



→ SESSION 2

Unity Sentis: Hugging Face 샘플들을 활용한 간단한 예제 만들기

**MONTHLY
TECH TALK**

김한얼, Senior Software Engineer, Unity



만나서
반갑습니다!



Sky Kim (김한얼)

Senior Software Engineer @Unity
AI and Simulations Technical Support
Disability Awareness Speaker

sky.kim@unity3d.com



오늘의 주제

- Hugging Face에서 사용할 수 있는 샘플 소개
- Hugging Face 샘플들을 활용해서 간단한 예제 만들기



UDay Seoul: Sentis 세션 다시보기



<https://www.unitysquare.co.kr/growwith/resource/form?id=519>

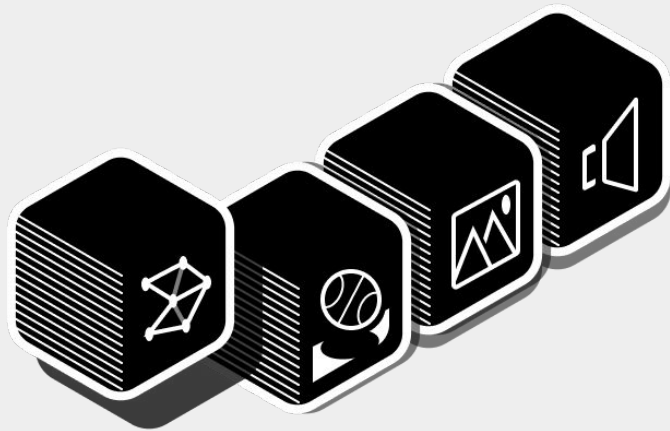


Sentis Intro



기존 딥러닝 모델 추론환경

- Front-end 환경과 호환되는 Inference Engine 사용
 - 플랫폼별 추론코드 구현의 제약 (모바일 / 콘솔 / 웹)
- Rest API를 통한 외부 서버에서의 호출
 - 외부 API 사용시 비용 발생 및 데이터 보안의 우려





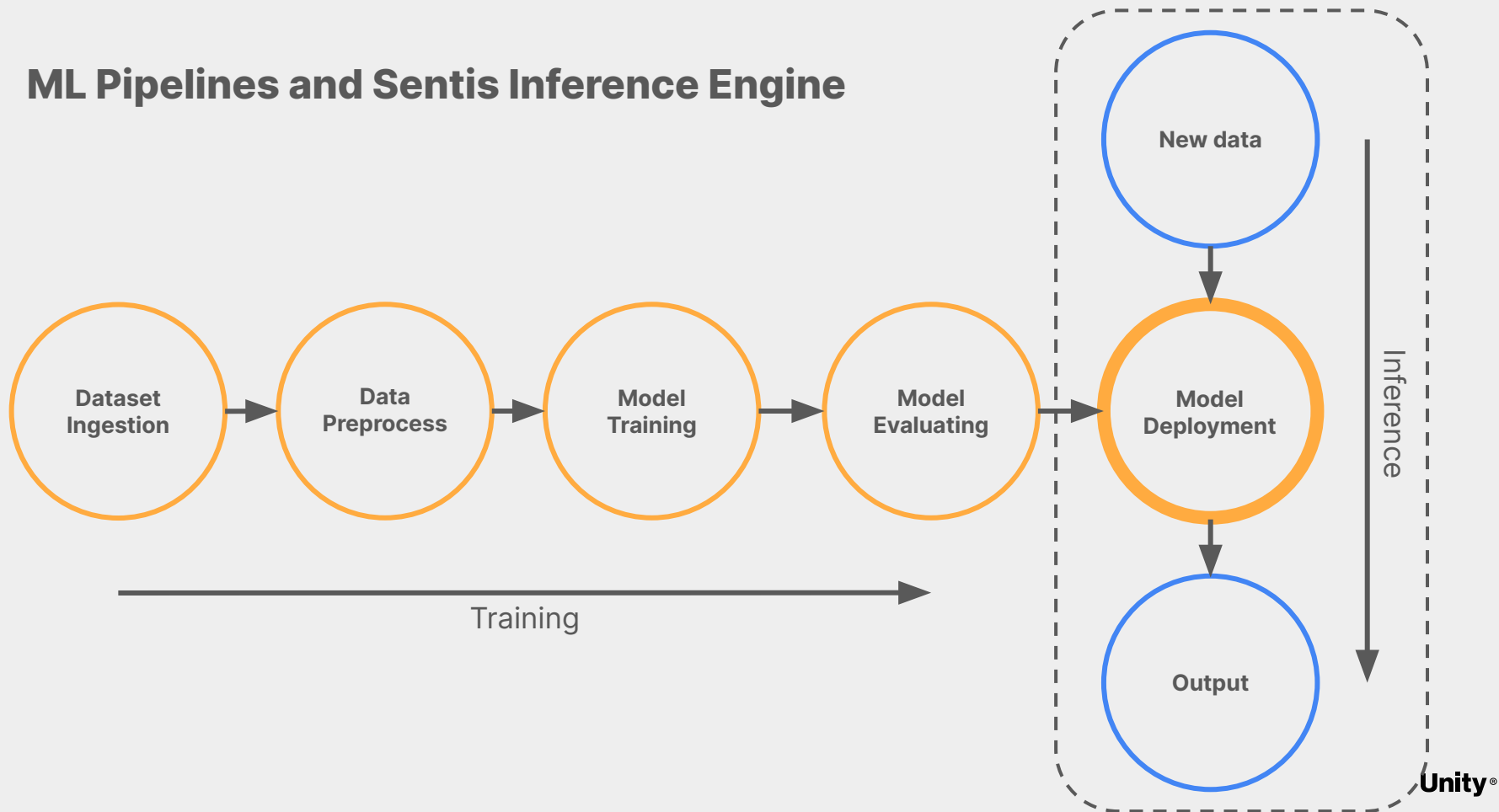
Sentis 가 제공하는 딥러닝 모델 추론환경

- Unity 6의 정식 패키지로 제공
- 호환성이 높은 Open Neural Network Exchange (ONNX) 포맷을 사용
- 빠르고 성능이 뛰어난 온디바이스 인퍼런스 엔진을 제공
- 하나의 코드로 다양한 플랫폼 (데스크탑, 콘솔, 모바일, Web)에 배포 가능
- Compute Shader, Compute Buffer 사용 가능
- 다양한 Pre-trained 모델과 C# 샘플 코드 제공 (Hugging Face)
- (To-be) PyTorch에서 Sentis로 더 쉽게 Export 가능 (ExecuTorch)
- (To-be) Neural Processing Units (NPU) 지원
(Microsoft Direct ML, Apple Core ML/MPS Graph, Google NN API)





ML Pipelines and Sentis Inference Engine



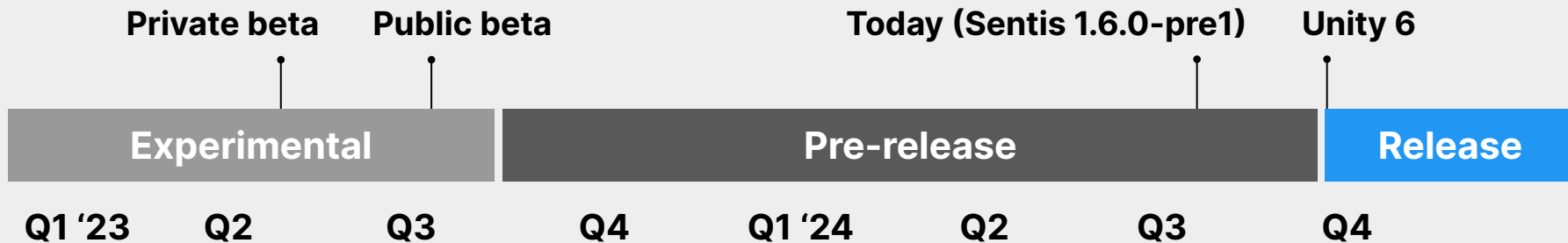


Sentis 로 할 수 있는 Tasks

Image segmentation	Image classification	Object detection
Hand gesture detection	Handwriting detection	Depth estimation
Image generation	Mask generation	Story generation
Text Summarization	Sentence similarity	Token classification
Text-to-speech	Audio-to-audio	Audio classification
Speech-to-text	Sound generation	Board game opponent
Sensor data classification	Zero-shot classification	Time series forecasting

