

OpenAI API

OpenAI API, một công cụ cực kỳ mạnh mẽ, linh hoạt và đang tạo nên những bước đột phá trong thế giới công nghệ hiện đại.

- [Giới thiệu](#)
- [OpenAI API for Text](#)
- [Tùy chỉnh đầu ra bằng các tham số trong OpenAI API](#)
- [Sử Dụng OpenAI API Để Hiểu, Phân Tích Và Mô Tả Hình Ảnh](#)

Giới thiệu

Tại sao phải học về API? Tại sao OpenAI API lại quan trọng đến vậy?

Hãy tưởng tượng như thế này: nếu trước đây chúng ta chỉ học cách sử dụng AI một cách đơn giản, thì bây giờ với OpenAI API, **bạn sẽ làm chủ một công cụ có thể tùy biến và xây dựng những hệ thống thông minh thực sự - từ chatbot, trợ lý ảo, cho đến các ứng dụng xử lý văn bản, hình ảnh và nhiều hơn nữa.**

Tăng tốc với công nghệ mới

Chúng ta đã từng cùng nhau tạo ra các hệ thống gợi ý cơ bản, tìm hiểu về các mô hình GPT và các khái niệm nền tảng. Nhưng giờ đây, **OpenAI API giống như nhiên liệu tên lửa**, giúp mọi thứ bạn học được **"cất cánh" lên một tầm cao mới.**

Với API này, bạn có thể:

- Tạo chatbot tương tác thông minh và hiểu ngữ cảnh
- Xây dựng trợ lý ảo cá nhân hóa
- Tự động hóa việc tạo nội dung bằng văn bản
- Phân tích dữ liệu hình ảnh và tạo ứng dụng đa phương tiện
- Và còn rất nhiều điều đang chờ bạn khám phá

Công nghệ mà các công ty hàng đầu thế giới đang sử dụng

Đây **không chỉ là công cụ học tập** - nó chính là công nghệ mà các tập đoàn lớn trên toàn cầu đang sử dụng để đổi mới, tăng tốc phát triển và tạo ra lợi thế cạnh tranh. Và **bạn cũng sẽ có trong tay những kỹ năng đó**, để ứng dụng vào công việc thực tế hoặc những dự án cá nhân đầy tiềm năng.

Hành trình thực hành - từ cài đặt đến triển khai

Ở phần tiếp theo, chúng ta sẽ:

- Thiết lập môi trường làm việc và lấy API Key
- Học cách gửi yêu cầu đến API và nhận phản hồi thông minh
- Viết prompt hiệu quả và điều chỉnh các tham số để có đầu ra tối ưu
- Khám phá khả năng xử lý cả văn bản và hình ảnh - tạo ứng dụng **đa phương thức (multimodal)**

Sáng tạo không giới hạn

Tôi muốn bạn cảm thấy **thật sự hứng khởi**, bởi vì **bạn đang bước vào một trong những lĩnh vực tiên tiến nhất của công nghệ hiện nay**. Và đây là cơ hội để bạn **sáng tạo không giới hạn**, mở rộng tư duy, và có thể bắt đầu xây dựng những ứng dụng AI thực sự thông minh.

Hãy nghĩ xem: bạn sẽ ứng dụng API này vào đâu? Một hệ thống chăm sóc khách hàng thông minh? Một công cụ tạo nội dung tự động? Hay một trợ lý AI cho doanh nghiệp của bạn?

Mục tiêu cuối cùng

Sau phần này, bạn sẽ:

- Hiểu cách hoạt động của OpenAI API
- Biết cách tích hợp nó vào dự án thực tế
- Và quan trọng nhất: **biết cách phát huy tối đa tiềm năng của công nghệ AI trong tay bạn**

Tác giả: Đỗ Ngọc Tú
Công Ty Phần Mềm VHTSoft

OpenAI API for Text

Mục tiêu bài học

Trong bài học này, bạn sẽ:

- Hiểu cách hoạt động cơ bản của **OpenAI API**
- Biết cách tạo **ứng dụng tạo văn bản** bằng `chat.completions` endpoint
- Nắm vai trò quan trọng của **system prompt**
- Cài đặt và kết nối với API để bắt đầu tạo phản hồi giống như ChatGPT

1. Tổng quan về OpenAI API

OpenAI cung cấp một bộ công cụ mạnh mẽ cho nhà phát triển, đặc biệt là:

- Dựa trên các **mô hình ngôn ngữ lớn (LLMs)** như GPT-4.0, GPT-4-turbo, GPT-3.5
- Có thể sinh:
 - Văn bản tự nhiên
 - Dữ liệu có cấu trúc
 - Đoạn mã code
 - Trả lời hội thoại (Chat)

API chính được dùng: `chat.completions.create`

Đây là endpoint mô phỏng giống **ChatGPT**.

