layout: page title: LUIS

permalink: luis.html

Create your first LUIS App

Language Understanding Intelligent Service (LUIS) allows your application to understand what a person wants in their own words. LUIS uses machine learning to allow developers to build applications that can receive user input in natural language and extract meaning from it. A client application that converses with the user can pass user input to a LUIS app and receive relevant, detailed information back.

Create a LUIS.ai account

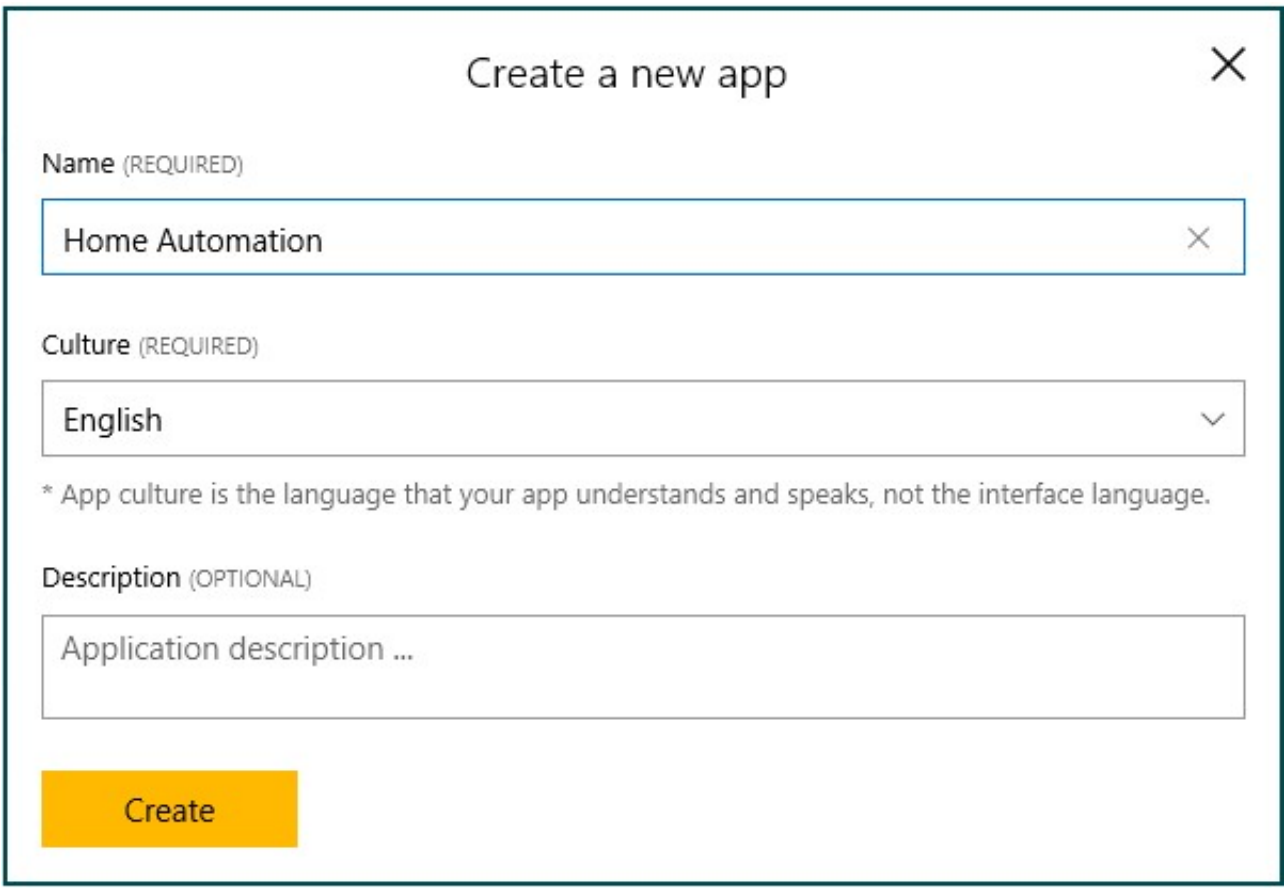
For this hands-on exercise, you will also need a **LUIS.ai** account.

1. In your browser, navigate to https://www.luis.ai/.
2. Click the **Sign in our create an account** button.
3. Authorize the **Luis.ai** application to access your account details.
4. Once your account is created, you will be taken to a simple signup form. Fill out the form with your **Country**, **Company** and **Referral** information.
5. Click the **Continue** button.
6. You will now arrive at the LUIS **My Apps** page.

Create a new LUIS app

You can create and manage your applications on My Apps page. You can always access this page by clicking My Apps on the top navigation bar of LUIS web page.

