# Creating a LUIS Service

* Go to Portal.Azure.com and log in
* Click on the create a resource button and search for “Language Understanding” and click on it
* Press the “create” button
* Under the create options select both and fill in the rest of the information. Using the free authoring and runtime pricing tiers should be sufficient. Press the “create” button.
* When the deployment is complete, open your new service

# Feature Extraction

* To set up our model open a new tab and browse to the LUIS portal at https://www.luis.ai/
* Log in with your Azure credentials
* Ensure that you are using the version that is tied to Azure services
* In the portal create a new app.
* Give it a unique name and make sure the language is English. Press the “Done” button.
* In the newly created app go to the build tab.
* Click on entities on the left.
* From the collateral folder open the LuisTraining.docx.
* Create a new Entity in Luis of type list and call it Reservation Type. Enter all the entity names from the list and add as many synonyms as you can think of from each
* When complete to go intents and create a new intent for each item in the list. Think of different ways to say each intent and enter them. Include words like hotel, airfare, car rental, etc.
* Add some nonsense examples under the None intent.

# Classification and Prediction

* When complete, press the train button.
* After training is done, press the Test button. Try out other variations of the phrase and see if you get the expected results. You can press the “inspect” button. You may want to select an intent if the probability is not high “enough”. If the entity is not being picked up, it may be a good idea to add new synonyms.
* Don’t forget to re-train your model and re-test any changes.

# Deploying your Model

* When you are finally happy that you are getting the proper intent, we are ready to publish our model.
* Click on the Manage tab and select “Azure Resources”
* Press “Add Prediction Resource”
* Select your account for the tenant, the subscription you set up the service in and finally the service name. Press “Assign Resource”
* Also add an Authoring resource. If none exists, create one.
* Go to the “Application Information” tab on the left and set the training and endpoint settings to public
* In the Versions tab, click on the checkbox next to the current version of the model and press rename. Name this version “Current”
* Press the publish button in the upper right.
* Leave the staging radio button selected and click change settings and turn on sentiment analysis.
* Press “Publish”

Compared to Custom Vision, how curated do you think this process needs to be? Which has the greater chance for error?