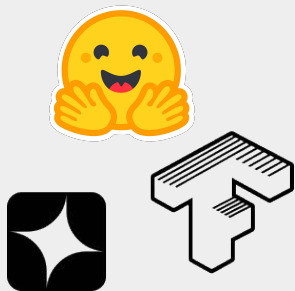


Sentis Roadmap

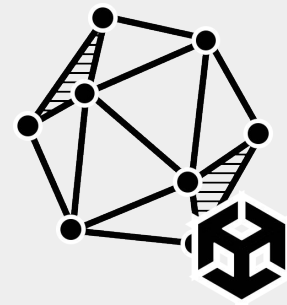




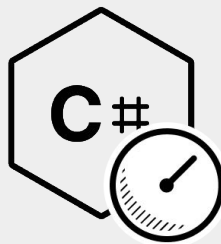
게임에 **Sentis** 를 사용하기



AI 모델 선택



Unity로 импорт 및
모델 최적화



추론 코드작성



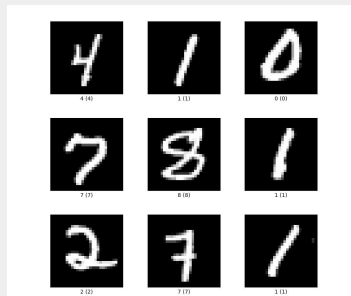
런타임 플랫폼에 배포
(Desktop, Console,
Mobile, WebGL,
WebGPU)



Hugging Face 에서 사용할 수 있는 샘플 소개



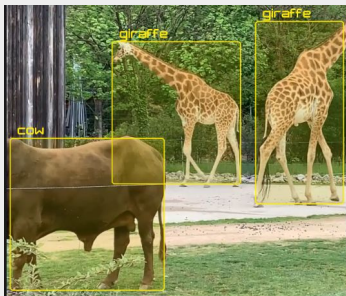
Hugging Face 에서 사용할 수 있는 샘플



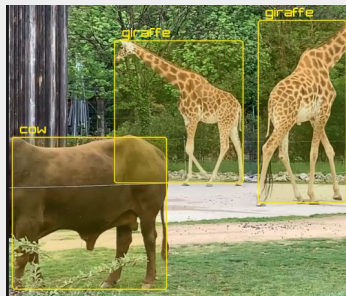
unity/sentis-MNIST-12
digit recognition



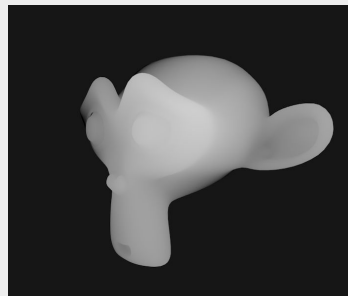
unity/sentis-mobilenet-v2
image classification



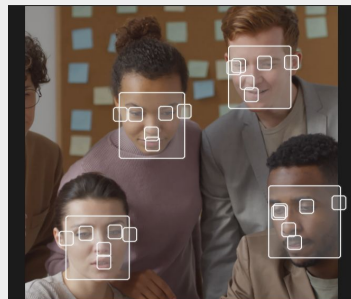
unity/sentis-yolotinyv7
real-time multi-object recognition



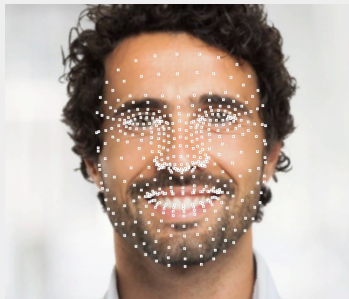
unity/sentis-YOLOv8n
real-time multi-object recognition



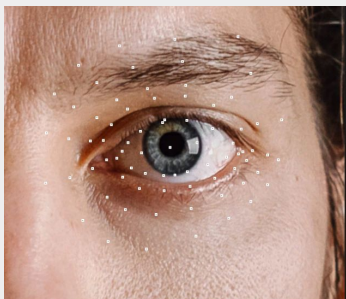
unity/sentis-MiDaS
depth estimation



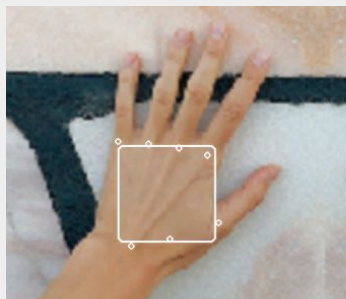
unity/sentis-blaze-face
face detector



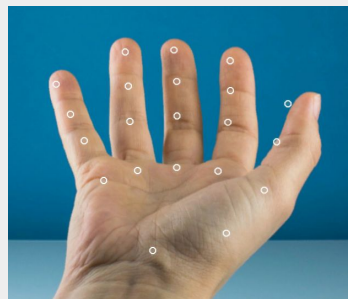
unity/sentis-face-landmarks
face landmark detection



unity/sentis-iris-landmark
iris landmark detection



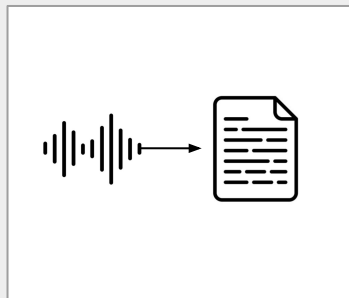
unity/sentis-blaze-palm
hand detection



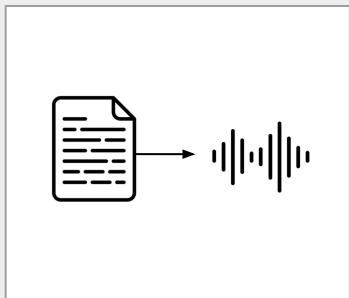
unity/sentis-hand-landmark
hand landmark detection



Hugging Face 에서 사용할 수 있는 샘플



unity/sentis-whisper-tiny
speech to text

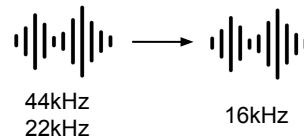


unity/sentis-jets-text-to-speech
text to speech

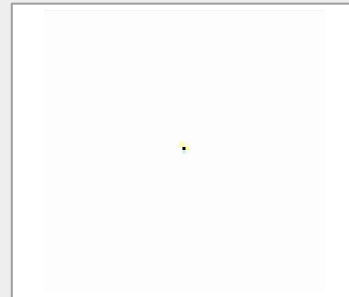
An 80s driving pop song with heavy drums and synth pads in the background



unity/sentis-MusicGen
text to audio



unity/sentis-audio-frequency-to-16khz
audio frequency converter



unity/sentis-neural-cellular-automata
Neural Cellular Automata

One day an alien came down from Mars. It saw a chicken and said, "Hello, little chicken. What are you doing here?" The chicken replied, "I'm looking for a place to stay. I'm very tired." The alien said, "You can stay here. I have a nice place for you. It's very comfortable."

unity/sentis-tiny-stories
text generation

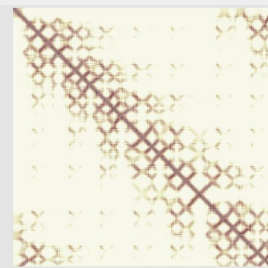
string1 = "That is a happy person"
string2 = "That is a happy dog"

Similarity Score: 0.6945773

unity/sentis-MiniLM-v6
sentence similarity

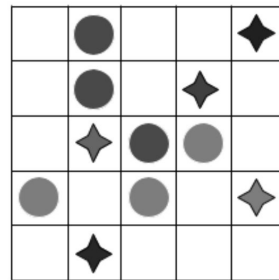
Once upon a time, there were three friends named Alice, Bob, and Carol. They were all passionate about mathematics and loved solving complex problems together. One day, they came across a challenging problem that required them to find the area of a triangle using the Pythagorean theorem.

unity/sentis-phi-1_5
text generation



Model Output

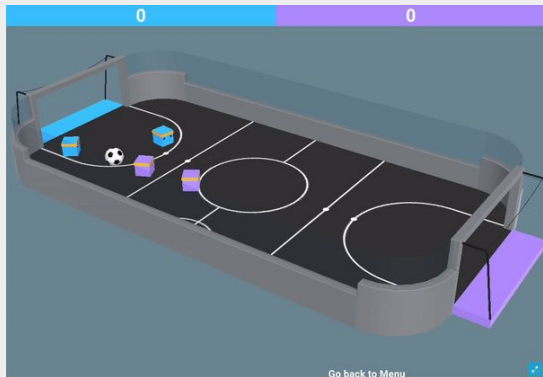
unity/sentis-alphafold-v1
AlphaFoldv1 model



unity/sentis-othello
Othello game model (AlphaZero)



