Coffee Brand Classifier

PIC32CX-BZ2 / WBZ451 and BME688 4-in-1 Gas Sensor

Objective:

Build an intelligent Coffee bean(powder) classifier using PIC32CX-BZ2 / WBZ451 and BME688 4-in-1 Gas Sensor in conjunction with the MPLAB Machine Learning Development Suite.

The ML Development Suite streamlines the development of machine learning solutions specifically tailored for application and deployment on Microchip's microcontrollers and microprocessors. This manual offers a comprehensive walkthrough, demonstrating the steps to gather Humidity, Gas(resistance), Pressure, Temperature using BME688 Gas sensor integrated with PIC32CX-BZ2 / WBZ451 Machine Learning Evaluation Kit, transmit the acquired data to the ML Model Builder, construct a personalized model capable of accurately classifying the input data, and ultimately deploy the model onto the Machine Learning Evaluation Kit.

Materials:

Hardware:

2 X PIC32CX-BZ2 and WBZ451 Curiosity Development Board

EEMB Lithium Polymer Battery 3.7V 820mAh 653042 Lipo Rechargeable Battery Pack with Wire JST Connector

Environment 3 Click (BME688 4-in-1 Gas Sensor)

Three different Coffee Brands powders/ beans

Glass Container with lid.

Software Tools:

- MPLAB X IDE (Integrated Development Environment) (v6.05 or later)
- XC 32 Compiler
- MPLAB ML Development Suite
    - ML Data Collector
    - ML Model Builder

Connection Diagram

1] Transmitter or data capturing device(peripheral):

- PIC32CX-BZ2 and WBZ451 Curiosity Development Board
- The Environment 3(Gas Sensor) click board is connected to the mikroBUS™ socket. Check the pin labels when connecting the boards to ensure that they are connected properly.
- Connect the lipo Rechargeable battery to the JST connector port.

2]

Connect the second PIC32CX-BZ2 and WBZ451 Curiosity Development Board with a USB to the PC which would act as the Host for receiving the data signal

Setup:

In a glass container having coffee beans/ powder keep the data capturing device set up inside the it. Put the lead on it and leave it for capturing the data. Connect the second board to the PC via USB cord.

Procedure

Before getting started, be sure to install all the necessary software tools. We will work within MPLAB X IDE and build the project with the MPLAB Machine learning Development Suite and compile with XC32 . Both MPLAB DV and the Machine Learning Plugin are available within the plugin update center in MPLAB X IDE. You will also need to sign up for mymicrochip account to access MPLAB X ML Development Suite.

Data Collection

The initial phase of constructing a machine-learning model involves the acquisition of data. This dataset serves as the foundation for training the ML model, enabling it to discern various types of exercises. In this particular instance, the model will be trained to identify three different coffee Brands.

Exercise Files:

- The firmware and dataset used in this tutorial and other MPLAB X project files can be found in the latest GitHub release.

Data Collection Overview

In this section, the recommended approach for gathering data samples to build the Coffee bean classifier model is discussed.

To begin the data collection process, it is essential to identify a suitable sensor configuration for your specific application. This will involve determining the optimal placement of the sensor, selecting an appropriate installation method, and configuring the signal processing parameters, such as the sample rate and sensitivity.

Sensor Installation

- Attach the bme688 sensor click board to the PIC32CX-BZ2 board and attach the battery to this board. By pressing the reset button simultaneously on both the pic place it inside the glass jar which contains the coffee beans/ powder. Close the lead of the jar

Data Collection Protocol

To proceed with the data collection process, the next essential step involves creating a protocol that outlines the necessary procedures for collecting data. This entails determining the number of samples to be collected, the metadata parameters to be captured and other relevant parameters that will govern the data collection procedure. The protocol will serve as a guide for executing the data collection process and ensure that the data collected is consistent and accurate.

Data Collection: Metadata

Initially, it is imperative to address metadata, as it significantly contributes to contextualizing our data. In this application, our metadata comprises solely the dataset, which is partitioned into training and testing subsets.

Data Collection: Sampling Method

At this stage, it is necessary to decide the data sampling process for our application. This involves determining the number of samples to be collected and defining the necessary steps for measuring the data. In the context of this example application, the methodology can be summarized in the following steps:

The sensor needs to be heated up for the initial 20mins and then place inside the coffee jar.

After that collect the data in sessions of 30mins each and save the data in multiple different files

Record and label segments for each of the Coffee brands .

In between changing the coffee let the sensor be disconnected form the power source (battery) for a period for it to cool down. Ideally 15 mins minimum.

How to Configure, Compile and Flash

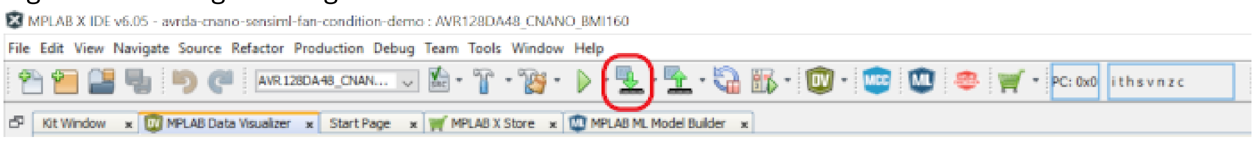
This document explains the steps involved in configuring the data logger firmware build, compiling it and flashing it to the PIC32CX-BZ2 device. Follow the instructions below:

1. Connect the Curiosity board to your PC using a USB cable.

2. Install the MPLAB X IDE and XC32 compiler on your PC. These are essential tools for loading the data logger project and programming the PIC32CX-BZ2 board.

3. Open the firmware project folder in MPLAB X.

Figure 1-19. Programming the device



Firmware Operation

The data streamer firmware will output sensor data over the UART port with the following UART settings:

- Baud rate 115200
- Data bits 8
- Stop bits 1
- Parity None

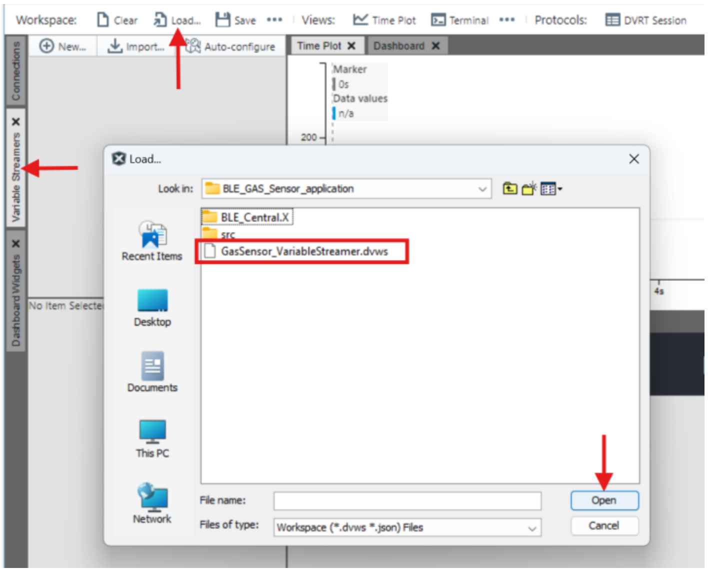
Usage with the MPLAB Data Visualizer and Machine Learning Plugins

This project can be used to generate firmware for streaming data to the MPLAB Data Visualizer plugin by setting the DATA_STREAMER_FORMAT macro to DATA_STREAMER_FORMAT_MDV as described above. When the firmware is flashed, follow the steps below to set up Data Visualizer.

