

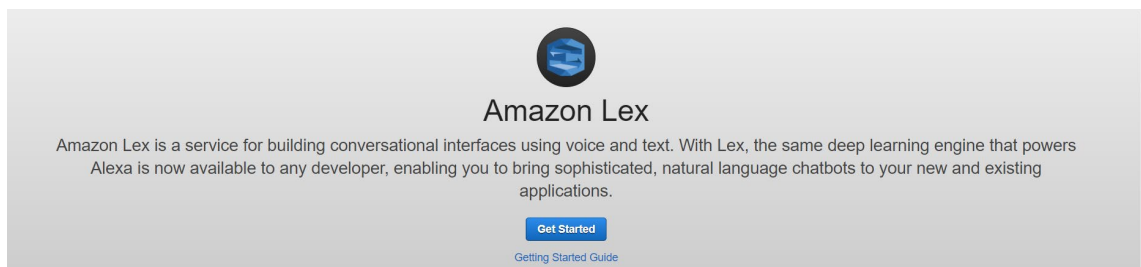
Amazon Lex ワークショップ

2021/06/04

シニアエバンジェリスト

亀田 治伸

1. Lex のマネージメントコンソールにアクセスします



2. [Get Started]を押します

Create your bot

Amazon Lex enables any developer to build conversational chatbots quickly and easily. With Amazon Lex, no deep learning expertise is necessary—you just specify the basic conversational flow directly from the console, and then Amazon Lex manages the dialogue and dynamically adjusts the response. To get started, you can choose one of the sample bots provided below or build a new custom bot from scratch.

CREATE YOUR OWN	TRY A SAMPLE		
Custom bot	BookTrip	OrderFlowers	ScheduleAppointment

Bot name

BookTrip


3. [Custom bot]を選びます


4. 以下のように値を入力します


Bot name

Language


Output voice




Session timeout 

Sentiment analysis ☒ Yes ☐ No 

5. はじめて Lex を起動する場合は、IAM ロールの作成画面が出ますので、画面の指示に従ってください。

IAM role [AWSServiceRoleForLexBots](#) 
Automatically created on your behalf

6. COPRA を No、[Confidence score threshold]に 0.4 を入力し、[Create]を押します

COPPA Please indicate if your use of this bot is subject to the [Children's Online Privacy Protection Act \(COPPA\)](#). [Learn more](#) 
☐ Yes ☒ No

Confidence score threshold

7. [Create Intent]を押します。Intent とは、人間の呼びかけに反応する処理群の塊のことを言います。一つの bot に複数の Intent を実装し、複雑な bot を作ることが可能ですが、その場合、最初の呼びかけでどの Intent を起動するか？の判断を Lex が行うため、呼びかけのキーワードを登録していくことになります。

8. さらに[Create intent]を押します

9. [BeefRiceBowl]と入力し[Add]をおします

10. [Sample utterances]に、皆さんが牛丼を注文する際に使う言葉を複数登録しましょう。最低 10 個以上登録することを商用環境では推奨しています。また登録されたキ

ワードをもとに学習を行い、さらに別の呼びかけにも反応するようになっていきます。思い浮かばない方は例えば「おなかすいた」「牛丼」「牛丼頼んで」等を登録してみましょう。

BeefRiceBowl Latest ▼

▼ Sample utterances ⓘ

e.g. I would like to book a flight.



11. 以下のような画面になれば正しくセットされています

BeefRiceBowl Latest ▼

▼ Sample utterances ⓘ

e.g. I would like to book a flight.



