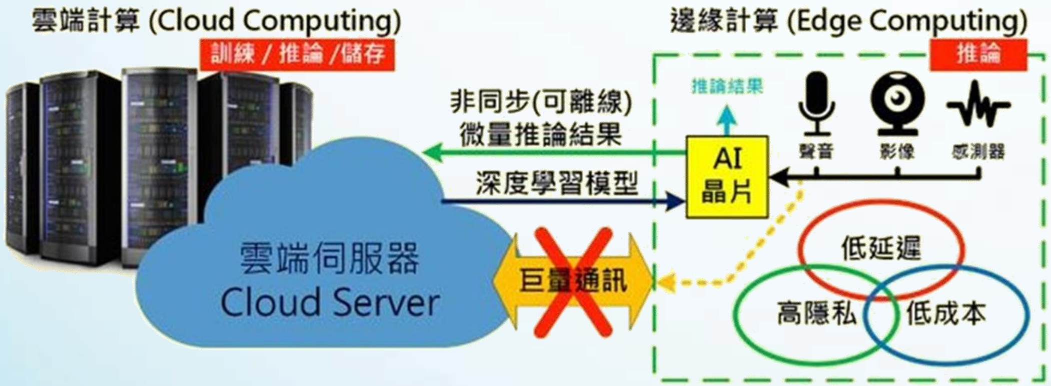
OmniXRI's Edge AI & TinyML 小學堂



Medium Edge
Mini Edge
Micro Edge

歡迎加入
邊緣人俱樂部

【第15講】
實作案例 –
文字語音生成

歐尼克斯實境互動工作室 (OmniXRI Studio)
許哲豪 (Jack Hsu)

Intel OpenVINO & Notebooks 回顧

5.2.
Intel OpenVINO
簡介

- 演進歷史
- 架構簡介
- 工作流程
- 重大革新
- 文件說明
- 下載安裝
- 範例來源

2024/04/02

OmniXRI's Edge AI & TinyML 小學堂_【第5講】開源模型推論工具

5.3.
OpenVINO
Notebooks簡介

- 功能簡介
- 下載安裝
- 執行畫面
- 範例練習

2024/04/02

OmniXRI's Edge AI & TinyML 小學堂_【第5講】開源模型推論工具 _OmniXRI_JackHsu

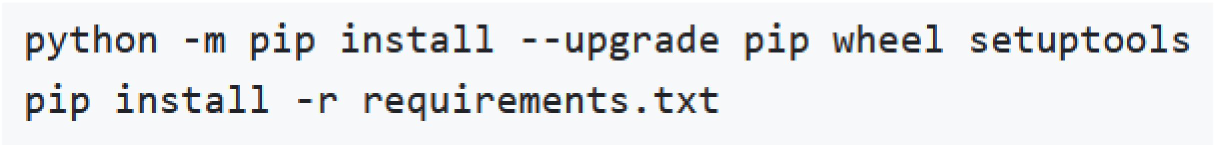
24

本週課程假設已在個人電腦上安裝好OpenVINO 2024版及 Notebooks。若尚未安裝者請參考第5講課程。

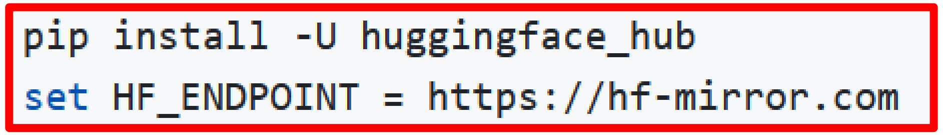
直播連結：https://youtu.be/6By3GXuEpFc

Intel OpenVINO Notebooks Windows安裝

1. Install Python (**3.9, 3.10, 3.11** 64bit)
2. Install Git
3. Install C++ Redistributable and **FFMPEG (Optional)**
4. Install Notebooks
5. Create Virtual Environment
6. Activate the Environment
7. Clone the Repository
8. Install the Packages
9. Launch the Notebooks



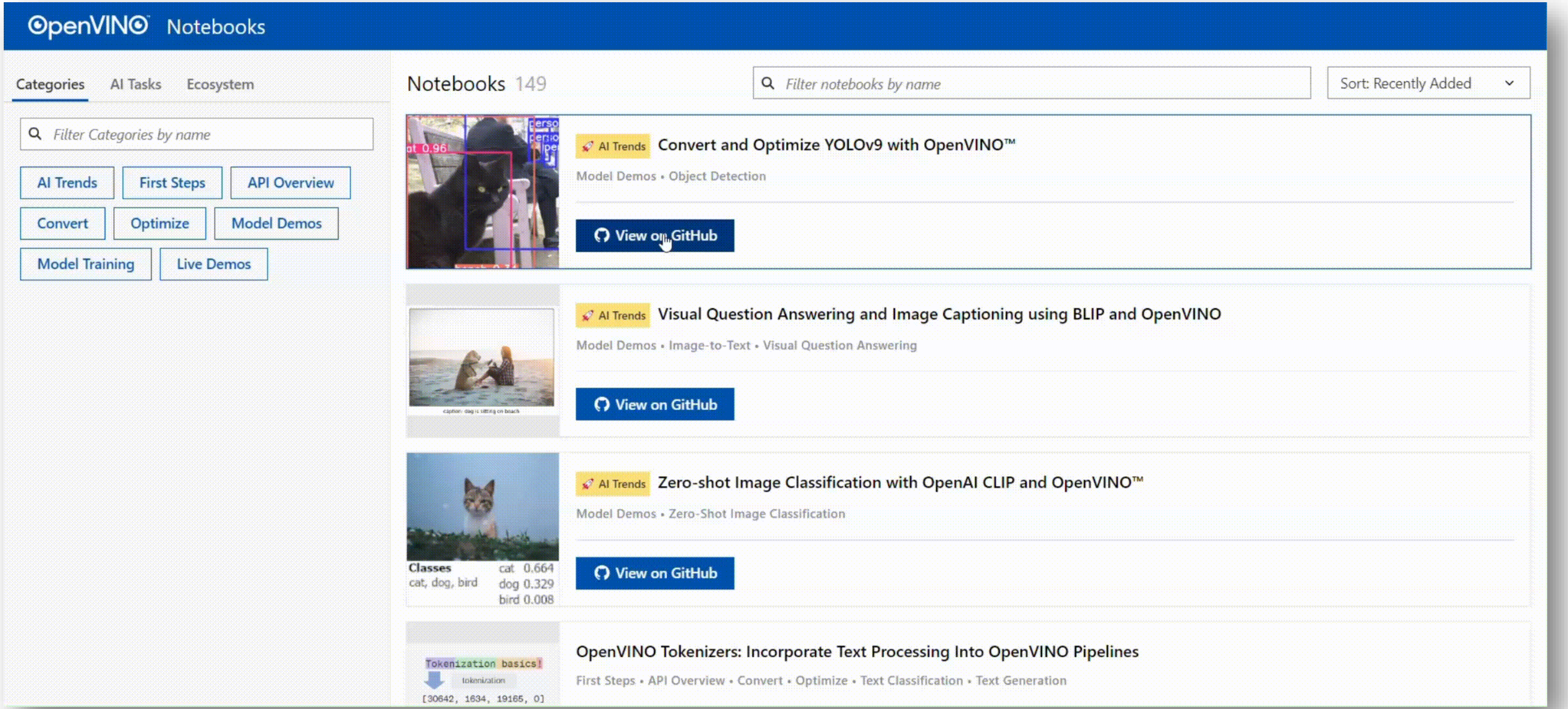
```
python -m pip install --upgrade pip wheel setuptools
pip install -r requirements.txt
```



```
pip install -U huggingface_hub
set HF_ENDPOINT = https://hf-mirror.com
```

https://github.com/openvinotoolkit/openvino_notebooks/wiki/Windows

Intel OpenVINO Notebooks 範例程式網頁



Notebooks 149

AI Trends Convert and Optimize YOLOv9 with OpenVINO™

Model Demos • Object Detection

AI Trends Visual Question Answering and Image Captioning using BLIP and OpenVINO

Model Demos • Image-to-Text • Visual Question Answering

AI Trends Zero-shot Image Classification with OpenAI CLIP and OpenVINO™

Model Demos • Zero-Shot Image Classification

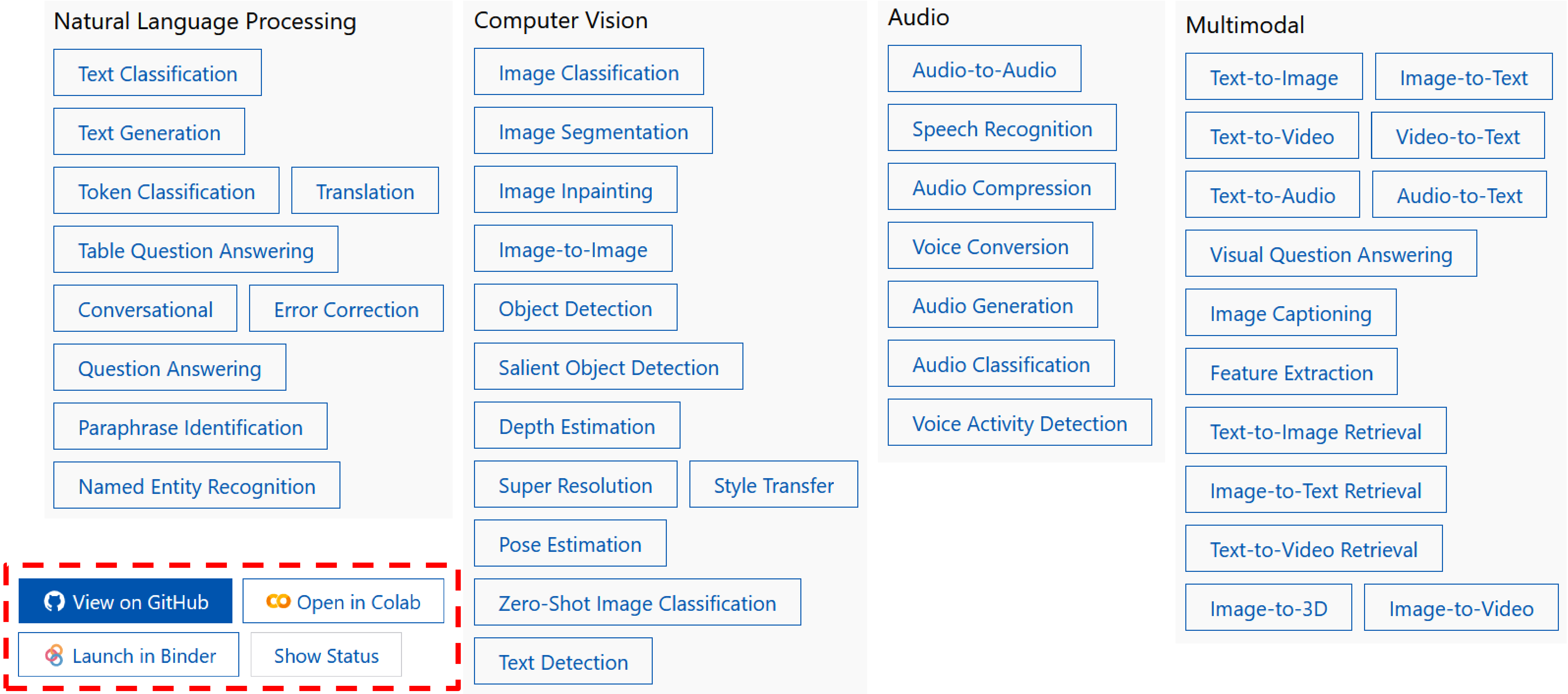
Classes	
cat, dog, bird	cat 0.664 dog 0.329 bird 0.008

