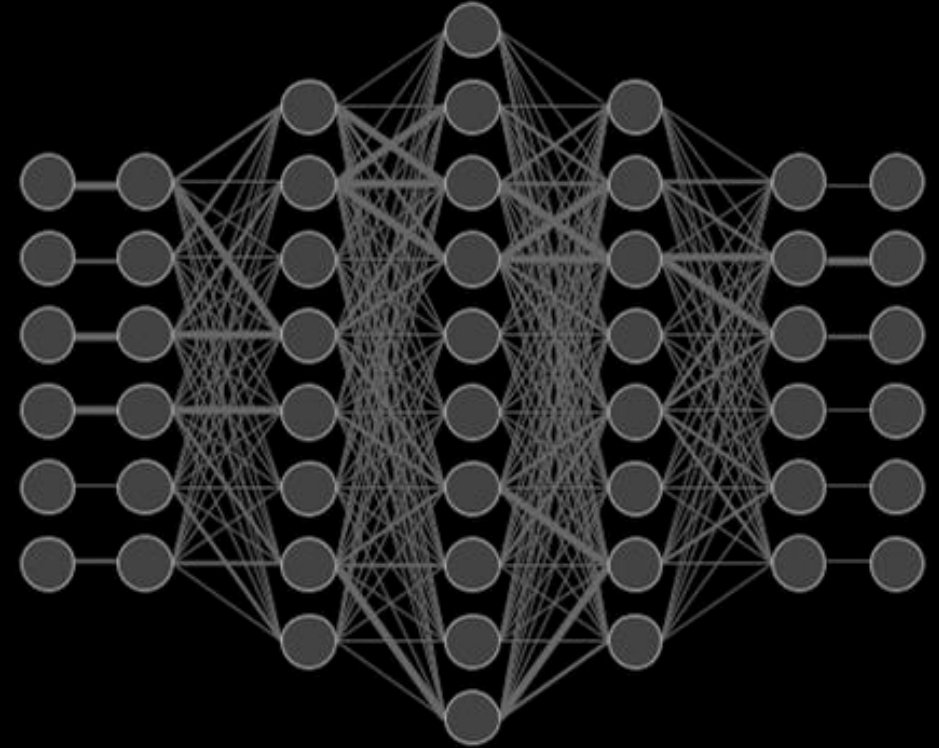


Azure OpenAI Update 2024 Q1



Microsoft 기술전략매니저 김현수

Agenda

1. Azure OpenAI 필수 정보
2. Azure OpenAI Update
3. Your SLMs and GPU Power
4. LLMOps - Prompt Flow
5. Azure OpenAI 모델 정보



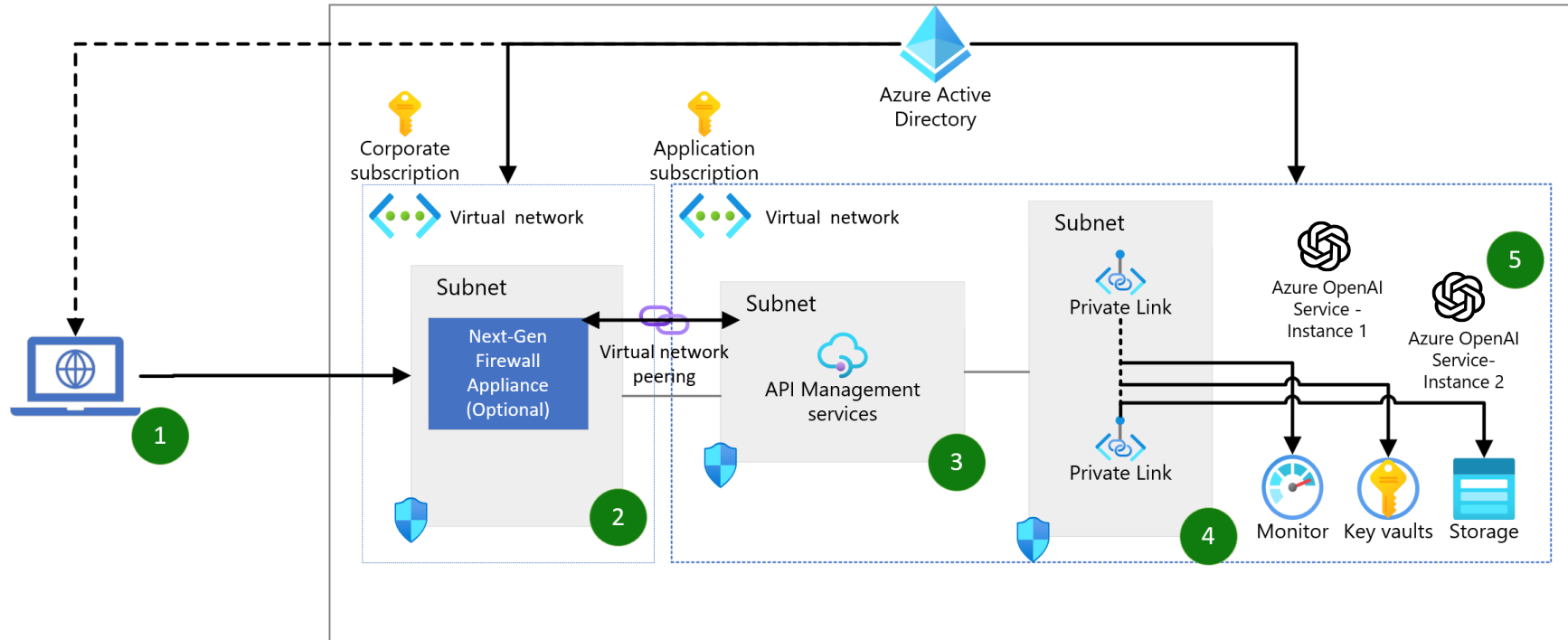
1. Azure OpenAI 필수 정보

Azure OpenAI Service

목적에 맞는 생성형 AI 기술을 채택

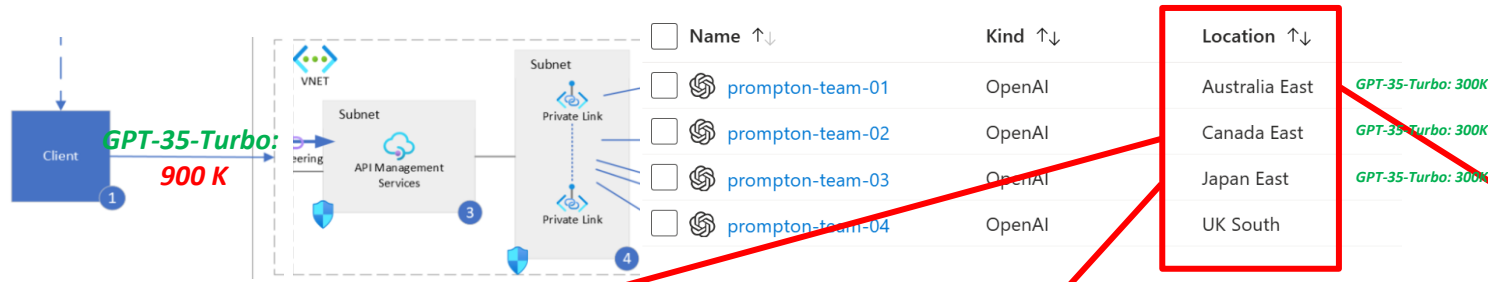


안정적 서비스를 위한 Enterprise OpenAI 환경



- 트래픽이 한도를 초과하는 경우 **API의 고가용성** 제공
- 다수 리전에 모델 배포하여 **기본 Quota 증가** 효과
- 책임감 있는 사용을 위한 로깅
- 다양한 사용자 그룹에 대한 세분화된 액세스 제어
- Azure Active Directory를 통한 역할 기반 액세스

리전별 모델 Quota 확장 방법



- 구독별로 설정 가능
- 리전별 OpenAI 리소스 생성 가능
 - 리소스별 모델 선택하여 배포 가능
 - 리소스별 모델의 기본 Quota 제공

Azure OpenAI Studio > Quotas

Canada East

GPT-35-Turbo: 300K TPM

Quota name	배포	Usage/Limit	Request quota
Tokens Per Minute (thousands) - GPT-35-Turbo		300 of 300 100%	
Tokens Per Minute (thousands) - GPT-35-Turbo-16K		300 of 300 100%	
Tokens Per Minute (thousands) - GPT-4		40 of 40 100%	
Tokens Per Minute (thousands) - GPT-4-32K		80 of 80 100%	
Tokens Per Minute (thousands) - Text-Embedding-Ada-002		350 of 350 100%	

Azure OpenAI Studio > Quotas

Japan East

GPT-35-Turbo: 300K TPM

Quota name	배포	Usage/Limit	Request quota
Tokens Per Minute (thousands) - GPT-35-Turbo		300 of 300 100%	
Tokens Per Minute (thousands) - GPT-35-Turbo-16K		300 of 300 100%	
Tokens Per Minute (thousands) - GPT-4		40 of 40 100%	
Tokens Per Minute (thousands) - GPT-4-32K		80 of 80 100%	
Tokens Per Minute (thousands) - Text-Embedding-Ada-002		350 of 350 100%	

Azure OpenAI Studio > Quotas

Australia East

GPT-35-Turbo: 300K TPM

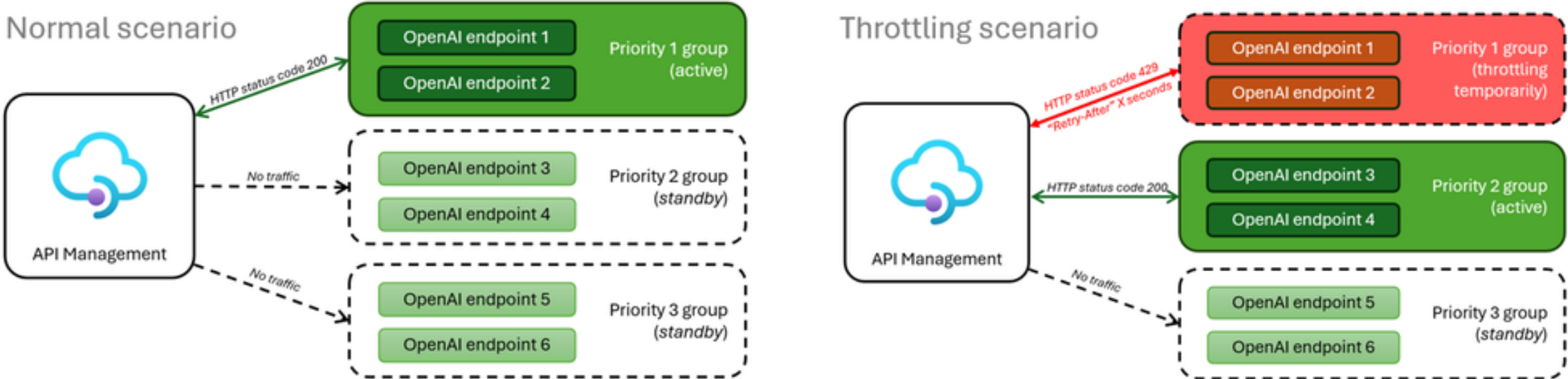
Quota name	배포	Usage/Limit	Request quota
Tokens Per Minute (thousands) - GPT-35-Turbo		300 of 300 100%	
Tokens Per Minute (thousands) - GPT-35-Turbo-16K		300 of 300 100%	
Tokens Per Minute (thousands) - GPT-4		40 of 40 100%	
Tokens Per Minute (thousands) - GPT-4-32K		80 of 80 100%	

GPT-35-Turbo: 300K TPM x 3 = 900K TPM

Smart load balancing for OpenAI endpoints

LLM은 리전별 시간대에 따라서 성능이 다를 수 있습니다. 따라서 리전별 API 호출 우선순위를 고려하세요.