牛丼頼んで



牛丼



おなかすいた



これにより、合致する呼びかけがあった場合、bot のなかの BeefRiceBowl という intent が起動することになります

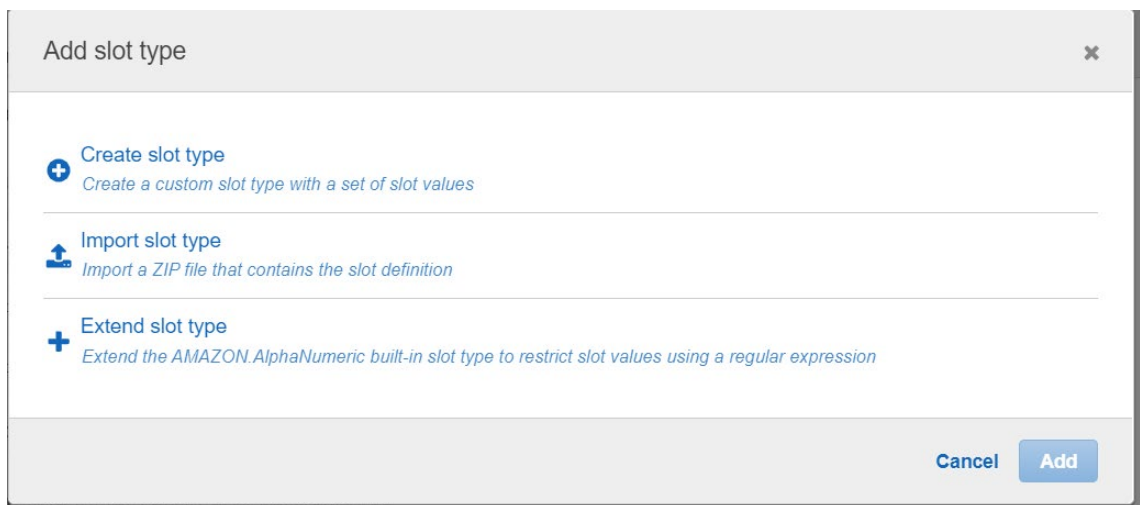
12. 次に、Slot を設定していきます。Slot とは、ユーザーが牛丼を頼む際に、Lex が取得しないといけない情報です。（並、大盛、卵の有無）などがそれに該当します。画面左ペインの[Slot types]のプラスボタンを押します

Slot types



No slots created

13. [Create slot type]をおします



14. 以下のように入力します

Slot type name

size

Description

size


Slot Resolution


☒ Expand Values ⓘ


☐ Restrict to Slot values and Synonyms ⓘ


15. [Value]には Lex がユーザーから取得したい情報を入力します

Value 














16. [Add slot to intent]をおします

17. 以下のように Slot が設定されました

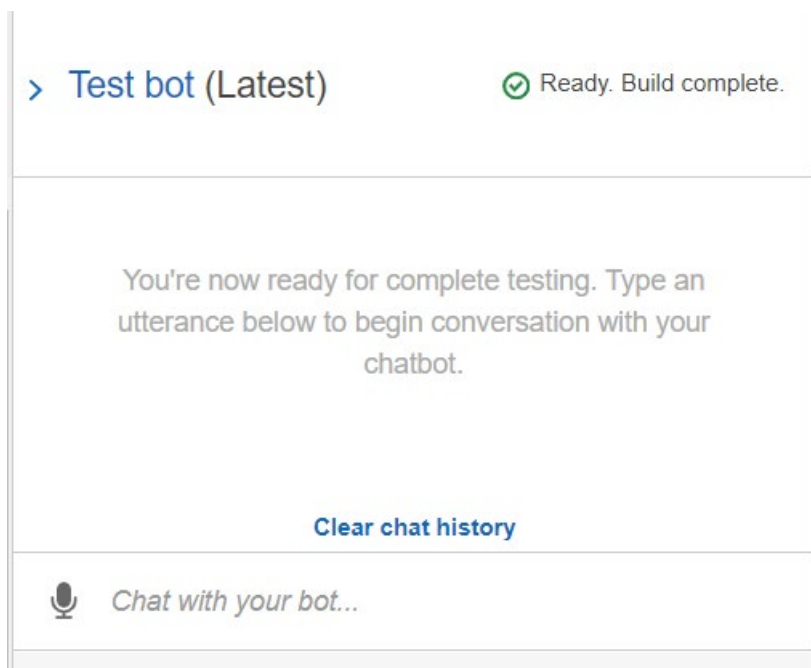
▼ Slots 							
Priority	Required	Name	Slot type	Version	Prompt	Settings	
		<input type="text" value="e.g. Location"/>	<input type="text" value="e.g. AMAZON.US_..."/>		<input type="text" value="e.g. What city?"/>		
1.		<input type="text" value="slotOne"/>	<input type="text" value="size"/>	Latest ▼	<input type="text" value="e.g. What city?"/>		

18. [slotOne]をわかりやすいように size に書き換えます。そして Prompt にユーザーへの Lex からの呼びかけを入力します。例えば「大きさはどうしますか？」などです

1.		<input type="text" value="size"/>	<input type="text" value="size"/>	Latest ▼	<input type="text" value="大きさはどうしますか？"/>		
----	---	-----------------------------------	-----------------------------------	----------	--	---	---

