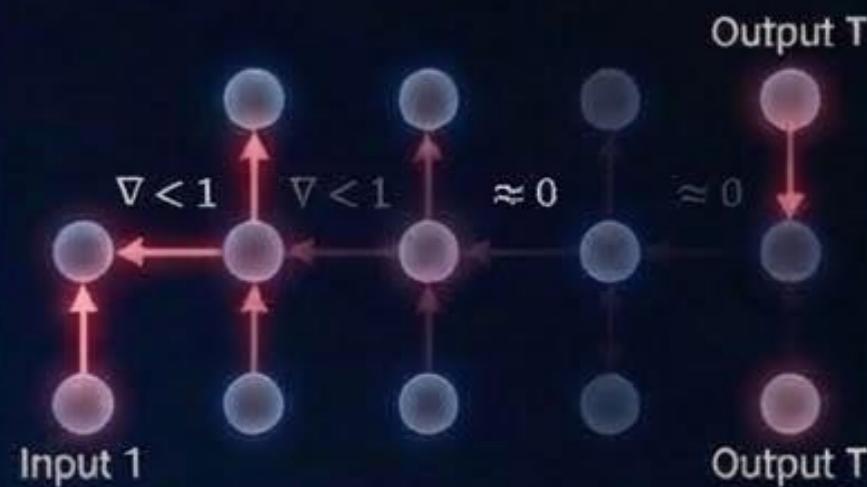
1. 장기 의존성 문제 (Long-Term Dependency Problem)



긴 시퀀스에서 초기 정보가 나중의 결과에 영향을 미치지 못하는 현상.

예시. 긴 문장의 시작 부분에 나온 주어 정보를 문장의 끝의 동사 해석에 활용하기 어려움.

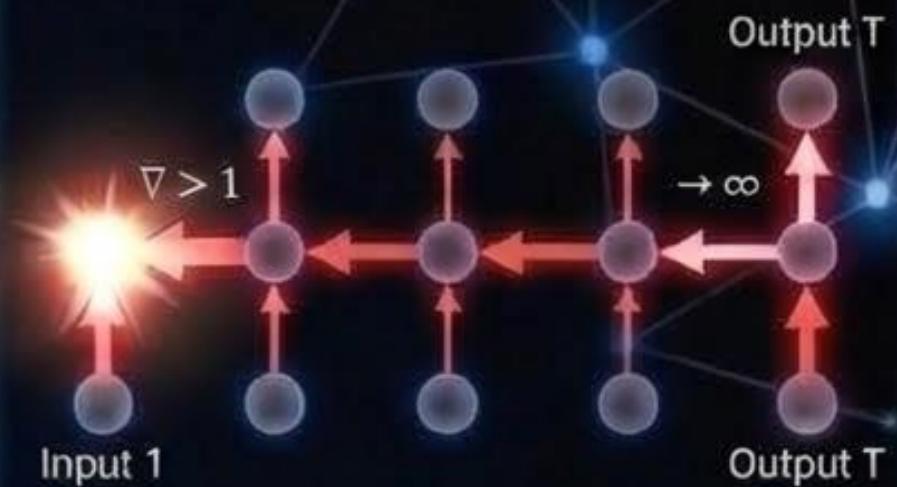
2. 그래디언트 소실 (Gradient Vanishing)



역전파 과정에서 그래디언트(기울기)가 점점 작아져 0에 가까워지는 현상.