ML-Agents Space



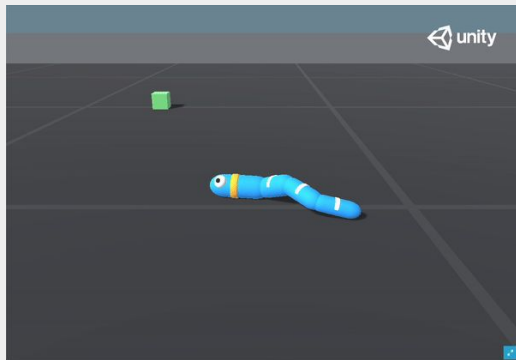
SoccerTwos



ML Agents Pyramids



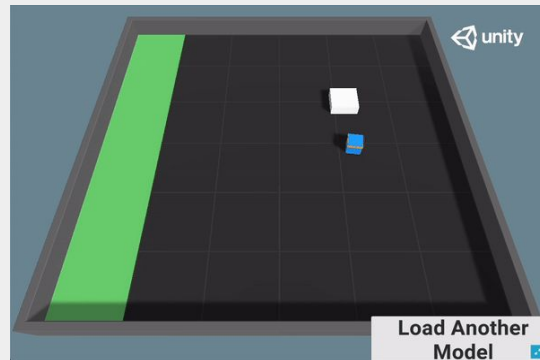
Huggy



ML Agents Worm



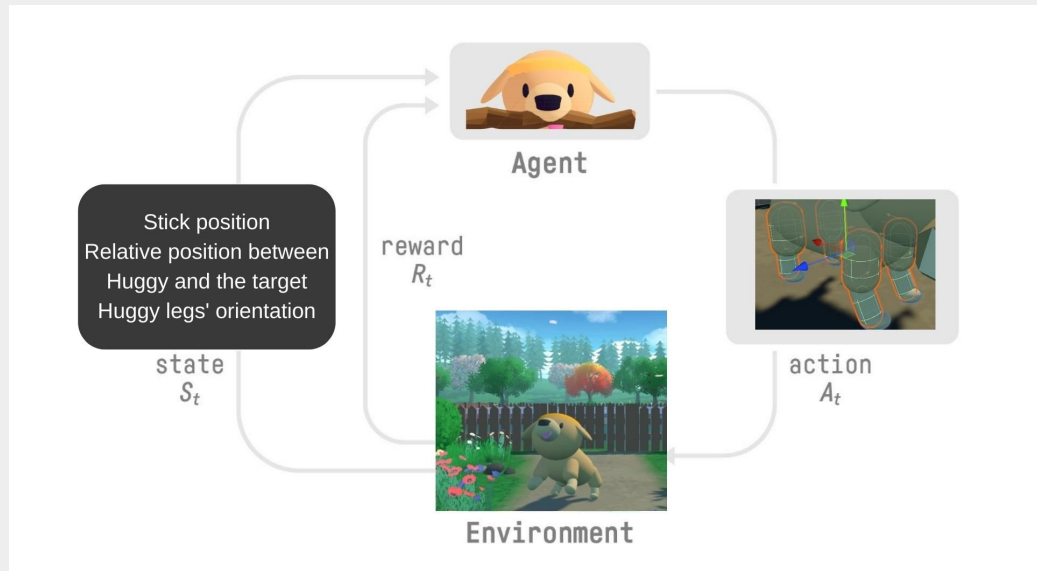
ML Agents Walker



ML Agents Push Block



Tutorial: How Huggy works?





Hugging Face

샘플들을 활용해서 간단한 예제 만들기

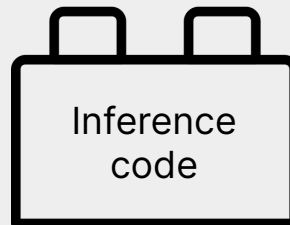
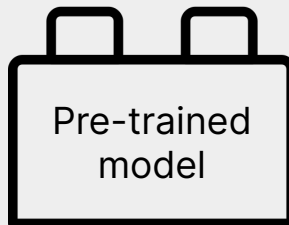


Hugging Face

샘플을 활용하는 방법

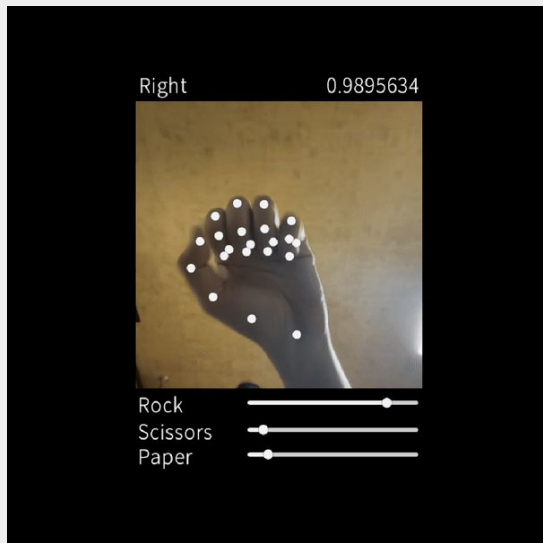
→ Hugging Face 샘플을 활용하는 방법

- 샘플 그대로 활용하기
- Pre-trained model만 변경하여 사용하기
- Inference 코드만 변경하여 사용하기
- Training data 생성용으로 사용하기



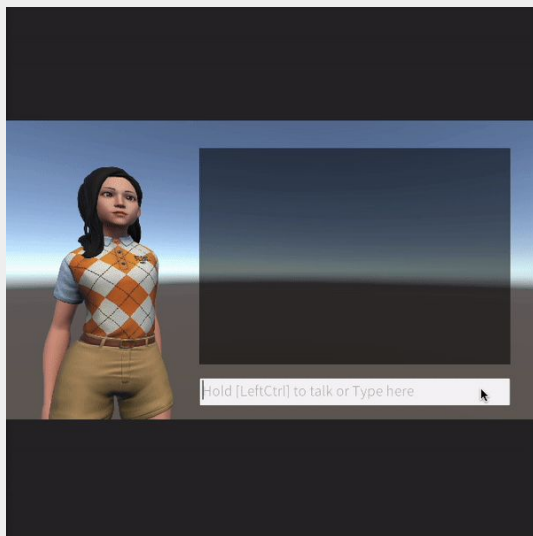


샘플들을 활용하여 간단한 예제를 만들어보기



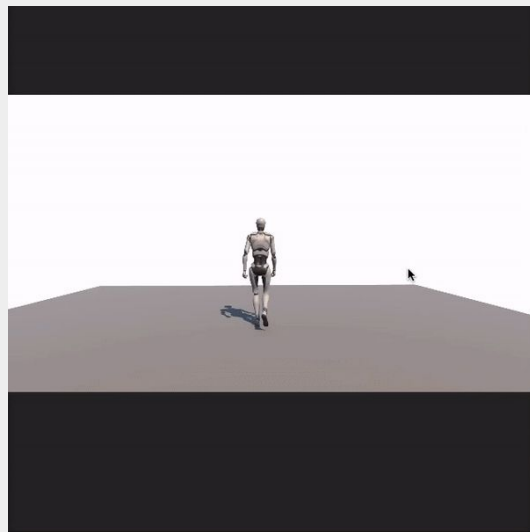
1. 손 제스처 학습을 통한 묵찌빠

- 3D Hand Landmark Recognition Model
- Custom Gesture Classification Model



2. 대화가 가능한 AI NPC 만들기

- Speech to Text (Whisper)
- Sentence Embedding Model (MiniLM)
- LLM (LLaMa3:8B)
- Text To Speech (Jets)



3. AI 갤러리 만들기

- Text To Image: Stable Diffusion XL (Hugging Face Inference API)



