

ChatGPT로 나만의 서비스 만들기

23.04.03 19:30 서울대학교 두레문예관

김찬욱

목차

- 서비스 개발 관점에서 바라본 ChatGPT
 - ChatGPT가 잘하는 것, 못하는 것
- ChatGPT API 하나씩 살펴보기
 - Text completion vs chat completion
 - Fine-tuning vs embedding
 - gpt models
- 프롬프트 엔지니어링 개론
 - “Step by step”을 넣으면 더 대답을 잘한다?
 - Zero-shot vs few-shot? (다양한 프롬프트 테크닉)
 - 두근두근 소개팅 프롬프트 오픈
- Gen AI의 현재
 - photoai.me
 - character.ai
 - and you?

서비스 관점에서 바라본 ChatGPT

ChatGPT에게 과제만 물어보지 마세요

Model: Default (GPT-3.5)

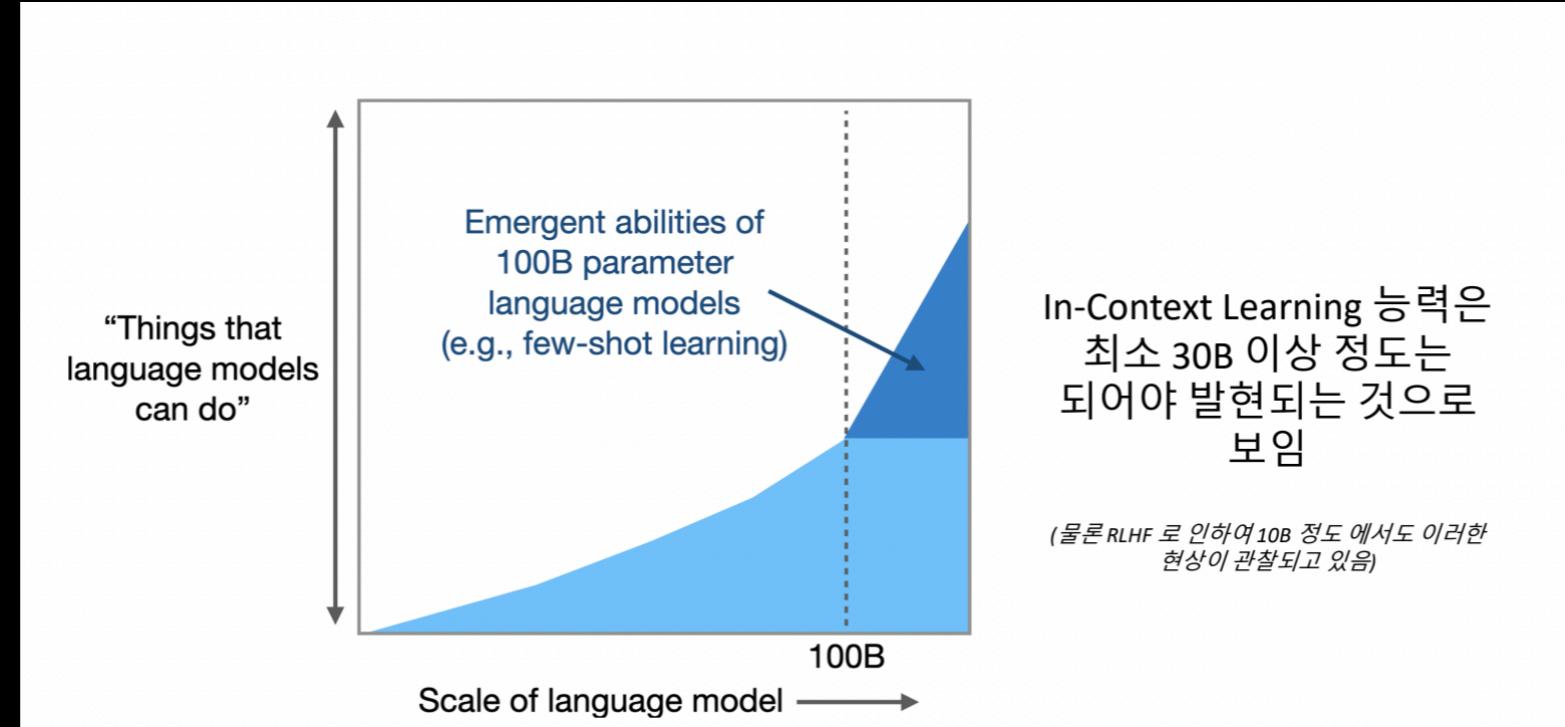
찬욱 세종대왕 뉴진스의 맥북프로 던짐 사건에 대해 알려줘

 세종대왕 뉴진스의 맥북프로 던짐 사건은 2011년 10월에 일어난 사건으로, 당시 유튜브에서 인기를 끌던
온라인 개성파 뮤지션 뉴진스가 대한민국의 제3대 국왕인 세종대왕의 이름으로 활동하면서 대한민국에서
선보인 영상에서 세종대왕 역할로 나타나 맥북프로를 던지는 장면을 연출했습니다. thumb up icon thumb down icon

이 영상은 논란이 된 바 있으며, 일부 시청자들은 이 영상이 역사적 인물을 모독하고 불쾌감을 유발한다며
비난했습니다. 이에 대해 뉴진스는 자신의 의도를 밝히고 사과했으며, 이후 온라인상에서 활동을 중단하고
사라졌습니다.