1. Connect the Curiosity board to your computer, open up MPLAB X, then open the Data Visualizer plugin.

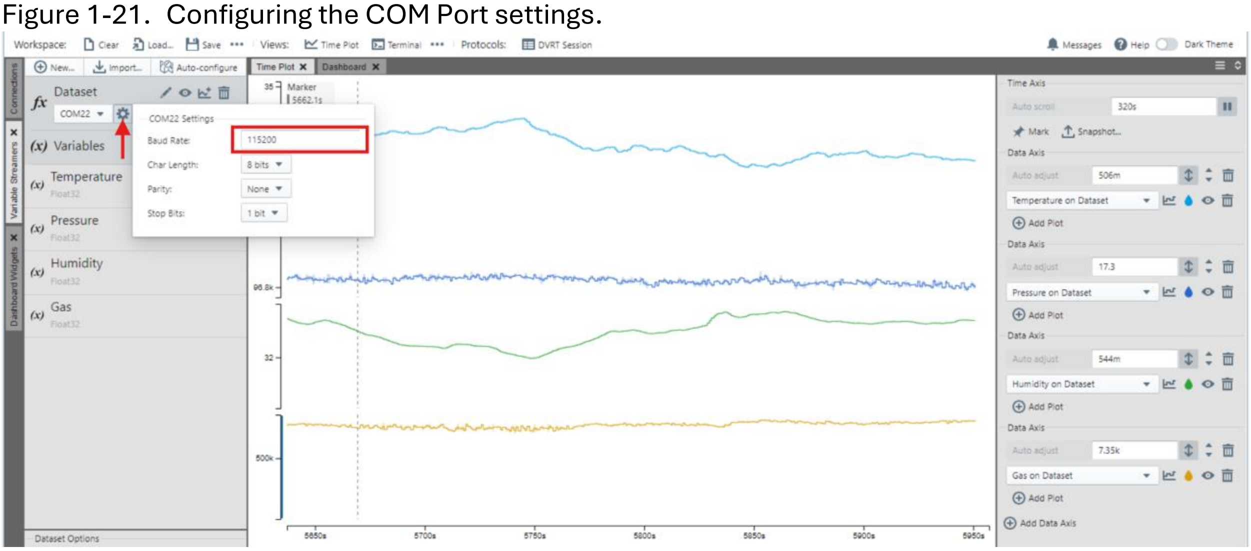
2. Click the **Load Workspace** button as highlighted in the image below. Select one of the workspace files included in this repository - located under the mplab-dv-workspaces folder - whose name most closely describes your sensor configuration; you can always modify the configuration when it is loaded if needed.

Figure 1-20. Loading .dvws /.json file to variable streamers.



1. Next, select the Serial/CDC Connection that corresponds to the Curiosity board, adjust the baud rate to 115200, then click **Apply**.

Figure 1-21. Configuring the COM Port settings.



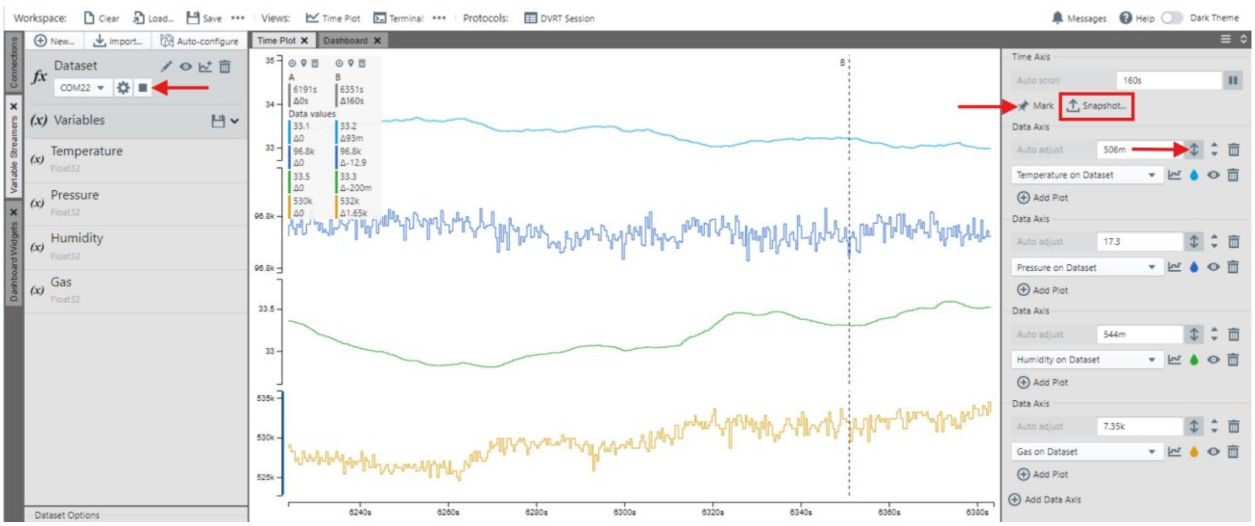
1. Use the play button on the Serial/CDC Connection to connect to the serial port. When the connection is made, the PIC32CX-BZ2 is available for use with the variable streamer.

2. Switch to the **Variable Streamers** tab, then select the aforementioned Serial/CDC Connection from the dropdown menu as the input data source for the BME688 variable streamer (see image below for reference); this will enable parsing of the PIC32CX-BZ2 data stream. You may delete or add variables here if your sensor configuration differs from the pre-loaded ones.

3. The Sensor data is now available in the time plot. Double click anywhere within the time plot to start/stop scrolling of the time axis.

4. Click the up down arrow button to auto set the time scale.

5. Click the mark icon button, then the snapshot icon button to snip the part of the time plot as data to save.



Visit the *Machine Learning Plugin* page to learn more about using the Data Visualizer plugin to export your data for machine learning applications.

Model Development

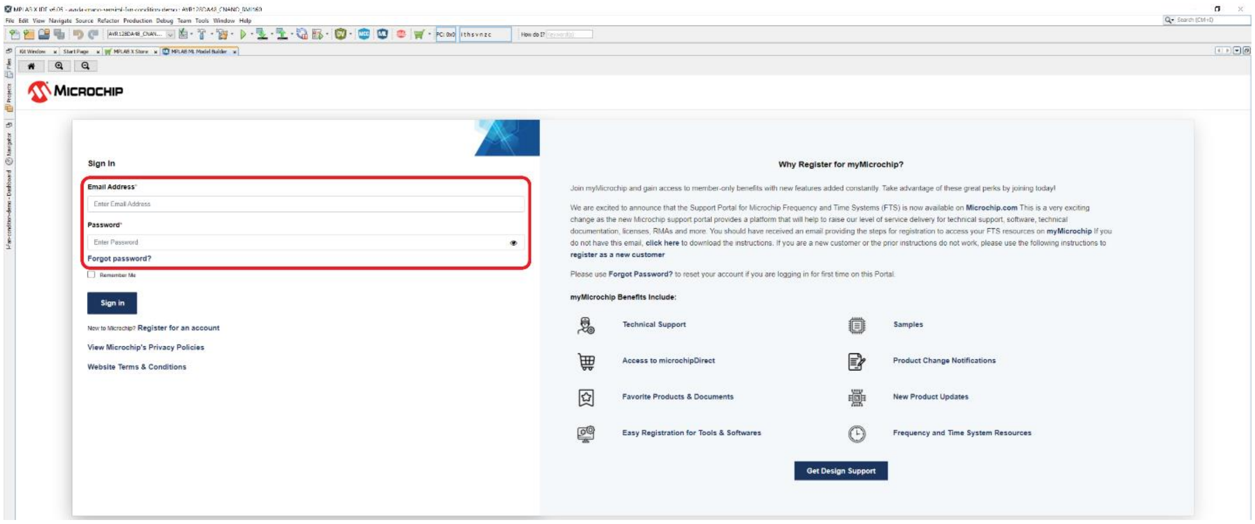
Once the initial dataset for your application has been gathered, the next step is to proceed to the ML Model Builder to create your classifier model. To generate your classifier model, proceed to MPLAB ML Development Suite.

