



MLOps Now

LLM in Production



쉽고 빠르게 커스텀 LLM을 파인튜닝, 배포하는 방법

안재만, Co-founder & CEO, VESSL AI

How to Build & Serve Private LLMs

쉽고 빠르게 커스텀 LLM을 파인튜닝 및 배포하는 방법

Jaeman An | VESSL AI | 2024-05-07

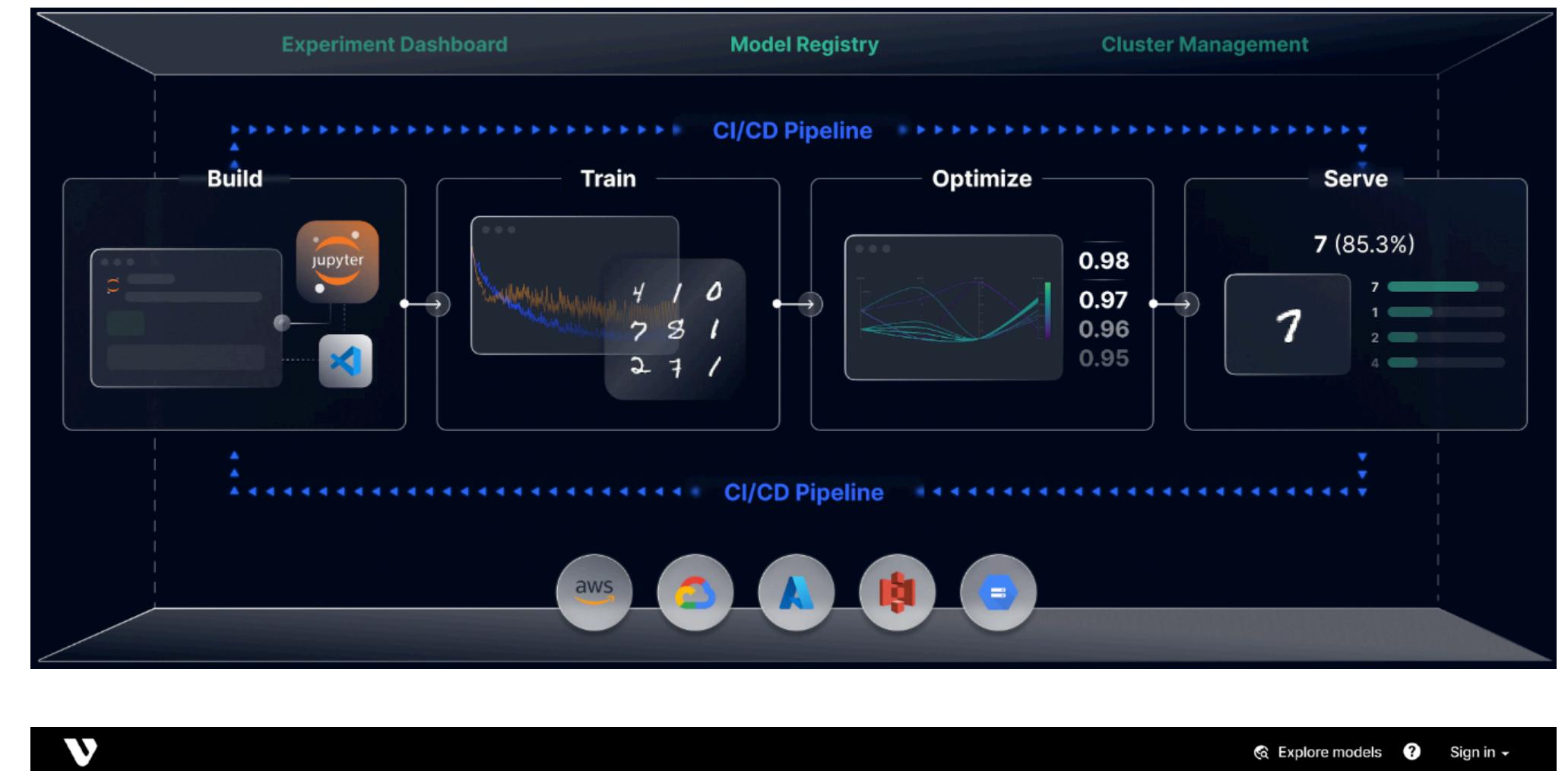
Intro

안녕하세요! 저는 안재만입니다. 🙌



- **VESSL AI (<https://vessl.ai/>)**

- MLOps for high-performance ML Teams
- 코딩, 모델 학습, 최적화, 배포와 자동화 파이프라인까지
end-to-end AI 서비스를 만들 수 있는 플랫폼



- **DevOps engineer building data platform**

- Design, develop & operate [MLOps platform](#) @ VESSL AI
- Lifesaving AI @ AITRICS
- Million-scale Game Lead @ DEVSISTERS Inc.

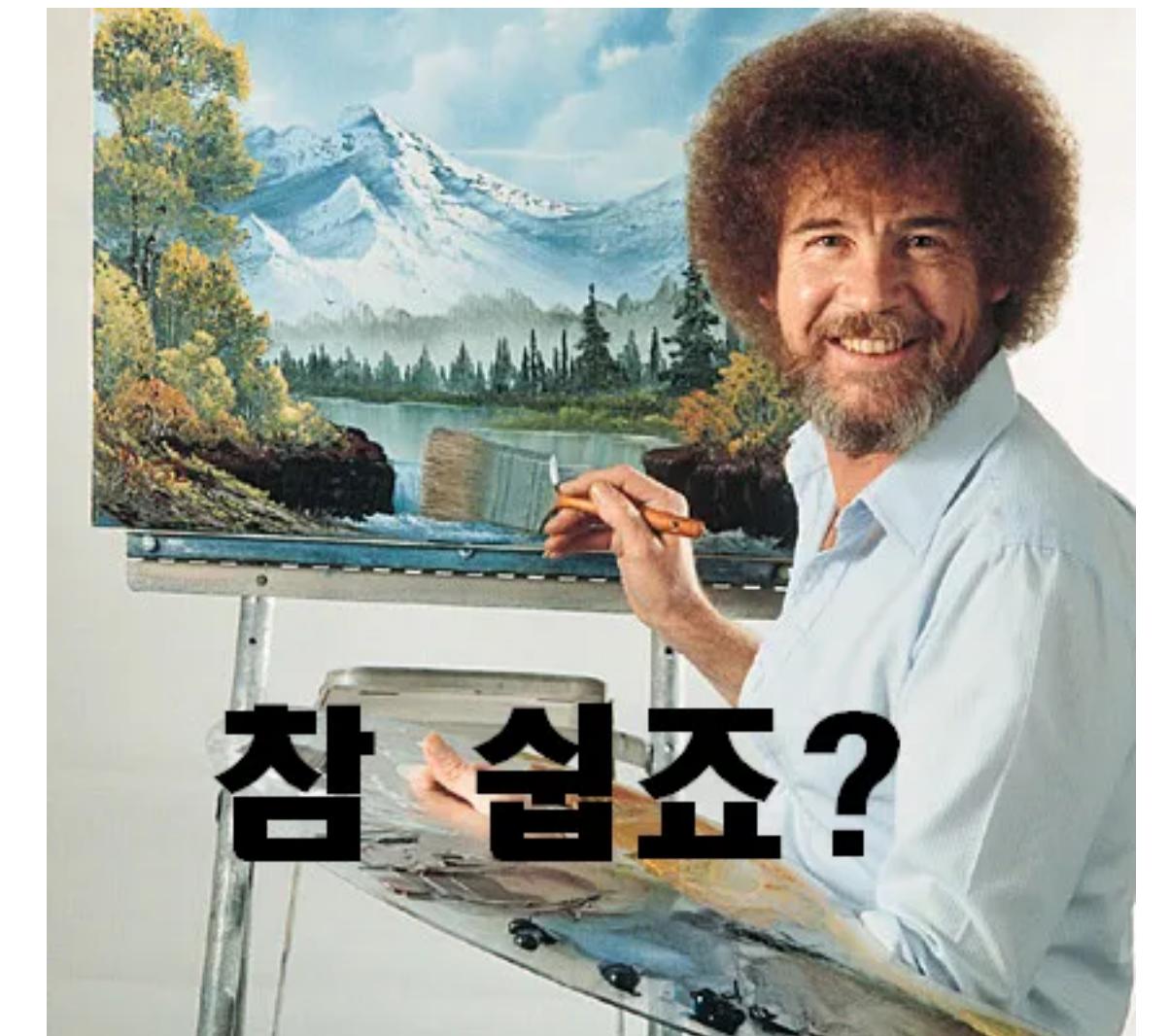
The screenshot shows the VESSL Hub interface. At the top, there's a navigation bar with a logo, 'Explore models', and 'Sign in'. Below it is a section titled 'VESSL Hub' with a sub-section 'Explore end-to-end YAML configurations of ready-to-run ML models.' There are six cards showcasing different AI models:

- Llama 2 fine-tuning**: Meta's Llama 2, a large language model with 70 billion parameters, is being fine-tuned on a 1.6k code instruction dataset using 8-bit quantization and LoRA. It features an image of a llama wearing sunglasses.
- Mistral 7B playground**: An open-source LLM which achieves both efficiency and high performance. It features an image of the number 7.
- SSD-1B playground**: Faster, smaller SSDL variant excelling in high-quality text-to-image generation with diverse dataset training, incorporating GPT and Midjourney data. It features a grid of generated images.
- Whisper V3 playground**: A model for automatic speech recognition and speech translation trained with large dataset of diverse audio. It features an image of a smartphone displaying an audio waveform.
- Stable Diffusion playground**: The Stable Diffusion WebUI from CivitAI offers an intuitive, user-friendly interface for generating high-quality AI-based art and images from a prompt. It features an image of an astronaut riding a horse.
- Your next model**: Starting with a YAML configuration, VESSL Run allows you to efficiently build and deploy your next model. It features a screenshot of the VESSL AI interface.

Agenda

우리 회사를 위한 LLM, 쉽고 빠르게 만들고 서비스하기 종합선물세트

- **Goals of Private LLM : Smarter, Faster & Cheaper**
 - Private LLM, 우리 회사에 왜 필요한가요?
 - 적어도 우리 회사의 업무에서만큼은 GPT-4보다 똑똑하게, GPT-3.5보다 빠르게 할 수는 없을까요?
- **How to Make LLM Smarter?**
 - 어떻게 내 모델을 더 똑똑하게 만들 수 있나요? 더 쉽고 빠르게 할 수 있는 방법은 없을까요?
- **How to Give LLM a Cheat Sheet?**
 - 모델이 생성하는 헛소리, 내가 가지고 있는 데이터로 대체할 수 있을까요?
- **How to Make LLM Service Cheaper & Faster?**
 - 서비스가 너무 느리고 운영 비용이 너무 비싸요! 어떻게 줄일 수 있을까요?
- **Further LLM Topics - Speculation, LLMOps, etc...**

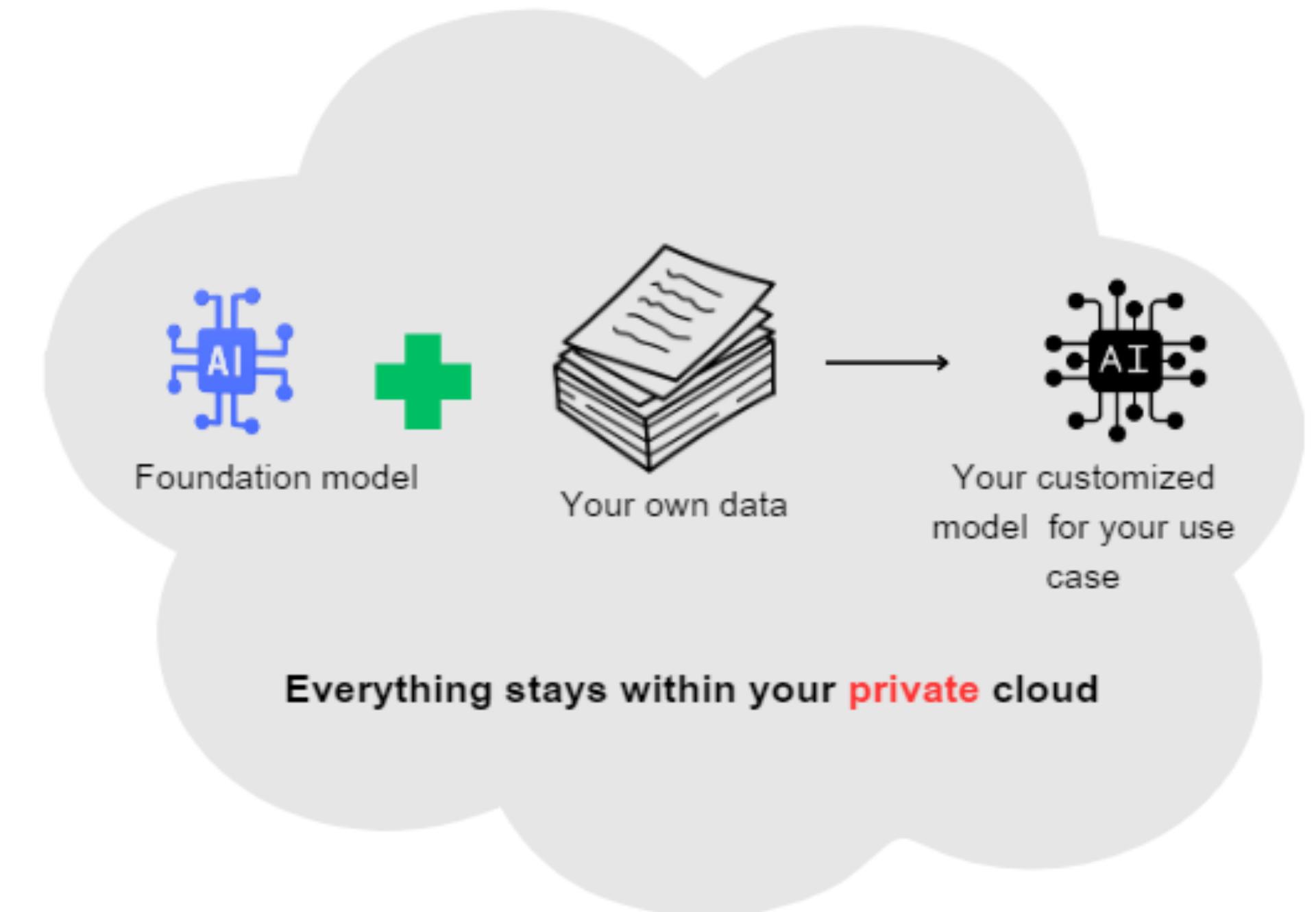


Goals of Private LLM: Smarter, Faster & Cheaper

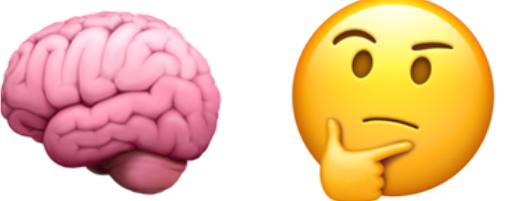
What's Private LLM?

Foundation Model + Private Data = Private LLM

- 데이터 보안을 위한 Private LLM
 - 외부 반출이 불가능한 민감데이터, 격리 환경 내 서비스 등
 - 적절한 성능의 Open-source LLM을 미세조정하여 사용
- 특정 도메인/작업 최적화를 위한 Private LLM
 - 코드, 법률/의료/금융, 사내 Q&A 서비스 등
 - 작업하려는 태스크에 맞게 모델 최적화가 필요
- 저비용 고효율을 위한 Private LLM
 - GPT-4: ~\$50 per million tokens
 - 일정 수준 이상 규모에서는 직접 호스팅이 저렴할수도?
(~2000 tokens per request == 1M tokens / hr?)



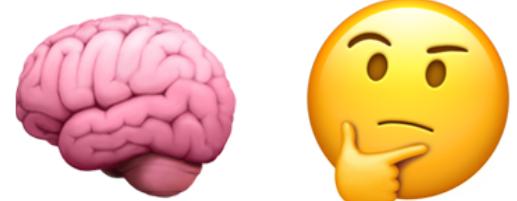
Goals of Private LLM



Smarter than GPT-4

적어도 내가 푸는 특정 문제에 대해서는 대답을 아주 잘 했으면 좋겠어요

Goals of Private LLM



Smarter than GPT-4

적어도 내가 푸는 특정 문제에 대해서는 대답을 아주 잘 했으면 좋겠어요



Cheaper than GPT3.5-Turbo

이왕 직접 만들어서 쓰는거 GPT3.5보다는 저렴하게 하고 싶어요

Goals of Private LLM



Smarter than GPT-4

적어도 내가 푸는 특정 문제에 대해서는 대답을 아주 잘 했으면 좋겠어요



Cheaper than GPT3.5-Turbo

이왕 직접 만들어서 쓰는거 GPT3.5보다는 저렴하게 하고 싶어요

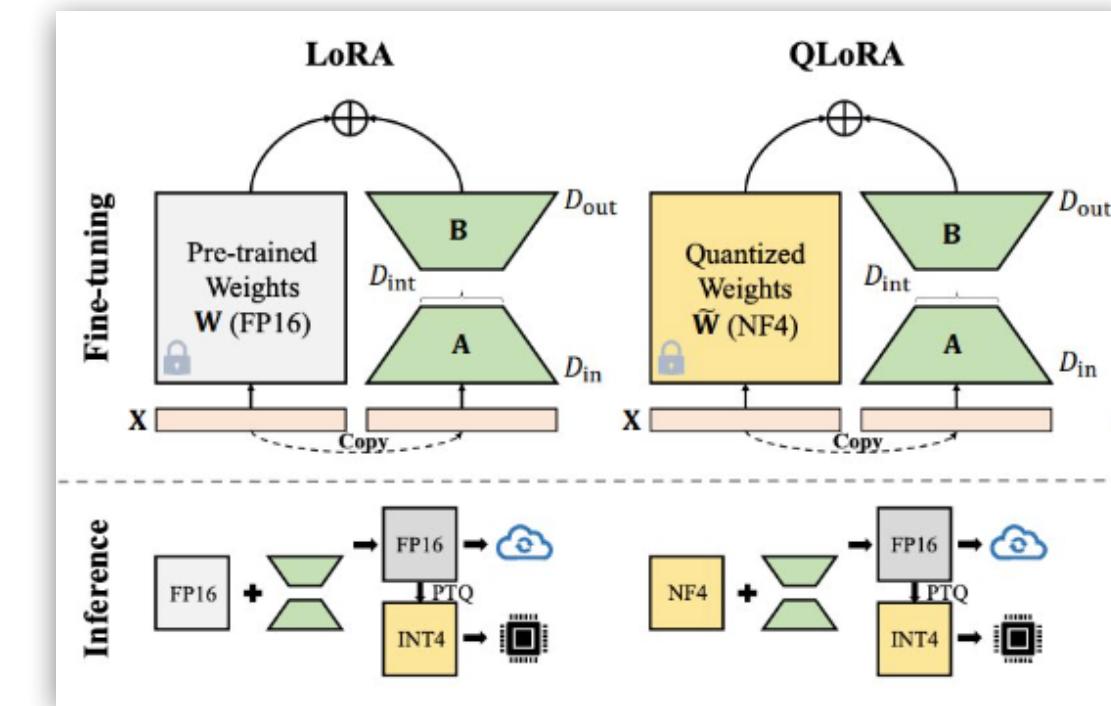


How to Make LLMs Smarter, Faster & Cheaper

LLM을 더 똑똑하면서 더 빠르게 만들기, 어떻게 했나요?

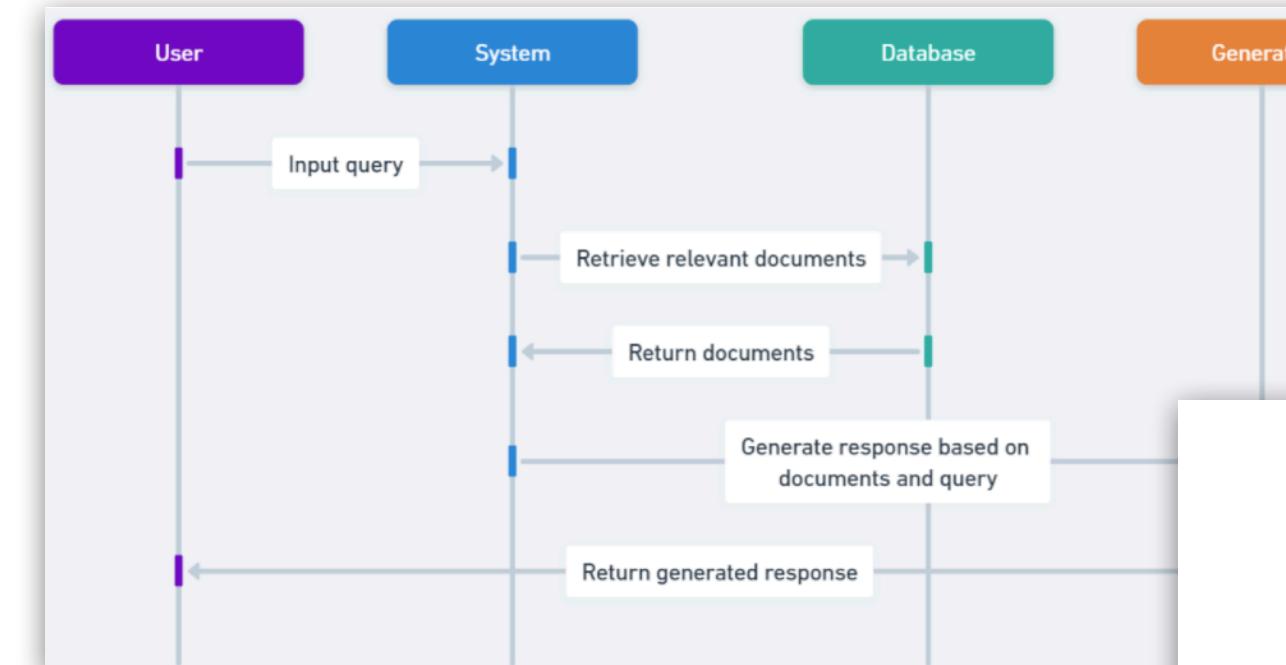
- **Make LLM “Smarter”: Fine-tuning**

- 추가로 학습할 부분만 효율적으로 학습하기 (PEFT)
- 파인튜닝이 적합한 작업을 잘 골라내기
- Fine-tuning의 효율을 두배 높이는 치트키



- **Give LLM a “Cheat-sheet”: RAG**

- RAG aka Retrieval Augmented Generation
- RAG의 성능을 50% 높이는 다양한 기법들



- **Reduce LLM's Burden: Quantize & Cache**

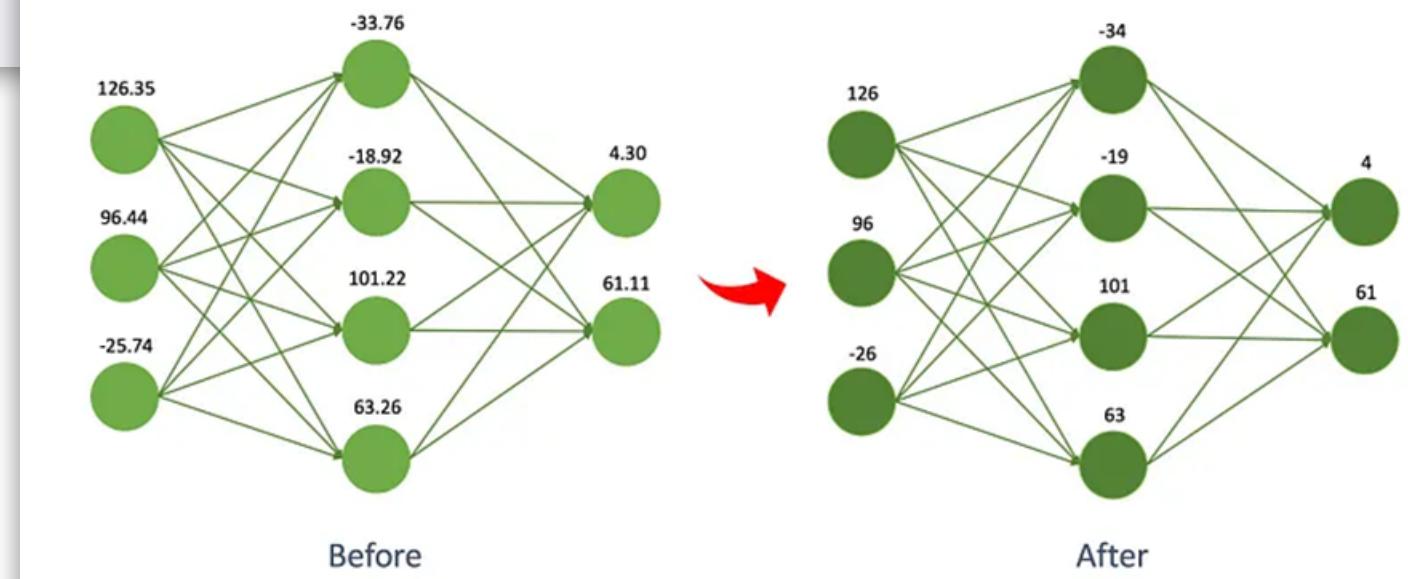
- 추론 성능을 가속화하기 위해 모델 사이즈 줄이고 캐싱하기
- Load-test 기반 실질적인 생성 비용 추정해보기

- **Structured output**

Another type of use case which works really well with structured information, in this case about sports

```
1 {"messages": [{"role": "system",  
2 {"role": "assistant", "content": "  
3 {"messages": [{"role": "system",  
4 {"role": "assistant", "content": "
```

Quantize LLMs Using AWQ



Fine-tuning: Make LLM Smarter

What & Why Fine-tune the Model?

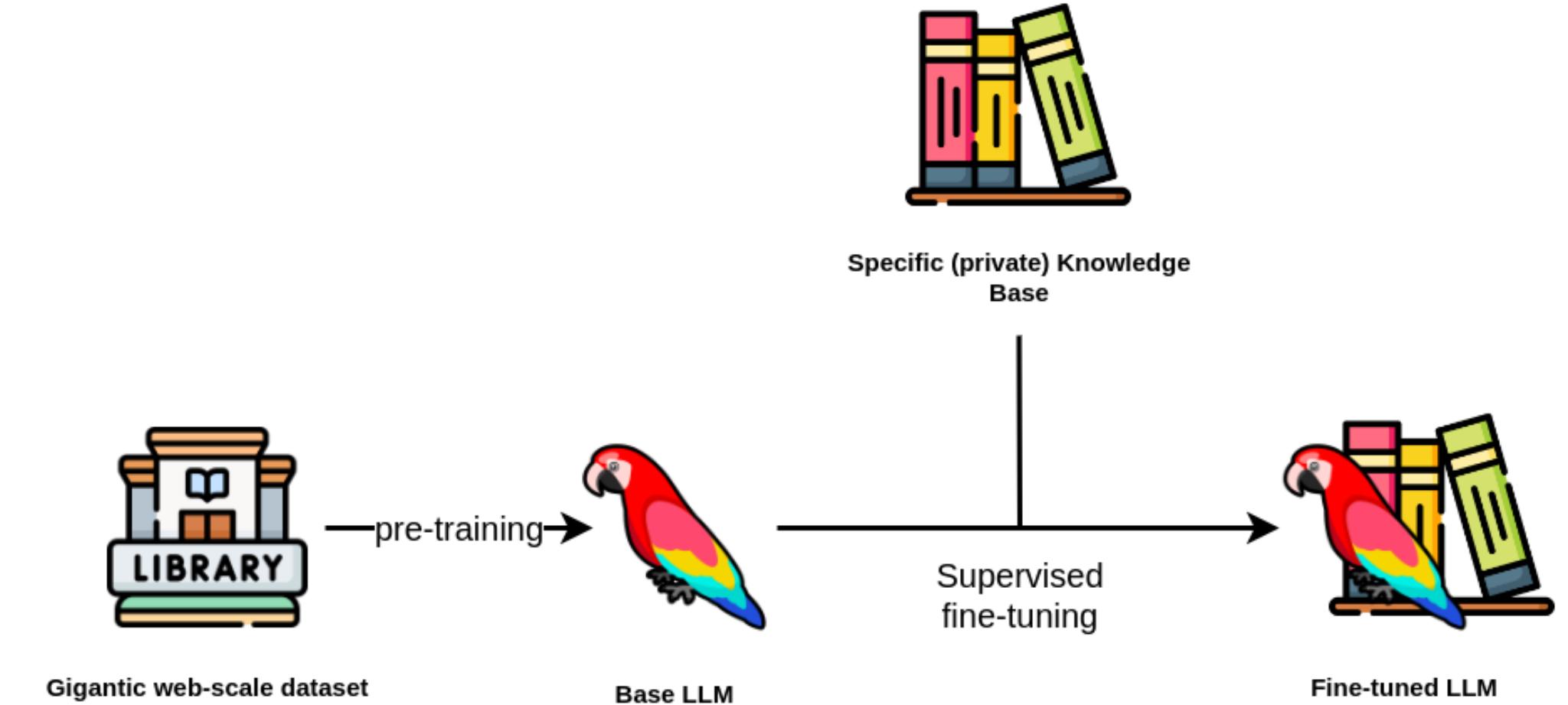
7B 사이즈 LLM도 학습하는데 \$300k 가까이 깨진다던데... [\(ref\)](#)

- **What is Fine-tuning?**

- 특정 작업을 더 잘 수행하기 위해 새로운 데이터로 모델을 추가로 학습시키는 과정

- **Why Fine-tuning?**

- 질문에 대한 대답 자체는 프롬프트 엔지니어링, RAG 등의 기법으로 가능하게 만들 수 있음
- “원하는 형태로 잘 할 수 있게” 하기 위해 fine-tuning이 필요함
- e.g. Structured output(JSON/YAML), 특정 도메인의 전반적인 지식과 자료 선제 학습, Prompt 길이 절약...



Base Model vs Fine-Tuned

Prompt alignment & Answer correctness

You are an AI assistant aiding a judge in a Korean court of law. Your task is to analyze the given 'fact' and 'reason', and subsequently estimate the judgement and the corresponding level of punishment.

Fact:

피고인 2020. 7. 4. 22:00경 남양주시 B호텔 부근 도로에서부터 남양주시 C 부근 도로에 이르기까지 약 200m 구간에서 혈중알코올농도 0.222%의 술에 취한 상태로 (차량번호 1 생략) 그랜저 승용차를 운전하였다.