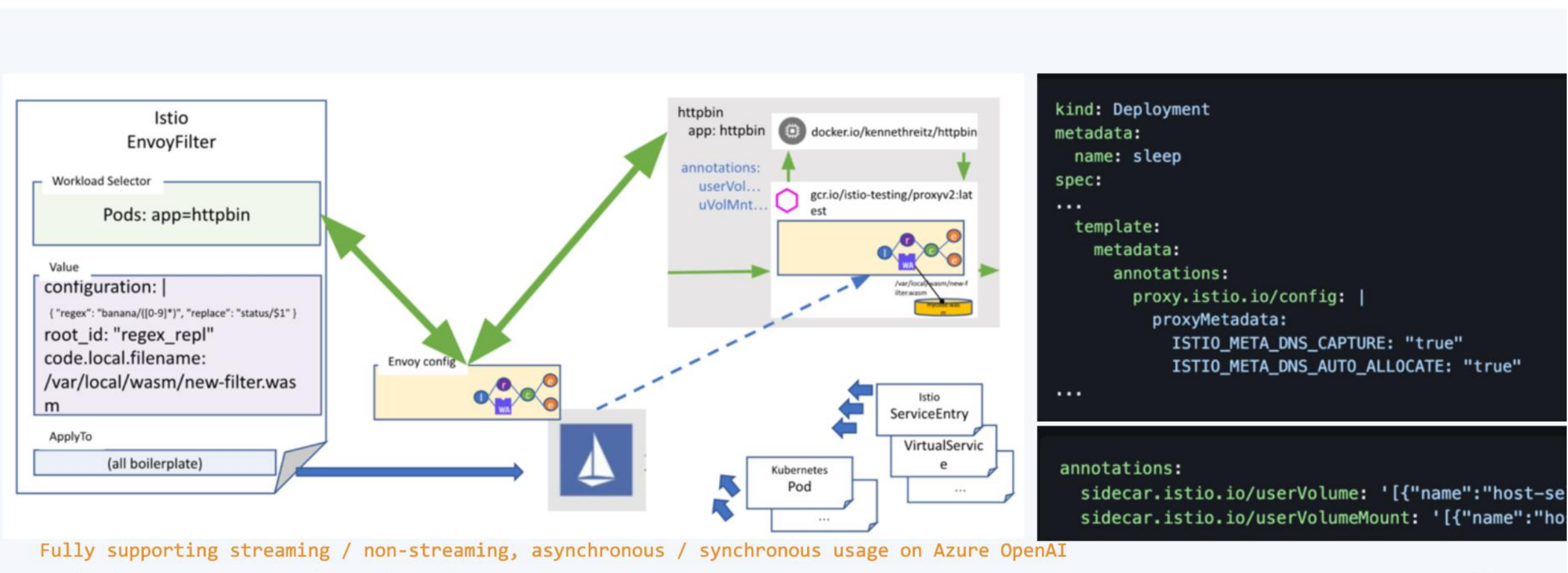
- Azure OpenAI는 토큰 제한(TPM)이 발생하면 **429 또는 TooManyRequests HTTP 상태 코드**와 다음 요청을 시도할 때까지 기다려야 하는 시간을 나타내는 Retry-After 응답 헤더를 발행합니다.
- 따라서, 이런 상황에서 **우선순위에 근거하여 다른 Endpoint를 활용하여 API를 호출**하도록 구성합니다. (API management가 이 기능을 제공하며, 일반적인 로드밸런싱 보다 더 스마트하게 구성 가능합니다.)



최적의 레이턴시 기반 라우팅 알고리즘 적용

동적으로 리전별 시간대에 따라 레이턴시가 가장 낮은 API를 호출하도록 구성

Source: <https://www.youtube.com/watch?v=MZyrCMSvKlc> | Skelter Labs Original



URL: 발표 영상 해당 부분: <https://youtu.be/qGs8DhEo7ww?t=13975>, 발표 자료에 참고한 영상: <https://www.youtube.com/watch?v=MZyrCMSvKlc>

모니터링 및 경고(알람)

HTTP Requests Tokens-Based Usage PTU Utilization Fine-tuning

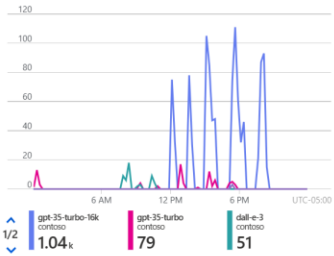


Metrics Description

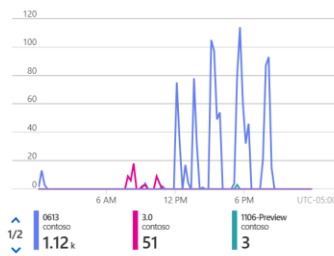
- Azure OpenAI Requests:** Number of calls made to the Azure OpenAI API over a period of time. Applies to PTU, PTU-Managed and Pay-as-you-go deployments.
- Time to Response (coming soon):** Recommended latency (responsiveness) measure for streaming requests. Applies to PTU and PTU-managed deployments. Calculated as time taken for the first response to appear after a user sends a prompt, as measured by the API gateway. This number increases as the prompt size increases and/or cache hit size reduces.

Show data for the last 15 minutes 1 hour 1 day 1 week 30 days

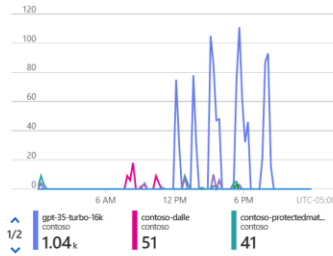
Azure OpenAI Requests by Model Name



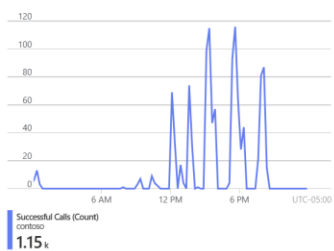
Azure OpenAI Requests by Model Version



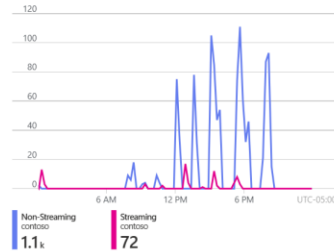
Azure OpenAI Requests by Deployment



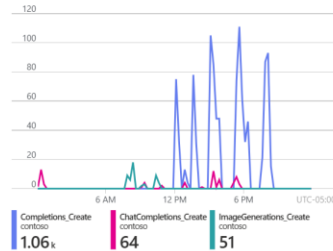
Azure OpenAI Requests by Status Code



Azure OpenAI Requests by StreamType



Azure OpenAI Requests by Operation



To examine the Azure Metrics (`AzureMetrics`) data for your resource, run the following query:

Kusto



AzureMetrics

| take 100

| project TimeGenerated, MetricName, Total, Count, Maximum, Minimum, Average, TimeGrain, UnitName

Results Chart

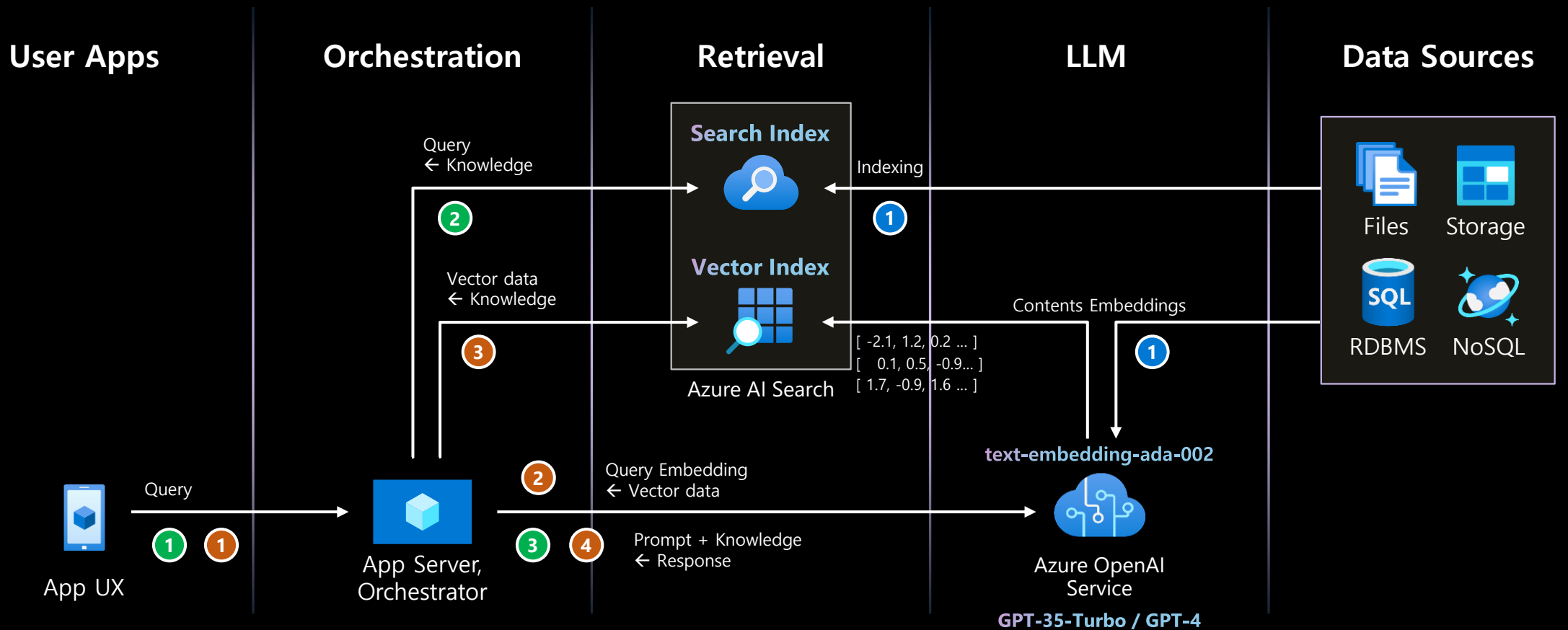
TimeGenerated [UTC] ↑↓	MetricName	Total	Count	Maximum	Minimum	Average	TimeGrain	UnitName
> 9/6/2023, 8:46:00.000 PM	Latency	105	1	105	105	105	PT1M	Milliseconds
> 9/6/2023, 8:46:00.000 PM	TotalCalls	1	1	1	1	1	PT1M	Count
▼ 9/6/2023, 8:46:00.000 ...	GeneratedTokens	150	1			150	PT1M	n/a
TimeGenerated [UTC]	2023-09-06T20:46:00Z							
MetricName	GeneratedTokens							
Total	150							
Count	1							
Average	150							
TimeGrain	PT1M							
UnitName	n/a							
> 9/6/2023, 8:46:00.000 PM	DataIn	497	1	497	497	497	PT1M	Bytes
> 9/6/2023, 8:45:00.000 PM	ProcessedPromptTokens	310	4			77.5	PT1M	n/a
> 9/6/2023, 8:45:00.000 PM	GeneratedTokens	257	4			64.25	PT1M	n/a
> 9/6/2023, 8:45:00.000 PM	DataIn	1925	4	829	321	481.25	PT1M	Bytes
> 9/6/2023, 8:45:00.000 PM	TokenTransaction	567	4			141.75	PT1M	n/a