The process for accessing the Microchip ML Development Suite requires the following steps:

- Launch the MPLAB X IDE software.

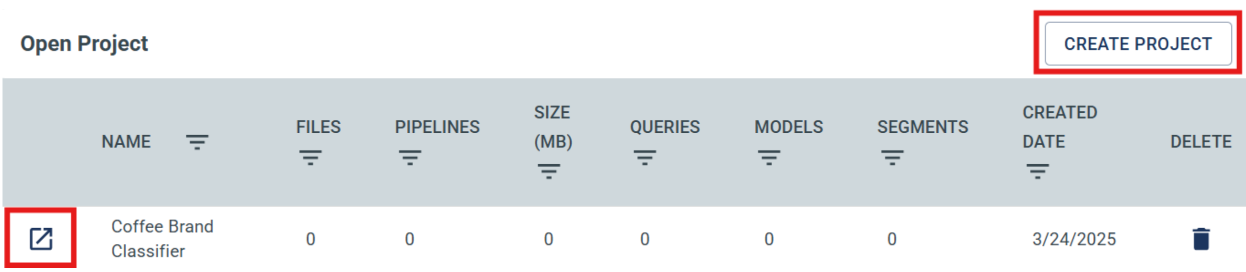
- Next, launch the ML Model Builder tool.

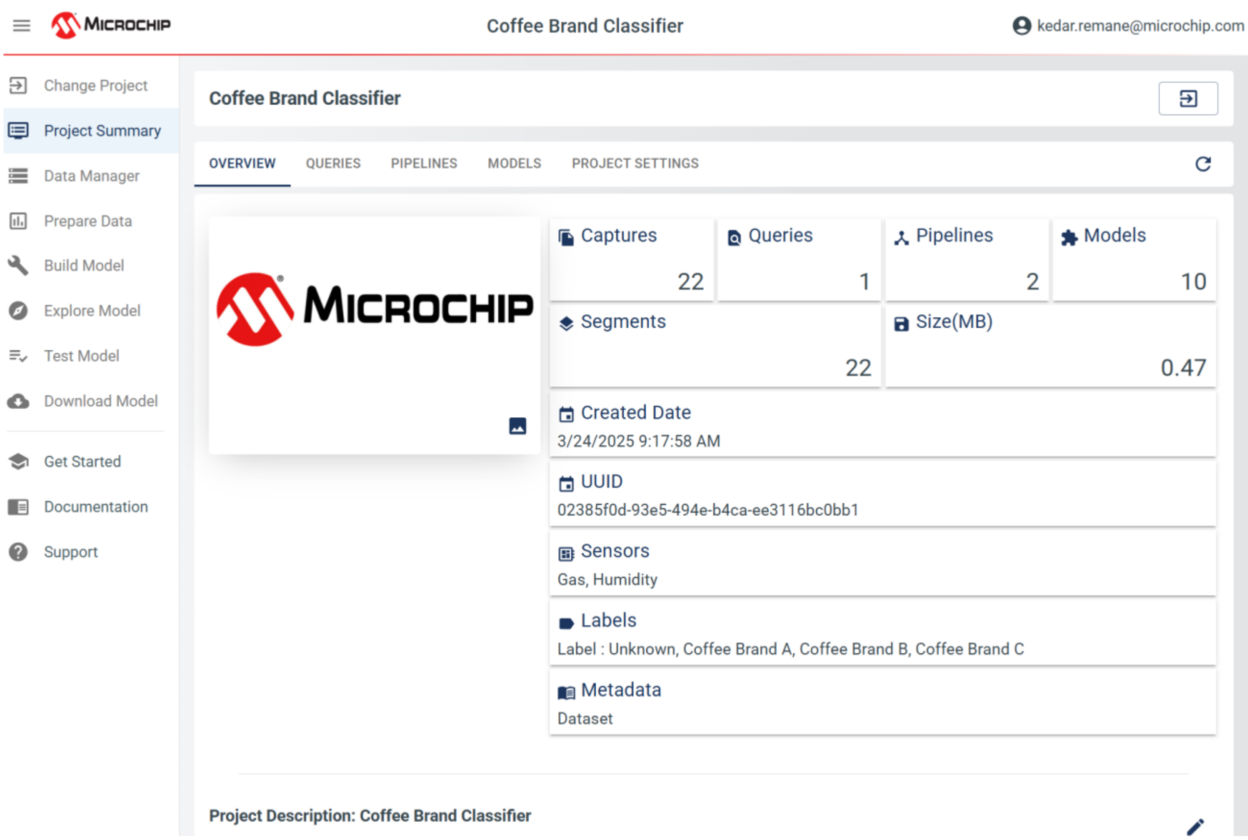
- When the ML Model Builder page loads, you will be prompted to authenticate your account by entering your myMicrochip login credentials.



Upon successful login, users can create a new project by selecting the "CREATE PROJECT" tab, which prompts a pop-up window to specify a project name. To access a created project, simply click the "Open Project" icon button located before the project name.

Figure 1-29. Creating a new ML Project.