1. If you are not already there, navigate to the **My Apps** page on the **Luis.ai** website and login: https://www.luis.ai/applications.
2. On **My Apps** page, click **New App**.
3. In the dialog box, name your application "Home Automation".

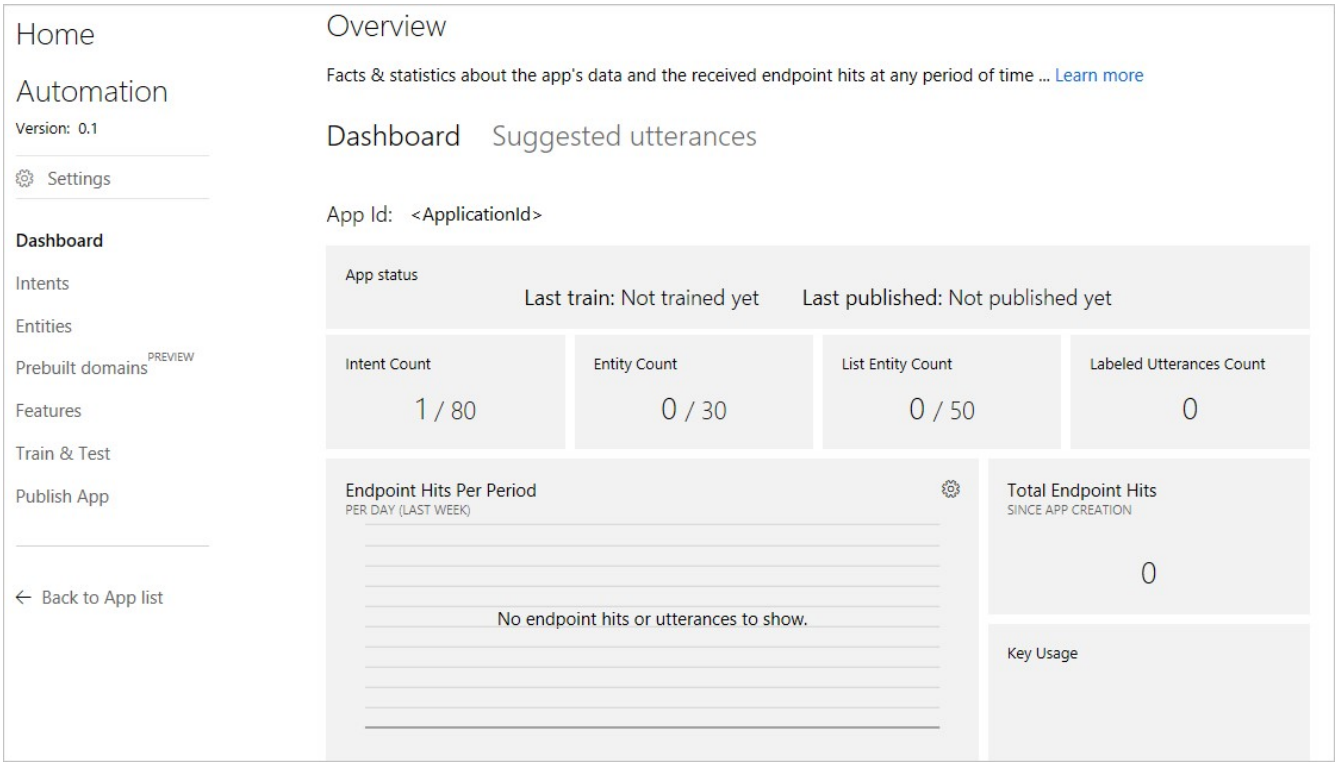


4. Choose your application culture (for this Home Automation app, we'll choose English), and then click **Create**.

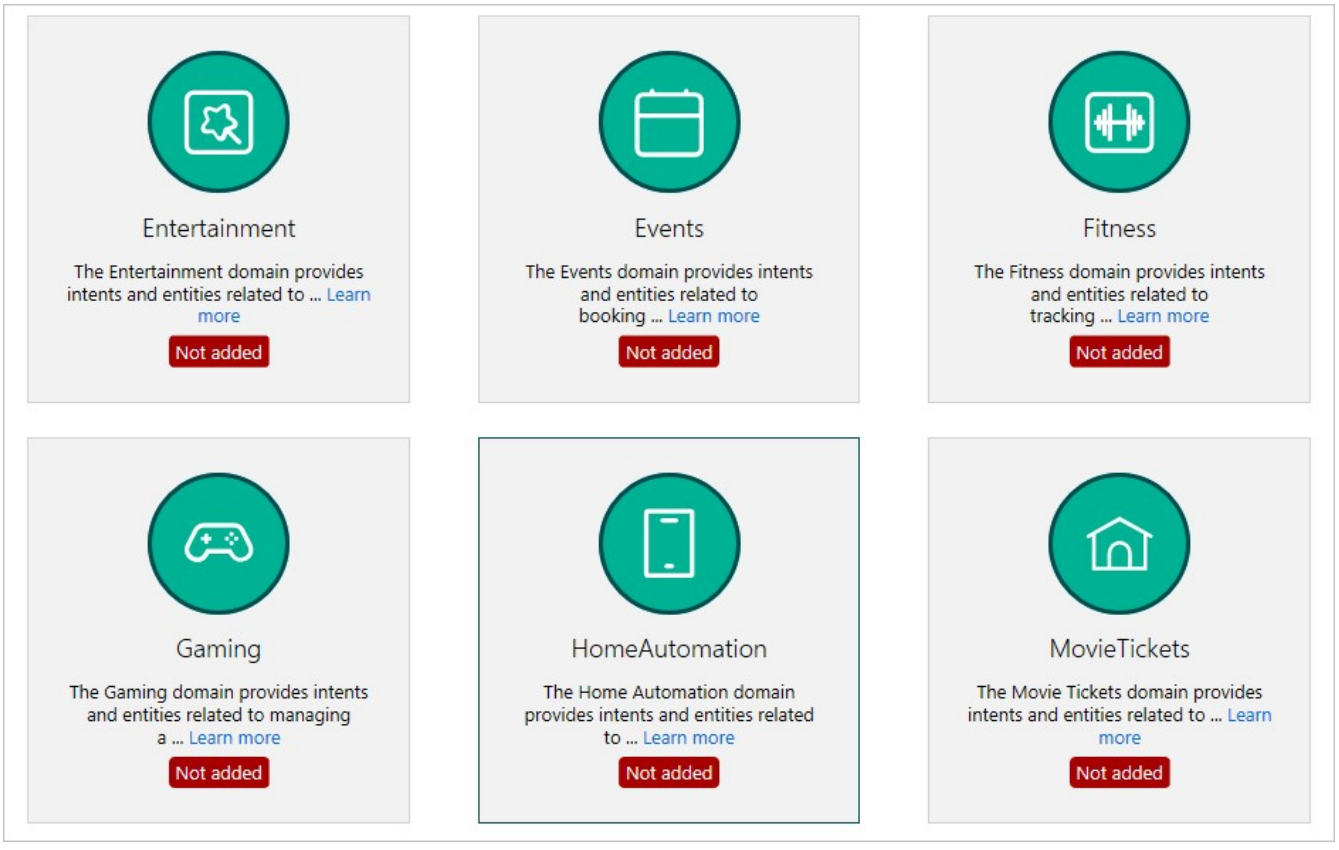