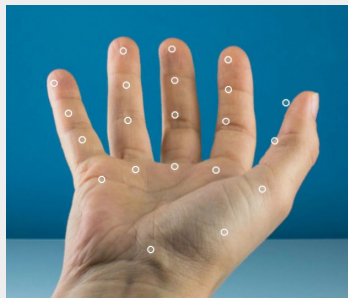
프로젝트 다운로드



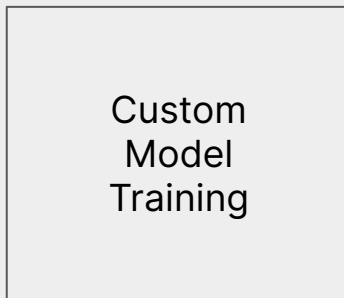
https://github.com/skykim/202407_TechTalk_SentisDemo



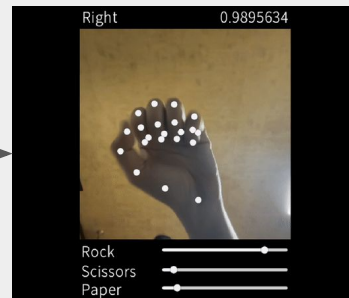
1. 손 제스처 학습을 통한 묵찌빠



unity/sentis-hand-landmark
hand landmark detection



MLP model

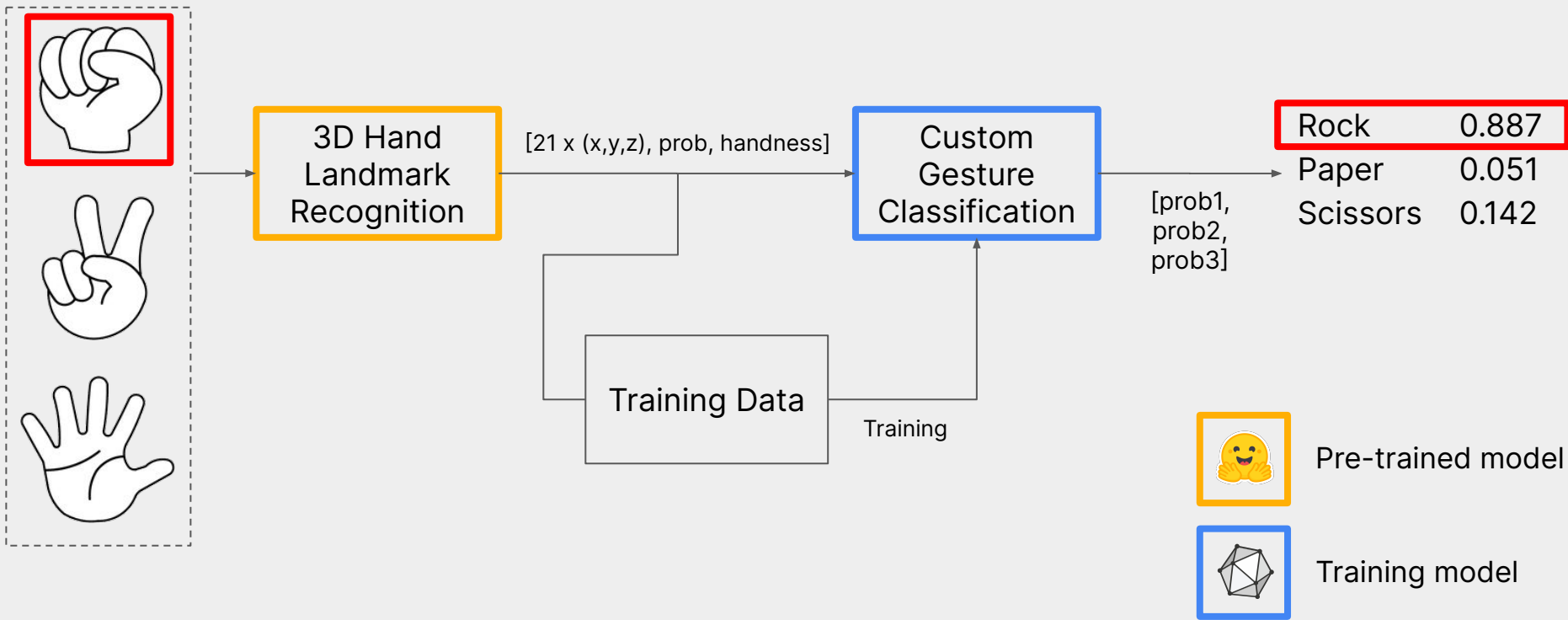


Rock-Paper-Scissors



Gesture Recognition Task

손의 21개의 랜드마크를 인식하고, 묵찌빠의 제스처를 Classification Model을 이용해서 학습하고, 제스처를 인식하기





3D Hand Landmark Detection Model

(HandLandmarkModel.cs)

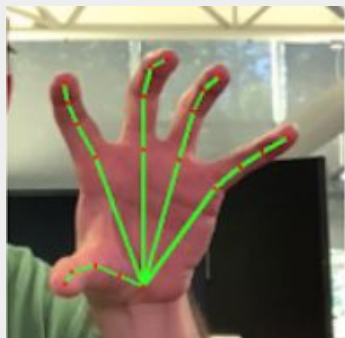


Image [1×3×224×224]

손이 중앙에 위치한 RGB 이미지

hand_landmark.onnx

Identity [1×63]

21개 랜드마크의 x,y,z 좌표

Identity_1 [1×1]

손 인식을 (0~1)

Identity_2 [1×1]

왼손~오른손 (0~1)


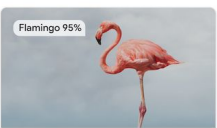


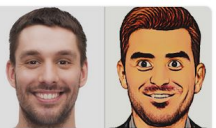
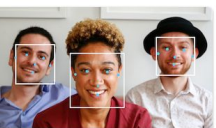
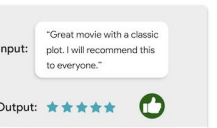





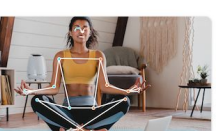
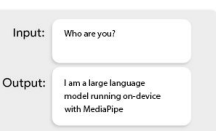
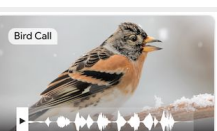
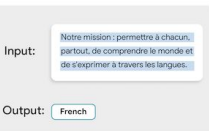


MediaPipe

Google에서 개발한 오픈소스 AI/ML 라이브러리

- Github: <https://github.com/google-ai-edge/mediapipe>
- Demo: <https://mediapipe-studio.webapps.google.com/home>

VISION, TEXT, AUDIO, GENERATIVE AI

 <p>Object Detection Track and label objects in images.</p> <p>See demo</p>	 <p>Image Classification Identify content in images.</p> <p>See demo</p>	 <p>Image Segmentation Locate objects and create image masks with labels.</p> <p>See demo</p>	 <p>Image Embedding Convert images into embedding vectors.</p> <p>See demo</p>	 <p>Face Stylization Stylize faces in an image.</p> <p>See demo</p>	 <p>Face Detection Detect faces in real time.</p> <p>See demo</p>	 <p>Text Classification Classify text into relevant tags.</p> <p>See demo</p>	 <p>Text Embedding Convert text into an embedding vector.</p> <p>See demo</p>
 <p>Interactive Segmentation Segment the object of interest in an image.</p> <p>See demo</p>	 <p>Gesture Recognition Identify and recognize hand gestures.</p> <p>See demo</p>	 <p>Hand Landmark Detection Detect hand landmarks.</p> <p>See demo</p>	 <p>Face Landmark Detection Detect face landmarks and blendshape scores in real time.</p> <p>See demo</p>	 <p>Pose Landmark Detection Identify key points on the body in real time.</p> <p>See demo</p>	 <p>LLM Inference Generate text with LLMs.</p> <p>See demo</p>	 <p>Audio Classification Identify sounds in audio clips.</p> <p>See demo</p>	 <p>Language Detection Identify the language of a given text.</p> <p>See demo</p>



3D Hand Landmark Detection Model: Get a ONNX

tf2onnx

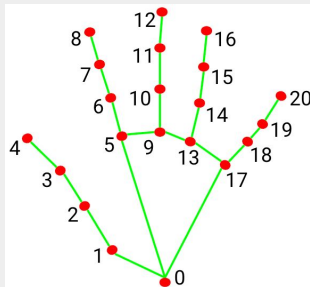
Hand Landmark Model

After the palm detection over the whole image our subsequent hand landmark **model** performs precise keypoint localization of 21 3D hand-knuckle coordinates inside the detected hand regions via regression, that is direct coordinate prediction. The model learns a consistent internal hand pose representation and is robust even to partially visible hands and self-occlusions.