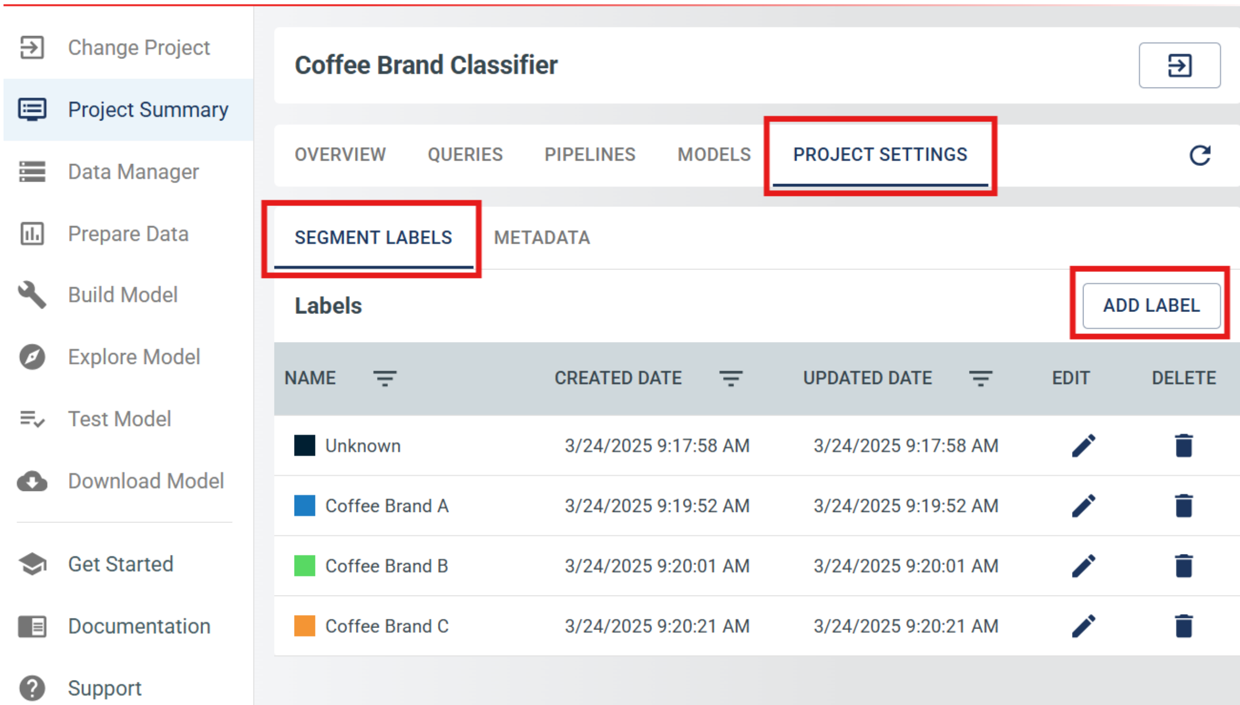


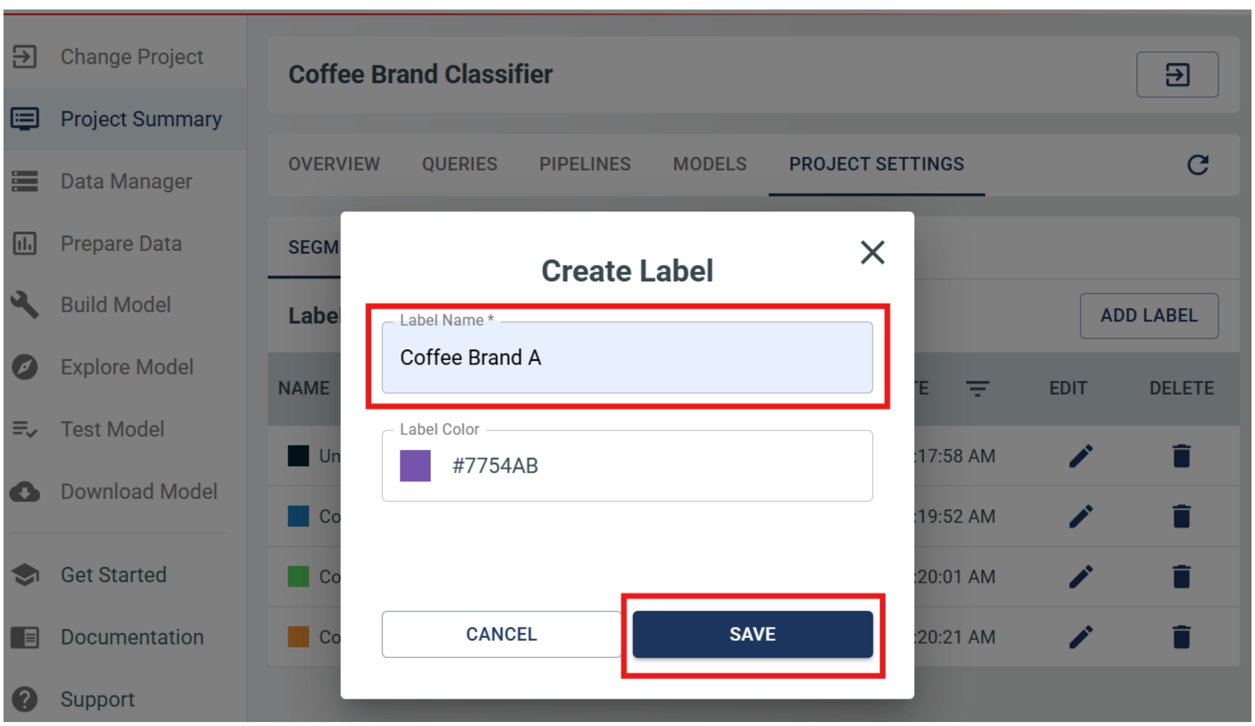


To create new labels and metadata fields for a project, follow these steps:

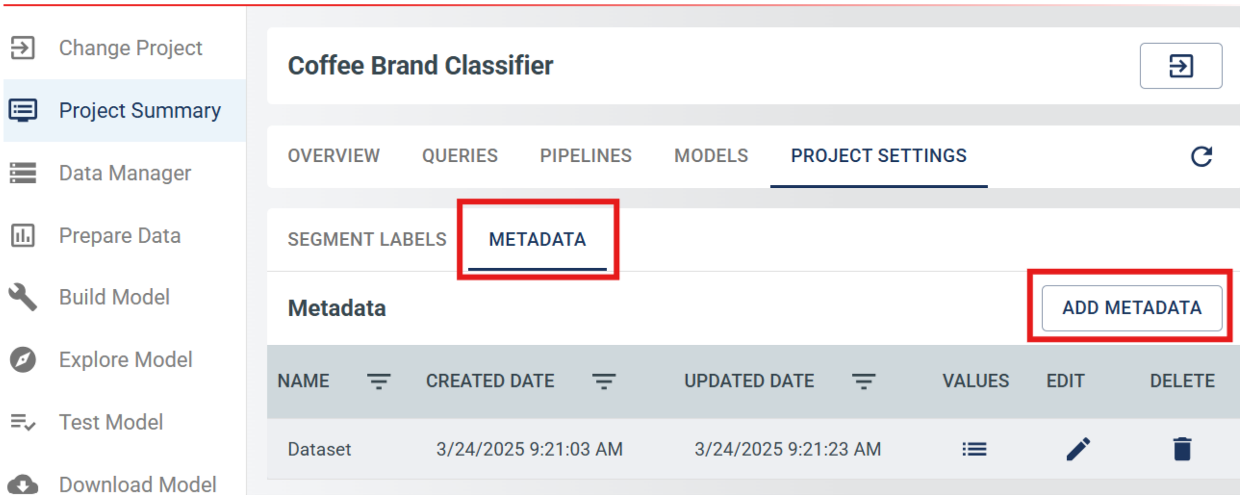
1. Navigate to the Project Summary page.
2. Access Project Settings.
3. Locate the "Segment Labels" and "Metadata" fields.
4. Select the "Add Label" and "Add Metadata" tabs positioned in the top right corner of the page.
5. Input the necessary details for the new label and metadata fields.
6. Save the information to generate the new label and metadata field.
7. Repeat steps 4-6 for each additional label or metadata field desired.

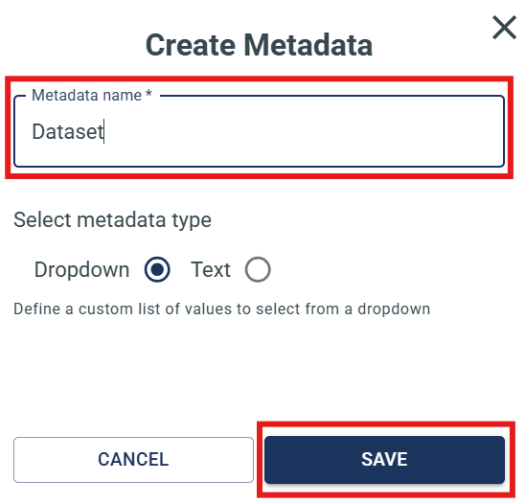
Figure 1-30. Configuring Project settings.

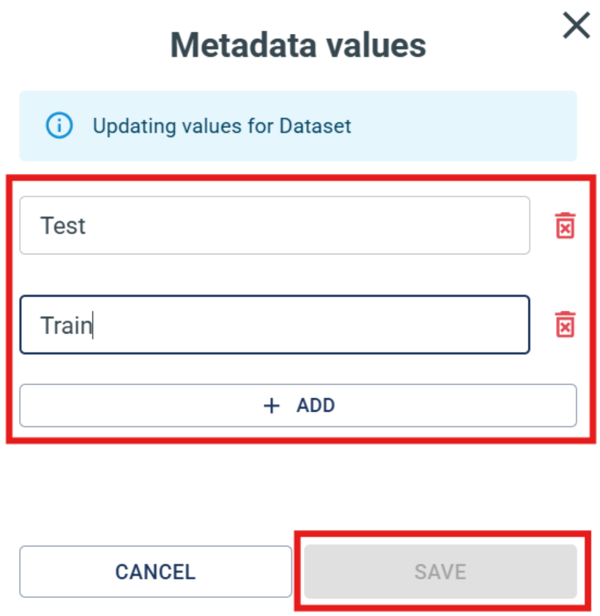




Next, we proceed to creating the Metadata:

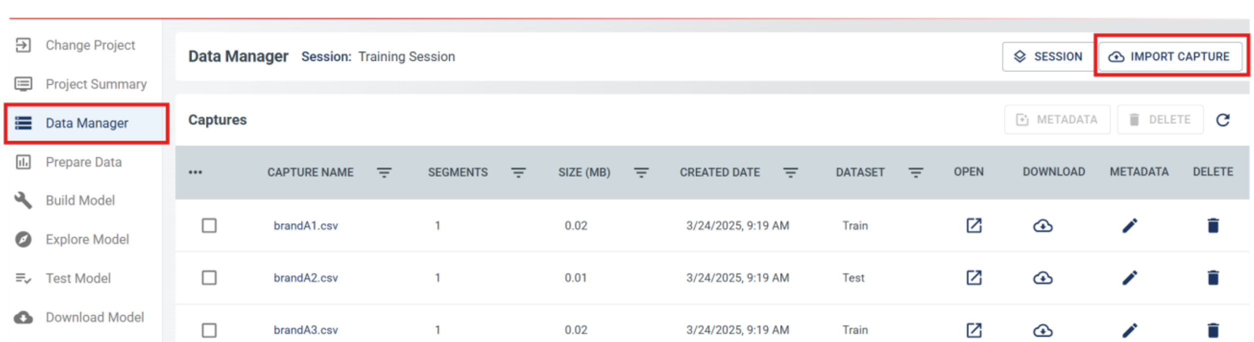

Change Project
Project Summary
Data Manager
Prepare Data
Build Model
Explore Model
Test Model
Download Model
Coffee Brand Classifier
OVERVIEW
QUERIES
PIPELINES
MODELS
PROJECT SETTINGS
SEGMENT LABELS
METADATA
Metadata
ADD METADATA
NAME
CREATED DATE
UPDATED DATE
VALUES
EDIT
DELETE
Dataset
3/24/2025 9:21:03 AM
3/24/2025 9:21:23 AM

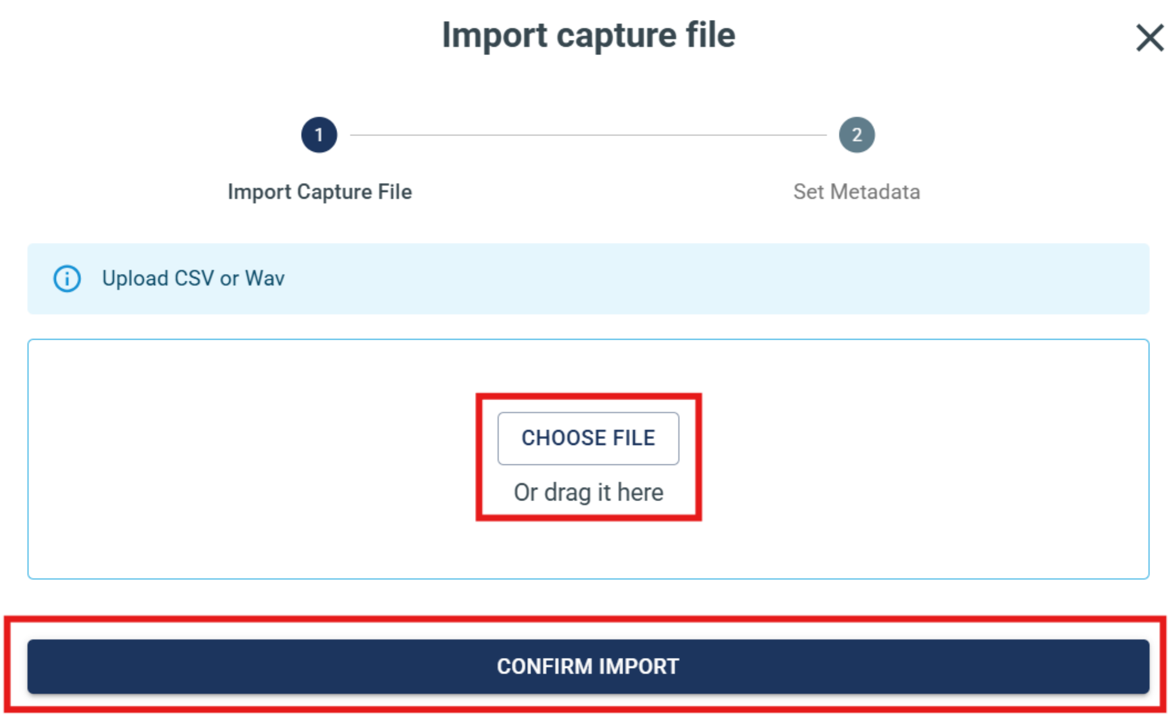

Create Metadata
Metadata name *
Dataset
Select metadata type
Dropdown
Text
Define a custom list of values to select from a dropdown
CANCEL
SAVE



Next, to import necessary data files for the project, follow these steps:

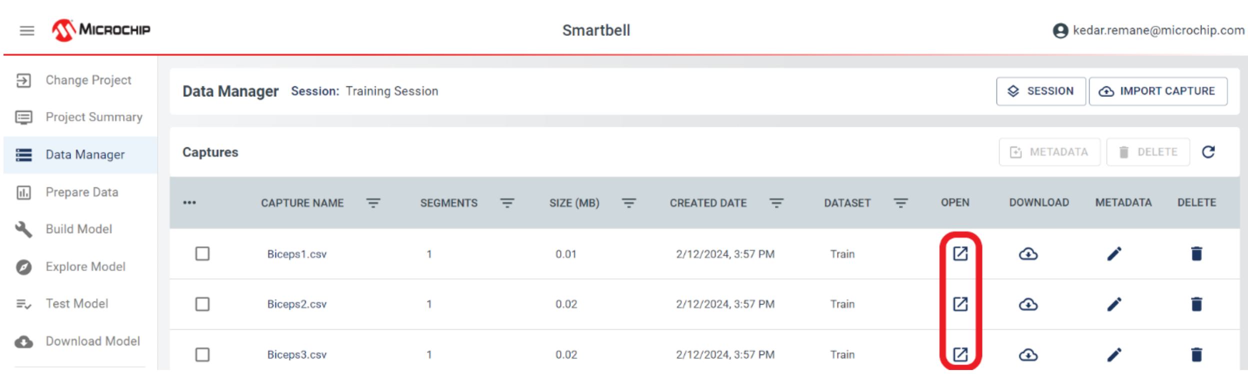
1. Navigate to the **Data Manager** tab.
2. Locate the **IMPORT CAPTURE FILE** button, then click it.
3. Ensure that the data files to be imported are in either CSV or Wav format.
4. Select all the files required for the project.
5. Begin the import process.





Upon completion of the import process, the imported data files will be ready for utilization within the project. It is crucial to confirm that all data files are correctly formatted and labeled to guarantee precise analysis and interpretation of project outcomes.

To access the imported data files, click the icons corresponding to the desired file under the OPEN column



Once the desired file is open, navigate to the **CREATE** tab to begin segmenting the data. To create segments, drag the segments axis to the desired locations within the data.

