# Category classification

Category classification is one of the fundamental natural language processing (NLP) challenges. With category classification, you can identify text entries with tags to be used for things like:

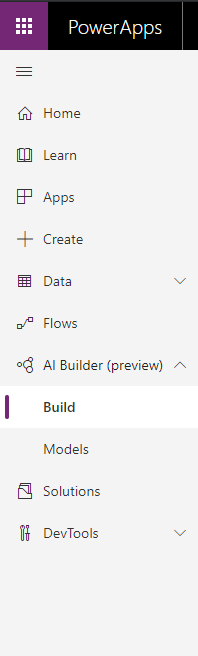
* Sentiment analysis
* Spam detection
* Customer request routing
* Other business needs

Automate and scale your business processes with AI Builder category classification in Power Automate and Power Apps. AI Builder models help free your employees to act on new insights. Use the results as an input for other AI capabilities, like subscription user churn and predictive analysis. AI Builder learns from your previously labeled text items and enables you to classify unstructured text data stored in Microsoft Dataverse into your own business-specific categories.

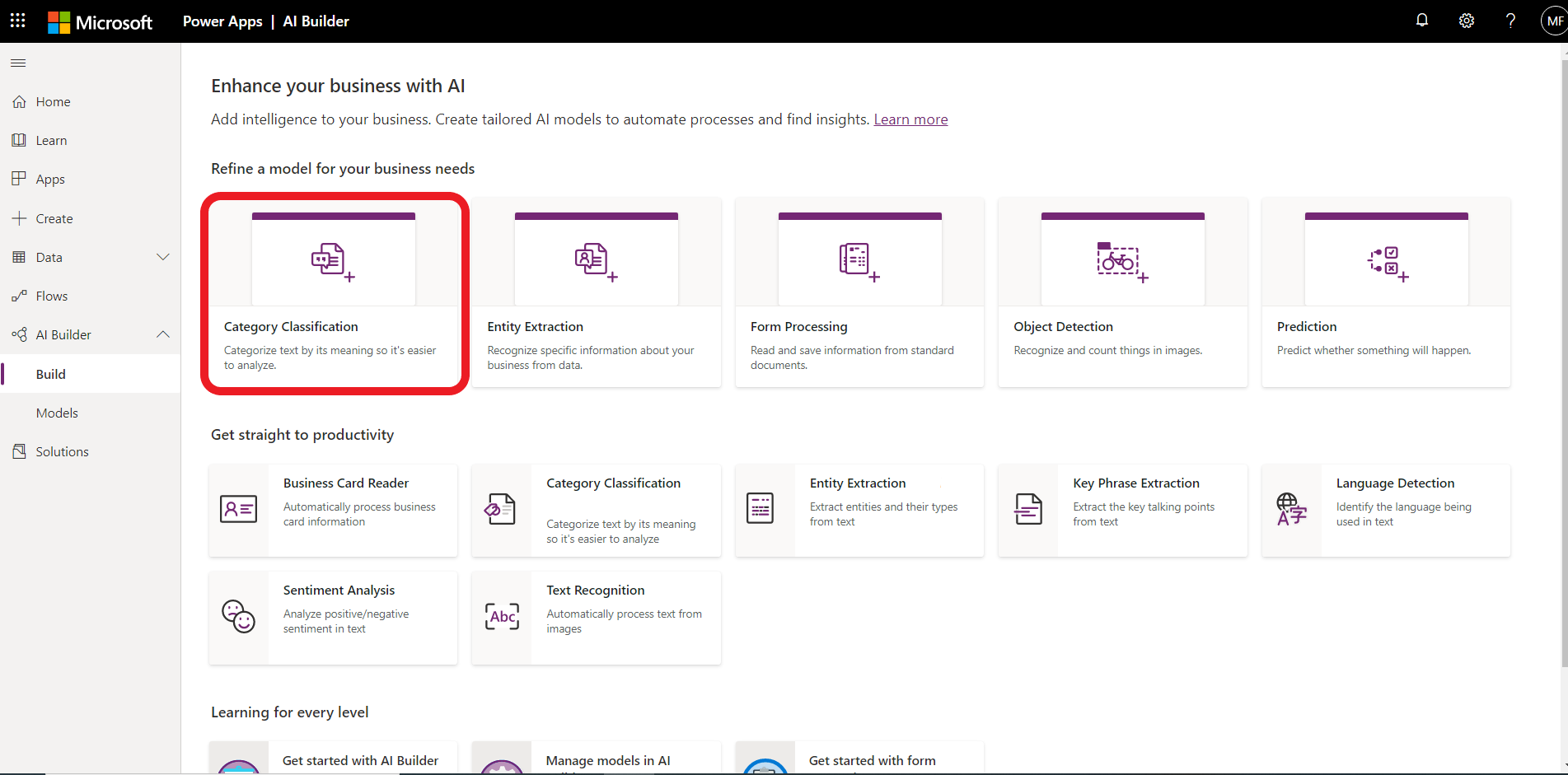
# Exercise 1

In the first exercise you will build and train your model.