Reason:

양형의 이유 이 사건 범행은 피고인이 혈중알코올농도 0.222%의 술에 취한 상태에서 약 200m 구간을 승용차로 운전한 사건으로, 이 사건 범행의 경위, 혈중알코올농도의 수치 정도 등에 비추어 볼 때 그 죄질이 가볍지 않은 점, 피고인은 재판 과정에 있어서도 피고인 소환장을 송달받고도 수회에 걸쳐 공판기일에 불출석 하는 등 이 사건 범행에 대하여 반성하고 있지도 않는 것으로 보이는 점 등을 피고인에게 불리한 정상이다. 다만, 피고인이 이 사건 범행을 인정하고 있는 점, 피고인이 1998년경 이후로는 동종 범죄로 처벌받은 전력이 없는 점 등을 피고인에게 유리한 정상이다. 위와 같은 정상들과 그 밖에 피고인의 나이, 성행, 환경, 이 사건 범행의 경위 및 범행 후의 정황 등이 사건 기록과 변론에 나타난 제반 양형 요소들을 참작하여, 주문과 같이 형을 정한다(이 사건 범행에 대해서는 양형기준이 설정되어 있지 아니하다).

To estimate the judgement, adhere to the following instructions:

Your output should be a JSON object with the structure as follows:

```
{  
  "judgement": "",  
  "fine_lv": ,  
  "imprisonment_with_labor_lv": ,  
  "imprisonment_without_labor_lv":  
}
```

1. }
2. The 'judgement' should contain simple Korean terminology referring to either a fine ("벌금 000원") or imprisonment ("징역 00월" or "금고 00월").
3. The levels of 'fine', 'imprisonment with labor', and 'imprisonment without labor' should be conveyed using a single integer value each.
4. Please keep your output concise, matching the length of the input for a balanced conversation.
5. The entire output must be in a well-formatted JSON structure without any other additional context.

Base model (7B)

Based on the given fact and reason, the judgement and the corresponding level of punishment can be estimated as follows:

```
{  
  "judgement": "징역 1년",  
  "fine_lv": 0,  
  "imprisonment_with_labor_lv": 12,  
  "imprisonment_without_labor_lv": 13  
}
```

The judgement is "징역 1년" which means imprisonment for one year. The levels of fine, imprisonment with labor, and imprisonment without labor are set to 0, 12, and 13 respectively.

Fine-tuned with 10M tokens

```
{  
  "judgement": "징역 12월",  
  "fine_lv": 0,  
  "imprisonment_with_labor_lv": 3,  
  "imprisonment_without_labor_lv": 0  
}
```

Base Model vs Fine-Tuned

Prompt alignment & Answer correctness

You are an AI assistant aiding a judge in a Korean court of law. Your task is to analyze the given 'fact' and 'reason', and subsequently estimate the judgement and the corresponding level of punishment.

Fact:

피고인 2020. 7. 4. 22:00경 남양주시 B호텔 부근 도로에서부터 남양주시 C 부근 도로에 이르기까지 약 200m 구간에서 혈중알코올농도 0.222%의 술에 취한 상태로 (차량번호 1 생략) 그랜저 승용차를 운전하였다.

Reason:

양형의 이유 이 사건 범행은 피고인이 혈중알코올농도 0.222%의 술에 취한 상태에서 약 200m 구간을 승용차로 운전한 사건으로, 이 사건 범행의 경위, 혈중알코올농도의 수치 정도 등에 비추어 볼 때 그 죄질이 가볍지 않은 점, 피고인은 재판 과정에 있어서도 피고인 소환장을 송달받고도 수회에 걸쳐 공판기일에 불출석 하는 등 이 사건 범행에 대하여 반성하고 있지도 않는 것으로 보이는 점 등을 피고인에게 불리한 정상이다. 다만, 피고인이 이 사건 범행을 인정하고 있는 점, 피고인이 1998년경 이후로는 동종 범죄로 처벌받은 전력이 없는 점 등을 피고인에게 유리한 정상이다. 위와 같은 정상들과 그 밖에 피고인의 나이, 성행, 환경, 이 사건 범행의 경위 및 범행 후의 정황 등이 사건 기록과 변론에 나타난 제반 양형 요소들을 참작하여, 주문과 같이 형을 정한다(이 사건 범행에 대해서는 양형기준이 설정되어 있지 아니하다).

To estimate the judgement, adhere to the following instructions:

Your output should be a JSON object with the structure as follows:

```
{  
  "judgement": "",  
  "fine_lv": ,  
  "imprisonment_with_labor_lv": ,  
  "imprisonment_without_labor_lv":  
}
```

1. }
2. The 'judgement' should contain simple Korean terminology referring to either a fine ("벌금 000원") or imprisonment ("징역 00월" or "금고 00월").
3. The levels of 'fine', 'imprisonment with labor', and 'imprisonment without labor' should be conveyed using a single integer value each.
4. Please keep your output concise, matching the length of the input for a balanced conversation.
5. The entire output must be in a well-formatted JSON structure without any other additional context.

Base model (7B)

Based on the given fact and reason, the judgement and the corresponding level of punishment can be estimated as follows:

```
{  
  "judgement": "징역 1년",  
  "fine_lv": 0,  
  "imprisonment_with_labor_lv": 12,  
  "imprisonment_without_labor_lv": 13  
}
```

Not aligning with prompt (JSON output)

The judgement is "징역 1년" which means imprisonment for one year. The levels of fine, imprisonment with labor, and imprisonment without labor are set to 0, 12, and 13 respectively.

Fine-tuned with 10M tokens

```
{  
  "judgement": "징역 12월",  
  "fine_lv": 0,  
  "imprisonment_with_labor_lv": 3,  
  "imprisonment_without_labor_lv": 0  
}
```

Base Model vs Fine-Tuned

Prompt alignment & Answer correctness

You are an AI assistant aiding a judge in a Korean court of law. Your task is to analyze the given 'fact' and 'reason', and subsequently estimate the judgement and the corresponding level of punishment.

Fact:

피고인 2020. 7. 4. 22:00경 남양주시 B호텔 부근 도로에서부터 남양주시 C 부근 도로에 이르기까지 약 200m 구간에서 혈중알코올농도 0.222%의 술에 취한 상태로 (차량번호 1 생략) 그랜저 승용차를 운전하였다.

Reason:

양형의 이유 이 사건 범행은 피고인이 혈중알코올농도 0.222%의 술에 취한 상태에서 약 200m 구간을 승용차로 운전한 사건으로, 이 사건 범행의 경위, 혈중알코올농도의 수치 정도 등에 비추어 볼 때 그 죄질이 가볍지 않은 점, 피고인은 재판 과정에 있어서도 피고인 소환장을 송달받고도 수회에 걸쳐 공판기일에 불출석 하는 등 이 사건 범행에 대하여 반성하고 있지도 않는 것으로 보이는 점 등을 피고인에게 불리한 정상이다. 다만, 피고인이 이 사건 범행을 인정하고 있는 점, 피고인이 1998년경 이후로는 동종 범죄로 처벌받은 전력이 없는 점 등을 피고인에게 유리한 정상이다. 위와 같은 정상들과 그 밖에 피고인의 나이, 성행, 환경, 이 사건 범행의 경위 및 범행 후의 정황 등이 사건 기록과 변론에 나타난 제반 양형 요소들을 참작하여, 주문과 같이 형을 정한다(이 사건 범행에 대해서는 양형기준이 설정되어 있지 아니하다).

To estimate the judgement, adhere to the following instructions:

Your output should be a JSON object with the structure as follows:

```
{  
  "judgement": "",  
  "fine_lv": ,  
  "imprisonment_with_labor_lv": ,  
  "imprisonment_without_labor_lv":  
}
```

1. }
2. The 'judgement' should contain simple Korean terminology referring to either a fine ("벌금 000원") or imprisonment ("징역 00월" or "금고 00월").
3. The levels of 'fine', 'imprisonment with labor', and 'imprisonment without labor' should be conveyed using a single integer value each.
4. Please keep your output concise, matching the length of the input for a balanced conversation.
5. The entire output must be in a well-formatted JSON structure without any other additional context.

Base model (7B)

Based on the given fact and reason, the judgement and the corresponding level of punishment can be estimated as follows:

```
{  
  "judgement": "징역 1년",  
  "fine_lv": 0,  
  "imprisonment_with_labor_lv": 12,  
  "imprisonment_without_labor_lv": 13  
}
```

Wrong answer

(Caused by unknown information)

The judgement is "징역 1년" which means imprisonment for one year. The levels of fine, imprisonment with labor, and imprisonment without labor are set to 0, 12, and 13 respectively.

Fine-tuned with 10M tokens

```
{  
  "judgement": "징역 12월",  
  "fine_lv": 0,  
  "imprisonment_with_labor_lv": 3,  
  "imprisonment_without_labor_lv": 0  
}
```

Base Model vs Fine-Tuned

Prompt alignment & Answer correctness

You are an AI assistant aiding a judge in a Korean court of law. Your task is to analyze the given 'fact' and 'reason', and subsequently estimate the judgement and the corresponding level of punishment.

Fact:

피고인 2020. 7. 4. 22:00경 남양주시 B호텔 부근 도로에서부터 남양주시 C 부근 도로에 이르기까지 약 200m 구간에서 혈중알코올농도 0.222%의 술에 취한 상태로 (차량번호 1 생략) 그랜저 승용차를 운전하였다.

Reason:

양형의 이유 이 사건 범행은 피고인이 혈중알코올농도 0.222%의 술에 취한 상태에서 약 200m 구간을 승용차로 운전한 사건으로, 이 사건 범행의 경위, 혈중알코올농도의 수치 정도 등에 비추어 볼 때 그 죄질이 가볍지 않은 점, 피고인은 재판 과정에 있어서도 피고인 소환장을 송달받고도 수회에 걸쳐 공판기일에 불출석 하는 등 이 사건 범행에 대하여 반성하고 있지도 않는 것으로 보이는 점 등을 피고인에게 불리한 정상이다. 다만, 피고인이 이 사건 범행을 인정하고 있는 점, 피고인이 1998년경 이후로는 동종 범죄로 처벌받은 전력이 없는 점 등을 피고인에게 유리한 정상이다. 위와 같은 정상들과 그 밖에 피고인의 나이, 성행, 환경, 이 사건 범행의 경위 및 범행 후의 정황 등 이 사건 기록과 변론에 나타난 제반 양형 요소들을 참작하여, 주문과 같이 형을 정한다(이 사건 범행에 대해서는 양형기준이 설정되어 있지 아니하다).

To estimate the judgement, adhere to the following instructions:

Your output should be a JSON object with the structure as follows:

```
{  
  "judgement": "",  
  "fine_lv": ,  
  "imprisonment_with_labor_lv": ,  
  "imprisonment_without_labor_lv":  
}
```

1. }
2. The 'judgement' should contain simple Korean terminology referring to either a fine ("벌금 000원") or imprisonment ("징역 00월" or "금고 00월").
3. The levels of 'fine', 'imprisonment with labor', and 'imprisonment without labor' should be conveyed using a single integer value each.
4. Please keep your output concise, matching the length of the input for a balanced conversation.
5. The entire output must be in a well-formatted JSON structure without any other additional context.

Base model (7B)

Based on the given fact and reason, the judgement and the corresponding level of punishment can be estimated as follows:

```
{  
  "judgement": "징역 1년",  
  "fine_lv": 0,  
  "imprisonment_with_labor_lv": 12,  
  "imprisonment_without_labor_lv": 13  
}
```

The judgement is "징역 1년" which means imprisonment for one year. The levels of fine, imprisonment with labor, and imprisonment without labor are set to 0, 12, and 13 respectively.

Fine-tuned with 10M tokens

```
{  
  "judgement": "징역 12월",  
  "fine_lv": 0,  
  "imprisonment_with_labor_lv": 3,  
  "imprisonment_without_labor_lv": 0  
}
```

Correct, well-structured response

Improving Fine-tuning Efficiency

파인튜닝을 훨씬 더 효율적으로, 저렴하게, 빠르게 해보자

• Parameter-Efficient Fine Tuning

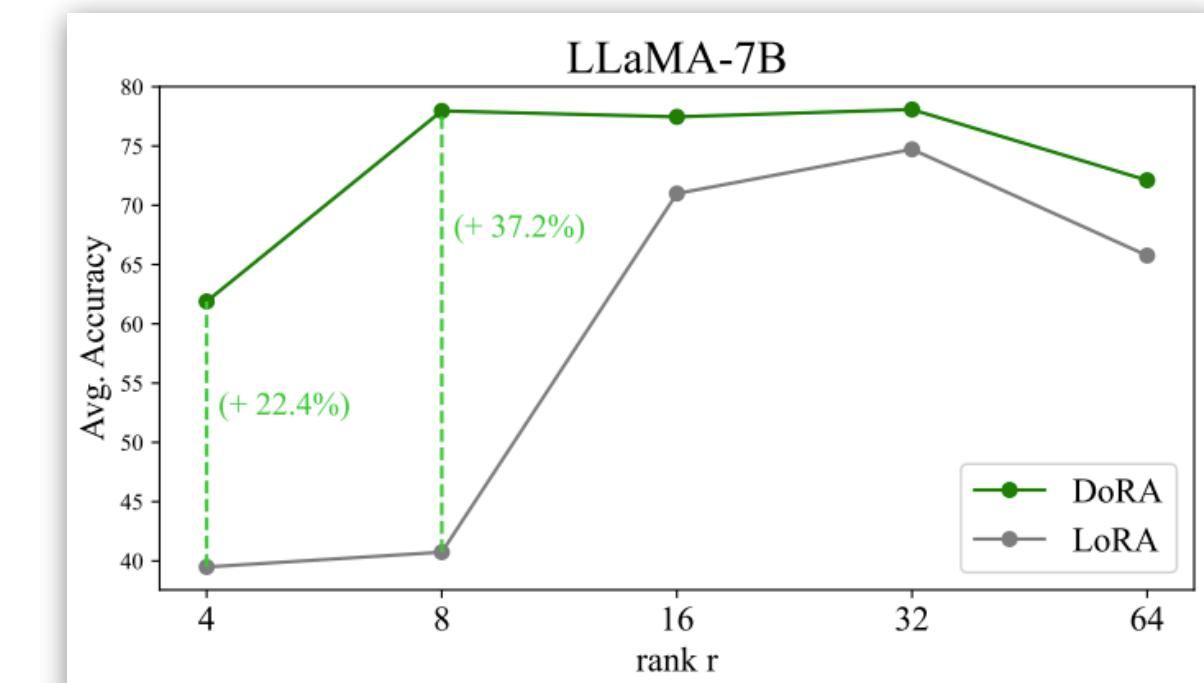
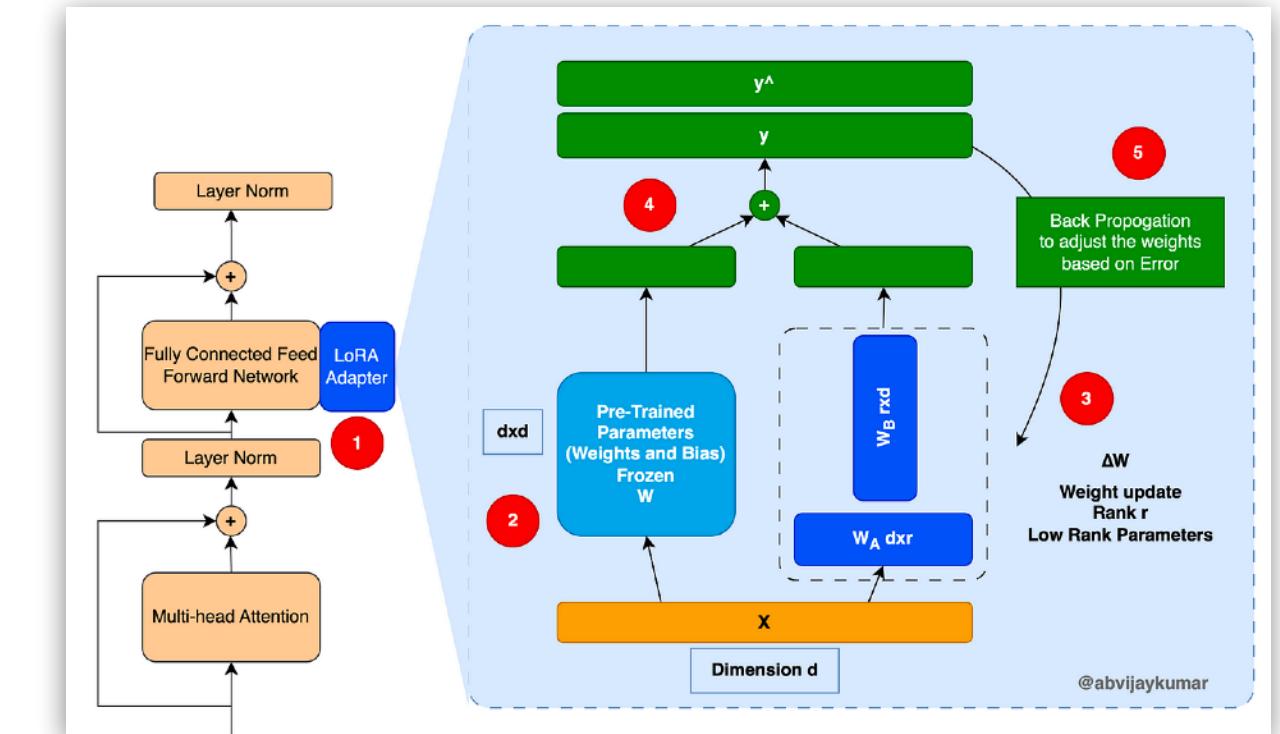
- 기존 모델의 대부분의 Parameter를 고정(freeze) 해 둔 상태에서 일부 weights만 학습하는 방식
- Low-Rank Adaptation(LoRA)라는 기법이 주로 쓰임
- 양자화(Quantized)된 모델에서 하는 QLoRA - 훨씬 더 적은 메모리로 학습

• DoRA

- LoRA의 가중치를 크기(M)와 방향(D) 구성 요소로 분해해서 더 세밀한 업데이트를 가능하게 만드는 기법
- 실험 결과 낮은 Rank(=더 적은 계산 오버헤드)에서도 상대적으로 고성능

• Unsloth (<https://unsloth.ai/>)

- 최적화된 GPU 커널 구현체로 LoRA / QLoRA 학습을 추가 가속 & GPU 메모리를 절약하는 라이브러리
- 사용시 약간 불안정했지만 2배 이상의 학습 속도 가속 가능



Free Colab T4	Dataset	Hugging Face	Pytorch 2.1.1	Unsloth	VRAM reduction
Llama-2 7b	OASST	1x	1.19x	1.95x	-43.3%
Mistral 7b	Alpaca	1x	1.07x	1.56x	-13.7%
Tiny Llama 1.1b	Alpaca	1x	2.06x	3.87x	-73.8%
DPO with Zephyr	Ultra Chat	1x	1.09x	1.55x	-18.6%

Evaluation: Experiment Setting

파인튜닝 비용과 성능, 직접 실험 해보았습니다

- Ibox/Ibox_open 데이터셋의 형사재판 기록

- 상황과 양형배경 바탕으로 벌금/징역 여부 및 형량의 수준 판단
- 프롬프트 포함 8400 chunks, 10M Tokens - 책 6-7권 분량
- 처리를 위한 JSON output 강제

- Fine-tuning with NVIDIA L4

- 시간당 \$1 내외에 클라우드로 사용 가능한 저렴한 GPU
- VRAM 24GB 규모로 주로 추론용으로 사용

- Model Configuration

- mistralai/Mistral-7B-Instruct-v0.2, rank=8, lora_alpha=32
- all linear layers (~1% trainable parameters)

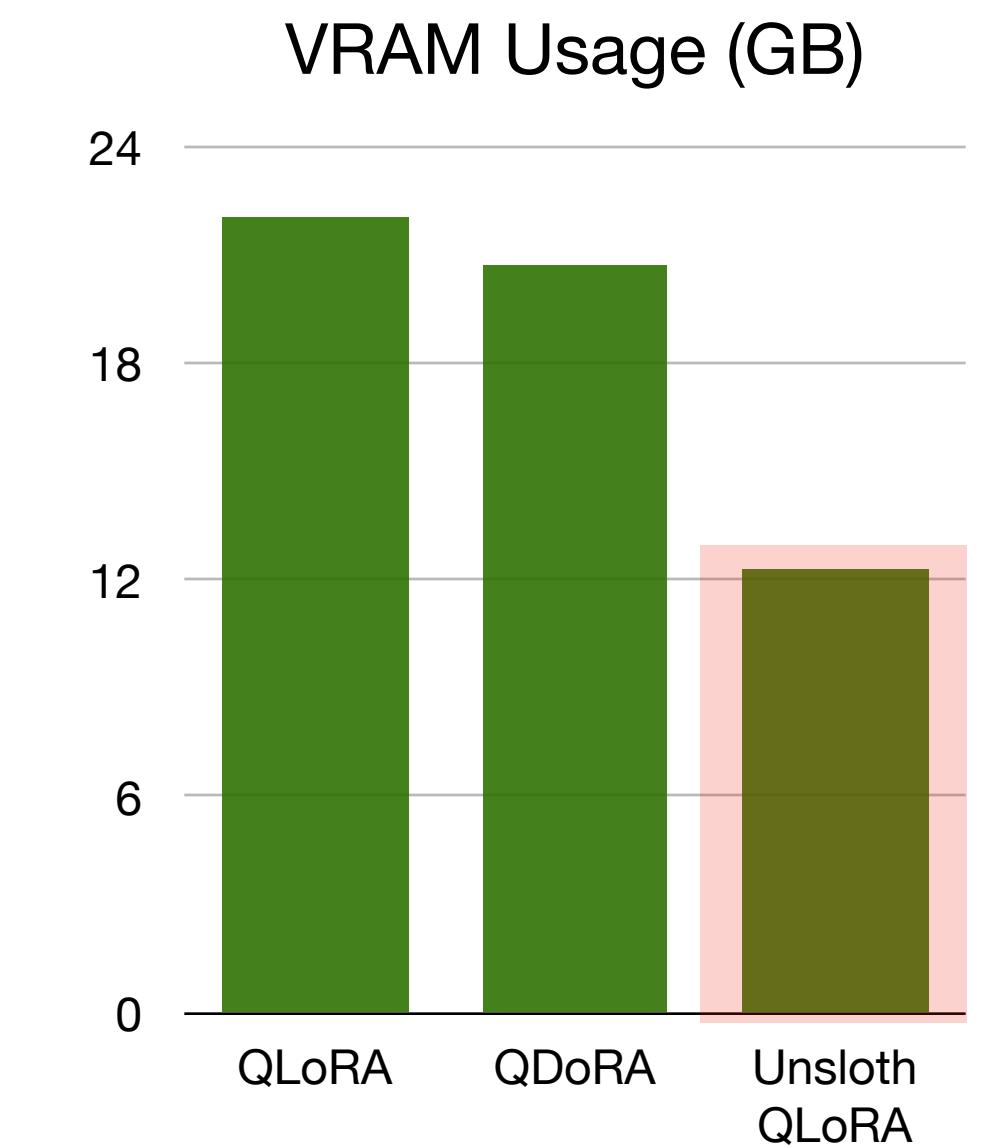
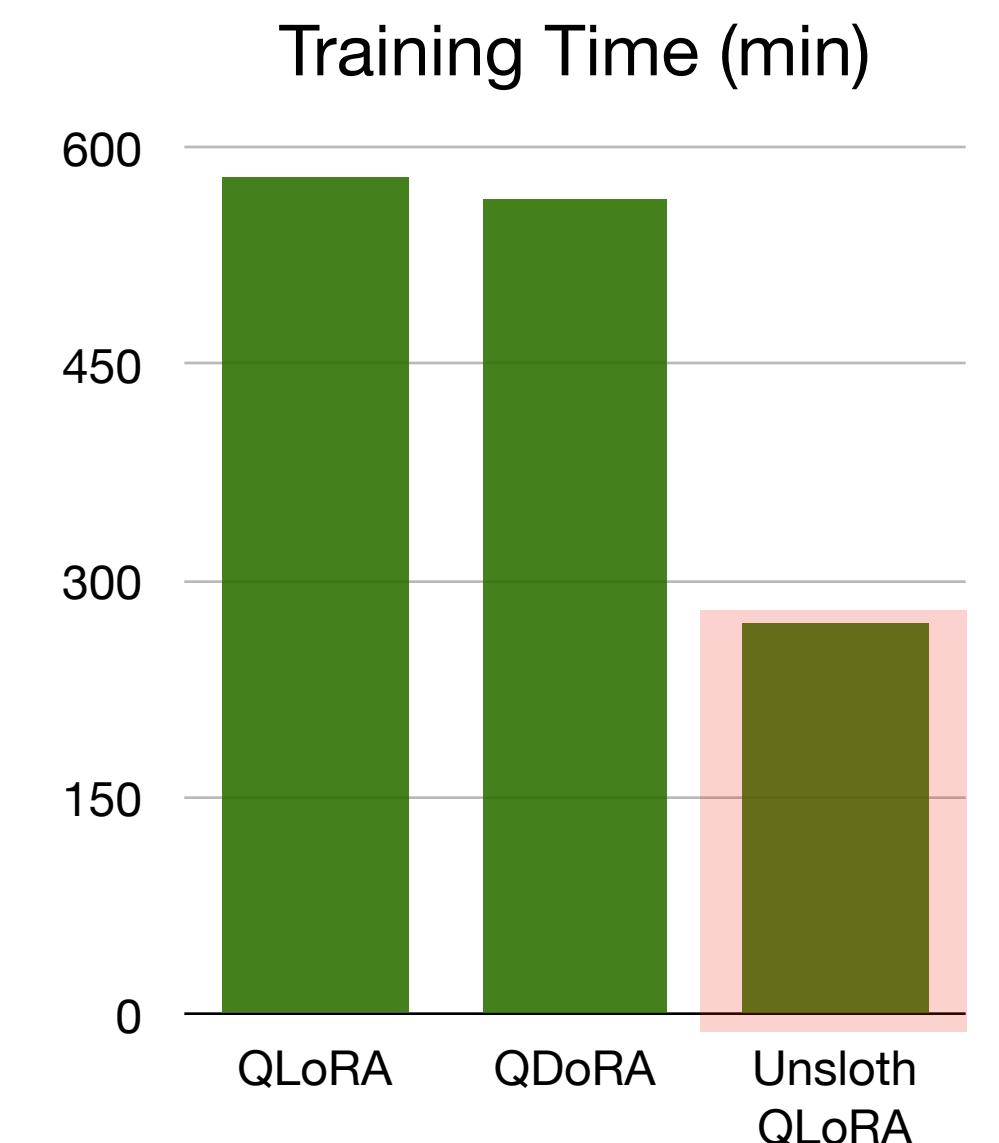
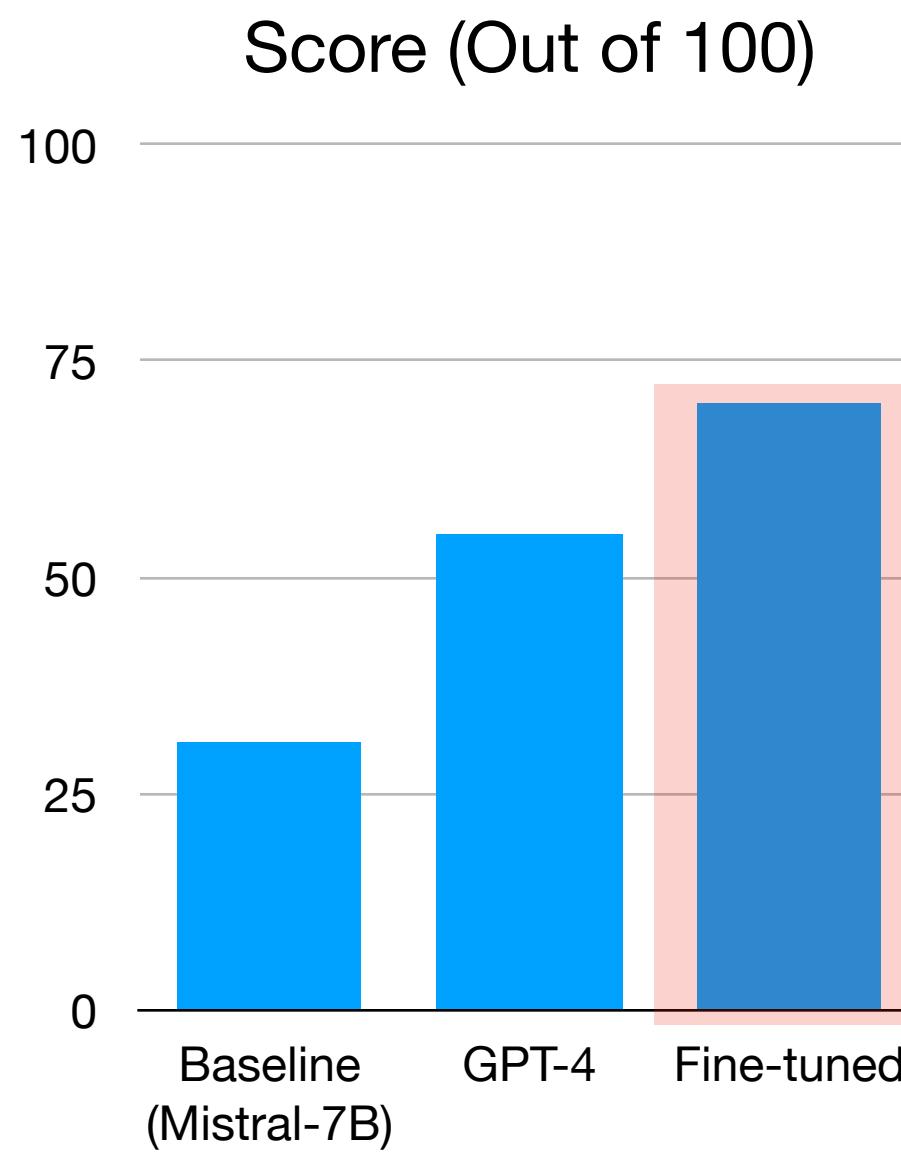
피고인은 2007. 9. 14. 수원지방법원 안산지 원에서 벌금 150만 원의 악식명령을 발령받았...	양형의 이유 피고인이 자신의 잘못을 인정하는 점 및 피고인의 범죄전력, 혈중알코올농도 수...	{ "text": "벌금 1000000원", "fine_lv": 4, "imprisonment_with_labor_lv": 0,.. }
범죄전력 피고인은 2006. 10. 12. 서울중앙 지방법원에서 도로교통법위반(음주운전)죄로 벌...	양형의 이유 피고인은 이미 음주운전으로 2회 악식명령의 처벌을 받은 전력이 있음에도 다시...	{ "text": "벌금 1200000원", "fine_lv": 4, "imprisonment_with_labor_lv": 0,.. }
피고인은 2020. 10. 7. 22:05경 혈중알코 올농도 0.094%의 술에 취한 상태로 김해시 ...	양형의 이유 아래와 같은 정상 및 피고인의 연 령, 성행, 환경, 범행의 동기, 수단과 결과,...	{ "text": "징역 10월", "fine_lv": 0, "imprisonment_with_labor_lv": 2,.. }
범죄전력 피고인은 2019. 6. 20. 수원지방법 원 안산지원에서 도로교통법위반(음주운전)죄로...	양형의 이유 피고인은 2019년 도로교통법위반 (음주운전)죄로 처벌받은 전력이 있음에도 재...	{ "text": "징역 12월", "fine_lv": 0, "imprisonment_with_labor_lv": 3,.. }
피고인은 2010. 12. 23. 수원지방법원 여주 지원에서 도로교통법위반(음주운전)죄로 벌금...	양형의 이유 피고인의 범죄전력(동종전력의 존부 및 그 횟수, 직전 동종전력과의 시간 간격...)	{ "text": "징역 12월", "fine_lv": 0, "imprisonment_with_labor_lv": 3,.. }
피고인은 2021. 10. 26. 02:08경 순천시 서면에 있는 불상의 도로에서부터 같은 시 백...	양형의 이유 이 사건 당시 피고인의 혈중알코올 농도가 비교적 높았고, 음주운전의 결과 도로...	{ "text": "벌금 1000000원", "fine_lv": 4, "imprisonment_with_labor_lv": 0,.. }
범죄전력 피고인은 2020. 7. 31. 인천지방법 원에서 도로교통법위반(음주운전)죄로 벌금...	양형의 이유 음주운전은 운전자 자신뿐만 아니라 도로를 이용하는 시민들의 생명·신체를 침해할...	{ "text": "징역 18월", "fine_lv": 0, "imprisonment_with_labor_lv": 4,.. }
피고인은 2011. 12. 19. 대전지방법원에서 도로교통법위반(음주운전) 등으로 징역 6월 ...	양형의 이유 생략	{ "text": "징역 18월", "fine_lv": 0, "imprisonment_with_labor_lv": 4,.. }
피고인은 2021. 6. 8. 00:19경 부천시 B 앞 도로에서 주차되어 있던 차량에 시동을 걸...	양형의 이유 범행을 모두 인정하고 있는 점과 혈중알코올농도의 수치, 운행 거리, 동종범죄...	{ "text": "징역 12월", "fine_lv": 0, "imprisonment_with_labor_lv": 3,.. }
피고인은 2021. 10. 30. 22:21 거제시 B 아파트 주차장에서부터 같은 시 C 앞 도로에...	양형의 이유 이 사건 음주운전에 이른 경위와 당시 음주수치, 피고인의 종전 동종범행으로 ...	{ "text": "징역 14월", "fine_lv": 0, "imprisonment_with_labor_lv": 3,.. }
피고인은 2009. 3. 19. 의정부지방법원에서 도로교통법위반(음주운전)죄로 벌금 70만 원...	양형의 이유 피고인은 2003년, 2009년에 각 음주운전으로 처벌받은 전력이 있음에도 또다시...	{ "text": "징역 12월", "fine_lv": 0, "imprisonment_with_labor_lv": 3,.. }
범죄전력 피고인은 2011. 7. 11. 창원지방법 원에서 도로교통법위반(음주운전)죄로 벌금...	양형의 이유 1. 법률상 처단형의 범위: 징역 1년~2년 6월 2. 선고형의 결정: 징역 1년,..	{ "text": "징역 12월", "fine_lv": 0, "imprisonment_with_labor_lv": 3,.. }
피고인은 2015. 12. 4. 서울동부지방법원에 서 도로교통법위반(음주운전)죄 등으로 벌금...	양형의 이유 음주운전은 언제든지 사람의 생명을 앗아갈 위험성을 내포한 행위로 죄질이 매우 ...	{ "text": "징역 18월", "fine_lv": 0, "imprisonment_with_labor_lv": 4,.. }
범죄전력 피고인은 2008. 3. 11. 수원지방법 원 안산지원에서 도로교통법위반(음주운전)죄로...	양형의 이유 피고인은 2008년, 2013년 각 도 로교통법위반(음주운전)죄로 처벌받은 전력이 ...	{ "text": "징역 12월", "fine_lv": 0, "imprisonment_with_labor_lv": 3,.. }
피고인은 2013. 2. 12. 대구지방법원에서 도 로교통법위반(음주측정거부)죄로 벌금 500만...	양형의 이유 피고인은 음주운전으로 2회, 음주 측정거부로 3회 처벌받은 전력이 있다. 그중...	{ "text": "징역 12월", "fine_lv": 0, "imprisonment_with_labor_lv": 3,.. }
피고인은 2014. 1. 24. 춘천지방법원 영월지	양형의 이유 피고인은 이미 동종 범행으로 여러	{ "text": "징역 16월", "fine_lv": 0, "imprisonment_with_labor_lv": 4,.. }