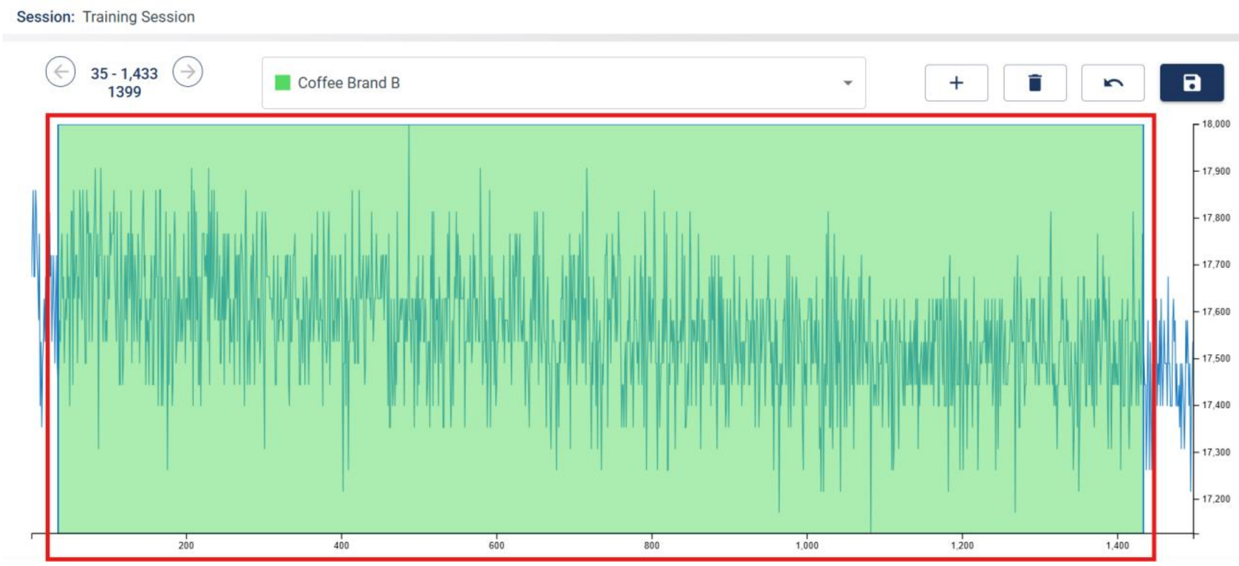


Figure 1-32. Creating segments inside data file.

It is, then, recommended to correctly categorize these segments using appropriate labels and metadata fields for ease of organization and future analysis.

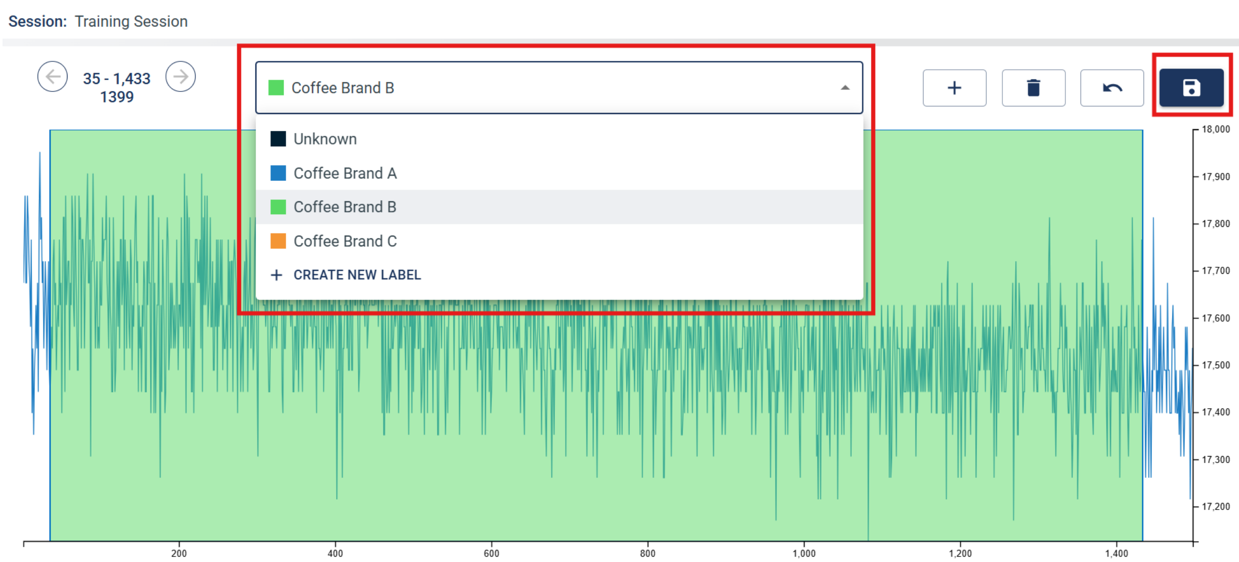
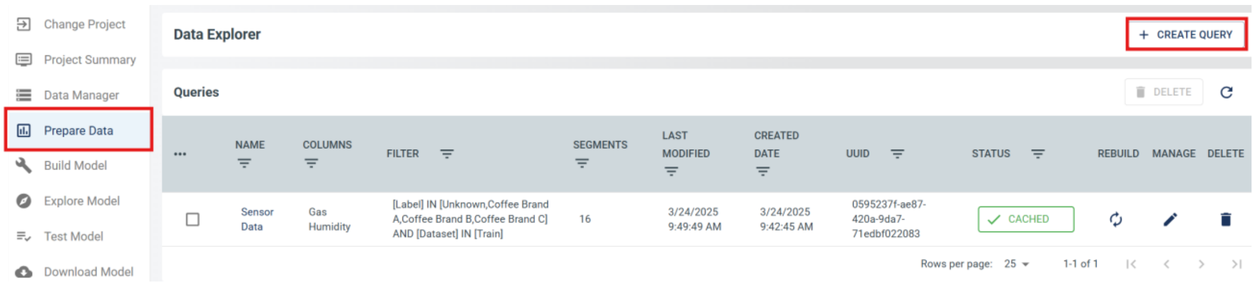


Figure 1-33. Labelling the segments.

To proceed, navigate to the Prepare Data view. From there, locate and click the **CREATE QUERY** button to initiate the data preparation process.

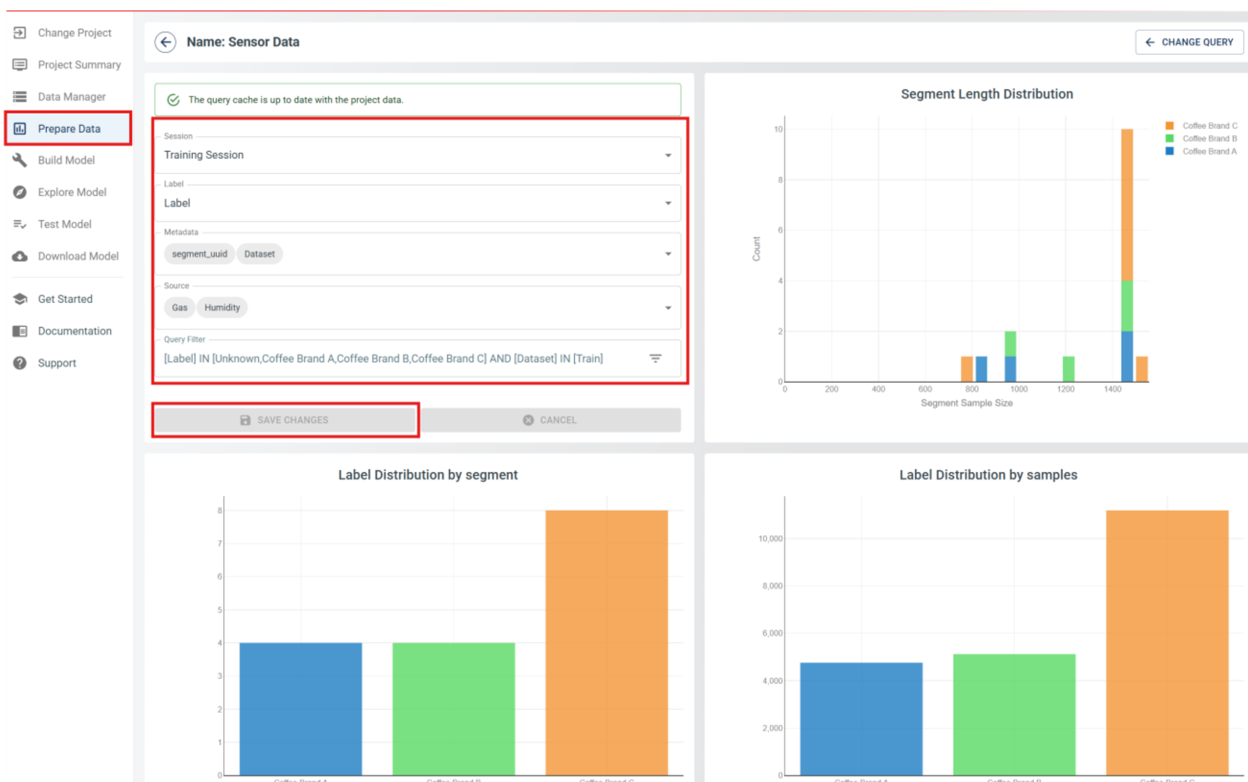
Figure 1-34. Creating new project query.