OpenVINO Tokenizers: Incorporate Text Processing Into OpenVINO Pipelines

First Steps • API Overview • Convert • Optimize • Text Classification • Text Generation

https://openvinotoolkit.github.io/openvino_notebooks/

OpenVINO Notebook 範例類型



Natural Language Processing

- Text Classification
- Text Generation
- Token Classification
- Translation
- Table Question Answering
- Conversational
- Error Correction
- Question Answering
- Paraphrase Identification
- Named Entity Recognition

Computer Vision

- Image Classification
- Image Segmentation
- Image Inpainting
- Image-to-Image
- Object Detection
- Salient Object Detection
- Depth Estimation
- Super Resolution
- Style Transfer
- Pose Estimation
- Zero-Shot Image Classification
- Text Detection

Audio

- Audio-to-Audio
- Speech Recognition
- Audio Compression
- Voice Conversion
- Audio Generation
- Audio Classification
- Voice Activity Detection

Multimodal

- Text-to-Image
- Image-to-Text
- Text-to-Video
- Video-to-Text
- Text-to-Audio
- Audio-to-Text
- Visual Question Answering
- Image Captioning
- Feature Extraction
- Text-to-Image Retrieval
- Image-to-Text Retrieval
- Text-to-Video Retrieval
- Image-to-3D
- Image-to-Video

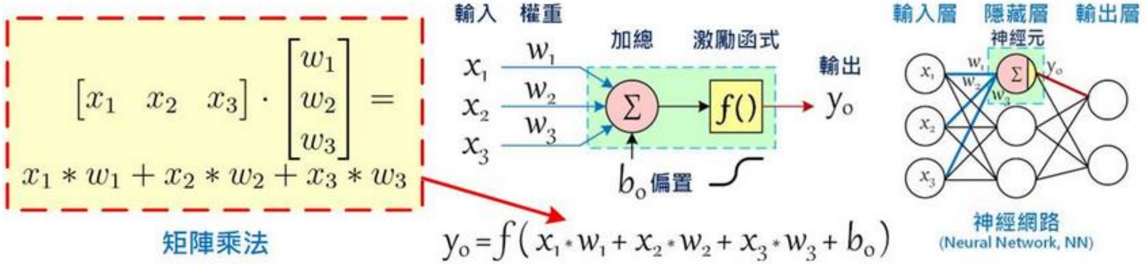
簡報大綱

- 15.1. 大語言模型簡介
- 15.2. 常見文字生成應用
- 15.3. 文字生成應用實例
- 15.4. 常見語音生成應用
- 15.5. 語音生成應用實例

本課程完全免費，請勿移作商業用途！
歡迎留言、訂閱、點讚、轉發，讓更多需要的朋友也能一起學習。

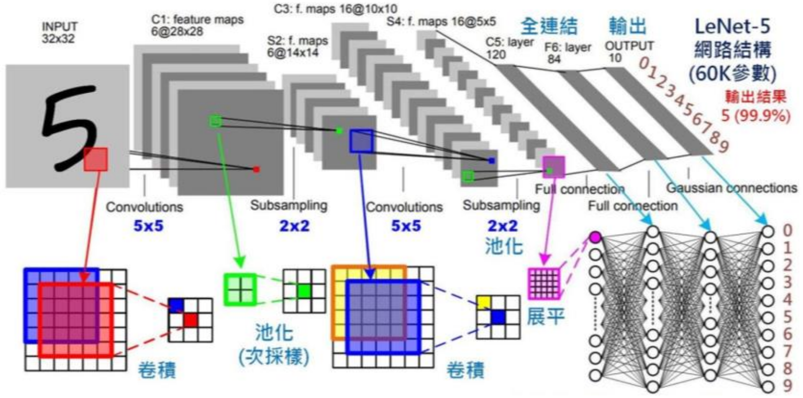
完整課程大綱：https://omnixri.blogspot.com/2024/02/omnixris-edge-ai-tinyml-0.html
課程直播清單：https://www.youtube.com/@omnixri1784/streams

神經網路技術發展

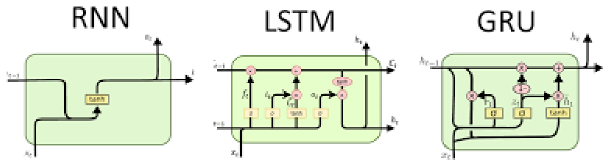


$[x_1\ x_2\ x_3] \cdot \begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix} =$
$x_1 * w_1 + x_2 * w_2 + x_3 * w_3$
矩陣乘法
輸入
權重
加總
激勵函式
輸出
x_1 x_2 x_3
w_1 w_2 w_3
Σ
$f()$
y_o
b_o 偏置
$y_o = f(x_1 * w_1 + x_2 * w_2 + x_3 * w_3 + b_o)$
輸入層
隱藏層
輸出層
神經元
神經網路
(Neural Network, NN)

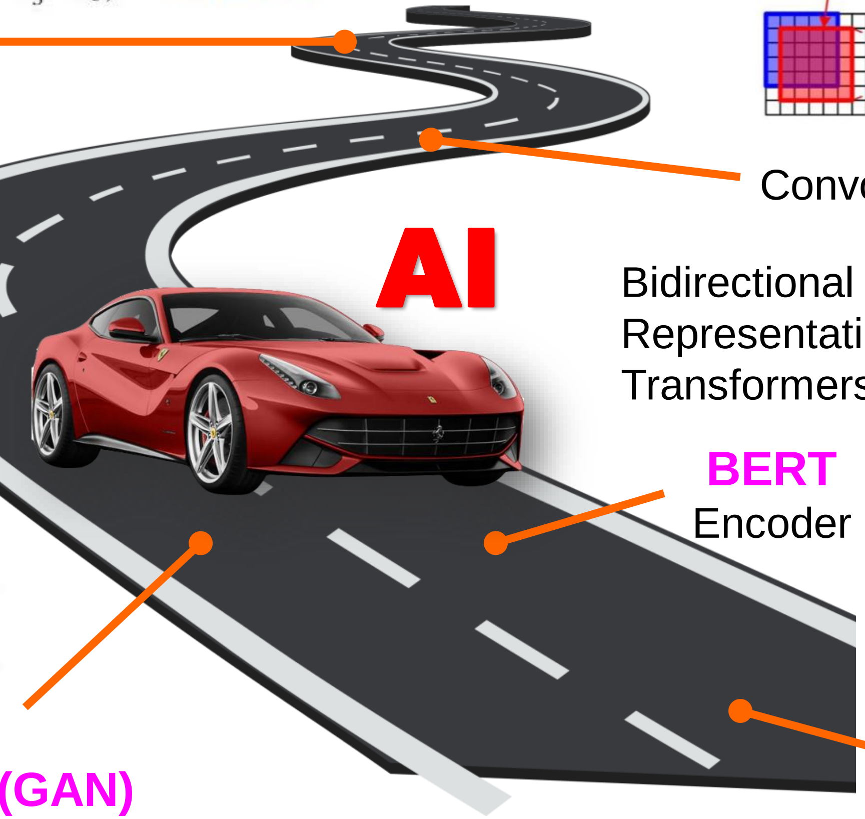
Artificial Neural Network **(ANN)**



Convolutional Neural Network **(CNN)**



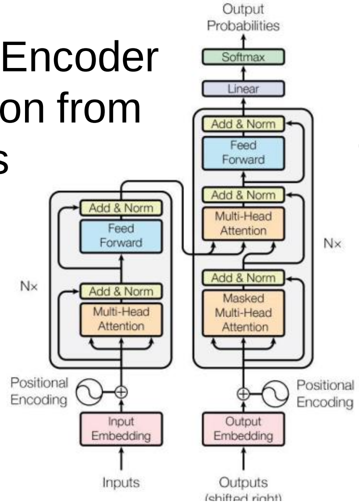
Recurrent Neural Network **(RNN)**
Long-Short Term Memory **(LSTM)**
Gated Recurrent Unit **(GRU)**



AI

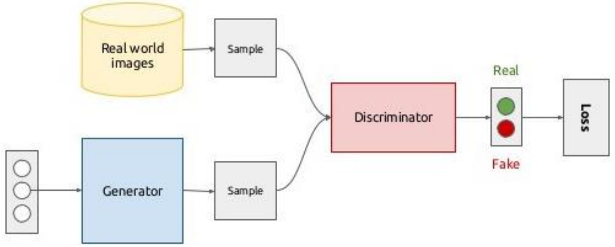
Bidirectional Encoder Representation from Transformers