Evaluation Result

Scoring - 예측의 유사도 기반 0~2점, 50개 evaluation sample

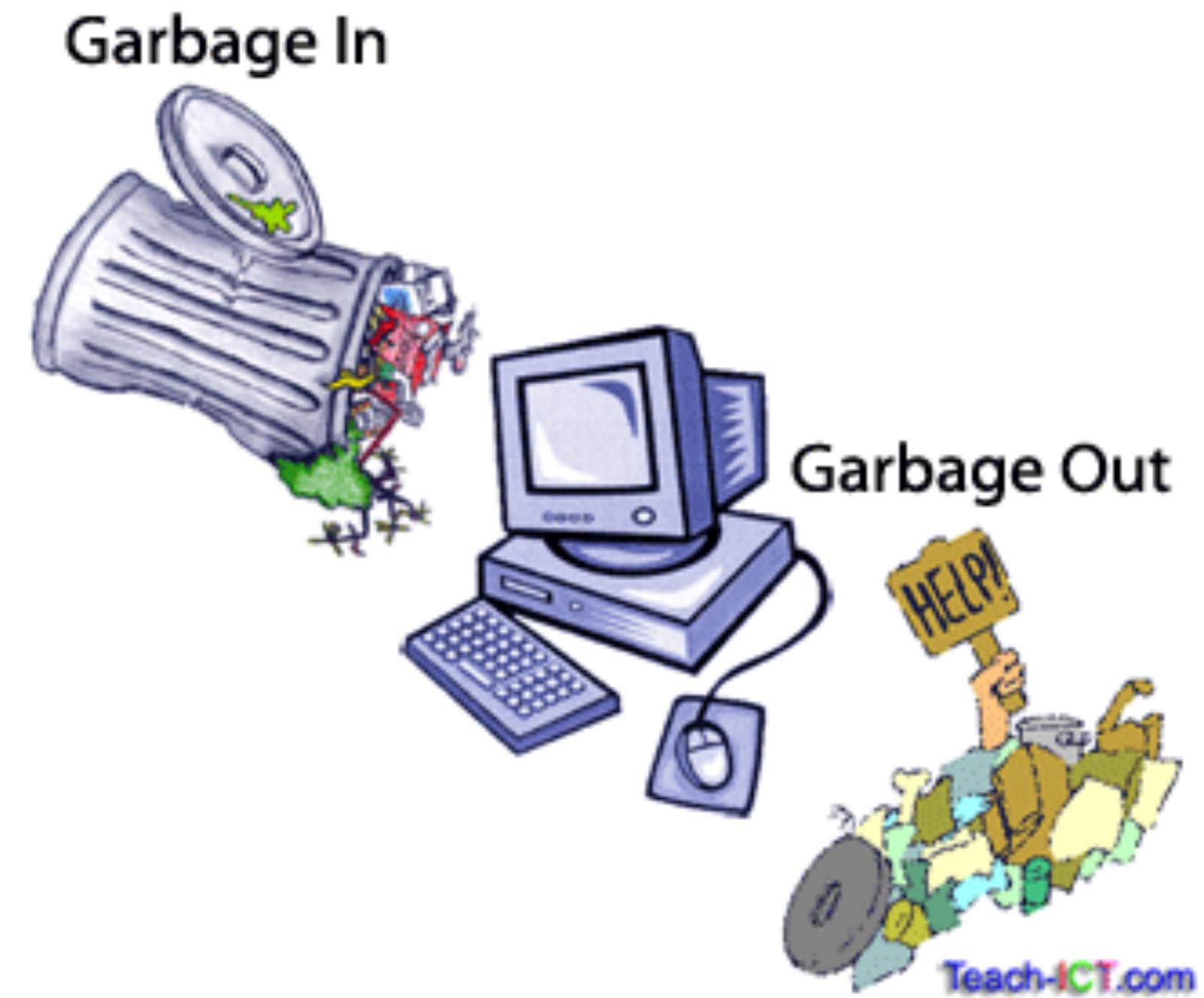
- 답변의 정형화에 Fine-tuning은 매우 강력
 - Baseline의 오답 중 상당 부분이 JSON 출력 실패
 - GPT-4는 Prompt를 상당히 잘 따랐으나, 모르는 정보에 대해 엉뚱한 답변을 내놓는 경우가 많았음
 - 형량을 완전히 잘못 판단, 의미없는 텍스트 등...
 - 4-5개 sample 기반 RAG는 개선이 크지 않았음
- QLoRA vs QDoRA vs Unslot
 - LoRA와 DoRA의 training cost는 큰 차이가 없음
 - DORA의 성능이 약간 더 좋았으나, 대동소이
 - Unslot: 학습 속도 2x, VRAM 사용량 50%
- Thoughts & Discussions
 - Alignment & Safety 등은 다른 기법을 고려 - DPO 등



Key Takeaways: Fine-tuning

LLM을 파인튜닝 할 때 고려해야 할 점들

- “Garbage in, garbage out”
 - 가장 중요한 것은 결국 태스크에 맞게 잘 정제된 데이터
 - 데이터의 품질을 높이는 것이 가장 어렵지만 가장 효과가 좋음
- GPT-4 level can be achieved in \$10~\$100
 - (물론 특정 태스크에 한해서)
 - GPU price \$0.978 / hr ([ref](#)) \times 4hr~9hr == less than \$10
 - 책 수십권 분량의 비교적 많은 데이터로 학습해도 \$100 내외
 - 더 고성능의 GPU 사용 시 실제 학습 비용은 더 낮아질 수도
- Fine tuning also needs extra prompts
 - 적당한 프롬프트를 학습 corpus에 포함시켰을 때 성능 크게 향상
 - 특히 JSON, API call 등 정형화된 output이 필요 시 필수적



Retrieval-Augmented Generation: Give LLM a Cheat-sheet

Quick RAG 101

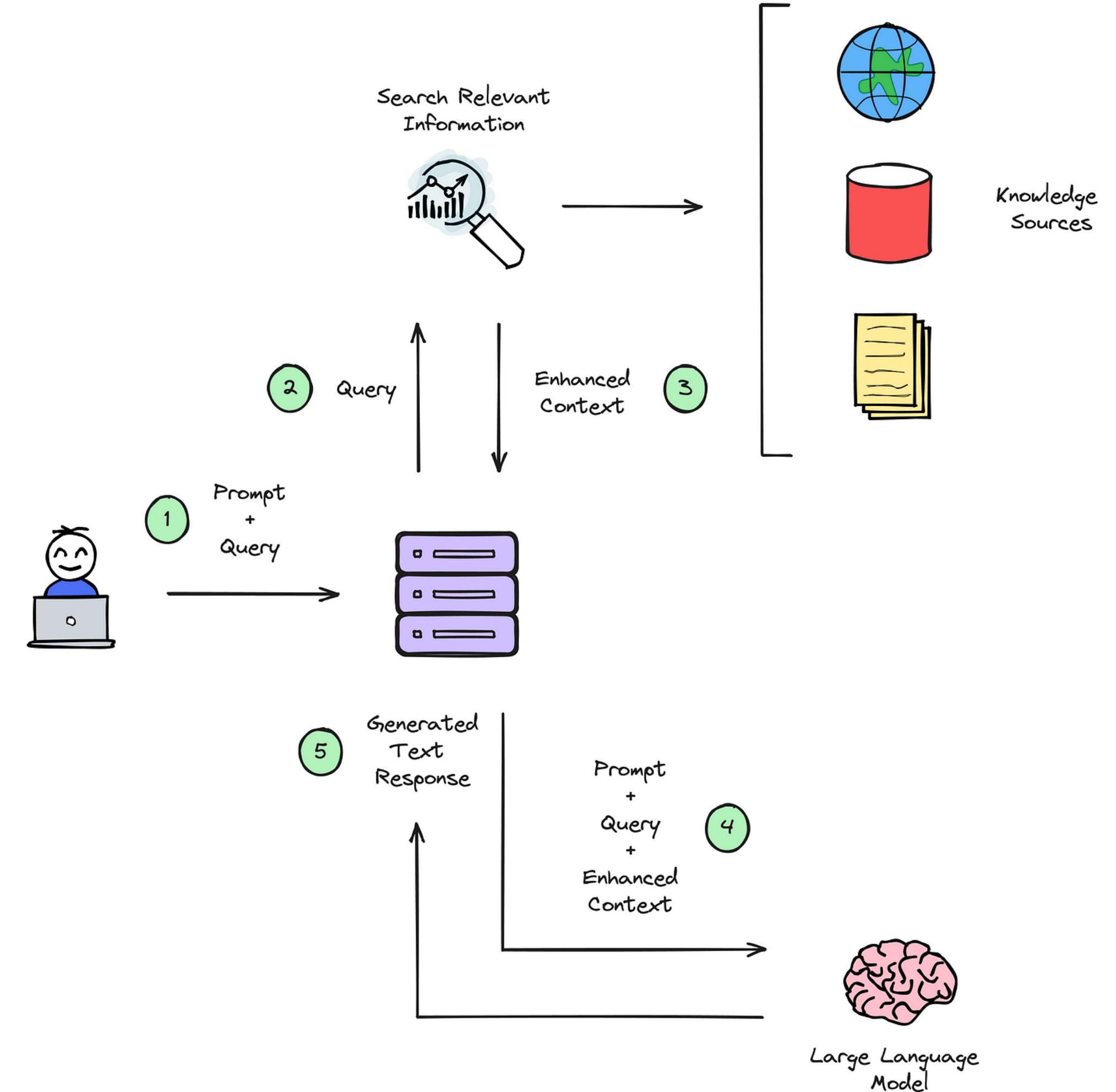
혹시 모르니 기본 개념을 설명하고 넘어갑니다

- **What is RAG?**

- Retrieval-Augmented Generation (검색 증강 생성)
- 입력할 질문과 상관이 있는 데이터를 검색해서 “가져온 맥락”과 질문을 함께 LLM의 생성에 사용
- 답변의 “출처”가 중요한 Q&A system 등에서 폭넓게 사용

- **Why RAG?**

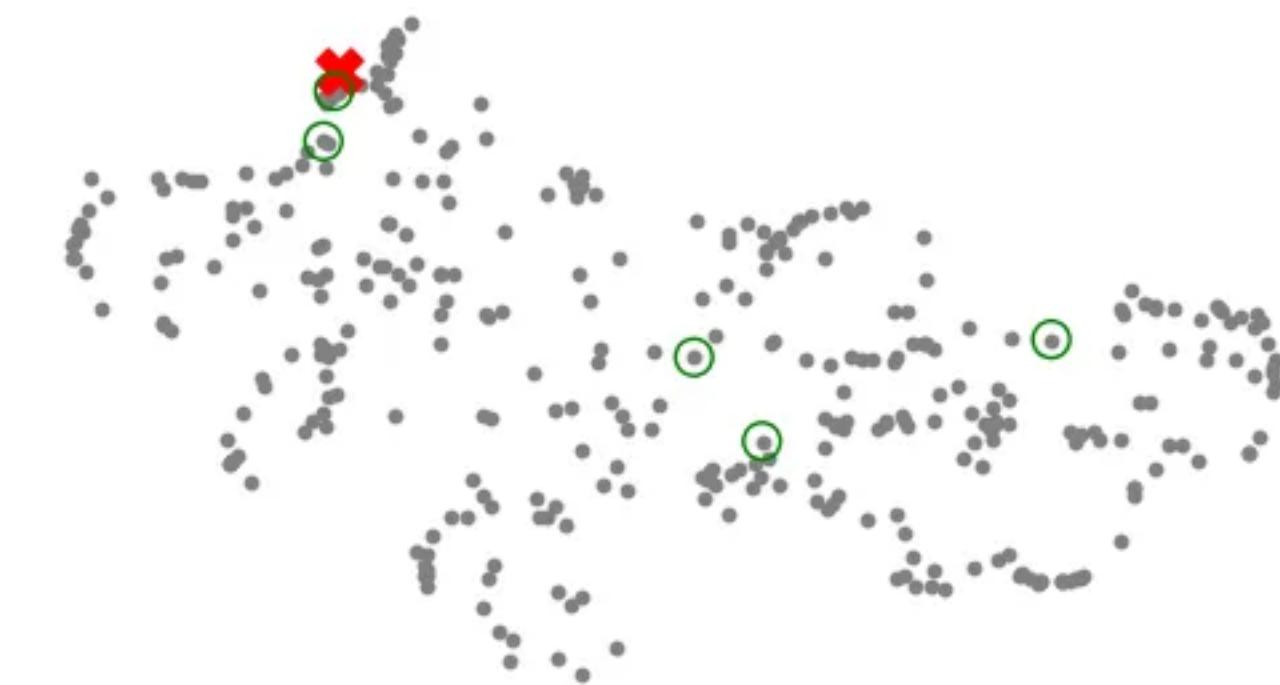
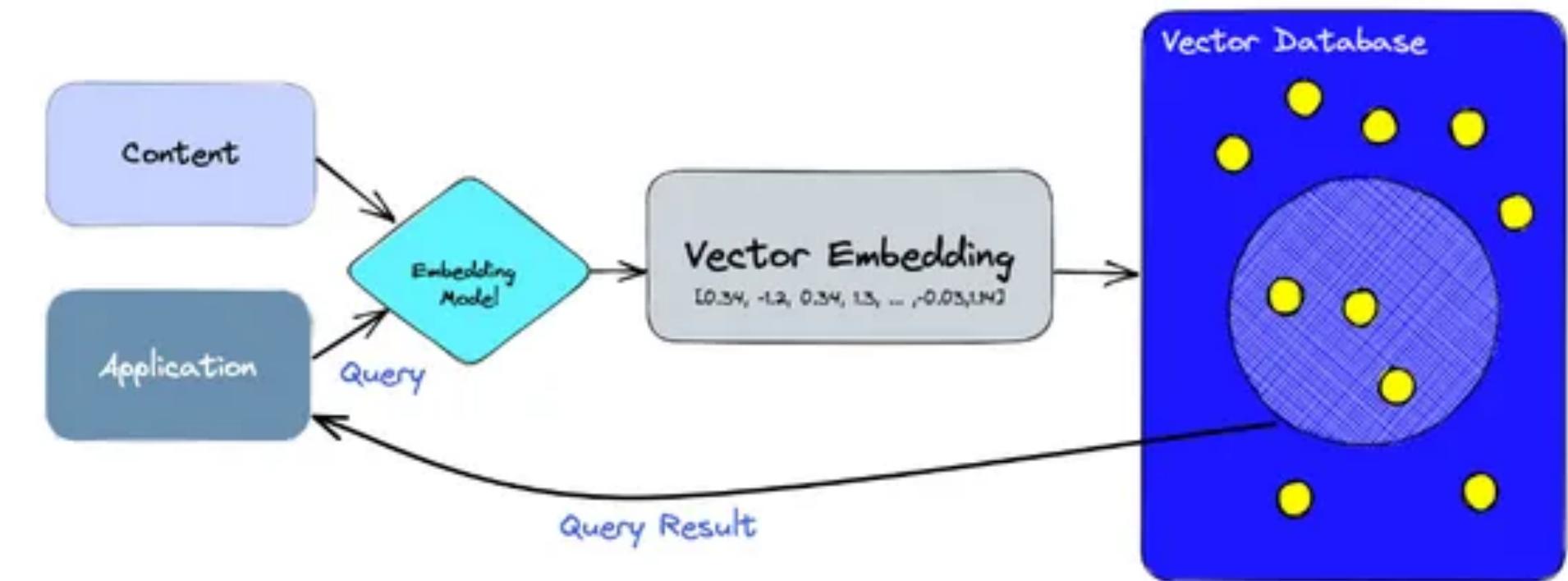
- 비용 효율 - “학습보단 데이터 적재가 싸지 않을까?”
- 확실한 출처 - 검색된 정보를 바탕으로 정확한 정보 생성
- 답변의 컨트롤 - 맥락 범위 안에서만 답변 시키기



TL;DR: Naive RAG Doesn't Work Well

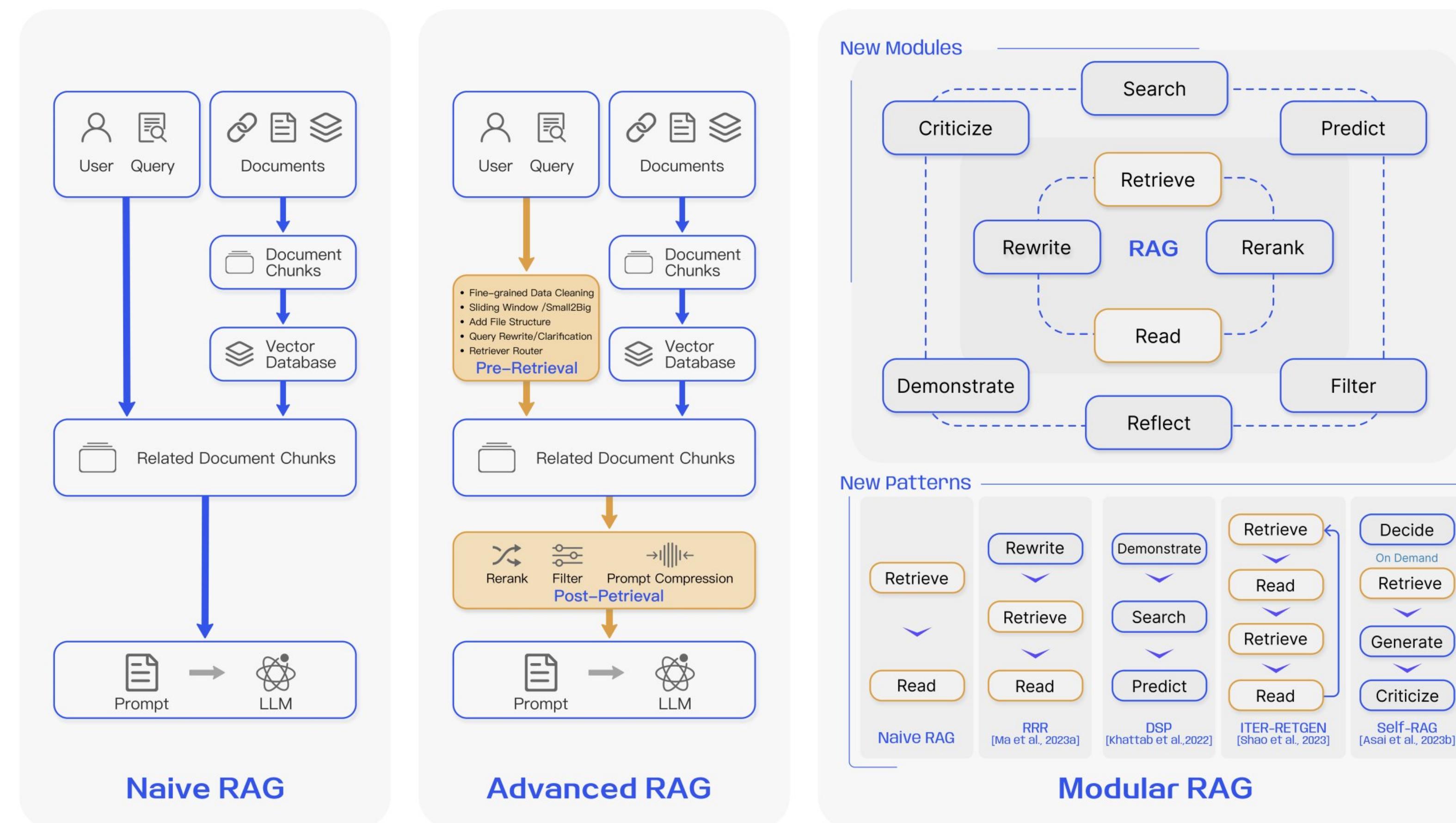
그냥 내 문서 때려박으면 알아서 해주는 것 아니었나요?

- **Indexing:** 경우에 따라 fine-tuning보다 비싼 작업
 - 어느 정도 사이즈로 데이터를 잘라야 할까?
 - 너무 크면 생성할때 넣어줄 컨텍스트가 많아 무거움
 - 너무 작으면 의미있는 정보를 담지 못해 retrieval이 작동 안함
 - 내 문서에서 긁어낸 텍스트를 바로 쓸 수 있을까?
 - 문서에 표, 그림 등이 섞여있는 경우, 파싱이 잘못 동작한 경우...
- **Retrieval의 결과물을 100% 신뢰할 수 없음**
 - 문서 자체가 아닌 “embedding”의 유사도를 비교하기 때문
 - 대개 의미상 비슷한 문서가 비슷한 embedding이 유사도가 높긴 하나, Query와의 유사도가 높다는 보장이 없음
 - 실제 가져온 정보 중 중요한 것이 무엇인지 모름



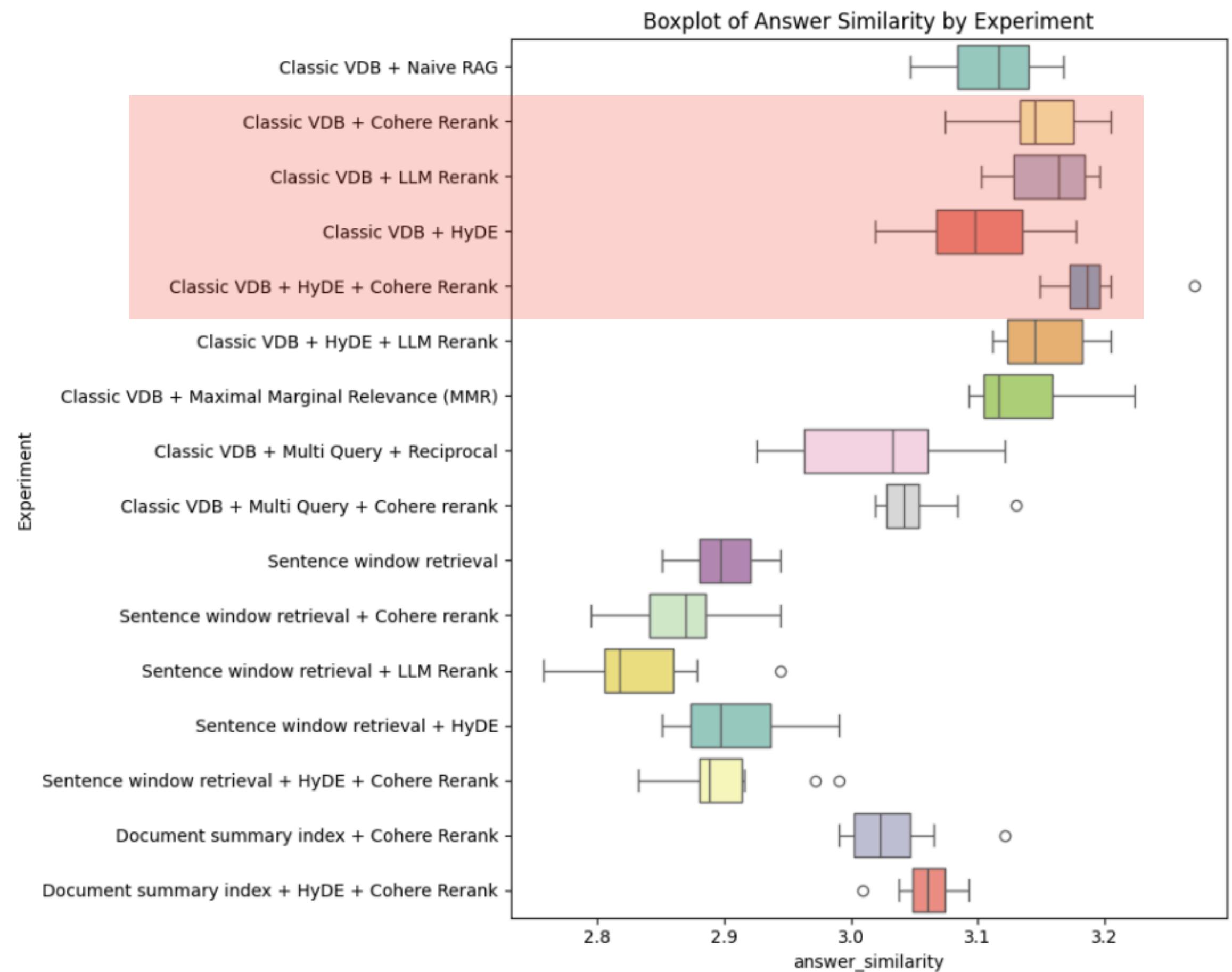
“Modular RAG”

이런저런 기법들을 잘 모아 패턴화한 RAG 방법들



Evaluating RAG Performance

- RAG의 성능을 측정하기 위한 도구들 (e.g. [Ragas](#))
 - RAG의 성능 지표
 - Faithfulness (얼마나 맞는 답을 생성했는지)
 - Context Recall (가져온 정보들 중 맞는 정보를 잘 썼는지)
 - Context Precision (엉뚱한 정보를 생성에 쓰지 않았는지)
 - Etc...
 - “LLM을 이용”해서 답변의 연관도, 정확도 등을 평가
 - → 느리고 100% 신뢰할 수는 없으나,
사람이 일일이 평가하는 것보다는 비교적 효율적

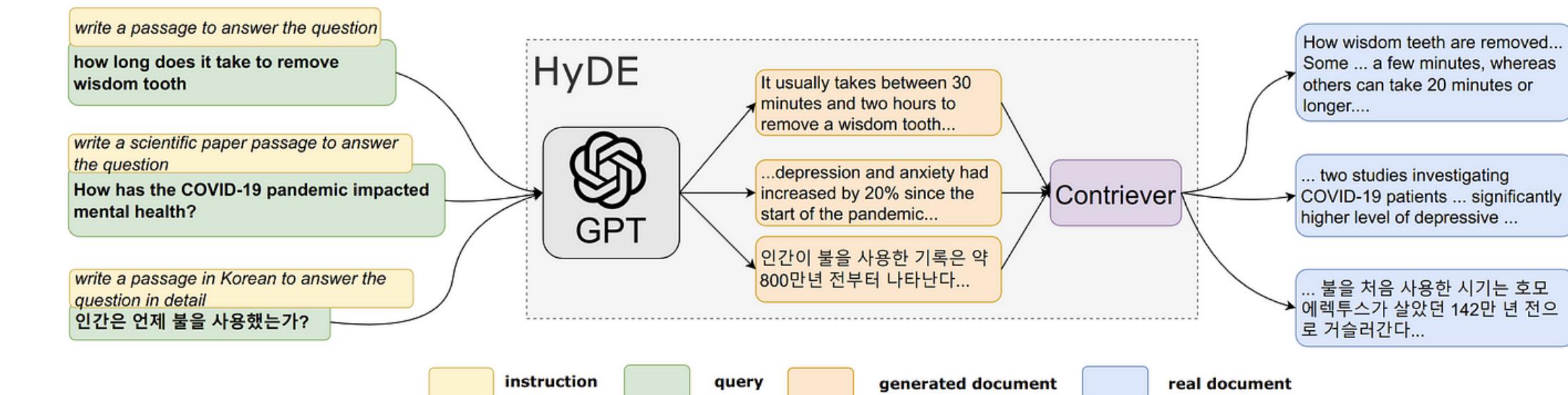


Improving RAG - Context Augmentation

질문과 어울리는 맥락을 찾는 좀 더 정교한 방법들

• “질문”을 증강하기

- 처음 주어진 query의 “가상의 답변”을 맥락 없이 생성
- “가상의 답변”的 임베딩이 실제 자료와 비슷할 가능성이 높음
→ Retrieval 성능 향상

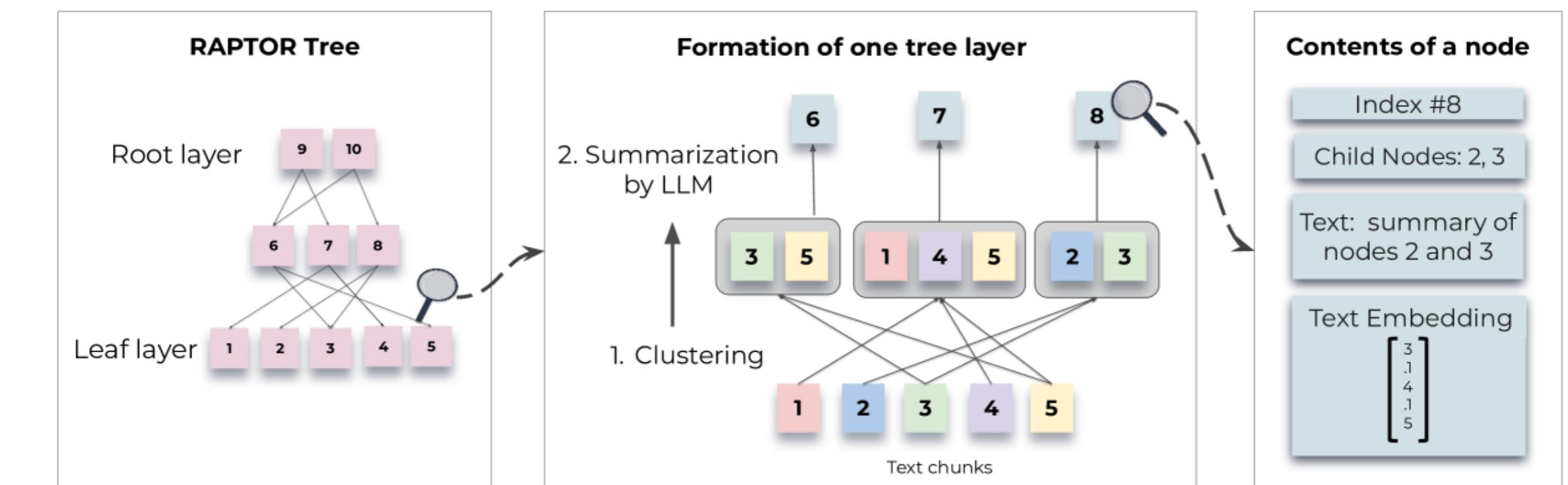


• “정보”를 증강하기

- 유사도가 높은 문서들을 서로 모은(clustering) 한 후 해당 문서의 “요약”을 새로운 문서로 DB에 추가
- 한정된 검색 기회에서 밀도 높은 정보를 획득할 가능성이 높음

• What else?