1. Fill out the block as displayed below.
2. Press the **CACHE** button to initiate the caching process.
3. Wait for the cache to build out completely.
4. Click the **SAVE** button when the cache is fully built.

Following these steps will result in the block saving cached information. Once the caching process is complete, details such as Segment Length Distribution, Label Distribution by segment, and Label Distribution by samples will be displayed.

Figure 1-35. Filling in details for the query.



To proceed with building the pipeline, navigate the Build Model module and, then locate the Create Pipeline tab

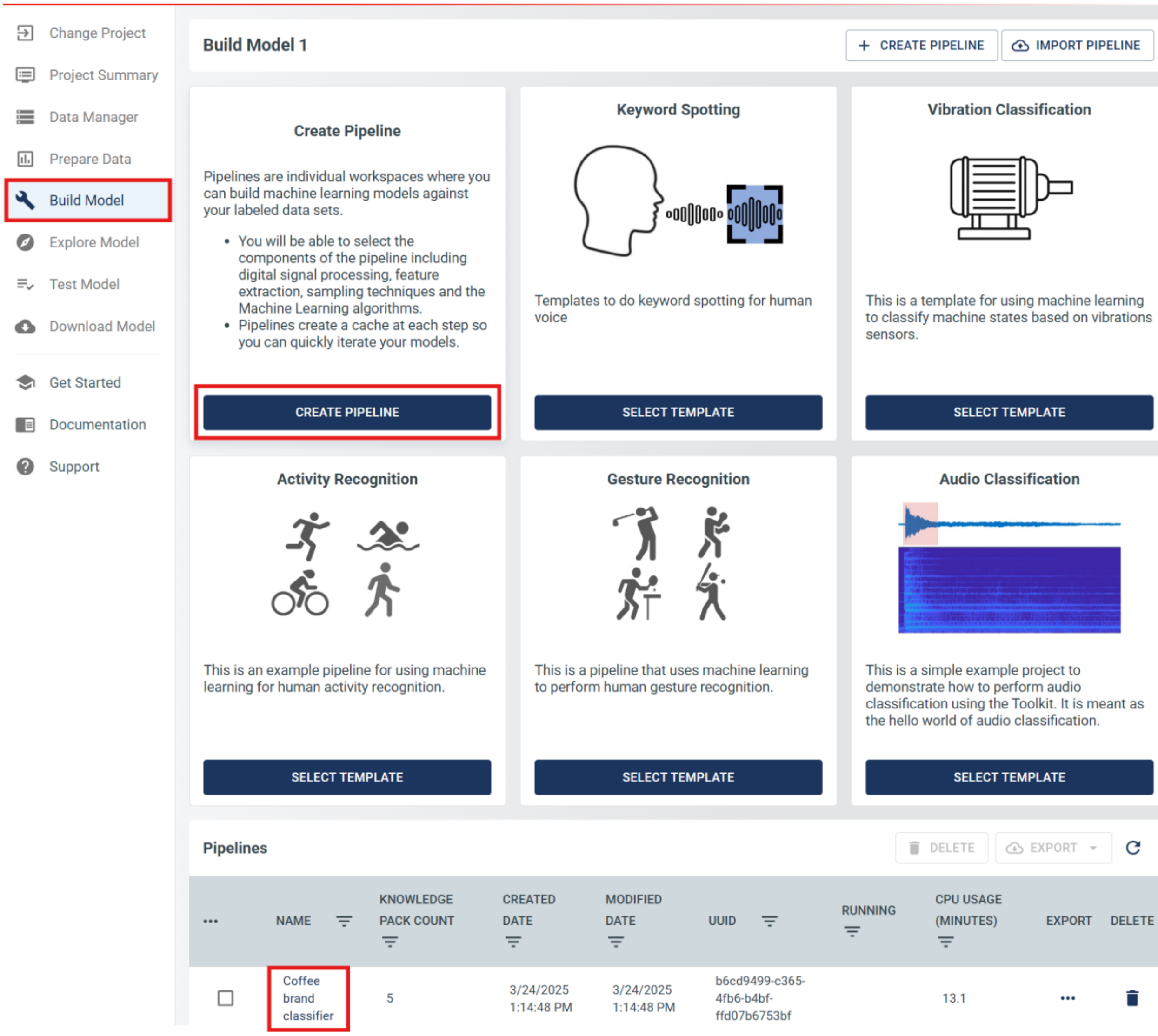


Figure 1-36. Creating a new pipeline.

After clicking **CREATE NEW PIPELINE**, information regarding Pipeline displays. Click **NEXT** to proceed.