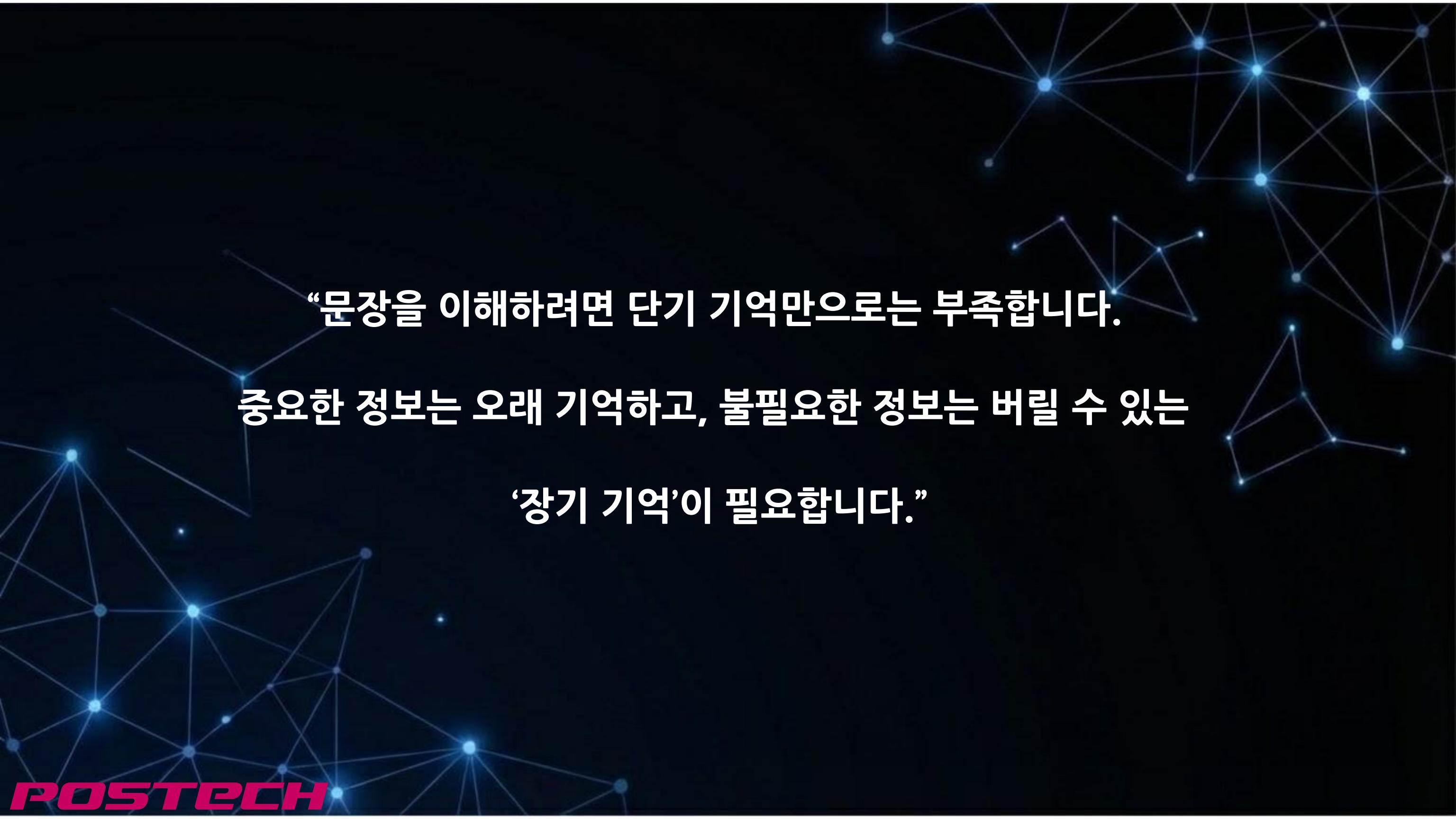
초기 층의 가중치가 업데이트되지 않아 학습이 제대로 이루어지지 않음

3. 그래디언트 폭발 (Gradient Exploding)



역전파 과정에서 그래디언트가 너무 커져 발산하는 현상.

가중치가 비정상적으로 커져 학습이 불안정해지고 발산함.