4s 53ms | Display time (UTC+00:00) ▼

Query details | 6 - 14 of 45

외부 데이터 활용

Retrieval Augmented Generation (RAG) – 하이브리드 기반의 검색 증강 생성





2. Azure OpenAI Update

GPT-4의 성능

✂ Chatbot Arena ✂ 에 입장한 사용자는 익명의 2개 모델에 채팅을 하여 더 우수한 모델을 선택

Rank ▲	🤖 Model ▲	★ Arena Elo ▲	🇮🇹 95% CI ▲	🗳 Votes ▲	🏢 Organization ▲	📄 License ▲	📅 Knowledge Cutoff ▲
1	GPT-4-1106-preview	1251	+5/-4	48226	OpenAI	Proprietary	2023/4
1	GPT-4-0125-preview	1249	+5/-6	22282	OpenAI	Proprietary	2023/12
1	Claude 3 Opus	1247	+6/-6	14854	Anthropic	Proprietary	2023/8
4	Bard (Gemini Pro)	1202	+6/-7	12623	Google	Proprietary	Online
4	Claude 3 Sonnet	1190	+6/-6	14845	Anthropic	Proprietary	2023/8
5	GPT-4-0314	1185	+4/-6	27245	OpenAI	Proprietary	2021/9
7	GPT-4-0613	1159	+4/-5	43783	OpenAI	Proprietary	2021/9
7	Mistral-Large-2402	1155	+5/-6	18959	Mistral	Proprietary	Unknown
8	Qwen1.5-72B-Chat	1146	+4/-5	16729	Alibaba	Qianwen LICENSE	2024/2
8	Claude-1	1145	+5/-6	21929	Anthropic	Proprietary	Unknown
8	Mistral Medium	1145	+5/-4	23931	Mistral	Proprietary	Unknown
12	Claude-2.0	1126	+7/-5	13679	Anthropic	Proprietary	Unknown
12	Mistral-Next	1123	+5/-5	11875	Mistral	Proprietary	Unknown
12	Gemini Pro (Dev API)	1118	+5/-6	12146	Google	Proprietary	2023/4
13	Claude-2.1	1115	+4/-6	32431	Anthropic	Proprietary	Unknown

DALL-E 3

```
from openai import AzureOpenAI
import os
import requests
from PIL import Image
import json
from dotenv import load_dotenv
load_dotenv()

client = AzureOpenAI(
    api_version=os.getenv("DALL_E_3_API_VER"),
    azure_endpoint=os.getenv("DALL_E_3_API_BASE"),
    api_key=os.getenv("DALL_E_3_API_KEY"),
)

result = client.images.generate(
    model="dalle3", # the name of your DALL-E 3 deployment
    prompt="In a medieval monastery, depicted in a fresco with low saturation and",
    n=1
)

json_response = json.loads(result.model_dump_json())
print(json_response)

✓ 17.2s

{'created': 1701091572, 'data': [{'b64_json': None, 'revised_prompt': 'Depict a fresco'}

# Set the directory for the stored image
image_dir = os.path.join(os.getcwd(), 'images')

# If the directory doesn't exist, create it
if not os.path.isdir(image_dir):
    os.mkdir(image_dir)

# Initialize the image path (note the filetype should be png)
image_path = os.path.join(image_dir, 'generated_image.png')

# Retrieve the generated image
image_url = json_response["data"][0]["url"] # extract image URL from response
print(image_url)

✓ 0.0s

https://dalleprodsec.blob.core.windows.net/private/images/590c32d2-04b4-46fc-8437-cda
```



GPT-4 Vision

채팅 세션

채팅 지우기 | 플레이그라운드 설정 | 코드 보기 | 원시 JSON 표시

이미지가 웃긴 이유를 알려줘.



이 이미지는 세 개의 패널로 이루어져 있으며, 각각의 패널에서 휴대폰 충전 케이블과 관련된 유머러스한 상황이 표현되고 있습니다.

첫 번째 패널: 여기서 우리는 휴대폰을 보고 있습니다. 휴대폰에는 커다란 블루 커넥터가 연결된 충전 케이블이 꽂혀 있습니다. 이 커넥터의 크기는 일반적인 휴대폰 충전 케이블보다 훨씬 크고 무거워 보입니다.

두 번째 패널: 이 패널은 커넥터의 포장을 보여줍니다. 포장에는 "CABLE MANIA"라는 문구가 적혀 있고 "For Lightning Cable"이라는 설명이 있어 이것이 애플 제품의 라이트닝 케이블을 위한 것임을 나타냅니다. 하지만 포장된 커넥터는 일반적으로 컴퓨터 하드웨어에서 볼 수 있는 블루 VGA 커넥터로 보입니다.

세 번째 패널: 마지막 패널에서는 실제로 라이트닝 케이블의 끝 부분을 보여줍니다. 여기서 유머는 이 커넥터가 실제로는 라이트닝 케이블이 아니라 VGA 모니터 케이블의 일부인데, 누군가가 라이트닝 케이블로 위장하려고 시도한 것처럼 보인다는 점에 있습니다.

이미지가 웃긴 이유는 충전 케이블과 커넥터의 예상치 못한 조합과 상황의 엉뚱함 때문입니다. 사용자가 실수로 잘못된 종류의 커넥터를 구매한 것처럼 보이며, 이는 일반적으로 기술에 익숙하지 않은 사람들이 할 법한 실수를 풍자하는 데서 오는 유머입니다.

채팅 세션

채팅 지우기 | 플레이그라운드 설정 | 코드 보기 | 원시 JSON 표시

각 컴포넌트에 대해서 설명을 추가해줘.



이 이미지는 전자 부품인 LCD 모듈의 구성 요소들을 설명하고 있습니다. 모듈은 녹색 회로 기판 위에 검은색으로 된 LCD 패널이 부착되어 있으며, 각각의 컴포넌트는 흰색 박스와 숫자로 표시되어 있습니다.

- 1번은 '인터페이스 판'으로, 모듈과 외부 기기 간의 신호를 주고받는 연결점입니다.
- 2번은 '명령 레지스터'로, 모듈에 어떤 명령을 실행할지 지칭하는 부분입니다.
- 3번은 'DD RAM'으로, 화면에 표시할 데이터를 저장하는 메모리입니다.
- 4번은 'CG ROM'으로, 기본 문자와 그래픽 패턴이 저장된 읽기 전용 메모리입니다.
- 5번은 'CG RAM'으로, 사용자가 정의한 문자나 그래픽 패턴을 저장할 수 있는 메모리입니다.
- 6번은 '시프트 레지스터'로, 화면에 표시할 데이터를 순차적으로 이동시키는 역할을 합니다.
- 7번은 'LCD 패널'로, 실제로 문자나 그래픽을 표시하는 부분입니다.
- 8번은 'Back light'로, LCD 패널 뒤에서 빛을 내어 화면을 밝게 해주는 역할을 합니다.

배경에는 한글로 된 문서가 보이며, 모듈의 핀 연결 부분은 확대된 모습이 표시되어 있습니다.

여기에 사용자 쿼리를 입력하세요.(줄 바꿈 입력: Shift + Enter)

구성

배포 매개 변수

배포 *
gpt-4-vision

고급 기능

비전
Azure AI 서비스

세션 설정

지난 메시지 포함 10

현재 토큰 수

입력 토큰 진행률 표시기

592/128000

Assistants API

코드 인터프리터를 사용하면 어시스턴트 API가 샌드박스 실행 환경에서 Python 코드를 작성하고 실행할 수 있습니다.

The screenshot displays the Azure AI Studio Playground interface. On the left, a sidebar contains navigation options: Overview, Tools (with Playground selected), Evaluation, Prompt flow, Custom Neural Voice, Fine-tuning, Components (with Data, Indexes, Deployments, and Content filters), and Settings. The main area is titled 'Build / contoso-chatbot / Assistant'. It shows the assistant's configuration, including its name 'MyAssistant' and instructions: 'You are a helpful financial analyst assistant for the company Contoso. You are tailored to help users track stock prices, analyze their financial documents, and answer questions about Contoso's financial data.' The deployment is set to 'gpt-4'. A list of functions includes 'get_stock_price'. The code interpreter is enabled, and a file named '2023_sales_data.csv' is loaded. The chat history shows a user asking for Microsoft's stock price, followed by the assistant's response and a 'get_stock_price' action. The user then asks for an analysis of sales data, leading to a 'code_interpreter' action and a bar chart titled 'Top Selling Items by Total Revenue'. The chart shows the following data:

Product Name	Total Revenue
MegaGadget	~85
PortableWhatsit	~75
FancyWidget	~65
ShinyDoohickey	~55
CoolGadget	~45
SmartThingy	~40

The right sidebar shows the 'Logs' section with a list of API requests and responses, including a POST request for creating a thread and a GET request for retrieving a thread.