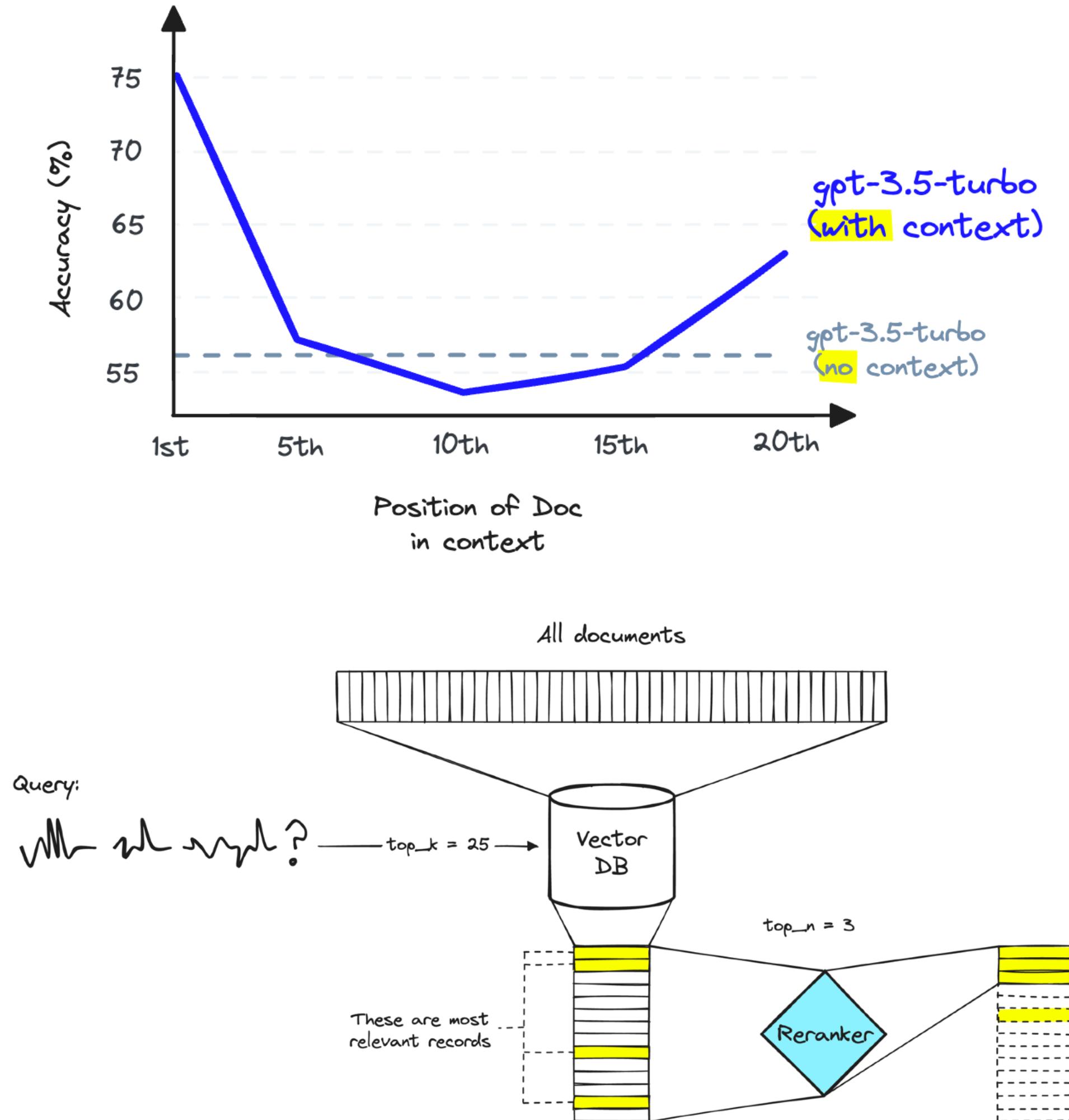
- 아직 개선의 여지가 많아 보이는 주제
- 데이터, 임베딩, retrieval 방법 등 여러 영역에서 새로운 기법들이 매일매일 쏟아지는 중



Improving RAG - Reranking

가져온 맥락 중 가장 쓸모있는 것을 정렬해서 LLM에게 알려주기

- 유용한 정보가 context의 상단에 있을수록 성능이 높아짐
 - Retrieve해온 모든 맥락을 밀어넣는 것보다는 중요한 것만 고르자
 - 어떻게 고를까? "Reranking"
- “Reranking”?
 - 실제 query와 관련도가 높은 순서대로 가져온 후보를 재정렬
 - 언어(docs)를 입력으로 받으나, 답변 생성 대신 여러 문서들 간의 유사도를 측정하는 모델을 사용 (Reranker model)
 - 사실 추천 시스템 등에서 많이 사용하던 기법들...

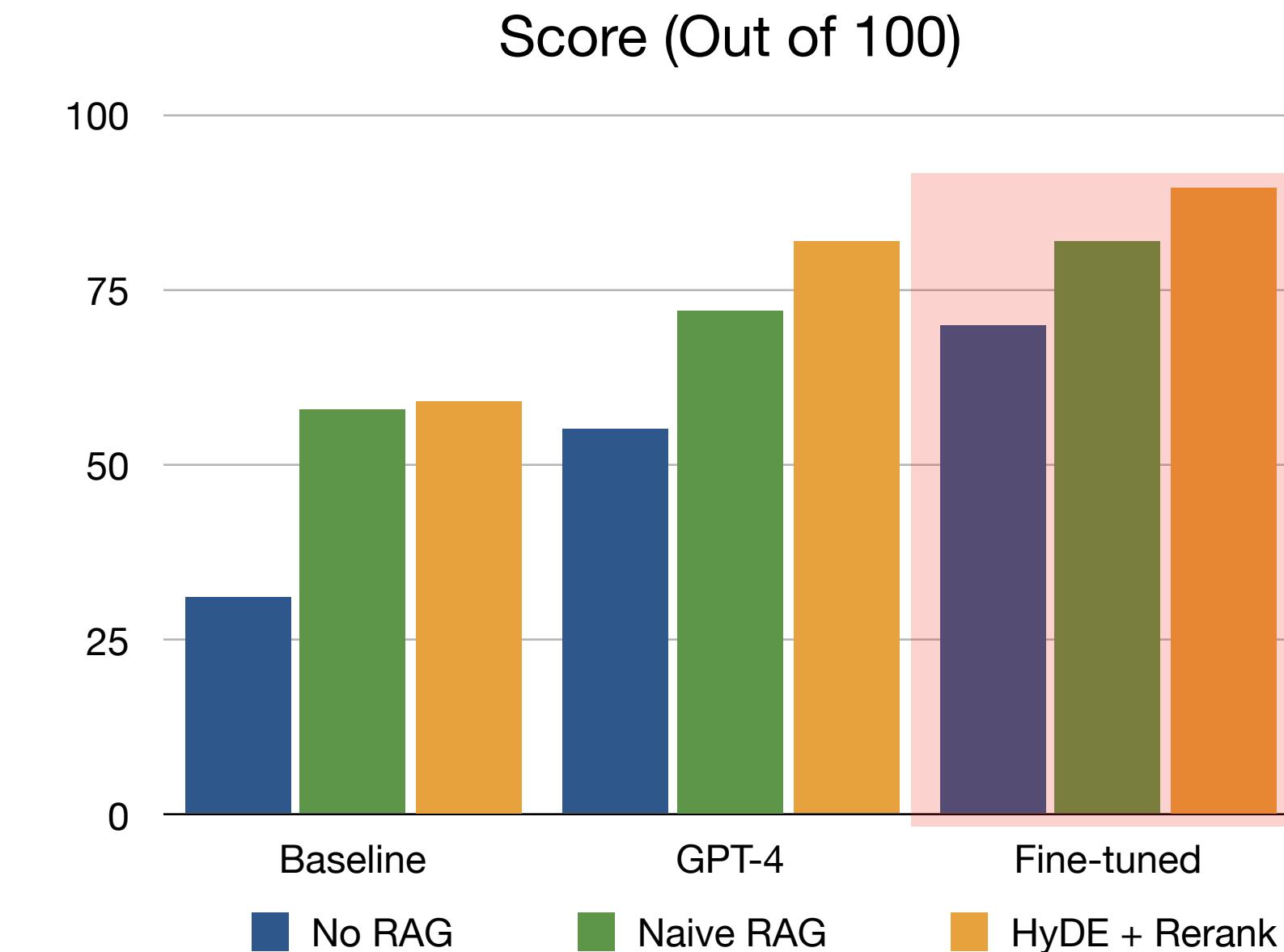


Evaluation Result

관련 사례 유사도 기반 top-k를 모델에 추가로 제공

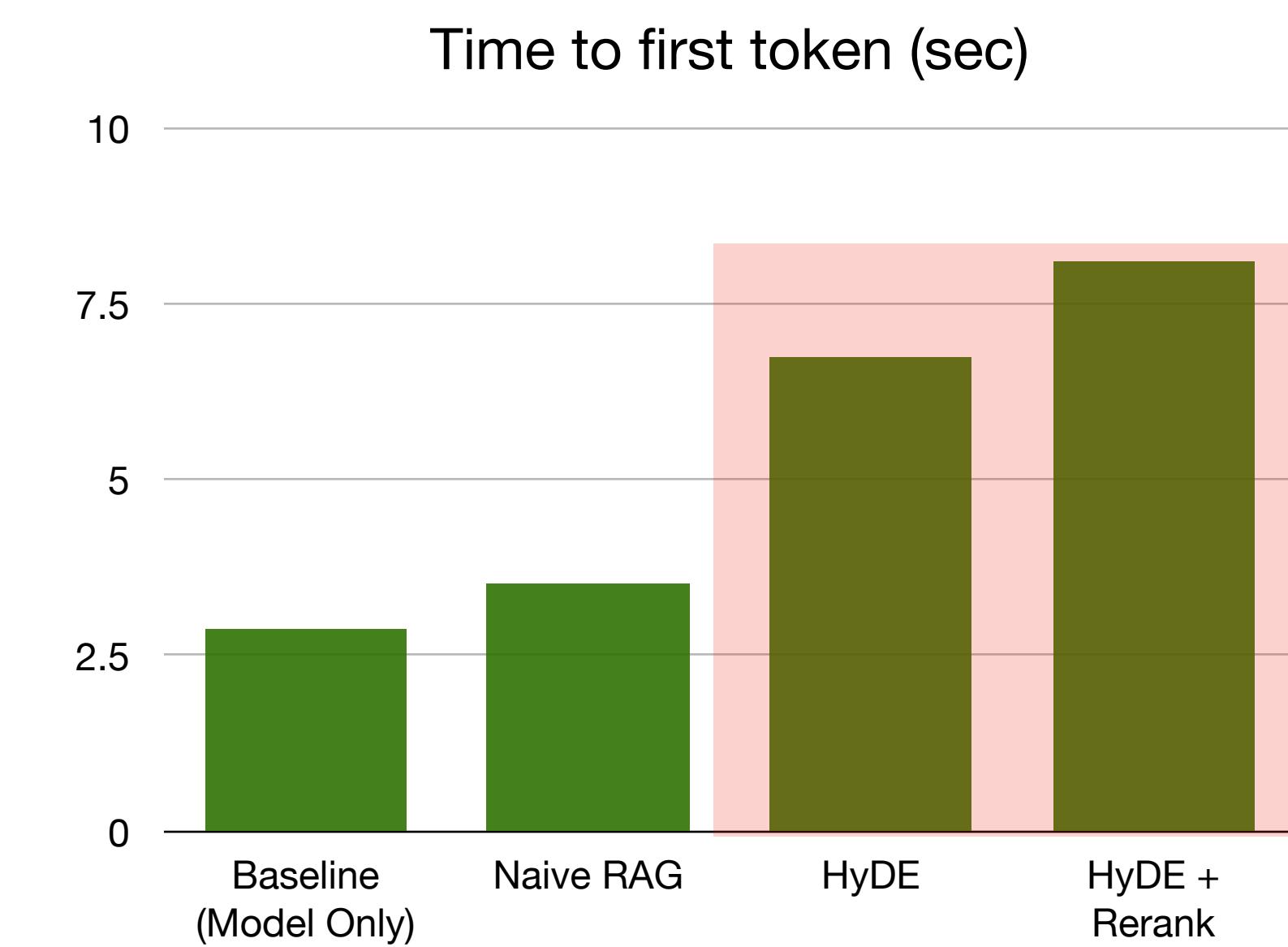
- 간단한 RAG만으로 답변의 수준이 크게 향상

- Document에 포함된 sample을 참조할 수 있게 됨 → 규격에 맞지 않는 답변이나, 완전히 엉뚱한 답변을 내놓는 경우가 감소
- 여전히 Fine-tuned > GPT >> Baseline



- Accuracy comes with cost

- HyDE와 Reranking은 특성상 LLM/Reranker model을 한 번 호출 → latency가 2배 가까이 증가
- 맥락을 포함하게 되면서 input token의 양도 많이 증가함 → 같은 질문이라도 비용 증가
- 그러나 비교적 까다로운 경우에 대해서도 답변의 정확도가 올라감 → 높은 신뢰도가 중요할 경우 검토해볼 만



- Caveats

- 답변이 비교적 많이 정형화된 task;
복잡한 답변이 요구되는 하는 경우는 결과가 달라질 수 있음

Key Takeaways: RAG

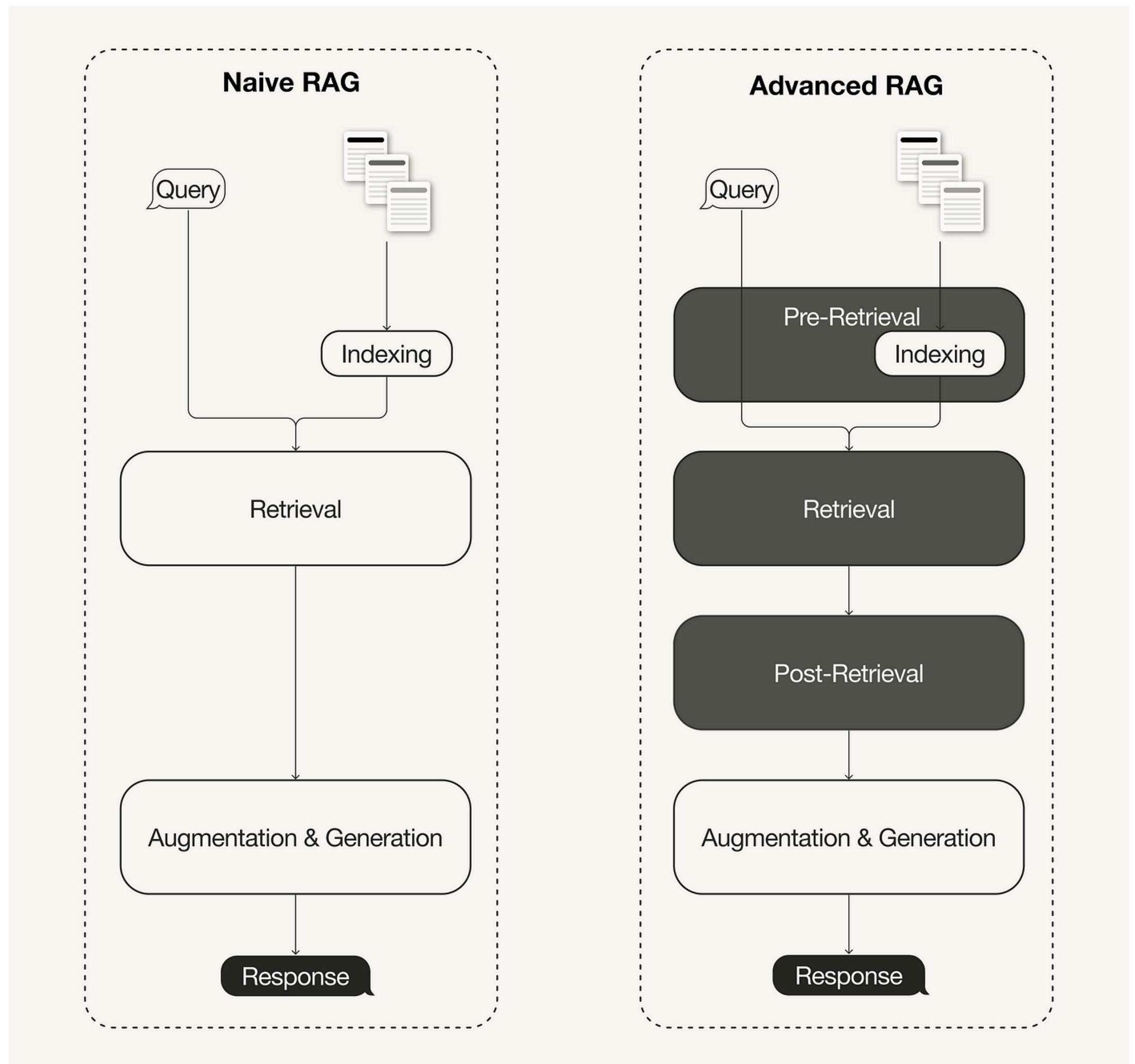
쉽게 할 수 있으나 잘 하려면 생각보다 할 게 많은 RAG

- **“Garbage in, garbage out”**

- RAG에서도 가장 중요한 것은 잘 정리된 데이터 - 데이터가 엉망이면 embedding이고 뭐고 의미가 없음
- 임베딩의 의미를 유지하는 적당한 사이즈의 chunk를 정하거나,
- 가져온 document의 metadata(id 등) 바탕으로 추가 정보를 가져올 수 있으면 좋음

- **Enhancing comes with cost**

- HyDE, Reranking 등은 전부 LLM / Reranker model을 사용
→ Time-to-first-token이 크게 증가
- 유저 경험을 해치지 않기 위한 방법을 고민하거나, latency를 감수할 수 있는 task 위주로 적용을 고려하기
- Reranking은 성능 향상의 체감이 커서 가치가 있는 듯



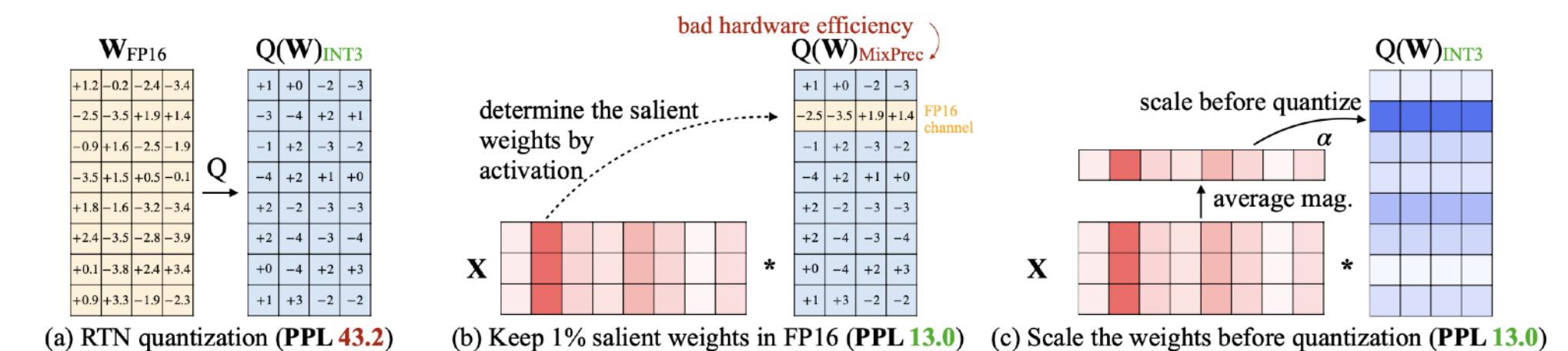
Reduce LLM's Burden: Quantize & Cache

Tip 1: Compress & Quantize LLM

Activation-aware Weight Quantization (AWQ)로 가벼운 LLM 만들기

- **What is AWQ?**

- 상대적인 중요도에 따라 중요한 weight만 남기고 나머지 weight를 4bit 압축해서 모델을 가볍게 만드는 기법

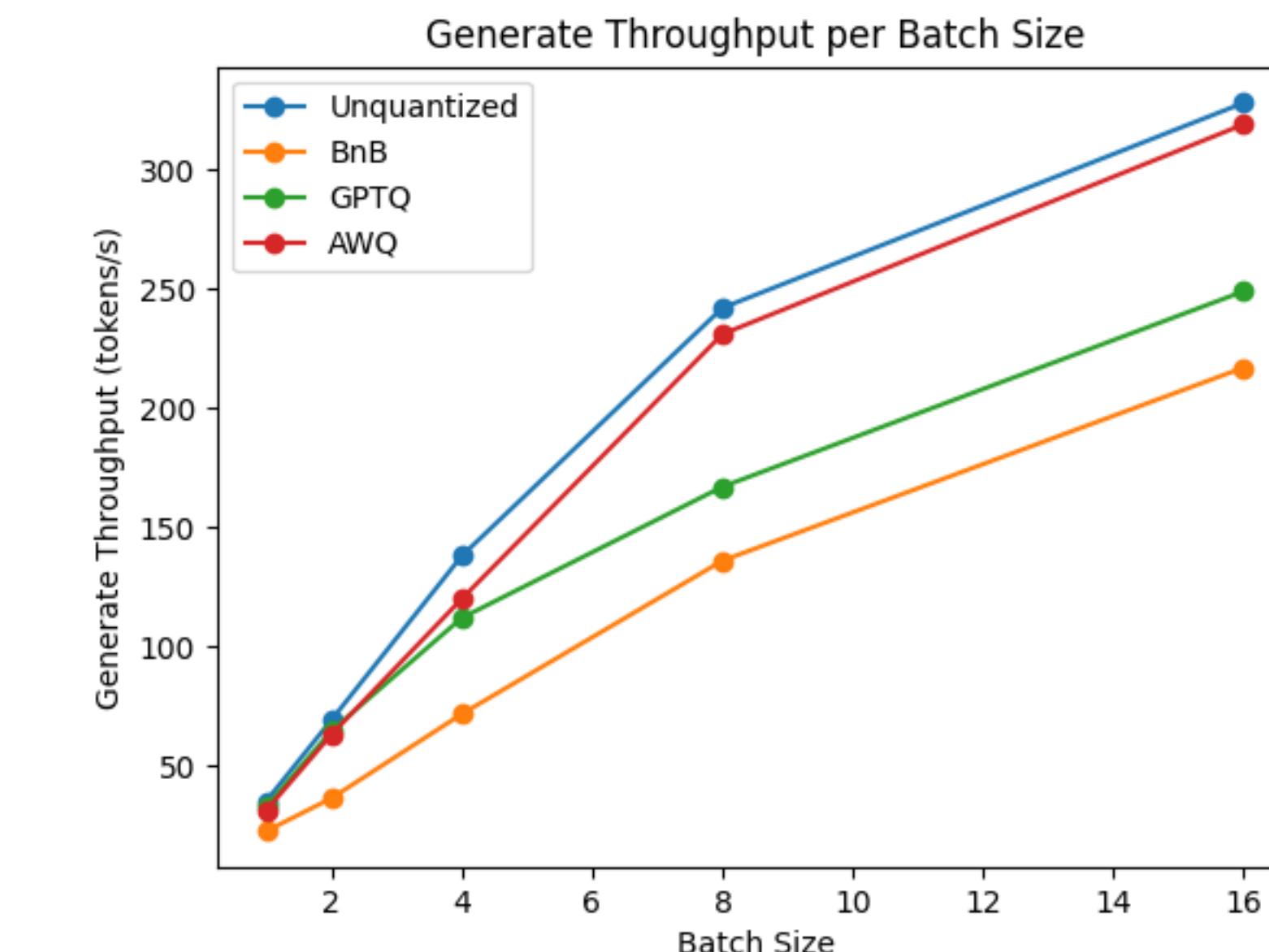


- **Why AWQ?**

- 모든 weight를 4bit 압축하는 것은 성능에서 손해를 크게 봄
- AWQ는 비교적 성능에서 손해를 덜 보면서...
 - 모델의 추론 속도를 크게 높이고
 - 메모리 사용량도 크게 줄여줌

- **Tip**

- 태스크 성능과 속도를 둘 다 잡기:
Fine-tuned adapter → 원본 모델 merge → AWQ quantize

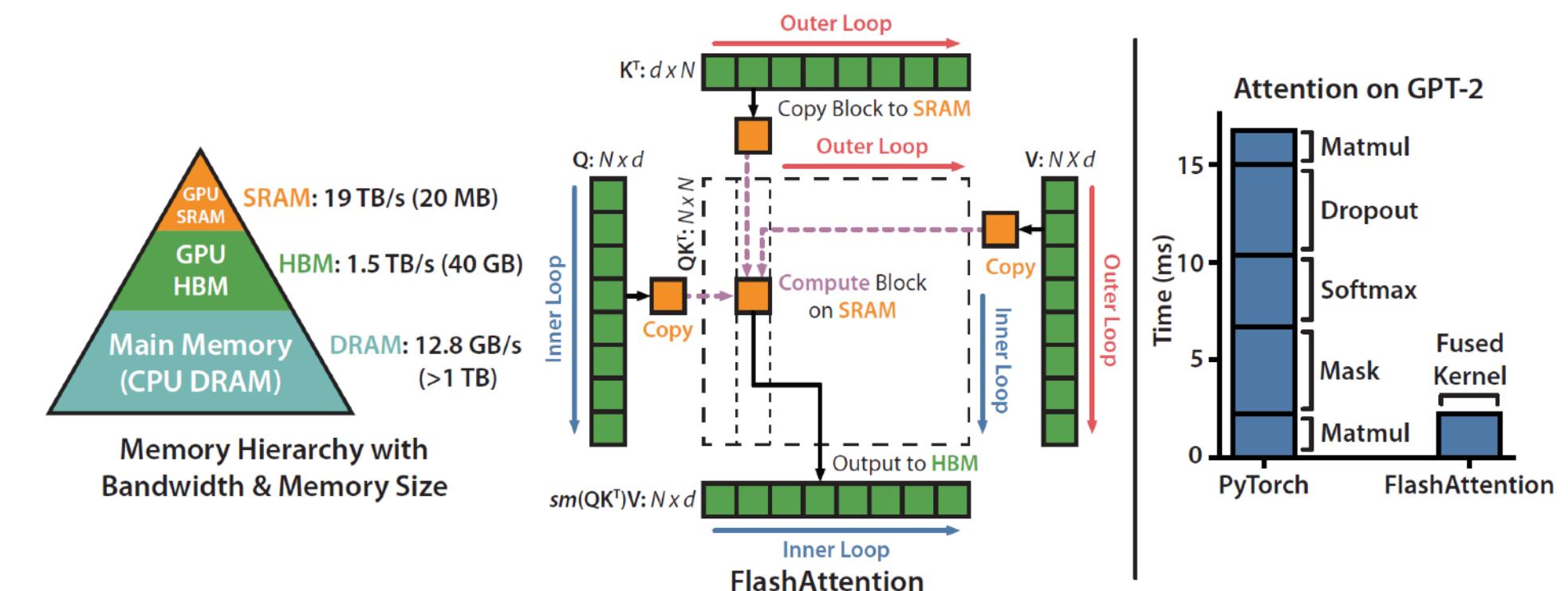


Tip 2: Optimize & Cache Attentions

FlashAttention과 PagedAttention(vLLM)으로 추론 성능 높이기

- 효율적인 연산과 캐싱으로 추론 성능 높이기

- FlashAttention2: Attention 연산을 최적화/병렬화
- PagedAttention(vLLM): KV cache를



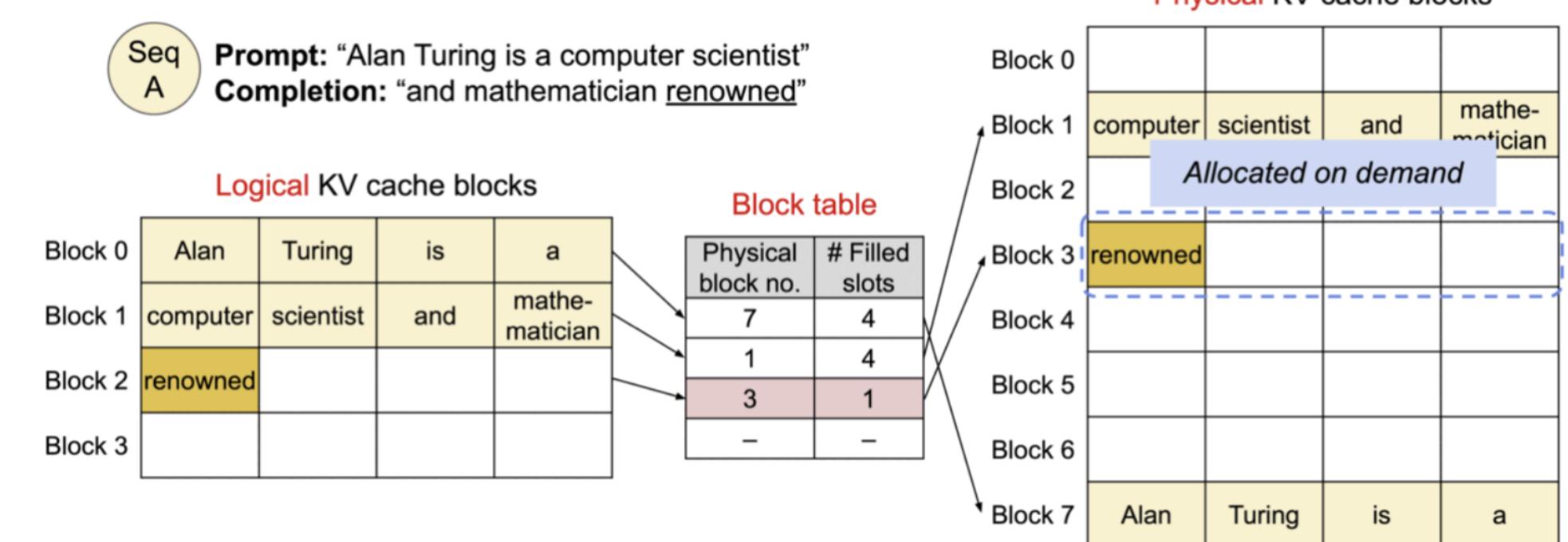
- 쉽게 가져와서 사용할 수 있음

- text-generation-inference 등 추론용 구현체들이 대부분 지원중

- 각각의 라이브러리에서도 서로의 구현체를 지원...

