Elile AI Task: Deep research Agent

AI Engineer Technical Assessment: Deep Research AI Agent

Overview

Design and implement an autonomous research agent capable of conducting comprehensive investigations on individuals or entities to uncover hidden connections, potential risks, and strategic insights. This challenge simulates real-world intelligence gathering scenarios critical to risk assessment and due diligence operations.

Technical Requirements

Core Architecture

- **Multi-Model Integration**: Implement at least two distinct AI models with different capabilities (Gemini 2.5, Claude Opus 4, OpenAI 4.1,..)
- **Consecutive Search Strategy**: Design an intelligent search progression that builds upon previous findings
- **Dynamic Query Refinement**: Agent must adapt search strategies based on discovered information

Functional Specifications

- **Deep Fact Extraction**: Identify and verify biographical details, professional history, financial connections, and behavioral patterns
- **Risk Pattern Recognition**: Flag potential red flags, inconsistencies, or concerning associations
- **Connection Mapping**: Trace relationships between entities, organizations, and events
- **Source Validation**: Implement confidence scoring and cross-referencing mechanisms

Before starting make sure to develop an evaluation set, have a name with deeply hidden facts about the person that one can find with so many searches and use those as evaluation of your AI agent. Audit each steps, read about prompt design and make sure your prompt design is up for the task.

Implementation Guidelines

Technical Stack

- Use LangGraph for agent orchestration

- Leverage available AI APIs, search engines, and real online data
- Implement proper error handling and rate limiting
- Design for scalability and maintainability

Deliverables

Phase 1: Development

- Complete codebase with comprehensive documentation
- Three test persona profiles with expected findings
- Execution logs demonstrating agent performance
- Risk assessment reports for each test case with details

Phase 2: Live Demonstration

- Real-time execution on provided test case
- Code walkthrough and architectural explanation
- Discussion of design decisions and trade-offs
- Q&A on scalability and production considerations

Evaluation Criteria

Technical Excellence

- Code quality, architecture, and best practices
- Effective multi-model orchestration
- Intelligent search progression logic
- Error handling and edge case management

Research Capability

- Depth and accuracy of information gathering
- Quality of risk assessment insights
- Ability to uncover non-obvious connections
- Source verification and confidence scoring

Innovation & Efficiency

- Creative approaches to complex research challenges
- Optimization of search strategies
- Novel use of available tools and APIs
- Scalability considerations

Once you are done, we do an assessment of the solution and then we invite you to present it and do a live demo. Be ready for that. We will go through your code and ask you about different parts

of it. You have to use AI tools effectively and we will test for your knowledge and proficiency of AI use as well.

Timeline: 4 days for development

References:

https://ali-khaledi1989.medium.com/technical-assessment-in-the-age-of-ai-finding-the-builders-not-test-takers-08bfebcc4368

https://docs.anthropic.com/en/docs/build-with-claude/prompt-engineering/overview