3. Your SLMs and GPU Power

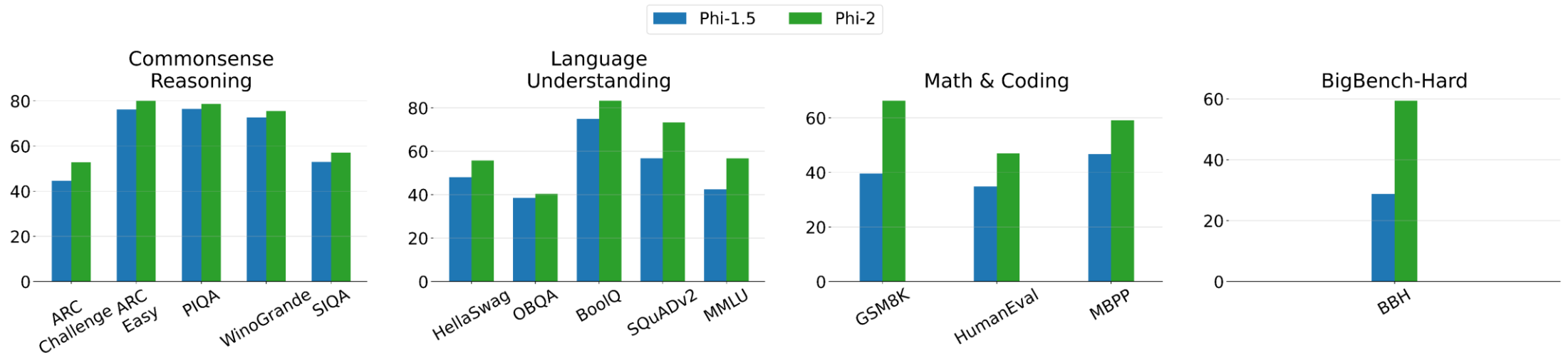
SLM으로 Phi-2 MIT 라이선스 제공

Microsoft는 안전하고 성능이 우수한 Small Language Models을 연구하고 제공

Phi-2(2.7B) 모델: MICROSOFT RESEARCH 라이선스에서 **MIT 라이선스로 변경**

1.4T 토큰 기반이며, 96 A100 GPUs 기반 14일간 훈련된 모델

QA 형식, 채팅 형식 및 코드 형식을 사용하는 프롬프트에 가장 적합

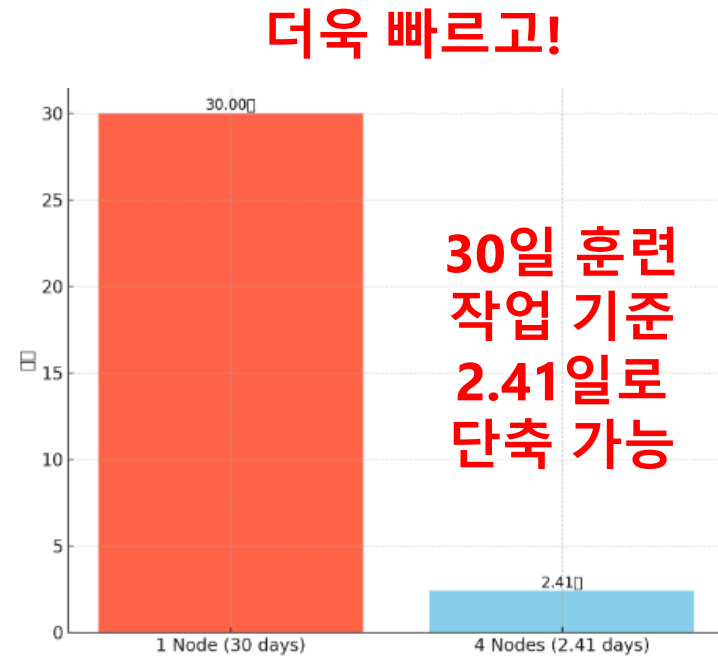
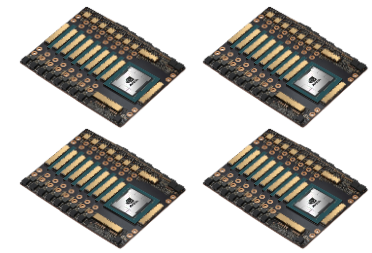
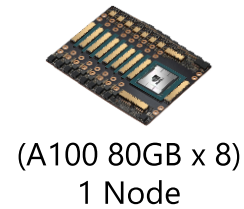


Nvidia A100 80GB x 8 = 1 Node vs 4 Nodes

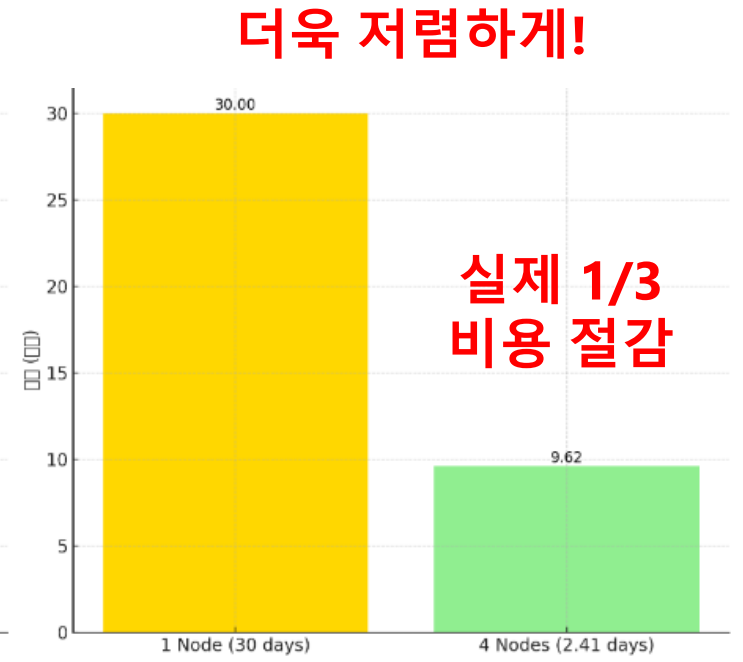
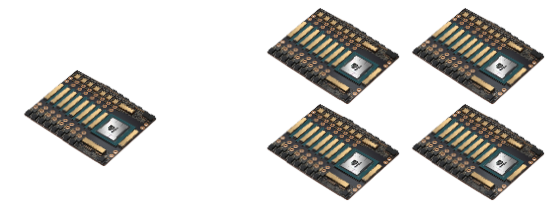
Meta Llama-2-70B-HF 모델 기반 훈련 벤치마킹 정보



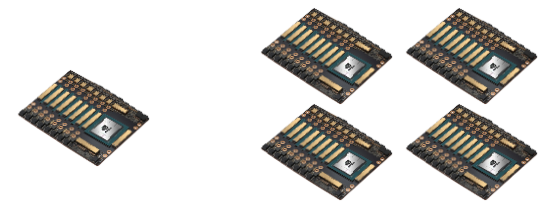
성능 차이
12.47배



30일 훈련
작업 기준
2.41일로
단축 가능



실제 1/3
비용 절감



VM 내의 각 GPU에는 고유한 전용 토폴로지 독립적
200GB/초 NVIDIA Mellanox HDR InfiniBand 연결이 제공

Assistants API (2024 Q1)

아래 제시한 문장은 Llama-2-70B-hf 모델을 학습할 때 Azure의 Azure A100 80GB GPU가 탑재된 노드 1대와 4대 썼을 때의 성능 차이를 문서화 해 놓은 데이터입니다.

국내 A사는 sLLM을 만들기 위해 Llama-2-70B-hf 모델 학습을 위해서 Azure A100 80GB GPU가 8개 탑재된 4대의 노드를 실험한 결과 싱글노드 대비 12.47배 학습 시간에 대한 성능 향상을 경험했습니다. 1대의 노드로 1달 걸릴 작업을 4대 노드로 할 경우 3일이면 끝낼 수 있음을 의미합니다. 비용은 30일이 아닌 3일*4대인 12일치에 대한 비용입니다. 시간과 비용 모두를 얻을 수 있습니다. 이는 OpenAI, Nvidia, Azure 3사의 노력이며 GPU 뿐만 아니라 InfiniBand를 통한 성능 향상이 극명합니다. A6000은 확장 및 전력 소모 측면에서 상대적으로 불리합니다. (이하 생략)

상세: 국내 A사 케이스

1. Llama-2-70B-hf 모델 학습 시간 비교시, Azure A100(80G*8) Single 대비 4-Multiple (InfiniBand) 활용시 약 12.47배 학습 성능 향상됨, 비용 대비 성능 고려시 노드 효율이 3.11배 증가함. Single은 1 Training Step 당 168.59s 걸리던 것이 4 nodes 기준 13.52s 걸림.

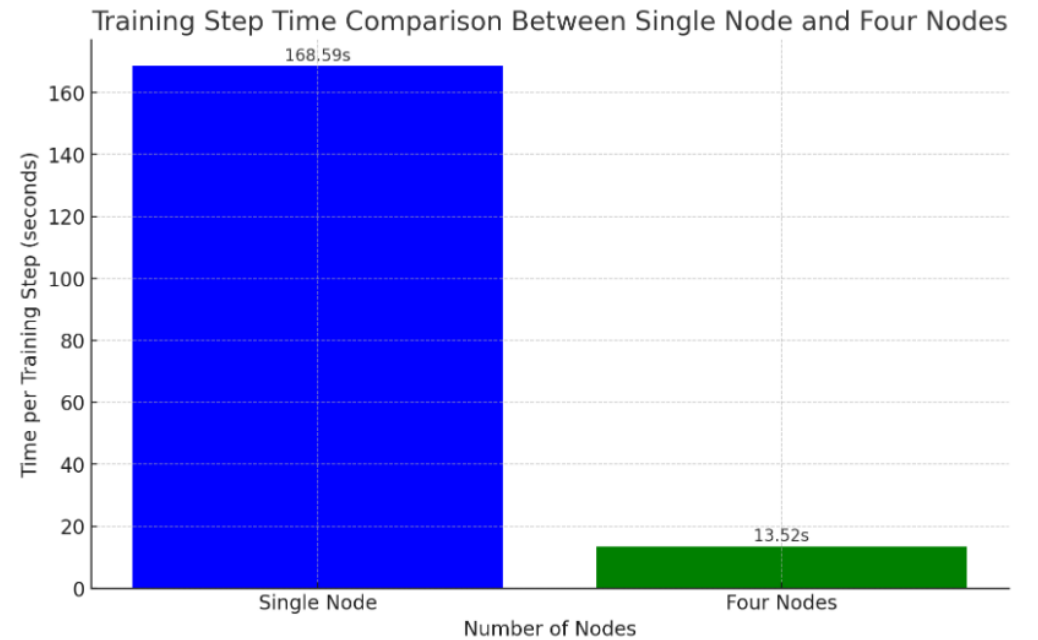
2. Llama-2 13B 모델 학습 시간 비교시, 0.399s vs 0.130s 나옴. 성능은 3.06배, 노드 효율에 대한 가성비는 0.765배

3. A6000(GA102-8nm)은 A100(GA100-7nm)처럼 Nvidia의 Ampere 아키텍처를 따르고 있으나, NDm A100은 80GB의 메모리로 머신러닝에 특화되어 있고, 200Gb/s 지원하는 InfiniBand에 의해 더 높은 성능을 제공함.