   > The culture cannot be changed once the application is created.

5. LUIS creates the Home Automation app and opens to the Dashboard. The application dashboard contains summary information about app usage.

6. You can explore your application using the links in the left panel.



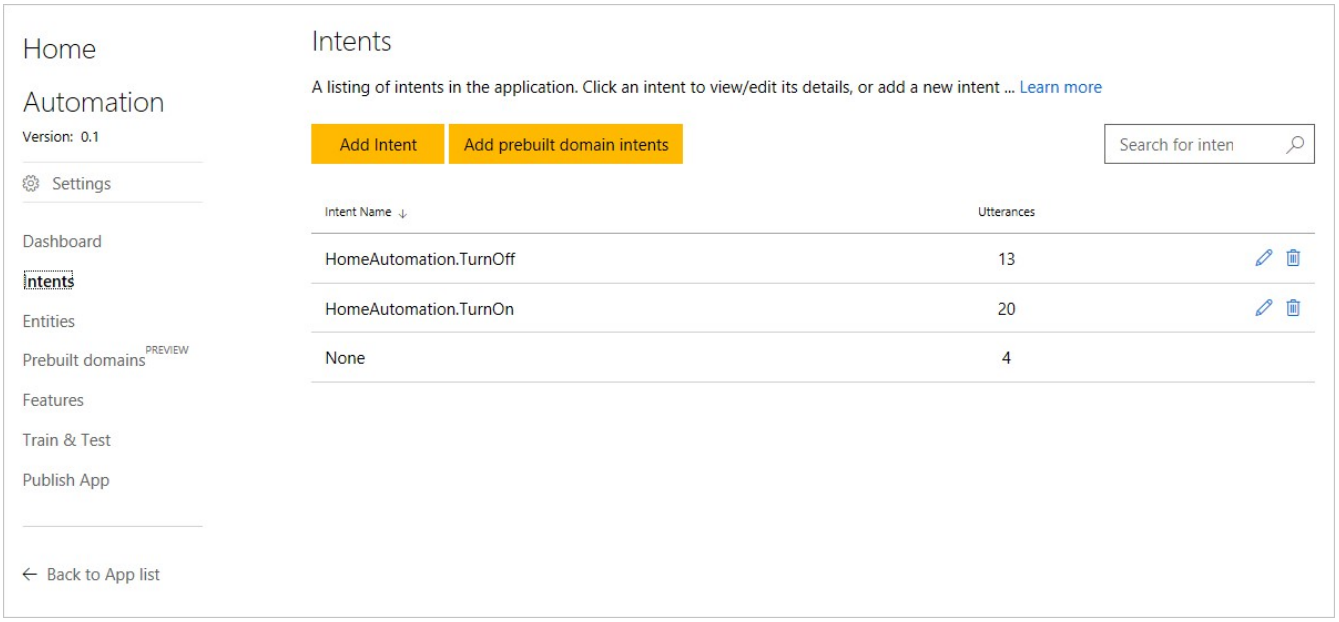
7. Click on **Prebuilt domains** in the left-side navigation pane. Then click on **HomeAutomation**.



