AI AGENTS

Autogen

Autogen is a framework that enables the creation of AI agents through language models. It emphasizes multi-agent collaboration where agents can handle complex workflows by sharing tasks and interacting seamlessly, offering a highly customization and scalable approach.

Pros

- Can easily work with non openai models like from groq, meta, ollama...
- There is some built in agent classes are there like user proxy agent, assistant agent.
- Simple to create agents
- Can register our custom tools like wikipedia search tool from langchain after agent is created
- Can create multiple agents and can group them to behave like team to finish the given task

Doc : https://microsoft.github.io/autogen/docs/tutorial/chat-termination

Crew AI

Crew AI allows users to create autonomous agents designed to perform tasks, make decisions, and collaborate with other agents. The system supports customization roles, goals, and tools for agents, enabling dynamic task delegation and communication across various tasks

Pros

- Create separate tasks to each agents
- Crew AI is offering some built in tools we can use them for tasks and agents
- While Creating agents its offers variables like goal, role, backstory. Which is help to the agent like how that it should behave.
- Can create custom tools using tool decorator in crew ai tools library.
- Using Hierarchal process techniques we can manage the all agents by providing the manageable llm.
- Using Sequential process technique we can run all the agents one by one.
- Also supports for async execution

Cons

- Getting sometimes library version comparability issues
- Getting litellm library errors even while giving correct api key, base_url and model names

Doc : https://docs.crewai.com/core-concepts/Agents/

AI AGENTS

LangGraph

LangGraph focuses on building AI agents with advanced task management capabilities. It provides a structured environment where agents work with tasks that are defined by input and output expectations, offering robust delegation and task execution features across agents.

Pros

- It is like totally graph data structure.
- There is a start node and end node for our particular task.
- Between these node we need add our custom nodes and edges. Like chat-bots or tools from langchain or even own custom functions
- Langgraph support for conditional edges. This will help to determine which tool should use for the prearticular task.
- Everything is customization like we can basic nodes like start and end node and we create our own nodes and them as edges between start and end nodes
- For multi-agents we can create functions those function will act as agents and we can add that function to langgraph state. So while processing the task agents will communicate each other to perform the task

Cons

- Hard to implement prompt engineering. To create any one agent need to specify system prompt. This should be dynamic from the user for create custom ai agent.
- There is no vars like role, role, backstory like in crew ai. So for this user need to provide everything in one var about the agents. But can be manageable.
- Hard to implement the agents in this langgraph. Everything is customization but need to write code from scratch for build complete agent.

Doc : https://langchain-ai.github.io/langgraph/tutorials/multi_agent/multi-agent-collaboration/

Conclusion

Need to Implement some agents with all these frameworks to understand how the frameworks will work for building custom ai agents.

Clarifications

1. Need more clarification about ai agents deployments
2. To create own custom ai agents can I use any framework mentioned above (autogen, crew ai)