위 데이터를 바탕으로 1개의 노드와 4개의 노드를 사용했을 때의 1 step 학습하는데 필요한 성능의 차이를 쉽게 인지할 수 있도록 바차트를 그려줘.



Here is the bar chart that visualizes the time required for 1 training step when using 1 node versus 4 nodes. As shown, using 4 nodes significantly reduces the time per training step from 168.59 seconds to 13.52 seconds, which clearly illustrates the performance difference.



Assistants API (2024 Q1)

2024-02
(Public Preview)

Azure AI | Azure OpenAI Studio

Kim Hyounsoo (KH)

Assistants playground (Preview)

Show panels

Assistant setup

Assistant

+ New Save Open Delete

Assistant ID: asst_HrAUGtVgaLeq3tWolu8Au05m

Assistant name

Instructions

Deployment: gpt-4

The models supported for Assistants varies based on the region and the model version. [Learn more](#)

Assistant tools: + Add function

Code interpreter:

Files: + Add files

Upload files to use with the Assistant tools. You can upload up to 20 files, with a max size of 512 MB. [Learn more](#)

Chat session

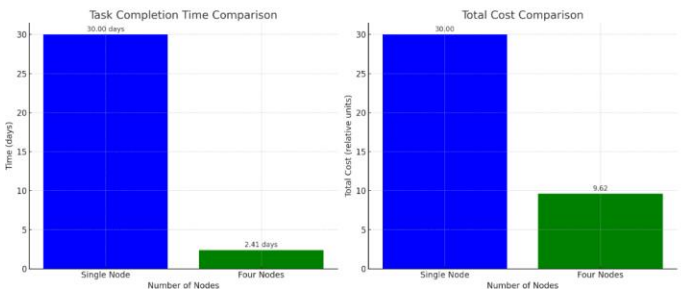
Clear chat

위 실험 결과를 보았을 때, 1대를 사용할 때 시간이 1달 걸리는 작업이 있을 때, 4대를 활용하면 얼마나 시간을 단축할 수 있을까. 동일한 작업을 완성하기 까지 사용되어지는 비용은 어떤 차이가 발생할지 차트로 그려서 비교해줘.

The bar charts illustrate the comparison between the time required and the total cost to complete a task with a single node versus using four nodes.

- For the task completion time, it's clear that using four nodes greatly reduces the duration from 30 days to approximately 2.40 days, based on the 12.47 times speedup mentioned earlier.
- In terms of total cost, using a single node would be equivalent to 30 units of cost (since it takes 30 days with a cost of 1 unit per day). However, with four nodes, due to the reduced time of around 2.40 days, even though we are using 4 times the resources per day, the total cost is approximately 9.60 units. This is due to the significantly reduced duration of the task, demonstrating that the higher daily cost is offset by the much shorter time requirement.

In summary, the bar charts demonstrate that while using four nodes can be a higher cost per day, the overall cost to complete the same task is much less due to the drastic reduction in time required.



Number of Nodes	Time (days)
Single Node	30.00
Four Nodes	2.41

Number of Nodes	Total Cost (units)
Single Node	30.00
Four Nodes	9.60

Type user query here. (Shift + Enter for new line)

Messages in the Assistants playground are visible to anyone with access to this resource and using the API.

Logs

Clear logs

Add message >

POST /openai/threads/thread_J9nqwH10t5t5AhDVM83H9nfq/messages

Run thread >

POST /openai/threads/thread_J9nqwH10t5t5AhDVM83H9nfq/runs

Get run steps >


GET /openai/threads/thread_J9nqwH10t5t5AhDVM83H9nfq/runs/run_ShBNiiaq8pFY63ImZ6Hqgf5/steps

4. LLMOps – Prompt Flow

시작: <https://microsoft.github.io/promptflow/how-to-guides/quick-start.html>, GitHub Repo: <https://github.com/microsoft/promptflow>

LLM App 제작을 더 쉽고 빠르게!

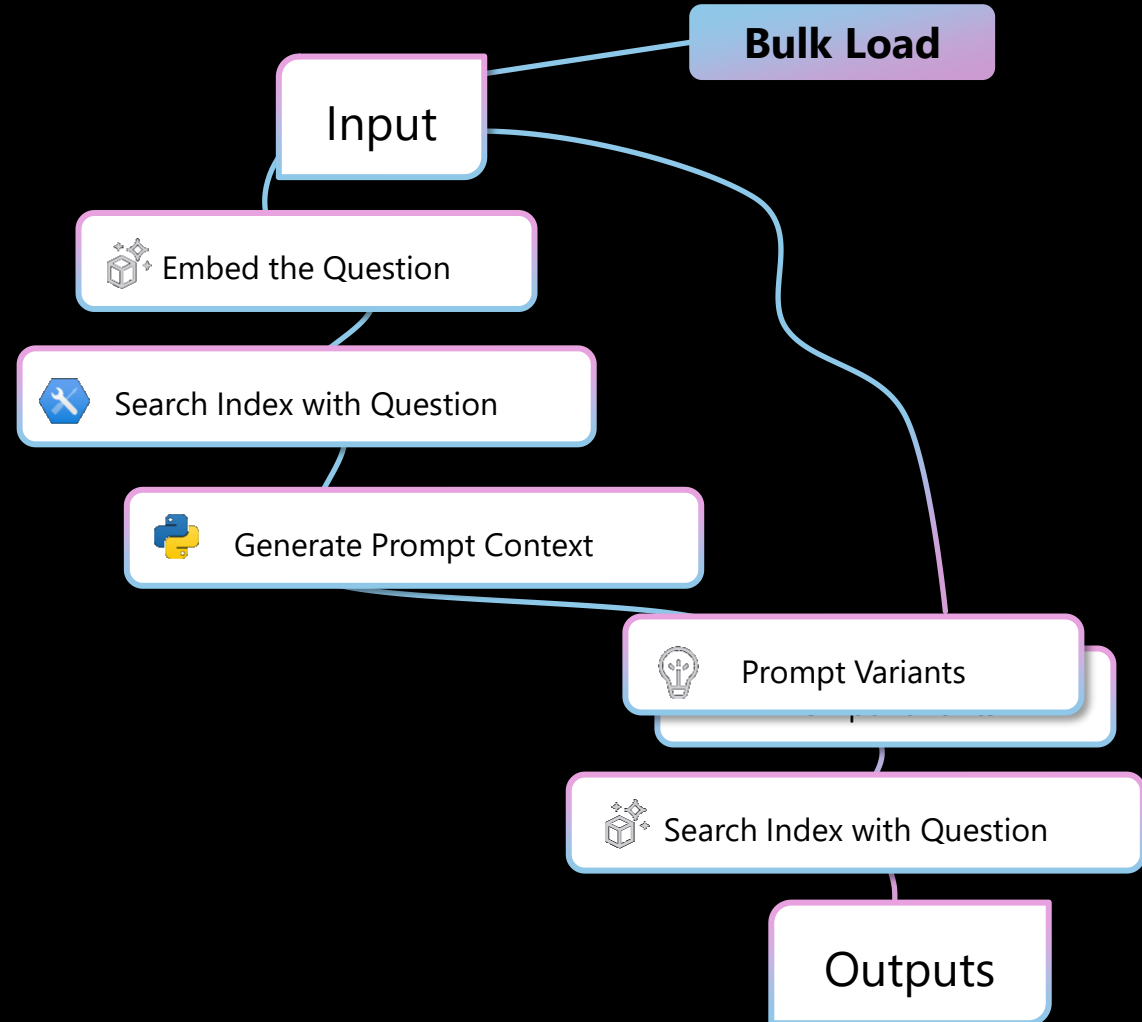
Prompt Flow – LLM 기반 응용 프로그램을 위한 전체 개발 주기를 단순화하는 도구

