LangChain & LangSmith Skills: Teach Your AI to Build Agents

4 MIN · YOUTUBE · HTTPS://WWW.YOUTUBE.COM/WATCH?V=657AGKGGA44
https://www.youtube.com/watch?v=657Agkgga44

SUMMARY

The video demonstrates how to utilize LangChain and LangSmith skills with Cloud Code to build, observe, and iterate on a deep research agent. It showcases the agent's ability to delegate tasks, run research queries, and create datasets while leveraging skills for efficient operation and evaluation.

- *Cloud Code generates a deep research agent capable of delegating tasks to workers.*
- *Skills provide Cloud with best practices, dependencies, and orchestration knowledge.*
- *The agent runs research queries on topics like Chinese restaurants and AI trends in 2026.*
- *Cloud Code uses skills to create a dataset and manage API keys seamlessly.*
- *The agent employs an evaluator skill to assess results using a match percentage tool.*
- *Traces of the agent's operations are captured, providing insights into its performance.*
- *The evaluation shows a strong match between expected and actual outputs, demonstrating the effectiveness of the agent.*

In this video, we'll be showing how to use LangChain and LangSmith skills with Cloud Code. We'll see how Cloud Code can use skills to effectively build, observe, and iterate on an agent. To start, we just asked Cloud Code to generate us a deep research agent that can delegate tasks to workers. You can see Cloud call her skills for guidance. Without them, Cloud wouldn't know best practices. With skills, it knows dependencies, structures, and orchestration out of the gate. And we can watch it put together our agent in one shot.

Once Cloud has finished putting together our agent, we can ask it to run some research for us. But to make sure that the research is concise, we'll also ask Cloud to ensure we're tracing to LangSmith. And with skills, Cloud knows what to check without having to search the docs or explore code. We can see that what it does is it reflects on the deep agent that it's put together to simplify the coordinator and sub-agent configurations. This allows research to be more concise without sacrificing functionality.

Let's go ahead and run some research to see the results. We're going to be using two test queries. The first will be about Chinese restaurants in New York and San Francisco. And the second will be about AI trends in 2026. Let's give Cloud a second to finish running the research. It looks like both queries have successfully returned. And at this point, we want to prepare our agent for testing. So, let's ask Cloud Code to help us make a data set for our agent.

We can see that the first thing it's going to do is invoke the data set skill, as well as the trace skill, to learn how to

export traces and use them in data sets. Cloud Code will need a second to find our API keys, but from that point on, it will be smooth sailing. Cloud will no longer need to fumble around for the right API format to successfully leverage LangSmith. Skills are what enable Cloud to efficiently and consistently find the right path in working with LangChain and LangSmith.

In our pre-builts, we cover LangChain, LangGraph, Deep Agents, and LangSmith to ensure that Claude has a deep understanding of all areas in our ecosystem. And in the tasks so far, it's used a variety of those skills to complete our requests. As we watch Claude, we can see that it has used our inbuilt CLI to find traces and understand the full trajectory of our agent from to-dos to delegation. With skills, it's then able to build a script to effectively upload that data set to LangSmith, as well as capture the trajectory that we need.

Without skills, Claude would need to reason itself over the right format to capture the trajectory successfully. Let's give Claude a second to finish uploading its data set to LangSmith. Now that our data set is done, the final step in the agent life cycle is to run our agent against the data set. To do so, we'll need an evaluator for judging our results. And specifically, we're interested in tool called match percentage. Let's watch Claude put together the evaluator.

The first thing it's going to do is invoke the evaluator skill. And at this point, it's going to read the skill to understand best practices for both making an evaluator, as well as uploading the evaluator into LangSmith to be auto-used on our data set. We see that Claude has finished reading the skill and has used it to put together an effective evaluator. It's then going to decide to test this evaluator before it uploads it to LangSmith.

Now that the evaluator is uploaded, Claude is going to run its tests, and then it's going to run the actual evaluation. So, let's go ahead and switch over to LangSmith to see what Claude created. Let's first take a look at the traces Claude generated. We can see that we've captured complex traces from our deep agent runs. And we've concisely summarized some results about AI trends in 2026 using our deep agent.

Now, we can open up and see a similar trace of our research query around Chinese restaurants in New York and San Francisco. And we can also see the data set that Claude uploaded, as well as the evaluation that it's run. It looks like the evaluator that it's made has shown us that we have a great match between our expected trajectory and our actual output.