

```
import asyncio
```

```
import base64
```

```
import time
```

```
from io import BytesIO
```

```
import aiohttp
```

```
import requests
```

```
from dotenv import load_dotenv
```

```
from PIL import Image
```

```
# Load environment variables
```

```
load_dotenv()
```

```
# Swarms Cloud API key
```

```
swarms_cloud_api_key = ""
```

```
# Convert image to Base64
```

```
def image_to_base64(image_path):
```

```
    with Image.open(image_path) as image:
```

```
        buffered = BytesIO()
```

```
        image.save(buffered, format="JPEG")
```

```
        img_str = base64.b64encode(buffered.getvalue()).decode("utf-8")
```

```
    return img_str
```

```
# Replace 'image.jpg' with the path to your image

base64_image = image_to_base64("trump.jpeg")

text_data = {

    "type": "text",

    "text": "Describe who is in the image, what is his name exactly?",

}

image_data = {

    "type": "image_url",

    "image_url": {"url": f"data:image/jpeg;base64,{base64_image}"},

}
```

Construct the request data

```
request_data = {

    "model": "cogvlm-chat-17b",

    "messages": [{"role": "user", "content": [text_data, image_data]}],

    "temperature": 0.2,

    "top_p": 0.9,

    "max_tokens": 8000,

}
```

```
headers = {

    "Authorization": f"Bearer {str(swarms_cloud_api_key)}",

    "Content-Type": "application/json",

}
```

Specify the URL of your FastAPI application

```
# url = "http://34.227.161.100:30001/v1/chat/completions"
```

```
url = "https://api.swarms.world/v1/chat/completions"
```

```
# Start the timer
```

```
start_time = time.time()
```

```
# Send the request
```

```
response = requests.post(url, json=request_data, headers=headers) # headers=headers)
```

```
# Stop the timer
```

```
end_time = time.time()
```

```
time_taken = end_time - start_time
```

```
# Print the response from the server
```

```
print(response.text)
```

```
print(f"Time taken: {time_taken} seconds")
```

```
print("Asyncio version")
```

```
# Start the timer
```

```
start_time = time.time()
```

```
async def send_request(session, url, headers, data):
```

```
    async with session.post(url, headers=headers, json=data) as response:
```

```
        return await response.text()
```

```
async def main():

    async with aiohttp.ClientSession() as session:

        tasks = []

        start_time = time.time() # Start the timer

        for _ in range(1):

            task = send_request(session, url, headers, request_data)

            tasks.append(task)

        responses = await asyncio.gather(*tasks)

        end_time = time.time() # Stop the timer

        for response in responses:

            print(response)

        time_taken = end_time - start_time # Calculate the time taken

        print(f"Time taken: {time_taken} seconds")


# Run the main function

asyncio.run(main())
```