

数据可视化 实践课15



1

智能体框架

LangGraph

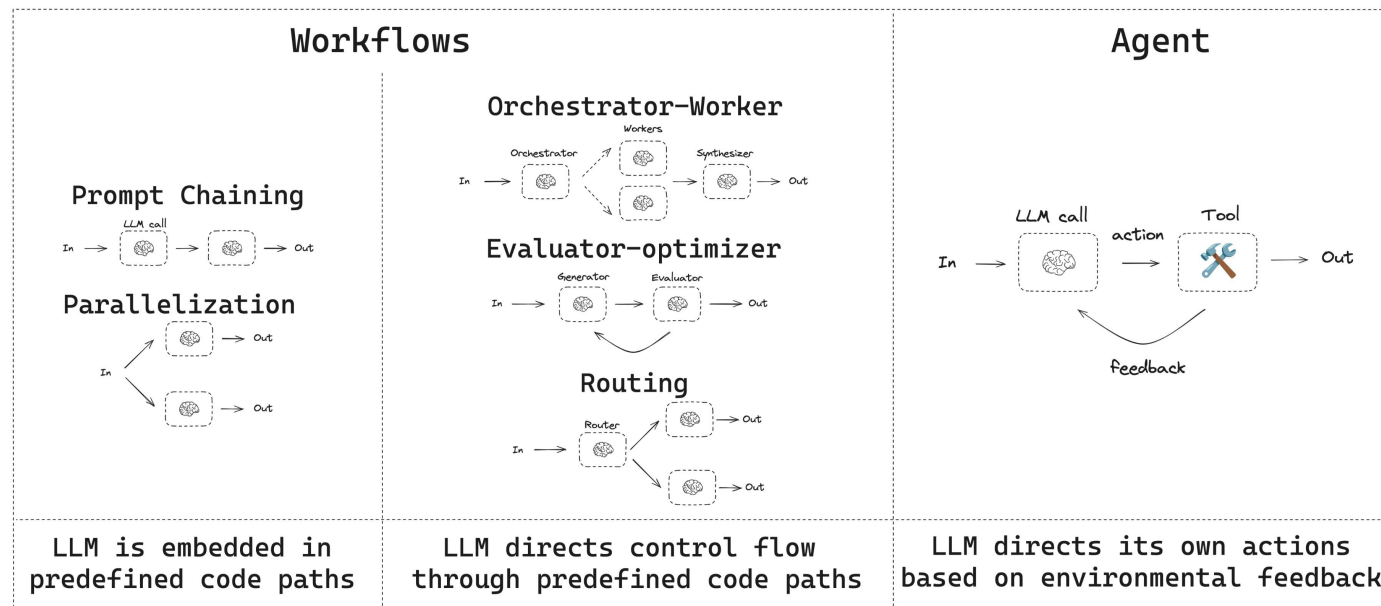
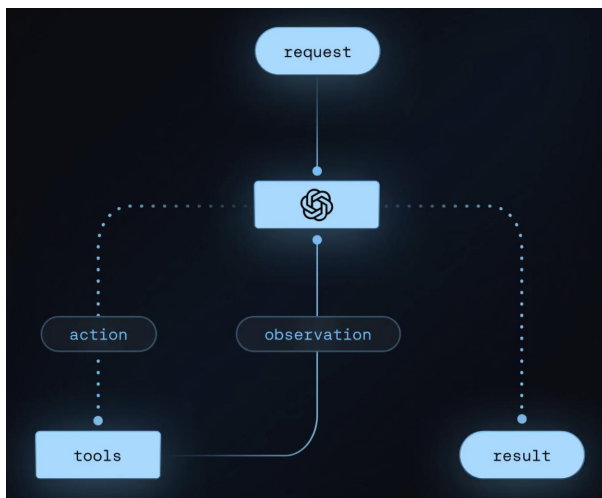


langgraph



langchain

- LangGraph 是面向 LLM 应用的流程编排框架，用图结构组织多个 Agent 节点。
- 节点表示一次具体操作，例如图表生成、图表评审、状态更新或代码修复。
- 边表示执行顺序；条件边可根据分数、轮次或评审结果决定继续还是结束。
- 本实践使用 LangGraph 将生成 Agent、评审 Agent 和循环迭代流程连接起来。





2

LangGraph Agent 实战



图表生成 Agent

练习1:

- 让 LLM 生成可运行的 ECharts 或 D3 代码。要求：
 1. 输入：参考图表图片 + CSV 数据文件，输出：复现图表的 echarts 或 d3 代码 + 新图表；
 2. 在 chart_agent_prompts.py 文件中填充 DEFAULT_GENERATION_PROMPT；
 3. 约束输出 JSON 格式和代码接口；
 4. 新图表应尽量复现原图样式，并且所有数据都来自给定 CSV。