To obtain ground truth data, we have manually annotated ~30K real-world images with 21 3D coordinates, as shown below (we take Z-value from image depth map, if it exists per corresponding coordinate). To better cover the possible hand poses and provide additional supervision on the nature of hand geometry, we also render a high-quality synthetic hand model over various backgrounds and map it to the corresponding 3D coordinates.

```
# install package
!pip install -U tf2onnx
!pip install git+https://github.com/onnx/tensorflow-onnx

# convert to onnx
!python -m tf2onnx.convert --opset 12 --tflite
"hand_landmark_full.tflite" --output "hand_landmark_full.onnx"
```



- | | |
|-----------------------|-----------------------|
| 0. WRIST | 11. MIDDLE_FINGER_DIP |
| 1. THUMB_CMC | 12. MIDDLE_FINGER_TIP |
| 2. THUMB_MCP | 13. RING_FINGER_MCP |
| 3. THUMB_IP | 14. RING_FINGER_PIP |
| 4. THUMB_TIP | 15. RING_FINGER_DIP |
| 5. INDEX_FINGER_MCP | 16. RING_FINGER_TIP |
| 6. INDEX_FINGER_PIP | 17. PINKY_MCP |
| 7. INDEX_FINGER_DIP | 18. PINKY_PIP |
| 8. INDEX_FINGER_TIP | 19. PINKY_DIP |
| 9. MIDDLE_FINGER_MCP | 20. PINKY_TIP |
| 10. MIDDLE_FINGER_PIP | |



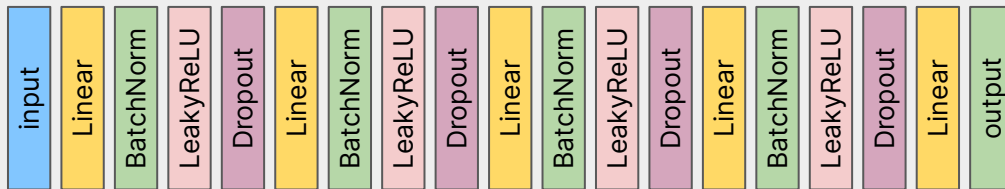
Hand Classifier Model Training

→ Unity에서 트레이닝 데이터 저장 (.csv)

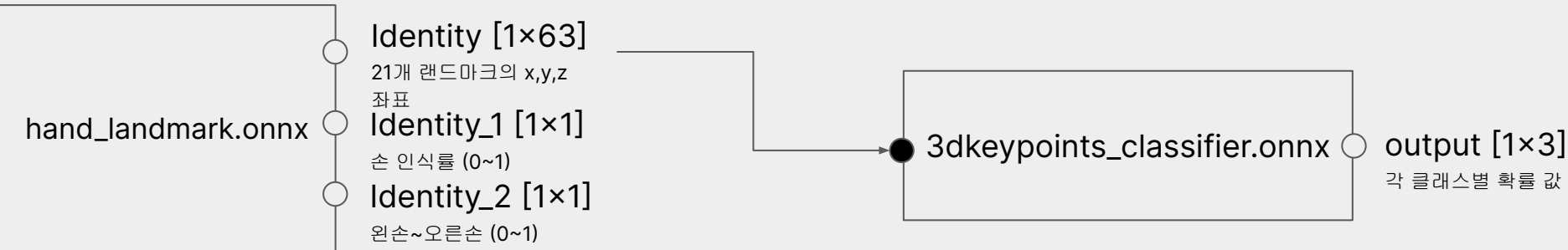
class	p0_x	p0_y	p0_z	p1_x	p1_y	...	p20_z
-------	------	------	------	------	------	-----	-------

...

