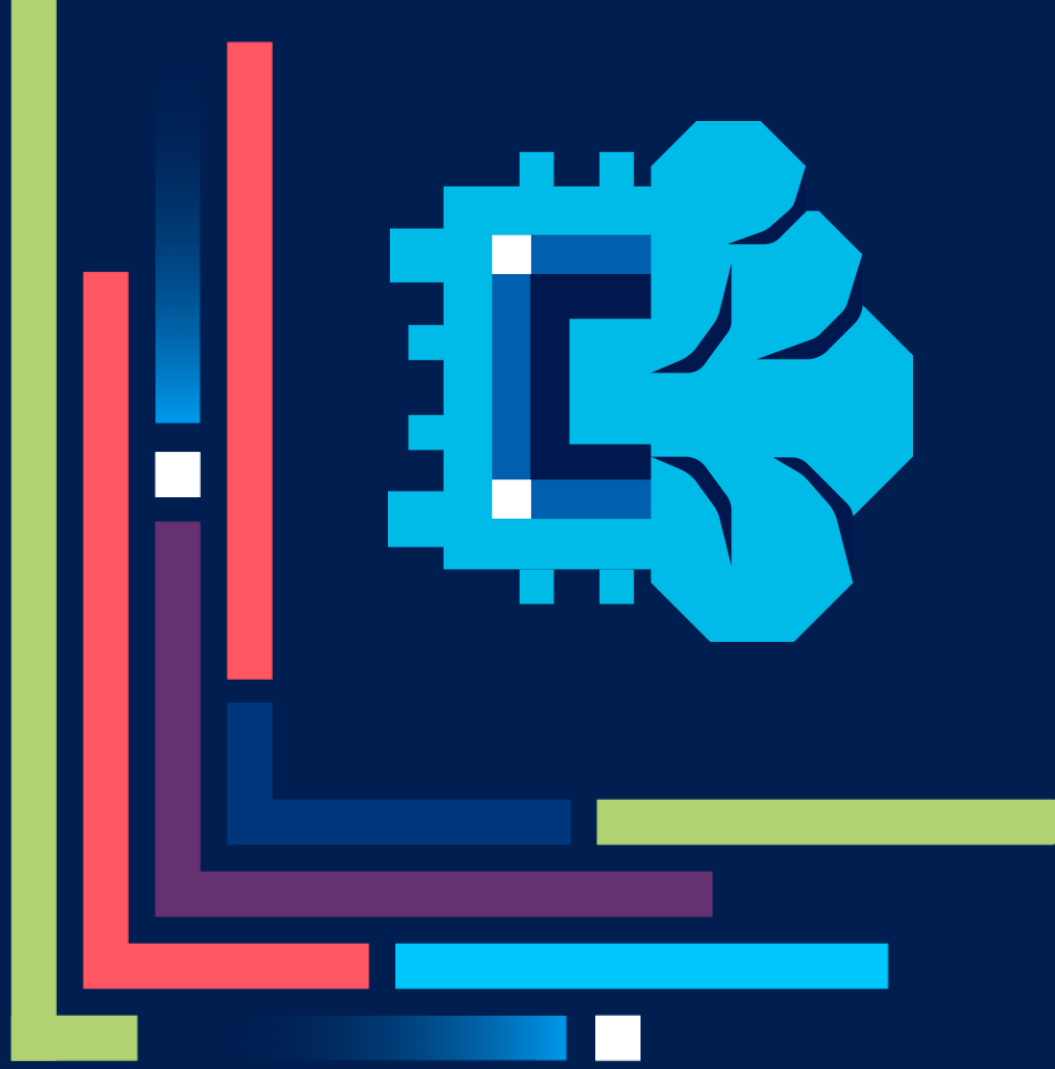
intel ai summit
英特爾 AI 科技論壇

Bringing AI Everywhere



Supercharge Inferencing of Gen AI & LLM on AI PC with OpenVINO™

Zhuo Wu
AI Software Evangelist
March 27th, 2024

Compelling Visual GenAI Use Cases

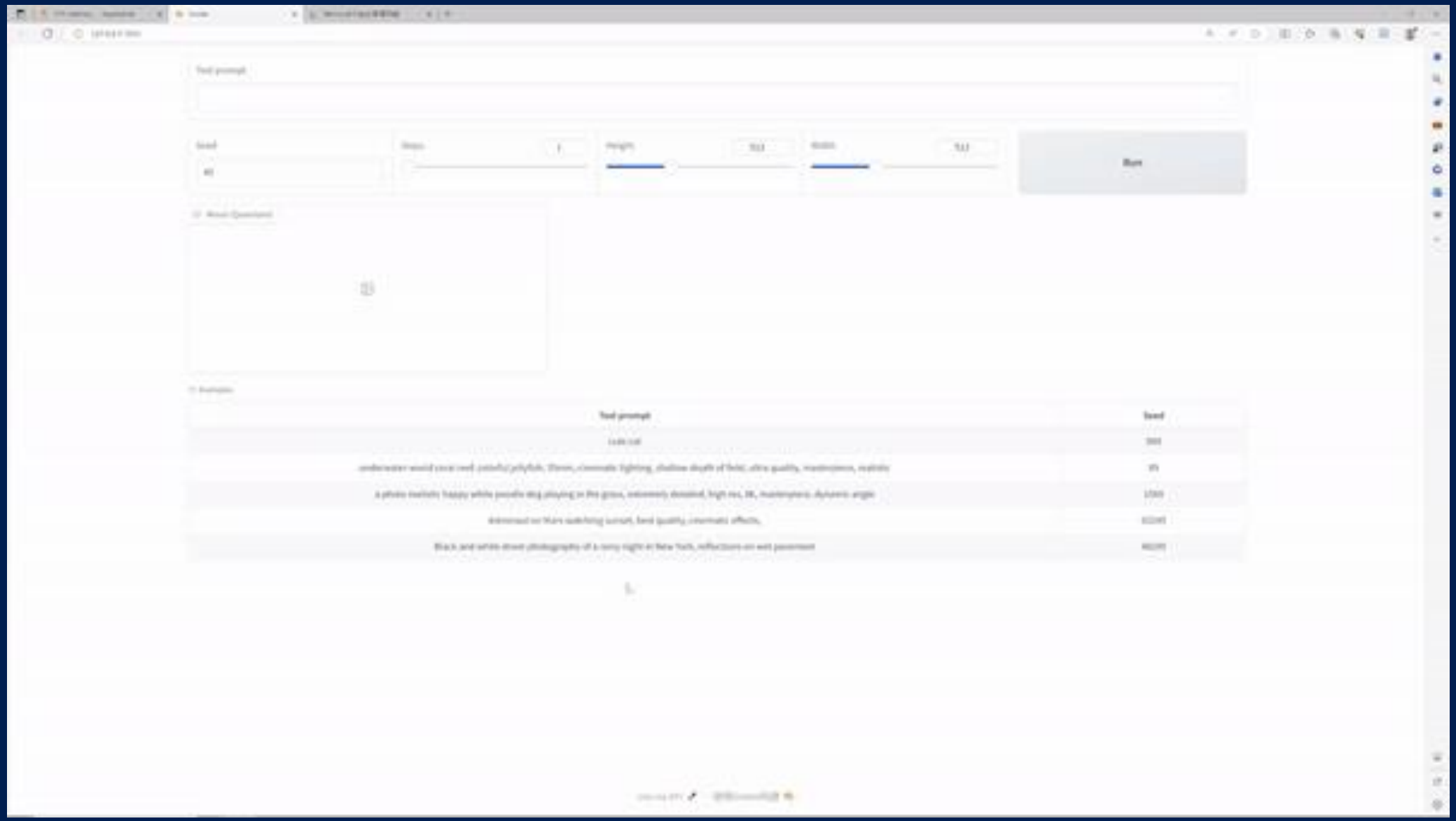
Gaming Experiences

Room Design

Novel Illustration

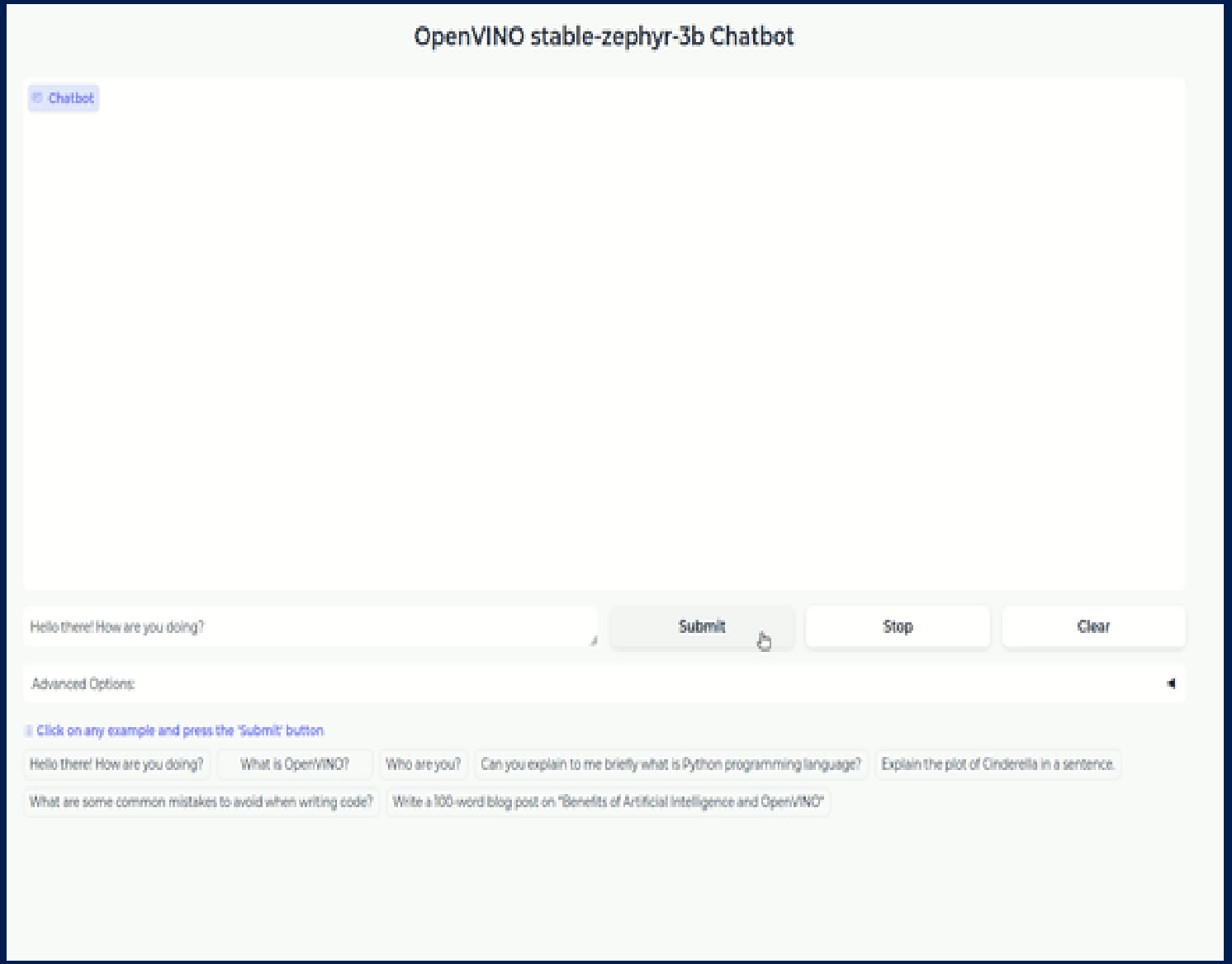
Fashion and E-Commerce

Web Design



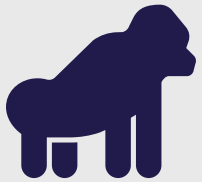
intel Ai summit


OpenVINO™
Large Language Model (LLM)