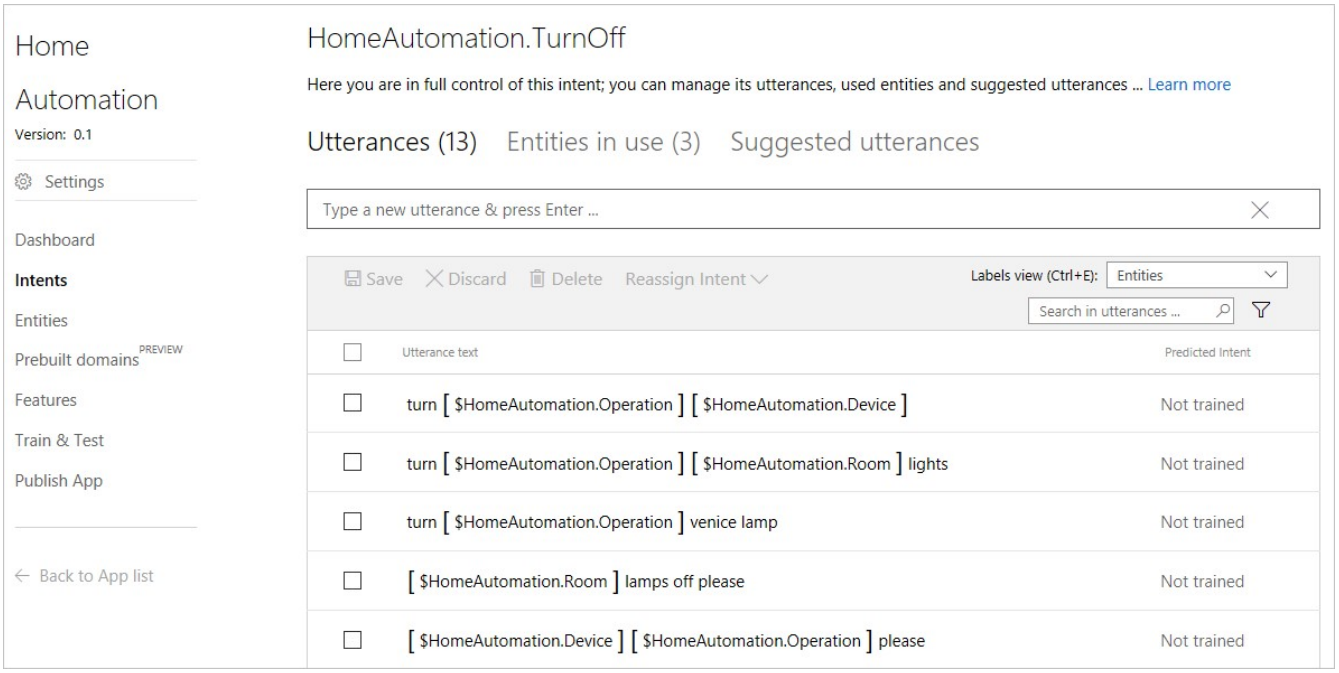
8. Click **Yes** when prompted to add the **"HomeAutomation"** domain to the app.

9. Click on **Intents** in the left-side navigation pane, and you can see that the HomeAutomation domain provides **HomeAutomation.TurnOff**, **HomeAutomation.TurnOn**, and **None** intents in your application. Each intent has sample utterances.

> **None** is an intent provided by all LUIS apps. You use it to handle utterances that don't correspond to functionality your app provides.