BERT
Encoder



Generative Pre-trained Transformer

GPT
Decoder

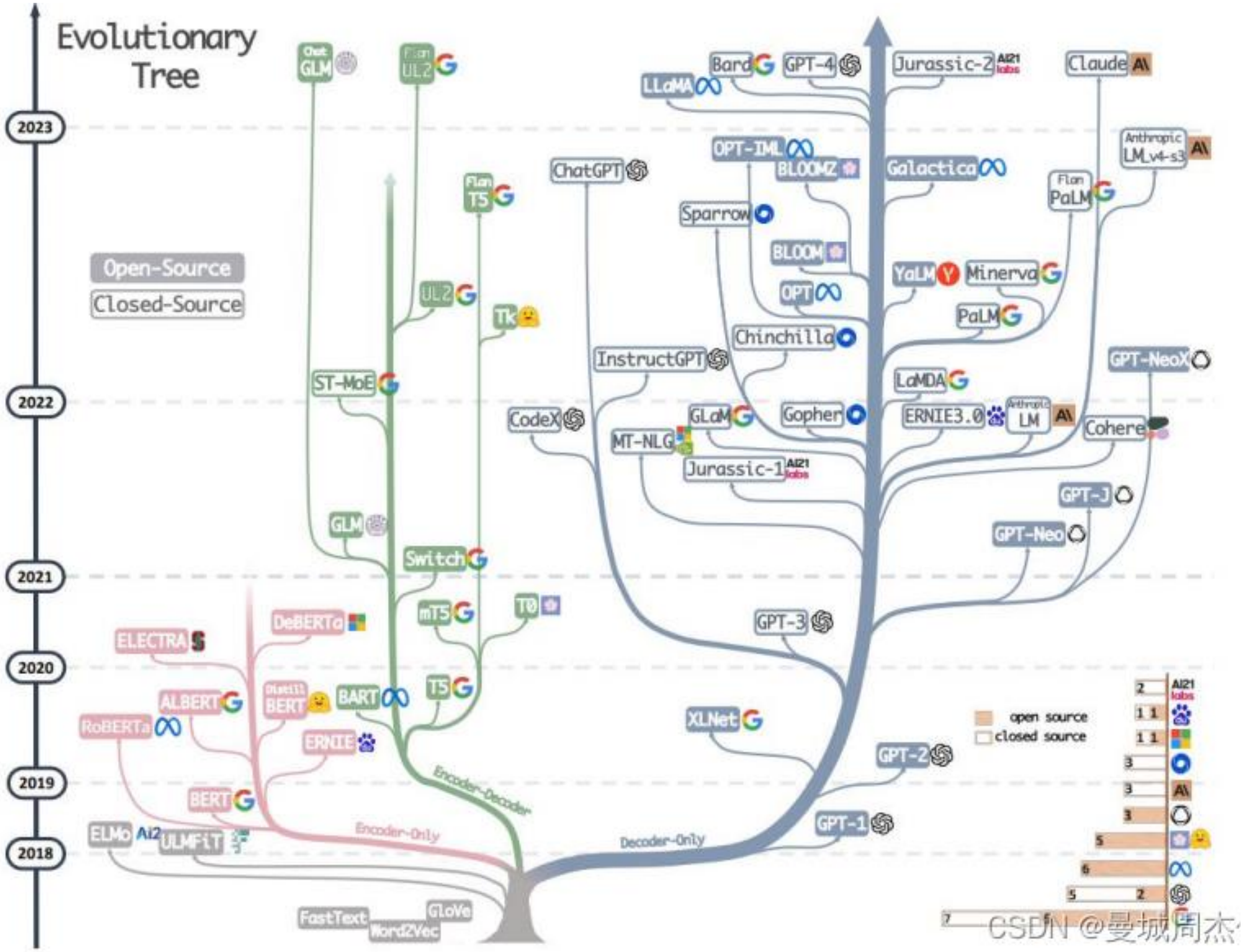


Generative Adversarial Networks **(GAN)**

Transformer

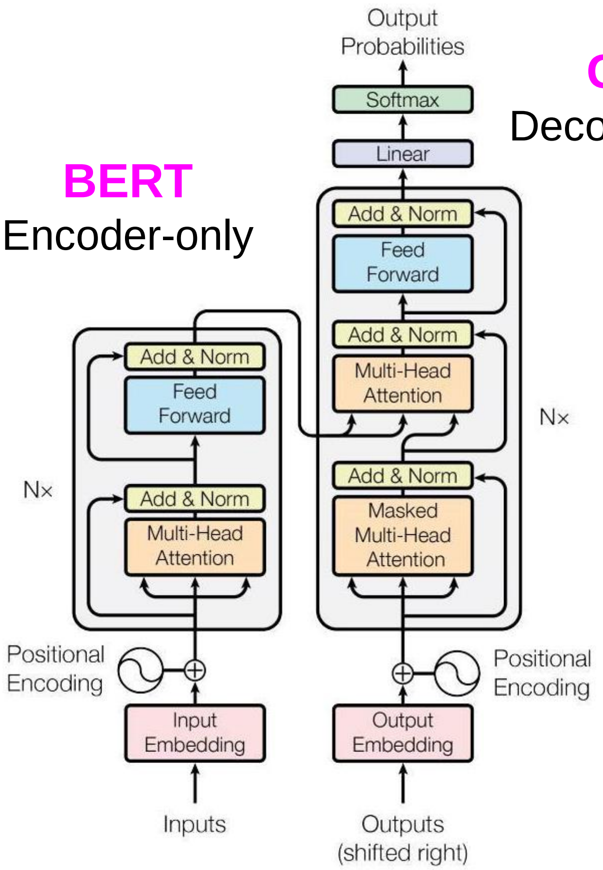
ChatGPT

大型語言模型技術演進



BERT
Encoder-only

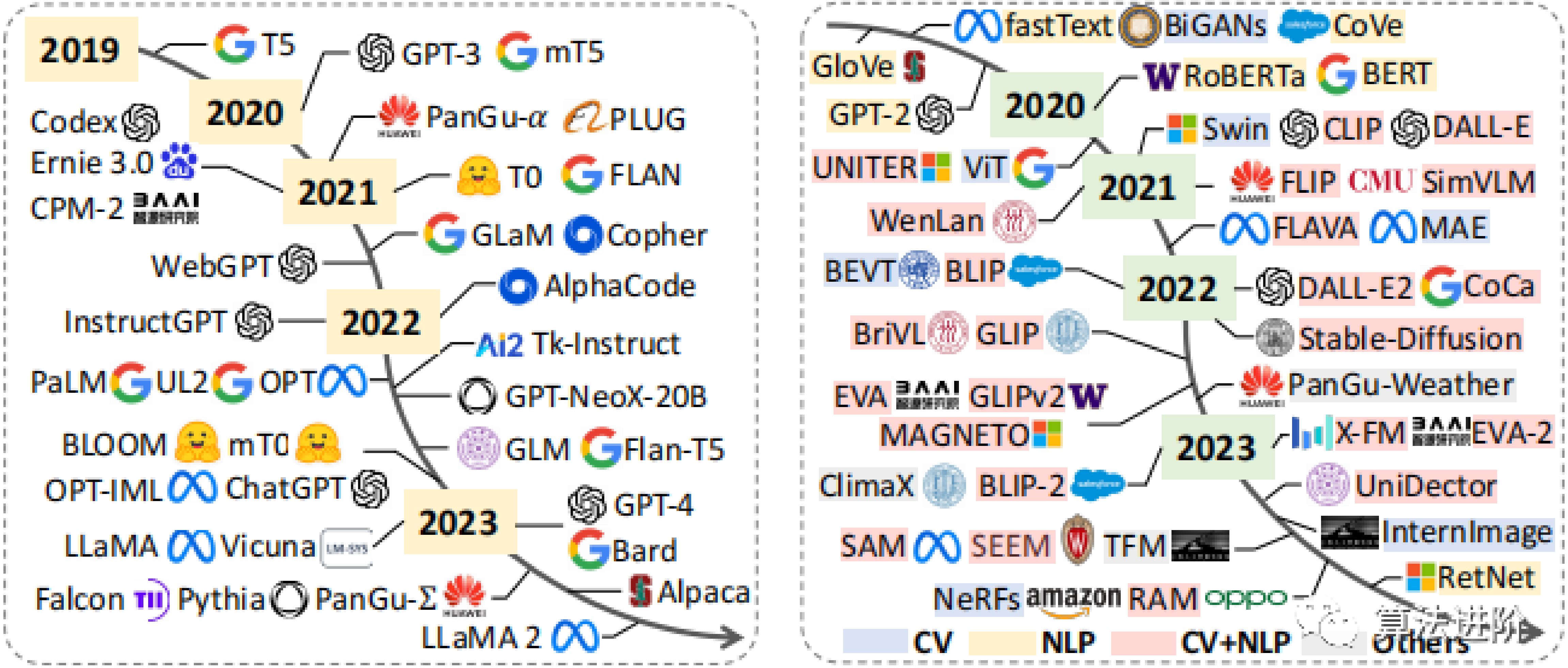
GPT
Decoder-only



Transformer
Encoder-Decoder

資料來源：https://github.com/Mooler0410/LLMsPracticalGuide

生成式模型發展



資料來源：https://blog.csdn.net/cvanlijingxuan/article/details/122260044

大型語言模型參數量

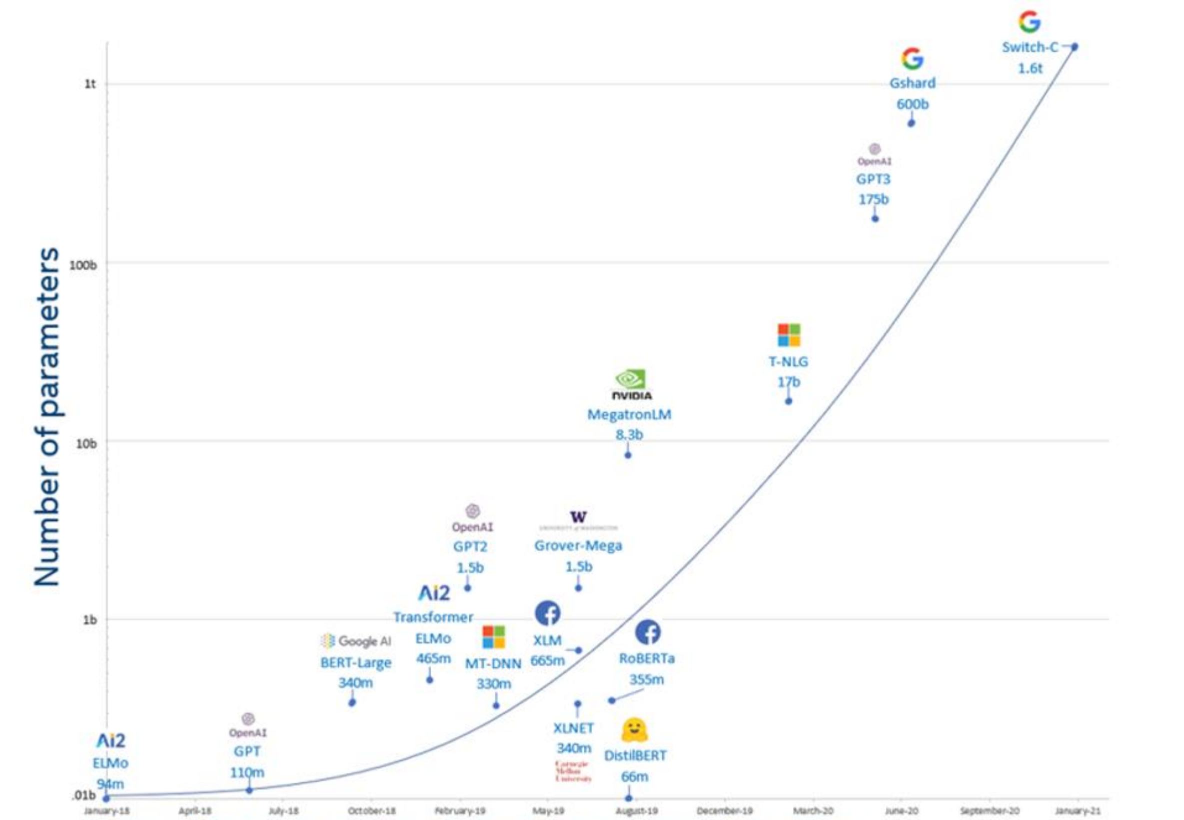


Figure 1: Exponential growth of number of parameters in DL models

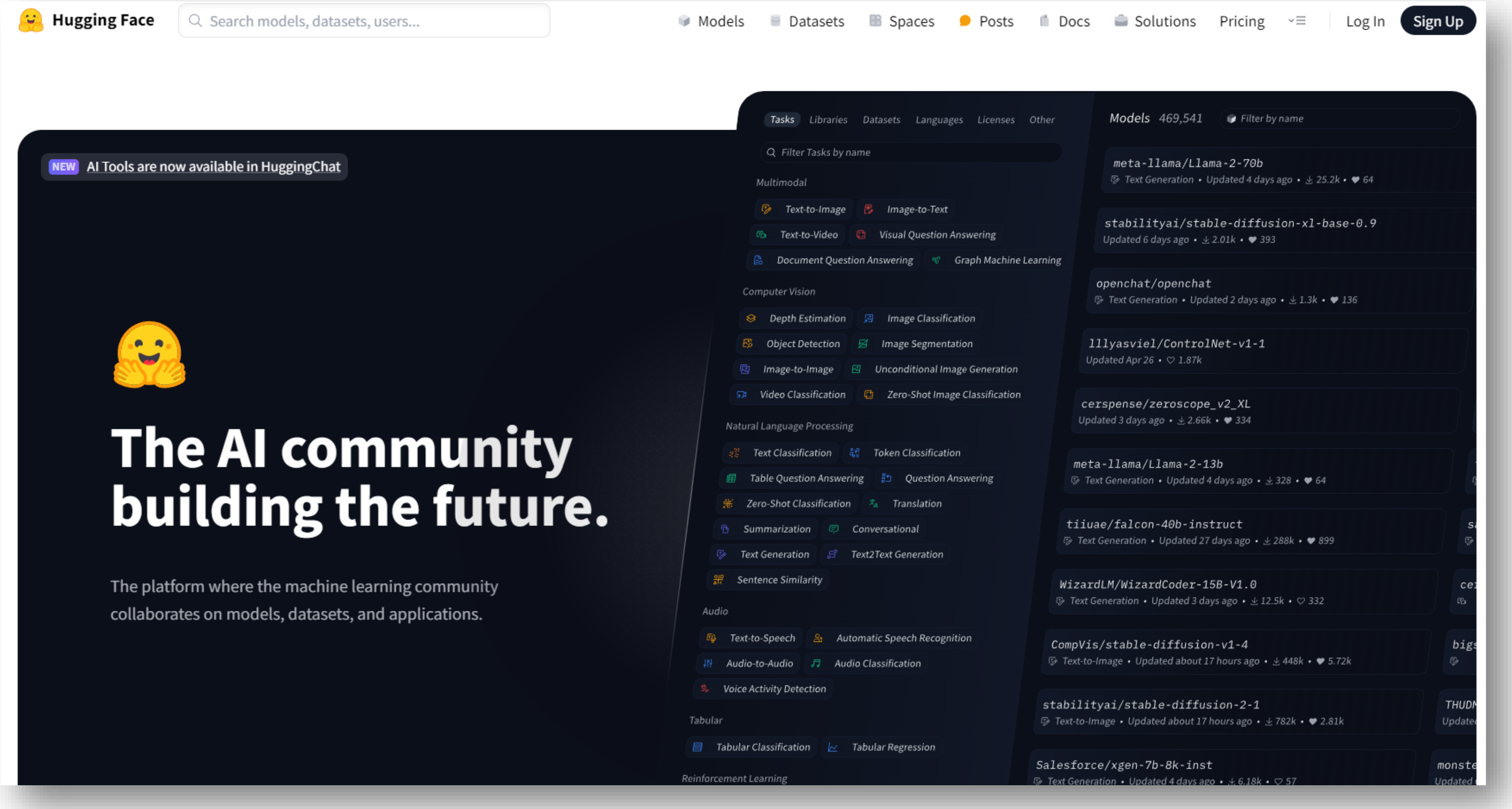
常見大型語言模型(GB~TB)

https://www.netadmin.com.tw/netadmin/zh-tw/trend/16482C89018F47719FE3DD037AF7CC9D

常見模型參數量 (B=10^9)

- GPT 3 : 175B
- GPT 4 : 8x220B
- Llama 2 : 7B, 13B, 70B
- Llama 3 : 8B, 70B, 400B
- Gemma : 2B, 7B
- Claude 3 : 20B, 70B, 2T
- Mistral : 8x7B, 8x22B
- **TAIDE : 7B, 8B**

大型語言模型庫 – Hugging Face



NEW AI Tools are now available in HuggingChat

The AI community building the future.