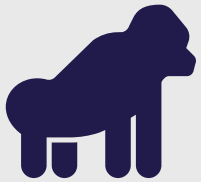

OpenVINO™
GenAI & LLM
Pain Points

Pain Points



Large
model size

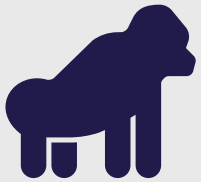
Pain Points



Large
model size

Large memory
footprint

Pain Points

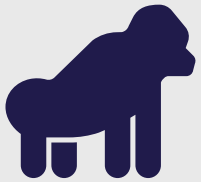


Large
model size

Large memory
footprint

Slow inference
speed

Pain Points



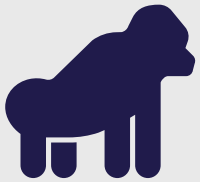
Large
model size

Large memory
footprint

Slow inference
speed

Difficulty training
+ optimizing

Pain Points

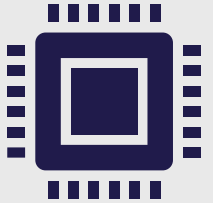


Large model size

Large memory footprint

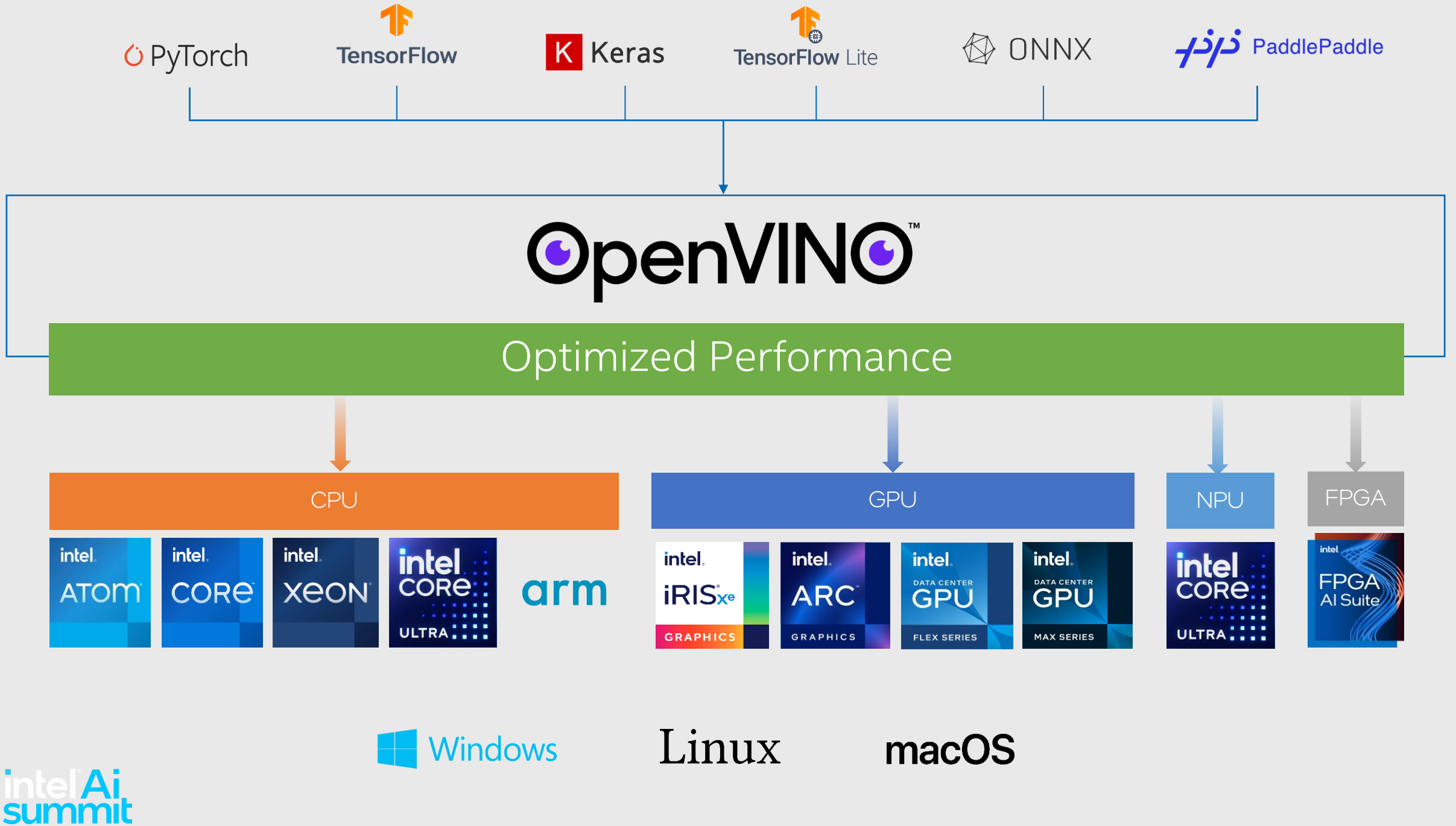
Slow inference speed

Difficulty training + optimizing



No flexibility to run workloads on different HW


OpenVINO™
Deploy and Optimize Visual GenAI with OpenVINO

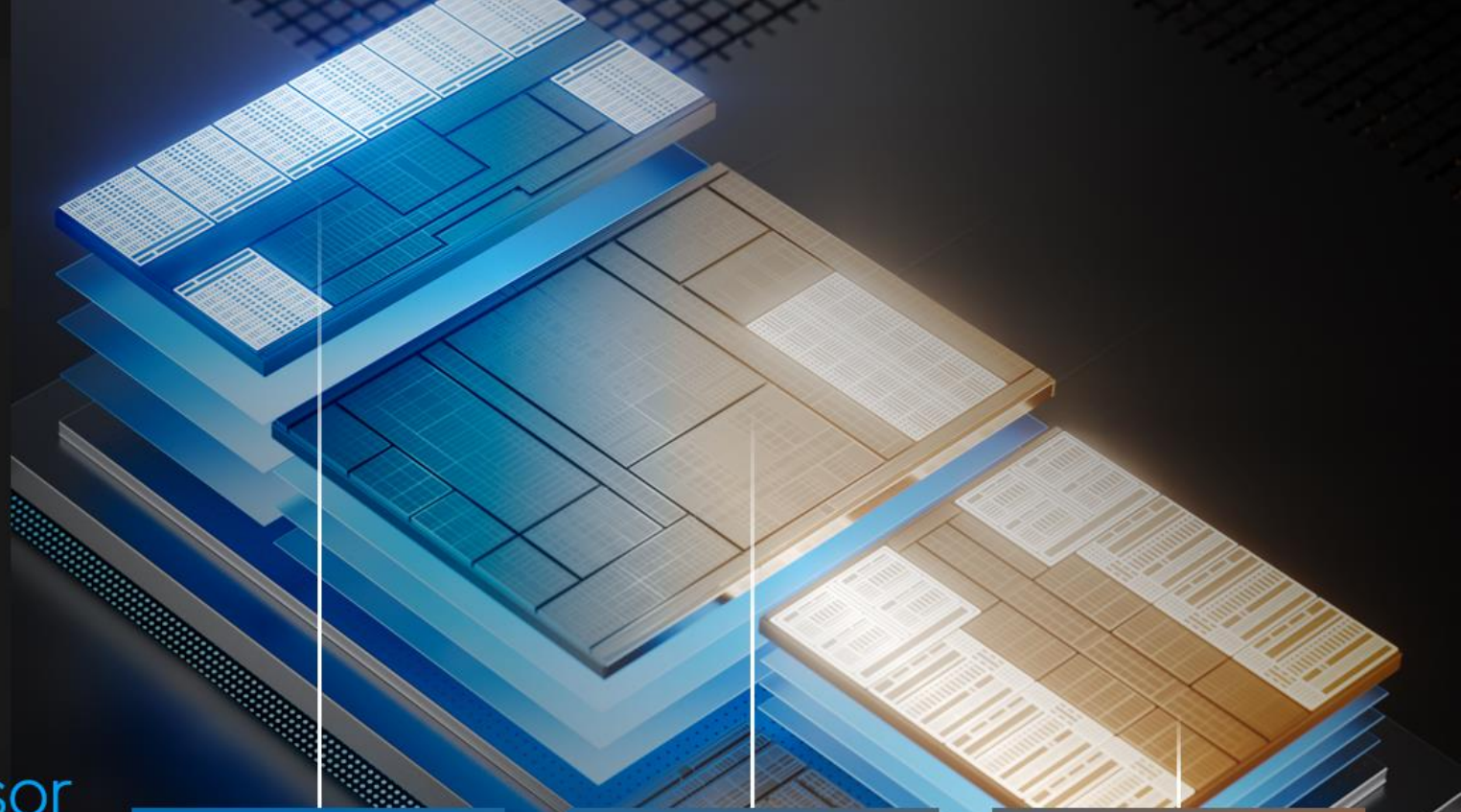

PyTorch
TensorFlow
Keras
TensorFlow Lite
ONNX
PaddlePaddle
OpenVINO™
Optimized Performance
CPU
GPU
NPU
FPGA
intel ATOM
intel CORE
intel XEON
intel CORE ULTRA
arm
intel iRISxe GRAPHICS
intel ARC GRAPHICS
intel DATA CENTER GPU FLEX SERIES
intel DATA CENTER GPU MAX SERIES
intel CORE ULTRA
intel FPGA AI Suite
Windows
Linux
macOS
intel AI summit

Three AI Engines

with Intel® Core™ Ultra Processor

Heterogenous execution of AI workloads embraces the best practices in AI software design

Deliver up to **34 TeraOPS**[1]



GPU

High Throughput
Ideal for AI-accelerated digital content creation

NPU

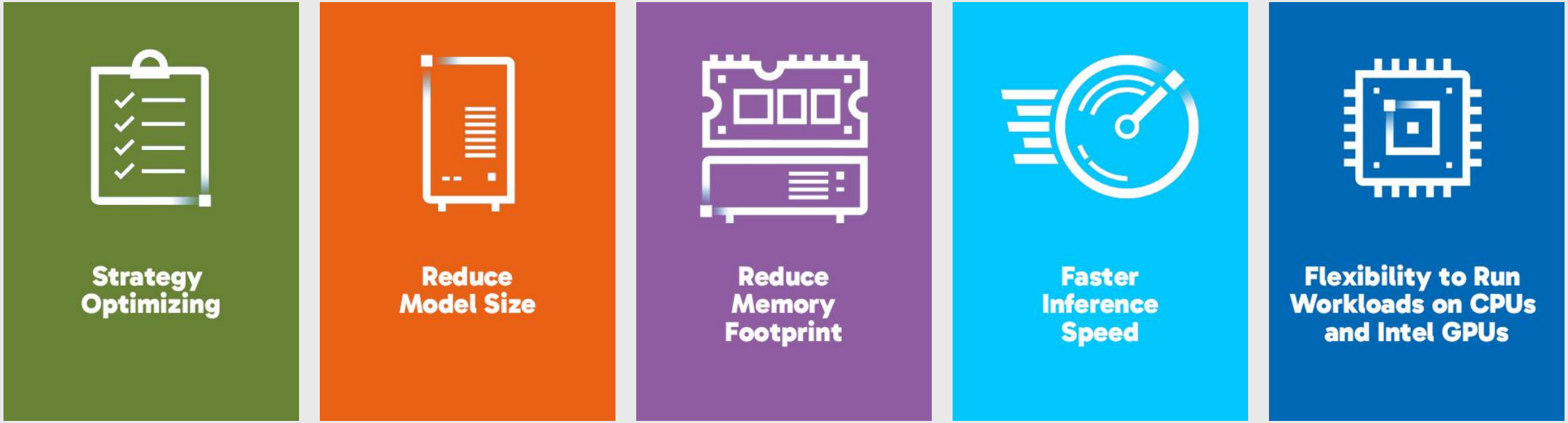
Low Power
Ideal for sustained AI workloads and AI offload for battery life