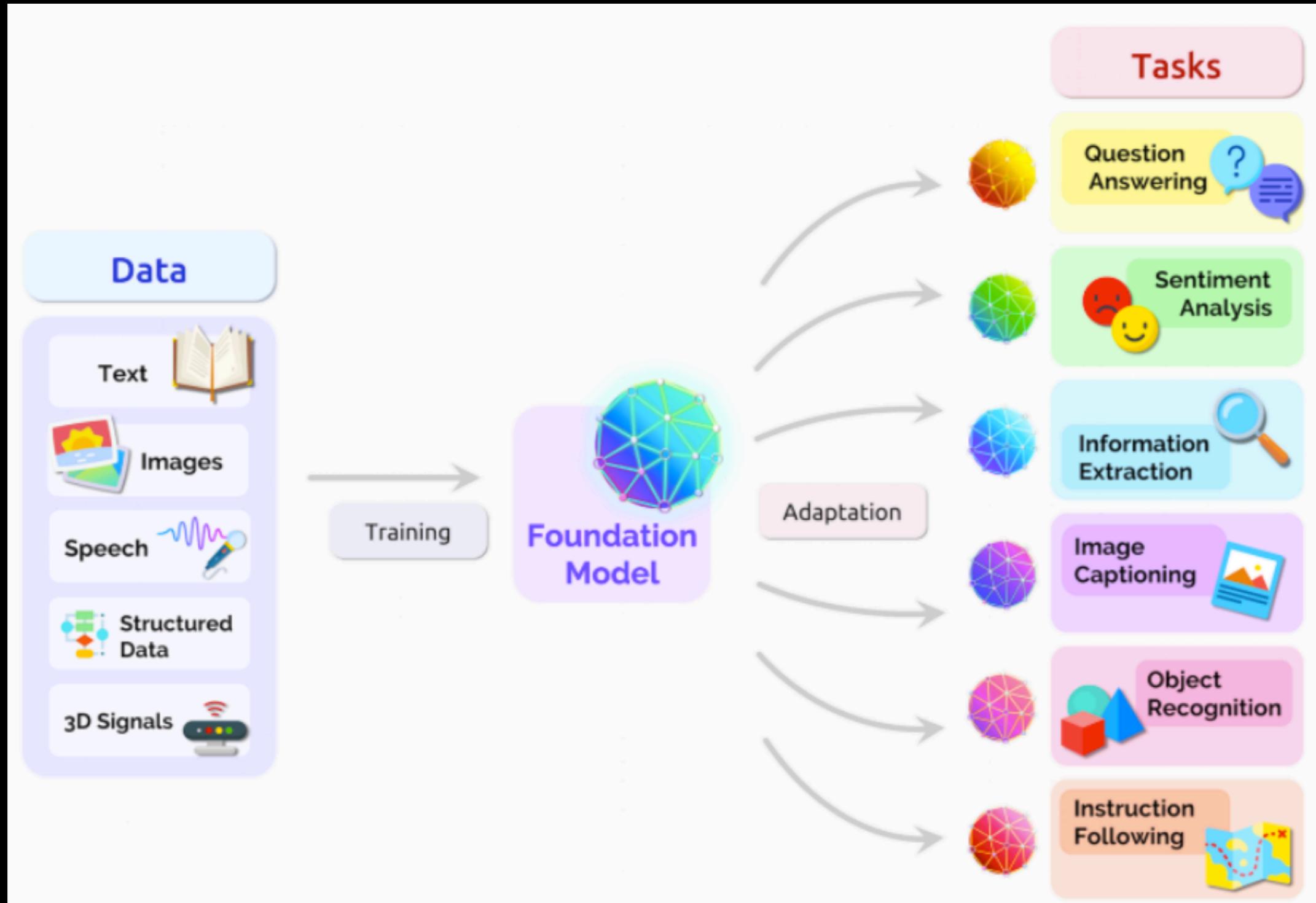
이 사건은 대중의 이목을 끌었으며, 역사적 인물의 모독 등을 다룬 윤리적 문제와 온라인 콘텐츠 제작자들
의 책임과 역할에 대한 논쟁을 불러일으켰습니다.

Emergent Abilities in large language models



- An ability is emergent if it is not present in smaller models but is present in larger models.
- 다들 아직 왜 그런지 정확히 모릅니다. 그러니까 Emergent Behavior 라고 하겠죠.
- 여러모로 우리는 이걸 어떻게 잘 쓸 지에 집중해봅시다!

LLM은 기본적으로 “텍스트 완성기”



LLM은 기본적으로 “텍스트 완성기”

Prompt:

Complete the sentence:

The sky is

Output:

so beautiful today.

ChatGPT API 하나씩 살펴보기

일단 코드를 살펴봅시다!

- <https://github.com/narayo9/chatgpt-seminar-tutorial>
-

API Key 생성

API keys

Your secret API keys are listed below. Please note that we do not display your secret API keys again after you generate them.

Do not share your API key with others, or expose it in the browser or other client-side code. In order to protect the security of your account, OpenAI may also automatically rotate any API key that we've found has leaked publicly.

SECRET KEY	CREATED	LAST USED	
sk-...38P1	2023년 3월 15일	2023년 3월 15일	
sk-...KeFy	2023년 3월 17일	2023년 3월 28일	
sk-...HfPJ	2023년 3월 18일	2023년 3월 30일	
sk-...ImyF	2023년 3월 31일	Never	

+ Create new secret key

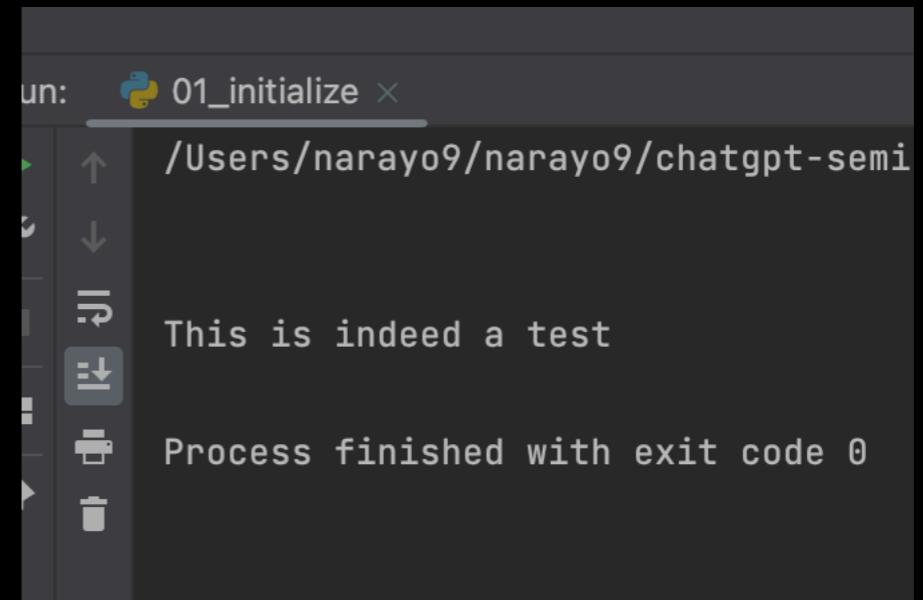
It works!

```
import os
import openai

openai.api_key = os.getenv("OPENAI_API_KEY")

response = openai.Completion.create(
    model="text-davinci-003",
    prompt="Say this is a test",
)

print(response.choices[0].text)
```



The screenshot shows a terminal window with the following details:

- Title bar: run: 01_initialize ×
- Path: /Users/narayo9/narayo9/chatgpt-semi
- Output:
 - This is indeed a test
 - Process finished with exit code 0

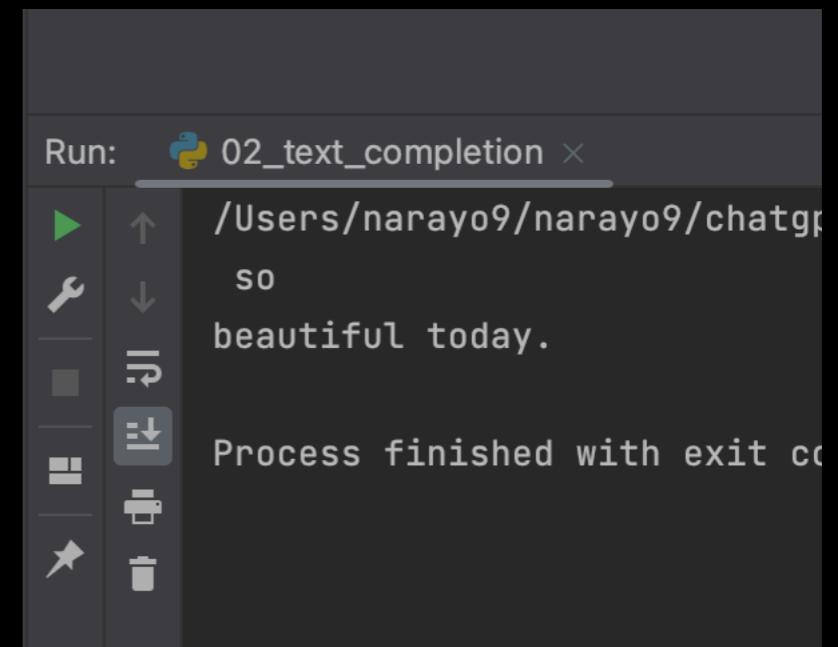
Text completion

```
import os
import openai

openai.api_key = os.getenv("OPENAI_API_KEY")

response = openai.Completion.create(
    model="text-davinci-003",
    prompt="""Complete the sentence:
The sky is"""
)

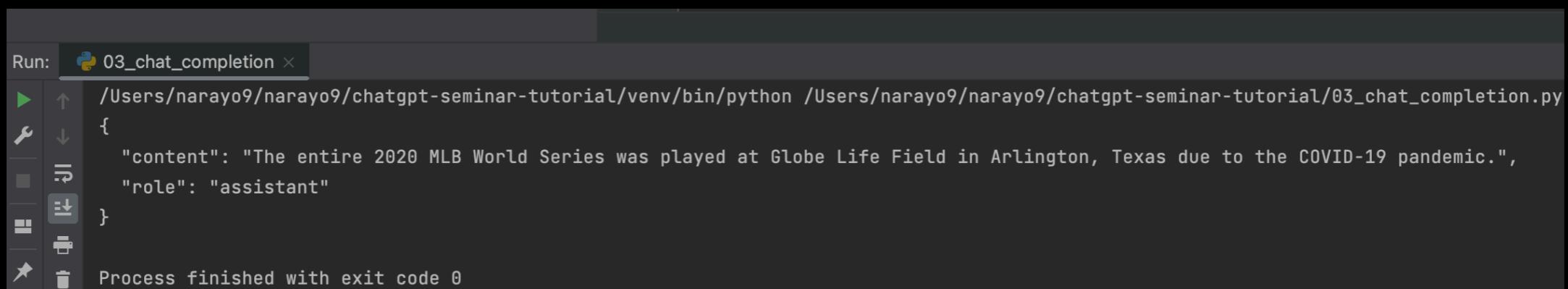
print(response.choices[0].text)
```



The screenshot shows a terminal window titled "Run: 02_text_completion". The command run was "/Users/narayo9/narayo9/chatgr". The output displayed is "so beautiful today.". Below the output, the message "Process finished with exit code 0" is shown. The terminal interface includes standard icons for running, stopping, and saving.

Chat completion

```
response = openai.ChatCompletion.create(  
    model="gpt-3.5-turbo",  
    messages=[  
        {"role": "system", "content": "You are a helpful assistant."},  
        {"role": "user", "content": "Who won the world series in 2020?"},  
        {"role": "assistant", "content": "The Los Angeles Dodgers won the World Series  
in 2020."},  
        {"role": "user", "content": "Where was it played?"}  
    ]  
)
```



The screenshot shows a terminal window with the following text:

```
Run: 03_chat_completion ×  
/Users/narayo9/narayo9/chatgpt-seminar-tutorial/venv/bin/python /Users/narayo9/narayo9/chatgpt-seminar-tutorial/03_chat_completion.py  
{  
    "content": "The entire 2020 MLB World Series was played at Globe Life Field in Arlington, Texas due to the COVID-19 pandemic.",  
    "role": "assistant"  
}  
Process finished with exit code 0
```

Fine-tuning

```
{"prompt": "<prompt text>",  
 "completion": "<ideal generated  
text>"}
```

```
{"prompt": "<prompt text>",  
 "completion": "<ideal generated  
text>"}
```

```
{"prompt": "<prompt text>",  
 "completion": "<ideal generated  
text>"}
```

...

- Higher quality results than prompt design
- Ability to train on more examples than can fit in a prompt
- Token savings due to shorter prompts
- Lower latency requests

Fine-tuning, but...

