Mentorship Report

Intern Info :

1) Name - Varad Dongarkar
2) Mentor name - Yash Chavan
3) Duration - March 25 to May 25
4) Github repository - https://github.com/VDongarkar/Mentorship_Project

Challenges :

1) The chatbot initially failed to maintain conversation continuity, resulting in disjointed or irrelevant responses.

2) Faced difficulty in guiding the AI agent's behavior, especially in task-specific scenarios.

3) The Gemini API could not generate relevant images or videos, which limited the agent's recommendation capabilities.

4) Encountered challenges while integrating multiple tools into a single AI agent and enabling dynamic tool usage based on the use case.

5) While developing the Stack Overflow agent, the system ignored the API/tool and relied on Gemini responses instead.

6) The reflection logic for the agent was initially over-engineered, causing the model to produce overly deterministic and rigid outputs.

Proposed Solution:

I developed an **AI-powered Stack Overflow Search Assistant** aimed at assisting developers in efficiently resolving programming errors by intelligently retrieving relevant solutions from Stack Overflow data.

To enhance the chatbot's coherence, I introduced a **history context mechanism** that allowed it to retain past interactions and maintain continuity across multi-turn conversations. I also explored **prompt engineering techniques** through deeplearning.ai resources to better control the behavior of the language model in task-specific scenarios.

For visual content generation, I integrated the **Google Serper API**, enabling the agent to provide real-time and relevant multimedia results. I also studied **LangGraph documentation** to understand the proper binding of multiple tools and dynamic task routing within a single agent framework.

During testing, I debugged the **Stack Overflow agent** by analyzing logs and correcting a misconfigured system prompt. To improve performance, I adopted the **Groq API** for faster, more cost-effective execution. Additionally, I referred to AI agent examples and tutorials to implement a simplified **reflection mechanism**, allowing more flexible and accurate responses.

Resources used :

1) https://langchain-ai.github.io/langgraph/
2) https://learn.deeplearning.ai/
3) https://www.youtube.com/@LangChain