CPU

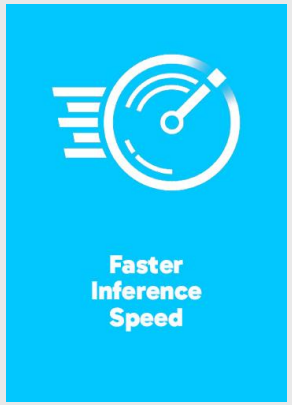
Fast Response
Ideal for low-latency AI workloads

1. Based on Intel® Core™ Ultra 7 165H processor combined TOPS of CPU, GPU, and NPU engines.

Accelerate Generative AI with OpenVINO™

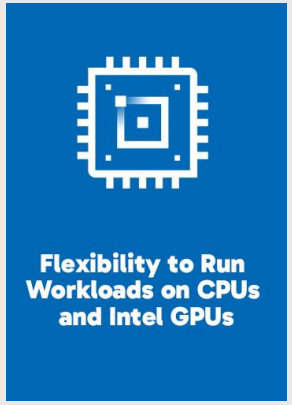


FP16 conversion on the fly for GPU devices



Python

```python
from openvino.tools.mo import convert_model
from openvino.runtime import serialize

ov_model = convert_model ("model.onnx")
serialize(model=ov_model, xml_path="model.xml")
```



Python

```python
from openvino.runtime import Core

core = Core()
text_enc = core.compile_model(TEXT_ENCODER_OV_PATH, 'GPU')
unet_model = core.compile_model(UNET_OV_PATH, 'GPU')
vae_decoder = core.compile_model(VAE_DECODER_OV_PATH, 'GPU')
```

Diffusers

How to use diffusers to build AI-inference solution

```python
import torch
from diffusers import DiffusionPipeline

pipe = DiffusionPipeline.from_pretrained(
          "runwayml/stable-diffusion-v1-5",
          torch_dtype=torch.float16,
          use_safetensors=True
)

pipe = pipe.to("cuda")

prompt = "a photo of an astronaut riding a horse on mars"
pipe.enable_attention_slicing()
image = pipe(prompt).images[0]
```

Library provides pipelines to support many use-cases

Kandinsky 3	text2image, image2image
Latent Consistency Models	text2image
Latent Diffusion	text2image, super-resolution
LDM3D	text2image, text-to-3D, text-to-pano, upscaling
MultiDiffusion	text2image
MusicLDM	text2audio
Paint by Example	inpainting
ParaDiGMS	text2image
Pix2Pix Zeo	image editing

Strategy
Optimizing

Optimum Intel + intel

Python

```
- from diffusers import StableDiffusionPipeline
+ from optimum.intel.openvino import OVStableDiffusionPipeline


model_id = "stabilityai/stable-diffusion-2-1-base"
- pipe = StableDiffusionPipeline.from_pretrained(model_id)
+ pipe = OVStableDiffusionPipeline.from_pretrained(model_id)


pipe.save_pretrained("./stabilityai_cpu")

prompt = "red car in snowy forest"

output_cpu = pipe(prompt, num_inference_steps=17).images[0]
output_cpu.save("image_cpu.png")
```

Reduce Memory Footprint

Optimization of AI-Models

FP32 Native Model	
8.0K	stabilityai_cpu/feature_extractor
1.3G	stabilityai_cpu/text_encoder
8.0K	stabilityai_cpu/scheduler
1.6M	stabilityai_cpu/tokenizer
3.3G	stabilityai_cpu/unet
320 M	stabilityai_cpu/vae
4.9G	stabilityai_cpu/

FP32 OpenVINO™ Model	
8.0K	openvino_ir/feature_extractor
1.3G	openvino_ir /text_encoder
131M	openvino_ir /vae_encoder
8.0K	openvino_ir /scheduler
1.6M	openvino_ir /tokenizer
3.3G	openvino_ir /unet
190M	openvino_ir /vae_decoder
4.9G	openvino_ir /

FP16 OpenVINO™ Model	
8.0K	modelSD21_dGPU_0V/feature_extractor
652M	modelSD21_dGPU_0V/text_encoder
8.0K	modelSD21_dGPU_0V/scheduler
1.6M	modelSD21_dGPU_0V/tokenizer
1.7G	modelSD21_dGPU_0V/unet
96M	modelSD21_dGPU_0V/vae_decoder
2.4G	modelSD21_dGPU_0V/

Reduce Memory Footprint

Faster Inference Speed

Quantization with Optimum Intel and NNCF

Weight Compression

Aims to reduce the memory footprint of a model.

Post-Training Quantization

Transforms the model into a more hardware-friendly representation without retraining or fine-tuning.

Training Aware Quantization

Improves model performance by applying optimizations (such as quantization) during the training.

Image generated after INT8 quantization

intel Ai summit

GenAI Model Workflow with OpenVINO

Optimum-Intel
(base on Transformers and Diffusers)

1 | Convert MODEL	2 | Optimize MODEL	3 | Deploy MODEL	4 | Build PIPELINE
PyTorch Frontend • openvino.model_convert • torch.compile	NNCF • Weight Compression • PTQ • QAT	Runtime(Backend) • C • C++ • Python •	• Text-generation • Text-to-image • ...

263-latent-c... (3) - JupyterLab | 24-nex-openvino-2023-3-lts-la | openvino notebooks github | openvino_notebooks: A coll | GitHub - openvinotoolkit/open

localhost:8888/lab/tree/263-latent-consistency-models-image-generation/263-latent-consistency-models-optimum-demo.ipynb

书签 手机书签

File Edit View Run Kernel Tabs Settings Help

254-llm-chatbot.ipynb | 263-latent-consistency-mc | 287-yolov9-optimization.i

Code | Python 3 (ipykernel)

```
Compiling the unet to GPU ...
Compiling the vae_encoder to GPU ...
Compiling the text_encoder to GPU ...
```

[*]:
```
prompt = "A cute squirrel in the forest, portrait, 8k"

image_ov = ov_pipeline(prompt=prompt, num_inference_steps=4, guidance_scale=8.0).images[0]
image_ov.save("image_opt.png")
image_ov
```

0% 0/4 [00:00<?, ?it/s]

[ ]:

Filter files by name

/ 263-latent-consistency-models-image-generation /

Name

- model
- openvino_ir
- 263-latent-consistency-model...
- 263-latent-consistency-model...
- 263-latent-consistency-model...
- 263-lcm-lora-controlnet.ipynb
- image_opt.png
- image_standard_pipeline.png
- README.md

Simple 0 4 Python 3 (ipykernel) | Busy Mode: Edit Ln 1, Col 1 263-latent-consistency-models-optimum-demo.ipynb 1