The platform where the machine learning community collaborates on models, datasets, and applications.

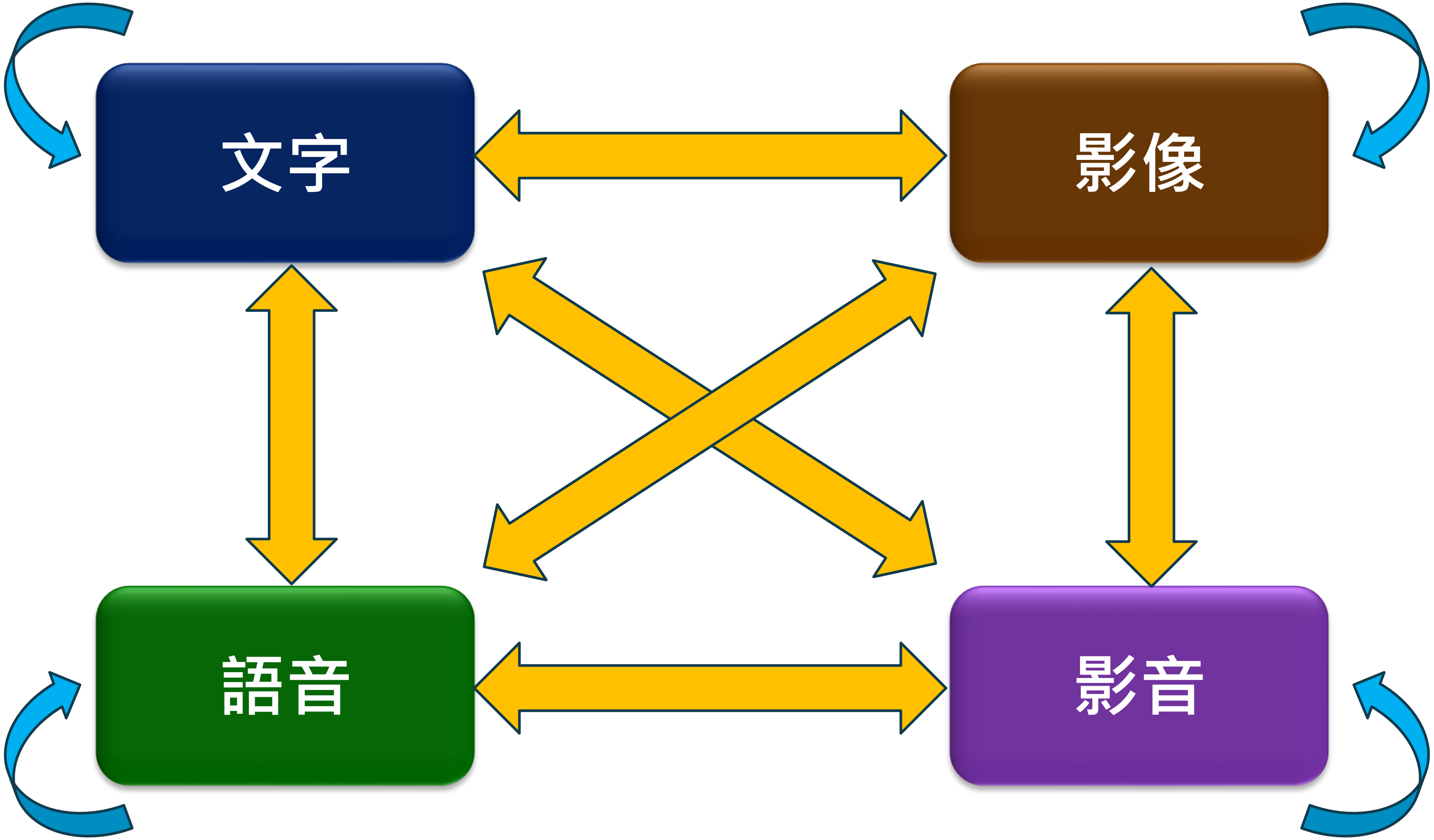
Models 469,541

- meta-llama/Llama-2-70b
  Text Generation • Updated 4 days ago • 25.2k • 64
- stabilityai/stable-diffusion-xl-base-0.9
  Updated 6 days ago • 2.01k • 393
- openchat/openchat
  Text Generation • Updated 2 days ago • 1.3k • 136
- lllyasviel/ControlNet-v1-1
  Updated Apr 26 • 1.87k
- cerspense/zeroscope_v2_XL
  Updated 3 days ago • 2.66k • 334
- meta-llama/Llama-2-13b
  Text Generation • Updated 4 days ago • 328 • 64
- tiiuae/falcon-40b-instruct
  Text Generation • Updated 27 days ago • 288k • 899
- WizardLM/WizardCoder-15B-V1.0
  Text Generation • Updated 3 days ago • 12.5k • 332
- CompVis/stable-diffusion-v1-4
  Text-to-Image • Updated about 17 hours ago • 448k • 5.72k
- stabilityai/stable-diffusion-2-1
  Text-to-Image • Updated about 17 hours ago • 782k • 2.81k
- Salesforce/xgen-7b-8k-inst
  Text Generation • Updated 4 days ago • 6.18k • 57