2. Cài đặt và kết nối API

Bước 1: Nhận API Key

- Truy cập <https://platform.openai.com/account/api-keys>
- Tạo một **API Key**
- Nạp tối thiểu **\$5** (rất rẻ, ví dụ: dịch 1.200 đoạn văn bản chỉ mất **\$0.04**)

Bước 2: Cài đặt thư viện

Trong terminal hoặc Google Colab:

```
pip install openai
```

Bước 3: Kết nối API trong Python

```
from openai import OpenAI

client = OpenAI(
    api_key="sk-..." # Thay bằng API key của bạn
)
```

3. Tạo phản hồi văn bản với `chat.completions`

Cấu trúc cơ bản:

```
response = client.chat.completions.create(
    model="gpt-4", # hoặc "gpt-4-turbo", "gpt-3.5-turbo"
    messages=[
        {"role": "system", "content": "You are a helpful assistant."},
        {"role": "user", "content": "Write a haiku about recursion in programming."}
    ]
)

print(response.choices[0].message.content)
```

3 phần quan trọng:

- `model`: tên mô hình
- `messages`: danh sách hội thoại
 - `system`: điều chỉnh **tính cách và hành vi**
 - `user`: prompt nhập từ người dùng
 - `assistant`: phản hồi (có thể thêm nếu muốn giữ trạng thái hội thoại)
- `choices[0].message.content`: kết quả đầu ra

4. System Prompt là gì?

System Prompt là phần đầu tiên trong `messages[]`, giúp điều chỉnh **hành vi của AI**.

Ví dụ:

```
{"role": "system", "content": "You are an expert in software engineering and provide concise technical answers."}
```

Giúp AI:

- Trả lời ngắn gọn
- Mang tính kỹ thuật
- Phù hợp cho ngữ cảnh cụ thể

Lưu ý:

- **ChatGPT (trên web):** không thể chỉnh `system prompt`
- **OpenAI API:** bạn **có thể** chỉnh được — tạo được trợ lý tùy biến hơn nhiều

So sánh giữa GPT-4 và GPT-4-Turbo

Mô hình	Chi phí	Tốc độ	Dung lượng context
GPT-4	Đắt hơn	Ổn định	8k hoặc 32k
GPT-4-Turbo	Rẻ hơn 5x	Nhanh hơn	128k
GPT-3.5-Turbo	Rất rẻ	Nhanh	16k

Kết luận

Nội dung chính	Ghi nhớ
<code>chat.completions.create</code> là công cụ chính để tạo phản hồi văn bản	
<code>system prompt</code> giúp định hình tính cách của AI	
Cài đặt bằng <code>pip install openai</code> và dùng <code>client.chat.completions.create()</code>	
Có thể dùng các model khác nhau để cân bằng giữa chi phí và chất lượng	
ChatGPT không cho chỉnh system prompt, API thì có thể	

Tác giả: **Đỗ Ngọc Tú**
Công Ty Phần Mềm **VHTSoft**

Tùy chỉnh đầu ra bằng các tham số trong OpenAI API

Mục tiêu

Trong bài học này, chúng ta sẽ:

- Khám phá cách tinh chỉnh đầu ra của mô hình ngôn ngữ bằng cách sử dụng các **tham số**.
- Hiểu rõ vai trò của từng tham số: `temperature`, `top_p`, `frequency_penalty`, `presence_penalty`, v.v.
- Trải nghiệm thực tế bằng cách thử nghiệm các tham số ngay trong mã nguồn hoặc trên Playground của OpenAI.

