

240717(수) Generative AI at the Edge

지금은 생성형 AI의 시대!!



stability.ai



Midjourney



Adobe Firefly



synthesia



Jasper



Bard



OpenAI



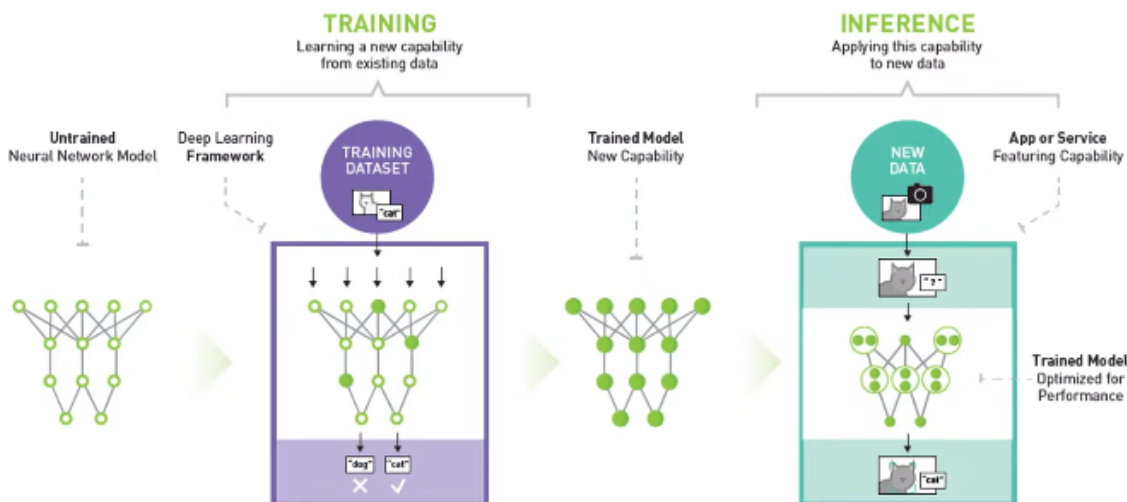
Nando.ai

The Edge



- 생성형 AI 연산은 고성능의 H/W와 대용량의 전력소비가 수반됨
- 대부분의 AI 서비스는 중앙의 'Cloud Server', 종말단의 'Edge Device' 형태로 구성
- Cloud Server와 Edge Device 간의 Response Time
- 빠른 반응 속도 / 높은 봉나 / 개인특화가 필요한 서비스는 Cloud Server에서 서비스하는데 문제가 있음
- 이를 해결하기 위한 Edge에서 생성형 AI를 수행하기 위한 다양한 시도가 진행됨

Deep Learning



다른 예시를 들어 보자면...



새로운 인공지능이 모델을 학습하는 모습
자전거를 타는 인공지능 학습 및 추론 과정

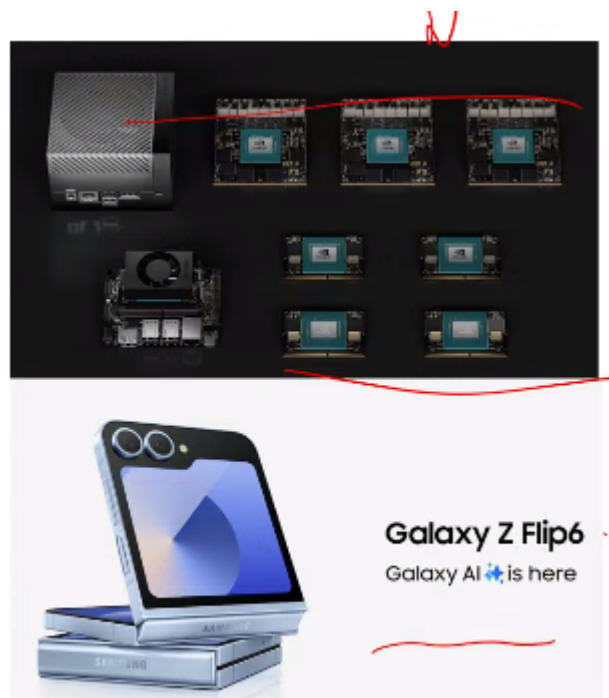


트레이닝 과정이 가장 많은 부하를 CPU, GPU에 부과한다.



