Question

Is there some tool/platform within Microsoft agentic stack that could be used as “marketplace” or “catalogue” for client employees to use the available agents in a business-user-friendly way?

Answer

Copilot for M365 has a "Get Agents" feature where available agents are published by the enterprise administrator. This feature allows users to discover and use agents in a business-user-friendly way.

https://support.microsoft.com/en-us/topic/get-started-with-agents-for-microsoft-365-copilot-169469d7-328d-4d37-9090-bfc2058a39bd

https://support.microsoft.com/en-us/topic/introducing-copilot-agents-943e563d-602d-40fa-bdd1-dbc83f582466

Question

how we can test this workflow since it includes many steps, how to test oevrall output and what if there is inaccuracy in output of any of the step in between?

Answer

The workflow represented as a directed graph in the example I provided was for demonstration purposes. When doing this is a real-world scenario, you would want to use a process frame work that allows you to define the workflow in a more structured way. Also you would capture the telemetry of the processing for each sessions to understand the completeness and accuracy of the process and output. You want to use best practice for software development and include the equivalent of unit test for your AI Agents and/workflow steps There are some common workflow patterns

- Fan In: The input for the next Step is supported by multiple outputs from previous Steps.
- Fan Out: The output of previous Steps is directed into multiple Steps further down the Process.
- Cycle: Steps continue to loop until completion based on input and output.
- Map Reduce: Outputs from a Step are consolidated into a smaller amount and directed to the next Step's input.

Examples of process framework

https://learn.microsoft.com/en-us/semantic-kernel/frameworks/process/process-framework

https://www.langchain.com/langgraph

For the evaluation and analytics you would want to consider an evaluation framework
https://learn.microsoft.com/en-us/azure/ai-foundry/concepts/evaluation-approach-gen-ai

Testing Agentic Systems please review AutoGenBench
https://microsoft.github.io/autogen/0.2/blog/2024/01/25/AutoGenBench/

Question

Is there anyway for these agents to discern accurate vs fictitious information, while it is sourcing information. (Think bad actors that flood channels, intentionally, to disseminate false information for the purpose of fooling AI)

Answer

We skipped over the Data Governance and AI Governance part of the discussion the slides are included in the deck. This is primarily a data governance and AI governance issue & Responsible AI Issue. The developer of the agent is responsible for adding the curated data source to the agent and the data sources that the AI applications use should be vetted and approved by the enterprise data governance team before being used. Considerations like Citations are important and alerting the user when output might be grounded on Web data so that the human reviewer can validate the grounding data is correct and or from a reliable source. So ensuring that malformed or incorrect data is not made available to the agent is important.

There are guardrails available on Azure such as:

Azure Content Safety

https://learn.microsoft.com/en-us/azure/ai-services/content-safety/overview

For more information, Please review the following links:

Prompt Shields

https://learn.microsoft.com/en-us/azure/ai-services/content-safety/concepts/jailbreak-detection

Protected Material Detection

https://learn.microsoft.com/en-us/azure/ai-services/content-safety/concepts/protected-material?tabs=text

Custom Categories

https://learn.microsoft.com/en-us/azure/ai-services/content-safety/concepts/custom-categories?tabs=standard

---

Question

Is there a simple overview of the agents a company has created and how they are improved? Is this done independently, or does it require script adjustments? Additionally, is there a central tool for managing agents, including roles and responsibilities? And is there a size constraint for creating agents?

Answer

When the agents are created in Copilot Studio the Governance and monitoring are baked into that system. The management of Agents would also be part of the Copilot Studio ecosystem, which would help manage access etc. We didnt get into the extending Copilot part of the presentation there are several options between plugins and agents that might be considered depending on the use case and data security requirements. In terms of scripting there are git integrations that can be leveraged if there is a development group using the pro-code tools to develop these agents.

There is no real size constraint however when using the Copilot studio tool considerations around the document size and number of documents being processed have recommended limits. If for example you are

using the LLM to process a 1000 page documents being process, with the current state of the art for copilot you might consider a custom copilot. It all depends on the use case. A large percentage of use cases can be solved with the Out of the Box Copilot Studio before considering a custom pro-code solution

---

Question *can we expand in isolation of the code interpreter? how does it manage prompt injection or malicious code?*

Answer Prompt injection should be handled by the safety systems before getting to a code interpreter. If there is a concern about malicious actors accessing a code interpreter enabled agent that agent should not have coder interpreter enabled. Limiting the agent to specific API calls and not allowing the agent to run arbitrary code is a good way to limit the risk of prompt injection.

Question *do this tool provide Jupyter editor as well?*

Answer If the question was does AI Foundry provide Jupyter editor the answer is no. However the Agent SDK is available here: https://learn.microsoft.com/en-us/azure/ai-services/agents/quickstart?pivots=programming-language-python-azure