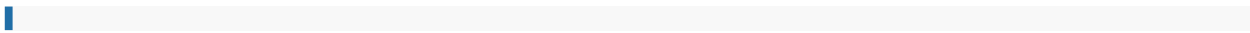
Tổng quan về Tham số trong Chat Completion

OpenAI API cung cấp một số tham số quan trọng để điều chỉnh cách mô hình tạo ra văn bản. Chúng ta sẽ tập trung vào những tham số **thường dùng nhất**:

Tham số	Mô tả ngắn gọn	Giá trị thường dùng
<code>temperature</code>	Điều khiển độ ngẫu nhiên của kết quả. Cao hơn → sáng tạo hơn.	<code>0</code> (chắc chắn), <code>1</code> (trung lập), <code>2</code> (siêu sáng tạo)
<code>top_p</code>	Sử dụng nucleus sampling , mô hình chỉ xét các token có xác suất tích lũy nằm trong <code>p</code> .	0.1 → chỉ lấy top 10%
<code>frequency_penalty</code>	Phạt nếu từ/ý lặp lại nhiều lần .	Từ <code>-2</code> đến <code>2</code>
<code>presence_penalty</code>	Phạt nếu từ/ý đã từng xuất hiện trước đó . Khuyến khích ý tưởng mới.	Từ <code>-2</code> đến <code>2</code>

Ví dụ: Điều chỉnh `temperature`

```
response = openai.ChatCompletion.create(
    model="gpt-4",
    messages=[
        {"role": "user", "content": "Viết một đoạn thơ ngắn về bầu trời đêm"}
    ],
    temperature=0.2
)
print(response['choices'][0]['message']['content'])
```



Với `temperature = 0.2`, kết quả sẽ **logic, nhất quán** và có vẻ "an toàn".

Giờ hãy thử với `temperature = 1.5`

```
response = openai.ChatCompletion.create(
    model="gpt-4",
    messages=[
        {"role": "user", "content": "Viết một đoạn thơ ngắn về bầu trời đêm"}
    ],
    temperature=1.5
)
```

“ Kết quả sẽ **sáng tạo, bất ngờ**, nhưng có thể "quá bay bổng" hoặc không chính xác – gọi là ảo giác(**hallucination**).

Kết hợp các tham số để điều chỉnh hành vi

Hãy xem ví dụ với `presence_penalty` và `frequency_penalty`

```
response = openai.ChatCompletion.create(
    model="gpt-4",
    messages=[
        {"role": "system", "content": "Bạn là một nhà văn sáng tạo."},
        {"role": "user", "content": "Hãy kể một câu chuyện về chú mèo khám phá vũ trụ."}
    ],
    temperature=1.0,
    frequency_penalty=1.2,
    presence_penalty=1.0
)
```

“ Điều này giúp GPT **tránh lặp lại** và **tìm ra nội dung mới mẻ**, tạo nên một câu chuyện hấp dẫn hơn.

Sử dụng Playground để dễ dàng điều chỉnh

Truy cập: <https://platform.openai.com/playground>

- Bạn có thể điều chỉnh tham số qua thanh trượt.
- Playground sẽ **tự động sinh ra mã code Python** để bạn sao chép về dùng.
- Cũng có tính năng **so sánh mô hình** (Compare), giúp bạn đối chiếu đầu ra giữa GPT-4-0125, GPT-4o, hay mô hình fine-tuned của bạn.

Lưu ý khi sử dụng

- **Không nên kết hợp** `top_p` và `temperature` **cùng lúc**, hãy chọn một trong hai.
- Cần **kiểm thử đầu ra nhiều lần**, vì đầu ra không cố định khi `temperature > 0`.
- Nếu GPT tạo ra kết quả kỳ lạ (hallucination), hãy **giảm temperature** và thử lại.

Việc điều chỉnh các tham số trong OpenAI API giúp bạn:

- Tạo ra nội dung **sáng tạo** hơn hoặc **ổn định** hơn tùy theo nhu cầu.
- Tránh việc GPT **lặp lại ý tưởng**.
- Tùy biến trải nghiệm người dùng cho chatbot, hệ thống gợi ý, hoặc sáng tạo nội dung.