https://huggingface.co/

提供各種自然語言處理 (NLP) 及 Transformers 模型庫及資料集，相當於 AI 界的 Github。

生成式智慧主要應用

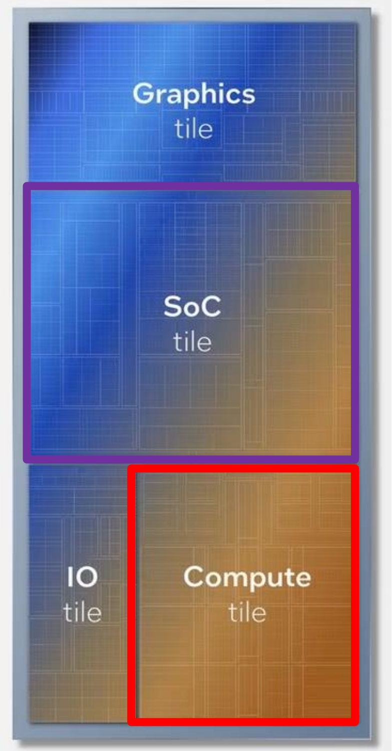


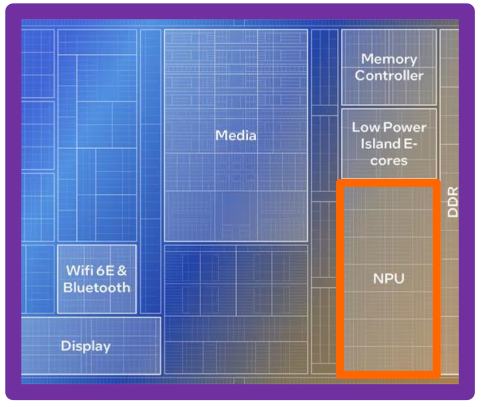
Intel AI PC 主CPU架構（AI混合加速）

Intel 14代CPU

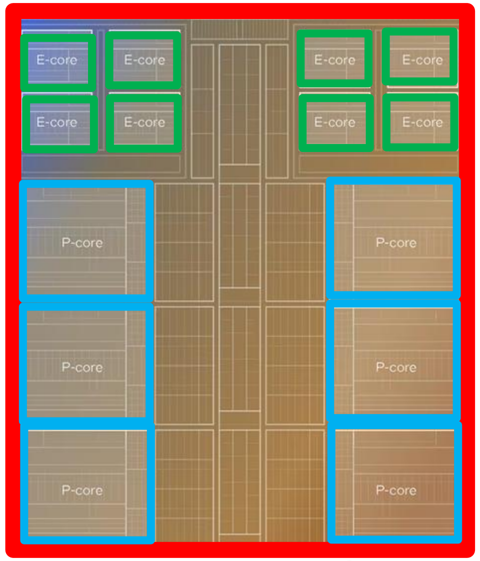
Core Ultra（Meteor Lake）

CPU + iGPU + NPU

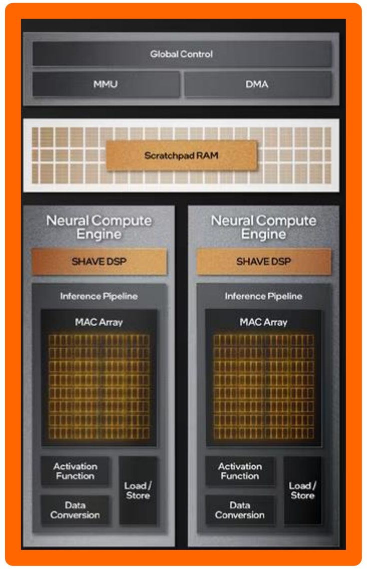




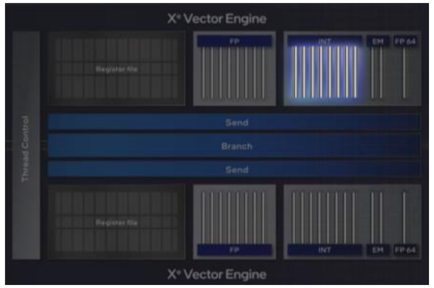
SoC tile



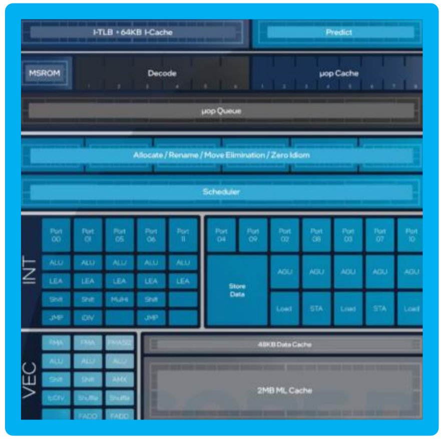
Compute tile



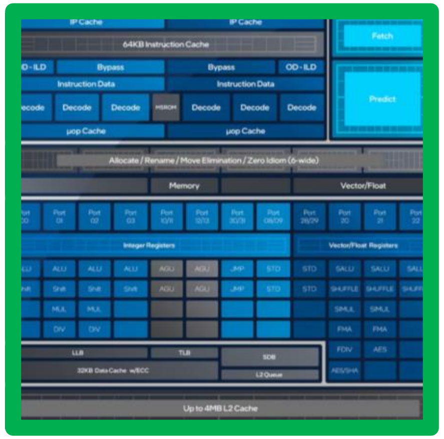
NPU（類似原Movidius神經加速棒VPU結構）



DP4A (GPU)



P-Core x6



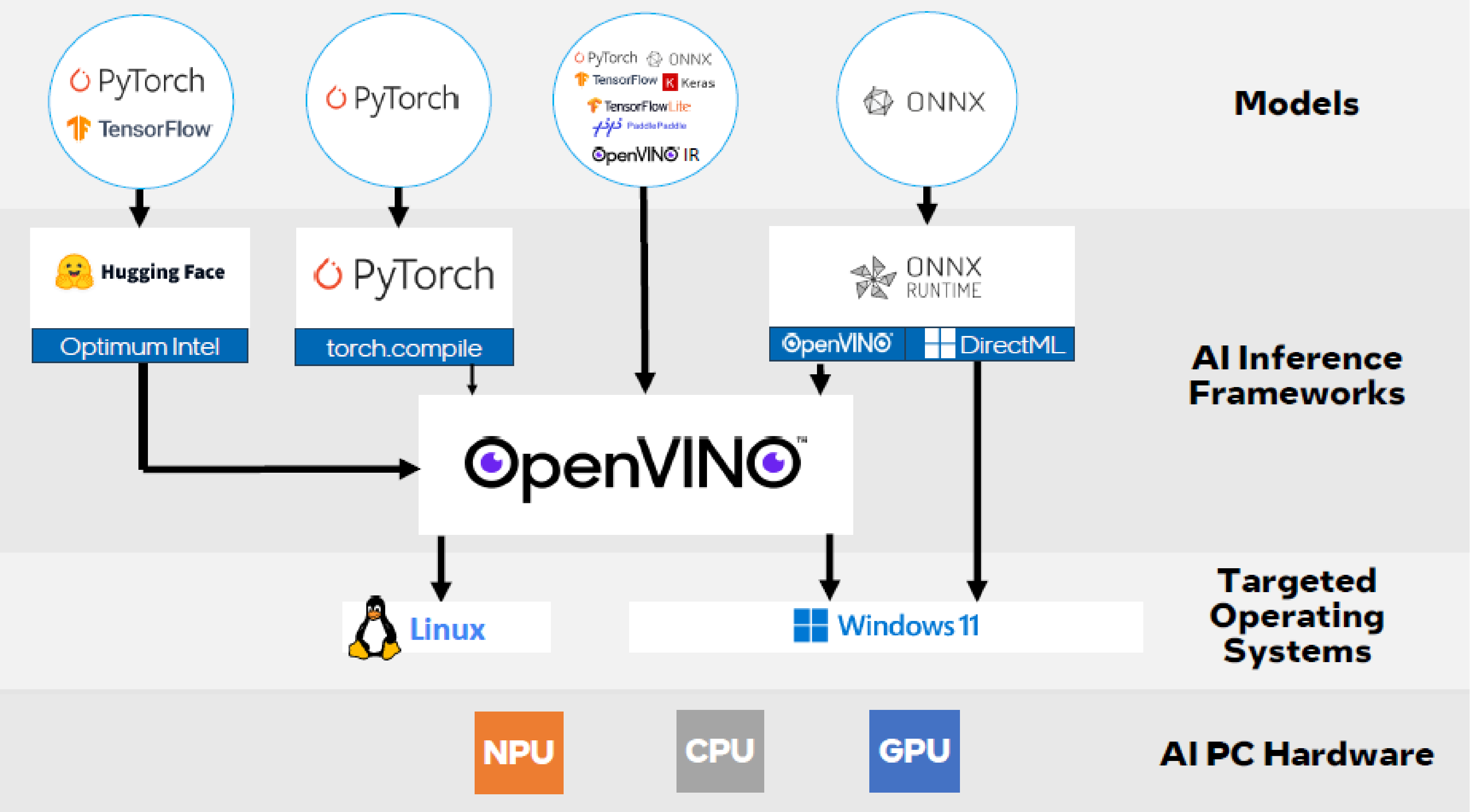
E-Core x8

影像來源：https://www.4gamers.com.tw/news/detail/59826/intel-meteor-lake-architecture-overview

常見文字生成應用

- 文章生成
- 對話聊天
- 長文摘要
- 語言翻譯
- 資料查詢
- 寫作助理

OpenVINO 推論架構



範例1 – 語句情緒辨識

Hugging Face Model Hub with OpenVINO

Converting a Model from the HF Transformers Package

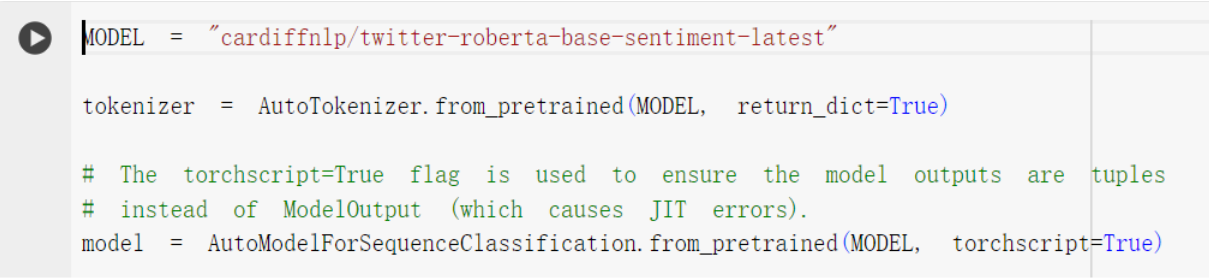
- Installing Requirements
- Imports
- Initializing a Model Using the HF Transformers Package
- Original Model inference
- Converting the Model to OpenVINO IR format
- Converted Model Inference