OpenVINO™
Visual Gen AI Models and OpenVINO Overview

OpenVINO Notebooks

Stable Diffusion

intel Ai summit

OpenVINO Notebooks

Text-to-Image:
Latent Consistency Model

intel Ai summit

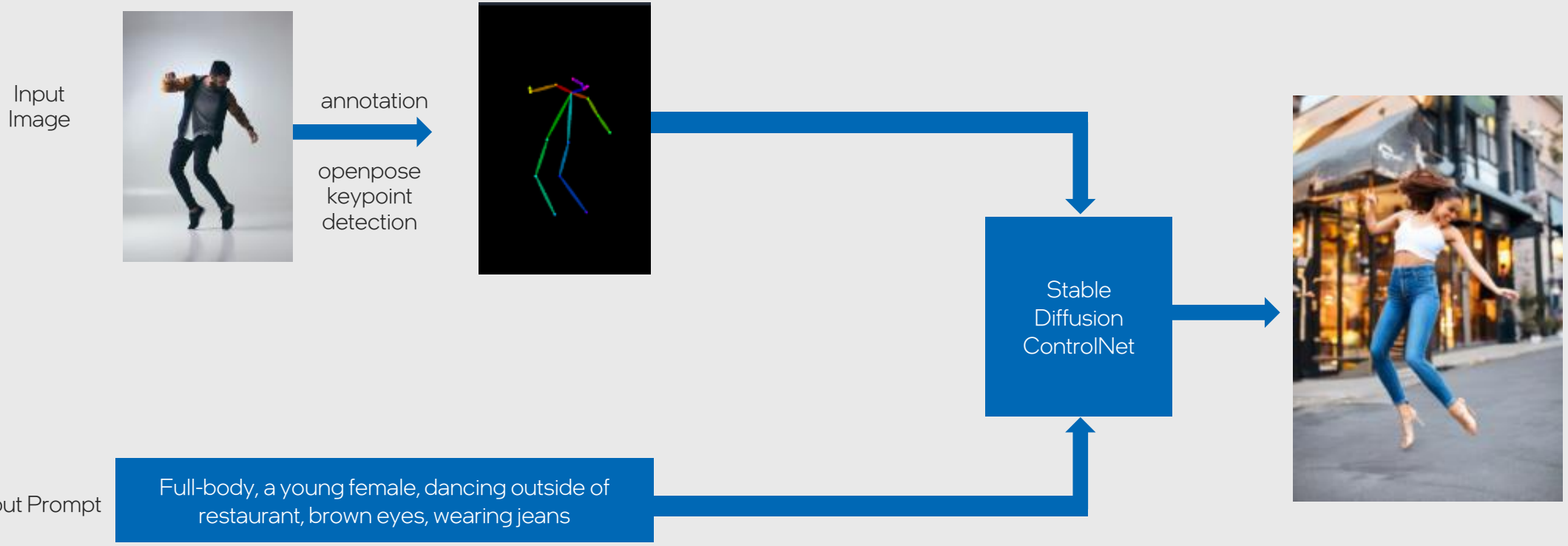
OpenVINO Notebooks

QR-Code Generation:
ControlNet QR Code Monster

intel Ai summit

OpenVINO Notebooks

Text-to-Image:
Stable Diffusion ControlNet Conditioning



intel Ai summit

OpenVINO Notebooks

Image-to-Image: InstructPix2Pix

Original Image	Prompt: Make It In Galaxy

intel Ai summit

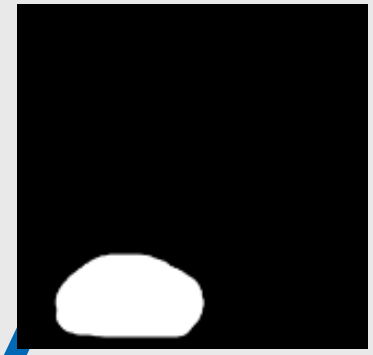
OpenVINO Notebooks

In-Painting: Paint by Example

Input Image

Draw Mask on Image

Extract Mask from Canvas and Invert



Reference Image

Stable Inpainting Pipeline

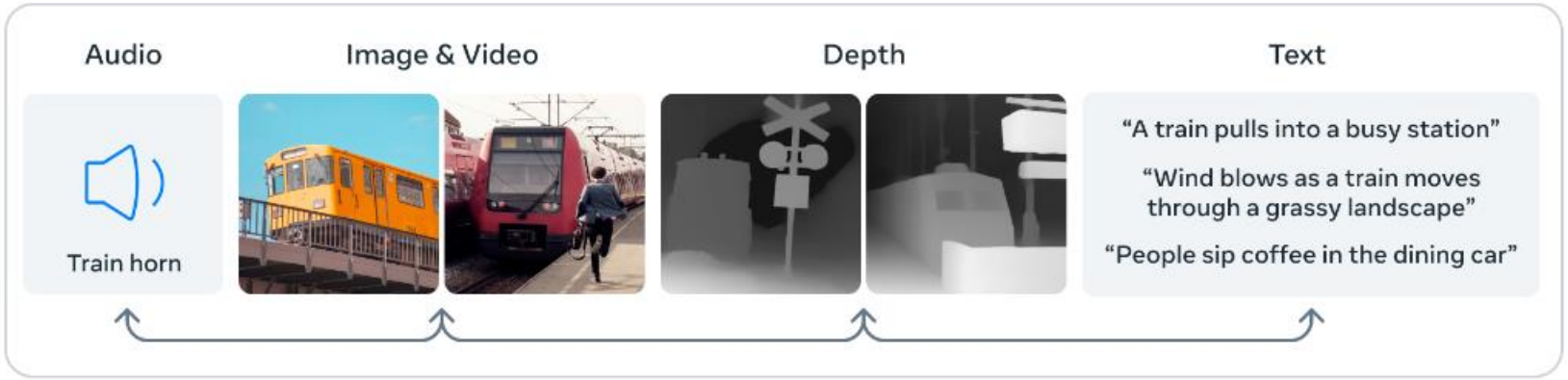
Result Image

intel Ai summit

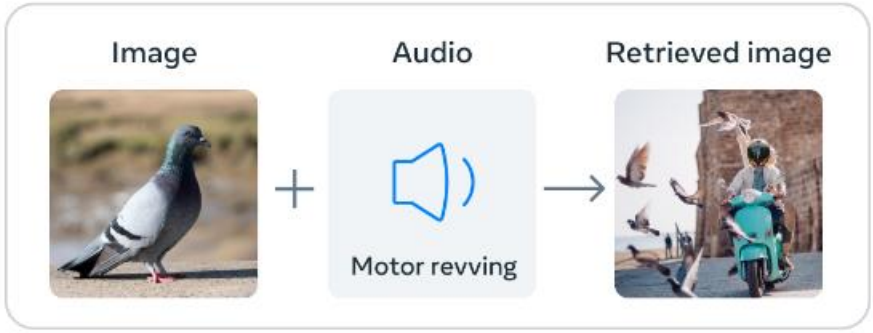
OpenVINO Notebooks

Multimodality: ImageBind

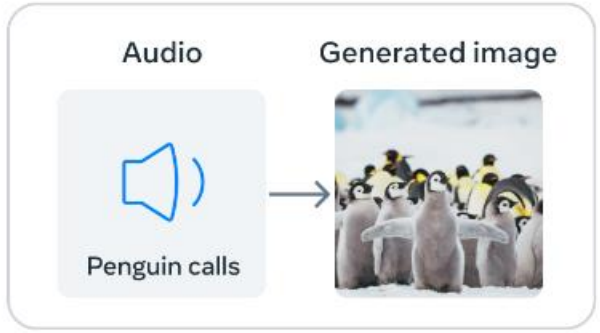
Cross-modal retrieval



Embedding-space arithmetic



Audio to image generation



intel AI summit

OpenVINO Notebooks

Text to Video: ZeroScope

Darth Vader is surfing on waves

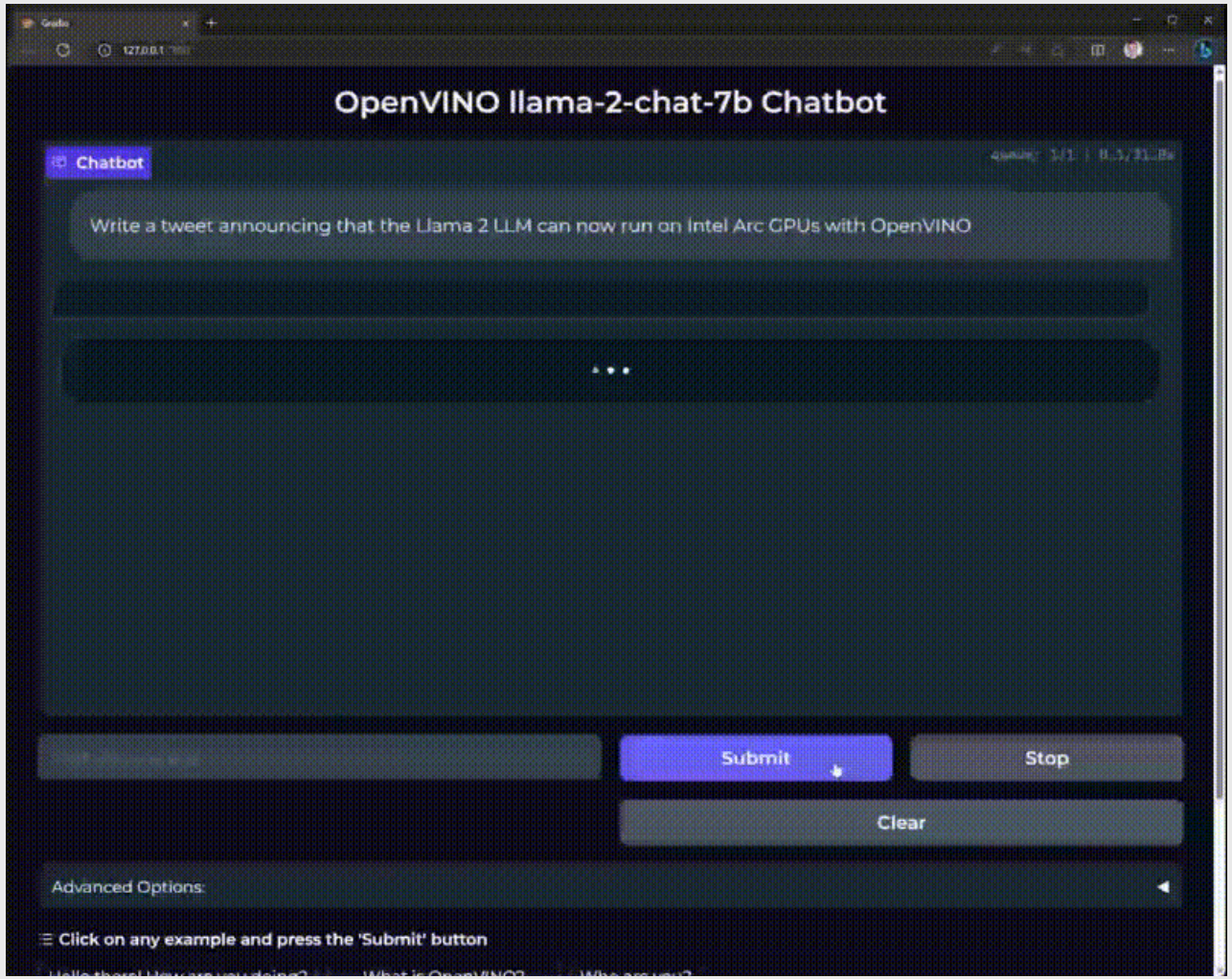
intel Ai summit

OpenVINO™

LLMs with OpenVINO

LLM Use Case

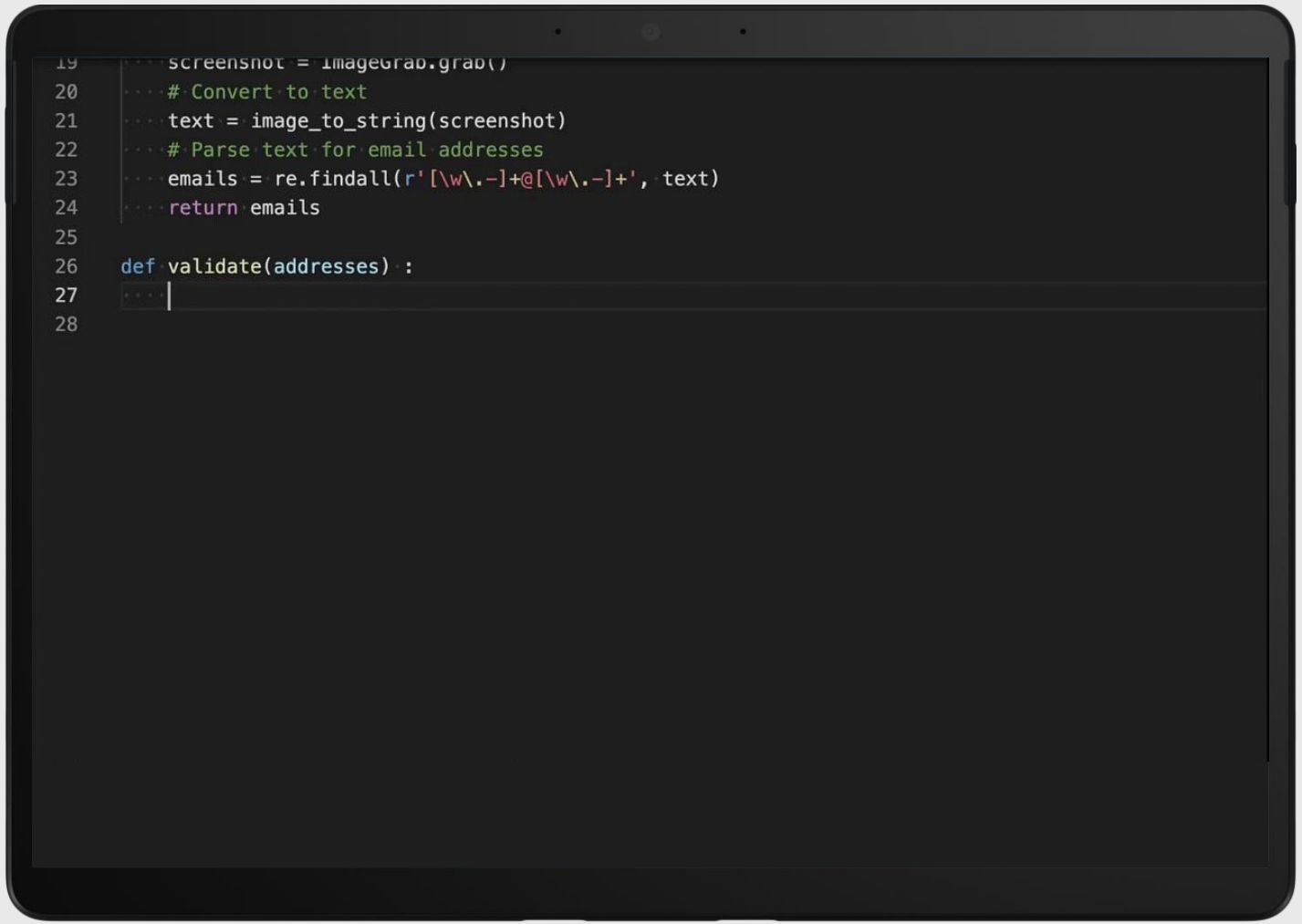
USE CASES
Agent Simulations
Agents
Autonomous Agents
Chatbots
Classification
Code Understanding
Code Writing
Evaluation
Extraction
Interacting with APIs
Multi-Modal
QA Over Documents
Self-Checking
SQL
Summarization
Tagging



intel Ai summit

LLM Use Case

USE CASES
Agent Simulations
Agents
Autonomous Agents
Chatbots
Classification
Code Understanding
Code Writing
Evaluation
Extraction
Interacting with APIs
Multi-Modal
QA Over Documents
Self-Checking
SQL
Summarization
Tagging

