Introduction to Broad AI

- Broad AI is a framework used to develop applications powered by generative AI.

- The framework utilizes a multi-agent system approach to enable AI systems to exhibit complex skills and collaboration amongst expert agents.

- Key parts of the Broad AI framework include the multi-agent system and agentic framework.

- Advantages of using the Broad AI framework include simplified development and seamless integration of agents.

- The structured approach to AI development empowers developers to create advanced AI systems efficiently.

- The application will consist of a user interface with a chat box and text box for typing messages, as well as an Express server hosting the web page and an API endpoint to interact with generative AI capabilities.

Broad AI Integration Process

- Once access to the app is obtained, developer tools must be accessed to ensure that the response from the API is received upon clicking the ask button.

- The basic integration for broad AI is set up, following which the documentation for broad AI's multi-agent system is to be reviewed.

- The initial step is to install broad AI using the terminal, followed by importing and instantiating it in the application.

- After importing the required agents, such as the researcher, the register method needs to be called to complete the registration process.

- Specific agent configuration parameters, such as Google search engine API key, programmable search engine ID, and Yelp API key, are required for the researcher agent, as outlined in the documentation.

- At this stage, the broad AI methods can be utilized as per the documentation, and a summary of the achieved progress can be made.

Setting Up Broad AI Agents and Configuration Parameters

- Imported the broad AI and broad AI agents, specifically the researcher agent, and created an object for Broad.

- Supplied the researcher agent so that Broad is aware of that agent and registered the agent by supplying the pra configuration and agent-specific configuration, which are API keys for interaction with external entities.

- Provided a bunch of Broad AI configuration parameters, including the llm API configuration (API provider endpoint) and data template (body expected by the API endpoint) containing placeholders for system context, task, and format.

- Need to specify the dot notation location in the response structure when the llm responds, and enable history for building a stateful application in a chat-like application.
- Building out the ask API endpoint for Broad to work and completing the connection from the front end to the API endpoint by modifying the script in the HTML page to pass the question provided in the text box to the ask API endpoint.
- The AI object provides various methods that are required in a specific order: plan function, critique function, and execute function.

Plan Method and Multi-Agent System

- The plan method not only prepares a plan to answer a question but also refines the plan by considering the history of the conversation.
- The plan method provides a result of the plan structure, including status and the actual plan containing sequence steps, objectives, and identified agents for each step.
- The multi-agent system utilizes the agent ecosystem to generate a proper plan for a proper response, calling multiple agents with various skills and parameters.
- Generative AI intelligently performs planning by providing specific agents and remains within the boundaries of those agents.
- Calling the plan method involves passing conversations as an array and receiving a promise, which is checked for the value of "right."
- The server needs to be running to create and update the plan, adding steps to engage the user and personalize the interaction.

Executing Broad AI Steps

- The results of the execution are used to generate a formal response back to the user.
- The next step is to engage in execute and in execute all, which requires passing a plan in the execute method as an array of steps.
- The plan is within the field plan, which is an array of steps, so only the plan needs to be passed, not the entire object.
- The results of the execution show that a plan with one step was generated, and it was executed successfully, including an update of the plan.
- The sequence zero is executed, which includes the general assistant agent and the get current time method, resulting in a response from the execution of that particular agent step.
- The final step involves asking AI to respond, and the response needs to include the original question and any conversations that need to be part of the response.

- The final response generated by Broad is "I am doing well, thank you. How can I assist you today?"

- Broad's property called conversations maintains a history of the conversation, which can be used to continue the conversation and make it more of a chat experience.

- The next step is to frame up the response to go back into the API on the front end for further use.

Building Reliability for Conversational AI

- To complete the API endpoint, ensure the server is running and test the full response back.

- The response from the API contains an AI response with an array of HTML tags and text. The response is "I'm doing well. How can I assist you today?"

- The metadata indicates that the status was complete, which is essential for building reliability in the system.

- After recording the conversation as an array, the next step is to store the conversation using session storage. The conversation will be stored as a JSON array, allowing it to be retrieved for the next API call.

- Additionally, the responses need to be printed back on the form to maintain contextual conversation.

- Testing with chat messages revealed that the response maintained context and provided relevant information, showing the ability to understand and continue the conversation.

- The system is able to provide suggestions for Chinese and Indian restaurants around the New York airport, demonstrating contextual understanding and the ability to maintain the conversation.

Building a Chat Application using Broad AI

- The application uses a sequence of steps to search for and extract ratings for Chinese and Indian restaurants.

- The highest-rated Chinese restaurant is Jing pong with a rating of 4.6, and the highest-rated Indian restaurant is Sy the with a rating of 4.6.

- It will then research and find the birth year of the father of AI, Johan mcari, who would be 105 years old today.

- The chat application is built using the agentic workflow system, specifically the multi-agent system and the framework called Broad AI.

- Broad AI allows the incorporation of various agents from the agent store, such as the researcher agent, and encourages the development of custom agents for specific use cases.

- The entire application is supported by Broad AI with minimal code required for writing the backend.