Converting a Model Using the Optimum Intel Package

- Install Requirements for Optimum
- Import Optimum
- Initialize and Convert the Model Automatically using OVModel class
- Convert model using Optimum CLI interface
- The Optimum Model Inference

https://colab.research.google.com/github/openvinotoolkit/openvino_notebooks/blob/latest/notebooks/hugging-face-hub/hugging-face-hub.ipynb#Converting-the-Model-to-OpenVINO-IR-format

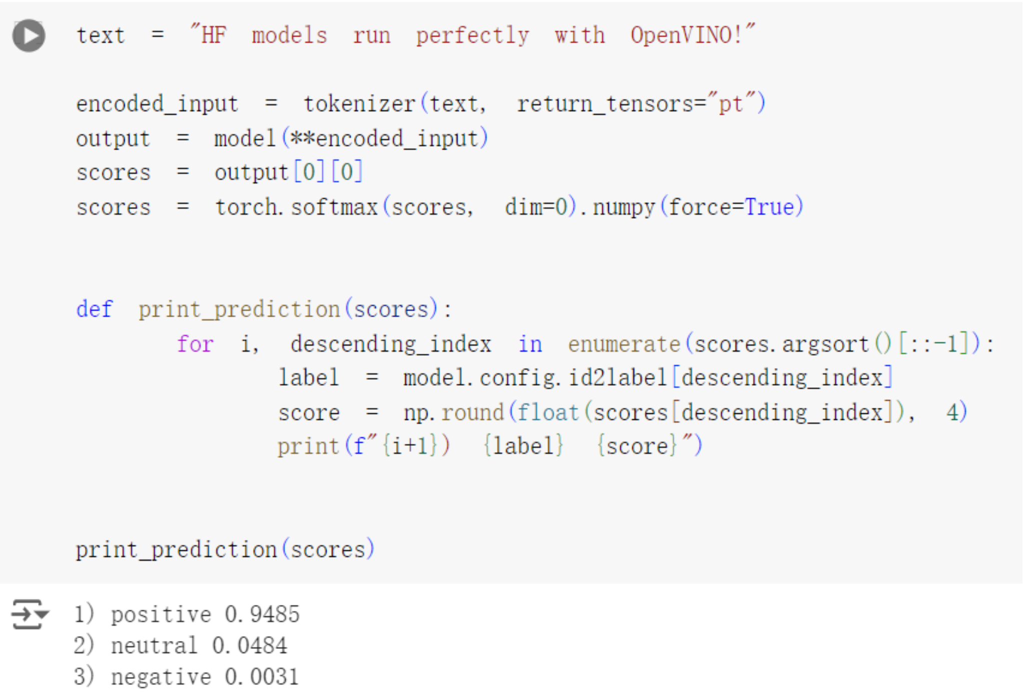
範例1 — 語句情緒辨識（HF）



```
MODEL = "cardiffnlp/twitter-roberta-base-sentiment-latest"

tokenizer = AutoTokenizer.from_pretrained(MODEL, return_dict=True)

# The torchscript=True flag is used to ensure the model outputs are tuples
# instead of ModelOutput (which causes JIT errors).
model = AutoModelForSequenceClassification.from_pretrained(MODEL, torchscript=True)
```

載入Hugging Face模型

"cardiffnlp/twitter-roberta-base-sentiment-latest"



```
text = "HF models run perfectly with OpenVINO!"

encoded_input = tokenizer(text, return_tensors="pt")
output = model(**encoded_input)
scores = output[0][0]
scores = torch.softmax(scores, dim=0).numpy(force=True)


def print_prediction(scores):
    for i, descending_index in enumerate(scores.argsort()[::-1]):
        label = model.config.id2label[descending_index]
        score = np.round(float(scores[descending_index]), 4)
        print(f"{i+1}) {label} {score}")


print_prediction(scores)
```

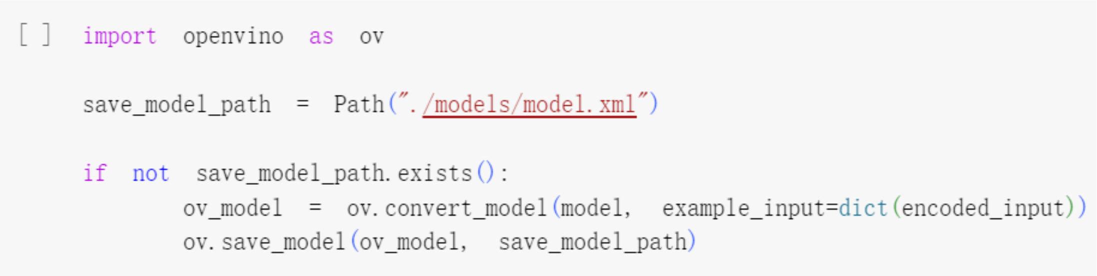
```
1) positive 0.9485
2) neutral 0.0484
3) negative 0.0031
```

提供輸入文字

進行推論

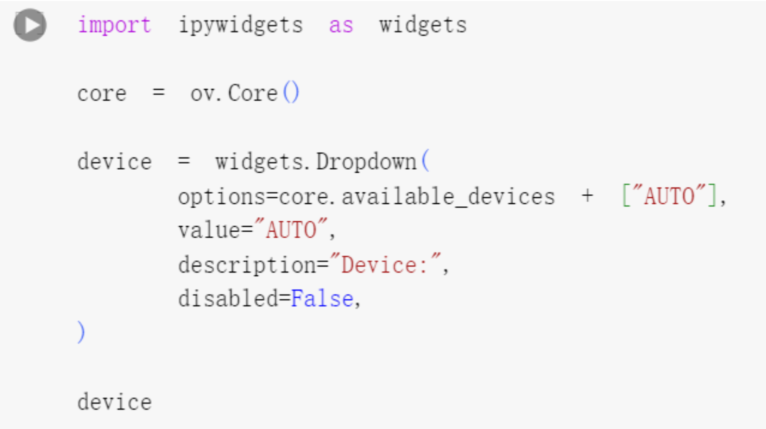
顯示結果

範例1 – 語句情緒辨識（IR）



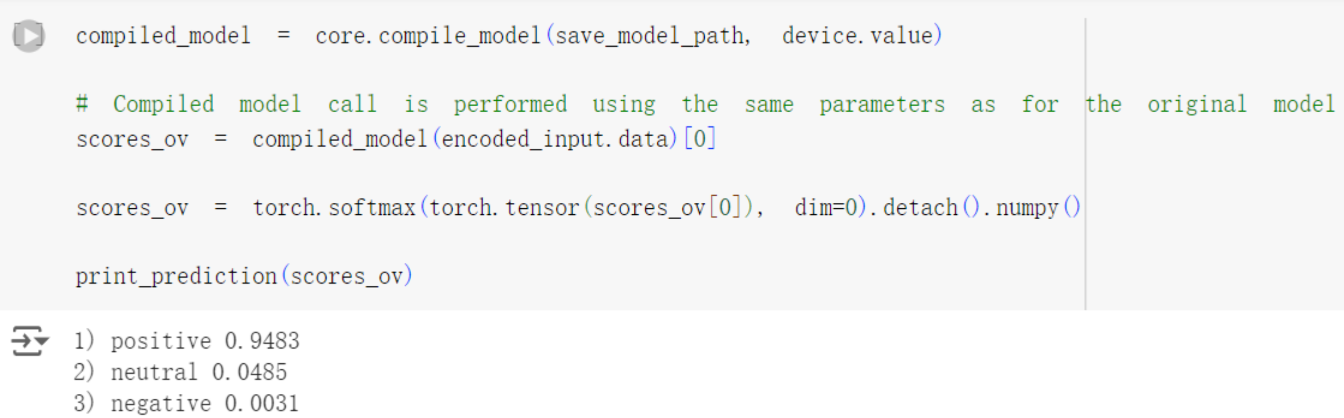
```
[ ] import openvino as ov

    save_model_path = Path("./models/model.xml")

    if not save_model_path.exists():
        ov_model = ov.convert_model(model, example_input=dict(encoded_input))
        ov.save_model(ov_model, save_model_path)
```

轉換並另存模型



```
import ipywidgets as widgets

core = ov.Core()

device = widgets.Dropdown(
    options=core.available_devices + ["AUTO"],
    value="AUTO",
    description="Device:",
    disabled=False,
)

device
```