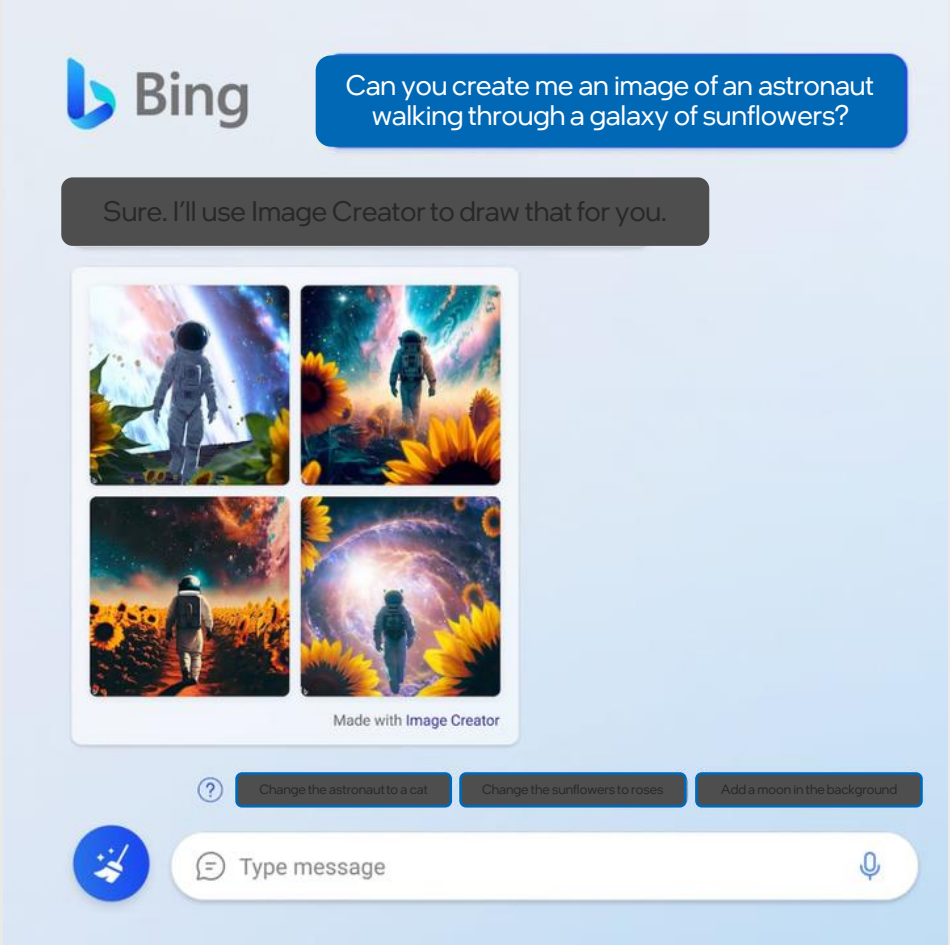


Code Generation

intel Ai summit

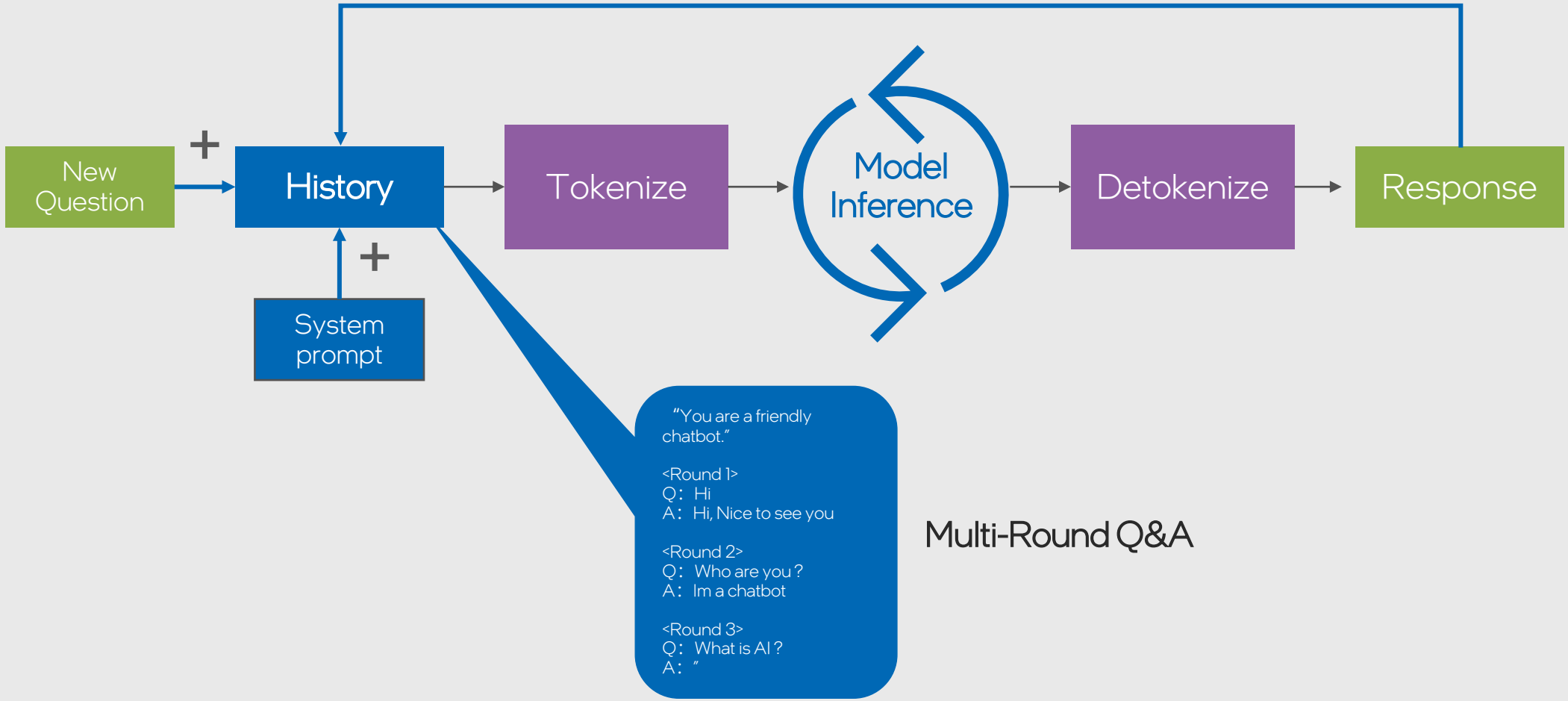
LLM Use Case

USE CASES
Agent Simulations
Agents
Autonomous Agents
Chatbots
Classification
Code Understanding
Code Writing
Evaluation
Extraction
Interacting with APIs
Multi-Modal
QA Over Documents
Self-Checking
SQL
Summarization
Tagging



intel AI summit

Example: How is a Chatbot Working?




OpenVINO™
Deploy LLM
with OpenVINO

LLM-Enabled

Use Cases

Chat Bot | Code Generation | Search | Text Classification | Content Creation | Instruction Following

Example Model Support Includes, But Not Limited to

GPT J | Notus | LLaVa | Llama 2 | BLOOM | chatGLM | Neural Chat | Baichuan | Mixtral
MPT | Dolly | Qwen | Mistral | Zephyr | RedPajama | LLM chatbot | Yi

OpenVINO™ Integration with OptimumAccelerate Transformers

Task	Auto Class
text-classification	OVModelForSequenceClassification
token-classification	OVModelForTokenClassification
question-answering	OVModelForQuestionAnswering
audio-classification	OVModelForAudioClassification
image-classification	OVModelForImageClassification
feature-extraction	OVModelForFeatureExtraction
fill-mask	OVModelForMaskedLM
text-generation	OVModelForCausalLM
text2text-generation	OVModelForSeq2SeqLM

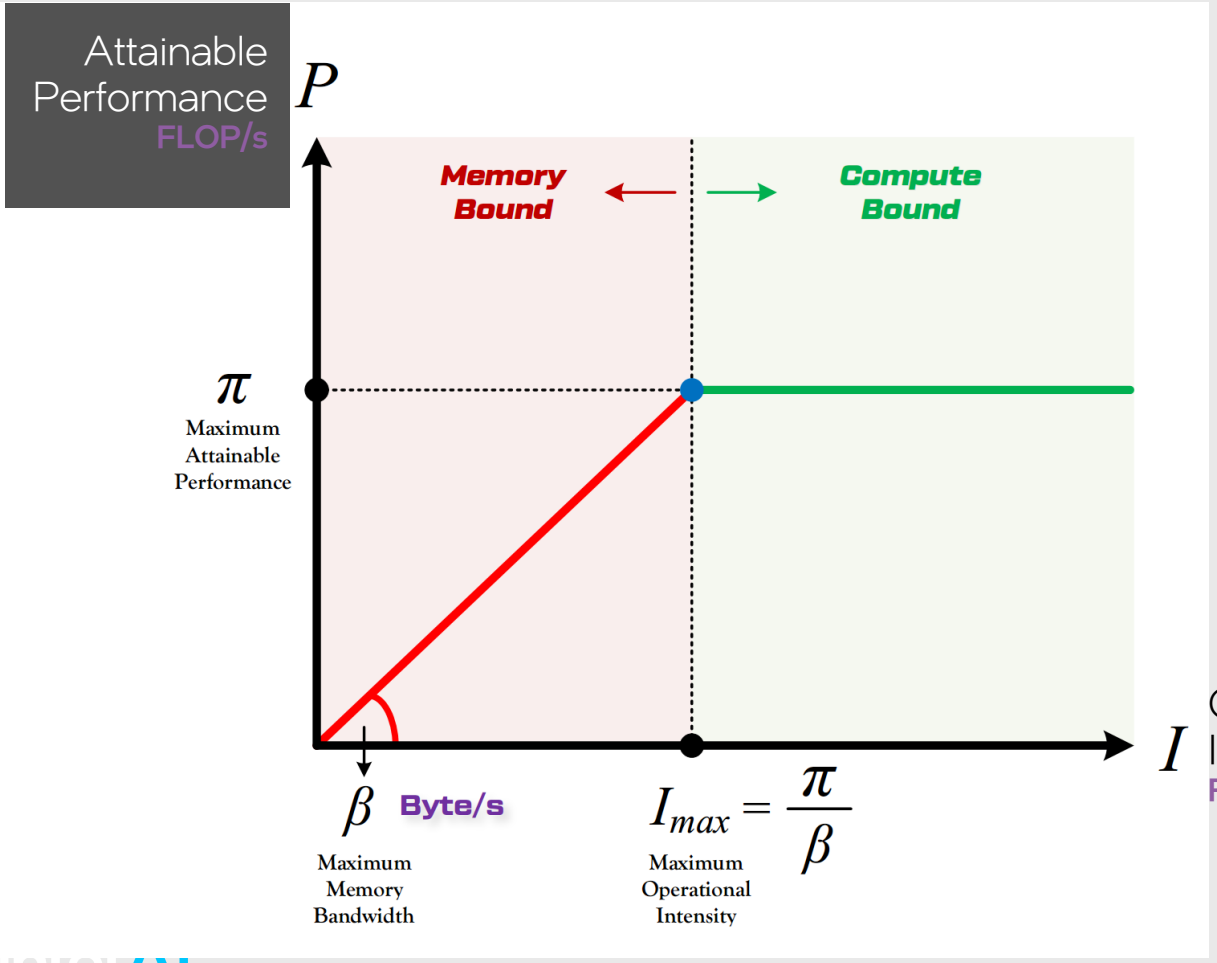
Python

```
- from transformers import AutoModelForCausalLM
+ from optimum.intel.openvino import OVModelForCausalLM

- model = AutoModelForCausalLM.from_pretrained(model_id)
+ ov_model = OVModelForCausalLM.from_pretrained(model_id)

generate_ids = ov_model.generate(input_ids)
```

intel Ai summit


OpenVINO™
Optimize LLM
with OpenVINO

Challenge for LLM Deployment
LLM Inference is a Memory Bound Task



1. Weight Compression (NNCF)

Model	Mode	Perplexity	Perplexity Increase	Model Size (GB)
databricks/dolly-v2-3b	fp32	5.01	0	10.3
databricks/dolly-v2-3b	int8	5.07	0.05	2.6
databricks/dolly-v2-3b	int4_asym_g32_r50	5.28	0.26	2.2
databricks/dolly-v2-3b	nf4_g128_r60	5.19	0.18	1.9
meta-llama/Llama-2-7b-chat-hf	fp32	3.28	0	25.1
meta-llama/Llama-2-7b-chat-hf	int8	3.29	0.01	6.3
meta-llama/Llama-2-7b-chat-hf	int4_asym_g128_r80	3.41	0.14	4.0
meta-llama/Llama-2-7b-chat-hf	nf4_g128	3.41	0.13	3.5
togethercomputer/RedPajama-INCITE-7B-Instruct	fp32	4.15	0	25.6
togethercomputer/RedPajama-INCITE-7B-Instruct	int8	4.17	0.02	6.4
togethercomputer/RedPajama-INCITE-7B-Instruct	nf4_ov_g32_r60	4.28	0.13	5.1
togethercomputer/RedPajama-INCITE-7B-Instruct	int4_asym_g128	4.17	0.02	3.6