Tác giả: Đỗ Ngọc Tú
Công Ty Phần Mềm VHTSoft

Sử Dụng OpenAI API Để Hiểu, Phân Tích Và Mô Tả Hình Ảnh

Trong phần này, chúng ta sẽ khám phá cách sử dụng **OpenAI API với hình ảnh**, một tính năng mạnh mẽ và linh hoạt, cho phép bạn:

- Hiểu nội dung hình ảnh
- Mô tả hoặc phân tích hình ảnh
- Trích xuất văn bản từ hình ảnh (OCR)
- Kết hợp giữa hình ảnh và văn bản để đưa ra câu trả lời chính xác

Tổng Quan: Tại Sao API này Lại Quan Trọng?

Thông thường, các mô hình ngôn ngữ chỉ làm việc với văn bản. Tuy nhiên, với khả năng **multimodal** (đa phương thức), OpenAI API hiện cho phép xử lý **hình ảnh** cùng với văn bản, mở ra rất nhiều ứng dụng thực tế như:

- Phân tích hóa đơn, biên lai, biểu đồ
- Trích xuất văn bản từ hình ảnh chụp màn hình
- Mô tả ảnh cho người khiếm thị
- Nhận diện nội dung ảnh trong hệ thống kiểm duyệt

Cách Gửi Hình Ảnh Cho API Phân Tích

Để gửi một yêu cầu đến API, bạn cần cấu trúc dữ liệu `messages` như sau:

```
{
  "model": "gpt-4-vision-preview",
  "messages": [
    {
      "role": "user",
      "content": [
        {
          "type": "text",
          "text": "Hãy mô tả những gì đang diễn ra trong hình ảnh sau."
        },
        {
          "type": "image_url",
          "image_url": {
            "url": "https://upload.wikimedia.org/wikipedia/commons/thumb/e/e9/Big_Ben_2012-06-23.jpg/800px-Big_Ben_2012-06-23.jpg"
          }
        }
      ]
    }
  ]
}
```

- `type: text`: Đây là đoạn văn bản mô tả yêu cầu (prompt).
- `type: image_url`: Đây là phần chứa hình ảnh (dưới dạng đường dẫn hoặc base64).
- `image_url.url`: Phải là **liên kết trực tiếp** tới ảnh (.jpg, .png,...), không phải trang HTML.

Các Tác Vụ Bạn Có Thể Làm Với Ảnh

1. Mô tả nội dung ảnh

```
"text": "Hãy mô tả bức ảnh này một cách chi tiết."
```



Kết quả sẽ là mô tả như: “Bức ảnh chụp Tháp đồng hồ Big Ben tại London vào một ngày nắng, với bầu trời trong xanh...”

2. Trích xuất nội dung văn bản trong ảnh (OCR)

"text": "Hãy đọc và trích xuất toàn bộ văn bản có trong hình ảnh sau."

“ Kết quả: Toàn bộ chữ có trong ảnh sẽ được AI trả lại, rất hữu ích khi xử lý biên lai, tài liệu scan, v.v.

Thay vì URL: Dùng Base64 Cho Ảnh Riêng Tư

Nếu ảnh bạn muốn phân tích **không được lưu công khai trên internet**, bạn có thể **chuyển ảnh sang dạng base64** và nhúng trực tiếp vào yêu cầu:

```
{
  "type": "image_url",
  "image_url": {
    "url": "data:image/jpeg;base64,/9j/4AAQSkZJRgABAQEASABIAAD..."
  }
}
```

“ Lợi ích của base64:

- Không cần tải ảnh lên server trung gian
- Bảo mật cao hơn (ảnh không công khai)
- Hoạt động trong các API nội bộ, offline hoặc giới hạn truy cập

Các Lưu Ý Khi Dùng API Với Ảnh

Yêu cầu	Mô tả
Ảnh rõ nét	Tránh ảnh mờ, tối hoặc có nhiễu
Định dạng ảnh	Hỗ trợ: <code>.jpg</code> , <code>.png</code> , <code>.gif</code>

Yêu cầu	Mô tả
Dung lượng ảnh	Không quá lớn (thường < 20MB)
Kết hợp text + image	Cung cấp prompt rõ ràng cùng với ảnh để tăng độ chính xác
Sử dụng hệ thống prompt	(Optional) Có thể dùng để hướng dẫn vai trò hoặc mục tiêu của AI

Ví Dụ: Phân Tích Biểu Đồ

```
"text": "Đây là biểu đồ. Hãy mô tả xu hướng chính và kết luận rút ra từ nó."
```

Với hình ảnh kèm theo là biểu đồ đường, biểu đồ cột, AI sẽ trả lời như một nhà phân tích dữ liệu.

