Jade Sanchez

2376 ITAI

5/6/2025

AI Agent Creation

Problem Statement

Students and professionals spend extensive time doing research work since they need to look for sources manually along with assessing credibility and extracting important data before generating citations. Students who have minimal research experience face high challenges and produce many mistakes during this lengthy manual process. An AI Research Assistant Agent under development by our project takes responsibility for extracting important information from multiple sources while summarizing results and determining source credibility before creating structured reports with accurate citations. This system enables users to accomplish high-quality research in both faster and more precise ways.

Project Option

Option 1: Research Assistant Agent

Agent Design

According to the proposed design the agent consists of these structural components:

Input Processing: The user interface supports text-based question entry (e.g., "Find academic sources on renewable energy policies in Europe").

Memory System: The system stores sources together with their summaries and user feedback in a vector database instance such as Pinecone or FAISS.

Reasoning Component: This system follows a Planning-then-Execution pattern to ensure it selects the right tools between search, summarize and citation functions. A multi-step task structure is achieved through Chain-of-Thought reasoning in the framework.

Output Generation: The system generates reports comprised of source summaries along with source interrelations that utilize correct citation formats (APA/MLA/Chicago).

Agent Pattern

Our system will deploy the Planning-then-Execution pattern together with Chain-of-Thought reasoning techniques for source analysis and summarization tasks.

Tool Selection

Our agent integrates at least two additional tools into its workflow.

The Web Search API serves as our primary tool either as SerpAPI or Bing Search API.

The tool serves to find current and reliable sources.

Agent activates searches according to specified query terms.

The system needs to operate correctly when API responses deliver empty data or when API requests encounter system setbacks.

The system relies on two comprehensive tools as its foundation: OpenAI GPT-3.5 Turbo or Azure OpenAI.

Summarizes content from retrieved sources.

Both summary output parsing and error data summaries fall under the component's responsibility.

Between Zotero or Citation.js the Citation Manager API functions as an optional third accessory.

Development Plan

PHASE	DATES	TASKS
Planning Phase	Mar 24-Apr 7	Architect system design while choosing tools to build before writing the proposal.
Implementation Phase	Apr 8-Apr 28	The team constructs a prototype and implements integrated tools together with building a memory system.
Finalization Phase	Apr 29-May 5	The team will test features after refinement while documenting code formats as they create a demonstration video.

Evaluation Strategy

Our evaluation of the AI agent focuses on the following criteria.

Testing users will evaluate the relevance of retrieved sources during evaluation.

Accuracy of summaries and connections

The citation system follows proper formatting according to style guide standards.

After user interaction with the agent, we will measure their satisfaction using feedback scoring systems.

Reinforcement Learning Element

The system will adapt its search procedures and summary guideline recommendations through ratings users provide using statements such as "this summary proved useful" or "this source proved irrelevant." A basic reward model operates through such mechanism together with a policy improvement framework.

Resource Requirements

Our project development happens through Azure AI Studio which we access using the included $100 student credit. Estimated resource needs:

Azure OpenAI API (GPT-3.5 Turbo)

Bing Search API

Cloud resources will consume between 50% to 75% of the available Azure funding when using caching and prompt optimization techniques.

Hardware: The system will use Azure as its cloud-computing platform while developers will perform local tests using their laptops.

Risk Assessment

Risk	Mitigation
API limits or budget exhaustion	The project will use GPT-3.5 as its base model while both caching data outputs and tracking resource utilization.
Tool integration issues	The approach begins with a basic tool configuration in combination with strong error prevention systems.
Scope creep	First develop the essential system functionalities (search capabilities, summarization and citation generation) before implementing additional features.

Data privacy concerns	Avoid storing sensitive user queries; use input validation