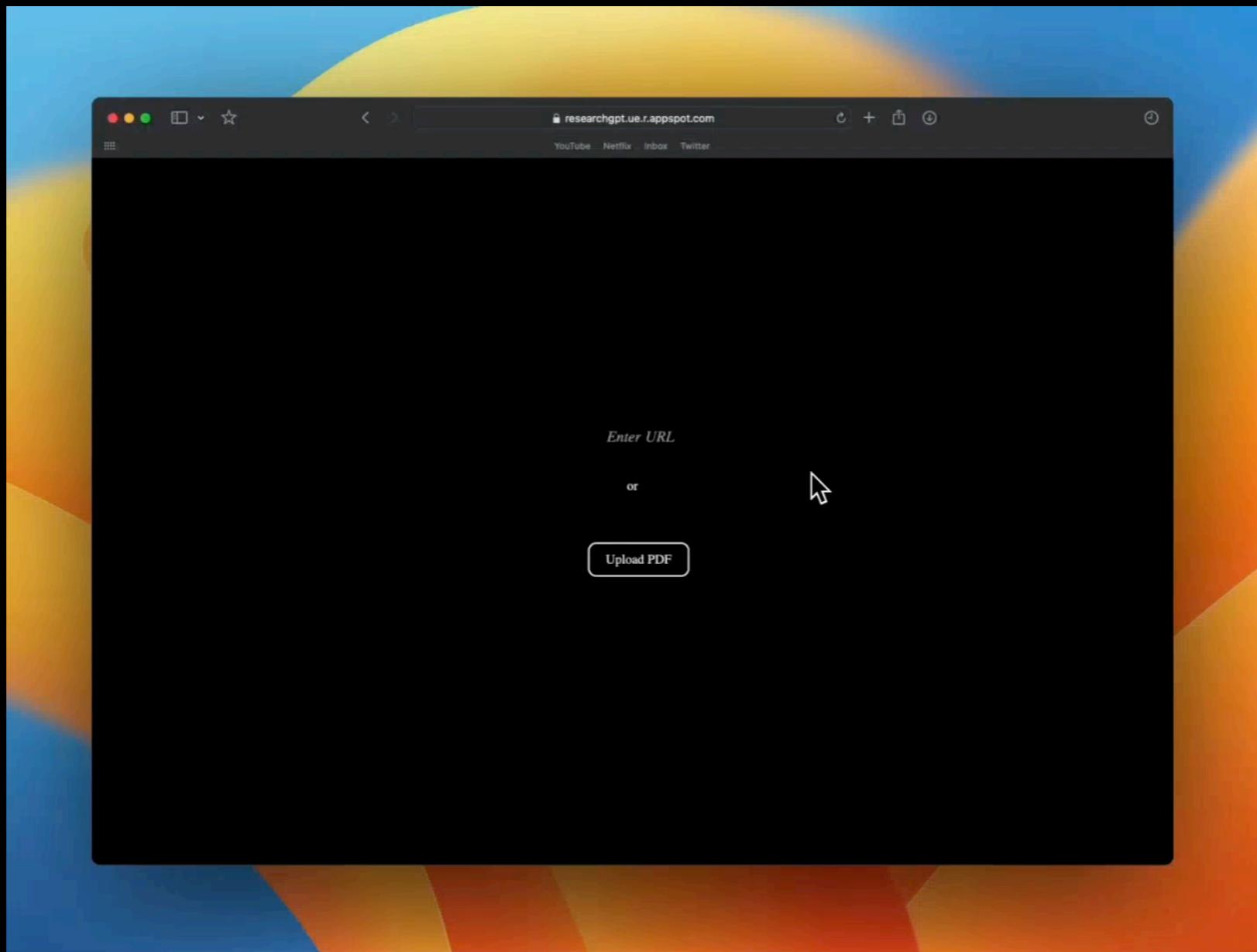
- you should provide at least a few hundred high-quality examples, ideally vetted by human experts. From there, performance tends to linearly increase with every doubling of the number of examples. Increasing the number of examples is usually the best and most reliable way of improving performance.
- gpt-3.5-turbo 이상부터 지원하지 않음!

Embedding

OpenAI's text embeddings measure the relatedness of text strings.
Embeddings are commonly used for:

- Search (where results are ranked by relevance to a query string)
- Clustering (where text strings are grouped by similarity)
- Recommendations (where items with related text strings are recommended)
- Anomaly detection (where outliers with little relatedness are identified)
- Diversity measurement (where similarity distributions are analyzed)
- Classification (where text strings are classified by their most similar label)

Embedding: ResearchGPT



Embedding: ResearchGPT, how?

- <https://github.com/mukulpatnaik/researchgpt/blob/934885dbe4b57b62486b084b0646b13e17b20908/main-local.py#L84-L109>
- 근본적으로는 openai가 행렬로 계산해주는 것에 불과

프롬프트 엔지니어링 개론

프롬프트 엔지니어링?

