



# LLM圖片理解服務

## LangChain + Gemini / Azure OpenAI

20260108



# LLM圖片理解服務

## LangChain + Gemini



# Gemini 限制

- 支援格式
  - PNG
  - JPEG
  - WebP

Property	Gemini 3 Pro, Flash	Gemini 3 Pro Image	Gemini 2.5 Pro, Flash, Flash-Lite	Gemini 2.5 Flash Image	Gemini 2.5 Flash- Live ***
Input token limit *	1,048,576 tokens	65,536 tokens	1,048,576 tokens	32,768 tokens	32K (default; upgradable to 128K) tokens
Output token limit *	65,536 tokens	32,768 tokens	65,536 tokens	8,192 tokens	64K tokens
Knowledge cutoff date	January 2025	January 2025	January 2025	June 2025	January 2025
<b>Images (per request)</b>					
Max number of <i>input</i> images	1,000 images	14 images	3,000 images	3 images	---
Max number of <i>output</i> images	---	10 images	---	10 images	---
Max size per input base64-encoded image	7 MB	7 MB	7 MB	7 MB	---

<https://firebase.google.com/docs/vertex-ai/gemini-models>



# config.ini

[Gemini]

API\_KEY = XXXXXXXXXX



# app.py

```
from langchain_core.messages import HumanMessage
from langchain_google_genai import ChatGoogleGenerativeAI
from IPython.display import Image, display
from configparser import ConfigParser

config = ConfigParser()
config.read("config.ini")

llm = ChatGoogleGenerativeAI(
    model="gemini-2.5-flash",
    google_api_key=config["Gemini"]["API_KEY"],
)

user_messages = []
# append user input question
user_input = "圖片中的生物是什麼？"
user_messages.append({"type": "text", "text": user_input + "請使用繁體中文回答。"})
```



# app.py

```
# append images
image_url = "https://i.ibb.co/KyNtMw5/IMG-20240321-172354614-AE.jpg"
user_messages.append({
    "type" : "image_url",
    "image_url" : {"url":image_url}
})

human_messages = HumanMessage(
    content = user_messages
)
result = llm.invoke([human_messages])

print("Q: " + user_input)
print("A: " + result.content)

# Display the image
display(Image(url=image_url))
```