- FlashAttention in vLLM ([ref](#))
- PagedAttention in flash-attn ([ref](#))

4. Generated 3rd token. Allocate new block.

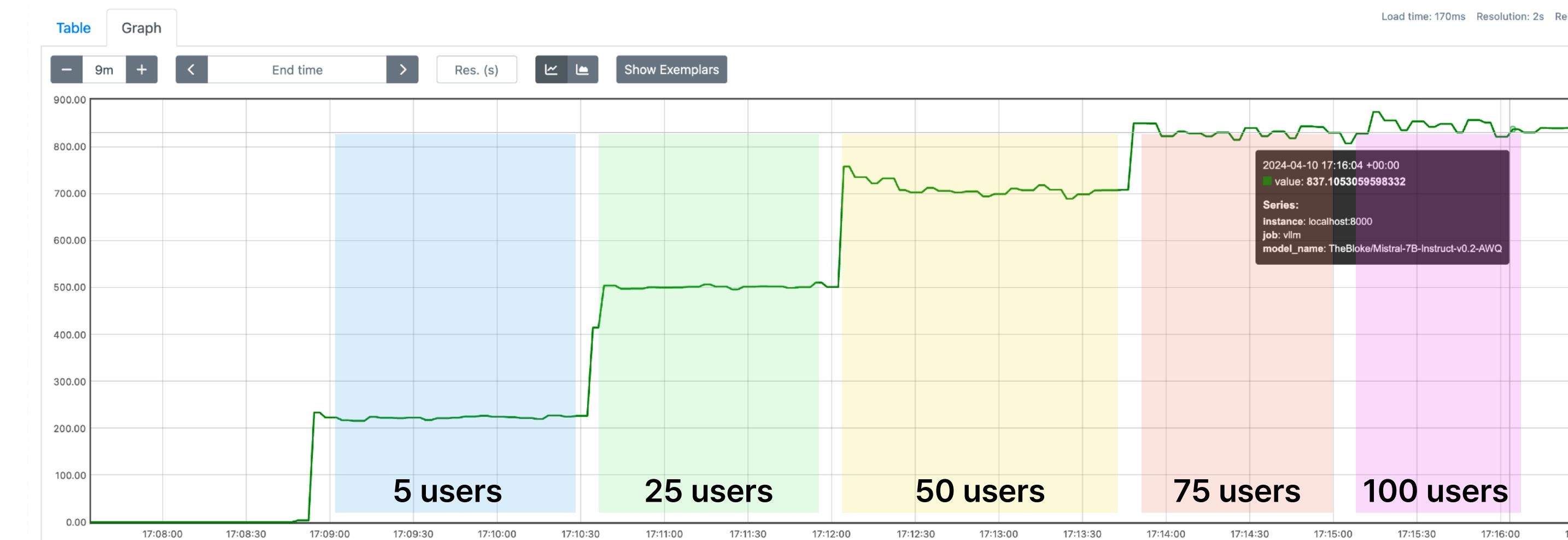
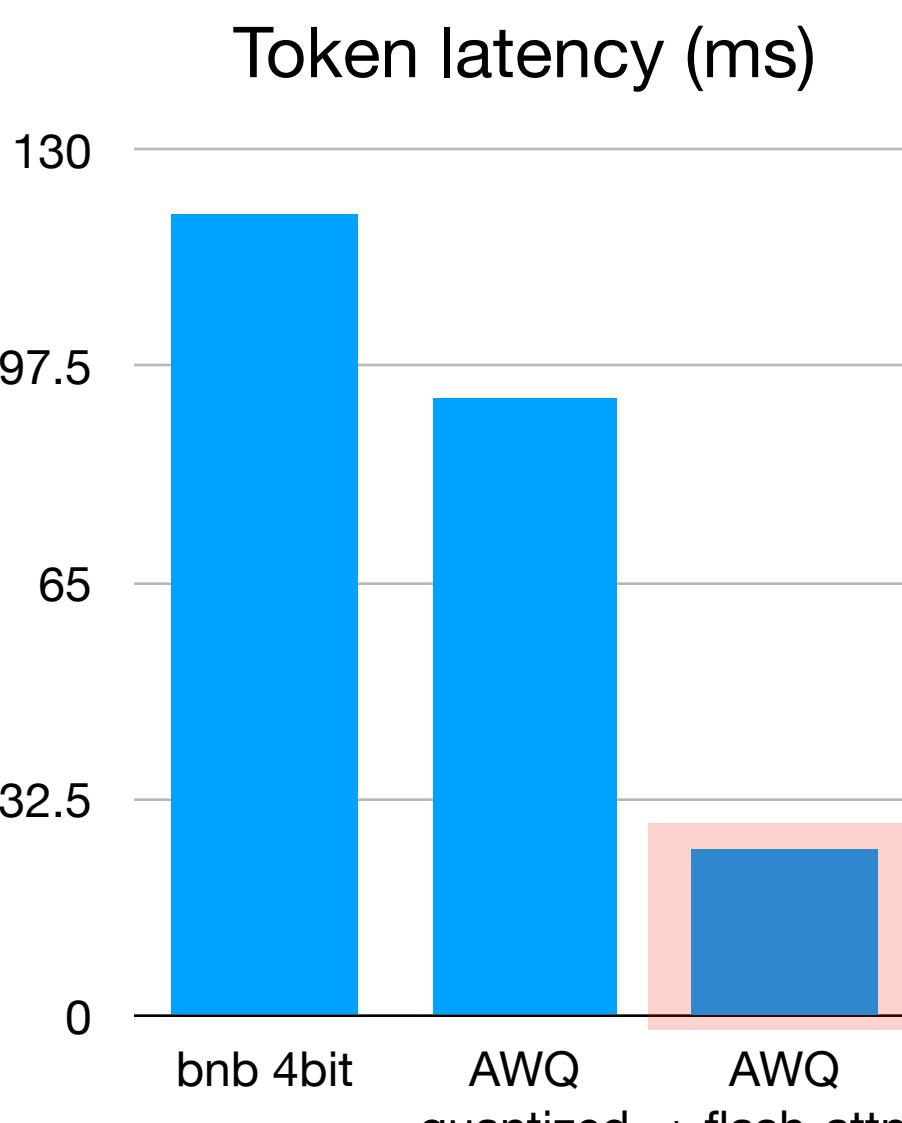


Example generation process for a request with PagedAttention.

Benchmark Result: Inference Performance

Calculating cost per million tokens with 7B models

- **5x performance with AWQ + FlashAttn**
 - 단순히 4bit quantize한 것 대비해서도 5배
 - 원본 모델(FP16)과 비교하면 훨씬 큰 차이
- **1 x Nvidia L4 GPU can handle 800 tokens/sec**
 - 7B 모델, 75명 동시 요청 기준 유저당 약 11.2 tokens / sec
 - ~ = \$0.3-\$0.4 / million tokens, 더 생성량을 높이는 것도 가능할듯
 - VS
 - GPT-4 = \$45 / 1M tokens
 - GPT-3.5-turbo = \$1 / 1M tokens
 - Claude Haiku = \$0.75 / 1M tokens
 - 운영 오버헤드 감안해도 적절한 가격에 서비스 가능



Summary & Lessons Learned

Private LLM: Before vs After



Baseline Model

GPT대신 쓸 이유가 없음

대답 잘 못함

무거워서 비싼 GPU 요구함

느리고 비쌈 (\$7-\$10 / 1M Tokens)

Private LLM: Before vs After



Baseline Model

GPT대신 쓸 이유가 없음

대답 잘 못함

무거워서 비싼 GPU 요구함

느리고 비쌈 (\$7-\$10 / 1M Tokens)



Fine-tune + RAG + Acceleration

내 태스크에선 GPT-4보다 고성능

만드는데 \$10-\$100 + Vector DB 운영 비용

Consumer GPU에서 사용 가능 (depends)

빠르고 저렴함 (\$0.3-\$1 / 1M Tokens)

You Got Your Own Model, Now What?

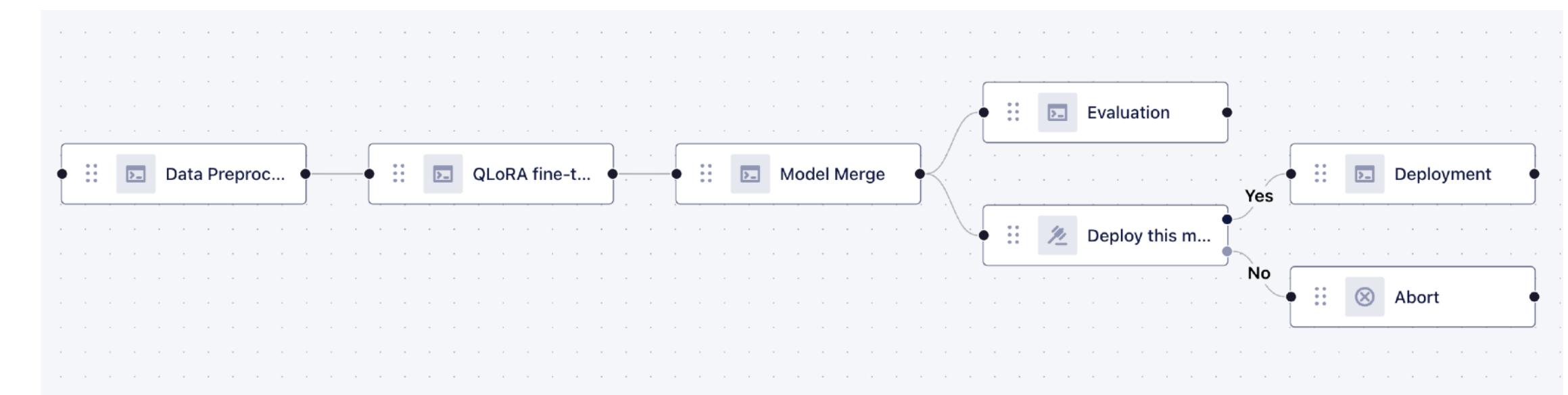
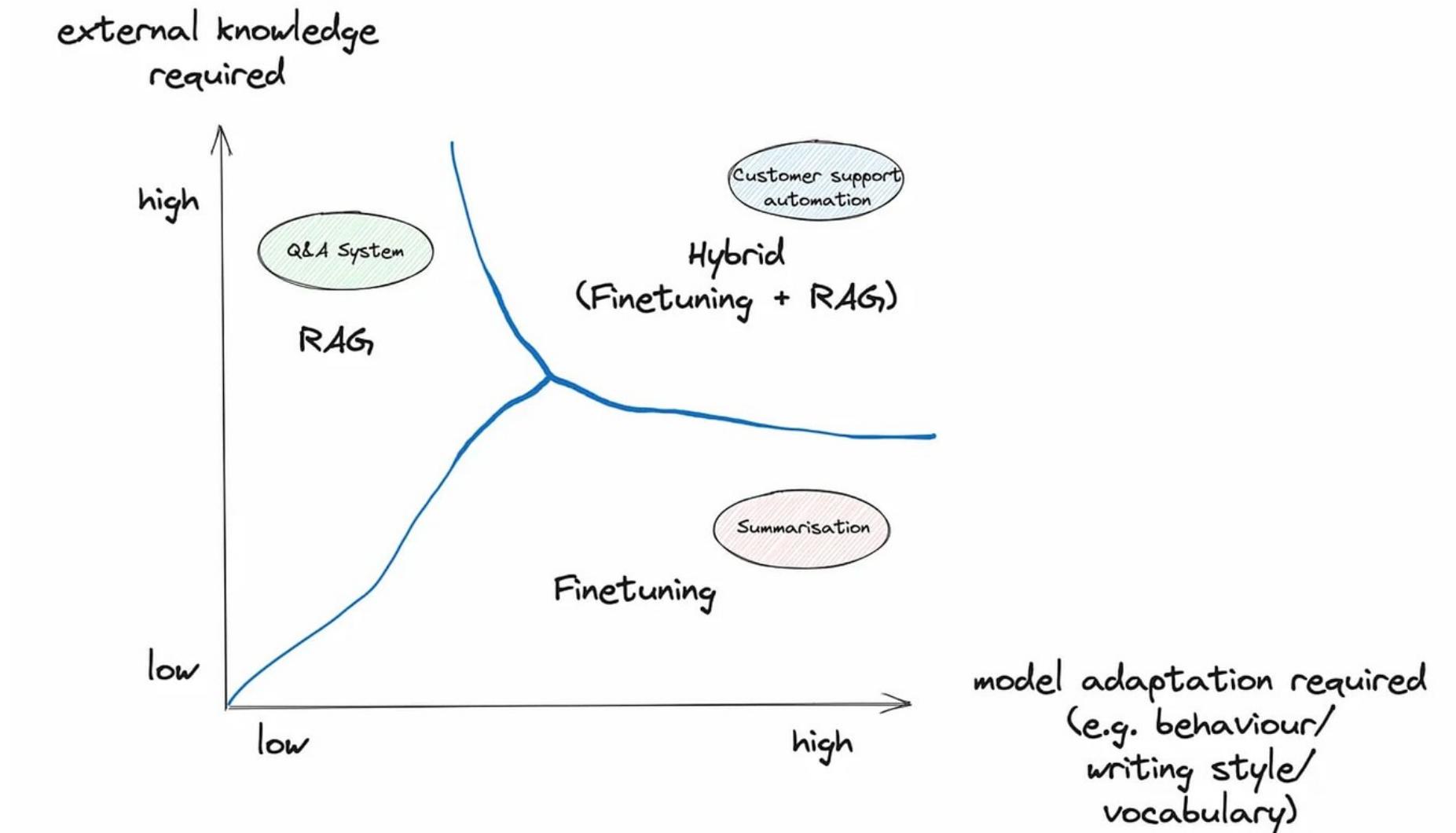
한번 만들어 본 건 좋은데... 실제로도 이 모든 과정을 다 해야 하나요?

- **Both fine-tuning and RAG is essential**

- 결국 원하는 수준으로 성능을 높이려면 둘 다 하는 것이 좋습니다
- RAG가 생각보다 싸지도, Fine-tuning이 생각보다 비싸지도 않습니다

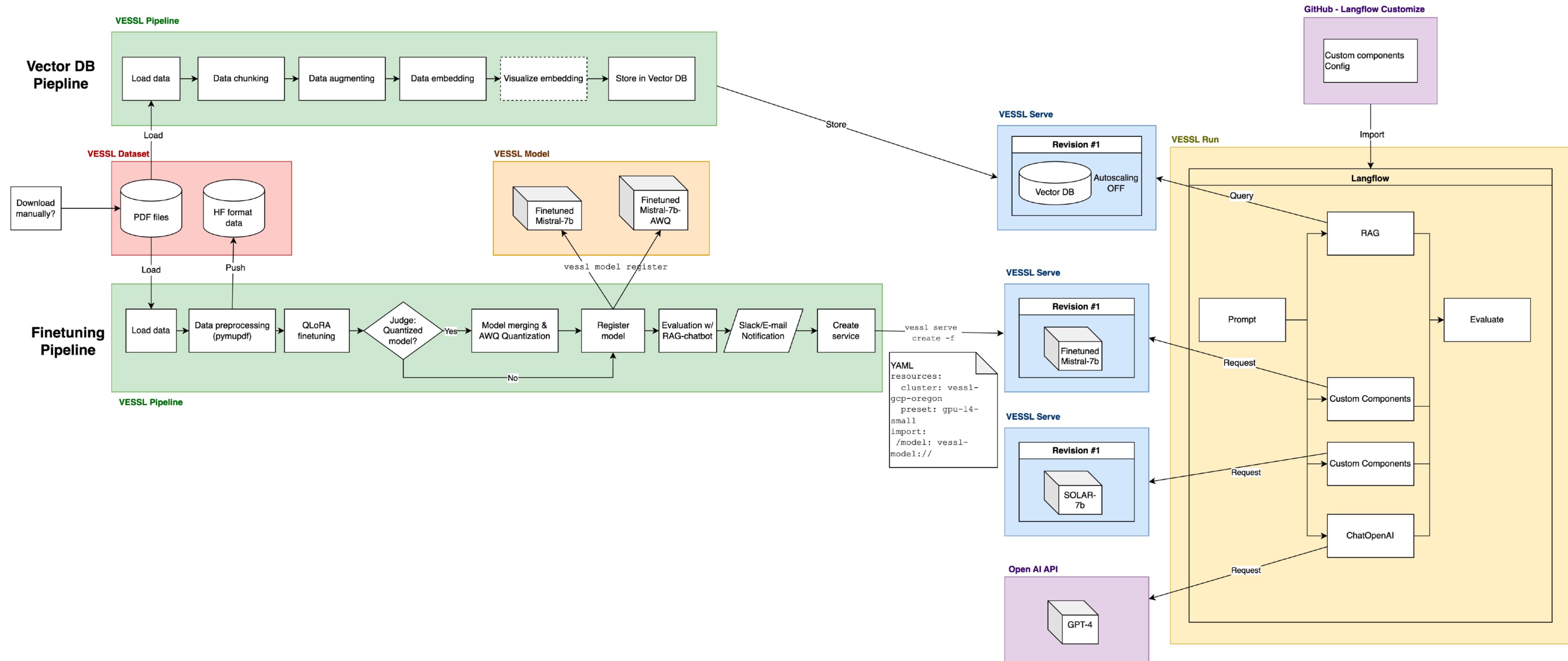
- **Procedures can get quite complex**

- → Needs reproducibility!
- 새로운 문서가 추가될때 인덱싱 / 파인튜닝 과정이 선택적으로 혹은 동시에 실행될 수 있게 파이프라인을 구성해두기
- 새 문서 → 파인튜닝된 모델과 새 인덱스가 자동으로 생성, 새 모델 기반으로 prompting / monitoring에 집중



Sample Architecture for Private LLM

Indexing Pipeline + Fine-tuning Pipeline + Service Endpoint





여전히 너무 어렵고 복잡해요



여전히 너무 어렵고 복잡해요



최신의 기술을 항상 따라잡고 있기도 쉽지 않아 보여요



여전히 너무 어렵고 복잡해요



최신의 기술을 항상 따라잡고 있기도 쉽지 않아 보여요



더 쉽고 편하게 만들 수 있는 방법은 없을까요?



여전히 너무 어렵고 복잡해요



VESSL AI

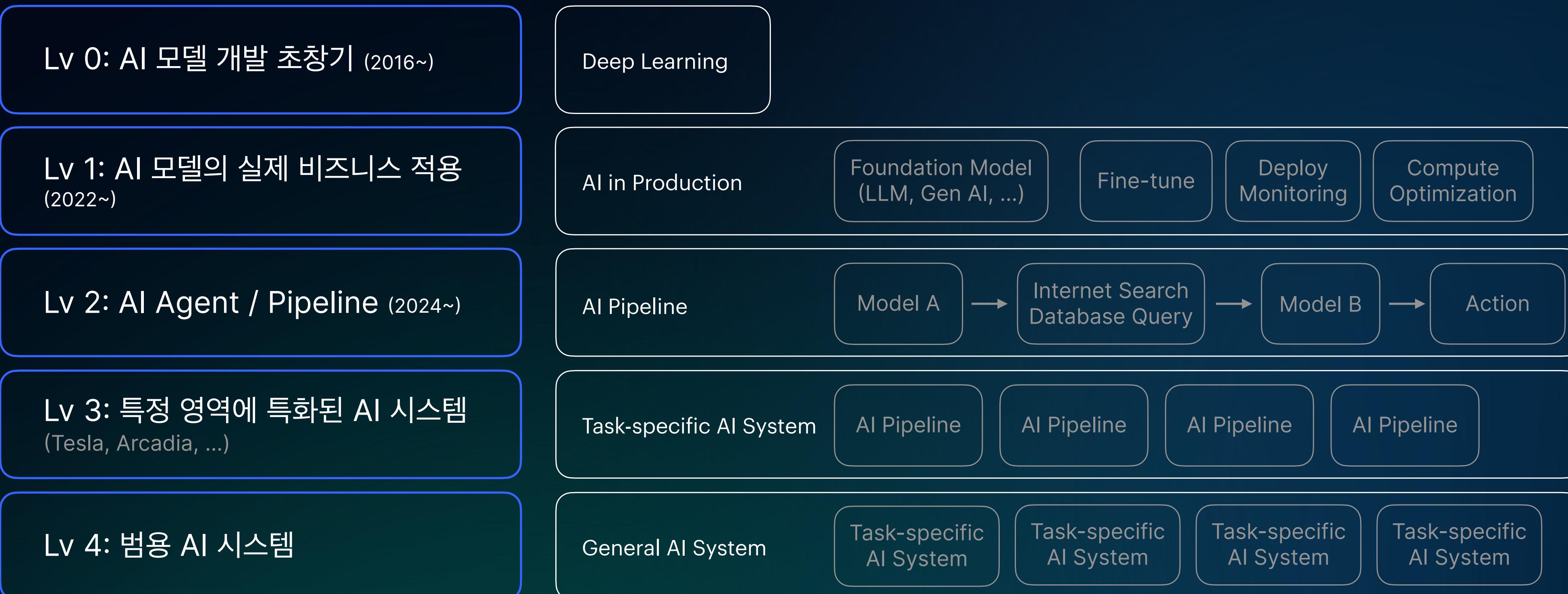


더 쉽고 편하게 만들 수 있는 방법은 없을까요?

Vision

General Intelligence System

범용 인공지능은, 하나의 큰 모델이 아닌 수백개의 AI agent를 통해 만들어 집니다.



Vision — 필요 요구 조건 1

수백개의 AI 모델이 자동으로 개발되고 운영되는 인프라

The complete software stack for AI

The screenshot shows the VLLM Hub interface, which is a web-based platform for managing and deploying AI models. The interface includes a sidebar with navigation icons (VE, Q, C, D, A, B, E, F, G) and a top bar with 'Hub' (selected), '30.00 credits', 'Explore models', and a user profile icon.

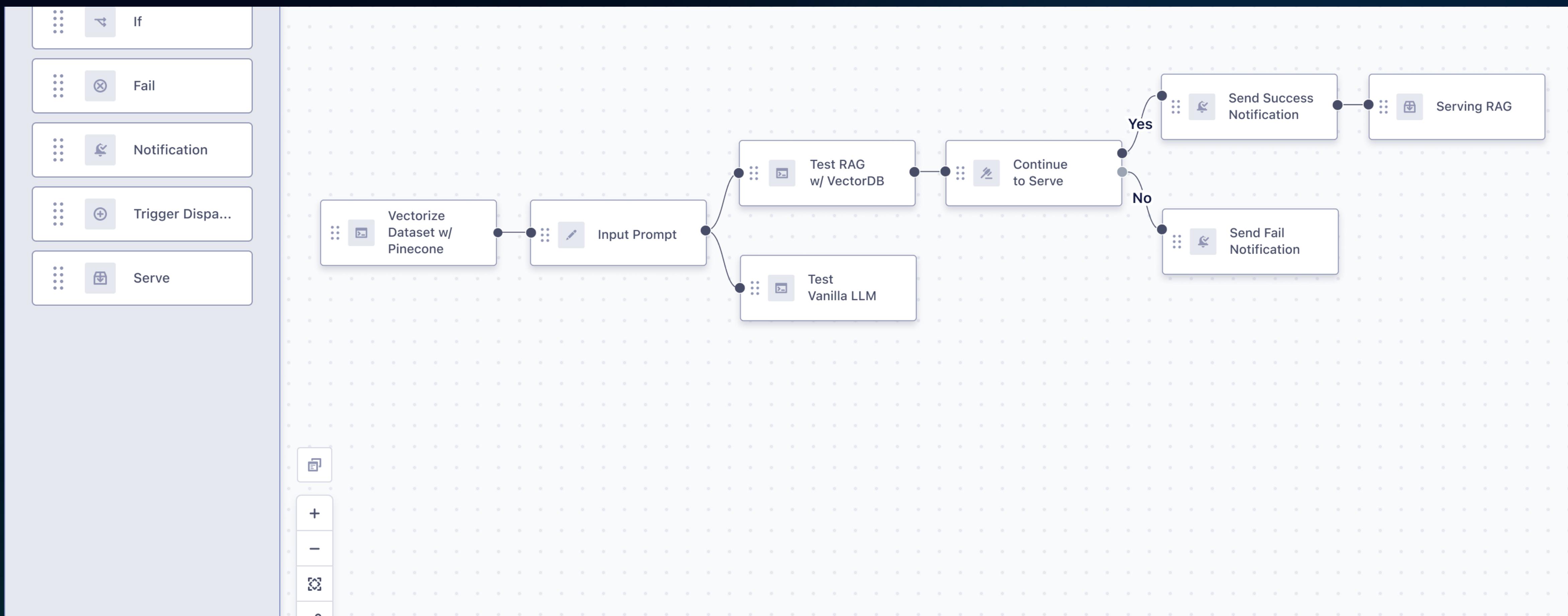
The main area displays six AI models:

- Llama2-Finetuning**: Finetuning Llama 2 with a small code instruction dataset in Alpaca format. Features: text generation, llama2, finetuning, quickstart. Image: A llama wearing sunglasses.
- Mixtral-8x7b.chatbot**: Mistral-7B: An open-source LLM which achieves both efficiency and high performance. Features: inference, LLM, text generation, quickstart. Image: A dark background with the text "Mixtral of experts" and a stylized "H" logo.
- Stable Diffusion WebUI**: Run stable diffusion with multiple checkpoints on web app. Features: inference, stable diffusion, text to image, +1. Image: An astronaut riding a horse.
- Whisper-v3**: A model for automatic speech recognition and speech translation with large dataset of diverse ... Features: Input (Form, Node.js, Python, Elixir, HTTP). Image: A smartphone displaying an audio waveform.
- SSD-1B**: Faster, smaller SDXL variant excelling in high-quality text-to-image generation with diverse dataset ... Features: Input (Form, Node.js, Python, Elixir, HTTP). Image: A grid of generated images including a forest scene, a golden robot, a man in a suit, a woman's face, an owl, a wizard, a person in a space suit, and a dragon.
- vLLM**: Serve Mistral-7B-Instruct-v0.2 with vLLM and VESSL Run.

Vision — 필요 요구 조건 2

여러 AI 모델과 데이터를 통한 의사결정 프로세스

The complete software stack for AI

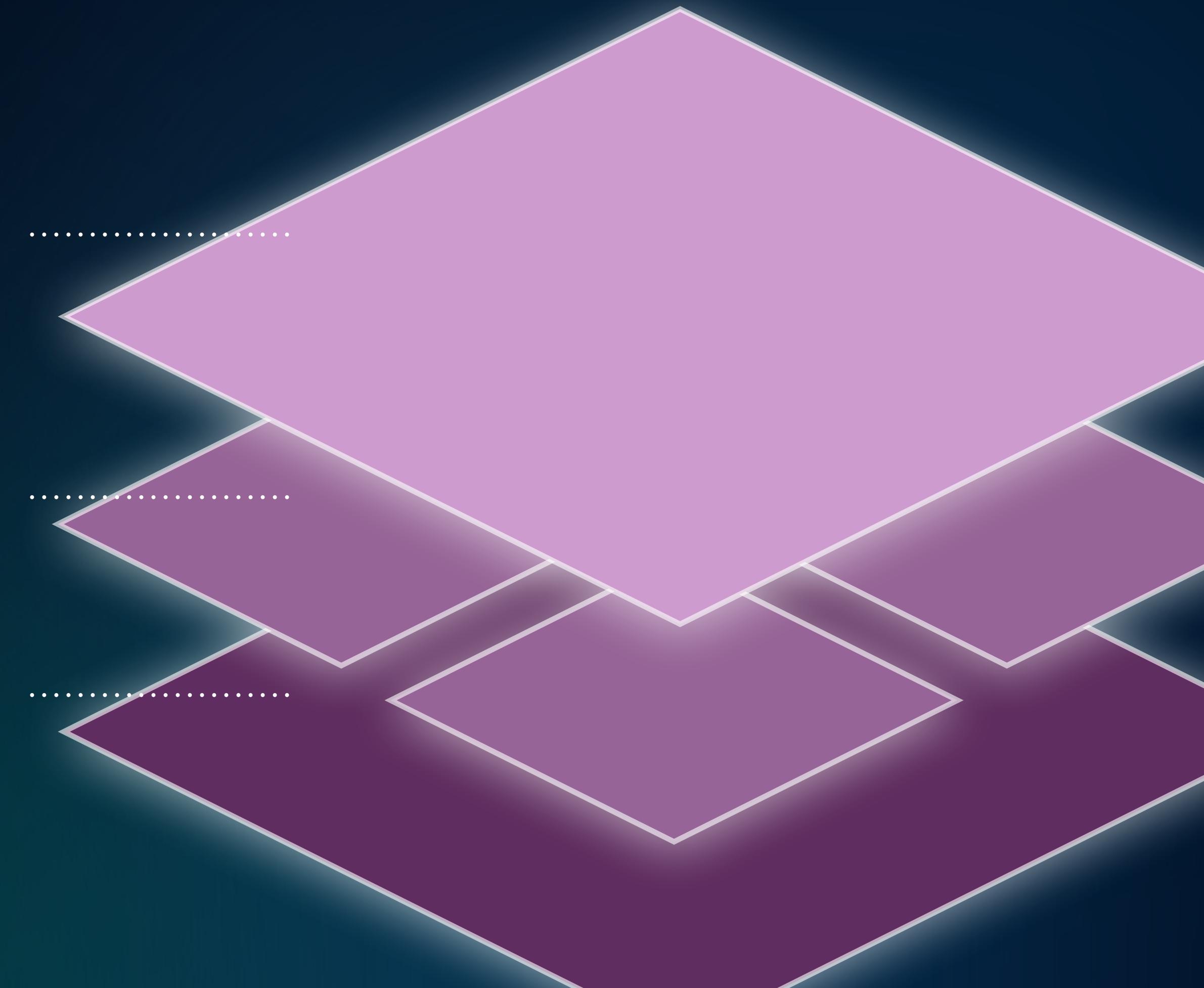


VESSL AI 의 해결 방식

VESSL AI — AI 를 위한 올인원 MLOps Platform

The complete software stack for AI

- ❖ VESSL Pipelines — 모델 학습 및 운영 자동화
- ❖ VESSL Hub — 한 번의 클릭으로 최신 AI 모델 실행
- ❖ VESSL Run — 인프라 고민 없이 모델 학습부터 배포까지
- ❖ VESSL Serve — 간단한 명령어 한 줄로 모델을 빠르게 배포
- ❖ VESSL Clusters — 멀티 클라우드 활용, GPU 사용 최적화 등
- ❖ VESSL Artifacts — 데이터셋, 모델 등의 머신러닝 자원 관리



단 한번의 클릭으로 최신 AI 모델 이용

클라우드와 GitHub 저장소, 그리고 데이터셋 설정만 해두고, Llama 2와 Stable Diffusion과 같은 최신 오픈소스 LLM과 생성형 AI 모델을 단 몇 초안에 빠르게 학습하고 파인튜닝 할 수 있습니다.