指定推論裝置

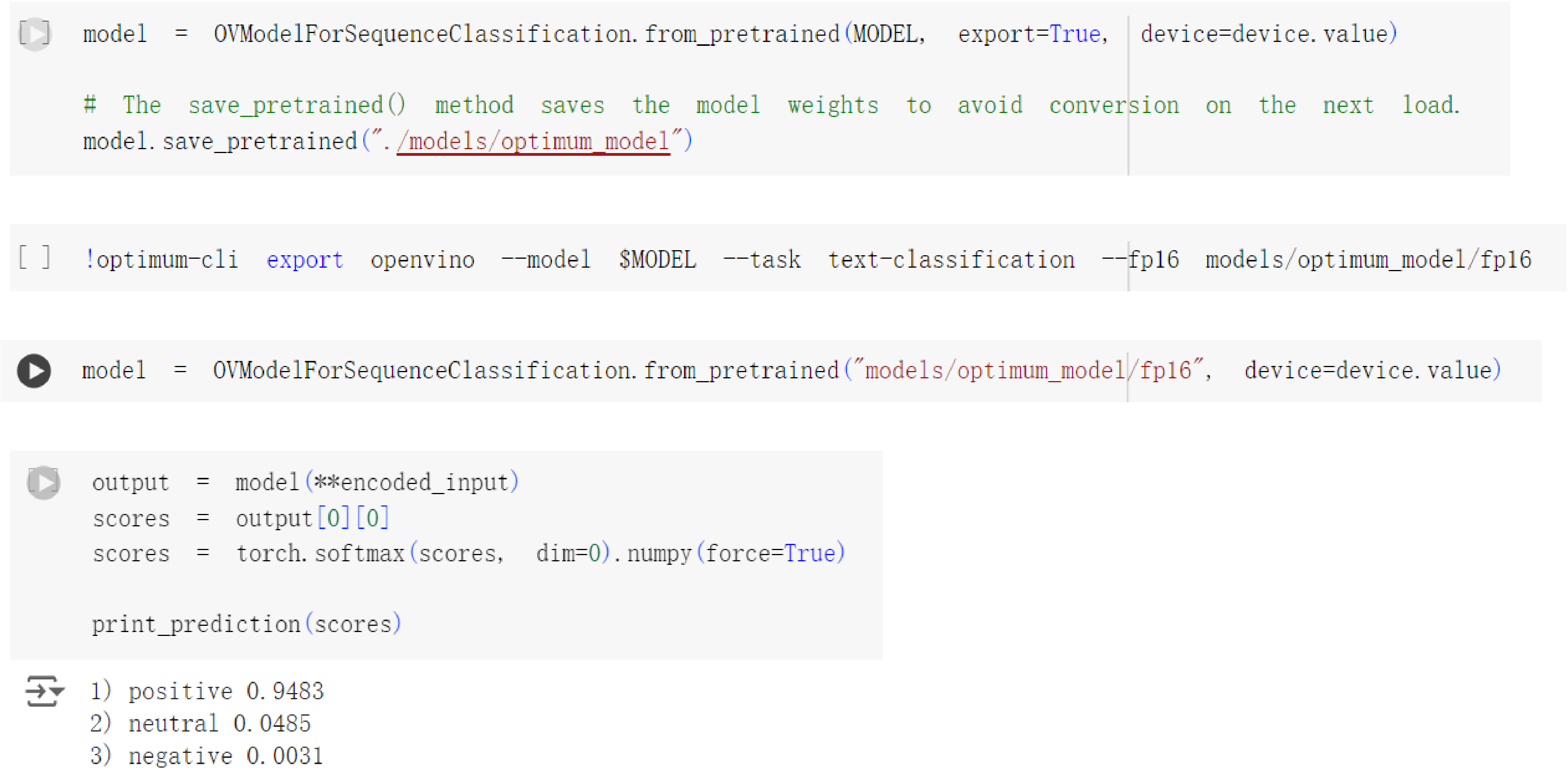


```
compiled_model = core.compile_model(save_model_path, device.value)

# Compiled model call is performed using the same parameters as for the original model
scores_ov = compiled_model(encoded_input.data)[0]

scores_ov = torch.softmax(torch.tensor(scores_ov[0]), dim=0).detach().numpy()

print_prediction(scores_ov)
```

```
1) positive 0.9483
2) neutral 0.0485
3) negative 0.0031
```

進行推論

範例1 – 語句情緒辨識（Optimum）



```
model = OVModelForSequenceClassification.from_pretrained(MODEL, export=True, device=device.value)

# The save_pretrained() method saves the model weights to avoid conversion on the next load.
model.save_pretrained("./models/optimum_model")
```

```
!optimum-cli export openvino --model $MODEL --task text-classification --fp16 models/optimum_model/fp16
```

```
model = OVModelForSequenceClassification.from_pretrained("models/optimum_model/fp16", device=device.value)
```

```
output = model(**encoded_input)
scores = output[0][0]
scores = torch.softmax(scores, dim=0).numpy(force=True)

print_prediction(scores)
```

```
1) positive 0.9483
2) neutral 0.0485
3) negative 0.0031
```

範例2 — LLM Chatbot

Create an LLM-powered Chatbot using OpenVINO

- Prerequisites
- Select model for inference ←
- Convert model using Optimum-CLI tool
- Compress model weights
  - Weights Compression using Optimum-CLI
  - Weight compression with AWQ
- Select device for inference and model variant
- Instantiate Model using Optimum Intel
- Run Chatbot

支援模型：

- **tiny-llama-1b-chat**
- **mini-cpm-2b-dpo**
- **gemma-2b-it**
- **phi3-mini-instruct**
- **red-pajama-3b-chat**
- **gemma-7b-it**
- **llama-2-7b-chat**
- **llama-3-8b-instruct**
- **qwen2-1.5b-instruct/qwen2-7b-instruct**
- **qwen1.5-0.5b-chat/qwen1.5-1.8b-chat/qwen1.5-7b-chat**
- **qwen-7b-chat**
- **mpt-7b-chat**
- **chatglm3-6b**
- **mistral-7b**
- **zephyr-7b-beta**
- **neural-chat-7b-v3-1**
- **notus-7b-v1**
- **youri-7b-chat**
- **baichuan2-7b-chat**
- **internlm2-chat-1.8b**

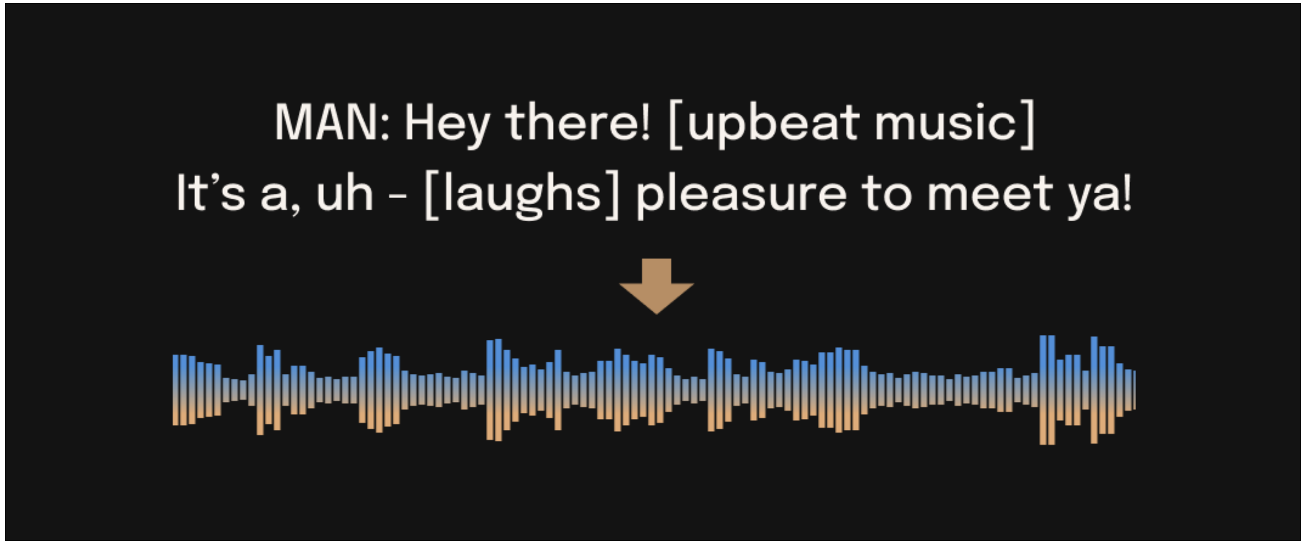
https://github.com/openvinotoolkit/openvino_notebooks/blob/latest/notebooks/llm-chatbot/llm-chatbot.ipynb **(不支援Colab)**

常見語音生成應用

自動讀稿

抑揚頓挫

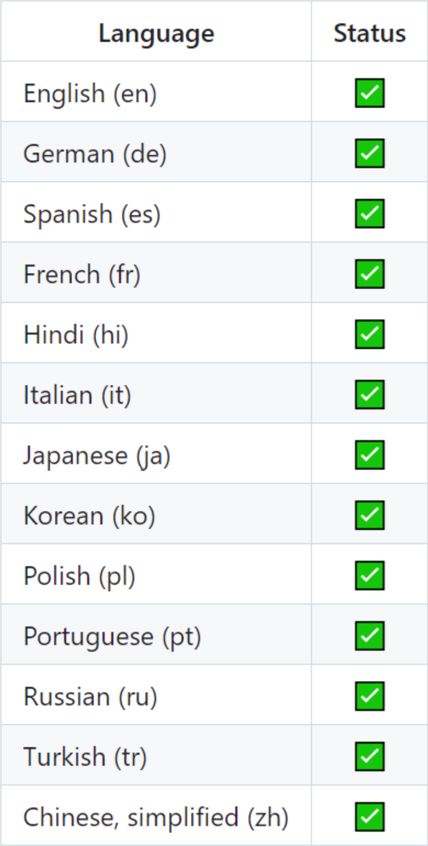
角色情緒



MAN: Hey there! [upbeat music]
It’s a, uh – [laughs] pleasure to meet ya!

文字轉語音模型 Suno-ai Bark

- 可讀多國文字

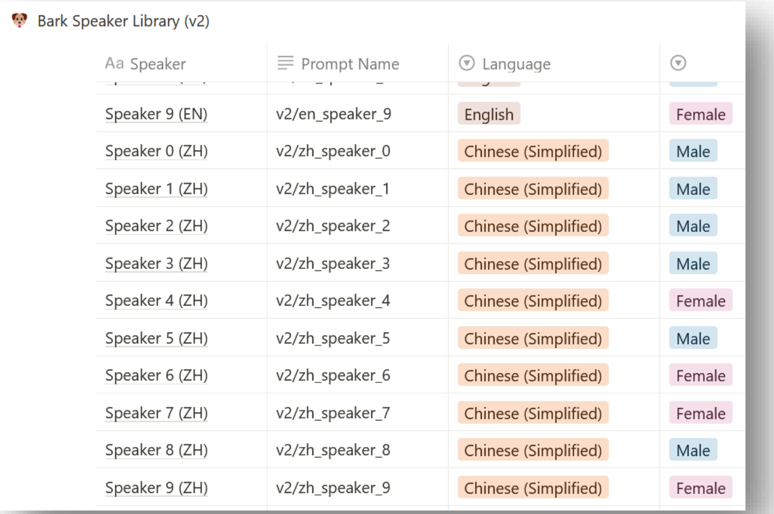


Language	Status
English (en)	✓
German (de)	✓
Spanish (es)	✓
French (fr)	✓
Hindi (hi)	✓
Italian (it)	✓
Japanese (ja)	✓
Korean (ko)	✓
Polish (pl)	✓
Portuguese (pt)	✓
Russian (ru)	✓
Turkish (tr)	✓
Chinese, simplified (zh)	✓

- 可加入文字細節
  - [laughter]
  - [laughs]
  - [sighs]
  - [music]
  - [gasps]
  - [clears throat]
  - — or ... for hesitations

- 可唱出文字
  - 在文字前後加上♪

- 可指定語者（含性別）



Bark Speaker Library (v2)