OpenVINO™ Integration with Optimum

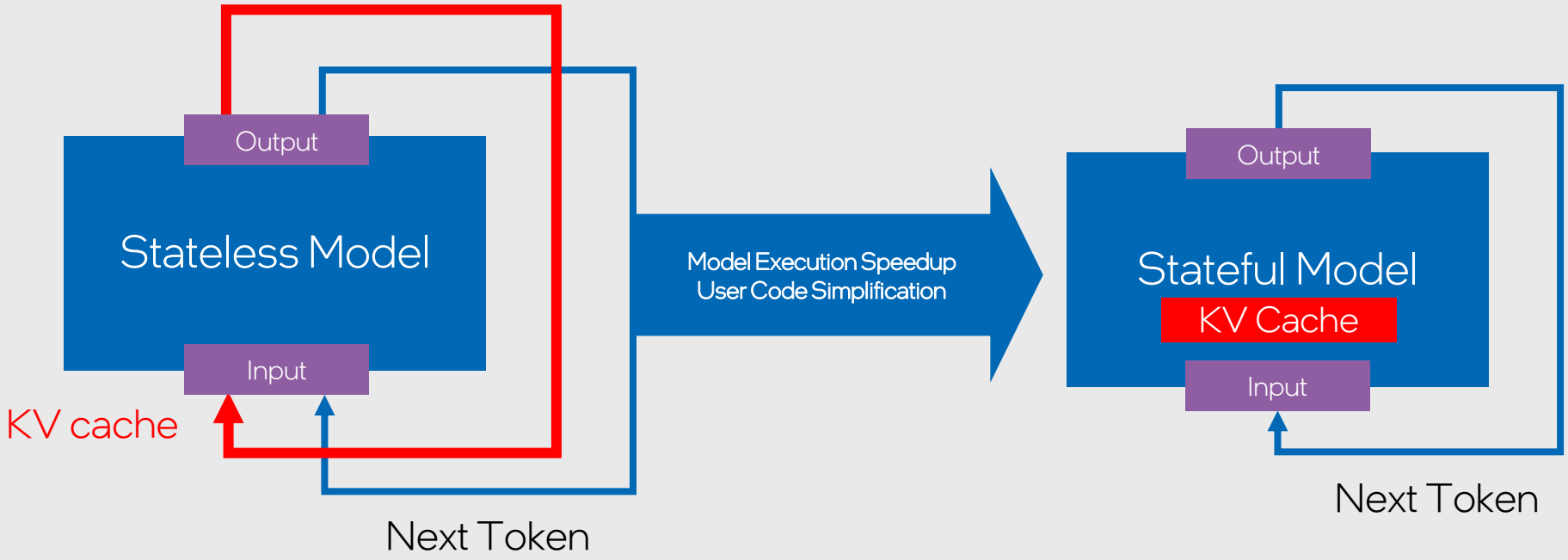
Using AutoGPTQ with int 4 precision:

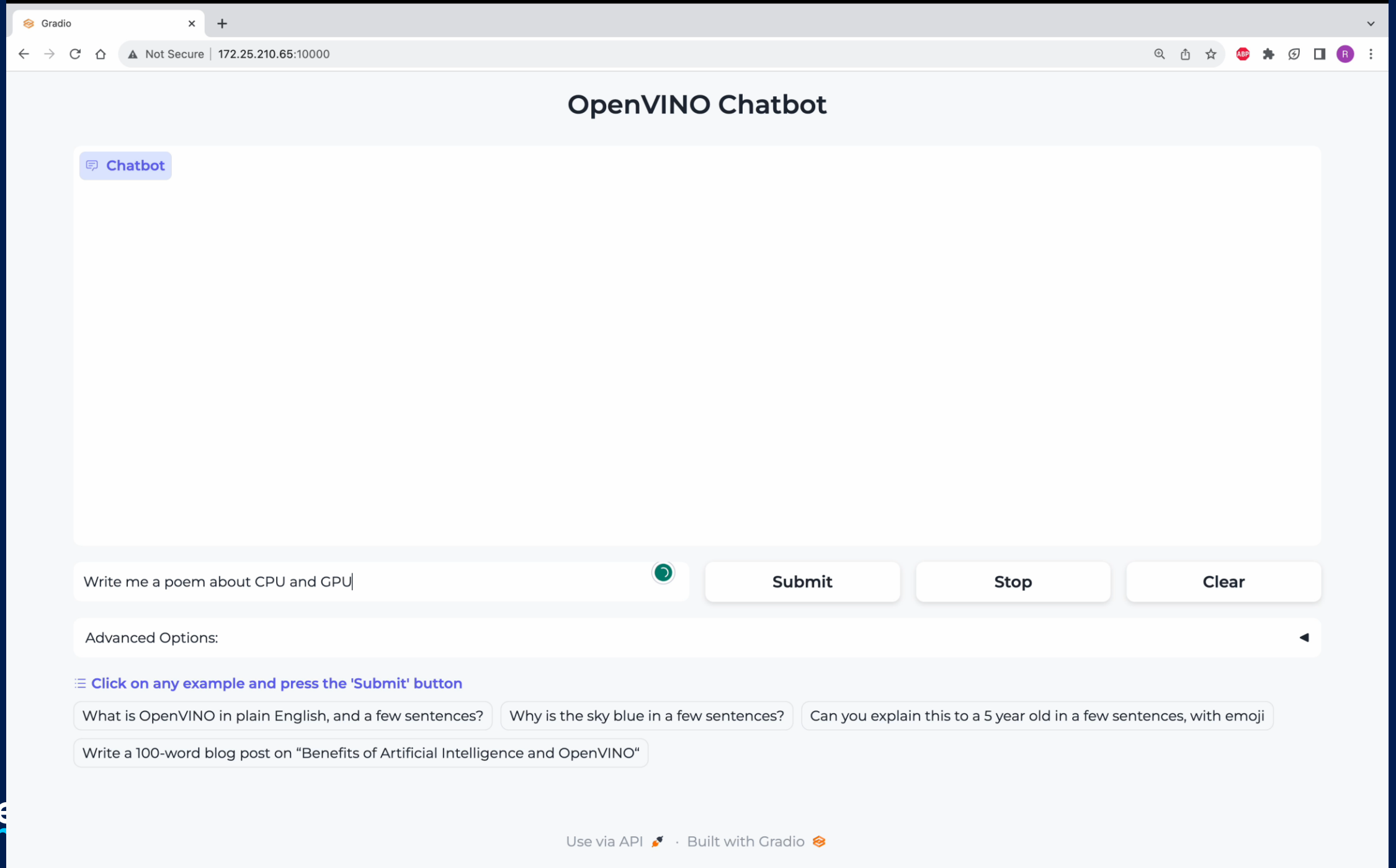
Python

```python
# make use of optimum-intel
from optimum.intel import OVModelForCausalLM

# load pretrained model, convert to OpenVINO representation
# with keeping weights in int4
model = OVModelForCausalLM.from_pretrained("TheBloke/Llama-2-7B-GPTQ",
                                           use_cache=True, export=True)

# store OpenVINO IR in a folder
model.save_pretrained("./Llama-2-7B-GPTQ")
```

2. Stateful Transformation

How to optimize KV cache with OpenVINO




Gradio
Not Secure | 172.25.210.65:10000
OpenVINO Chatbot
Chatbot
Write me a poem about CPU and GPU
Submit
Stop
Clear
Advanced Options:
Click on any example and press the 'Submit' button
What is OpenVINO in plain English, and a few sentences?
Why is the sky blue in a few sentences?
Can you explain this to a 5 year old in a few sentences, with emoji
Write a 100-word blog post on “Benefits of Artificial Intelligence and OpenVINO“
Use via API · Built with Gradio


OpenVINO™
NEW Benchmark Tool and C++ Examples for LLM

openvino.genai

Pinned

openvino Public

OpenVINO™ is an open-source toolkit for optimizing and deploying AI inference

C++ 5.4k 1.9k

nncf Public

Neural Network Compression Framework for enhanced OpenVINO™ inference

Python 729 187

openvino_notebooks Public

Jupyter notebook tutorials for OpenVINO™

Jupyter Notebook 1.7k 651

openvino.genai Public

Run Generative AI models using native OpenVINO C++ API

Python 35 33

model_server Public

A scalable inference server for models optimized with OpenVINO™

C++ 619 185

training_extensions Public

Train, Evaluate, Optimize, Deploy Computer Vision Models via OpenVINO™

Python 1.1k 437

openvino.genai
Benchmark Tool

Input Length – Output Length

```
numactl -N 1 --membind=1 python benchmark.py -m /mnt/llm_irs/models_db24bab9_stateful/llama-2-13b-chat/pytorch/dldt/compressed_weights/OV_FP16-INT8_ASYM -d cpu -r ./test_report_db24bab9/ll
INFO:nncf:NNCF initialized successfully. Supported frameworks detected: torch, onnx, openvino
[ INFO ] ==SUCCESS FOUND==: use_case: text_gen, model_type: llama-2-13b-chat
[ INFO ] OV Config={'PERFORMANCE_HINT': 'LATENCY', 'CACHE_DIR': '', 'NUM_STREAMS': '1'}
[ INFO ] OPENVINO_TORCH_BACKEND_DEVICE=CPU
[ INFO ] Model path=/mnt/llm_irs/models_db24bab9_stateful/llama-2-13b-chat/pytorch/dldt/compressed_weights/OV_FP16-INT8_ASYM, openvino runtime version: 2024.0.0-14166-db24bab90ae-pr_22392
Compiling the model to CPU ...
[ INFO ] From pretrained time: 5.70s
[ INFO ] Read prompts from /home/build/jenkins_home/workspace/test-llm-model-imp/llm_prompts/32_1024/llama-2-13b-chat.jsonl
[ INFO ] Numbeams: 1, benchmarking iter nums(exclude warm-up): 3, prompt nums: 2
[ INFO ] [warm-up] Input text: Once upon a time, there existed a little girl who liked to have adventures. She wanted to go to places and meet new people, and have fun
[ INFO ] [warm-up] Input token size: 32, Output size: 128, Infer count: 128, Tokenization Time: 4.60ms, Detokenization Time: 0.40ms, Generation Time: 9.91s, Latency: 77.42 ms/token
[ INFO ] [warm-up] First token latency: 649.84 ms/token, other tokens latency: 72.89 ms/token, len of tokens: 128
[ INFO ] [warm-up] First infer latency: 649.01 ms/infer, other infers latency: 72.43 ms/infer, inference count: 128
[ INFO ] [warm-up] Max rss memory cost: 26407.07MBytes, max shared memory cost: 12568.50MBytes
[ INFO ] [warm-up] Result MD5:['c2ba8223fe81b826cf3e69f0b9af51c3']
[ INFO ] [warm-up] Generated: <s> Once upon a time, there existed a little girl who liked to have adventures. She wanted to go to places and meet new people, and have fun. She was always a
```

First Token Latency (ms)