The screenshot shows the VESSL Hub interface with a sidebar on the left containing icons for Home, VE, Search, Collection, Model, Data, and Help. The main area displays several AI model cards:

- Llama2-Finetuning**: Finetuning Llama 2 with a small code instruction dataset in Alpaca format. Features: text generation, llama2, finetuning, quickstart. Preview image: A llama wearing sunglasses.
- Mixtral-8x7b.chatbot**: Mistral-7B: An open-source LLM which achieves both efficiency and high performance. Features: inference, LLM, text generation, quickstart. Preview image: A dark background with the text "Mixtral of experts" and a stylized "H".
- Stable Diffusion WebUI**: Run stable diffusion with multiple checkpoints on web app. Features: inference, stable diffusion, text to image, +1. Preview image: An astronaut riding a horse.
- Whisper-v3**: A model for automatic speech recognition and speech translation with large dataset of diverse ... Features: text generation, inference, LLM, quickstart. Preview image: A yellow speech bubble icon.
- SSD-1B**: Faster, smaller SDXL variant excelling in high-quality text-to-image generation with diverse dataset ... Features: inference, LLM, text generation, quickstart. Preview image: A collage of generated images.
- vLLM**: Serve Mistral-7B-Instruct-v0.2 with vLLM and VESSL Run. Features: inference, LLM, text generation, quickstart. Preview image: A collage of generated images.

VESSL Run

AI 모델 학습 · 배포를 단 한 줄의 명령어로

The complete software stack for AI

VESSL Run은 모델 학습과 파인튜닝을 단 몇 초 안에 실행할 수 있는 통합 인터페이스를 제공합니다. 인프라에 대한 고민 없이, 오류 대응에 대한 걱정 없이 학습과 배포를 빠르게 진행해보세요.

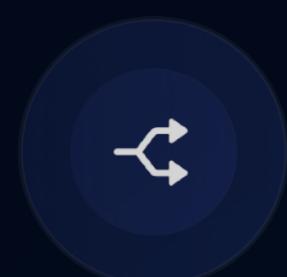
```
vessel run create -f llama2_c.yaml
```

The screenshot shows the VESSL AI / Projects interface. At the top, there is a navigation bar with links: Home, Pipelines, Projects (which is highlighted), Datasets, Workspaces, Clusters, Models, and Settings. Below the navigation bar, the main content area has a title "1. Select instance" and a sub-instruction "Configure a compute instance for your workload." It includes a "Learn more" link. There are four options for "Resource type": VESSL Cloud (selected), AWS, GCP, and On-premise. To the right of the instance selection, there is a section titled "2. Mount volumes" with the instruction "Connect your codebase and datasets". It shows a "Select Repository" button and a file tree on the left containing "detectron2", "facebookresearch", "coco_dataset", "training", "label", and "output". At the bottom, there are sections for "3. Configure arguments" (with a "Hyperparameters" sub-section) and "4. Deploy workload" (with a command-line interface showing "python main.py --dist").

VESSL Serve

AI 모델 운영에서의 확장성 · 안정성

The complete software stack for AI



다중 모델, 다중 포트 배포

다수의 포트를 생성하여 여러 모델을 운영해보세요.



서비스 엔드포인트

비용 최적화를 위해 Serverless 엔드포인트를 생성하세요.



오토스케일링

사용량 증가에 따라 자원을 자동으로 조절하여 최적의 성능을 유지하세요.



모델 모니터링

GPU 사용량과 모델 성능에 대하여 지속적으로 모니터링하세요.

Serve > RAG Chatbot with Mixtral-7B

Endpoint: model-service-gateway-bef6jufoo3mt.seoul.aws-cluster.vessl.ai [Copy](#)

Requests

```

1 {
2   "id": 1782920123719,
3   "status": "completed",
4   "message": "Retrieval Augmented Generation (RAG) is natural al, delaytime: 80823, executionTime: 49, status: \"COMPLETED\""
5 }
6

```

Response

```

1 {
2   "id": 1782920123719,
3   "status": "completed",
4   "message": "Retrieval Augmented Generation (RAG) is natural al, delaytime: 80823, executionTime: 49, status: \"COMPLETED\""
5 }
6

```

Request ID	Prompt	Status	Start time	Latency	Tokens	Actions
1782920123721	Write a dialogue between a tree and a sq...	In queue	2024-01-13 20:31:38	-	423	Refresh Edit
1782929123720	Describe the sensation of flying on a ma...	Completed	2024-01-13 20:35:32	3.21s	255	Refresh Edit
1782920123719	Create a poem inspired by the sound of ...	Completed	2024-01-13 20:42:24	80.82s	804	Refresh Edit

Metrics

Token Throughput

Time	Prompt Token/Sec	Generation Token/Sec
00:15:45	~250	~150
00:16:15	~350	~220
00:16:30	~450	~280
00:16:45	~400	~320
00:17:00	~480	~350
00:17:15	~420	~330
00:18:00	~380	~320

Time to First Token Latency

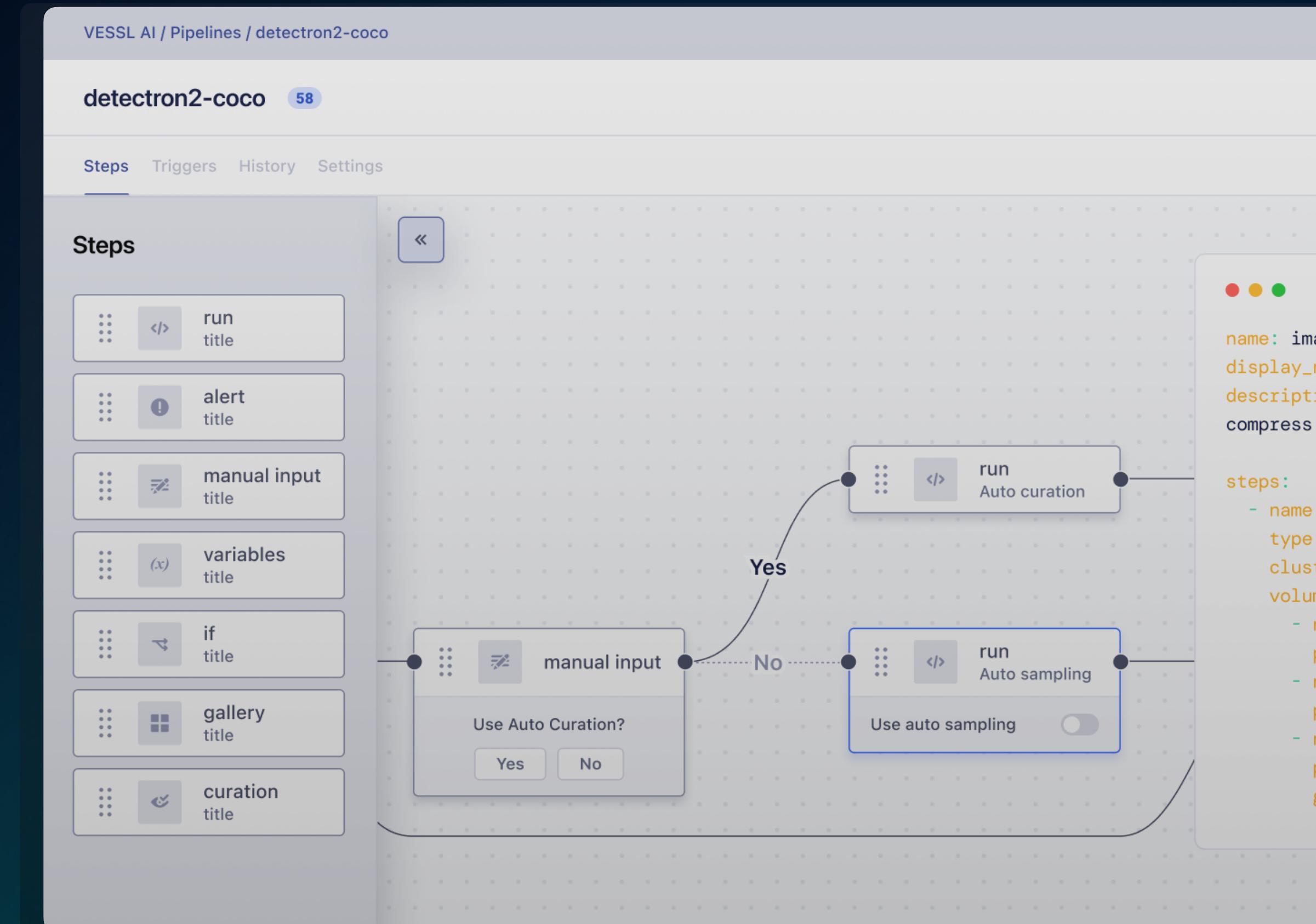
Time	P99	P95	P90	Average
00:15:45	~350ms	~250ms	~200ms	~100ms
00:16:15	~550ms	~350ms	~250ms	~150ms
00:16:30	~480ms	~320ms	~220ms	~120ms
00:16:45	~450ms	~300ms	~200ms	~100ms
00:17:00	~520ms	~350ms	~250ms	~150ms
00:17:15	~480ms	~320ms	~220ms	~120ms
00:18:00	~450ms	~300ms	~200ms	~100ms



VESSL Pipeline

AI 모델 개발 · 운영 자동화

데이터 수집부터 모델 배포, 모니터링에 이르는
AI 개발의 모든 단계를 파이프라인을 통해 자동화해보세요.
반복되는 모델 개발 과정을 드래그 앤 드롭만으로, 지속적인
학습 · 배포를 위한 자동화 파이프라인을 구축할 수 있습니다.

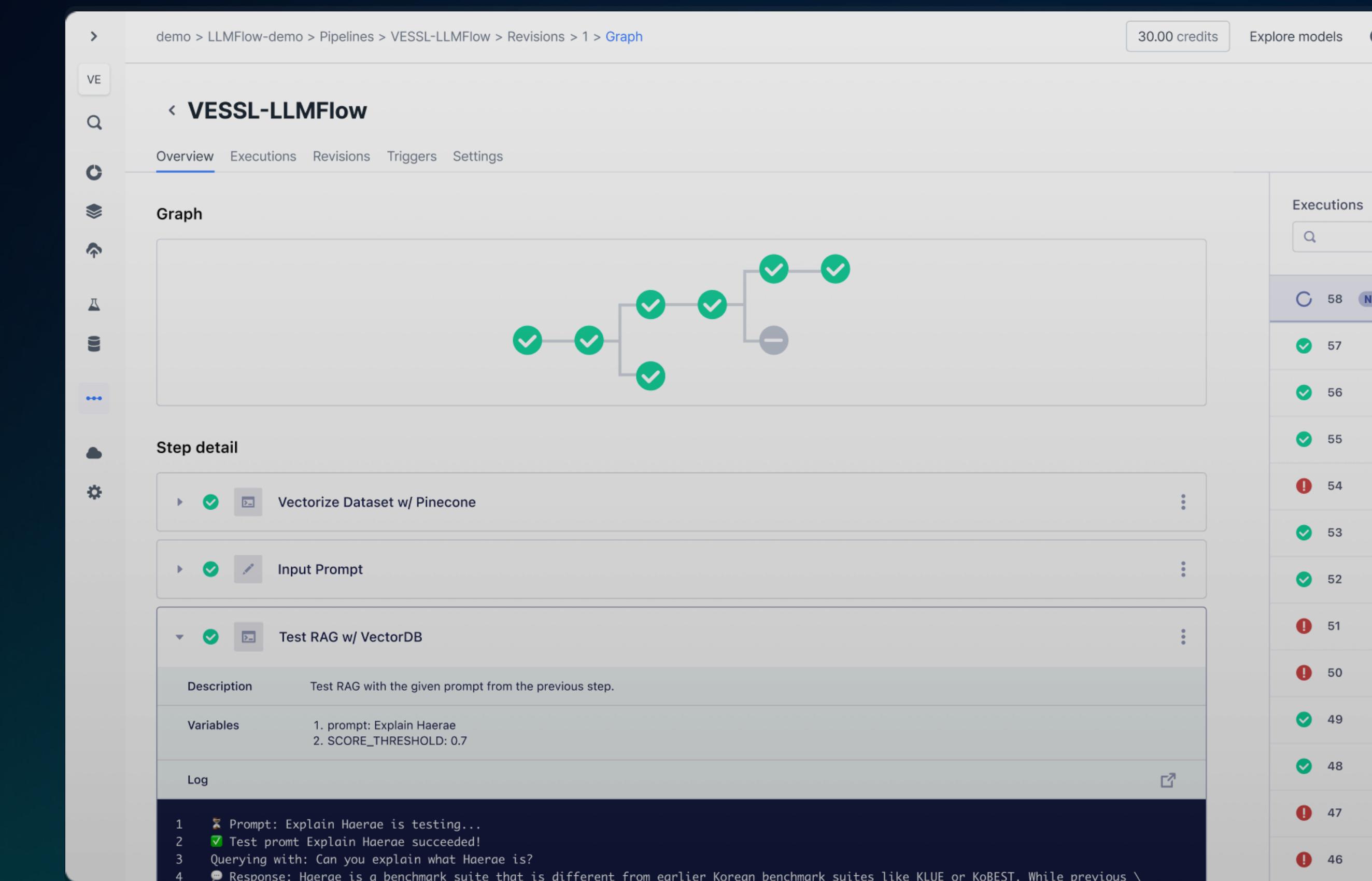


VESSL Pipeline

파이프라인을 통한 LLM Agent 개발 · 운영

The complete software stack for AI

파이프라인에 사내 Knowledge Base, Vector DB 등과 함께 구글 검색 등 외부 서비스를 연결하여, 나만의 LLM Agent를 만들 수 있습니다.



우리 회사 기술팀의 10년 비전

수백개의 AI 모델이 자동으로 개발되고 운영되는 인프라 구축까지

The complete software stack for AI

The screenshot shows the VESSL AI Hub interface. On the left is a sidebar with icons for different features: VE (selected), Q, C, H, A, E, and a cloud icon. The main area is titled "Hub" and shows a grid of AI models:

- Llama2-Finetuning**: Finetuning Llama 2 with a small code instruction dataset in Alpaca format. It includes a thumbnail of a llama wearing sunglasses, and buttons for "text generation", "llama2", "finetuning", and "quickstart".
- Mixtral-8x7b.chatbot**: Mixtral-7B: An open-source LLM which achieves both efficiency and high performance. It includes a thumbnail of a dark background with the text "Mixtral of experts" and "A high quality Sparse Mixture-of-Experts.", and buttons for "inference", "LLM", "text generation", and "quickstart".
- Stable Diffusion WebUI**: Run stable diffusion with multiple checkpoints on web app. It includes a thumbnail of a person in a spacesuit riding a horse, and buttons for "inference", "stable diffusion", "text to image", and "+1".
- Whisper-v3**: A model for automatic speech recognition and speech translation with large dataset of diverse ... It includes a thumbnail of a yellow speech recognition interface.
- SSD-1B**: Faster, smaller SDXL variant excelling in high-quality text-to-image generation with diverse dataset ... It includes a thumbnail of several generated images.
- vLLM**: Serve Mistral-7B-Instruct-v0.2 with vLLM and VESSL Run. It includes a thumbnail of a person's face.

At the top right of the main area, there are buttons for "30.00 credits", "Explore models", and a user profile icon. The overall theme is a dark blue background with white text and light-colored cards for each model.

End of Document

VESSL - made with ❤️ in Seoul, South Korea



Building context-augmented LLMs with RAG & Vector DB

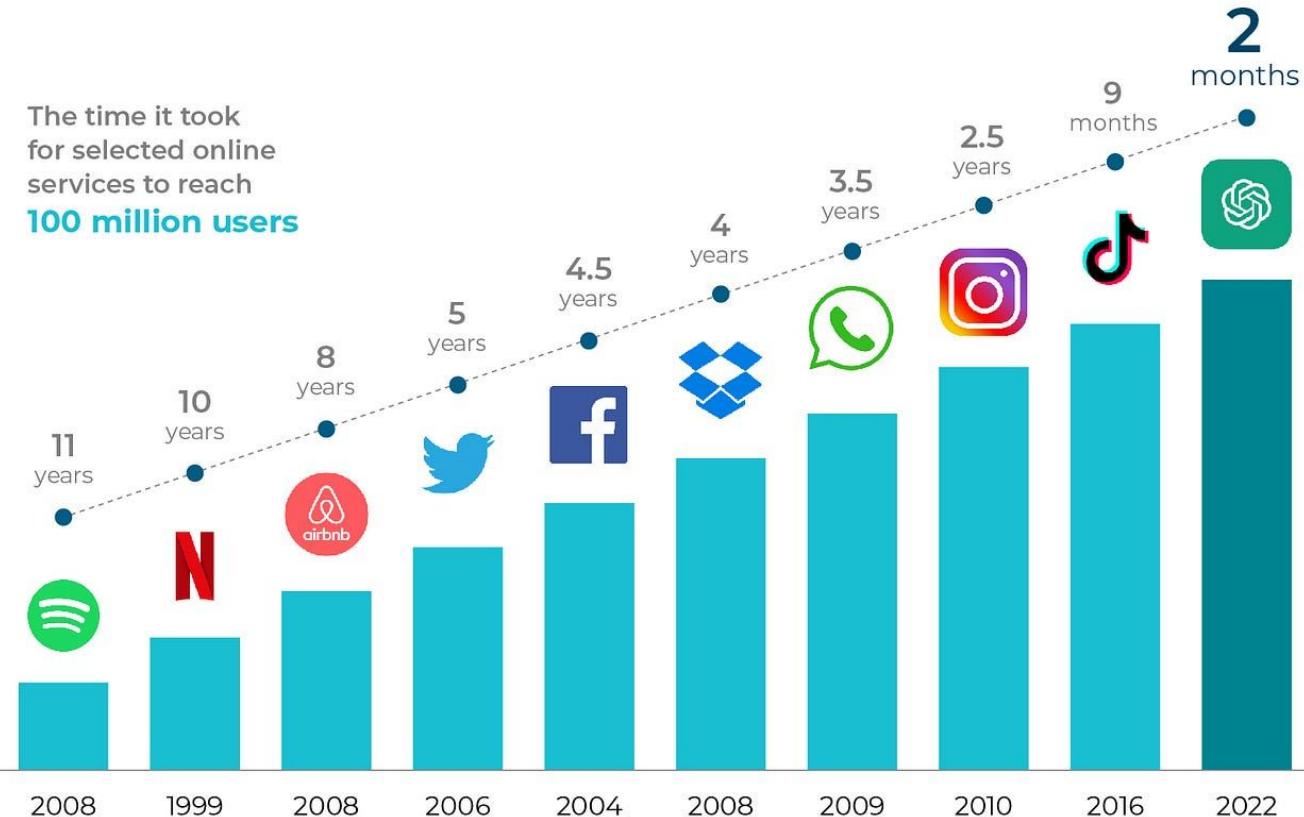
Michael Makaröv, Staff Software Engineer, Pinecone

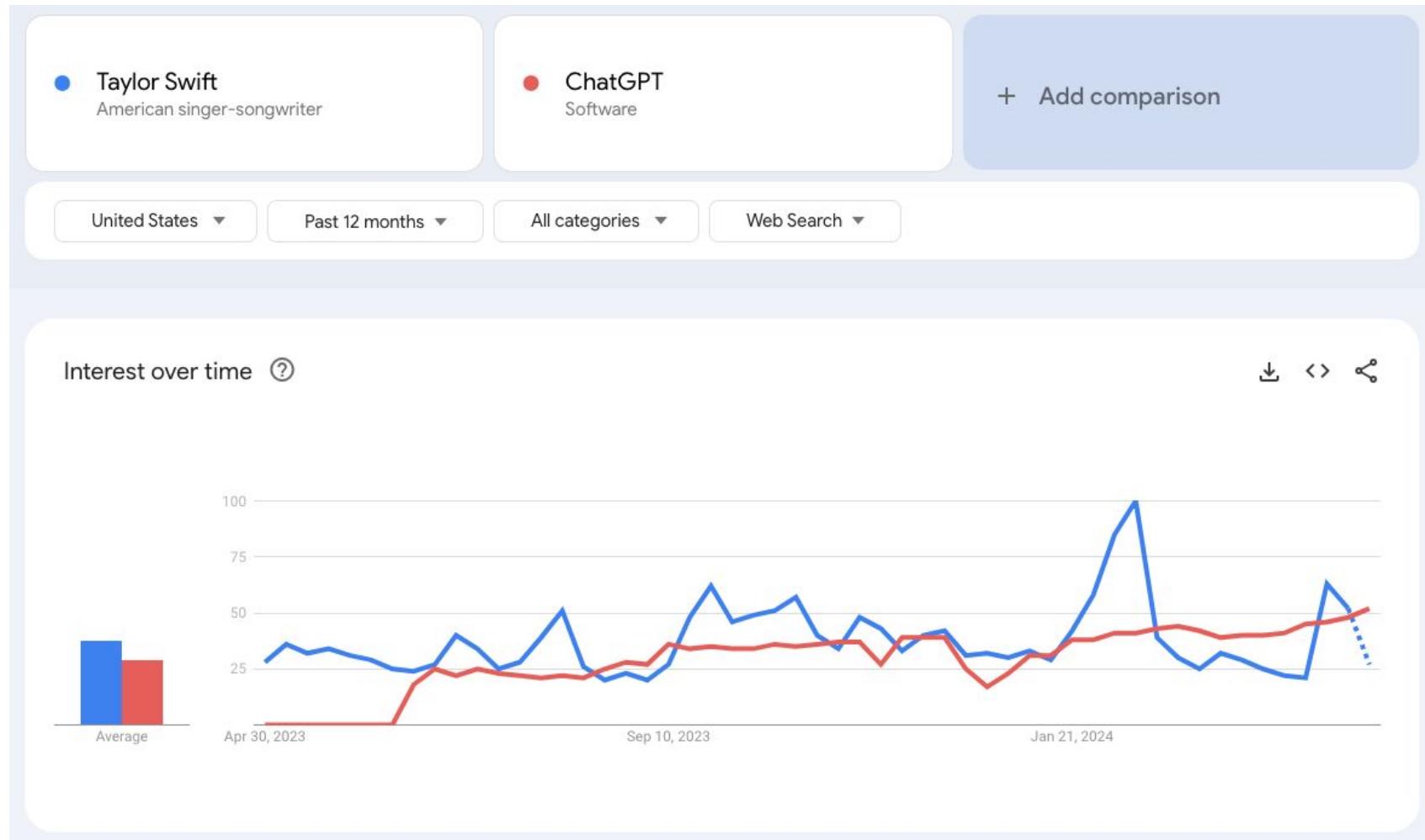
Flavors of RAG

Powering Generative AI with Vector Databases and Semantic Search

LLM Recap

Chat-GPT sprints to 100 million users





● Taylor Swift

American singer-songwriter

● ChatGPT

Software

+ Add comparison

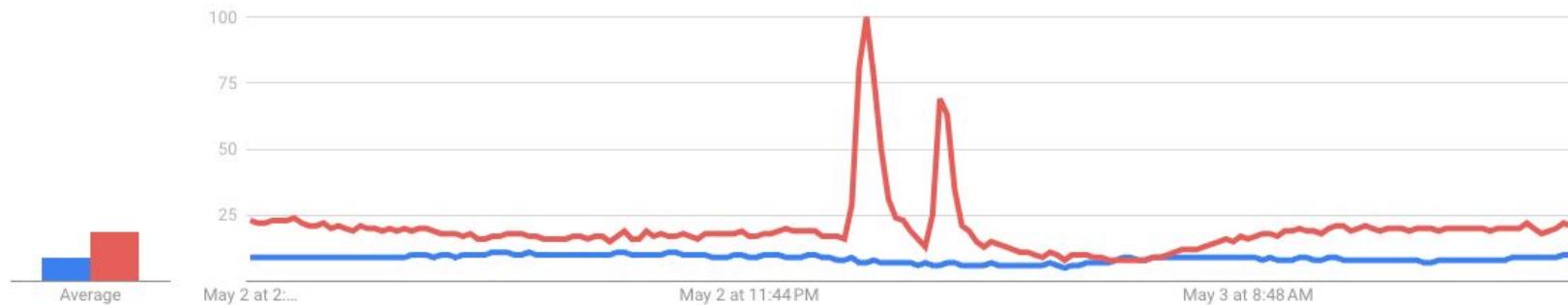
United States ▾

Past day ▾

All categories ▾

Web Search ▾

Interest over time ⓘ



● Taylor Swift
American singer-songwriter

● ChatGPT
Software

+ Add comparison

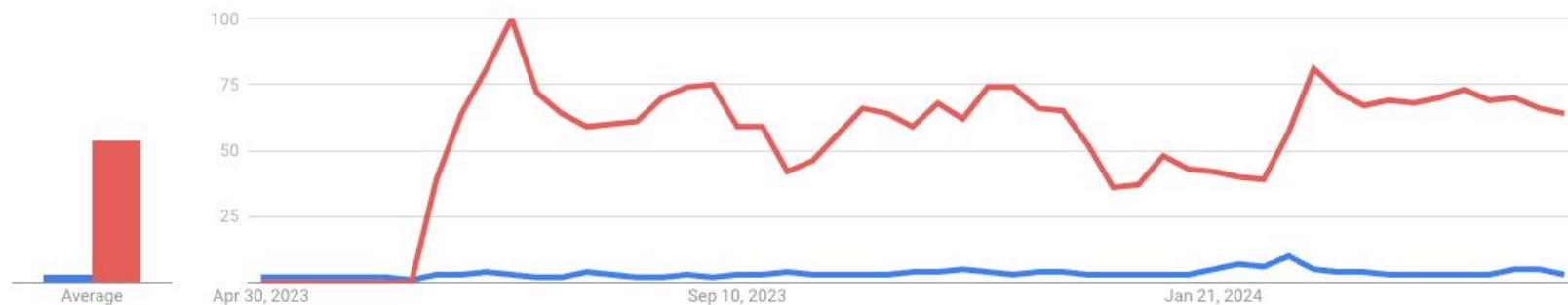
South Korea ▾

Past 12 months ▾

All categories ▾

Web Search ▾

Interest over time ?



● Taylor Swift
American singer-songwri...