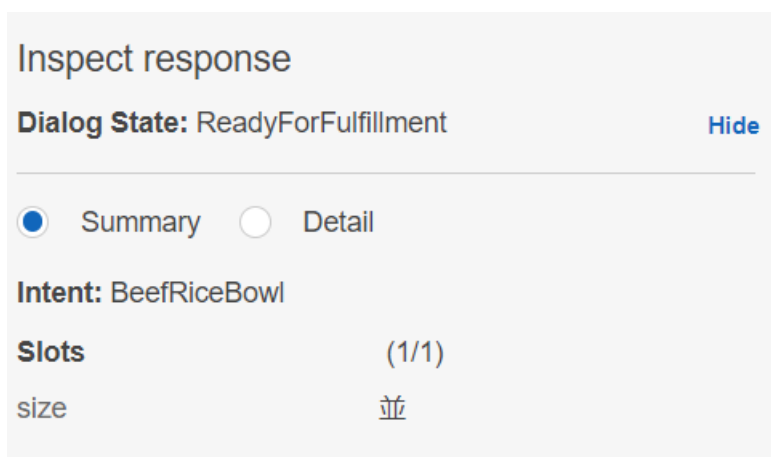
19. 一旦ここでテストのために画面右上の[Build]を押します

20. ブラウザの右側でテストが行えるようになります。テストをしてみます。



例えば、牛丼、とタイプし、そのあと、並、とタイプしてみてください

21. 以下のように、Lex が正しく値を取得できたことがわかります



Detail を見ると感情分析結果が出ていることがわかります。例えば、「うるさい」など
と答えると以下のように NEGATIVE と出力されます

```

{
  "responseCard": null,
  "sentimentResponse": {
    "sentimentLabel": "NEGATIVE",
    "sentimentScore": "{Positive: 0.0021450138,Negative: 0.9978549861}"
  },

```

22. では同じ要領でもう少し Slot をふやしてみましょう。例えば以下です。

▼ Slots ⓘ

Priority	Required	Name	Slot type	Version	Prompt	Settings
		<input type="text" value="e.g. Location"/>	<input type="text" value="e.g. AMAZ..."/>		<input type="text" value="e.g. What city?"/>	
1. ▼	<input checked="" type="checkbox"/>	<input type="text" value="size"/>	<input type="text" value="size"/>	1 ▼	<input type="text" value="大きさはどうしますか？"/>	
2. ^ ▼	<input checked="" type="checkbox"/>	<input type="text" value="vendor"/>	<input type="text" value="vendor"/>	Latest ▼	<input type="text" value="どの店に頼みますか？"/>	
3. ^	<input checked="" type="checkbox"/>	<input type="text" value="egg"/>	<input type="text" value="egg"/>	Latest ▼	<input type="text" value="卵はつけますか？"/>	

vendor:吉野家、松屋、すき屋

egg:お願いします、いらないです

23. 再度[Build]を押してテストします

24. 続いて、[Confirmation prompt]を設定します。これは、Lex が取得した値をもとに処理を続行させる前にユーザーに確認を行う機能です。

▼ Confirmation prompt ⓘ

☒ Confirmation prompt

Confirm



Cancel (if the user says "no")



25. [Confirm]に以下を入力して下さい


ありがとうございます。{vendor}にサイズ{size}を頼みますね。

[Cancel]には以下を入力してください


またよんでくださいね

26. 再度[Build]を押してテストします

27. 画面左ペインから[Error Handling]をクリックします

Intents 

BeefRiceBowl

Slot types 

egg
size
vendor

Error Handling

28. 以下のように設定し[Save]を押します

Error handling

☒ Clarification prompts

e.g. Sorry, can you please repeat that? 

もう一度お願いします 

Maximum number of retries

5

Hang-up phrase

e.g. Sorry, I could not understand. Please contact customer support 

ごめんなさい。一度終了します 

29. もう一度 Build してからテストを行うと、エラー処理が日本語になったことがわかります

30. 最後に画面右上の[Publish]を押し bot を公開します

Publish BeefRiceBowl

Your bot is published! You can now connect to your mobile app or continue to chatbot deployment.

Bot Name	BeefRiceBowl
Bot Version	1
Alias	beefricebowl

What to do next?
Here are some resources to help you progress once your bot is published.

[How to connect to your mobile app](#)
Learn how to connect to your bot to your mobile app.

[Integrate with Mobile hub.](#)
Please create a project and choose the Conversational Bots feature in Mobile Hub

[How to deploy your bot to other services](#)
Learn how to deploy your bot to other services like Facebook Messenger, Slack, Twilio, and Kik.

[Go to channels](#)

Close

https://docs.aws.amazon.com/ja_jp/lex/latest/dg/example1.html

こちらに Facebook や Twilio 等との連携方法が記載されています