<https://gist.github.com/ryanchung403/ed6f0abd79ad91702955056b15b20d95>

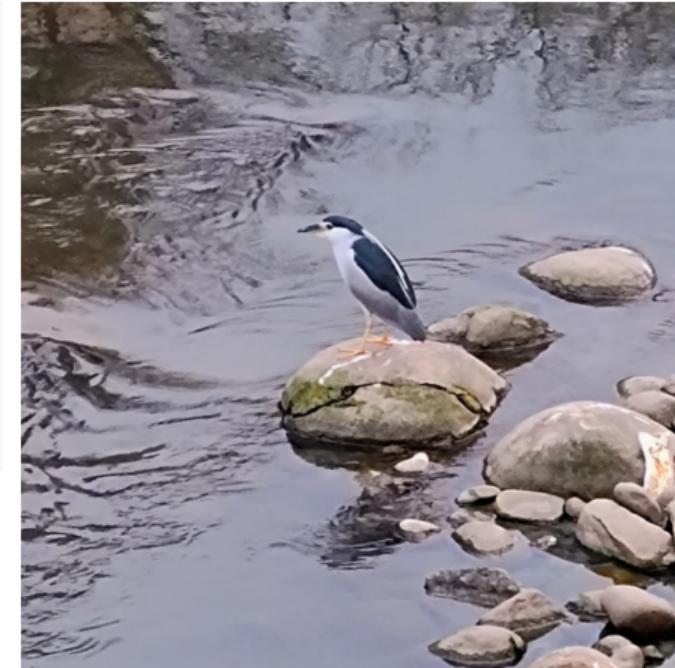
# 觀察結果 – 網址圖片

Q: 圖片中的生物是什麼？

A: 圖片中的生物是\*\*夜鷺\*\* (Black-crowned Night Heron)。

牠的特徵包括：

- \* 頭頂和背部呈黑色。
- \* 臉部和腹部為白色。
- \* 翅膀為灰黑色。
- \* 腳為黃色或橘黃色。



<https://i.ibb.co/KyNtMw5/IMG-20240321-172354614-AE.jpg>



# 測試本地端圖片

- 先到網路上下載一張圖片至開發資料夾中
- 或是準備好一張圖片放在開發資料夾中



# 本地端圖片 -> 格式轉換 -> 添加

```
from langchain_core.messages import HumanMessage
from langchain_google_genai import ChatGoogleGenerativeAI
from IPython.display import Image, display
from configparser import ConfigParser
import base64

config = ConfigParser()
config.read("config.ini")

llm = ChatGoogleGenerativeAI(
    model="gemini-2.5-flash",
    google_api_key=config["Gemini"]["API_KEY"],
    max_tokens=8192,
)
def image4LangChain(image_url):
    if "http" in image_url:
        return {"url":image_url}
    else:
        with open(image_url, "rb") as image_file:
            image_data = base64.b64encode(image_file.read()).decode("utf-8")
        return {"url":f"data:image/jpeg;base64,{image_data}"}
```



# 新版LangChain會擋本地端檔案

```
user_messages = []
# append user input question
user_input = "圖片中的生物是什麼？請詳細描述。"
user_messages.append({"type": "text", "text": user_input + "請使用繁體中文回答。"})
# append images
# image_url = "https://i.ibb.co/KyNtMw5/IMG-20240321-172354614-AE.jpg"
image_url = "cat.jpg"

user_messages.append(
    {
        "type": "image_url",
        "image_url": image4LangChain(image_url),
    }
)

human_messages = HumanMessage(content=user_messages)
result = llm.invoke([human_messages])

print("Q: " + user_input)
print("A: " + result.content)
```

<https://gist.github.com/ryanchung403/8ffefa3fa3df7b7d6372270a756577d3>

# 本地端圖片理解

Q: 圖片中的生物是什麼？請詳細描述。

A: 圖片中的生物是一隻\*\*橘色虎斑貓 (Orange Tabby Cat)\*\*。

以下是牠的詳細描述：

1. \*\*物種識別\*\*：這是一隻家貓，從其毛色和花紋判斷，屬於橘色虎斑貓。
2. \*\*毛色與花紋\*\*：
  - \* 牠的毛色主體是明亮的橘色或薑黃色。
  - \* 全身覆蓋著清晰可見的深色虎斑紋路，這些紋路在額頭、臉頰和頸部尤為明顯，呈現出深淺不一的橫棕色條紋。額頭上似乎有典型的「M」形虎斑紋。
3. \*\*眼睛\*\*：
  - \* 牠擁有一雙引人注目的綠色眼睛，顏色介於翠綠色和淺黃綠色之間。
  - \* 瞳孔呈垂直的狹縫狀，顯示牠處於明亮的光線下或警覺狀態。
  - \* 眼神專注、明亮，直視前方。
4. \*\*鼻子\*\*：
  - \* 鼻子是粉紅色的，小巧而略帶濕潤感。
5. \*\*鬍鬚\*\*：
  - \* 兩頰長著許多又長又白的鬍鬚，從鼻翼兩側向外伸展，非常顯眼，是貓咪重要的感官器官。
6. \*\*耳朵\*\*：
  - \* 耳朵直立且尖，呈三角形。
  - \* 外側覆蓋著與身體相同的橘色短毛，內側則可見粉紅色的皮膚，且有一些細小的白色絨毛。
7. \*\*臉部特徵\*\*：
  - \* 臉型圓潤，表情平靜而警覺。
  - \* 嘴巴閉合，嘴角線條柔和。
8. \*\*身體與姿態\*\*：
  - \* 圖片主要聚焦在貓的頭部和部分頸部及肩部。
  - \* 從可見的部分判斷，牠的毛髮短而密，看起來健康且有光澤。
  - \* 牠似乎正坐著或站著，身體朝向鏡頭。
9. \*\*背景\*\*：
  - \* 背景是模糊的綠色，呈現出柔和的散景效果，突顯了貓咪作為主體。這暗示牠可能在戶外，例如草地或灌木叢前。

總體而言，這是一隻美麗、毛色鮮豔且神情專注的橘色虎斑貓。





# LLM圖片理解服務

## LangChain + Azure OpenAI



# 設定 config.ini

[AzureOpenAI]