 Orchestrates AI models, prompts, and APIs

 Support for prompt tuning and experimentation

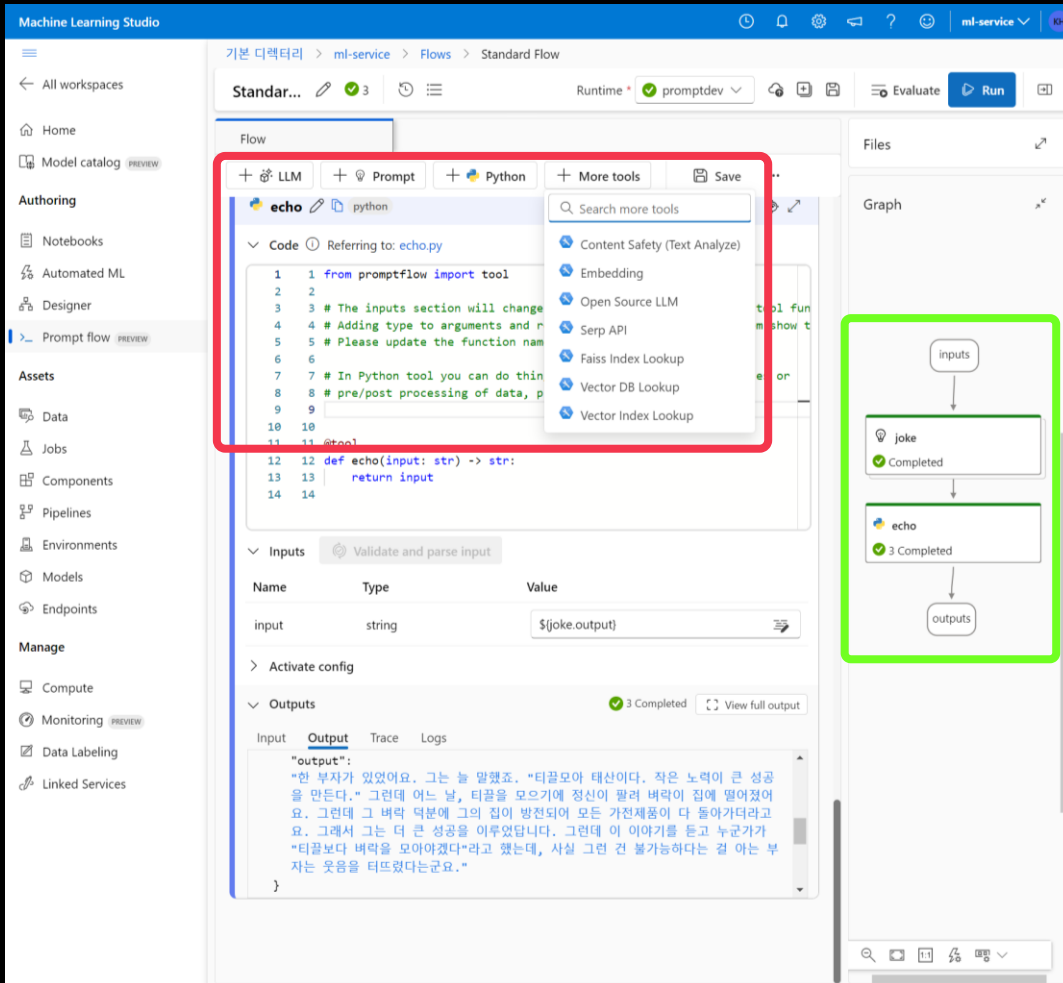
 Blue/green deployments and testing

 Supports Semantic Kernel, LangChain, Plugins



각 기능들을 연결하여 해결

Prompt Flow – LLM 기반 응용 프로그램을 위한 전체 개발 주기를 단순화하는 도구



워크플로우 개발

- 다양한 LLM 모델, 외부 데이터와 연결하고, 코드를 작성 시각화된 맞춤형 워크플로우 설계

테스트 및 평가

- 성능, 근거, 정확성과 같은 측정 항목을 사용하여 워크플로의 AI 품질을 평가
- 대규모 데이터 세트를 병렬로 테스트

프롬프트 튜닝

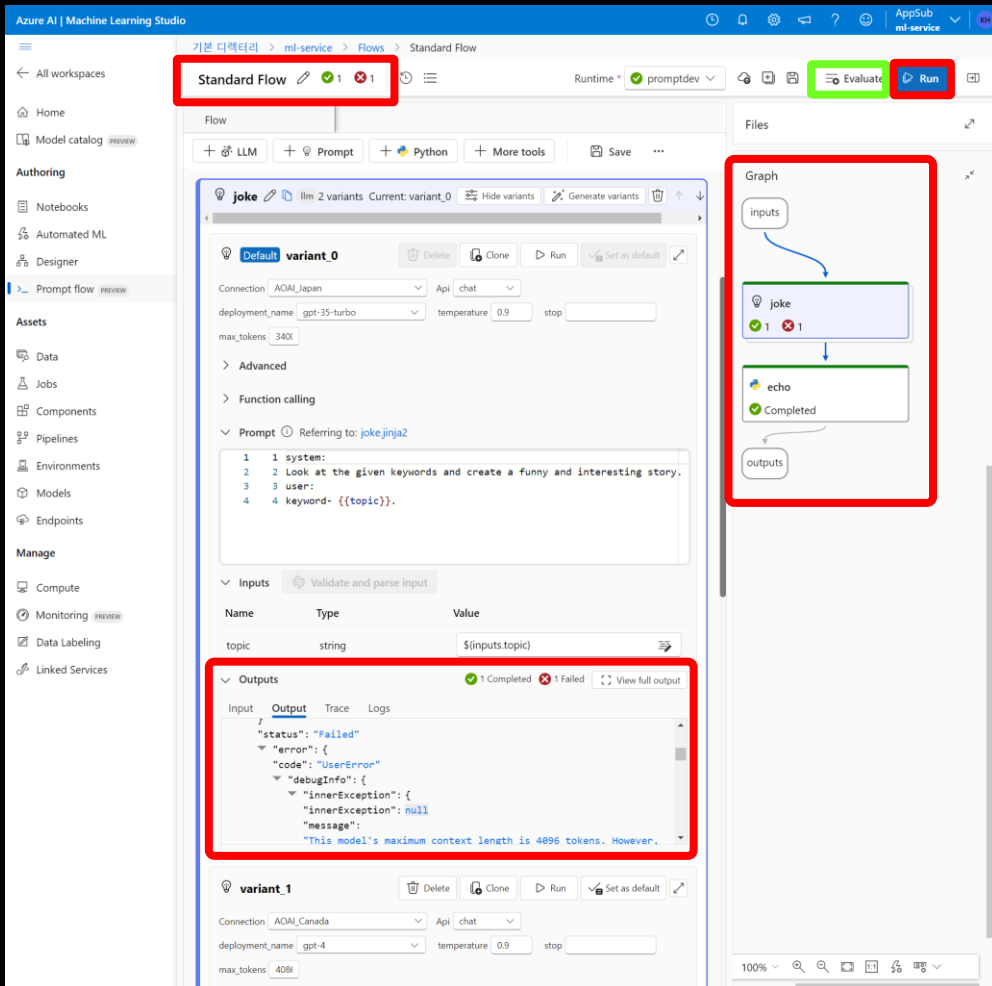
- 변형 및 버저닝으로 간단하게 프롬프트를 조정

비교 및 배포

- 여러 실험을 시각적으로 비교
- 신속한 통합을 위해 관리형 엔드포인트에 원 클릭 배포

각 기능들을 연결하여 해결

Prompt Flow – LLM 기반 응용 프로그램을 위한 전체 개발 주기를 단순화하는 도구



워크플로우 개발

- 다양한 LLM 모델, 외부 데이터와 연결하고, 코드를 작성하여 시각화된 맞춤형 워크플로우 설계

테스트 및 평가

- 성능, 근거, 정확성과 같은 측정 항목을 사용하여 워크플로의 AI 품질을 평가
- 대규모 데이터 세트를 병렬로 테스트

프롬프트 튜닝

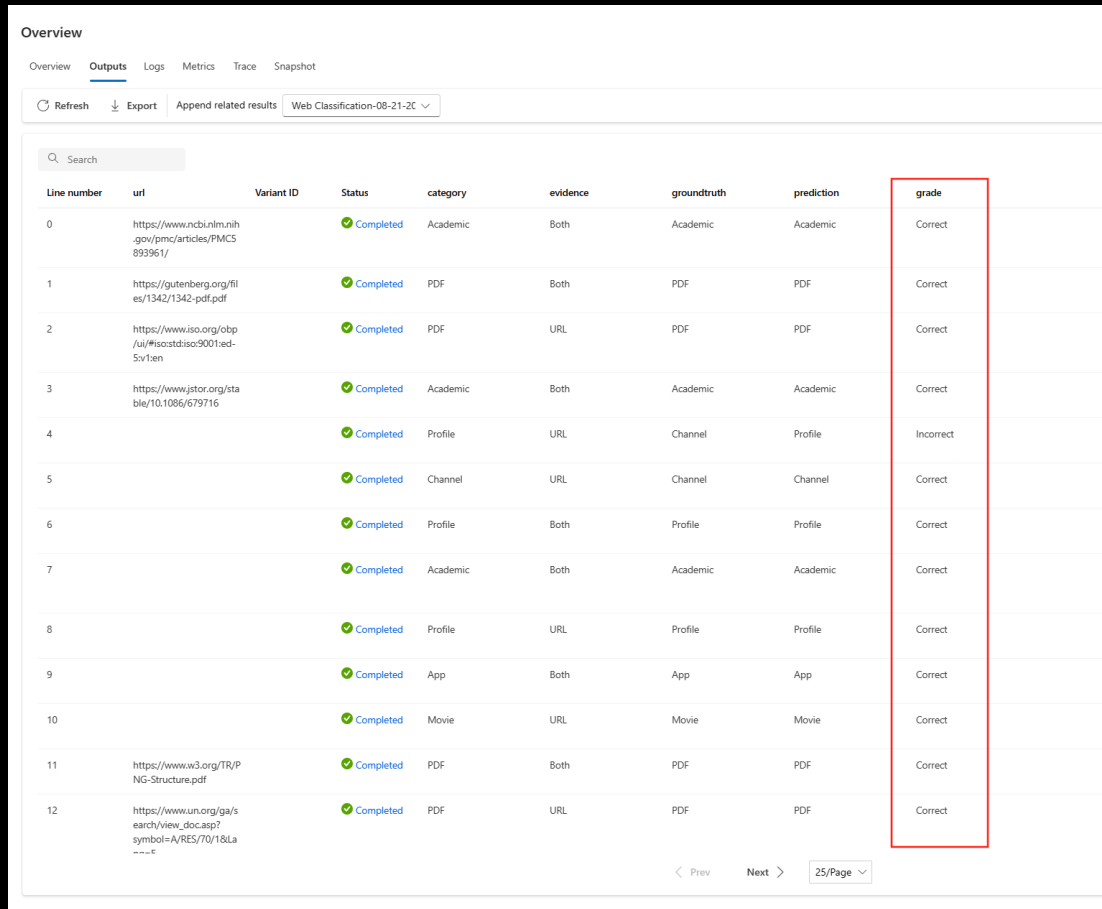
- 변형 및 버저닝으로 간단하게 프롬프트를 조정

비교 및 배포

- 여러 실험을 시각적으로 비교
- 신속한 통합을 위해 관리형 엔드포인트에 원 클릭 배포

각 기능들을 연결하여 해결

Prompt Flow – LLM 기반 응용 프로그램을 위한 전체 개발 주기를 단순화하는 도구



Overview

Overview **Outputs** Logs Metrics Trace Snapshot

Refresh Export Append related results Web Classification-08-21-20

Search

Line number	url	Variant ID	Status	category	evidence	groundtruth	prediction	grade
0	https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5893961/		Completed	Academic	Both	Academic	Academic	Correct
1	https://www.gutenberg.org/files/1342/1342-pdf.pdf		Completed	PDF	Both	PDF	PDF	Correct
2	https://www.iso.org/obp/ui/#iso:std:iso:9001:ed-5:vi:en		Completed	PDF	URL	PDF	PDF	Correct
3	https://www.jstor.org/stable/10.1086/679716		Completed	Academic	Both	Academic	Academic	Correct
4			Completed	Profile	URL	Channel	Profile	Incorrect
5			Completed	Channel	URL	Channel	Channel	Correct
6			Completed	Profile	Both	Profile	Profile	Correct
7			Completed	Academic	Both	Academic	Academic	Correct
8			Completed	Profile	URL	Profile	Profile	Correct
9			Completed	App	Both	App	App	Correct
10			Completed	Movie	URL	Movie	Movie	Correct
11	https://www.w3.org/TR/PNG-Structure.pdf		Completed	PDF	Both	PDF	PDF	Correct
12	https://www.un.org/ga/search/view_doc.asp?symbol=A/RES/70/1&La...		Completed	PDF	URL	PDF	PDF	Correct