그래서 부하를 나눌 방법을 찾는다.

Edge Device에서의 생성형 AI



- 고성능의 Large Language Model들이 출시되자 이를 Device 상에서 동작시키기 위한 노력이 지속
- GPU / NPU 상에서 해당 기술들을 적용한 제품들과 Develop Kits들이 출시

Edge 기반의 생성형 AI를 프로젝트에 어떻게 적용할 수 있을까?

프로젝트에 어떻게 적용할 수 있을까?

1. AIoT 트랙을 선택하였다면

NVIDIA Jetson Orin Nano 보드를 중심으로 프로젝트 기획



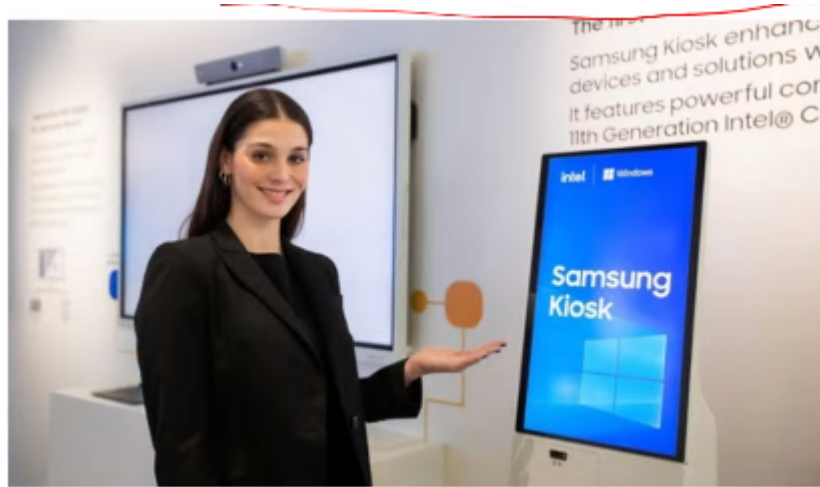
2. 모바일 서비스에 관심이 많다면

NPU가 포함된 휴대폰에서 생성형AI 기반의 모바일 프로젝트 기획



3. Web기반으로 Edge에서 생성형AI 기술을 사용하고자 한다면

NVIDIA GPU가 내장된 노트북에 로컬서버와 생성형 AI를 설치하고
웹기반 UI를 바탕으로 '디지털 사이니지' 등의 프로젝트를 기획



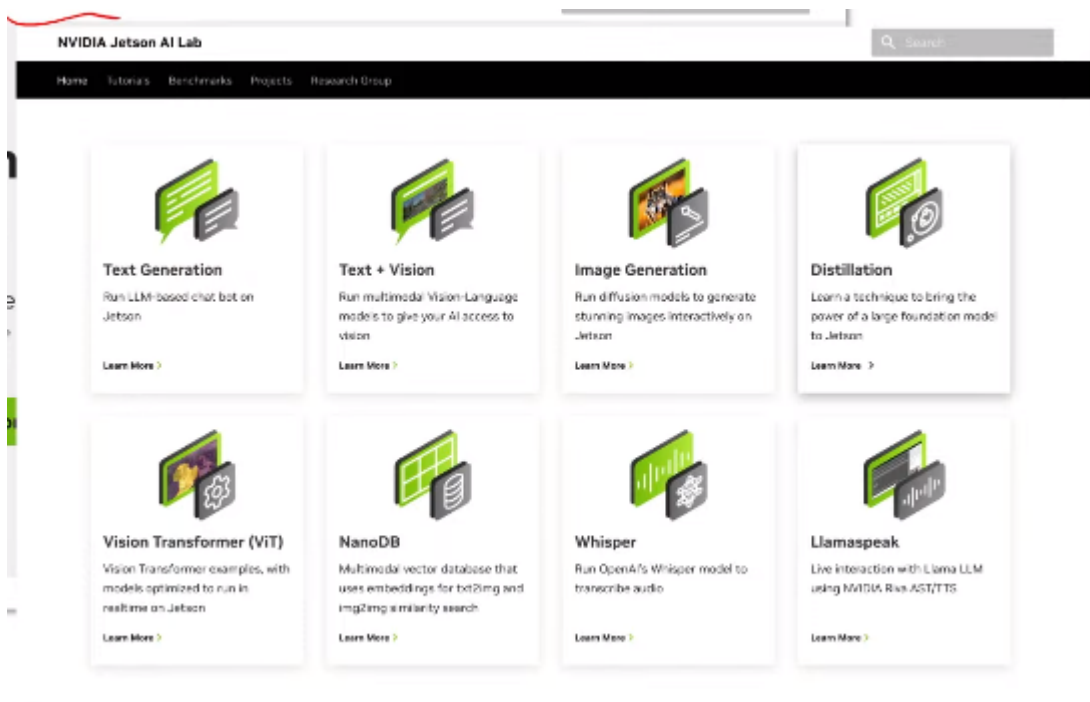
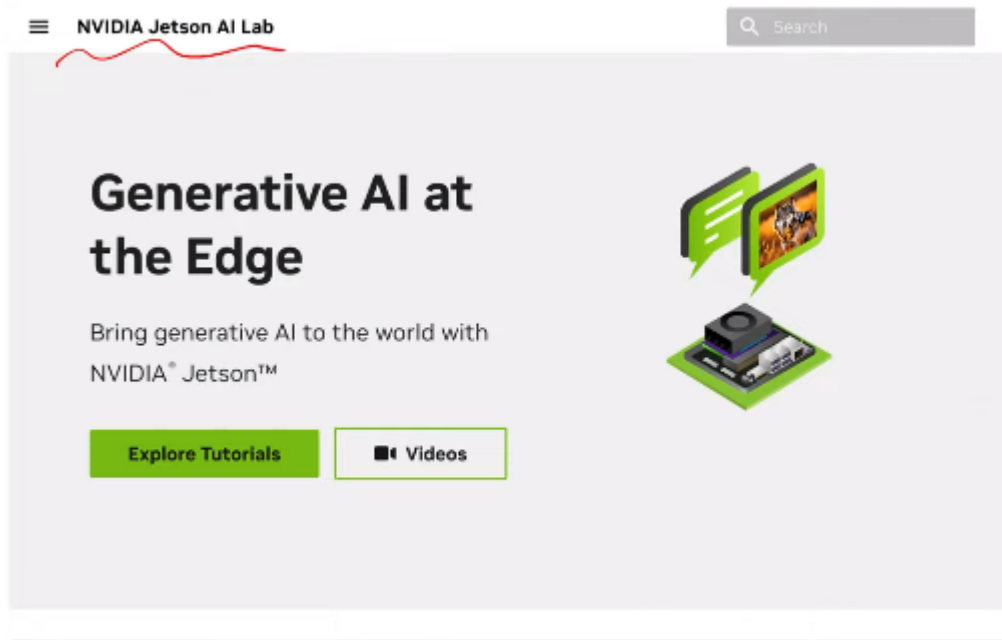
NVIDIA Jetson Orin Nano Development Kit

NVIDIA Jetson Orin Nano



- 11기 신설된 공통PJT AIoT 트랙에서 교보재로 제공
- 이전 버전 Jetson Nano 대비 80배 정도의 성능향상
- JetPack 6.0 등을 통해 OS와 생성형 AI 수행에 필요한 환경이 기본제공

NVIDIA Generative AI Models



Text + Vision (VLM) - LLaVA

LLaVA

Home Tutorials Benchmarks Projects Research Group

Tutorials

- Introduction
- Hello AI World
- Agent Studio
- Text (LLM)
 - text-generation-webui
 - ollama
 - llamaspeak
 - NanoLLM
 - Small LLM (SLM)
 - API Examples
- Text + Vision (VLM)
 - LLaVA**
 - Live LLaVA
 - NanoVLM
- Vision Transformers (ViT)
 - EfficientViT
 - NanoOWL
 - NanoSAM
 - SAM
 - TAM
- RAG & Vector Database
 - NanoDB
 - LlamaIndex
 - Jetson Copilot
- Audio
 - Whisper

Model	Quantization	Size	Memory
text-generation-webui	4-bit (GPTQ)	~ 7	9.7 GB
llava-serve-c11	FP16 (None)	4.2	27.7 GB
llava-serve	4-bit (G4_K)	10.1	9.2 GB
NanoVLM	4-bit (MLC)	21.1	8.7 GB

In addition to Llava, the [NanoVLM](#) pipeline supports [ViLA](#) and mini vision models that run on Orin Nano as well.