● ChatGPT
Software

● BTS
Boy band

+ Add comparison

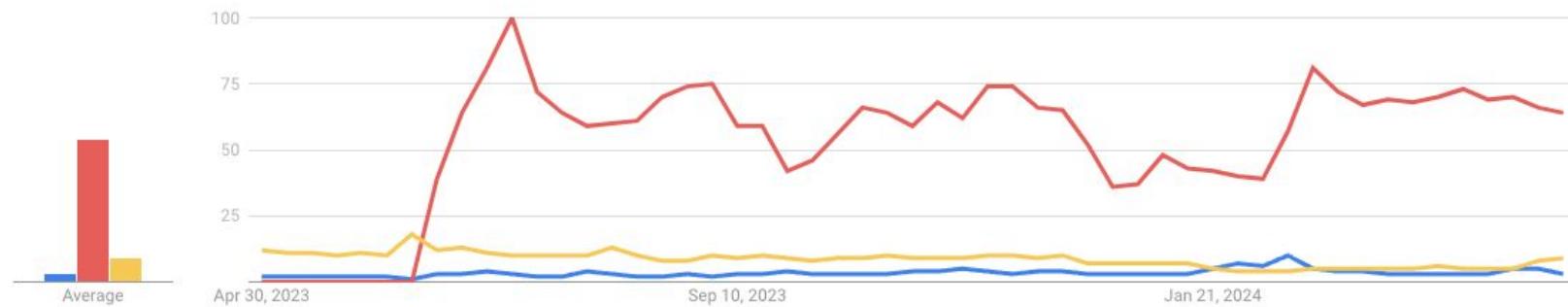
South Korea ▾

Past 12 months ▾

All categories ▾

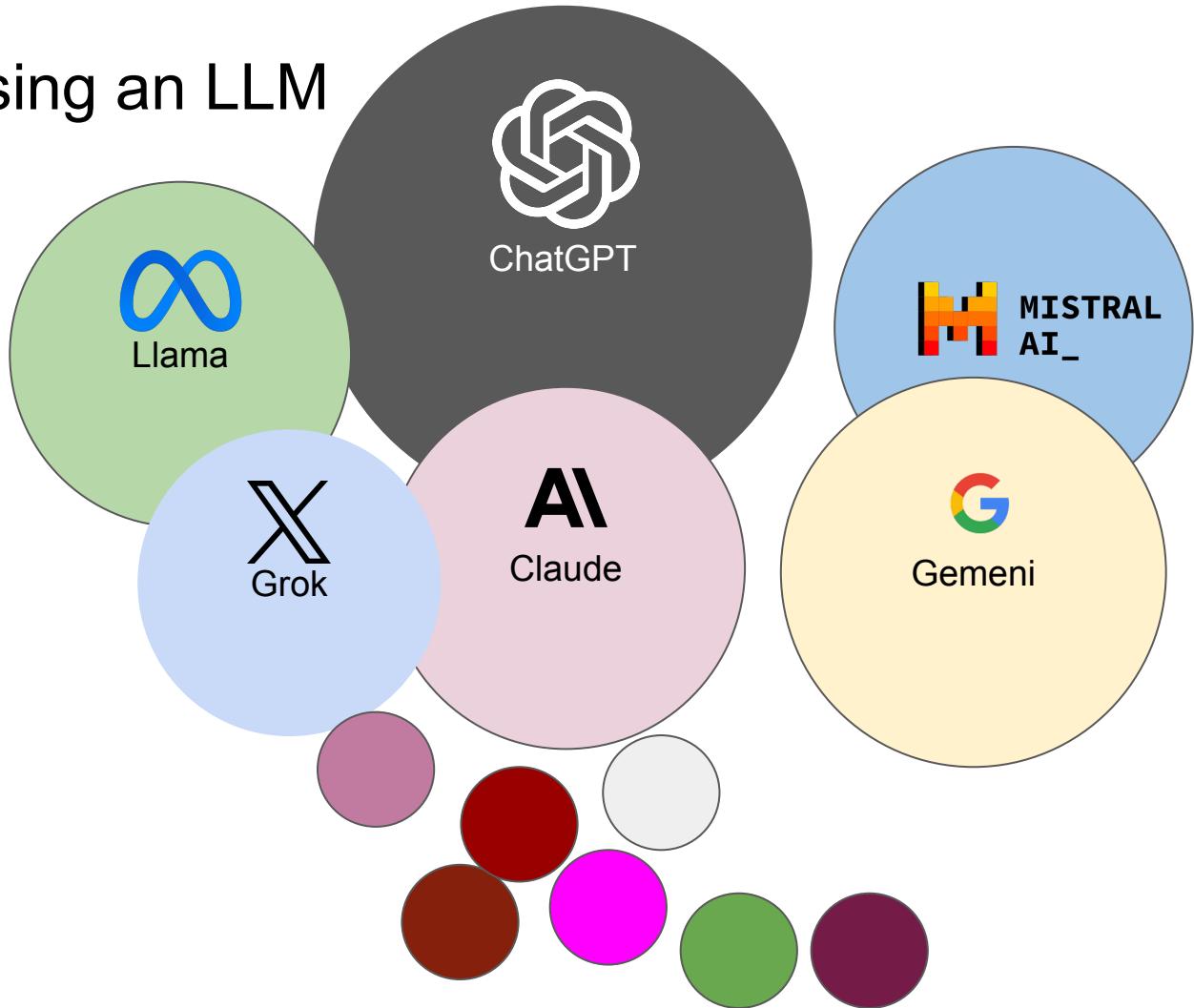
Web Search ▾

Interest over time ⓘ

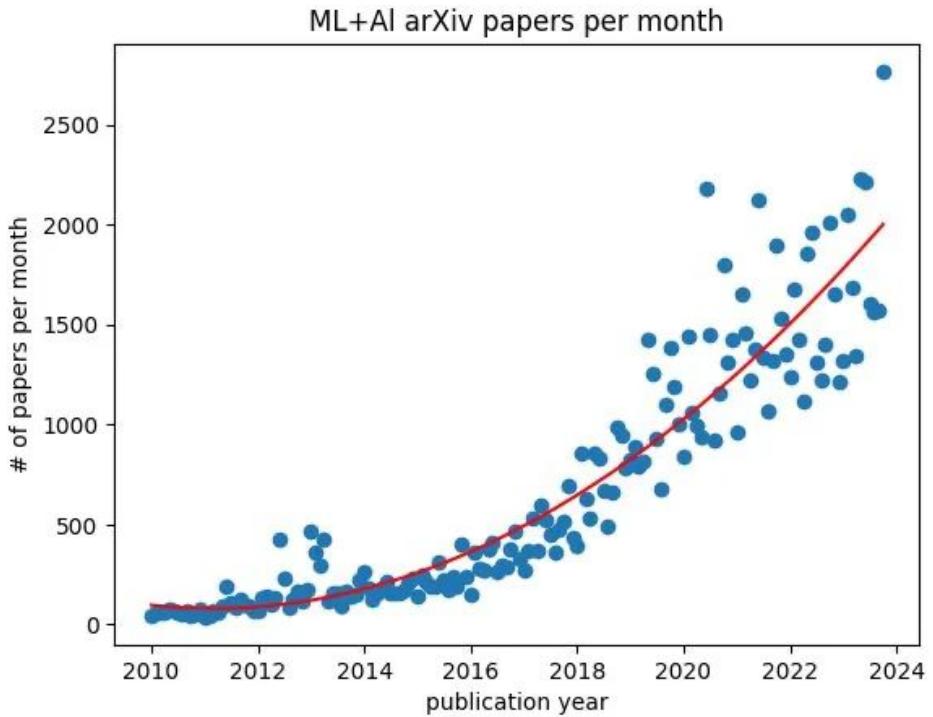


Someone is releasing an LLM As we speak

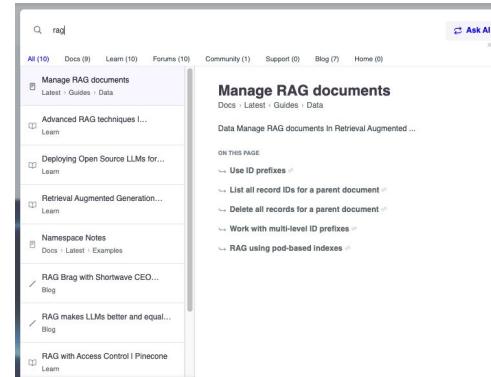
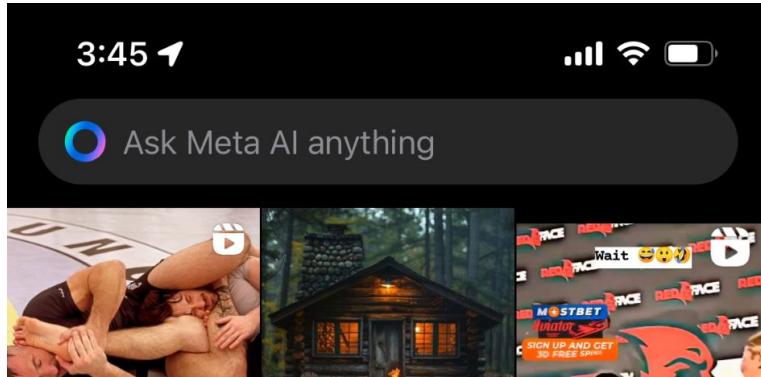
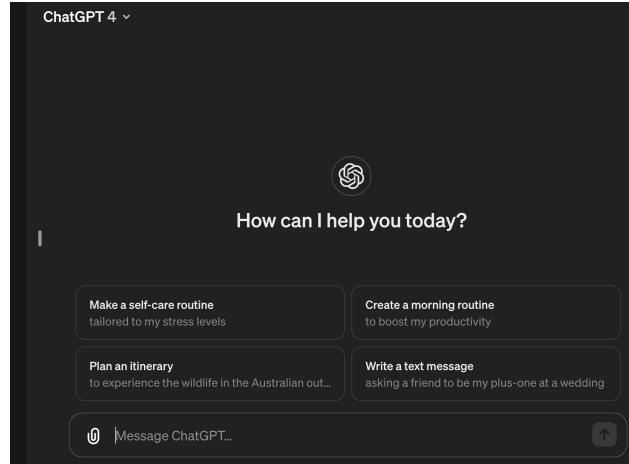
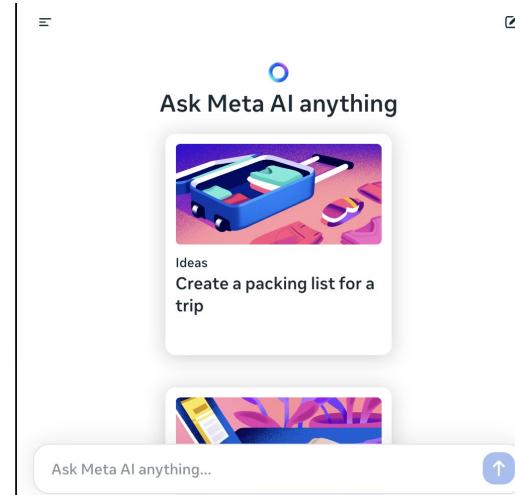
Rank* (UB)	Model	Arena Elo
1	GPT-4-Turbo-2024-04-09	1259
2	GPT-4-1106-preview	1253
2	Claude_3_Opus	1251
2	Gemini_1.5_Pro_API-0409-Preview	1250
2	GPT-4-0125-preview	1247
6	Llama-3-70b-Instruct	1210
6	Bard_(Gemini_Pro)	1209
7	Claude_3_Sonnet	1201
9	Command_Rt	1191
9	GPT-4-0314	1190
11	Claude_3_Haiku	1181
12	GPT-4-0613	1165



Someone publishing an ML research paper As we speak



Impacting customer interacting w/ products



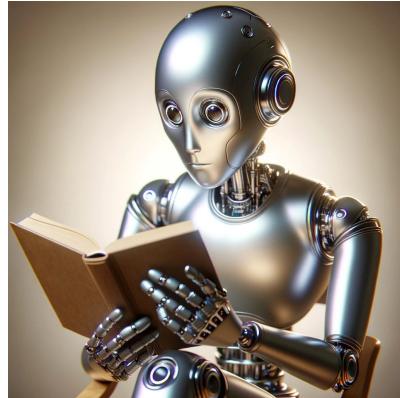


At its highest level, a typical GenAI workflow revolves around a user asking a large language model (LLM) a question.



“I last updated half a year ago”
“I don’t have access to your proprietary data”
“I hallucinate”

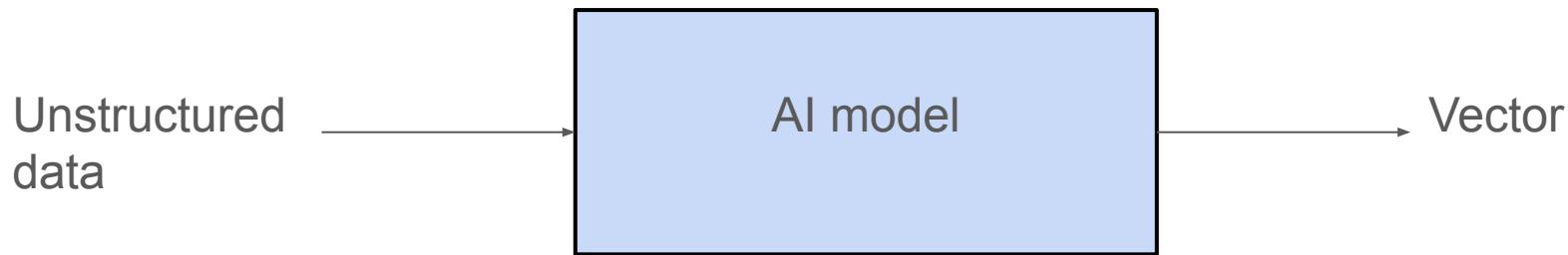
How do you teach LLMs



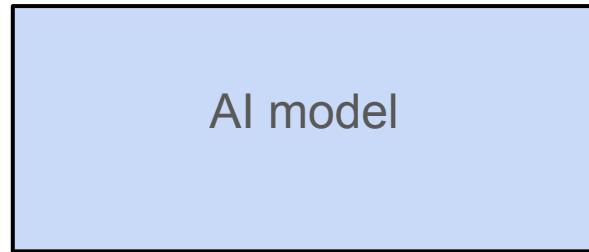
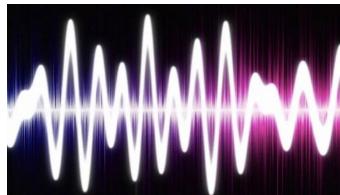
to have your information?

Vector Recap

Everything can be a vector



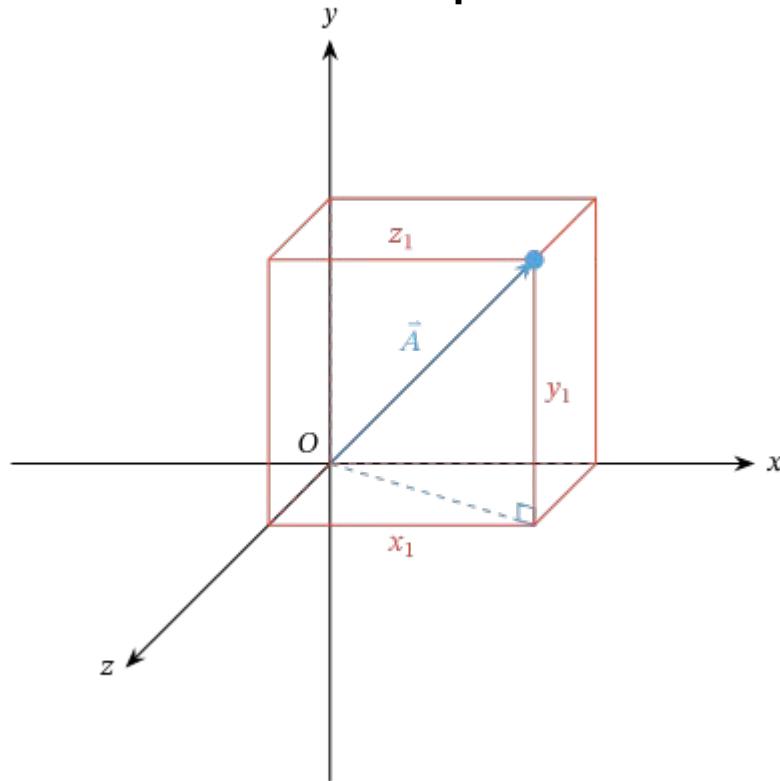
Anything can be a vector



[0.5, 0.7, -0.3,
0.7, 0.11]

“I’m a vector”

A point in a multi-dimensional space



Many dimensions...

[0.1, 0.7, 0.51, 0.71, 0.36, 0.54, 0.21, 0.99, 0.85, 0.77, 0.89, 0.14, 0.15, 0.24, ...]

THE JOURNAL OF CLIMATE

卷之三

卷之三

THE JOURNAL OF CLIMATE

卷之三

卷之三

卷之三

卷之三

卷之三

卷之三

卷之三

ପ୍ରକାଶକ ପତ୍ର ମହିନେ ପରିଚୟ

卷之三

ପ୍ରକାଶକ ପତ୍ର ପରିଚୟ

DODGE HIGH SCHOOL CHAMPS

THEORY OF THE SPHERICAL HARMONICS

卷之三

THE JOURNAL OF CLIMATE

卷之三

ପ୍ରକାଶକ ମେଳିକା ପରିଷଦ ଓ ବ୍ୟାପକ ନିର୍ମାଣ କାନ୍ତି

三

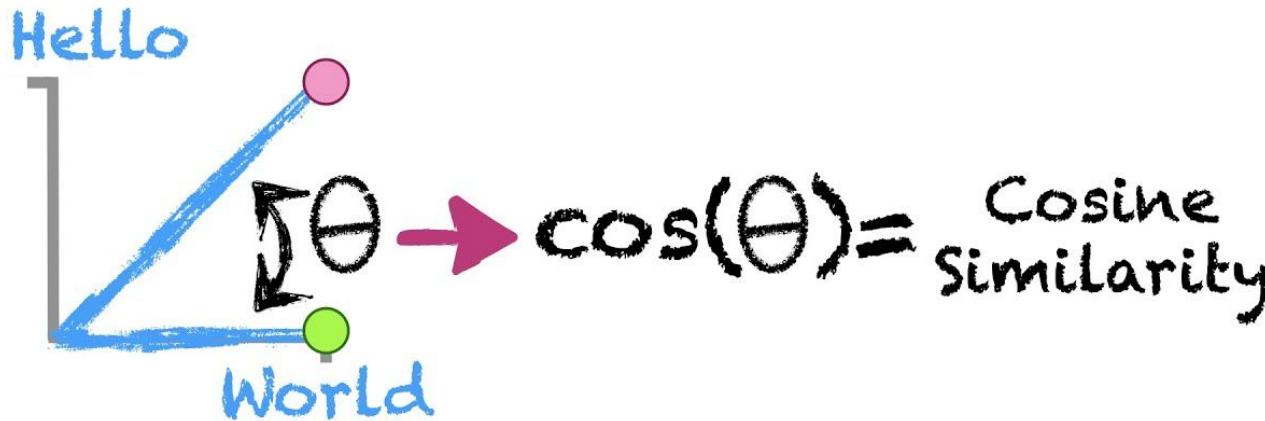
בְּנֵי יִשְׂרָאֵל בְּמִזְבֵּחַ וְבְמִזְבֵּחַ תְּמִימָה

卷之三

Vectors are the new
data structure that
enables similarity
search

Similarity Search

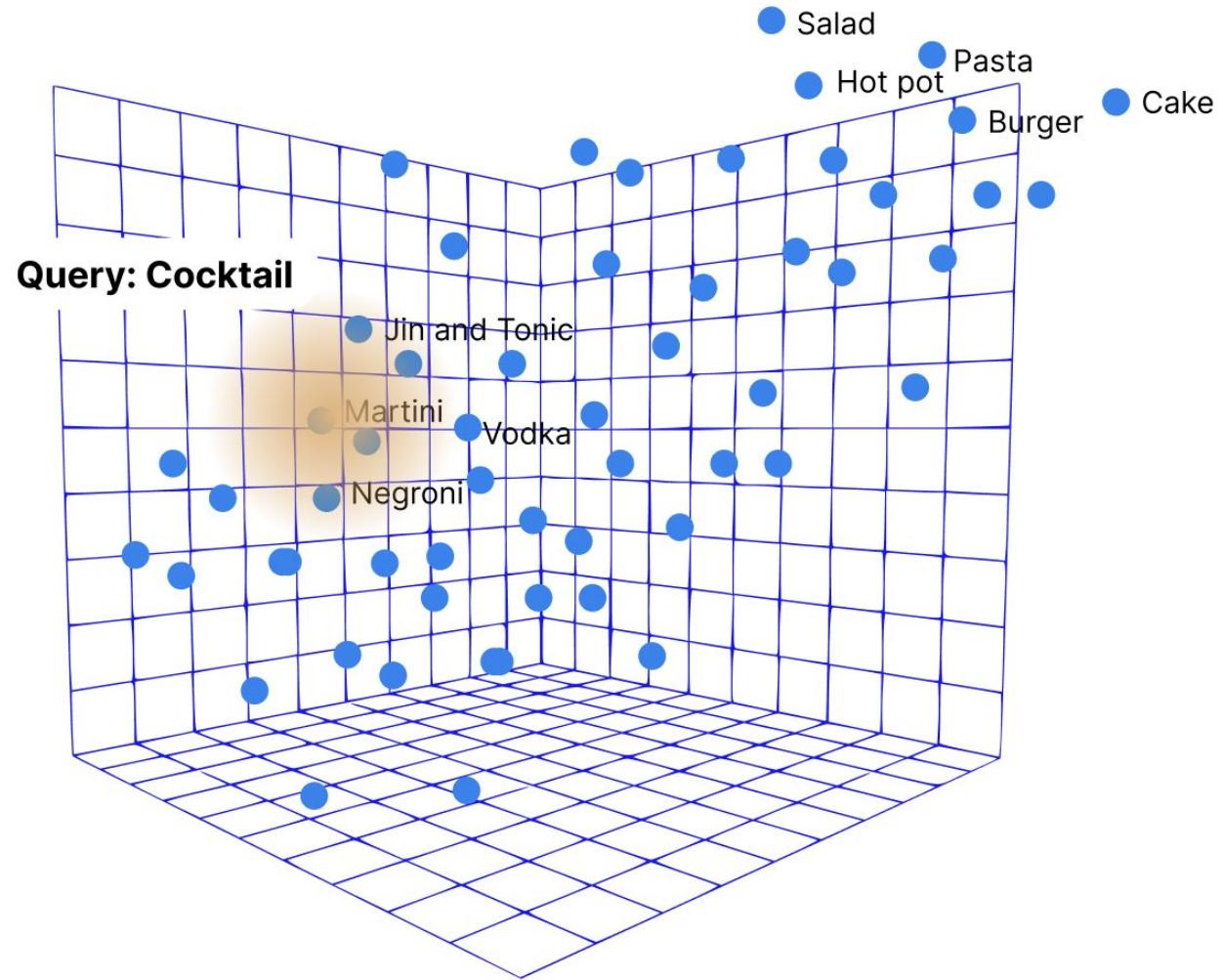
Cosine Similarity...



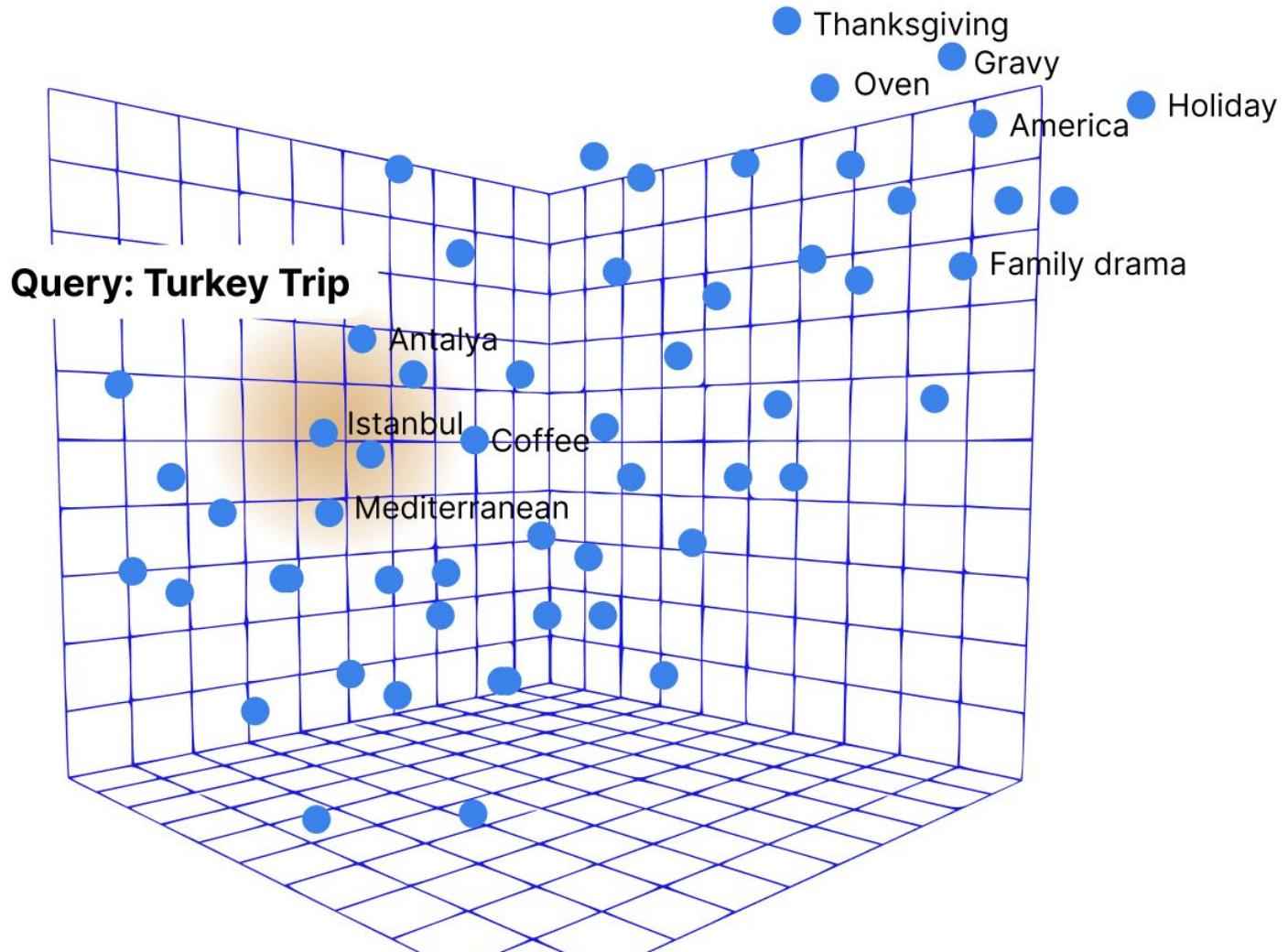
...Clearly Explained!!!

Vector Database

top_k



Semantic Search = Search by Meaning



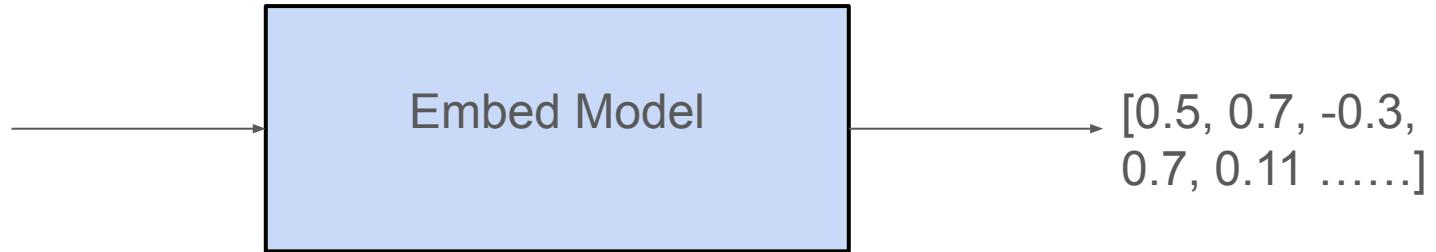
Semantic Search Recap

Text can be a vector

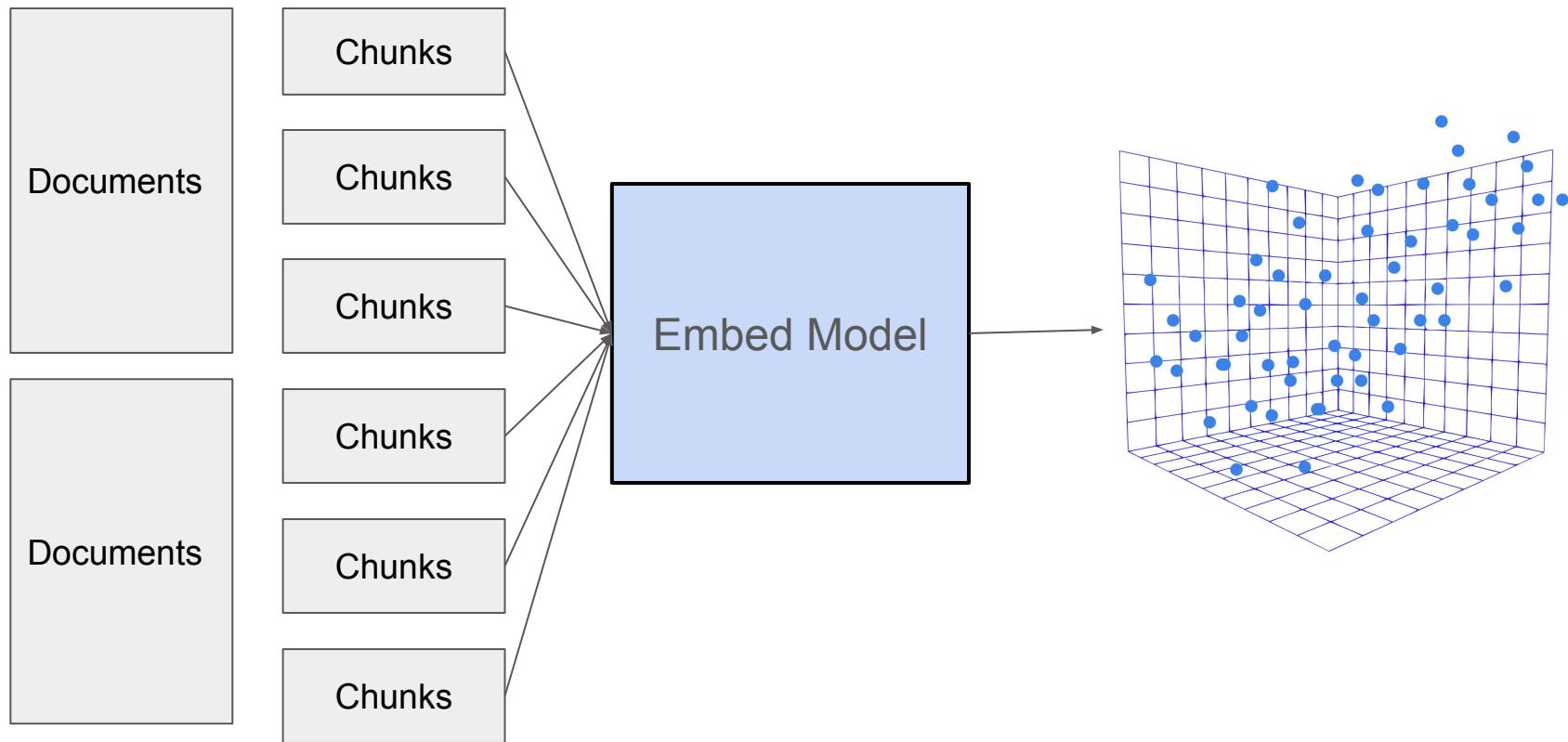


Text can be a vector

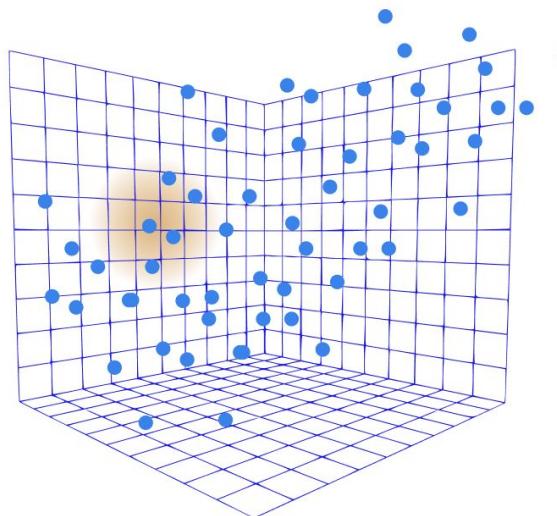
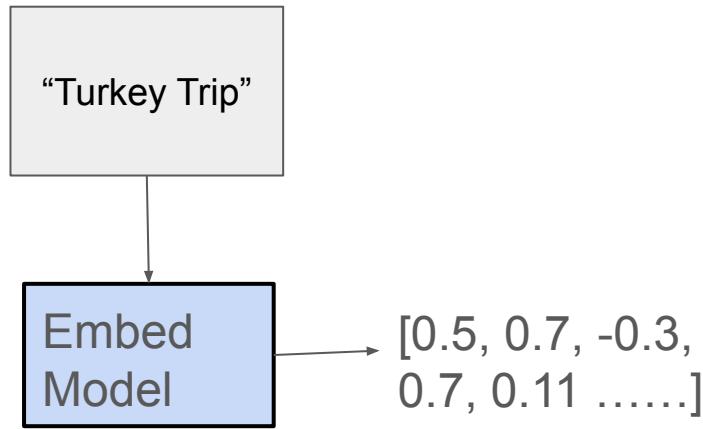
“Can this
be a
vector?”



