

# Kết nối các công cụ (tools) bằng bind\_tools

`bind_tools` là một tính năng trong LangChain (thường dùng với các mô hình như OpenAI's GPT) để **kết nối các công cụ (tools)** với LLM. Nó cho phép mô hình gọi những tool đó khi cần, như tính toán, truy xuất thông tin, tra cứu, v.v.

`ind_tools` giúp bạn nói với LLM rằng:

“ “Ê, ngoài việc trả lời bằng văn bản, mày có thể **gọi** mấy tool này nếu thấy cần.”

## Cú pháp cơ bản

```
llm_with_tools = chat_model.bind_tools(  
    tools=[tool_1, tool_2],  
    parallel_tool_calls=False # hoặc True nếu muốn gọi song song nhiều tool  
)
```

- `tools`: danh sách các tool bạn định sử dụng (được định nghĩa trước).
- `parallel_tool_calls`: nếu `True`, mô hình có thể gọi nhiều tool một lúc (song song). Nếu `False`, gọi từng cái một theo thứ tự.

## Ví dụ đơn giản

### Bước 1: Định nghĩa các tool

```
from langchain.tools import tool  
  
@tool  
def add(a: int, b: int) -> int:  
    """Cộng hai số"""  
    return a + b  
  
@tool  
def multiply(a: int, b: int) -> int:
```

```
"""Nhân hai số"""
```

```
return a * b
```

## Bước 2: Bind tools vào mô hình

```
from langchain.chat_models import ChatOpenAI

chat_model = ChatOpenAI(model="gpt-4")

llm_with_tools = chat_model.bind_tools(
    tools=[add, multiply],
    parallel_tool_calls=False # toán học thường nên làm tuần tự
)
```

## Bước 3: Gọi mô hình và để nó tự chọn tool

```
response = llm_with_tools.invoke("Hãy cộng 3 và 5 giúp tôi")
print(response.content)
```

Nếu LLM hiểu rằng cần thực hiện phép cộng, nó sẽ tự động gọi tool `add(3, 5)` và trả lại kết quả là `8`.

`bind_tools` hữu ích vì

- **Modular hóa:** LLM không cần biết cách hoạt động nội bộ của tool, chỉ cần mô tả chức năng.
- **Mở rộng dễ dàng:** Bạn có thể thêm bao nhiêu tool tùy thích.
- **Tự động lựa chọn:** LLM tự chọn tool phù hợp dựa trên yêu cầu người dùng.

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

Phiên bản #1

Được tạo 22 tháng 4 2025 02:07:32 bởi Đỗ Ngọc Tú

Được cập nhật 22 tháng 4 2025 02:13:50 bởi Đỗ Ngọc Tú