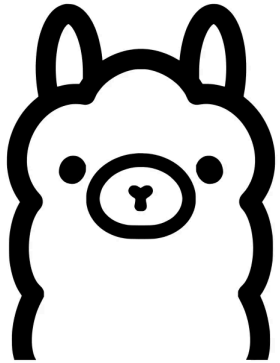

Ollama

本地快速上手大型语言模型

(Llama 3.1, Mistral, Gemma 2, ...)

📚 LangChain - 工具调用

操作步骤

- 实现 Ollama 模型调用工具
- 运行确认

操作演示

main.py

```python
import time, pprint, warnings
from langchain_ollama import ChatOllama
from langchain_core.tools import tool
from langchain import hub
from langchain.agents import AgentExecutor, create_tool_calling_agent

warnings.simplefilter("ignore")

def evalEndTime(start_time):
    end_time = time.time()  # 获取结束时间
    execution_time = "(程序运行时间：%.2f 秒)" % (
        end_time - start_time
    )  # 计算程序运行时间
    return execution_time

@tool
def get_weather(location: str):
    """Call to get the current weather."""
    if location.lower() in ["tyo", "tokyo", "东京", "東京"]:
        return location + "→" + "下雨"
    else:
        return location + "→" + "晴天"

print("=" * 100)
start_time = time.time()  # 获取开始时间

# 定义工具
tools = [
    get_weather,
]

# 创建Ollama
llm = ChatOllama(
    model="llama3.1:8b",  # 支持工具调用
    # model="gemma2:9b", # 不支持工具调用
    temperature=0.5,
)
print(">", llm)

# ReAct提示词
prompt = hub.pull("hwchase17/openai-tools-agent")

# 创建Agent
agent = create_tool_calling_agent(
    llm=llm,
    tools=tools,
    prompt=prompt,
)

# 创建Agent执行器
agent_executor = AgentExecutor.from_agent_and_tools(
    agent=agent,
    tools=tools,
    verbose=True,
    handle_parsing_errors=True,
)

################################################################################
# 调用Agent
response = agent_executor.invoke({"input": "Can you tell me the weather in Tokyo?"})
# response = agent_executor.invoke({"input": "Can you tell me the weather in Boston?"})
# response = agent_executor.invoke({"input": "请告诉我东京的天气？"})
pprint.pprint(response)

print()
# 打印结束时间
```

下课时间

课件下载： https://github.com/komavideo/lesson_ollama