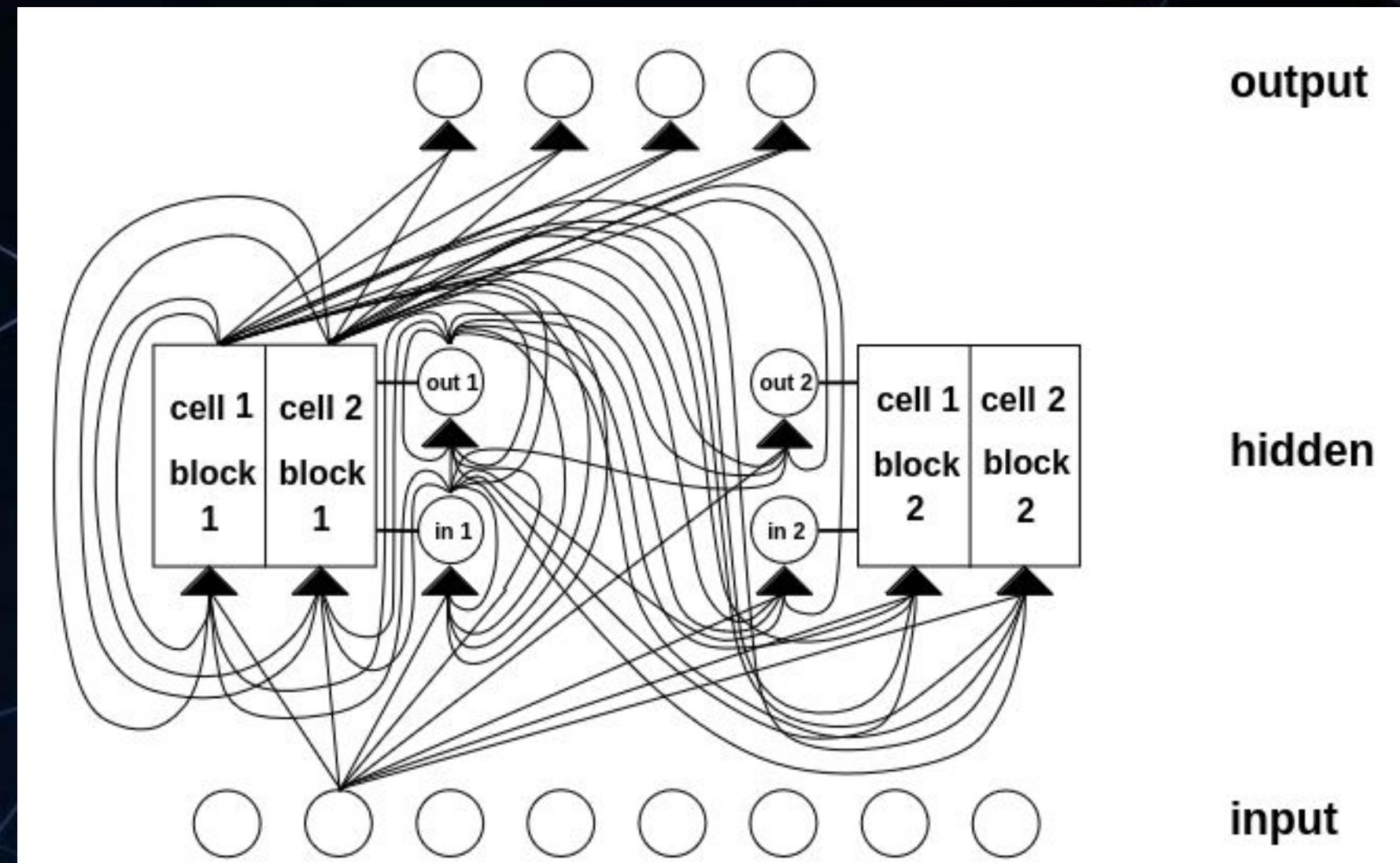


“문장을 이해하려면 단기 기억만으로는 부족합니다.

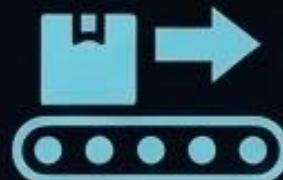
중요한 정보는 오래 기억하고, 불필요한 정보는 버릴 수 있는

‘장기 기억’이 필요합니다.”

LSTM: 중요한 정보를 오래 기억하는 구조



LSTM의 핵심: 셀 상태와 게이트를 통한 정보 제어



셀 상태 (Cell State)

정보가 변질 없이 흘려갈 수 있는 ‘고속도로’. 장기 기억을 저장하고 전달합니다.



망각 게이트 (Forget Gate)

이전 상태에서 불필요한 정보를 삭제합니다.



입력 게이트 (Input Gate)

새로운 정보 중 중요한 것을 선택하여 셀 상태에 추가합니다.



출력 게이트 (Output Gate)

현재 셀 상태를 바탕으로 최종 출력을 결정합니다.

망각 게이트
(Forget Gate)



입력 게이트
(Input Gate)



출력 게이트
(Output Gate)



망각 게이트
(Forget Gate)

입력 게이트
(Input Gate)

출력 게이트
(Output Gate)

LSTM의 한계와 Attention의 등장: 모든 것을 기억할 수는 없다

LSTM의 문제점: 장기 의존성 & 정보 병목



핵심은
'필요한 정보에만
집중'하는 것!