1. From the left navigation, expand **AI Builder** and select **Build**.



1. Select **Category Classification**.



1. Name your model as **Healthcare Classification**. Because you are working in a shared environment make sure to include your name as part of the model name. This will make it easier to find later. Click **create**.

Graphical user interface, diagram

Description automatically generated

1. Your screen should look like the following image.

Diagram

Description automatically generated with medium confidence

1. Notice the **progress indicator** on the left.

Timeline

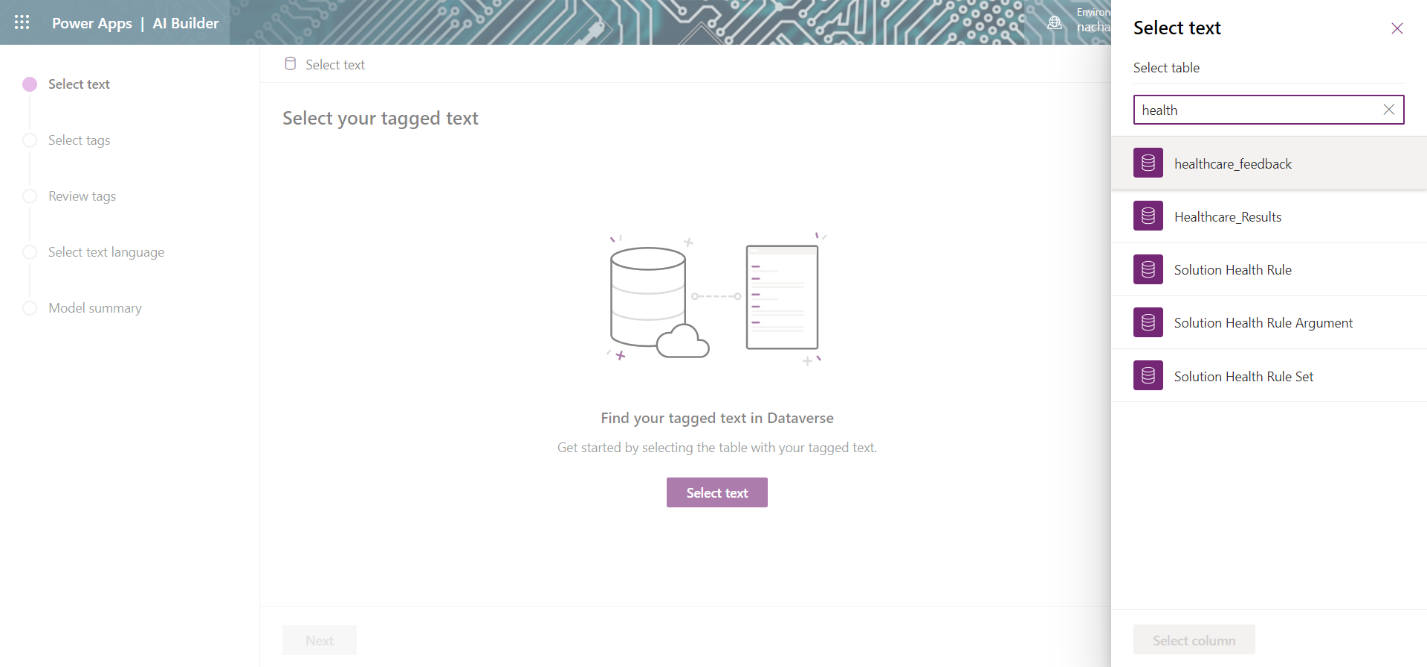
Description automatically generated

1. You will see **Quick tips** on the right.

Graphical user interface, text, application

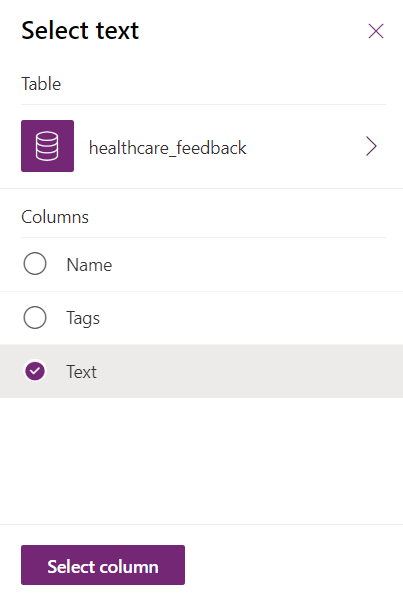
Description automatically generated

1. In the center we will build our model. Click on the **Select text** and select **healthcare\_feedback**.



**Note:** If you do not see heathcare\_feedback table, create a new one in Dataverse and upload data from Excel at Lab Files\Lab Data\TextClassification\pai\_healthcare\_feedbacks.csv