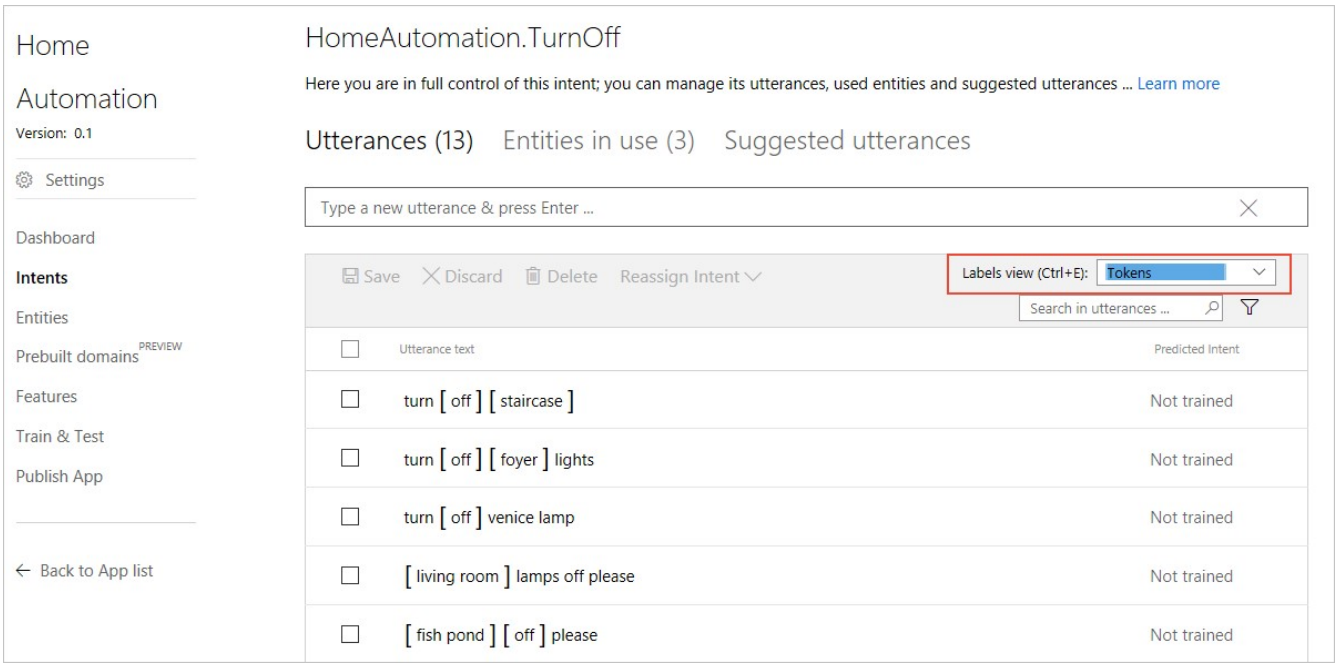
10. Click on the **HomeAutomation.TurnOff** intent. You can see that the intent contains a list of utterances which are labeled with entities.



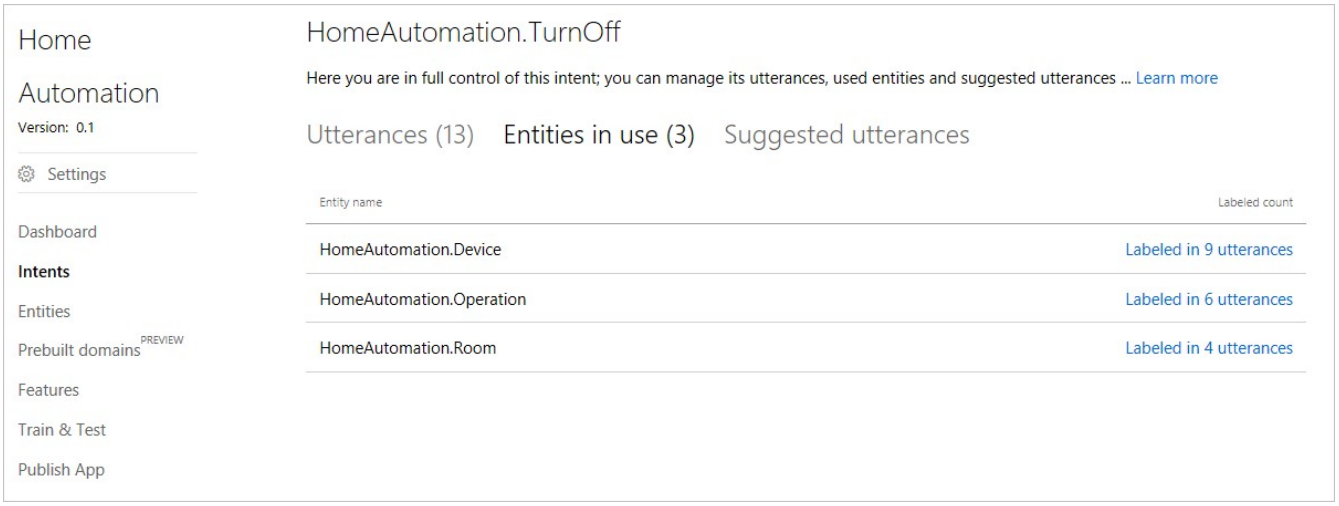
11. Click on the **Labels view** and select **tokens**. This shows the text tokens that make up each labeled entity, instead of the name of the entity type.