Attention의 등장: 필요한 정보에 스포트라이트



- 문장이 길어질수록 초기 정보 소실 (Vanishing Gradient)
- 고정된 크기의 백터에 모든 정보 압축 한계

- 모든 입력 정보를 동시에 고려하여 가중치 부여
- 문맥에 따라 동적으로 정보에 집중 (Attention Mechanism)

All you need is attention: 핵심 정보에 집중하라!

나는 학교에 가서 공부를 했다



Attention
Mechanism

핵심 내용 파악:
공부, 학교



중요한 정보에 ‘헝겊펜’ 칠하기

긴 글을 읽을 때, 모든 단어를 똑같이 중요하게 여기지 않습니다.
핵심 단어에 더 ‘집중’합니다.



단어 간의 관계 파악

문장의 각 단어가 서로 어떤 연관이 있는지
계산하여 문맥을 이해합니다.



효율적인 정보 처리

RNN처럼 순서대로 처리할 필요 없이,
중요한 정보를 동시에 파악하여 속도가
빠릅니다.

ChatGPT는 어떻게 학습할까?

ChatGPT는 문장을 작은 단위(토큰)로 나누고 숫자로 바꾼 뒤, ‘다음에 올 표현의 확률’을 계산합니다.

이 과정을 수백억 번 반복하면서 언어의 패턴과 지식을 배웁니다.

사전 학습(Pre-training)

인터넷의 글, 책, 코드 등을 읽으며 '언어의 규칙'을 익힘

미세 조정(Fine-tuning)

사람이 평가한 데이터를 통해 '좋은 대답'을 배우는 단계

ChatGPT가 '생각'하는 방식 – Chain of Thought

ChatGPT는 단번에 답을 내는 대신, 중간 단계를 말로 풀어가며 사고(Chain of Thought) 합니다.

이렇게 하면 복잡한 문제도 더 정확히 해결할 수 있습니다.

1. 기본 원리 : 다음 단어 예측

GPT는 기본적으로 "다음에 어떤 단어가 올까?"만 반복적으로 학습합니다.

2. 데이터 속 사고과정

훈련 데이터에 단계별 설명 예시가 포함되어 있습니다.

예: "12 + 37 = ? 먼저 10의 자리끼리 더하면 40, 1의 자리끼리 더하면 9, 그래서 49."

3. 패턴 학습과 습득

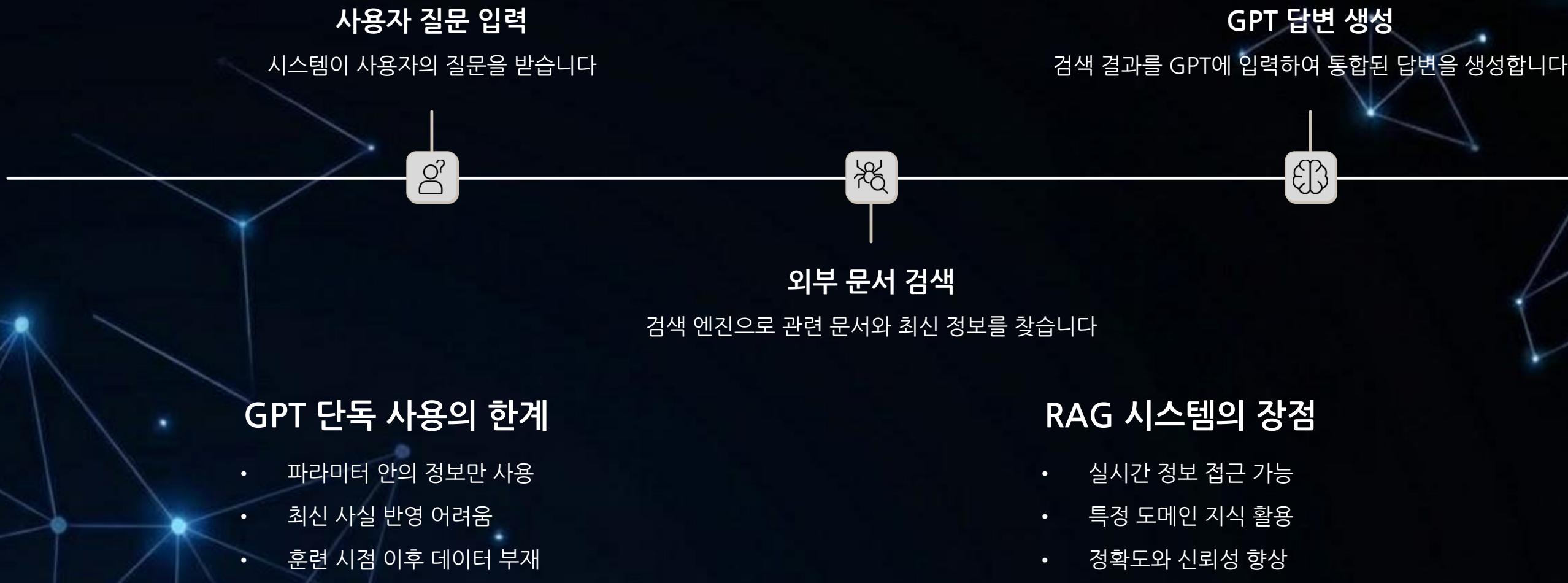
이런 예시를 반복적으로 보면서 모델은 "답을 내기 전에 사고의 단계를 적어야 한다"는 패턴을 자연스럽게 습득합니다.

