

计算思维 专题研究

第 11 组

小组成员：赵乐朋

代码链接

https://github.com/one-step-beh1nd/computational_thinking

视频演示链接

<https://cloud.tsinghua.edu.cn/d/73d5d9ea09b74bda9e01/>

选题方向：棋类AI的计算思维

从深蓝到 AlphaGo，棋类 AI 的发展不仅是人工智能技术进步的缩影，也是计算思维在现实问题中的生动体现。计算思维强调用算法、逻辑和分解问题的方法来解决复杂任务。在棋类游戏中，这种思维方式表现为如何将庞大的棋局空间抽象、分解，并通过策略搜索、模式识别和评估函数进行决策。从规则枚举到启发式搜索，再到深度学习与强化学习结合的 AlphaGo，每一次进步都让我们更直观地理解：计算思维不仅是程序员的工具，也是一种普适的思考方法，能帮助人们在面对复杂问题时更系统、更高效地分析和决策。

技术细节

1. RAG

在 wiki 上搜索相关关键词的页面，将内容复制下来。

并且让 ChatGPT 生成有关内容。

使用 `pyserini` 库构建 RAG 库。

2. Tool Calling

使用 `BaseTool` 抽象基类定义工具接口，每个工具需要实现 `to_function()` 方法返回 OpenAI 函数格式的定义。

通过 `ToolRegistry` 管理所有工具，提供注册和执行功能。

工具执行结果封装为 `ToolResult`，可以渲染为 LLM 可读的格式。

3. ReAct Agent

使用 `ReactAgent` 类实现 ReAct 风格的 agent，通过循环（最多 `max_turns` 轮）与 LLM 交互。

每次调用 LLM 时传入工具定义，LLM 可以选择调用工具或直接回答。

如果 LLM 返回 `tool_calls`，则执行工具并将结果加入消息历史，继续循环。

如果 LLM 返回最终答案，则结束循环。