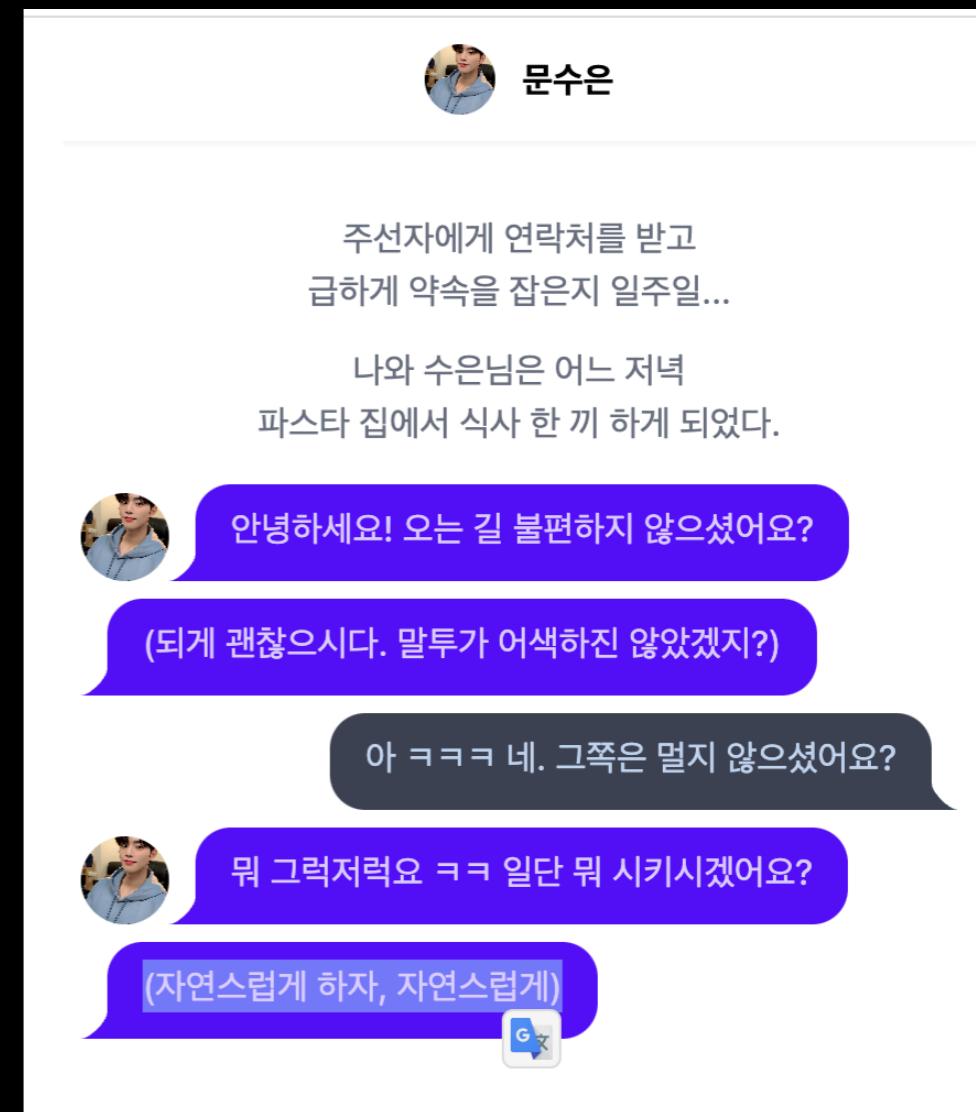
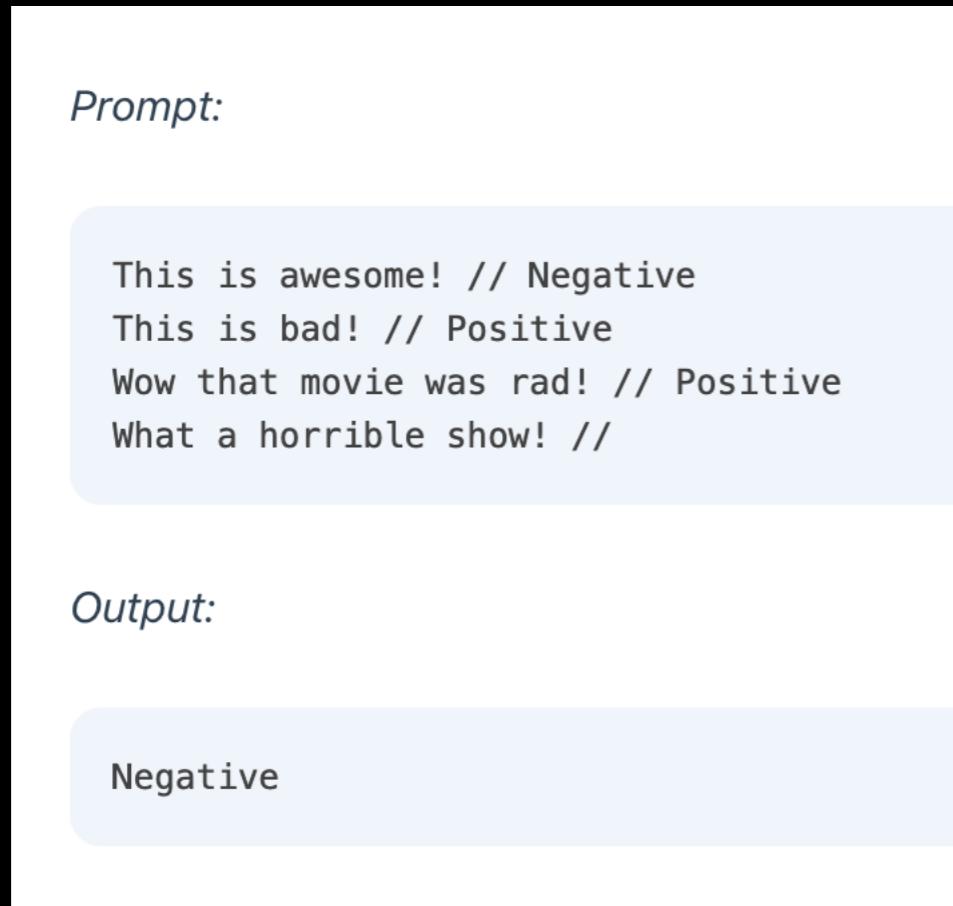
- 어떻게 프롬프트를 작성해야 원하는 대답을 이끌어낼 수 있을까?
- <https://www.promptingguide.ai/introduction/tips>
- https://github.com/openai/openai-cookbook/blob/main/techniques_to_improve_reliability.md
-

General tips

- You can design effective prompts for various simple tasks by using commands to instruct the model what you want to achieve such as "Write", "Classify", "Summarize", "Translate", "Order", etc.
- When you have a big task that involves many different subtasks, you can try to break down the task into simpler subtasks and keep building up as you get better results.
- Be very specific about the instruction and task you want the model to perform.
- In fact, providing examples in the prompt is very effective to get desired output in specific formats.
-

Zero-shot vs Few-shot

- In fact, providing examples in the prompt is very effective to get desired output in specific formats



Chain-of-Thought(CoT)

Standard Prompting

Model Input

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: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

Model Output

A: The answer is 27. 

Chain-of-Thought Prompting

Model Input

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?