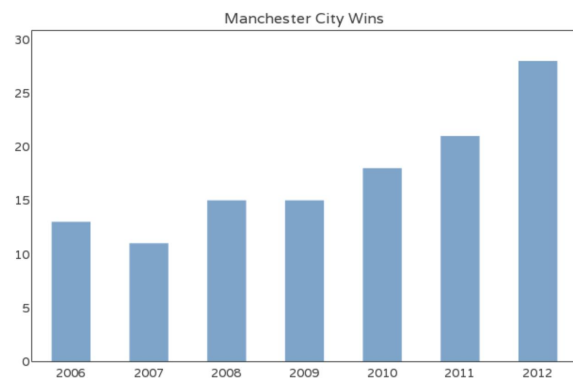
输入

参考图表图片

reference.png
2.9KB

CSV 文件

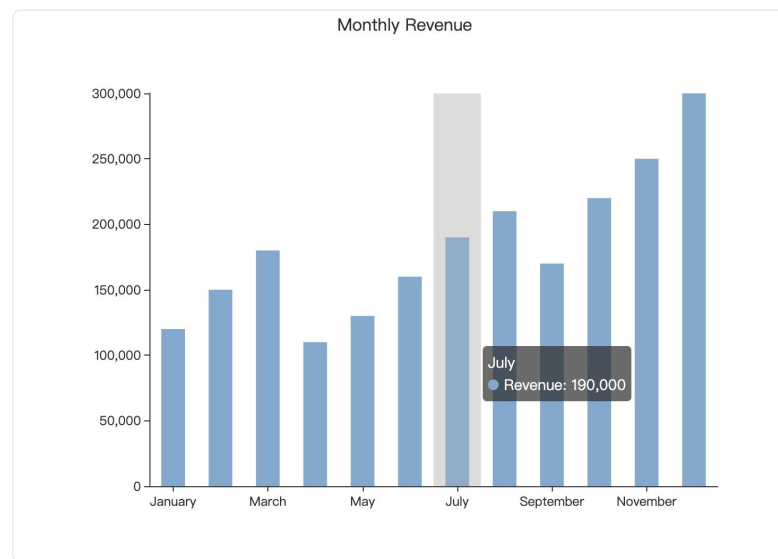
data.csv
404.0B



参考图表

输出

生成完成



图表评审 Agent

练习2:

- 在练习1基础上增加评审 Agent，对生成结果进行一轮评价。要求：
 - 在 `chart_agent_prompts.py` 文件中填充 `DEFAULT_REVIEW_PROMPT`；
 - 评审内容：评价生成图表是否复现参考图，并检查代码是否适配 CSV 数据，是否写死数据、代码鲁棒性；
 - 重点：设计评审维度、评分规则和修改建议格式。

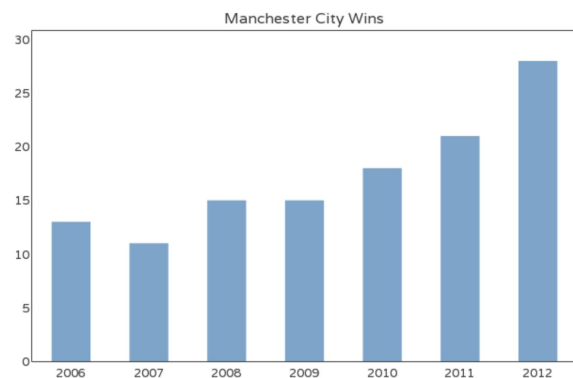
输入

参考图表图片

reference.png
2.9KB

CSV 文件

data.csv
404.0B



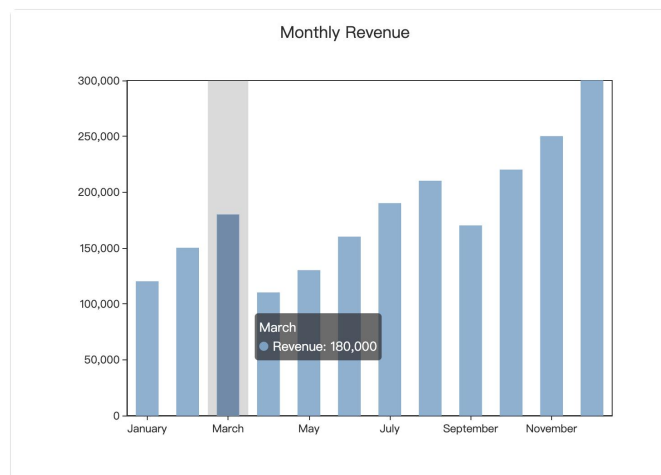
参考图表

输出

运行完成

评审总分

88



双 Agent 循环迭代

练习3:

- 在练习2基础上增加循环机制，对生成结果进行评价生成迭代。要求：
 - 在 `chart_agent_prompts.py` 文件中填充 `DEFAULT_ITERATIVE_GENERATION_PROMPT` 和 `DEFAULT_ITERATIVE_REVIEW_PROMPT`；
 - 修改 `exercise3_iterative_agents.py` 文件，用 LangGraph 条件边实现最多 3 轮的自动改进流程。

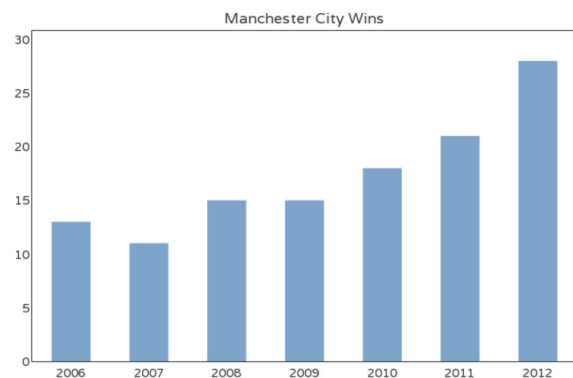
输入

参考图表图片

reference.png
2.9KB

CSV 文件

data.csv
404.0B



参考图表

当前状态

- 正在解析 CSV 和参考图表图片...
- 第 1 轮生成 Agent 正在生成图表代码...
- 第 1 轮生成 Agent 已完成。
- 第 1 轮评审 Agent 正在评审...
- 第 1 轮评审 Agent 已完成，得分：88。
- 已达到目标分数，准备结束。
- 双 Agent 循环迭代完成。

输出

迭代完成

最终分数	迭代轮次	目标分数
88	1	85