Chat Default Notebook Parameters Model Training Session

Send a message Show controls (Ctrl+G)

Vision Transformers - NanoOWL

NVIDIA Jetson AI Lab

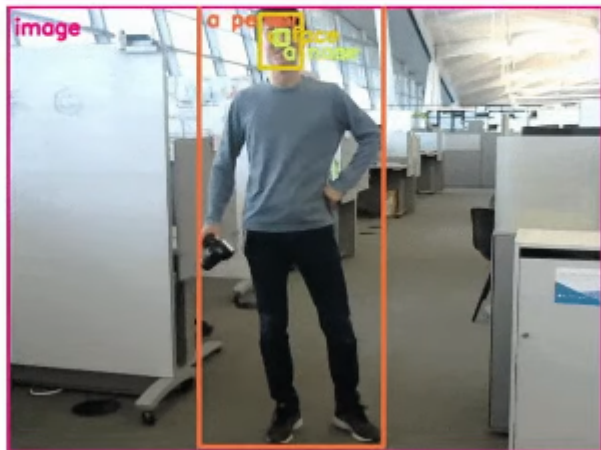
Home Tutorials Benchmarks Projects Research Group

Tutorials

- Text (LLM)
 - text-generation-webui
 - ollama
 - llamaspeak
 - NanoLLM
 - Small LLM (SLM)
 - API Examples
- Text + Vision (VLM)
 - LLaVA
 - Live LLaVA
 - NanoVLM
- Vision Transformers (ViT)
 - EfficientViT
 - NanoOWL**
 - NanoSAM
 - SAM
 - TAM
- RAG & Vector Database
 - NanoDB
 - LlamaIndex
 - Jetson Copilot
- Audio
 - Whisper
 - AudioCraft
 - VoiceCraft

Tutorial - NanoOWL

Let's run [NanoOWL](#), [OWL-ViT](#) optimized to run real-time on Jetson with [NVIDIA TensorRT](#).



[a person [a face [a nose]]]

RAG & Vector Database - NanoDB

Tutorials

Small LLM (SLM)

API Examples

Text + Vision (VLM)

LLaVA

Live LLaVA

NanoVLM

Vision Transformers (ViT)

EfficientViT

NanoDWT

NanoSAM

SAM

TAM

RAG & Vector Database

NanoDB

LlamaIndex

Jetson Copilot

Audio

Whisper

AudioCraft

VoiceCraft

Image Generation

Stable Diffusion

Stable Diffusion XL

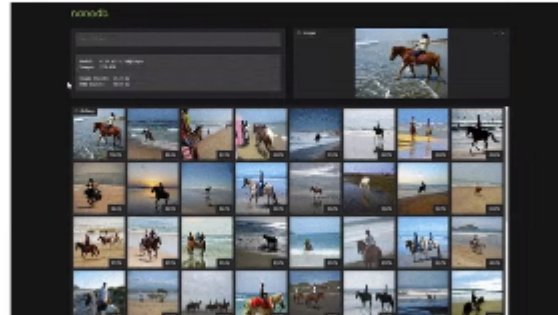
Tips

Knowledge Distillation

SSD + Docker

Tutorial - NanoDB

Let's run NanoDB's interactive demo to witness the impact of Vector Database that handles multimodal data.



What you need

- One of the following Jetson devices:
 - Jetson AGX Orin (64GB)
 - Jetson AGX Orin (32GB)
 - Jetson Orin NX (16GB)
 - Jetson Orin Nano (8GB)
- Running one of the following versions of JetPack:
 - JetPack 5 (L4T r35.x)
 - JetPack 5 (L4T r36.x)
- Sufficient storage space (preferably with NVMe SSD):
 - 16GB for container image

Image Generation - Stable Diffusion

Stable Diffusion

Tutorials

API Examples

Text + Vision (VLM)

LLaVA

Live LLaVA

NanoVLM

Vision Transformers (ViT)

EfficientViT

NanoGWL

NanoSAM

SAM

TAM

RAG & Vector Database

NanoDB

LlamaIndex

Jetson Copilot

Audio

Whisper

AudioCraft

VoiceCraft

Image Generation

Stable Diffusion

Stable Diffusion XL

Tips

Knowledge Distillation

SSD + Docker

Memory optimization