→ PyTorch 환경에서 MLP 기반 모델 학습



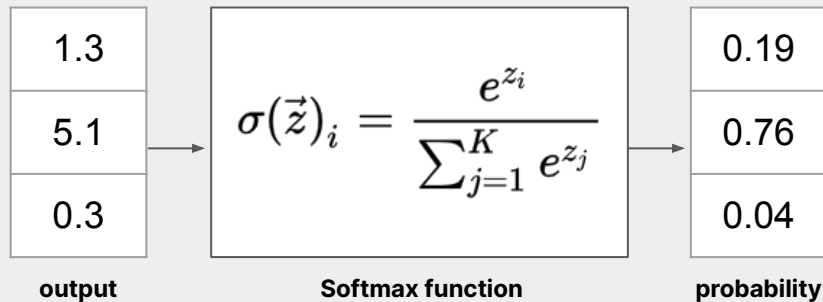
→ 모델 학습 후 ONNX 저장





Hand Classifier SoftMax Layer 추가

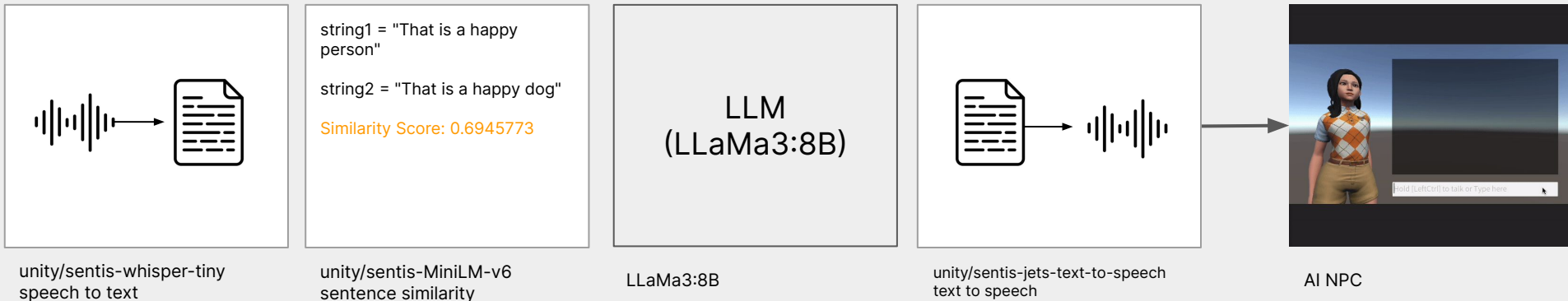
→ Softmax Layer 추가 후 Inference 진행



Data Acquisition



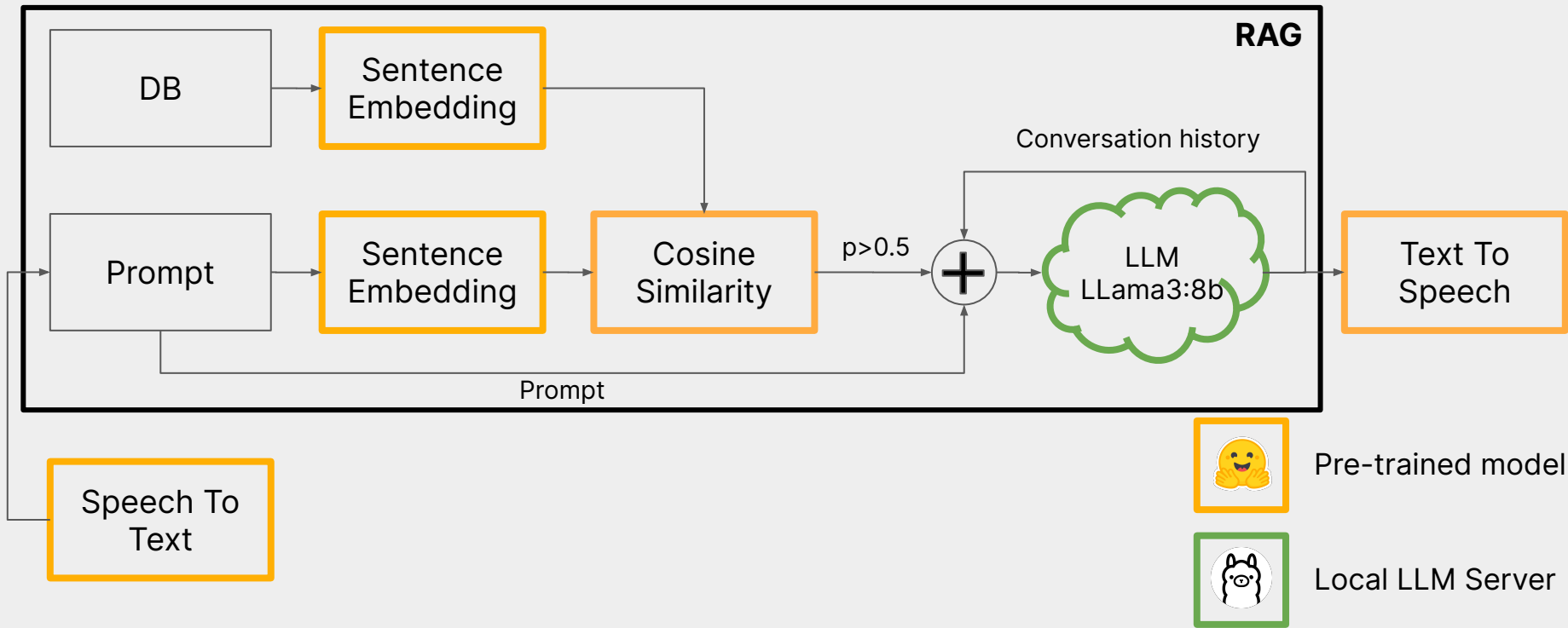
2. 대화가 가능한 **AI NPC** 만들기





AI NPC Dialogue Task

STT를 이용하여 음성을 인식하고, 간단한 RAG를 통해 다음 대화를 생성하여 TTS로 말하기





Why is RAG necessary?

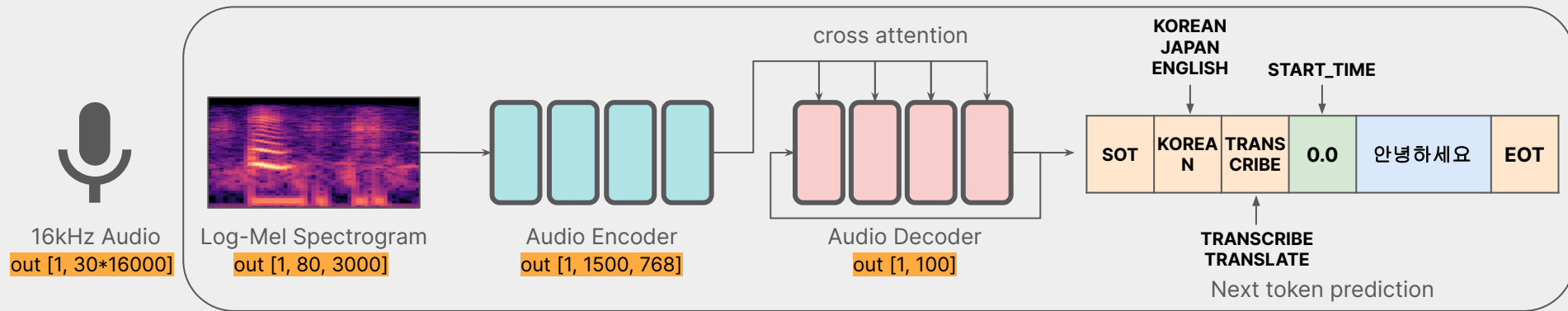
- LLM의 Fine-tuning이 필요하지 않음
- 환각증상 (Hallucination)을 줄여줌
- 실시간 정보를 검색하여 처리가능
- 효율적으로 Prompt 입력 길이를 관리



Speech To Text Model

(WhisperModel.cs)

- Whisper Model (OpenAI, 2022)
 - Log-Mel Spectrogram
 - Transformer Encoder/Decoder
- Size: tiny (39M), base (74M), small (244M), medium (769M), large (1550M)
- Task: 다국어 지원, 번역 기능





Speech To Text Model: Get a ONNX

→ Log-Mel Spectrogram model

- https://colab.research.google.com/drive/1AIH37wtF1WSU6AeZtFy_nG923cSAavmG?usp=sharing

→ Whisper model (Encoder/Decoder)

- <https://colab.research.google.com/drive/1byrBznepFbIn4hRNHRFLIHGXGhXq3nEU?usp=sharing>

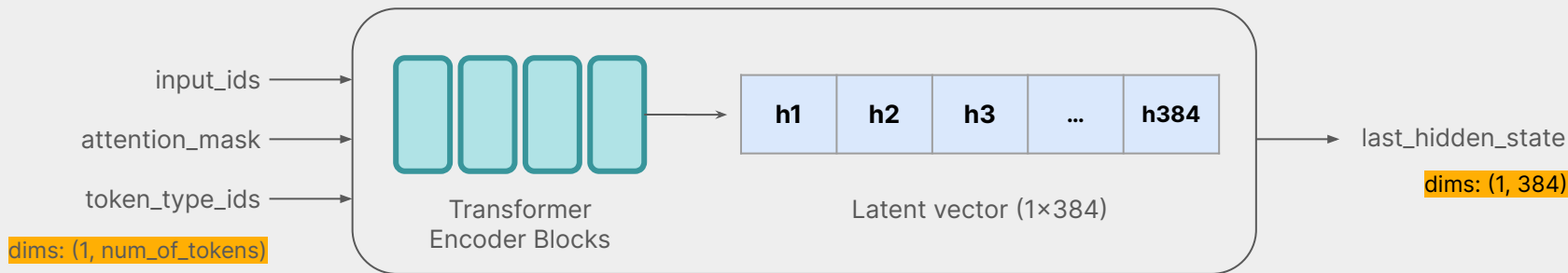
```
#tiny, base, small, medium, large  
model = whisper.load_model("small", device="cpu")
```



Sentence Embedding Model

(MiniLMModel.cs)

- all-MiniLM-L12-v2 (MS, 2021)
 - Transformer (Encoder)
- Distillation 방식으로 학습 진행
- 문장을 토큰 단위로 변환 후, 벡터로 변환
- 해당 모델은 영어만 지원





Sentence Embedding Model : Get a ONNX

→ Sentence Embedding model

- https://colab.research.google.com/drive/1ziKi_6rzW-nGCfcvslKYSzSC-3QwJEw9?usp=sharing

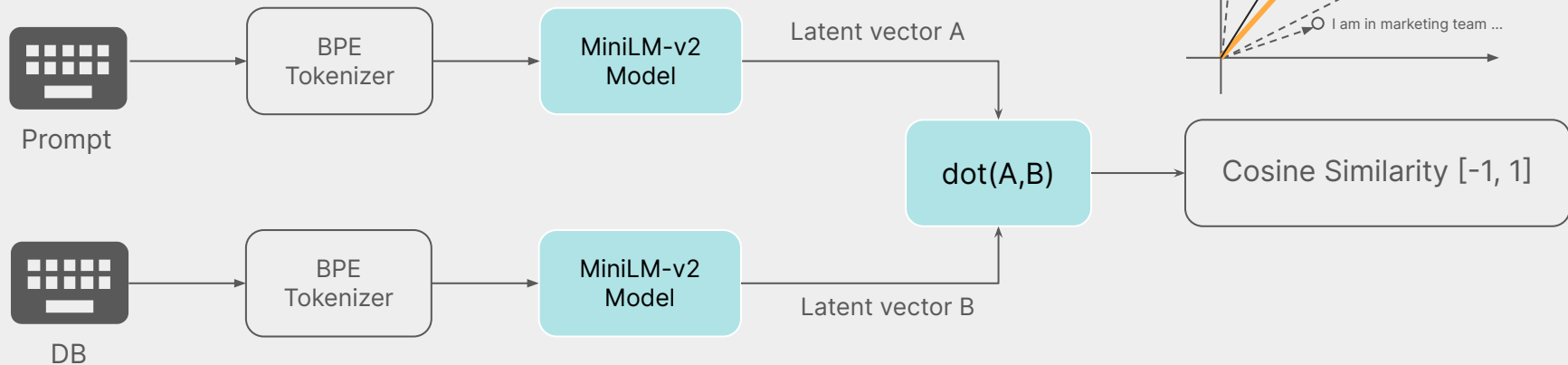
Model Name	Performance Sentence Embeddings (14 Datasets) ⓘ	Performance Semantic Search (6 Datasets) ⓘ	🏆 Avg. Performance ⓘ	Speed ⓘ	Model Size ⓘ
all-mpnet-base-v2 ⓘ	69.57	57.02	63.30	2800	420 MB
multi-qa-mpnet-base-dot-v1 ⓘ	66.76	57.60	62.18	2800	420 MB
all-distilroberta-v1 ⓘ	68.73	50.94	59.84	4000	290 MB
all-MiniLM-L12-v2 ⓘ	68.70	50.82	59.76	7500	120 MB
multi-qa-distilbert-cos-v1 ⓘ	65.98	52.83	59.41	4000	250 MB
all-MiniLM-L6-v2 ⓘ	68.06	49.54	58.80	14200	80 MB
multi-qa-MiniLM-L6-cos-v1 ⓘ	64.33	51.83	58.08	14200	80 MB
paraphrase-multilingual-mpnet-base-v2 ⓘ	65.83	41.68	53.75	2500	970 MB
paraphrase-albert-small-v2 ⓘ	64.46	40.04	52.25	5000	43 MB
paraphrase-multilingual-MiniLM-L12-v2 ⓘ	64.25	39.19	51.72	7500	420 MB
paraphrase-MiniLM-L3-v2 ⓘ	62.29	39.19	50.74	19000	61 MB
distiluse-base-multilingual-cased-v1 ⓘ	61.30	29.87	45.59	4000	480 MB
distiluse-base-multilingual-cased-v2 ⓘ	60.18	27.35	43.77	4000	480 MB