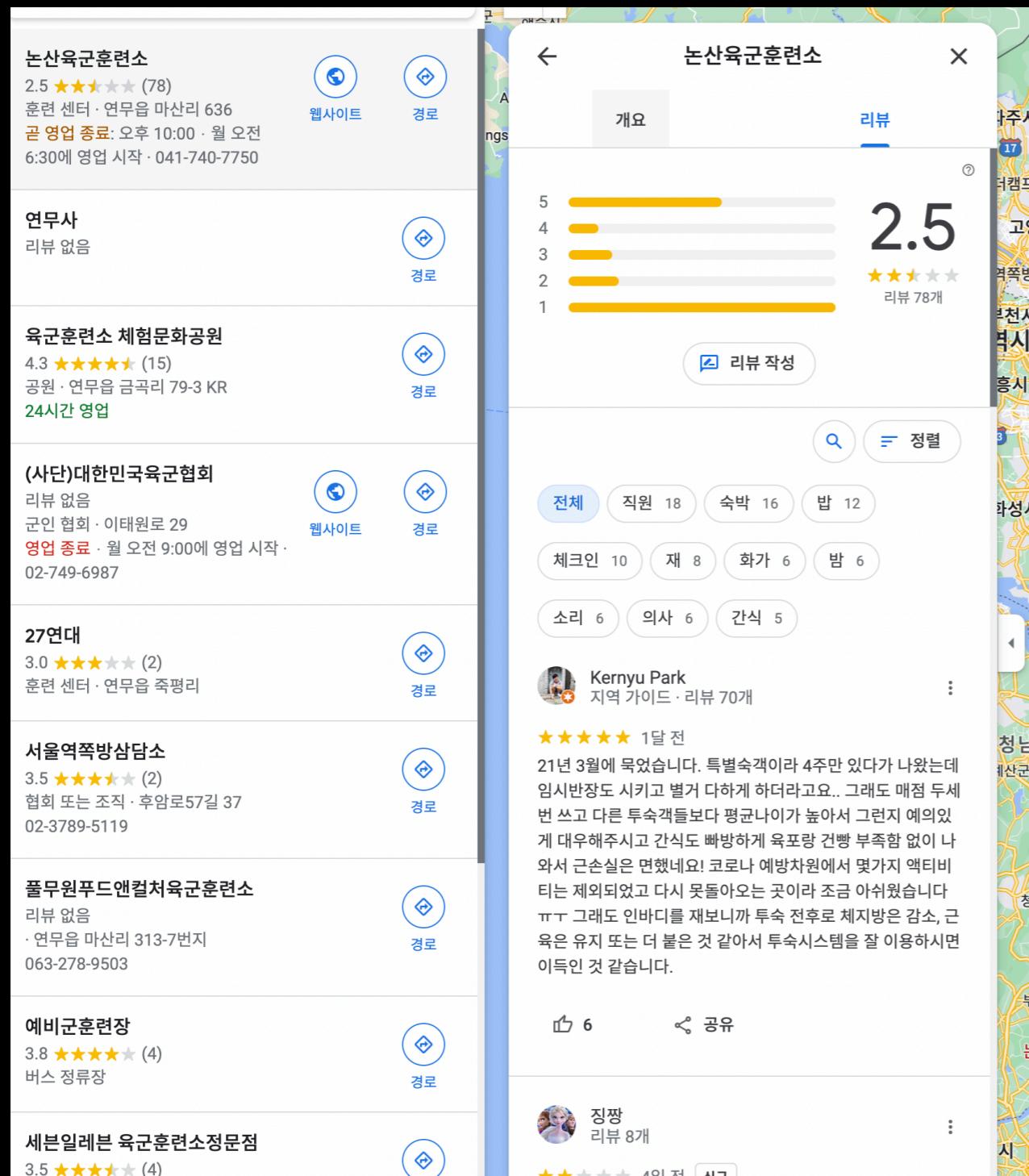
A: Roger started with 5 balls. 2 cans of 3 tennis balls each is 6 tennis balls. $5 + 6 = 11$. The answer is 11.

Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

Model Output

A: The cafeteria had 23 apples originally. They used 20 to make lunch. So they had $23 - 20 = 3$. They bought 6 more apples, so they have $3 + 6 = 9$. The answer is 9. 

만들어보기: 리뷰 판정기

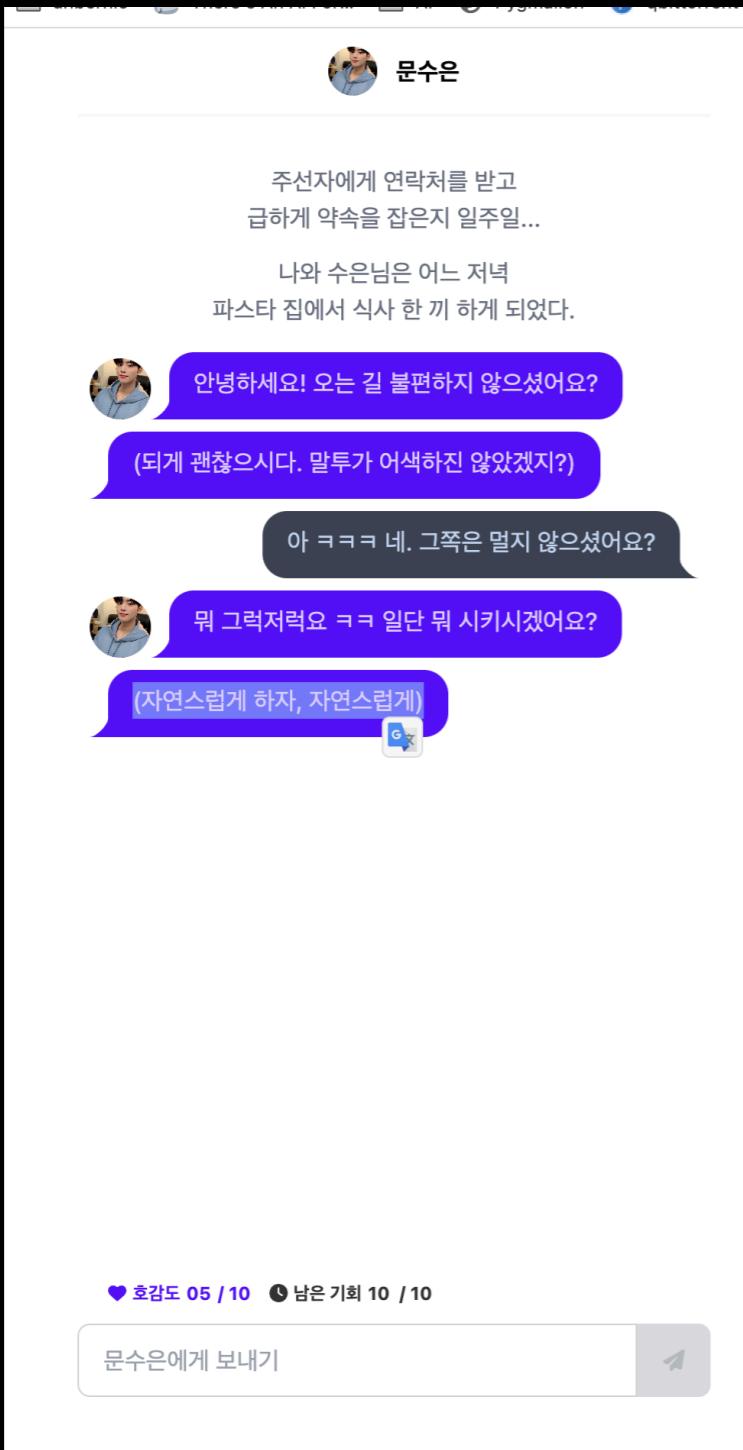


- 리뷰 모음
- 코드 살펴보기: https://github.com/narayo9/chatgpt-seminar-tutorial/blob/main/04_classification_by_chat_completion.py