Điểm chính	Nội dung
Đa năng	OpenAI API không chỉ làm việc với văn bản, mà còn hiểu được hình ảnh
Mô tả & Đọc ảnh	Có thể mô tả cảnh vật hoặc trích xuất chữ từ ảnh
Hỗ trợ base64	Hữu ích cho ảnh nhạy cảm, không công khai
Ứng dụng thực tế	Từ kiểm tra tài liệu, hỗ trợ người khiếm thị đến phân tích hình ảnh trong hệ thống doanh nghiệp

Dưới đây là **một ví dụ thực tế hoàn chỉnh** về cách **trích xuất thông tin từ hóa đơn** bằng cách sử dụng **OpenAI API với hình ảnh**, kết hợp cùng đoạn mã Python để thực thi yêu cầu này.

Tình huống giả định:

Bạn có một **ảnh chụp hóa đơn** và muốn tự động trích xuất các thông tin sau:

- Tên công ty
- Ngày xuất hóa đơn
- Tổng tiền
- Mã số thuế
- Số hóa đơn

Yêu cầu gửi đến OpenAI

Ví dụ về `messages` gửi tới GPT-4 API:

```
"messages": [  
  {  
    "role": "user",  
    "content": [  
      {
```

```
"type": "text",
"text": "Hãy trích xuất các thông tin sau từ hóa đơn trong hình ảnh: Tên công ty, Mã số thuế, Ngày lập hóa đơn, Số hóa đơn, và Tổng tiền thanh toán. Vui lòng trả về kết quả dưới dạng JSON."
},
{
  "type": "image_url",
  "image_url": {
    "url": "https://example.com/hoa_don_mau.jpg"
  }
}
]
}
```

Kết quả GPT trả về có thể là:

```
{
  "company_name": "Công Ty TNHH ABC Việt Nam",
  "tax_code": "0301234567",
  "invoice_date": "01/04/2024",
  "invoice_number": "0000123",
  "total_amount": "5.500.000 VND"
}
```

Đoạn mã Python thực tế

Bạn có thể dùng đoạn mã sau để gửi ảnh hóa đơn đến GPT-4 và nhận kết quả JSON:

```
import openai

openai.api_key = "YOUR_API_KEY"

response = openai.ChatCompletion.create(
  model="gpt-4-vision-preview",
  messages=[
    {
      "role": "user",
      "content": [
        {
          "type": "text",
```

```
"text": "Hãy trích xuất các thông tin sau từ hóa đơn trong hình ảnh: Tên công ty, Mã số thuế, Ngày  
lập hóa đơn, Số hóa đơn, và Tổng tiền thanh toán. Vui lòng trả về kết quả dưới dạng JSON."
```

```
    },  
    {  
      "type": "image_url",  
      "image_url": {  
        "url": "https://example.com/hoa_don_mau.jpg" # Thay bằng URL ảnh thật  
      }  
    }  
  ]  
}  
],  
max_tokens=500  
)  
  
print(response['choices'][0]['message']['content'])
```

Nếu bạn dùng ảnh nội bộ (Base64):

Bạn có thể chuyển ảnh hóa đơn thành Base64 và gửi đi:

```
import base64  
  
with open("hoa_don_mau.jpg", "rb") as img:  
    b64 = base64.b64encode(img.read()).decode("utf-8")  
  
image_url = f"data:image/jpeg;base64,{b64}"
```

Sau đó thay dòng:

```
"url": "https://example.com/hoa_don_mau.jpg"
```

bằng:

```
"url": image_url
```

Ứng dụng thực tế

Bạn có thể áp dụng kỹ thuật này để:

- Tự động lưu hóa đơn vào hệ thống kế toán

- Gửi cảnh báo nếu tổng tiền vượt ngưỡng
- Kiểm tra hóa đơn trùng
- Phân loại hóa đơn theo nhà cung cấp

Tác giả: Đỗ Ngọc Tú
Công Ty Phần Mềm VHTSoft