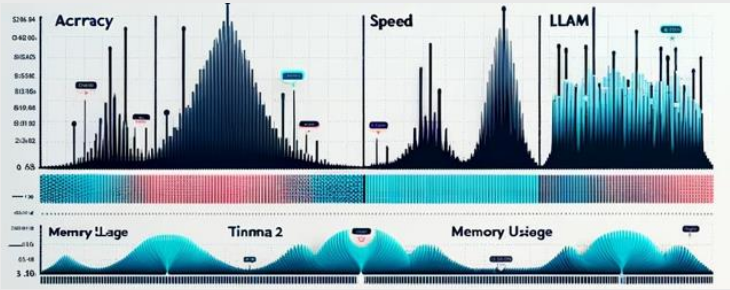
Other Token Latency (ms)

Memory Usage (GB)

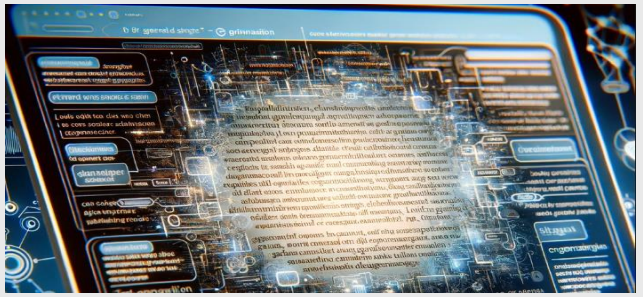
intel Ai summit

OpenVINO™ GenAI Pipeline Repo

OpenVINO Native C&C++ Pipeline for Gen AI and LLM



Benchmarking for LLMs



Text generation C++ samples

Stable Diffusion (with LoRA) C++ image Generation Pipeline

LCM (with LoRA) C++ image Generation Pipeline

intel Ai summit


OpenVINO™
Customize
Your LLM

Challenge for LLM Application

Fabricate Facts

Knowledge is Out of Date

Lack Knowledge on Specific Domains

How to Enable New Knowledge on LLM?

Fine-Tunning

Model Adaptation Required

RAG

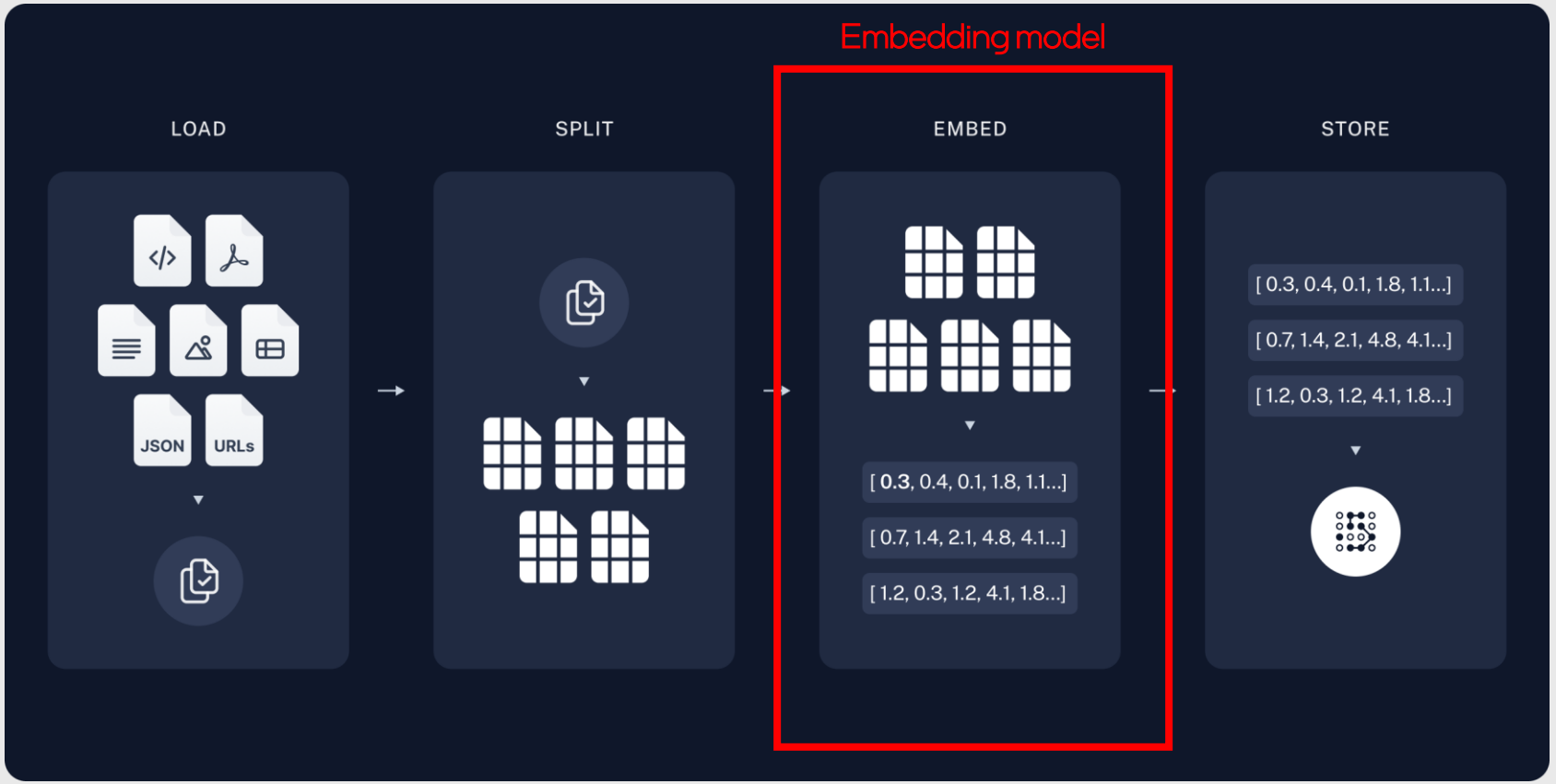
External Knowledge Required

Comparison

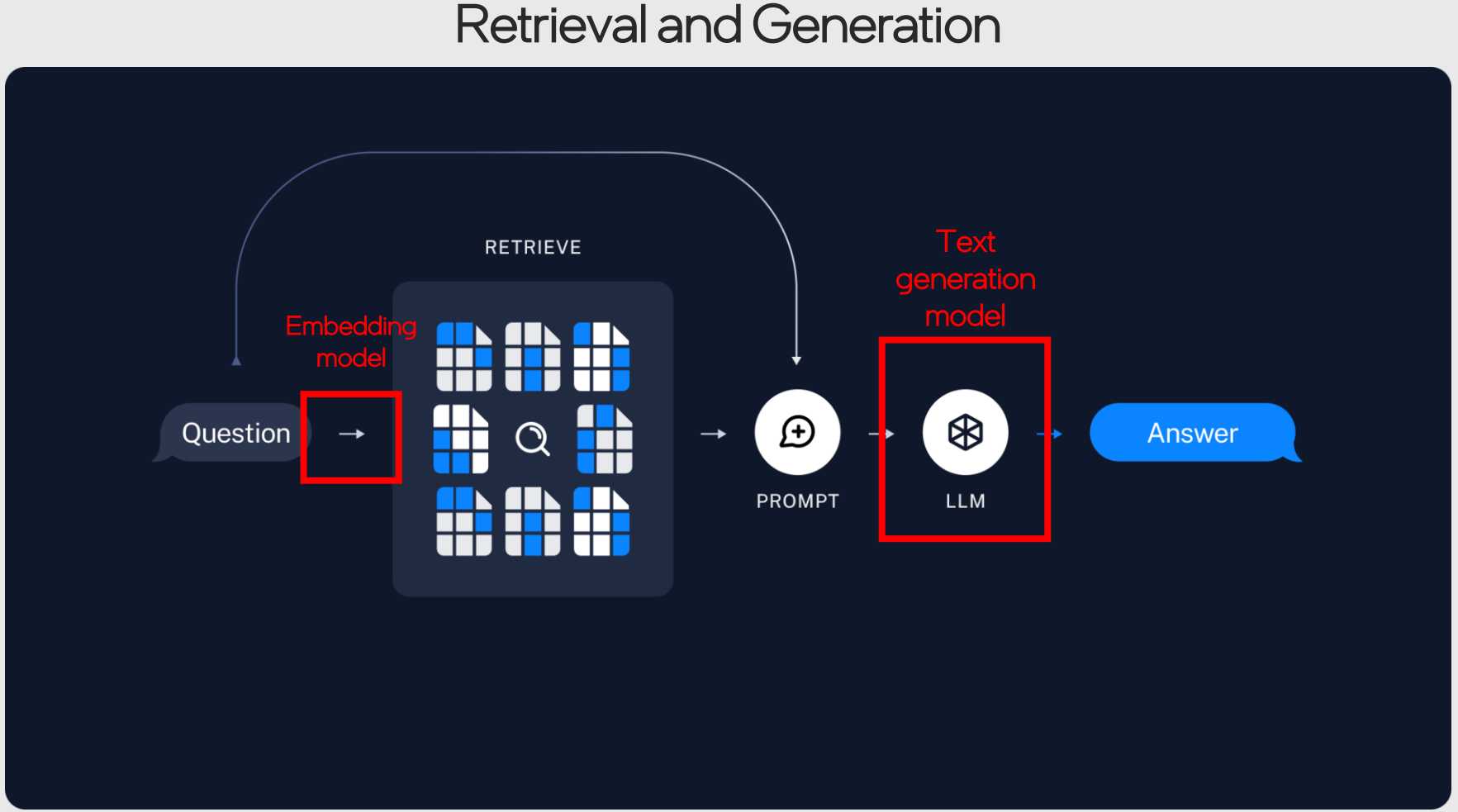
	Knowledge Update	Interpretability	Latency
Fine-Tunning	Require retraining	Black box, lower Interpretability	**Lower**
RAG	**Directly** update to retrieval knowledge base	Answer is **traceable**	Higher