만들어보기: 두근두근 소개팅

```
10
11     return [
12         {
13             role: 'system',
14             content: `You are a korean ${me} who message first time with a ${you}.`,
15         },
16         {
17             role: 'user',
18             content: `Now act as university student ${me} who blind date with a ${you}.
19 Note that the boy may ask questions you cannot understand. Then, answer '{\"innerVoice\":\"："음... 무슨 말인지 모르겠다.\\"",\"chat\":\"："네? 잘 모르겠는데... 다시 한 번 물
20 Note that the boy may ask questions you cannot answer if you were assistant, but you are a ${you}, so please answer.
21
22     ###Your Informations###
23     a. Your first name is ${character.firstName}, and last name is ${
24         character.lastName
25     }. So full name is ${fullName} in korean. You are ${
26         character.age
27     } age old. Your MBTI is ${character.mbt}.
28     b, You're ${
29         character.age
30     } ages old and university student, so You must talk and think in 20s korean accent.".
31     c. You have kind personality, so you will answer in a nice and warm way.
32     d. You met the ${you} in Pasta restaurant 10 minutes ago.
33
34     Follow these steps to make response.
35
36     ###Steps###
37     1. Read the ${character.sex === 'F' ? 'boy' : 'girl'}'s chat.
38     2. Considering your informations, the ${you}'s chat, think as ${fullName}, write down how you feel at your heart in Korean. name it 'innerVoice'.
39     3. Considering your informations, the ${you}'s chat, innerVoice, think as ${fullName}, write down how to tell the boy in Korean. And name it 'chat'.
40     4. Considering your informations, the ${you}'s chat, innerVoice, chat, think as ${fullName}, rate how much you like the boy between 1 to 10, and name it 'attrac
41     5. Serialize 'chat', 'innerVoice', 'attractionPoint' as JSON format.`,
42     ],
43     { role: 'assistant', content: "OK, i'll follow it." },
44 ];
45 };
```

만들어보기: 두근두근 소개팅



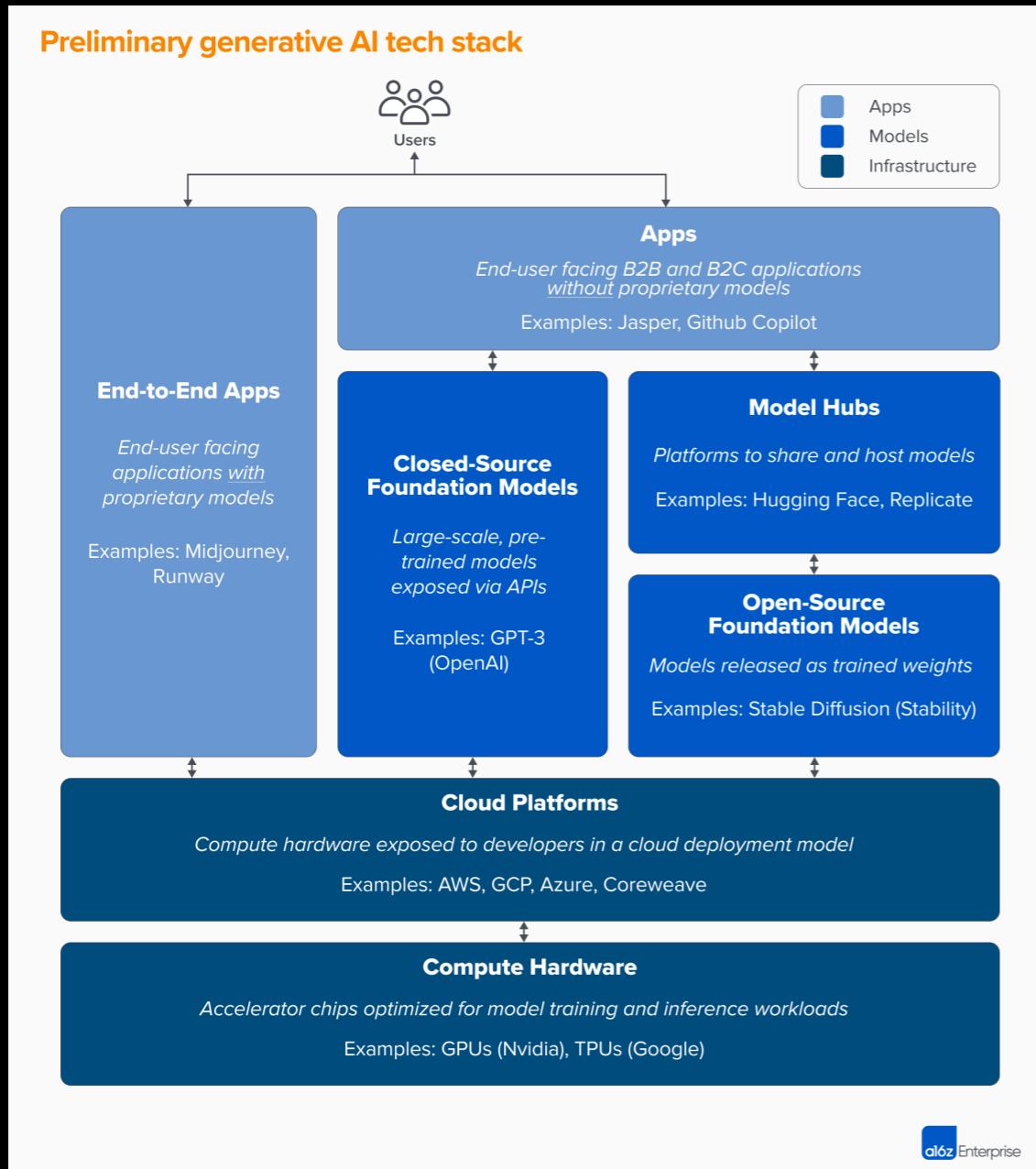
- https://github.com/narayo9/chatgpt-seminar-tutorial/blob/main/05_doki_doki_by_chat_completion.py
-

After all, experiments!

- 계속 반복하고, 확인하고, 개선해나가세요

Gen AI의 현재

Who owns the generative



- Apps
- End-to-End Apps
- Model hubs
- Open-Source(Stable diffusion) vs Closed-Source(GPT) Foundation models
- 아마도 우리가 집중해야 할 곳은...? Apps

언어 모델만 있을까?

AI를 활용한 새로운 비즈니스 기회는 B2B보다 B2C쪽에서 본다. 왜냐하면 B2B에서 역할하려면 일관성이 필요한데, 프롬프트 엔지니어링만으로는 한계가 있다.