1. Select **Text** column. Click **Select Column**.



1. The next screen will show us preview. Click **Next**.

Graphical user interface, text, application, email

Description automatically generated

1. In the next screen, select **Tags**.

Graphical user interface

Description automatically generated

1. Select **comma** as tag separator. Click **Next**.

Graphical user interface, text, application, email

Description automatically generated

1. Review your text and tags. Click **Next**.

Graphical user interface, application

Description automatically generated

1. Select your text language as **English**. Click **Next**.

A picture containing table

Description automatically generated

1. Review the model summary. Click **Train**.

Graphical user interface, text, application

Description automatically generated

1. Publish your model. Once published the data will get scored, and scoring will happen daily for a published model.

Graphical user interface, application, Teams

Description automatically generated

# Exercise 2: Use the model in Power Automate

1. Create a Power Automate Flow as shown below:

Graphical user interface, text, application, email

Description automatically generated

1. Next, because the Predict action can work with different types of models, the output can contain different data. So future steps can easily use the output. We are going to use a Parse JSON action that will parse the output and make it available to subsequent steps with known names.
2. Click + New Step and type Parse JSON in the search field.
3. Select the Parse JSON action.

Graphical user interface, text, application, email

Description automatically generated

1. Click in the Content field and select Response Payload from the Dynamic content panel.

Graphical user interface, application

Description automatically generated

1. Next click in the Schema field and paste in the following. This is a schema that tells the parse action what to expect in the content. You will be able to find this in the docs site in the future.

{

"type": "object",

"properties": {

"predictionOutput": {

"type": "object",

"properties": {

"results": {

"type": "array",

"items": {

"type": "object",

"properties": {

"type": {

"type": "string"

},

"score": {

"type": "number"

}

},

"required": [

"type",

"score"

]

}

}

}

},

"operationStatus": {

"type": "string"

},

"error": {}

}

}

1. Let’s check to make sure things are good so far. Click Test in the upper right corner.
2. Choose I’ll perform the action and click Save & Test.
3. In the Feedback field type The facilities are wonderful.
4. Click Run Flow and then click Done.
5. Your flow should have run successfully and should look like the following. Click Edit to resume making changes.

Chart

Description automatically generated

## Task 3 Process the output

In this task we are going to take the output and use it to email the different departments. The Predict might classify multiple tag matches, each returned with a type and a score. We only want to send a copy of the feedback to the departments when the score is greater than 63. To accomplish this, we will add a condition action and check the score and the type. For each one that passes the criteria and we will send an email.

1. Click + New Step and search for Condition.
2. Select the Filter array action.

Graphical user interface, application, Word

Description automatically generated

1. Click in the Choose a value field and select type from the Dynamic content panel.

Graphical user interface, application

Description automatically generated

1. Because there could be multiple type and score records returned, flow will wrap your condition in an Apply for each action and it should now look like the following

Graphical user interface, application, Teams

Description automatically generated

1. Click on the Condition action to expand it.
2. Click in the second Choose a value field type Staff.

Graphical user interface, application

Description automatically generated

1. Click + Add and then + Add Row to add a second check in the condition for type.

Graphical user interface

Description automatically generated with medium confidence

1. Click in the Choose a value field, and select score in the Dynamic content pane.

Graphical user interface, application, Teams

Description automatically generated

1. Change the operator to is greater than and type 0.63 in the second value field.

Graphical user interface, application

Description automatically generated

***Double check that the value is 0.63 and not .63 otherwise the flow will fail.***

1. Inside the If Yes path of the condition, click Add an action

Graphical user interface, text, application

Description automatically generated

1. Search on Send an Email.
2. Select Send an **email (v2).**
3. In the To field type your lab user.
4. In the subject type Staff Feedback.
5. Click in the body and select Feedback in the Dynamic content panel.

Graphical user interface, text, application

Description automatically generated

1. Click Test and then choose I’ll perform the action and then click Save and Test.
2. In the feedback paste the following:

The doctors were very professional, nurses were kind and attentive, the room was nice but the food at the cafeteria and in room is terrible

1. Click Run Flow.
2. Your flow should run successfully and should look something like the following. You should also now have an email with the feedback.

Graphical user interface

Description automatically generated with medium confidence