12. If you compare the same utterance in the tokens view and the entities view, you can see that some of the words of each utterance have already been labeled.

13. The first utterance is "turn off staircase." The word "off" has been labeled as the type of HomeAutomation.Operation. The word "staircase" has been labeled as the type of "HomeAutomation.Device."



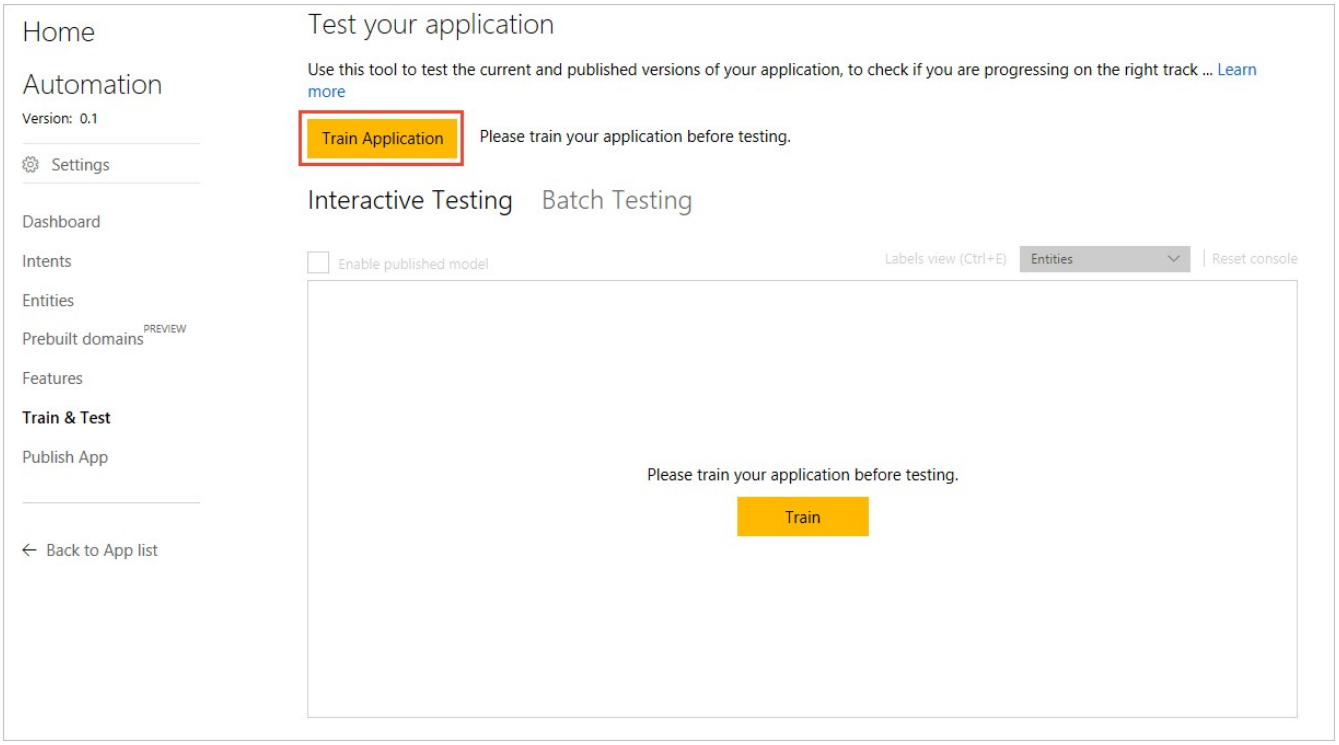
14. Click **Entities in use**. This shows the entities this app identifies in the utterances.



Train & Test your LUIS App

To validate our app, we should train it. Once we've trained the app, you can test it immediately. We can then publish our app and test the published endpoint in a browser using the generated URL.