ENDPOINT = <https://xxxxxxxx.azure.com/openai/v1/>

KEY = xxxxxxxxx

GPT4o\_DEPLOYMENT\_NAME = xxxxxxxx



# 使用 Azure OpenAI - GTP4o-mini

- 可使用相同的human\_messages

```
from langchain_openai import ChatOpenAI

llm_gpt4o_mini = ChatOpenAI(
    model=config["AzureOpenAI"]["GPT4o_DEPLOYMENT_NAME"],
    base_url=config["AzureOpenAI"]["ENDPOINT"],
    api_key=config["AzureOpenAI"]["KEY"],
)
result_gpt4o_mini = llm_gpt4o_mini.invoke([human_messages])
print("Q: " + user_input)
print("A: " + result_gpt4o_mini.content)

# Display the image
display(Image(url=image_url))
```

<https://gist.github.com/ryanchung403/cba1a0999a836f4bba6ef516e19f5f81>



# 補充：Gemini 原生寫法(不用LangChain)

```
from google import genai
from google.genai import types
import requests
from configparser import ConfigParser
from IPython.display import Image, display

config = ConfigParser()
config.read("config.ini")

image_url = "https://i.ibb.co/KyNtMw5/IMG-20240321-172354614-AE.jpg"
image_url_2 = "cat.jpg"

def image4Gemini(this_image):
    if "http" in this_image:
        image_bytes = requests.get(this_image).content
    else:
        with open(this_image, "rb") as image_file:
            image_bytes = image_file.read()
    image = types.Part.from_bytes(
        data=image_bytes, mime_type="image/jpeg"
    )
    return image
```



# 補充：Gemini 原生寫法(不用LangChain)

```
client = genai.Client(  
    api_key = config["Gemini"]["API_KEY"]  
)  
  
user_question = """  
請描述這張圖片。  
使用繁體中文回答。  
"""  
  
response = client.models.generate_content(  
    model='gemini-2.5-flash',  
    contents=[  
        image4Gemini(image_url),  
        user_question  
    ]  
)  
  
print(f"Q: {user_question}")  
print(f"Gemini: {response.text}")  
# Display the image  
display(Image(url=image_url))
```

<https://gist.github.com/ryan403/e7f9b24c0de2da35932c8441c41e9cdd>



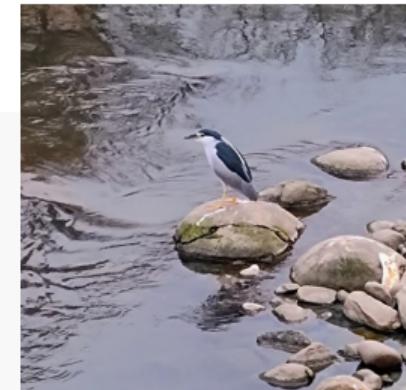
# 觀察結果

Q:

請描述這張圖片。

使用繁體中文回答。

Gemini: 這張圖片展示了一幅寧靜的自然景象：一隻鳥站在流動的河水中的一顆石頭上。



\*\*主體描述：\*\*

- \* \*\*鳥：\*\* 這是一隻黑冠麻鷺（Black-crowned Night Heron）。牠的頭頂和背部呈現深藍黑色，臉部和頸部下方是顯眼的白色，而腹部則是淺灰色。牠的腳呈鮮明的橙黃色。
- \* \*\*姿態：\*\* 這隻鳥正筆直地站立著，身體挺拔，頭部微微轉向左側，眼神專注地凝視著前方的水面，似乎在觀察或尋找獵物。

\*\*環境描述：\*\*

- \* \*\*石頭：\*\* 牠所站立的石頭圓潤而平滑，表面覆蓋著一些綠色的青苔，中間有一道明顯的裂痕。
- \* \*\*河水：\*\* 鳥的周圍是平靜而緩緩流動的河水。水面呈現灰藍色調，反射著天空的光線，形成細微的波紋和漣漪，顯示出水流的輕微移動。
- \* \*\*其他石頭：\*\* 水中還散佈著許多大小不一的鵝卵石或圓石，有些部分露出水面，有些則完全沒入水中，與水中的倒影交織在一起。
- \* \*\*背景：\*\* 遠處的背景是更廣闊的水面，顯得有些模糊，進一步襯托出前景中的鳥和石頭。

\*\*整體氛圍：\*\*

整幅畫面給人一種自然、寧靜與和諧的感覺，展現了野生動物在自然棲息地中靜謐的一刻。

THE END!