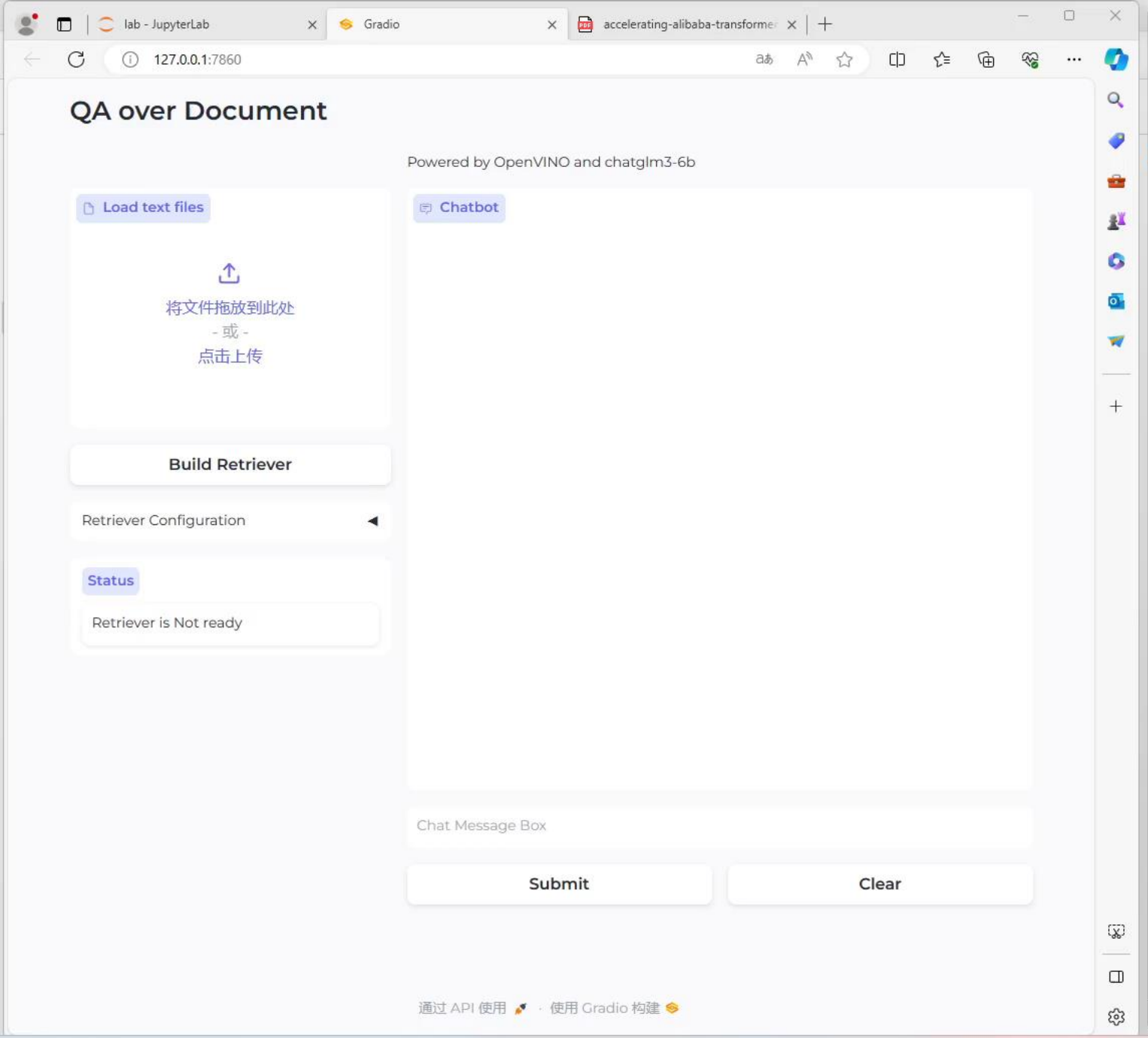
Retrieval-Augmented Generation (RAG)

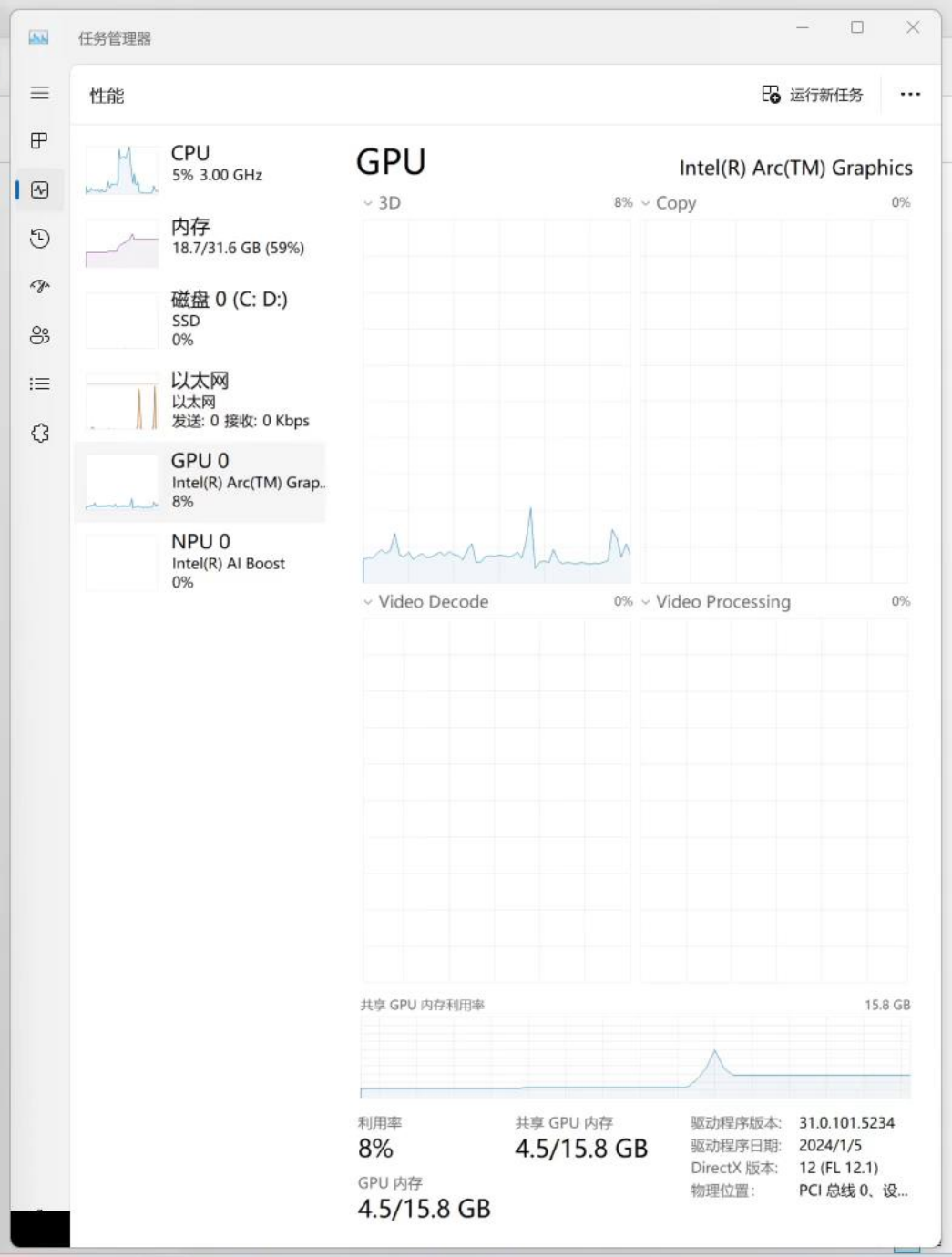
Indexing



Retrieval-Augmented Generation (RAG)



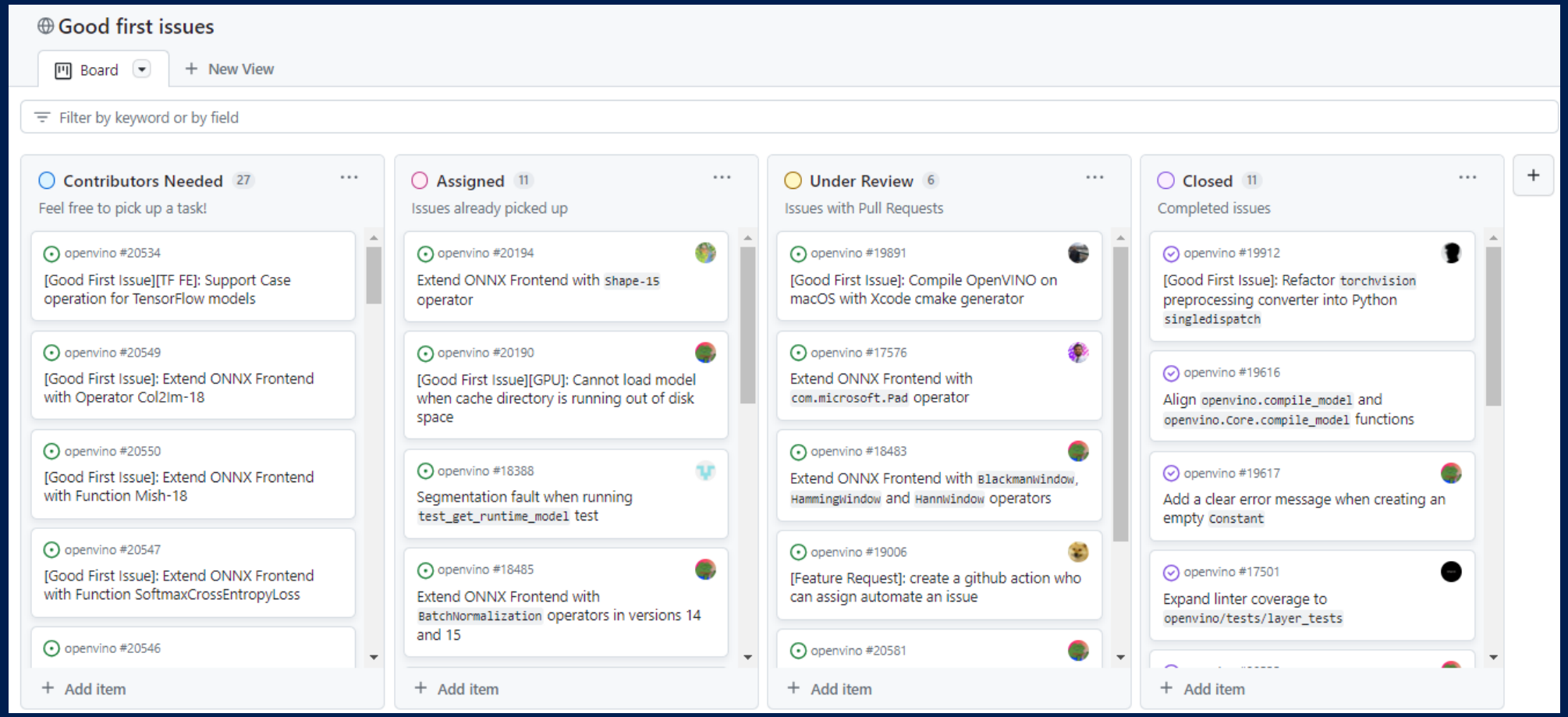

lab - JupyterLab
Gradio
accelerating-alibaba-transforme
127.0.0.1:7860
QA over Document
Powered by OpenVINO and chatglm3-6b
Load text files
将文件拖放到此处
- 或 -
点击上传
Chatbot
Build Retriever
Retriever Configuration
Status
Retriever is Not ready
Chat Message Box
Submit
Clear
通过 API 使用 · 使用 Gradio 构建


任务管理器
性能
运行新任务
CPU
5% 3.00 GHz
内存
18.7/31.6 GB (59%)
磁盘 0 (C: D:)
SSD
0%
以太网
以太网
发送: 0 接收: 0 Kbps
GPU 0
Intel(R) Arc(TM) Grap..
8%
NPU 0
Intel(R) AI Boost
0%
GPU
Intel(R) Arc(TM) Graphics
3D
8%
Copy
0%
Video Decode
0%
Video Processing
0%
共享 GPU 内存利用率
15.8 GB
利用率
8%
共享 GPU 内存
4.5/15.8 GB
GPU 内存
4.5/15.8 GB
驱动程序版本: 31.0.101.5234
驱动程序日期: 2024/1/5
DirectX 版本: 12 (FL 12.1)
物理位置: PCI 总线 0、设...


7°C
多云
搜索
17:40
2024/1/26


Try It Yourself
openvino.ai
OpenVINO
intel

Contribute to OpenVINO™ Toolkit



How to start: https://medium.com/openvino-toolkit/how-to-contribute-to-an-ai-open-source-project-c741f48e009e

intel Ai summit

Introducing Intel's AI PC Developer Program

A new initiative by Intel to grow AI PC software ecosystem by empowering developers with best tools and resources

- Early access to Intel's leading-edge XPUs and optimization tools
- Best AI deployment frameworks on Intel XPUs
- Seamless, consistent and reliable developer experience
- A developer kit to make it easy to adopt new AI technologies
- Opportunity to scale through Intel's broad and open ecosystem

intel Ai summit

Join the Future of AI PCs!

- Attend the AI PC training sessions this afternoon
- Sign up for the AI PC Developer Kit lab sessions at tomorrow's Ecosystem Symposium
- Join the AI PC Developer Program at *Program Website (TBD)* and take home a developer kit

intel Ai summit

Notices and Disclaimers

For notices, disclaimers, and details about performance claims, visit www.intel.com/PerformanceIndex or scan the QR code:

intel Ai summit

Thank You!