→ Prompt와 DB 간에 Cosine Similarity (dot product)를 계산
($\cos 0^\circ = 1$, $\cos 180^\circ = -1$)

Cosine Similarity

(MiniLMModel.cs)





LLM

(NPCManager.cs)

→ Ollama

- Get up and running with large language models
- <https://ollama.com>
- MIT License

→ Supported models

- LLaMa 3, Phi 3, Mistral, Gemma 2, ...
- LLaMa3:8b (FP16)의 경우 용량이 16GB

→ Localhost Rest API 지원

- <http://localhost:11434/api/generate>



LLM: Prompt

Cosine similarity
=0.5944023

System

You are an Assistant in a game world. Answer the ### Question ### section by referring to the ### Context ### section. Keep your response in character, very brief, and limited to two short sentences at most. Absolutely avoid mentioning that you're an NPC, and respond as if you're truly the person fitting the given role. Do not use any emojis or emoticons.

Assistant Role

Hi there! I'm Park Sojin, a Marketing Specialist in the Marketing Team at Unity Technologies Korea. I'm kind and love gaming, which helps me make good marketing plans that cover both tech and creative parts. I pay close attention to details, get along well with coworkers and clients, and always try my best to do great work. Because of this, I'm seen as a promising team member in our marketing group.

Context

Name and Nationality: My name is Sojin. I am from South Korea.

Conversation History

User: Hi, there!

NPC: Hi! It's great to meet you. How can I assist you today?

Question

User: What is your name?



LLM: Prompt

Question ###
User: What is your name?

→ Context

Cosine similarity

- 0.594 • Name and Nationality: My name is Sojin. I am from South Korea.
- 0.371 • Occupation: work in the marketing department at Unity. My role involves creating marketing strategies and presentations for gaming technologies.
- 0.183 • Professional Skills: I specialize in digital illustration and marketing strategies for the gaming industry. I'm known for my ability to explain complex technical concepts in simple terms and create visually appealing presentations.
- 0.341 • Language Abilities: I'm fluent in English and comfortable using it in professional settings. I also speak Korean and occasionally use Korean expressions, which adds a unique touch to my communication style.
- 0.023 • Hobbies and Interests: I'm passionate about indie games and enjoy creating digital art in my free time. I love reading, especially fantasy novels and marketing books, and I dream of traveling the world someday.

...



Text To Speech Model (JetsModel.cs)

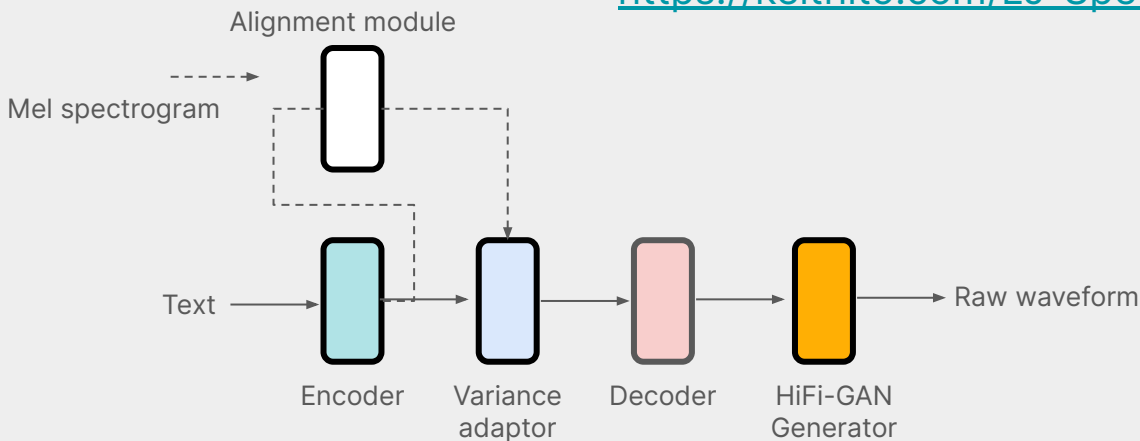
→ JETS: Neural TTS (2022, Kakao)

- Jointly Training FastSpeech2 and HiFi-GAN for End to End Text to Speech

- <https://github.com/imdanboy/jets>

→ Dataset: LJSpeech (미국 여자 목소리)

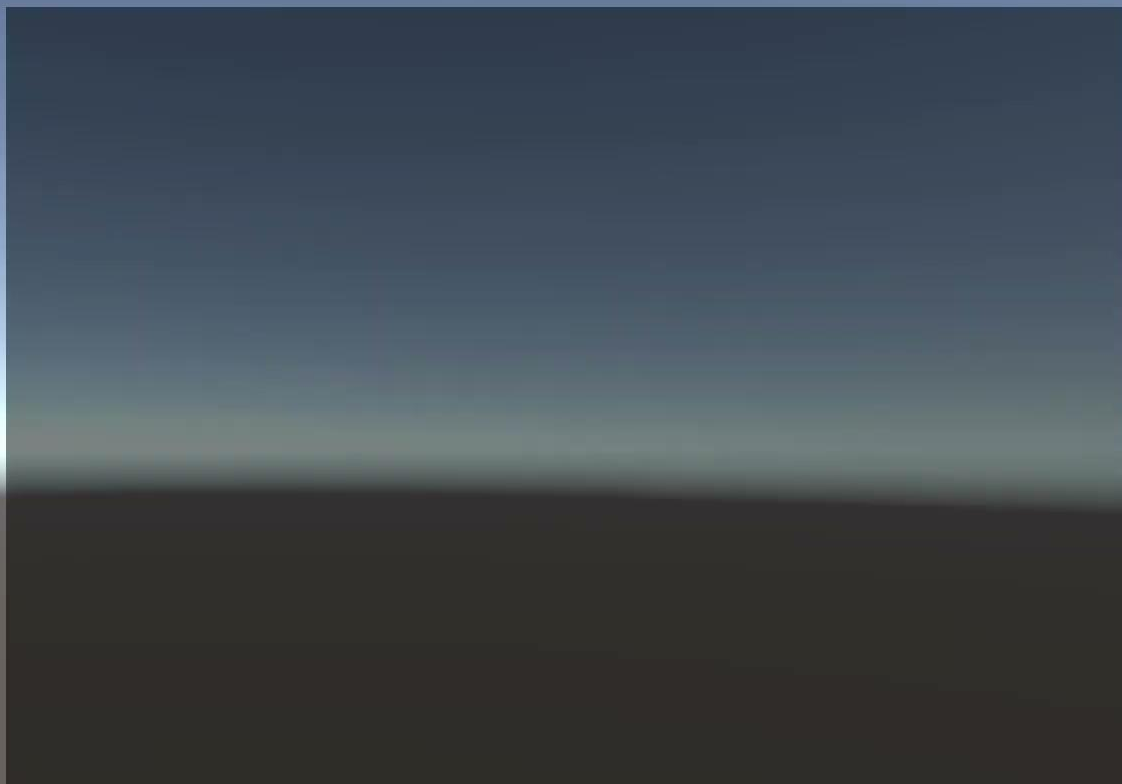
- <https://keithito.com/LJ-Speech-Dataset/>