< Prev Next > 25/Page

워크플로우 개발

- 다양한 LLM 모델, 외부 데이터와 연결하고, 코드를 작성하여 시각화된 맞춤형 워크플로우 설계

테스트 및 평가

- 성능, 근거, 정확성과 같은 측정 항목을 사용하여 워크플로의 AI 품질을 평가
- 대규모 데이터 세트를 병렬로 테스트

프롬프트 튜닝

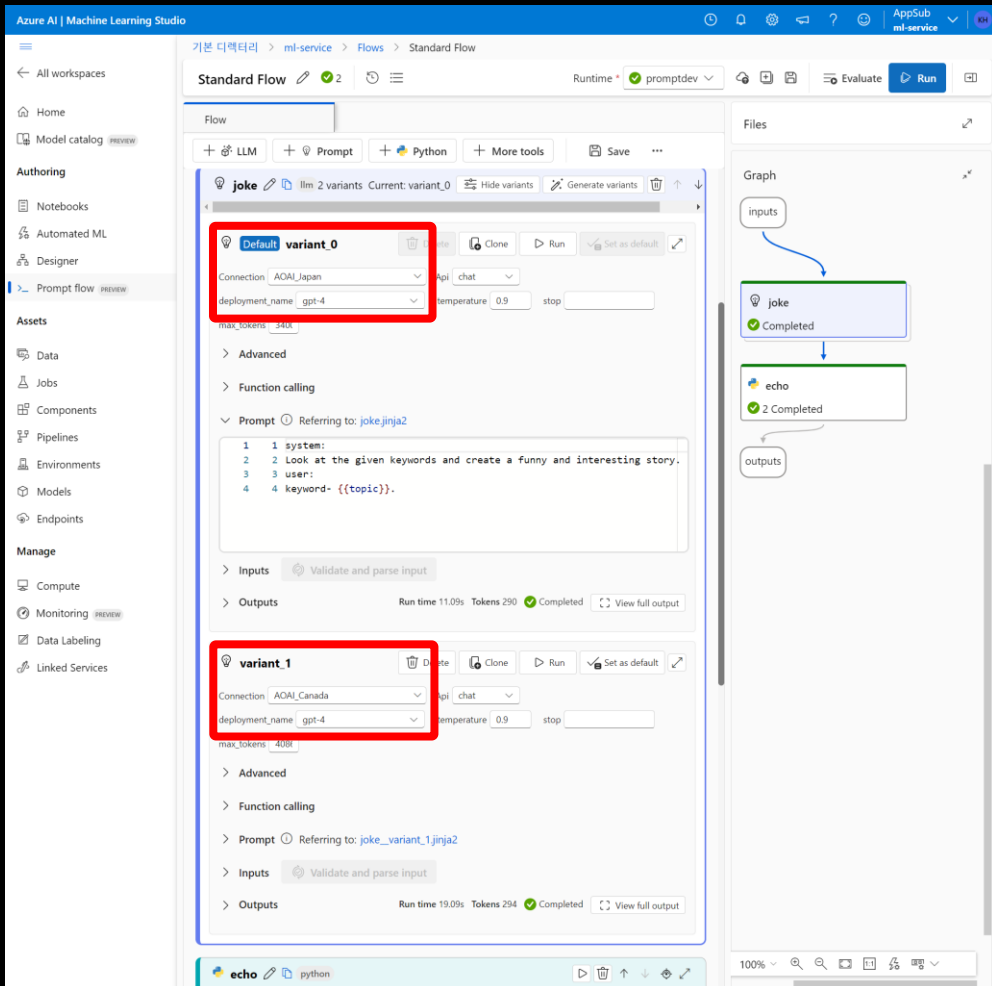
- 변형 및 버저닝으로 간단하게 프롬프트를 조정

비교 및 배포

- 여러 실험을 시각적으로 비교
- 신속한 통합을 위해 관리형 엔드포인트에 원 클릭 배포

각 기능들을 연결하여 해결

Prompt Flow – LLM 기반 응용 프로그램을 위한 전체 개발 주기를 단순화하는 도구



워크플로우 개발

- 다양한 LLM 모델, 외부 데이터와 연결하고, 코드를 작성하여 시각화된 맞춤형 워크플로우 설계

테스트 및 평가

- 성능, 근거, 정확성과 같은 측정 항목을 사용하여 워크플로의 AI 품질을 평가
- 대규모 데이터 세트를 병렬로 테스트

프롬프트 튜닝

- 변형 및 버저닝으로 간단하게 프롬프트를 조정

비교 및 배포

- 여러 실험을 시각적으로 비교
- 신속한 통합을 위해 관리형 엔드포인트에 원 클릭 배포

각 기능들을 연결하여 해결

Prompt Flow – LLM 기반 응용 프로그램을 위한 전체 개발 주기를 단순화하는 도구

The screenshot displays the 'Outputs' window in Azure AI Machine Learning Studio. A table with columns 'name', 'duration', 'start time (UTC)', and 'end time (UTC)' is highlighted with a red border. Below the table, a JSON configuration for an LLM call is visible, including fields like 'api_base', 'api_type', 'api_version', 'engine', 'max_tokens', and 'messages'.

name	duration	start time (UTC)	end time (UTC)
▼ AzureOpenAI.chat	11,084 ms	2023-10-25 11:43:0:590	2023-10-25 11:43:11:674
openai.api_resources.chat_completion.ChatCor	11,081 ms	2023-10-25 11:43:0:593	2023-10-25 11:43:11:674
echo	0 ms	2023-10-25 11:43:11:694	2023-10-25 11:43:11:694
▼ AzureOpenAI.chat	19,090 ms	2023-10-25 11:43:12:486	2023-10-25 11:43:31:576
openai.api_resources.chat_completion.ChatCor	19,087 ms	2023-10-25 11:43:12:489	2023-10-25 11:43:31:576
echo	0 ms	2023-10-25 11:43:31:600	2023-10-25 11:43:31:601

```
{
  "inputs": {
    "api_base": "https://japanopenai.azure.com/",
    "api_type": "azure",
    "api_version": "2023-07-01-preview",
    "engine": "gpt-4",
    "frequency_penalty": 0,
    "logit_bias": {}
  },
  "max_tokens": 3400,
  "messages": [
    {
      "content": "Look at the given keywords and create a funny and interesting story. Limit the length of your story to a maximum of 100 words. Output language is Korean.",
      "role": "system"
    },
    {
      "content": "keyword- 티끌, 비락, 부자, 성공.",
      "role": "user"
    }
  ],
  "n": 1,
  "presence_penalty": 0,
  "stop": null,
  "stream": true,
  "temperature": 0.9,
  "top_p": 1,
  "user": ""
}
```