Speaker	Prompt Name	Language	
Speaker 9 (EN)	v2/en_speaker_9	English	Female
Speaker 0 (ZH)	v2/zh_speaker_0	Chinese (Simplified)	Male
Speaker 1 (ZH)	v2/zh_speaker_1	Chinese (Simplified)	Male
Speaker 2 (ZH)	v2/zh_speaker_2	Chinese (Simplified)	Male
Speaker 3 (ZH)	v2/zh_speaker_3	Chinese (Simplified)	Male
Speaker 4 (ZH)	v2/zh_speaker_4	Chinese (Simplified)	Female
Speaker 5 (ZH)	v2/zh_speaker_5	Chinese (Simplified)	Male
Speaker 6 (ZH)	v2/zh_speaker_6	Chinese (Simplified)	Female
Speaker 7 (ZH)	v2/zh_speaker_7	Chinese (Simplified)	Female
Speaker 8 (ZH)	v2/zh_speaker_8	Chinese (Simplified)	Male
Speaker 9 (ZH)	v2/zh_speaker_9	Chinese (Simplified)	Female

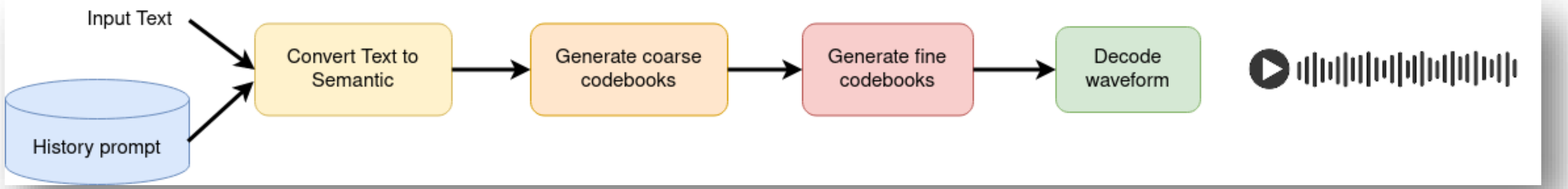
https://github.com/suno-ai/bark

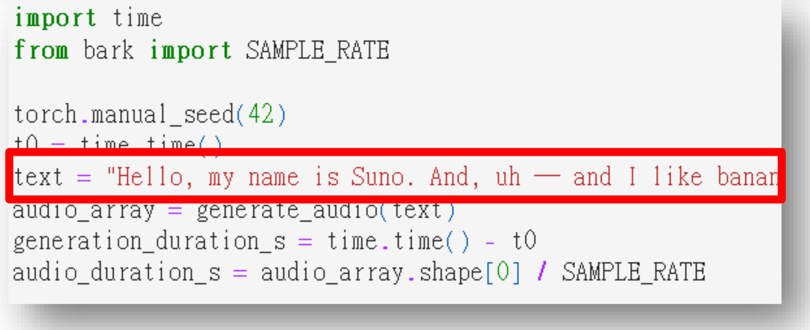
語音生成應用實例 – TTS Bark

Text-to-speech generation using Bark and OpenVINO

- Prerequisites
- Download and Convert models
  - Text Encoder
  - Coarse encoder
  - Fine encoder
  - Prepare Inference pipeline
- Run model inference
  - Select Inference device
- Interactive demo

模型		
	text.pt	2.32G Byte
	coarse.pt	1.25G Byte
	fine_2.pt	3.74G Byte



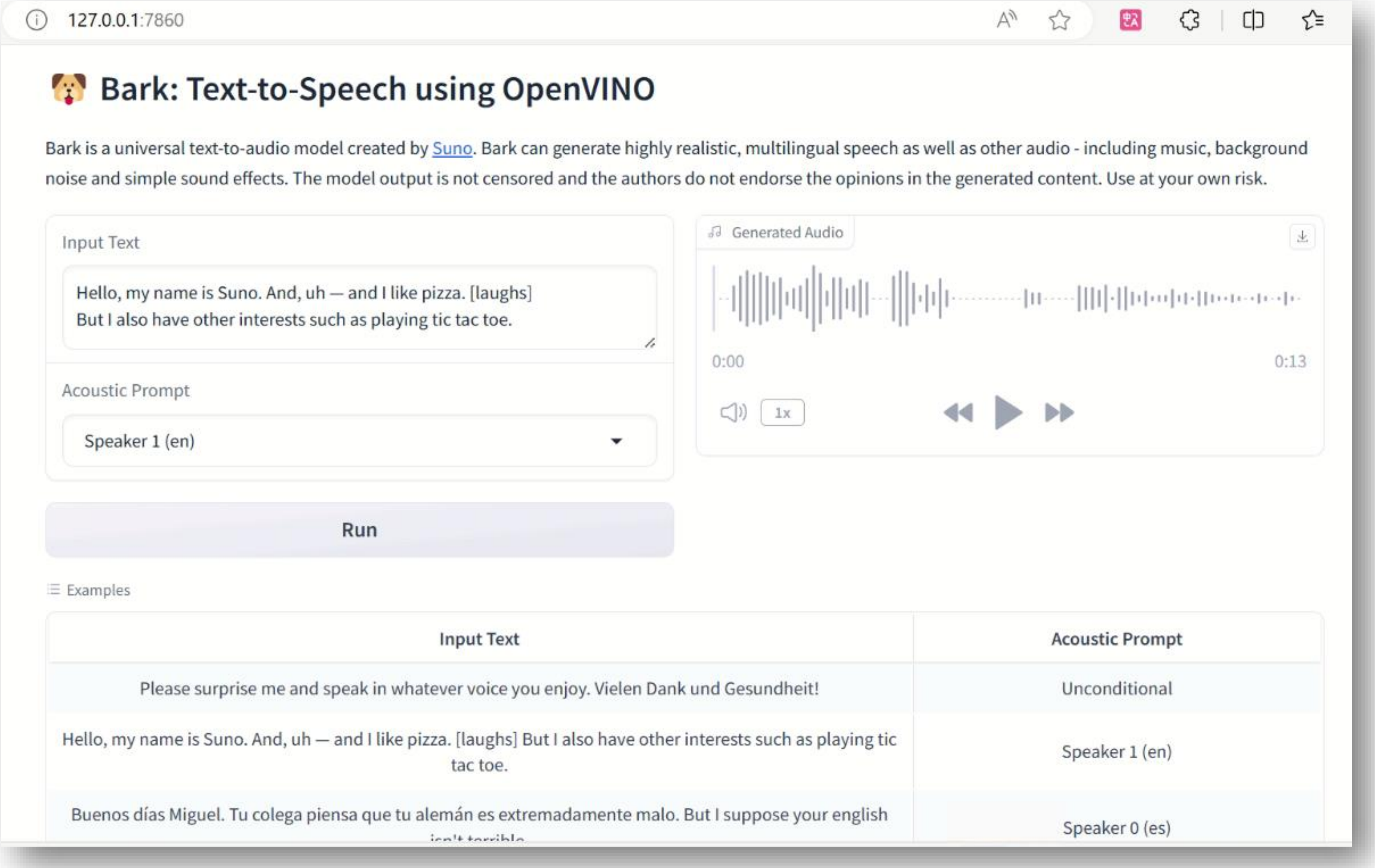


```
import time
from bark import SAMPLE_RATE

torch.manual_seed(42)
t0 = time.time()
text = "Hello, my name is Suno. And, uh — and I like banan
audio_array = generate_audio(text)
generation_duration_s = time.time() - t0
audio_duration_s = audio_array.shape[0] / SAMPLE_RATE
```

輸入文字

https://github.com/openvinotoolkit/openvino_notebooks/blob/latest/notebooks/bark-text-to-audio/bark-text-to-audio.ipynb (不支援Colab)

語音生成應用實例 – TTS Bark GUI



🐶 Bark: Text-to-Speech using OpenVINO

Bark is a universal text-to-audio model created by Suno. Bark can generate highly realistic, multilingual speech as well as other audio - including music, background noise and simple sound effects. The model output is not censored and the authors do not endorse the opinions in the generated content. Use at your own risk.

Input Text

Hello, my name is Suno. And, uh — and I like pizza. [laughs]
But I also have other interests such as playing tic tac toe.

Acoustic Prompt

Speaker 1 (en)

Examples

Input Text	Acoustic Prompt
Please surprise me and speak in whatever voice you enjoy. Vielen Dank und Gesundheit!	Unconditional
Hello, my name is Suno. And, uh — and I like pizza. [laughs] But I also have other interests such as playing tic tac toe.	Speaker 1 (en)
Buenos días Miguel. Tu colega piensa que tu alemán es extremadamente malo. But I suppose your english isn't terrible.	Speaker 0 (es)

參考文獻

- 許哲豪，臺灣科技大學資訊工程系「人工智慧與邊緣運算實務」（2021~2023）

https://omnixri.blogspot.com/p/ntust-edge-ai.html

- 許哲豪，【課程簡報】20231209_DevFest Taichung_如何結合Google Colab及Intel OpenVINO來玩轉AIGC

https://omnixri.blogspot.com/2023/12/20231209devfest-taichunggoogle.html

- Jingfeng Yang etc., Harnessing the Power of LLMs in Practice: A Survey on ChatGPT and Beyond

https://arxiv.org/abs/2304.13712

延伸閱讀

- Intel OpenVINO DevCon （Youtube 中文講座）

https://www.youtube.com/watch?v=jnYNJVvghgE&list=PLJhgRo1wc4K9LRAUUgG-48BxJVqXEXhhH

- Intel OpenVINO™ 生成式 AI 系列 (Bilibili 教學影片)

https://space.bilibili.com/38566875/channel/collectiondetail?sid=2301246

- 許哲豪，【vMaker Edge AI專欄 #15】 從MWC 2024看AI手機未來發展

https://omnixri.blogspot.com/2024/03/vmaker-edge-ai-15-mwc-2024ai.html

- 許哲豪，【vMaker Edge AI專欄 #17】 開發者如何選擇 Edge AI 開發方案

https://omnixri.blogspot.com/2024/05/vmaker-edge-ai-17-edge-ai.html

歐尼克斯實境互動工作室
(OmniXRI Studio)
許哲豪 (Jack Hsu)

Facebook : Jack Omnixri
FB社團： Edge AI Taiwan邊緣智能交流區
電子信箱：omnixri@gmail.com
部落格：https://omnixri.blogspot.tw
開　源：https://github.com/OmniXRI
YOUTUBE 直播： https://www.youtube.com/@omnixri1784/streams