1. Click on **Train & Test** in the left-side navigation, then click **Train application**.

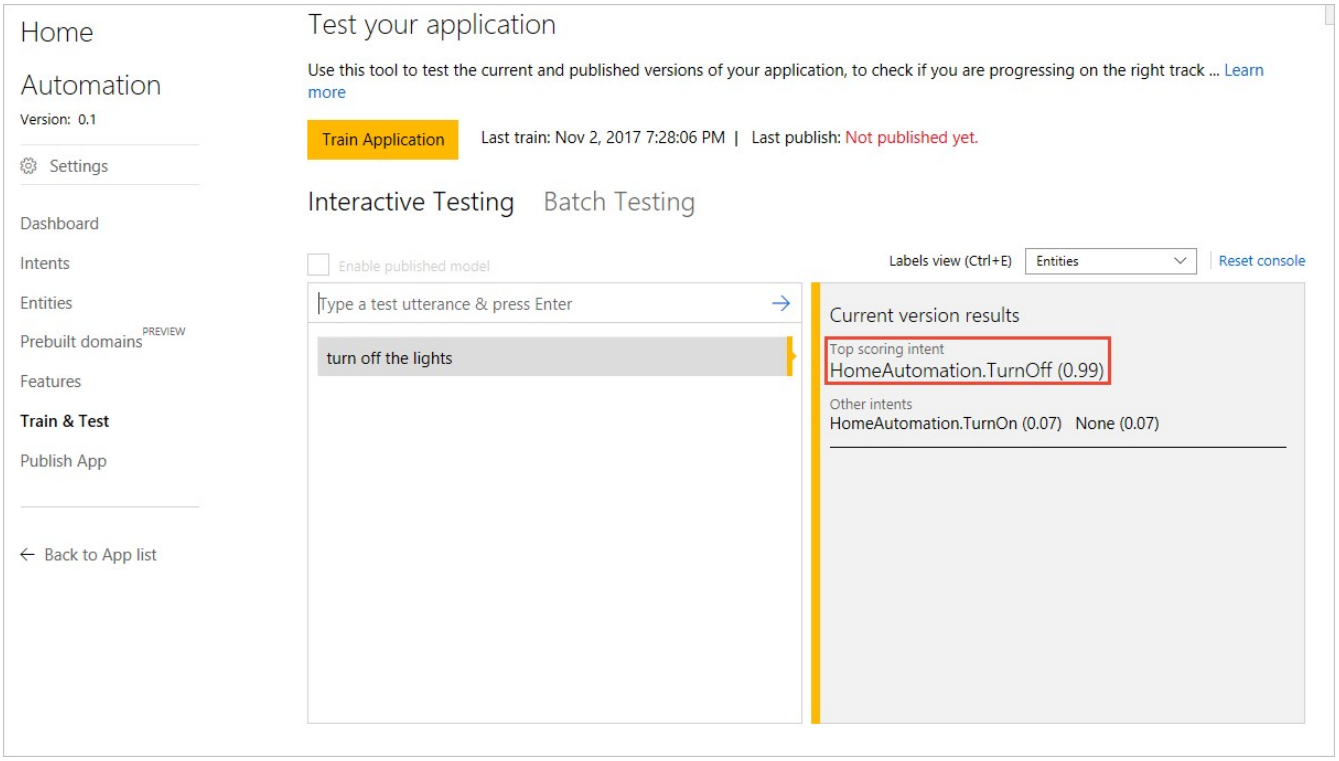


2. Type a test utterance like "Turn off the lights" into the Interactive Testing pane, and press Enter.

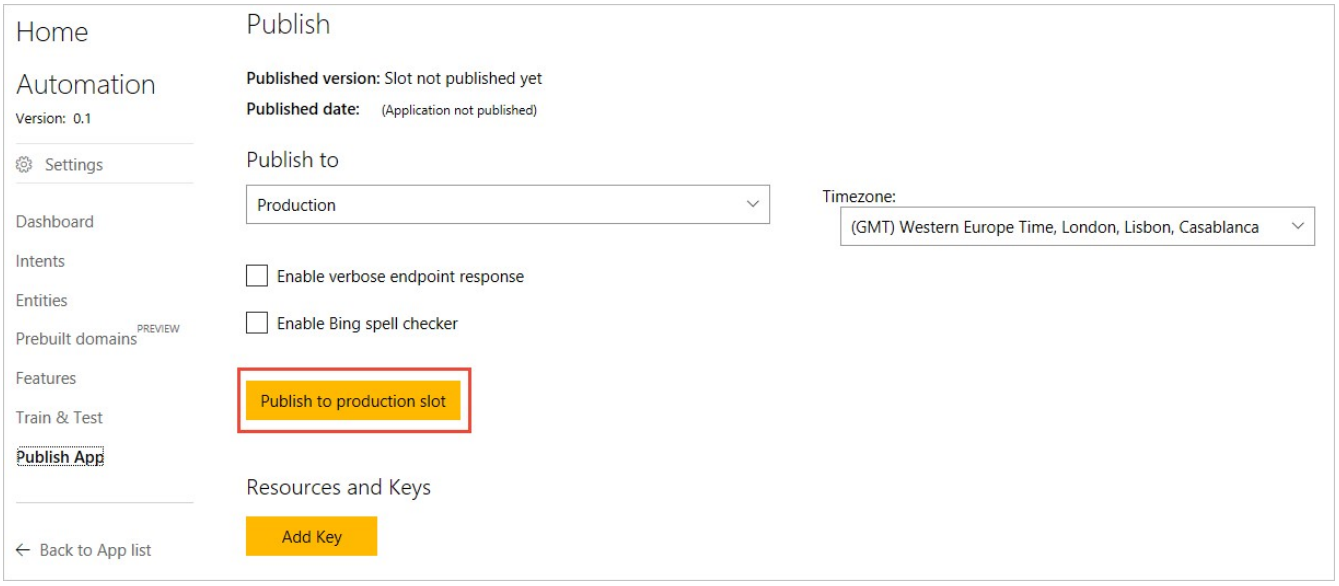
```
Turn off the lights
```

3. The results display the score associated with each intent. Check that the top scoring intent corresponds to the intent you expected for each test utterance.

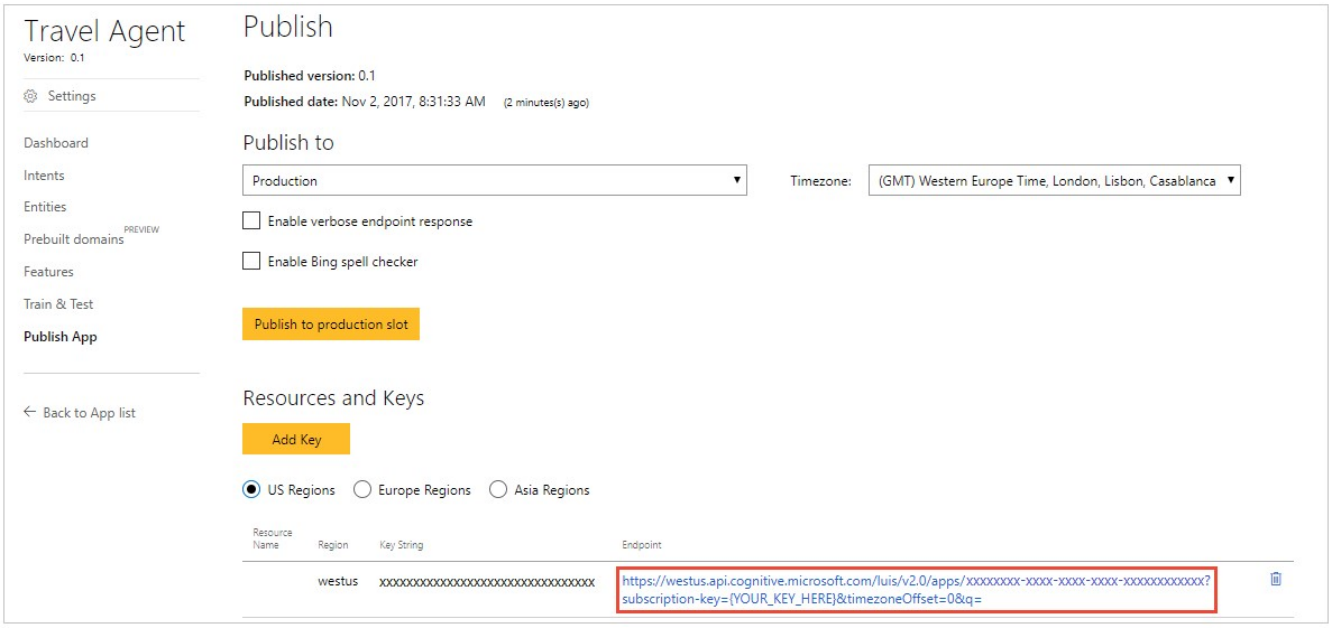
4. In this example, "Turn off the lights" is correctly identified as the top scoring intent of "HomeAutomation.TurnOff."



5. Select **Publish App** from the left-side menu and click the **Publish** button.



6. After you've successfully published, you can use the Endpoint URL that the **Publish App** page displays.



7. Copy the URL, then replace the `{YOUR-KEY-HERE}` with one of the keys listed in the **Key String** column for the resource you want to use. To open this URL in your browser, set the URL parameter "&q" to your test query. For example, append `&q=turn off the living room light` to your URL, and then press Enter. The browser displays the JSON response of your HTTP endpoint.

> **Up Next:** Create a QNA Maker Service