4. 의도적 강화 : Fine-tuning

연구자들은 "Let's think step by step."와 같은 prompt로 모델을 추가 학습시켜 단계적 사고를 더욱 강화합니다.

Retrieval-Augmented Generation(RAG): 외부 구조로 추가된 기능

RAG는 GPT 자체의 기능이 아니라, GPT를 감싸는 외부 시스템 구조로 최신 정보를 활용할 수 있게 합니다.

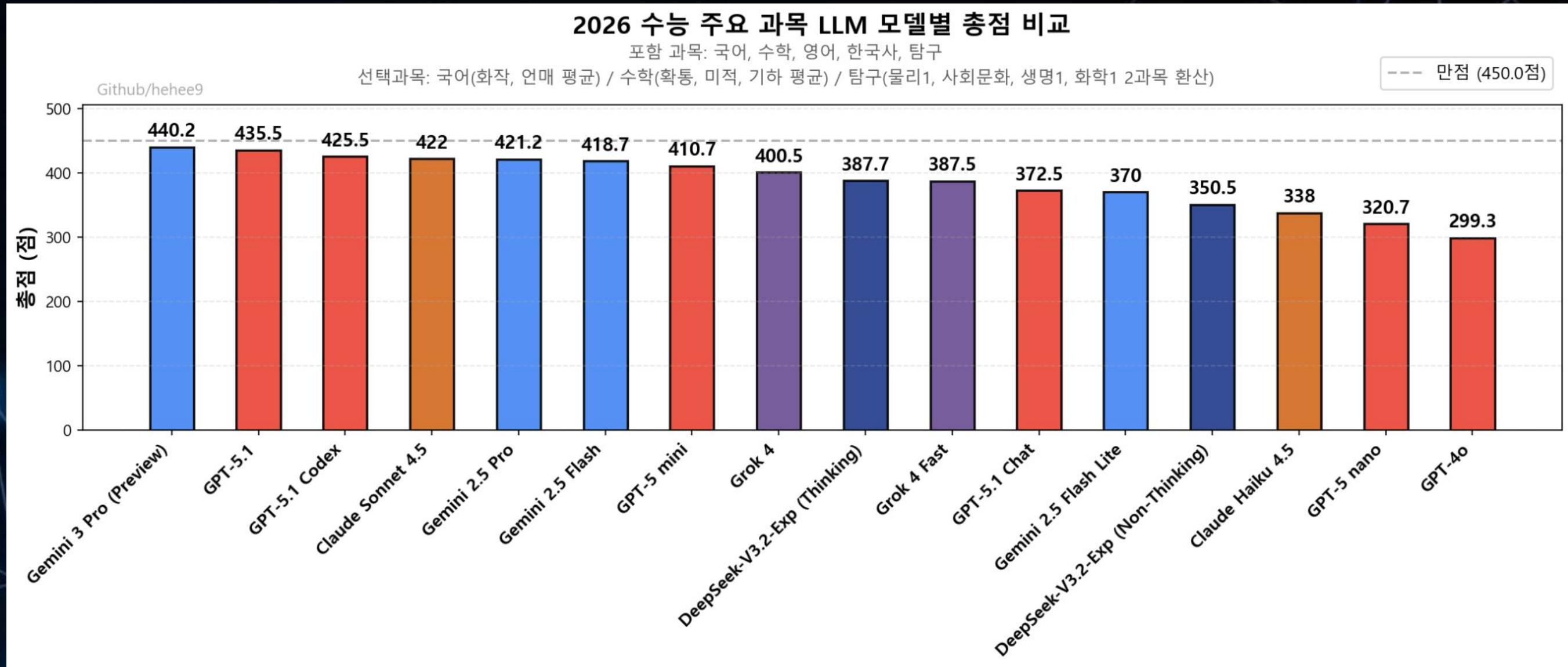


RAG는 GPT가 외부 지식을 참조할 수 있도록 보조하는 구조입니다. GPT 본체는 변하지 않지만, RAG 시스템이 GPT를 감싸서 능력을 확장합니다.