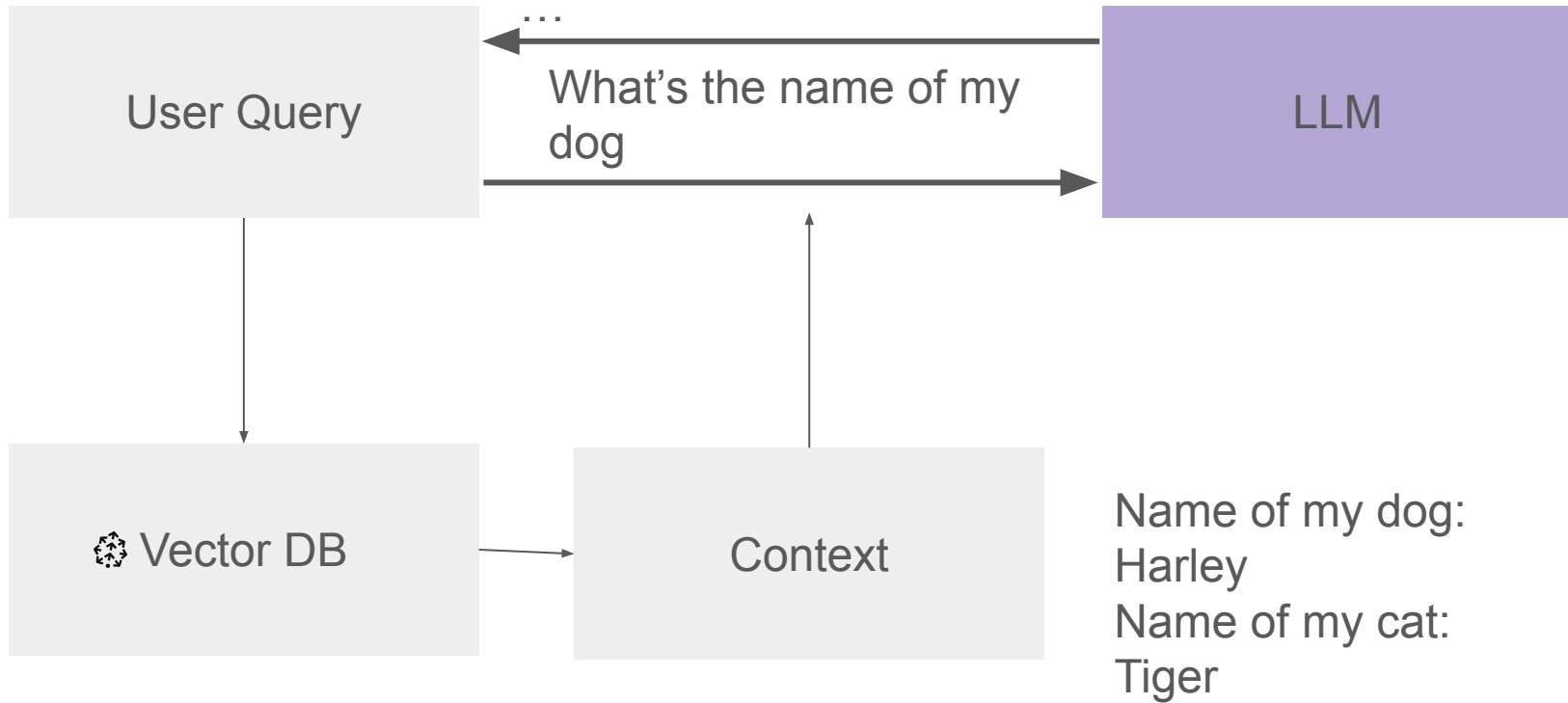
You can vectorize your documents



And retrieve most relevant documents by similarity



When you build a RAG application, the typical GenAI workflow gets interrupted by a call to an external knowledge base. By doing so, you are giving the LLM access to information it otherwise would not know.



<https://www.pinecone.io/learn/advanced-rag-techniques/>

RAG

Retrieval-Augmented Generation



Question ambiguity

Users sometimes use questions that aren't well defined, and may lead to irrelevant retrieval.

Low retrieval accuracy

The retrieved documents may not all be equally relevant to the question in hand.

Limited knowledge

The knowledge base may not include the information the user is looking for.

Context window performance limitations

Trying to "over-retrieve" may hit on the capacity of the context window, or otherwise produce a context window that is too big to return a result in a reasonable amount of time.

<https://arxiv.org/abs/2402.03367>

RAG Fusion

Wanna learn make latte art

Wanna learn make latte art (Original)

Best online courses for learning latte art

Step-by-step guide to making latte art for beginners

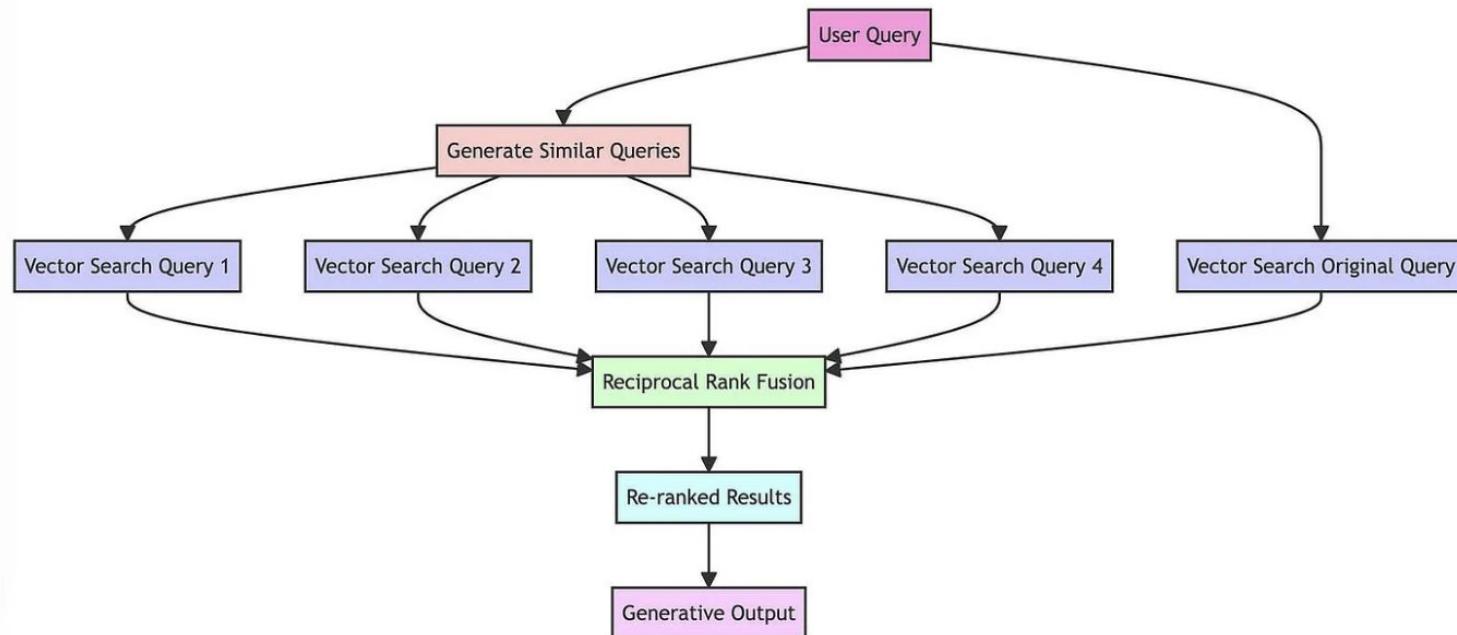
Tips and tricks for perfecting latte art designs

You are a helpful assistant that generates multiple search queries based on a single input query. Generate <NUMBER> search queries, one on each line, related to the following input query:

Query: "How can I learn how to make latte art?"

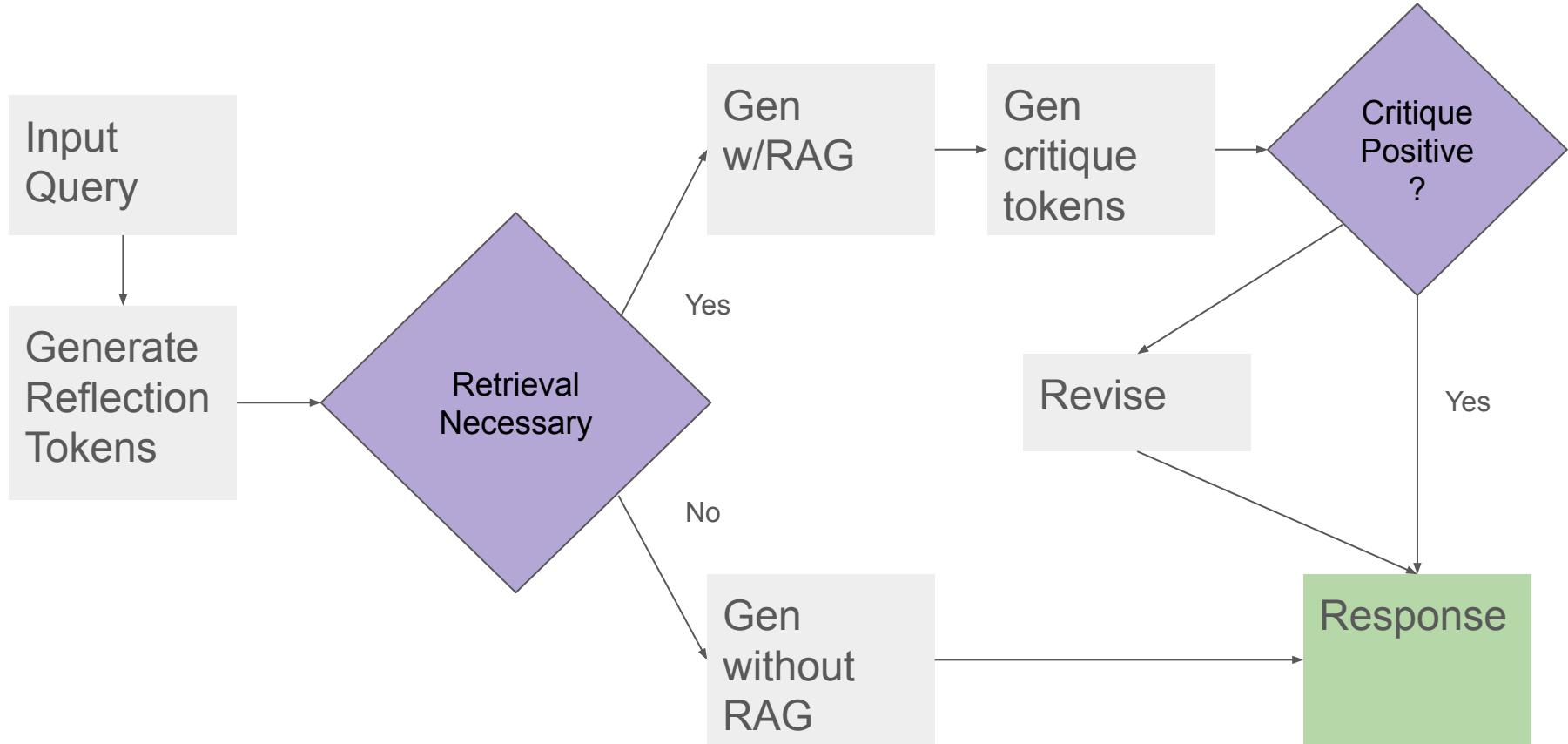
Queries:

Rag Fusion



Self Reflective RAG

Self-Reflective RAG



Type	Input	Output	Definitions
Retrieve	$x / x, y$	{yes, no, continue}	Decides when to retrieve with \mathcal{R} .
ISREL	x, d	{relevant, irrelevant}	d provides useful information to solve x .
ISSUP	x, d, y	{fully supported, partially supported, no support}	All of the verification-worthy statement in y is supported by d .
ISUSE	x, y	{5, 4, 3, 2, 1}	y is a useful response to x .

Retrieval-Augmented Generation (RAG)

Prompt How did US states get their names?

Step 1: Retrieve K documents

① Of the fifty states, eleven are named after an individual person.

② Popular names by states. In Texas, Emma is a popular baby name.

Retriever ③ California was named after a fictional island in a Spanish book.

Step 2: Prompt LM with K docs and generate

Prompt How did US states get their names? + ①②③

LM US states got their names from a variety of sources. Eleven states are named after an individual person (e.g., California was named after Christopher Columbus). Some states including Texas and Utah, are named after American tribes. No information in passages

Prompt: Write an essay of your best summer vacation



Ours: Self-reflective Retrieval-Augmented Generation (Self-RAG)

Prompt How did US states get their names?

Step 1: Retrieve on demand

① US states got their names from a variety of sources. [Retriever]



Step 2: Generate segment in parallel

Prompt + ①



Retriever 11 of 50 state names

Supported

come from persons.

Irrelevant Texas is named after a Native American tribe.

Relevant California's name has its

origins in a 16th-century novel

Partially

Las Sergas de Esplendán.

Prompt + ②



Retriever 11 of 50 state names

Supported

Irrelevant Texas is named after a Native American tribe.

Relevant California's name has its

origins in a 16th-century novel

Partially

Las Sergas de Esplendán.

Prompt + ③



Retriever 11 of 50 state names

Supported

Irrelevant Texas is named after a Native American tribe.

Relevant California's name has its

origins in a 16th-century novel

Partially

Las Sergas de Esplendán.

Step 3: Critique outputs and select best segment

① > ③ > ②

LM US states got their names from a variety of sources. 11 of 50 states names are come from persons. 11 of 50 states are named after Native Americans, including Utah.

Prompt: Write an essay of your best summer vacation



Corrective RAG

Lightweight
Relevancy
Evaluator
(confidence)

Evaluator based on a
small model
T5 (fine-tuned)

Q: What is Henry Feilden's occupation?

Q: Who was the screenwriter for Death of a Batman?



Retriever

Accurate Documents



Henry Feilden
(Conservative politician):
Henry Master Feilden
was an Conservative
Party politician...

Inaccurate Documents



Batman (1989 film):
of the murder of Bruce
Wayne's parents. When
Hamm's script was
rewritten, ...



Politician.



Generator



Hamm.



Generator

Correct



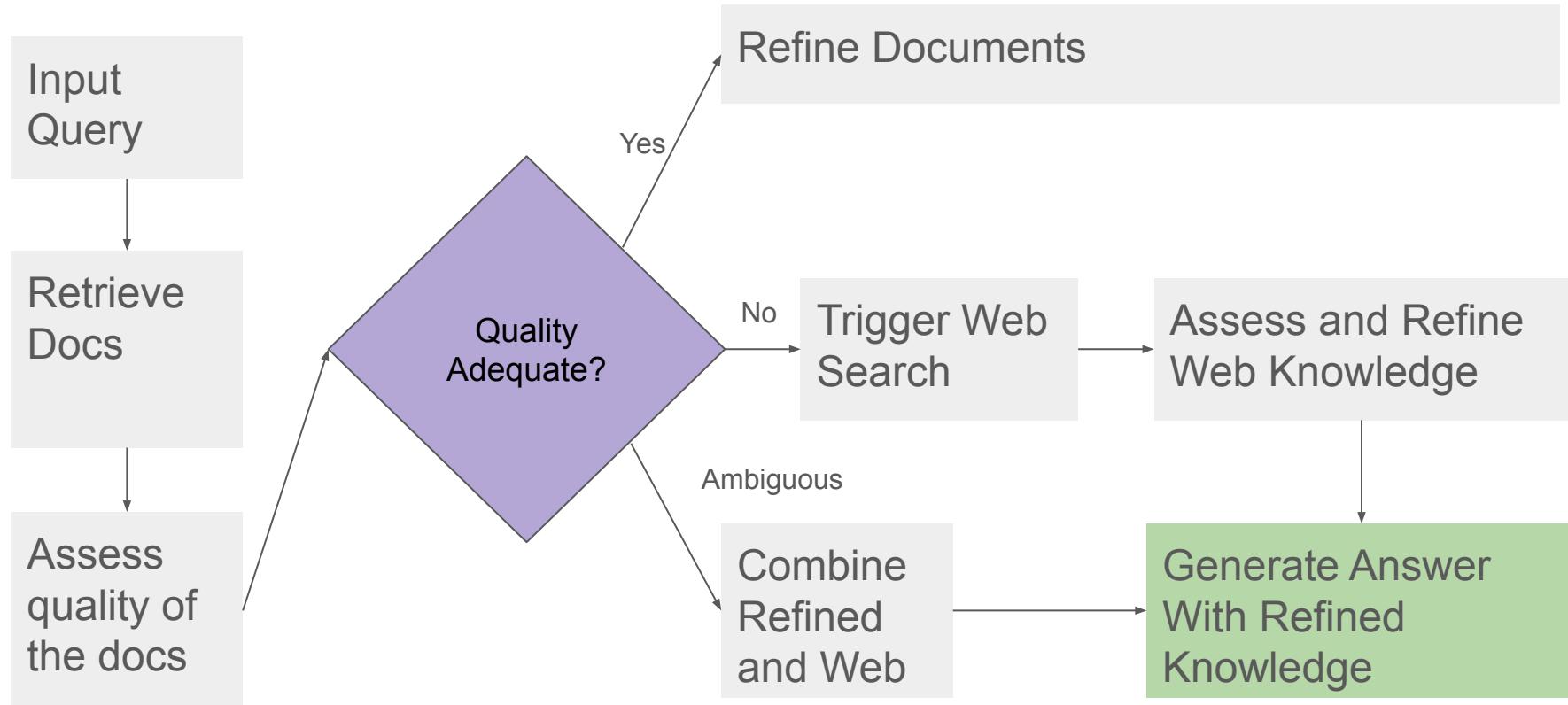
Ambiguous



Incorrect



Corrective Rag



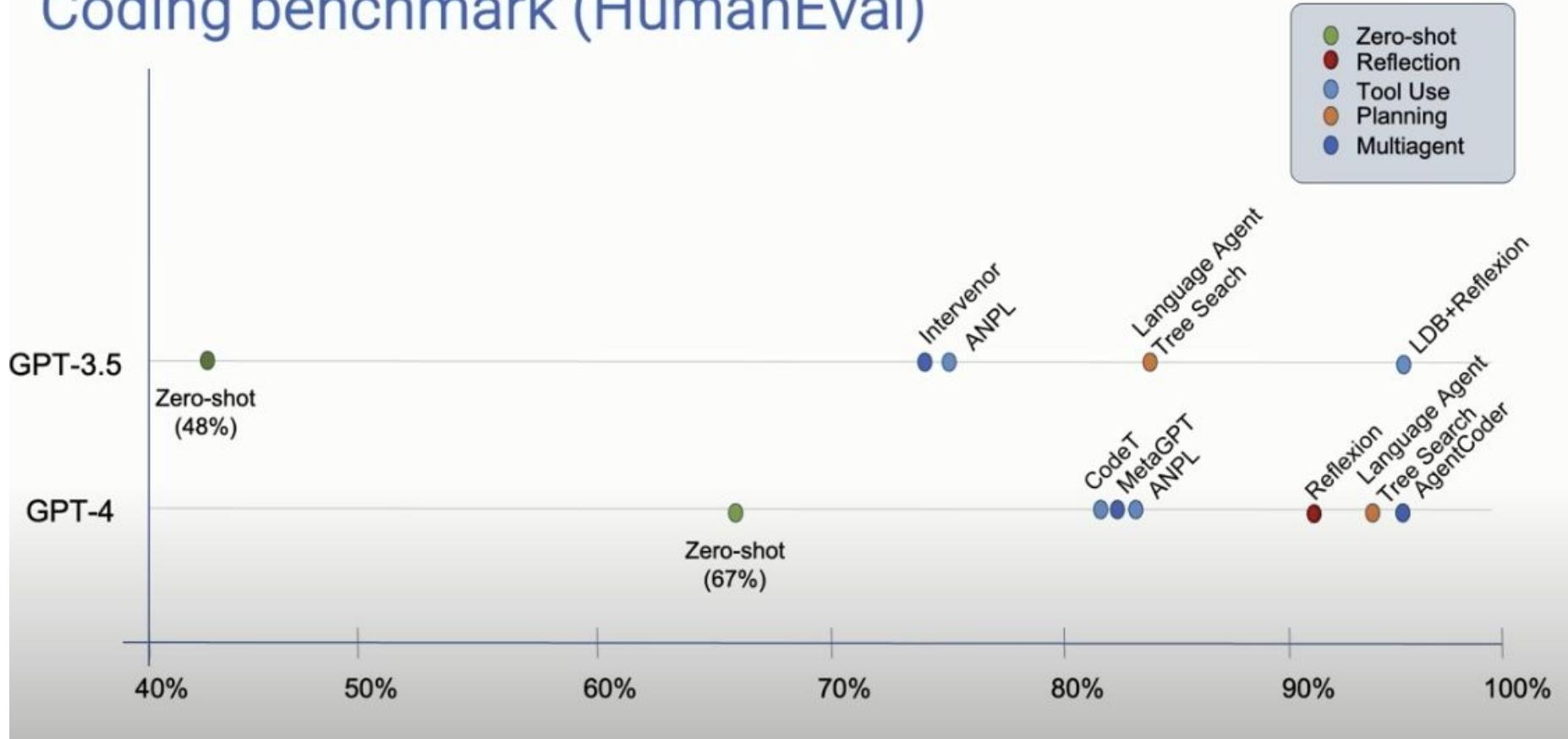
Feature	Self RAG	Corrective RAG	RAG Fusion
Overview	Enhances LM's quality and factuality through retrieval and self-reflection. Uses special tokens for adaptive retrieval and critique of its own generations.	Improves robustness of generation with a lightweight retrieval evaluator and a decompose-then-recompose algorithm for document refinement. Utilizes large-scale web searches for augmentation.	Combines RAG and Reciprocal Rank Fusion (RRF) by generating multiple queries, reranking with reciprocal scores, and fusing documents and scores for more accurate and comprehensive answers.
Key Mechanism	Fine-tuned with reflection tokens and critique tokens for on-demand retrieval and generation quality assessment.	Employs a retrieval evaluator to assess document quality and triggers actions (Correct, Incorrect, Ambiguous) based on confidence scores.	Generates multiple queries from the original query, reranks documents using RRF, and fuses them for the final output.
Advantages	Increases factuality and versatility of responses. Adaptable to diverse task requirements.	Significantly improves the performance of RAG-based approaches in both short- and long-form generation tasks.	Provides more accurate and comprehensive answers by contextualizing the original query from various perspectives.

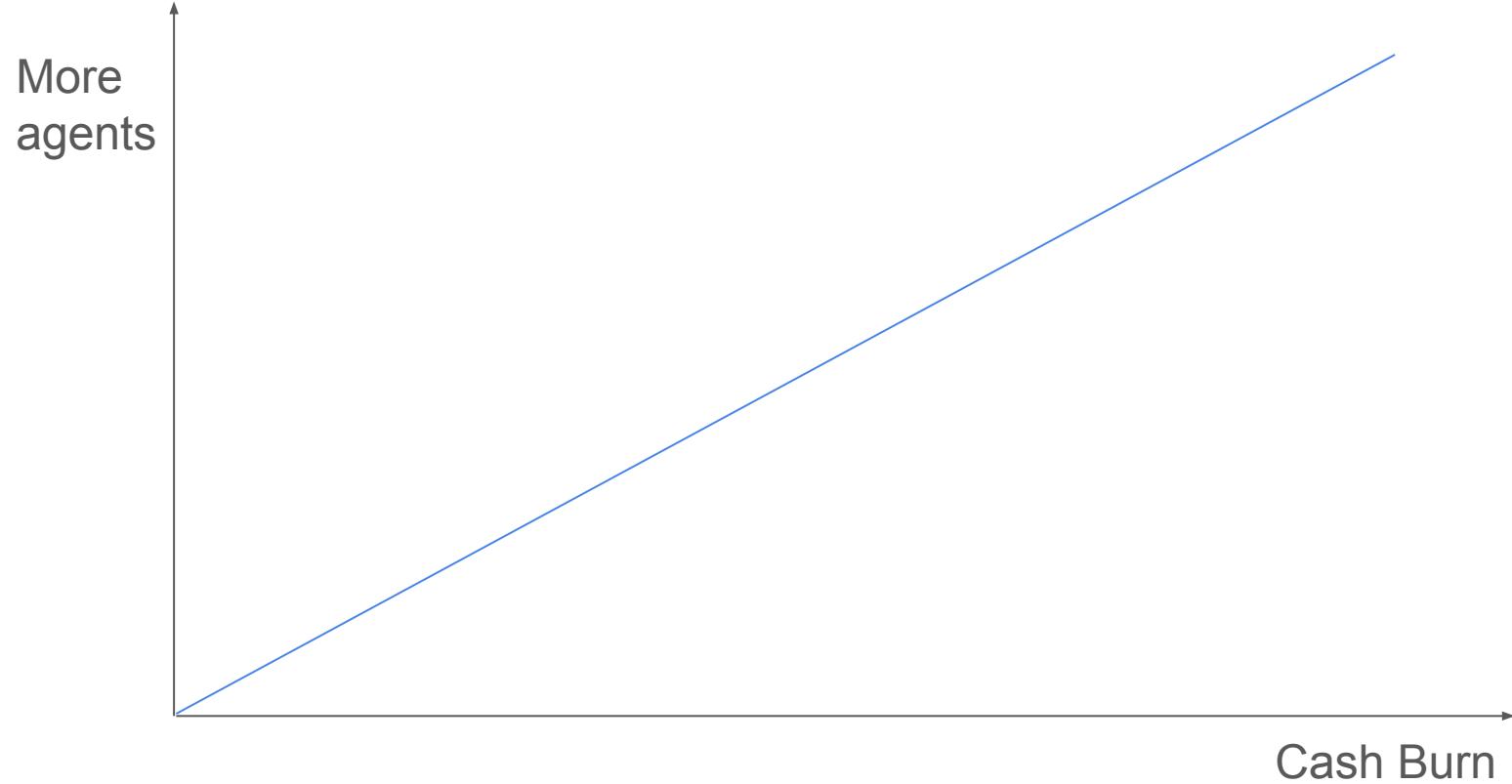
What's best RAG?

Try it yourself and combine!

<https://www.pinecone.io/learn/advanced-rag-techniques/>

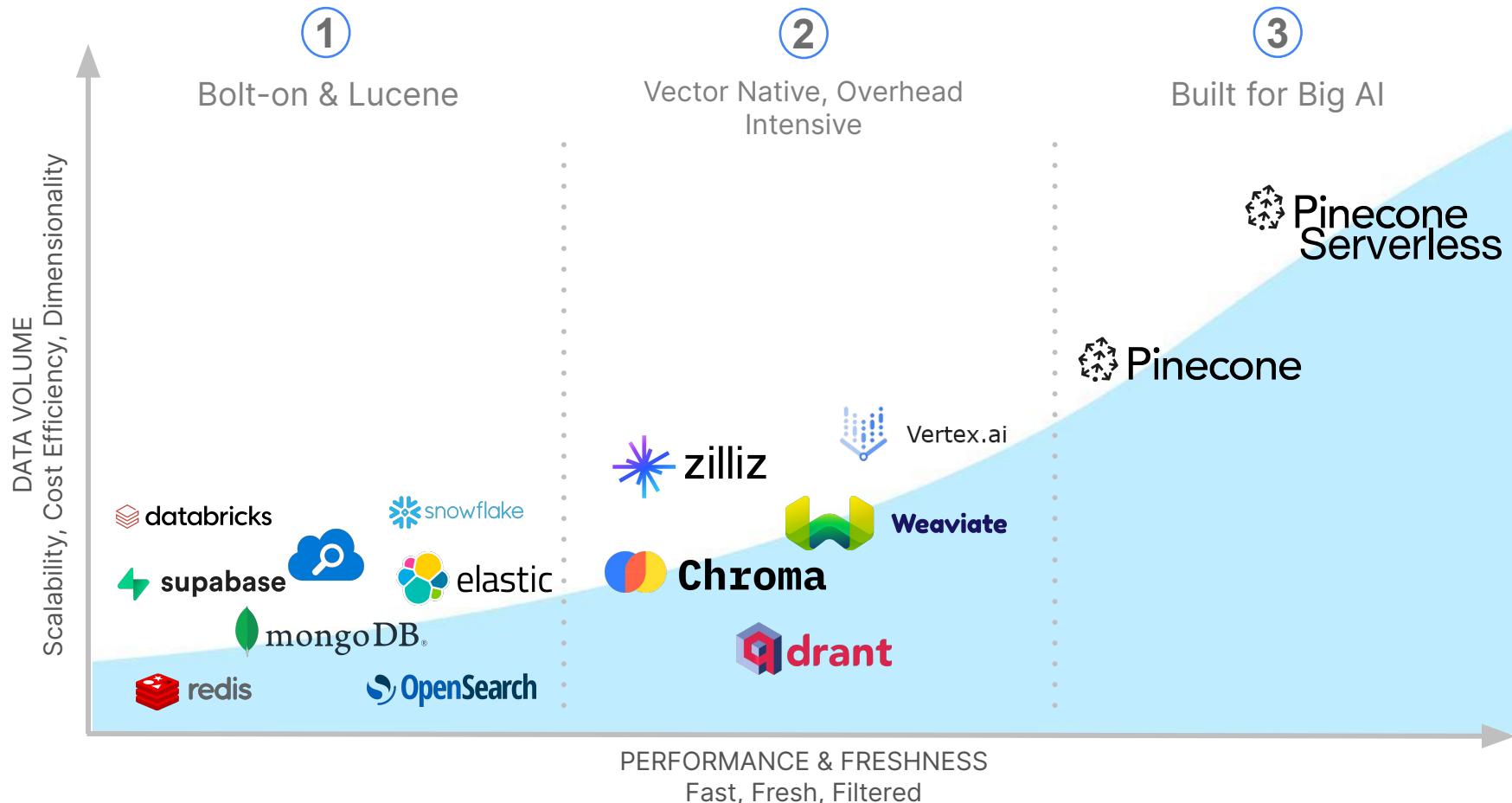
Coding benchmark (HumanEval)





The Vector Search Landscape

AC



Thank you!



Michael Makarov
Staff Engineer @ Pinecone

Twitter (X!): [@micmakarov](https://twitter.com/micmakarov)

Linkedin: [@micmakarov](https://www.linkedin.com/in/micmakarov)

Resources

[Pinecone Blog](#)

Rag Brag

[Andrew NG on Agents](#)

[LLM Leaderboard](#)



Advancing Vector Search for LLM in Production

고석현, CEO, Sionic AI

AI Acceleration Program

0 to full spectrum use case in 2 weeks



V VESSL AI × Pinecone

LinkedIn



YouTube



후기를 남겨주신 5분에게는 VESSL AI 굿즈를 선물로 드립니다