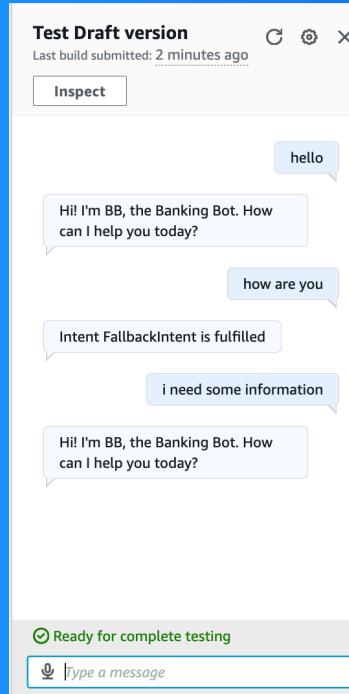




Build a Chatbot with Amazon Lex



sailavanyapudi08@gmail.com



Introducing Today's Project!

What is Amazon Lex?

Amazon Lex is a service for creating conversational interfaces using natural language understanding, allowing developers to build chatbots and voice assistants that can automate interactions and improve customer engagement.

How I used Amazon Lex in this project

Today, I used Amazon Lex to develop a chatbot that automates customer support responses, streamlining common inquiries for improved efficiency.

One thing I didn't expect in this project was...

One thing I didn't expect in this project was the amount of fine-tuning needed to improve the chatbot's accuracy in understanding varied user inputs

This project took me...

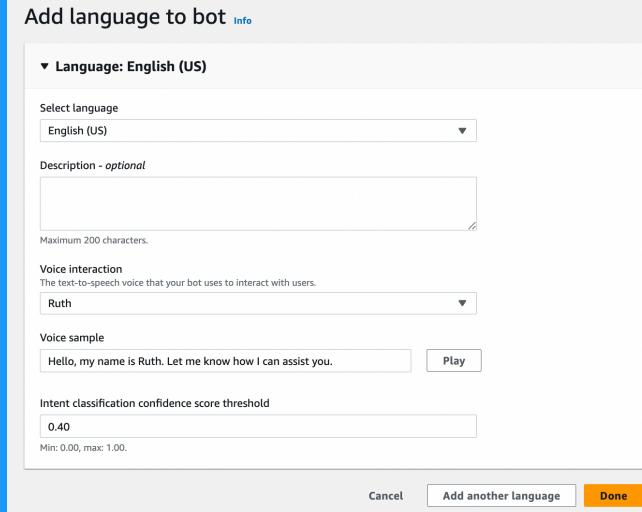
It took me around half an hour to get detailed understanding of AmazonLex.

Setting up a Lex chatbot

I created my chatbot from scratch with Amazon Lex. Setting it up took me around 5 minutes.

While creating my chatbot, I also created a role with basic permissions because Amazon Lex needs the permission to call other AWS services like lambda.

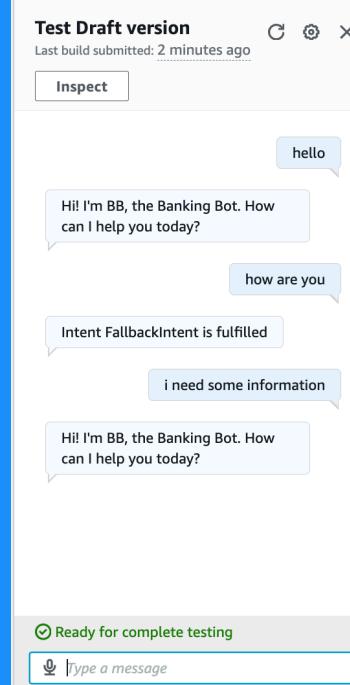
In terms of the intent classification confidence score, I kept the default value of 0.40. This means that your chatbot needs to be at least 40% confident that it understands what the user is asking to be able to give a response.



Intents

An intent is what the user is trying to achieve in their conversation with the chatbot. For example, checking a bank account balance; booking a flight; ordering food.

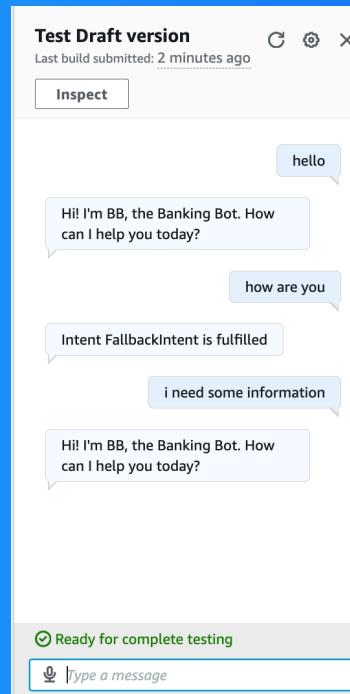
I created my first intent, WelcomeIntent, to welcome a user when they say hello.



FallbackIntent

I launched and tested my chatbot, which could respond successfully if I enter the above given sample utterances.

My chatbot returned the error message 'Intent FallbackIntent is fulfilled' when I entered any text other than the provided sample utterances.



Configuring FallbackIntent

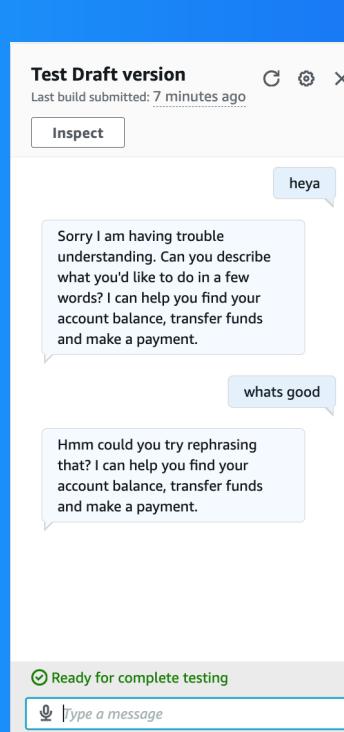
FallbackIntent is a default intent in every chatbot that gets triggered when the chatbot has a confidence score below the Intent classification confidence score threshold selected. In our case it was 40 %

I wanted to configure FallbackIntent because it is a custom error message that the chatbot will use to tell the user it doesn't understand their input.

Variations

To configure FallbackIntent, I created a FallbackIntent and set up responses for the users to understand.

I also added variations! What this means for an end user is a dynamic range of responses, making them sound more conversational!





NextWork.org

Everyone should be in a job they love.

Check out nextwork.org for
more projects