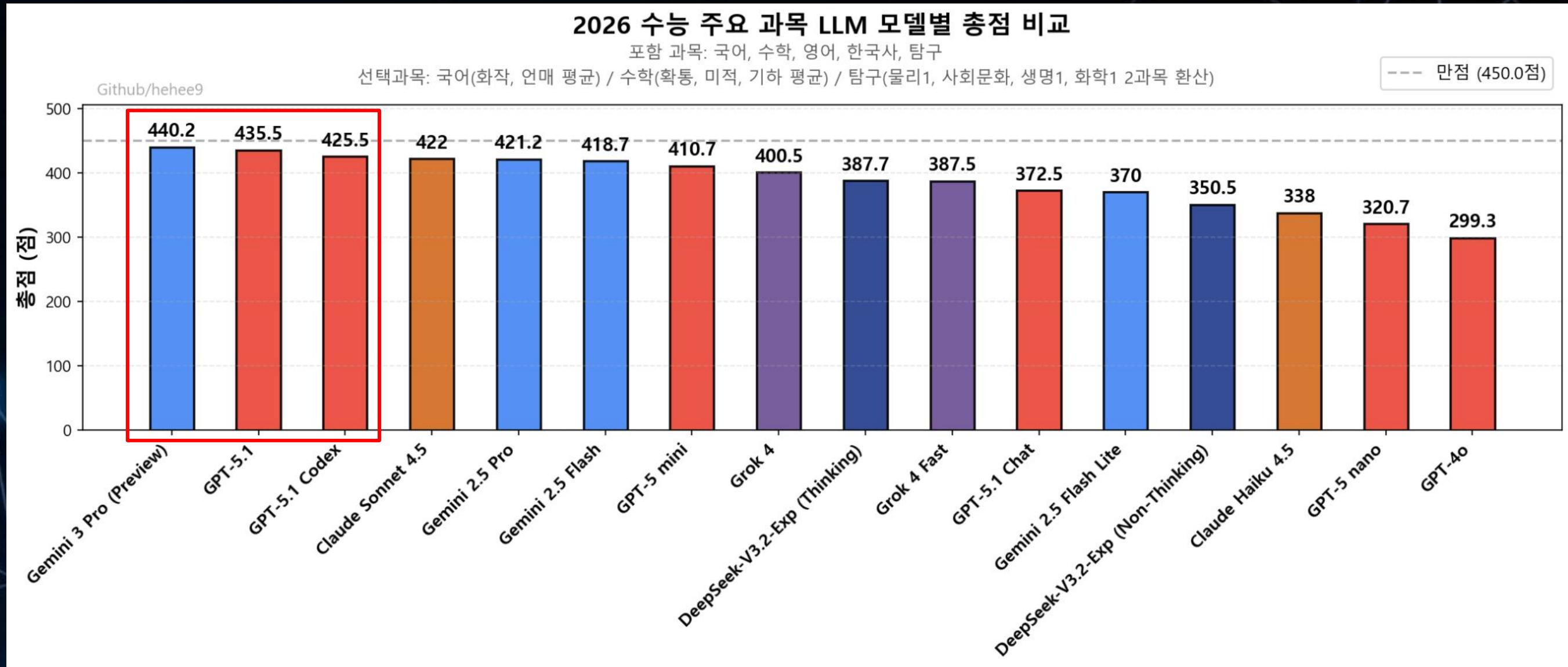


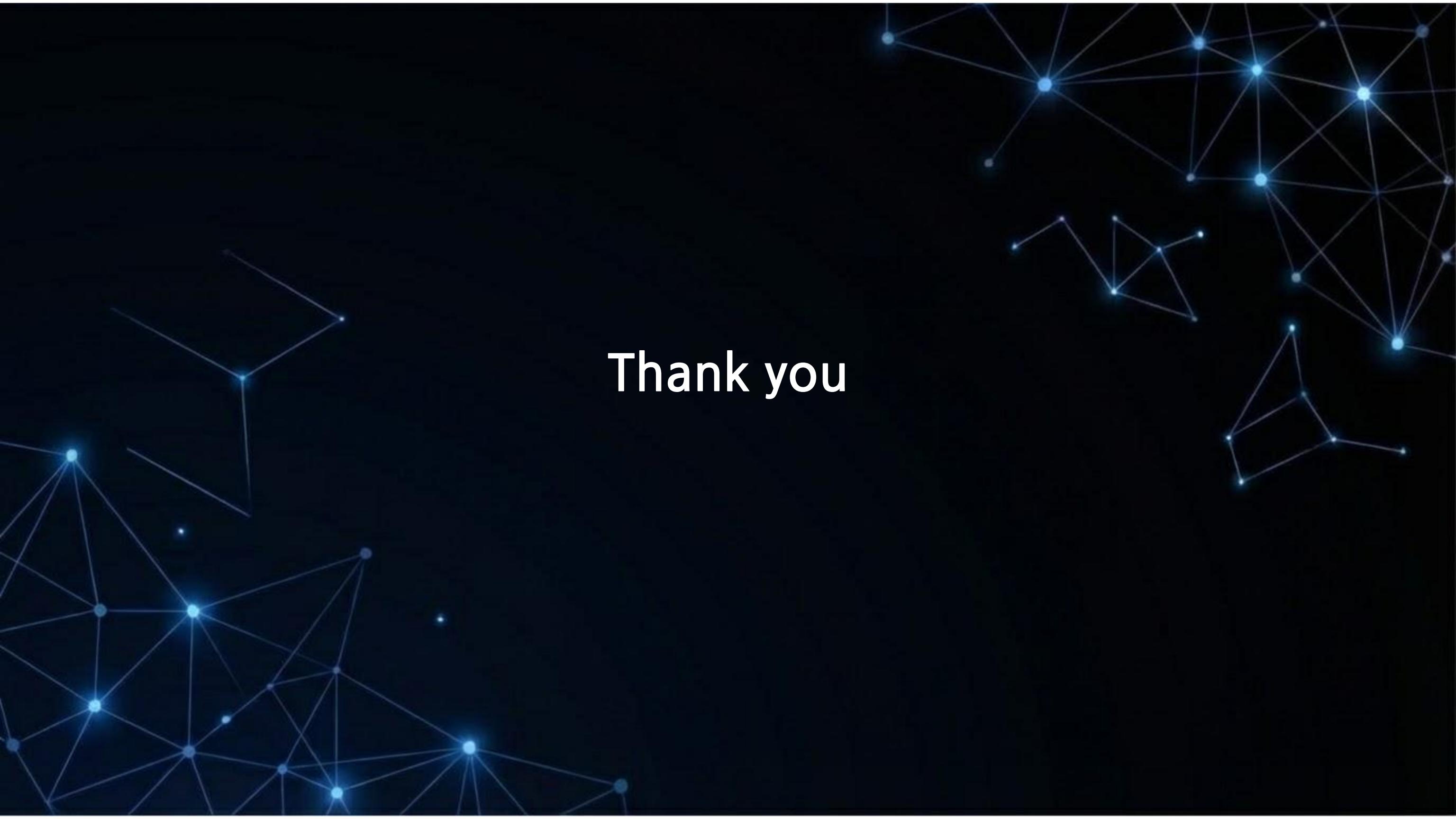
LLM을 이용해서 2026 수능을 풀어본다면?

LLM을 이용한 2026 수능 문제 풀이 결과



LLM을 이용한 2026 수능 문제 풀이 결과





Thank you