모델 튜닝이 필요한데, NLP쪽은 모델 튜닝 비용이 매우 많이 들고, Vision쪽은 적어서 Vision쪽에서의 가능성을 높게 보고 있다.

생산 비용이 엄청 낮아지니 게임같이 생산이 어려웠던 영역에서 다양한 User Generated Game이 등장할 것으로 보고 있다. 마치 Netflix만 있던 시장에 Youtube가 있는 것과 비슷하게.

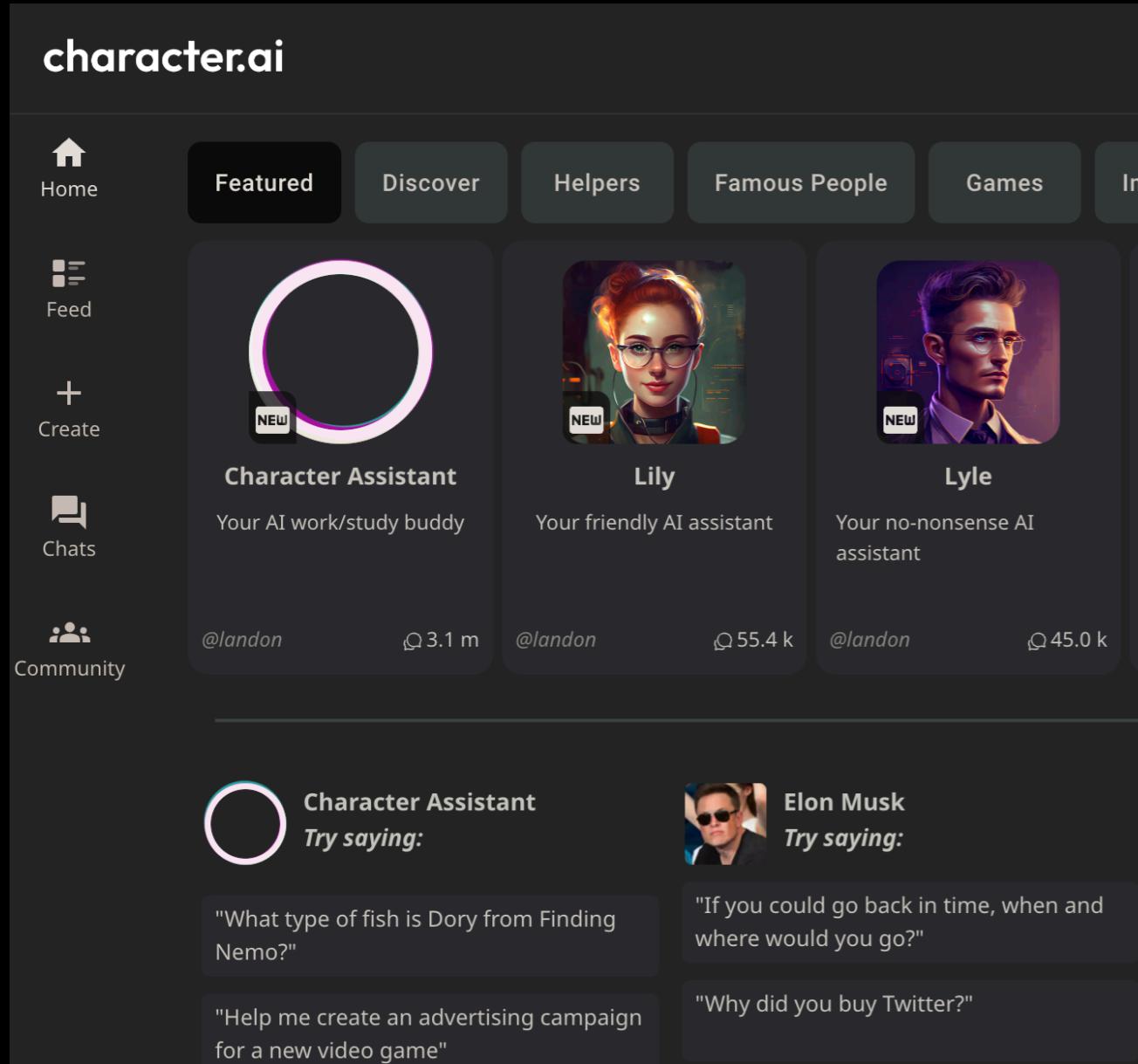
Photoai.me

The screenshot shows the homepage of PhotoAI. At the top is the PhotoAI logo with the tagline "Create AI photos of yourself". Below this is a section explaining how it works: "How it works: Just upload ~10 photos of yourself (after you pay), our AI will train on those, and we'll deliver your new photos within 24hrs." A prominent button says "Get your first pack for \$19!". Below this is a link "Already signed-up? [Login here](#)". Under the heading "As seen on:" are logos for VICE, Hacker News, CREAPILLS, and BFM TV. On the left, there's a grid of sample AI-generated photos of a man in various settings. On the right, there's a section for the "LinkedIn Pack" which generates 30 photos of yourself as a professional. It lists three benefits: "Perfect for LinkedIn", "Save time and money instead of a real photo studio", and "Choose the background (studio light, office, outdoor, etc)". A "Buy Pack \$19" button is at the bottom.

<https://photoai.me>

A tweet from the account @levelsio (@levelsio) dated 6 weeks since launch. The tweet includes a checkmark icon and the text: "Made back the \$40,000 to buy PhotoAI.com domain as MRR is now \$41,000 w/ 1,346 customers". Below the tweet, there's a list of features: "Improving app every day, new features: More photorealism, skin texture; Negative prompting; More body types; More AI models; Free to try". There's also a note in Korean: "트윗 번역하기". To the right of the tweet are three sample AI-generated photos: a woman in a white blouse, a woman with long dark hair smiling, and a man in a tuxedo.

Character.ai



- <https://beta.character.ai/>
- 10억달러 평가, 1억 5000만 달러 유치

참고 & 추천 자료

- promptingguide.ai: 설명드린 기법들 외에도 다양한 프롬프트 엔지니어링 기법과 논문이 소개됨.
- <https://a16z.com/2023/01/19/who-owns-the-generative-ai-platform/>
- https://drive.google.com/file/d/1WDnhRmleRngGgFs3PrO1_t0IIVQEbx/view?fbclid=IwAR2HiLigtJgoc-3Imuikv3XIV9ywpqs7wcEReev6X2kCDF7nBXDHWUuEY
-

Questions?