워크플로우 개발

- 다양한 LLM 모델, 외부 데이터와 연결하고, 코드를 작성하여 시각화된 맞춤형 워크플로우 설계

테스트 및 평가

- 성능, 근거, 정확성과 같은 측정 항목을 사용하여 워크플로의 AI 품질을 평가
- 대규모 데이터 세트를 병렬로 테스트

프롬프트 튜닝

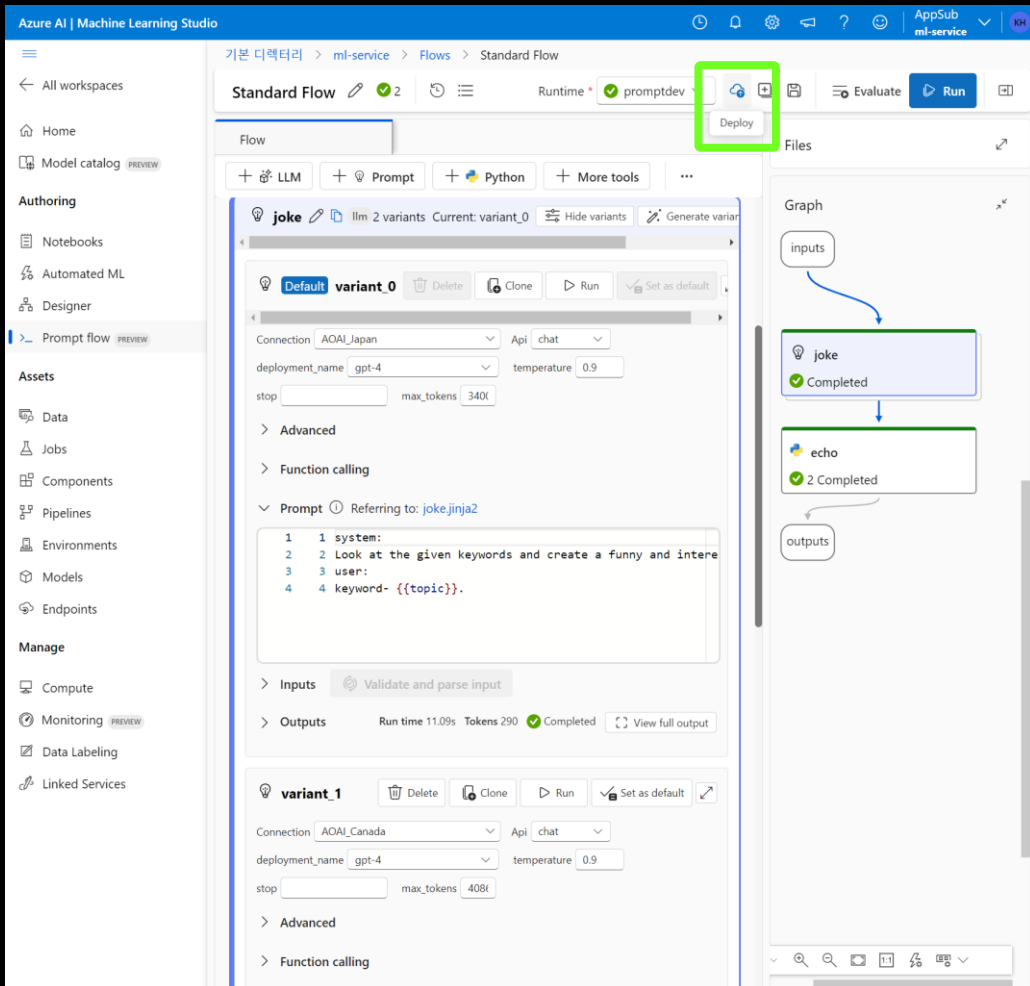
- 변형 및 버저닝으로 간단하게 프롬프트를 조정

비교 및 배포

- 여러 실험을 시각적으로 비교
- 신속한 통합을 위해 관리형 엔드포인트에 원 클릭 배포

각 기능들을 연결하여 해결

Prompt Flow – LLM 기반 응용 프로그램을 위한 전체 개발 주기를 단순화하는 도구



워크플로우 개발

- 다양한 LLM 모델, 외부 데이터와 연결하고, 코드를 작성하여 시각화된 맞춤형 워크플로우 설계

테스트 및 평가

- 성능, 근거, 정확성과 같은 측정 항목을 사용하여 워크플로의 AI 품질을 평가
- 대규모 데이터 세트를 병렬로 테스트

프롬프트 튜닝

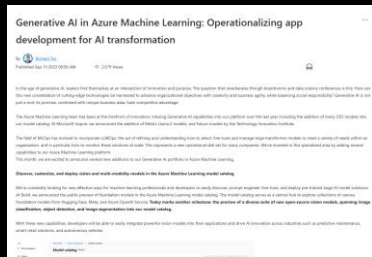
- 변형 및 버저닝으로 간단하게 프롬프트를 조정

비교 및 배포

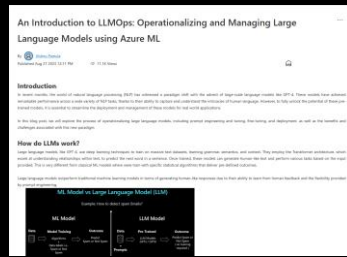
- 여러 실험을 시각적으로 비교
- 신속한 통합을 위해 관리형 엔드포인트에 **원 클릭 배포**

LLMOps 배우기

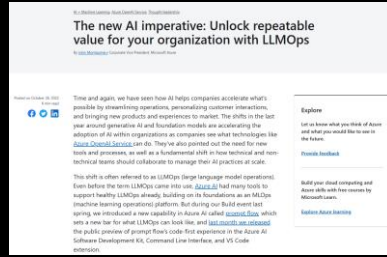
Blogs



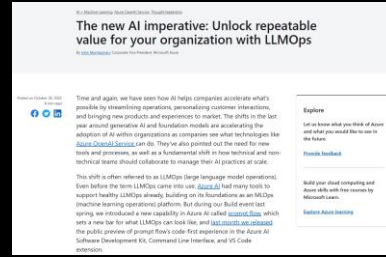
[TC Blog - GenAI in AzureML](#)



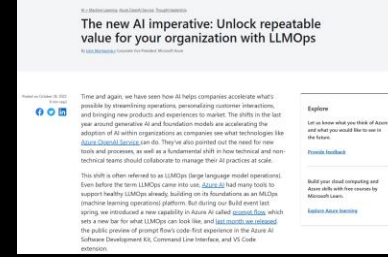
[TC Blog - Intro to LLMOps](#)



[Azure Blog - LLMOps Series 1](#)

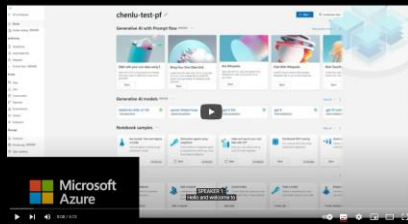


[Azure Blog - LLMOps Series 2](#)



[Azure Blog - LLMOps Series 3](#)

Demos/ Videos



[LLMOps with PF demo video](#)



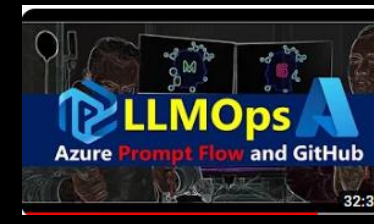
[PF snackable video](#)



[Getting started with PF](#)



[Model Monitoring for LLMs](#)

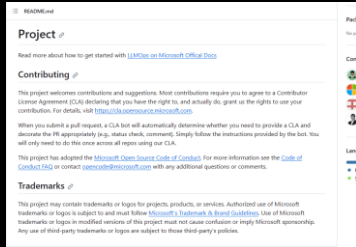


[LLMOps with prompt flow & GH](#)

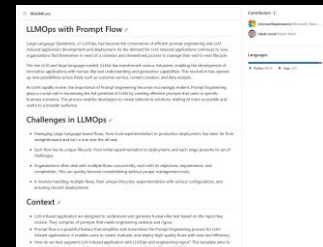
Technical Docs/ SA



[End-to-end LLMOps with PF+GH](#)



[LLMOps Solution Accelerator](#)



[LLMOps with prompt flow template](#)



5. Azure OpenAI 모델 정보

모델별 리전 지원 및 특성

Model ID	Base model Regions	Deprecation Date	Default Limit Value (TPM)	Max Request (tokens)	Training Data (up to)	Pricing (per 1,000 tokens)	
text-embedding-ada-002 (v2)	East US, South Central US, West Europe, France Central	2025-04-03	240K	8,191	Sep 2021	\$0.0001	
	Australia East, Canada East, East US2, Japan East, North Central US, Norway East, Sweden Central, Switzerland North, UK South, West US, South India, Brazil South	2025-04-03	350K				
text-embedding-3 (S/L)	Canada East, East US, East US 2	2025-02-02	350K	8,191		Large: \$0.00013	Small: \$0.00002
gpt-35-turbo-instruct (0914)	East US, Sweden Central	2024-09-30	240K	4,097		\$0.0015	\$0.0020
gpt-35-turbo (0301)	East US, France Central, South Central US, UK South, West Europe	2024-07-05	240K	4,096		\$0.0015	\$0.0020
gpt-35-turbo (0613)	East US, South Central US, West Europe, France Central, UK South North Central US, Australia East, East US 2, Canada East, Japan East, Sweden Central, Switzerland North	2024-06-13 2024-06-13	240K 300K			\$0.0015	\$0.0020
gpt-35-turbo-16K (0613)	East US, South Central US, West Europe, France Central, UK South	2024-06-13	240K	16,384		\$0.0030	\$0.0040
	North Central US, Australia East, East US 2, Canada East, Japan East, Sweden Central, Switzerland North	2024-06-13	300K				
gpt-35-turbo (1106)	Australia East, Canada East, France Central, South India, Sweden Central, UK South, West US	2024-11-06	120K	16,385 / 4,096		\$0.0010	\$0.0020
gpt-35-turbo (0125)	East US, Canada East, North Central US, South Central US	2025-02-15	120K	16,385 / 4,096		\$0.0005	\$0.0015
gpt-4 (0613)	East US, South Central US, West Europe, France Central	2024-07-05	20K	8,192		\$0.0300	\$0.0600
	North Central US, Australia East, East US 2, Canada East, Japan East, UK South, Sweden Central, Switzerland North	2024-07-05	40K				
gpt-4-32k (0613)	East US, South Central US, West Europe, France Central	2024-07-05	60K	32,768	\$0.0600	\$0.1200	
	North Central US, Australia East, East US 2, Canada East, Japan East, UK South, Sweden Central, Switzerland North	2024-07-05	80K				
gpt-4 (1106-preview)	Australia East, Canada East, East US 2, France Central, UK South, West US	2024-03-31	80K	128,000 / 4,096	Apr 2023	\$0.0100	\$0.0300
	South India, Norway East, Sweden Central	2024-03-31	150K				
gpt-4 (1106-vision-preview)	Sweden Central, West US, Japan East, Switzerland North, Australia East	2024-03-31	30K	128,000 / 4,096	Apr 2023	\$0.0100	\$0.0300
gpt-4 (0125-preview)	East US, North Central US, South Central US	2024-04-15	80K	128,000 / 4,096	Dec 2023	\$0.0100	\$0.0300
dall-e-3	East US, Sweden Central, Australia East	2024-07-15	12 req / 1m	4,000 (Max Char)	-	\$4~8(SD) / \$8~12(HD)	
whisper	East US 2, North Central US, West Europe, South India, Norway East, Sweden Central	2026-02-01	-	25 MB (Audio)	-	\$0.36 (per hour)	
tts, tts-hd (preview)	North Central US, Sweden Central	2025-03-01	-	-	-	\$15/\$30(HD) 1M characters	

API 명세

- 2022-12-01 [Swagger spec](#)
- 2023-03-15-preview (retiring April 2, 2024) [Swagger spec](#)
- 2023-05-15 [Swagger spec](#)
- 2023-06-01-preview (retiring April 2, 2024) [Swagger spec](#)
- 2023-07-01-preview (retiring April 2, 2024) [Swagger spec](#)
- 2023-08-01-preview (retiring April 2, 2024) [Swagger spec](#)
- 2023-09-01-preview (retiring April 2, 2024) [Swagger spec](#)
- 2023-12-01-preview (retiring April 2, 2024) [Swagger spec](#)
- 2024-02-15-preview [Swagger spec](#)
- 2024-03-01-preview [Swagger spec](#)

참고할 수 있는 자료 모음

Azure OpenAI GPT 활용 Prompthon 자료

1. [소개 자료](#), [AOAI 소개](#), [프롬프트 엔지니어링](#), [실습 소개](#)
2. [실습 자료: https://github.com/HyounsooKim/azure-openai-samples-kr/](https://github.com/HyounsooKim/azure-openai-samples-kr/)

Azure OpenAI 기초

1. [2023 한국 고객 Azure OpenAI 사용 사례 모음](#)
2. [Azure OpenAI Service News](#), [AOAI 리전별 지원 모델 정보](#), [AOAI Quotas 정보](#)
3. [Introduction to prompt engineering - Azure OpenAI](#)
4. [Prompt engineering techniques - Azure OpenAI](#)
5. [Azure OpenAI Samples](#), [OpenAI cookbook](#), [Prompt Examples](#)

LLMOps

1. [Prompt flow Quick Start](#)
2. [Prompt flow in Azure AI Studio](#)

운영의 묘

1. [Azure OpenAI Service embeddings tutorial - Azure OpenAI](#)
2. [부하 분사 및 Quotas 확보: Smart load balancing for Azure OpenAI endpoints](#)
3. [한글 프롬프트 오동작 방지를 위한 Contents Filtering 사용 승인](#)



Thank you