Text To Speech Model: Get a ONNX

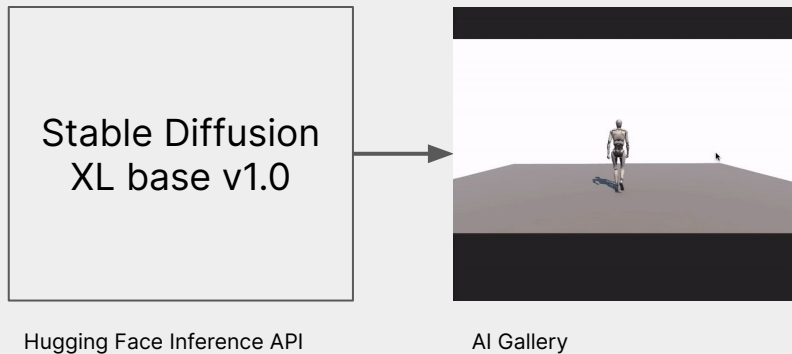
- ESPNet_onnx
 - https://github.com/Masao-Someki/espnet_onnx
- ESPNet Model zoo
 - https://github.com/espnet/espnet_model_zoo/blob/master/espnet_model_zoo/table.csv
- 추가적인 모델 편집 필요 (If operator 제거)



| Hold [LeftCtrl] to talk or Type here



3. AI 갤러리 만들기





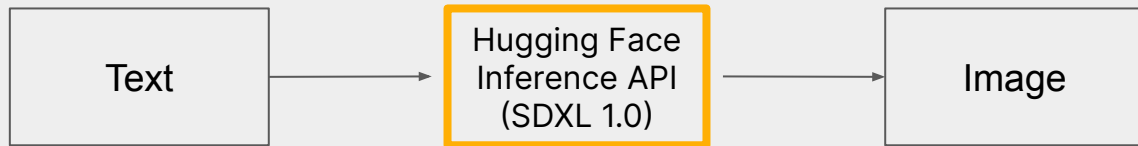
SDXL 1.0-base (AIGalleryScript.cs)

- Stable Diffusion XL (StabilityAI, 2023)
- Improving Latent Diffusion Models for High-Resolution Image Synthesis
 - 모델용량: 약 7GB



Text To Image Task

Stable Diffusion XL 1.0 모델을 Hugging Face Inference API로 호출하여, 텍스트로 갤러리의 AI 작품을 완성시키기



HF Inference API



Hugging Face Inference API 활용하기

- Hugging Face Serverless Inference API
 - Access Token 발급 (API Key)
- Hugging Face API package 설치 (Unity)
 - <https://github.com/Hugging Face/unity-api.git>
- Window > Hugging Face API Wizard >
Install Examples 클릭하여 예제 참고



Hugging Face Inference API Tasks

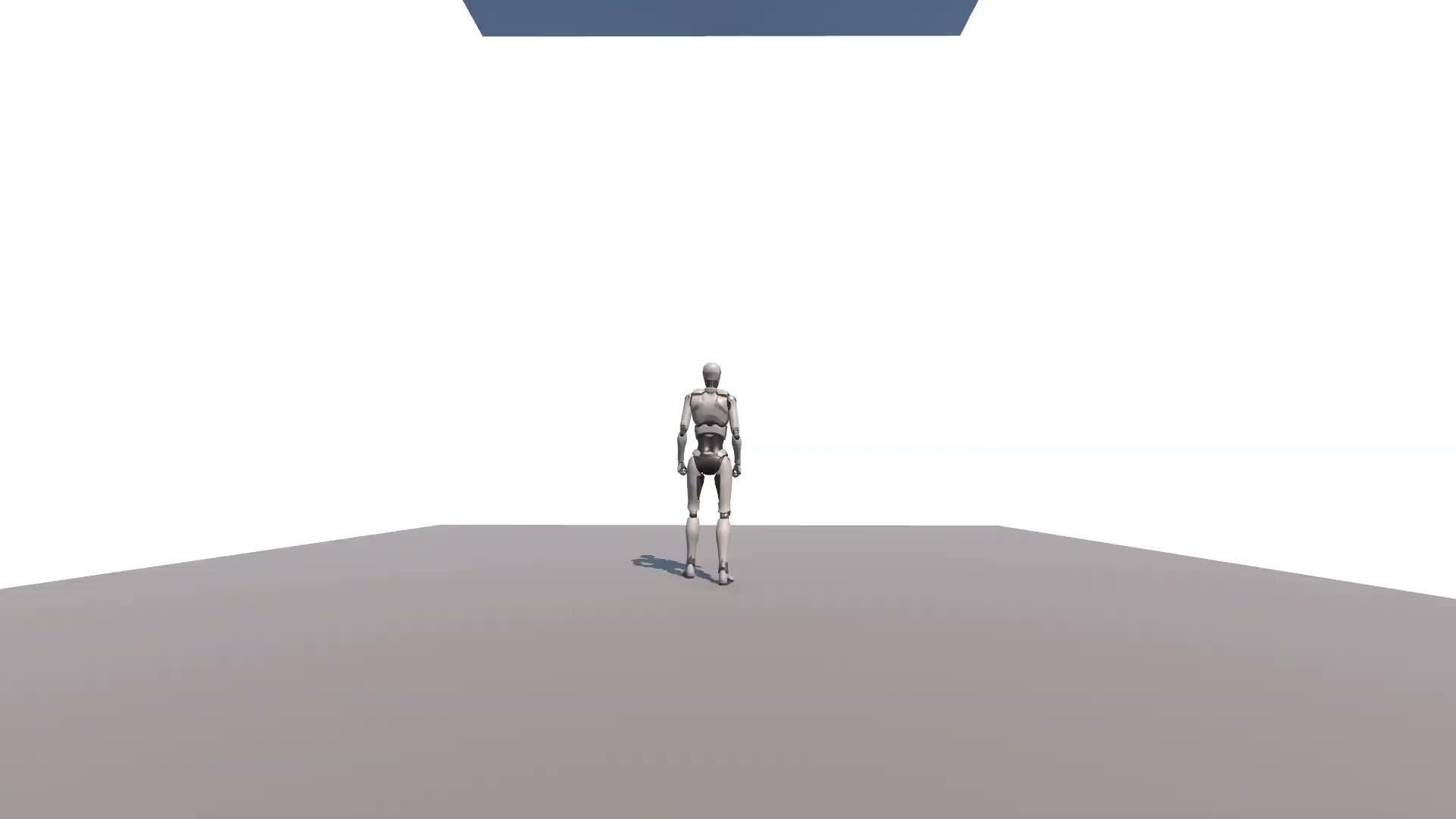
Task	Status
Conversation	✓
Text Generation	✓
Text to Image	✓
Text Classification	✓
Zero Shot Text Classification	✓
Question Answering	✓
Translation	✓
Summarization	✓
Sentence Similarity	✓
Speech Recognition	✓



사용하고자 하는 모델주소로 변경하기

- Task Endpoints > TextToImage > 모델 지정
- <https://api-inference.huggingface.co/models/stabilityai/stable-diffusion-xl-base-1.0>

Task Endpoints	
AutomaticSpeechRecog	https://api-inference.huggingface.co/models/openai/whisper-tiny
Conversation	https://api-inference.huggingface.co/models/facebook/blenderbot-400M-distill
QuestionAnswering	https://api-inference.huggingface.co/models/deepset/roberta-base-squad2
SentenceSimilarity	https://api-inference.huggingface.co/models/sentence-transformers/all-MiniLM-L6-v2
Summarization	https://api-inference.huggingface.co/models/facebook/bart-large-cnn
TextClassification	https://api-inference.huggingface.co/models/distilbert-base-uncased-finetuned-sst-2-english
TextGeneration	https://api-inference.huggingface.co/models/gpt2
TextToImage	https://api-inference.huggingface.co/models/stabilityai/stable-diffusion-xl-base-1.0
Translation	https://api-inference.huggingface.co/models/t5-base
ZeroShotTextClassificat	https://api-inference.huggingface.co/models/facebook/bart-large-mnli
Reset to Defaults	





Recap

- Hugging Face의 샘플들을 활용한다면,
 - 다양한 어플리케이션을 빠르게 제작 가능
 - 비슷한 모델들을 찾아서 프로젝트에 쉽게 도입 가능
- On-Device 처리가 어려운 모델은,
 - Rest API를 제공하는 자체 외부서버 구현
 - Hugging Face Inference API를 사용 가능

Thank you