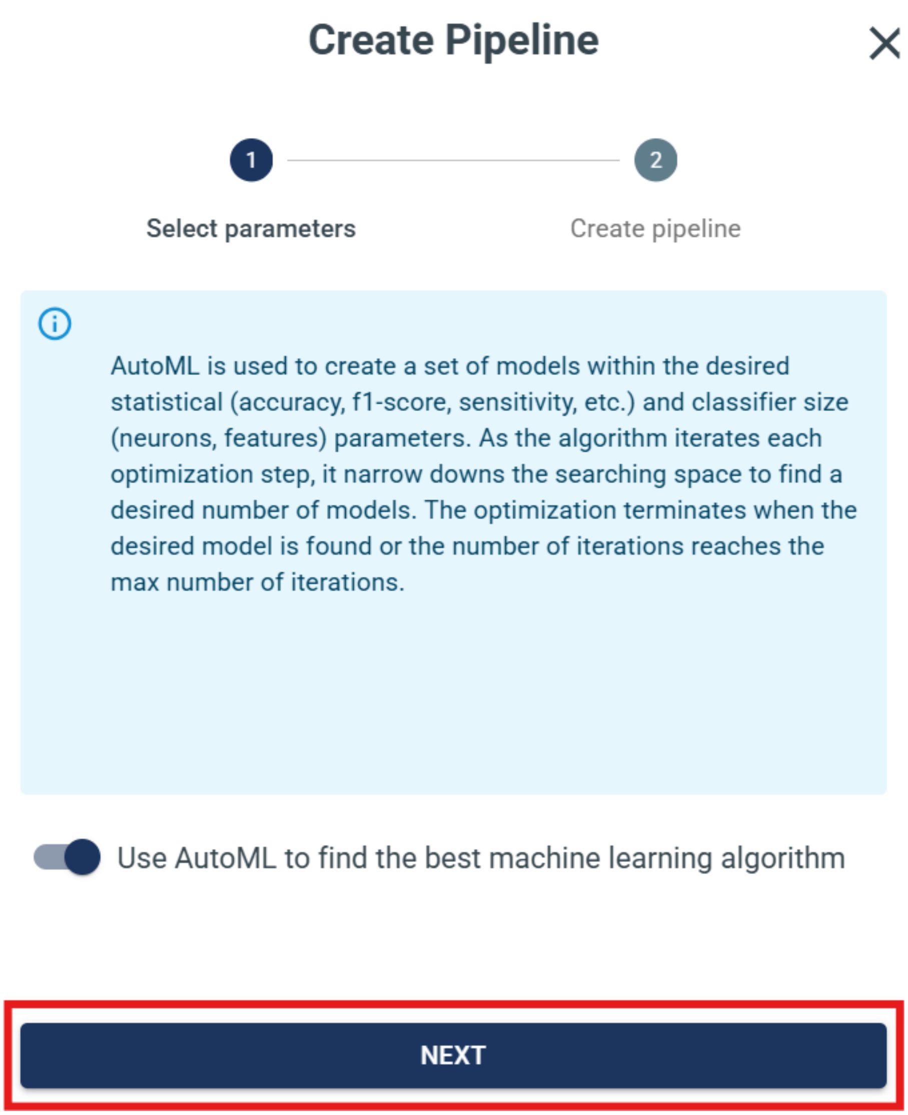
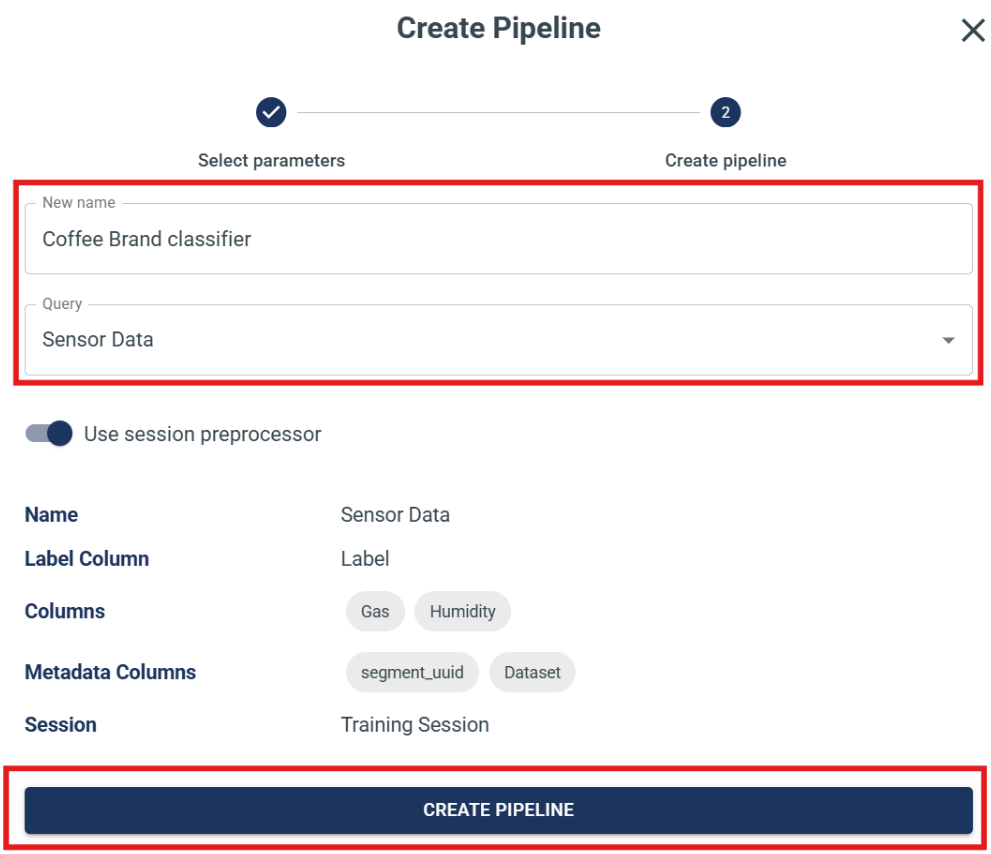


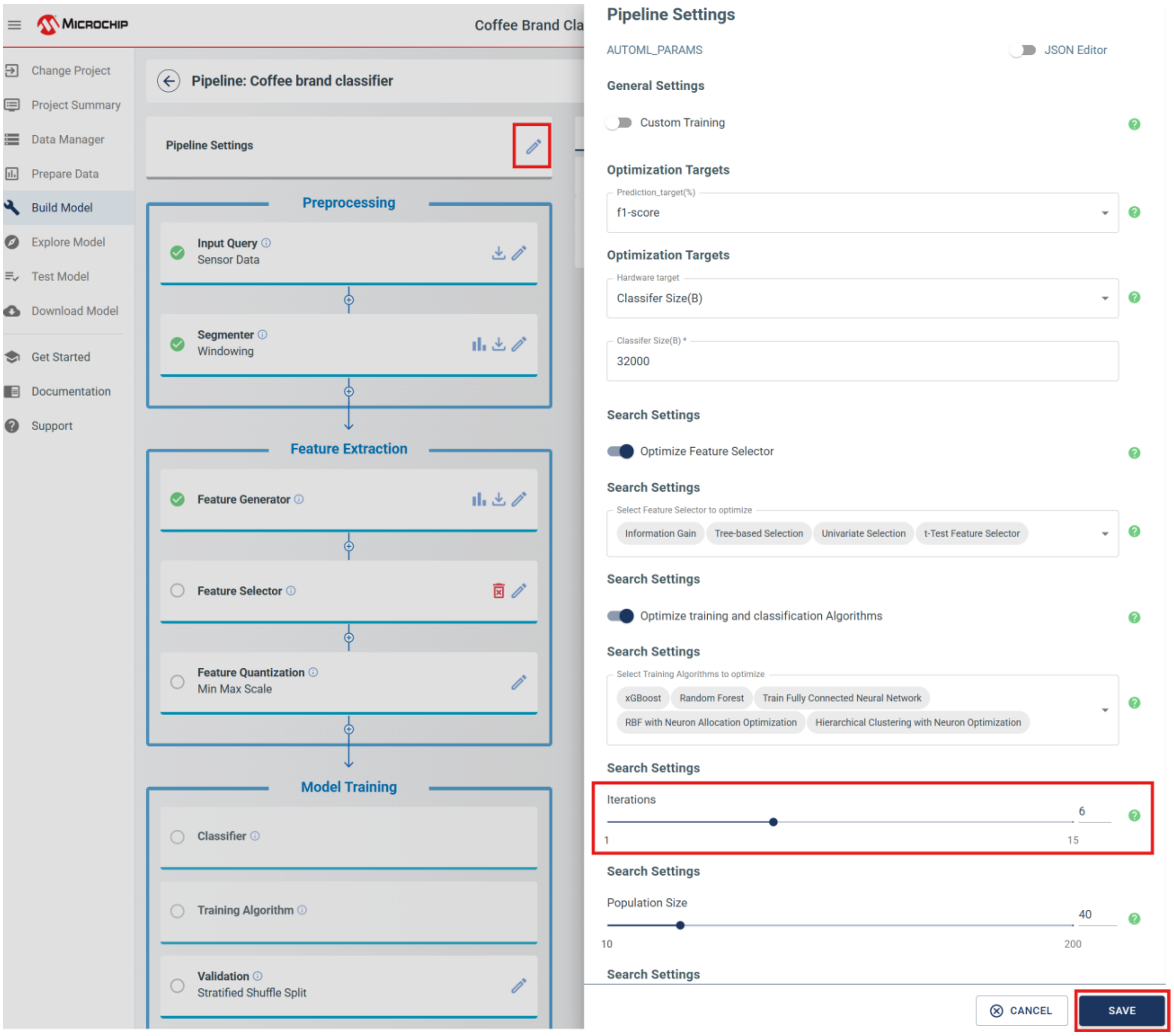
Figure 1-37. Naming the new Pipeline and selecting the query.



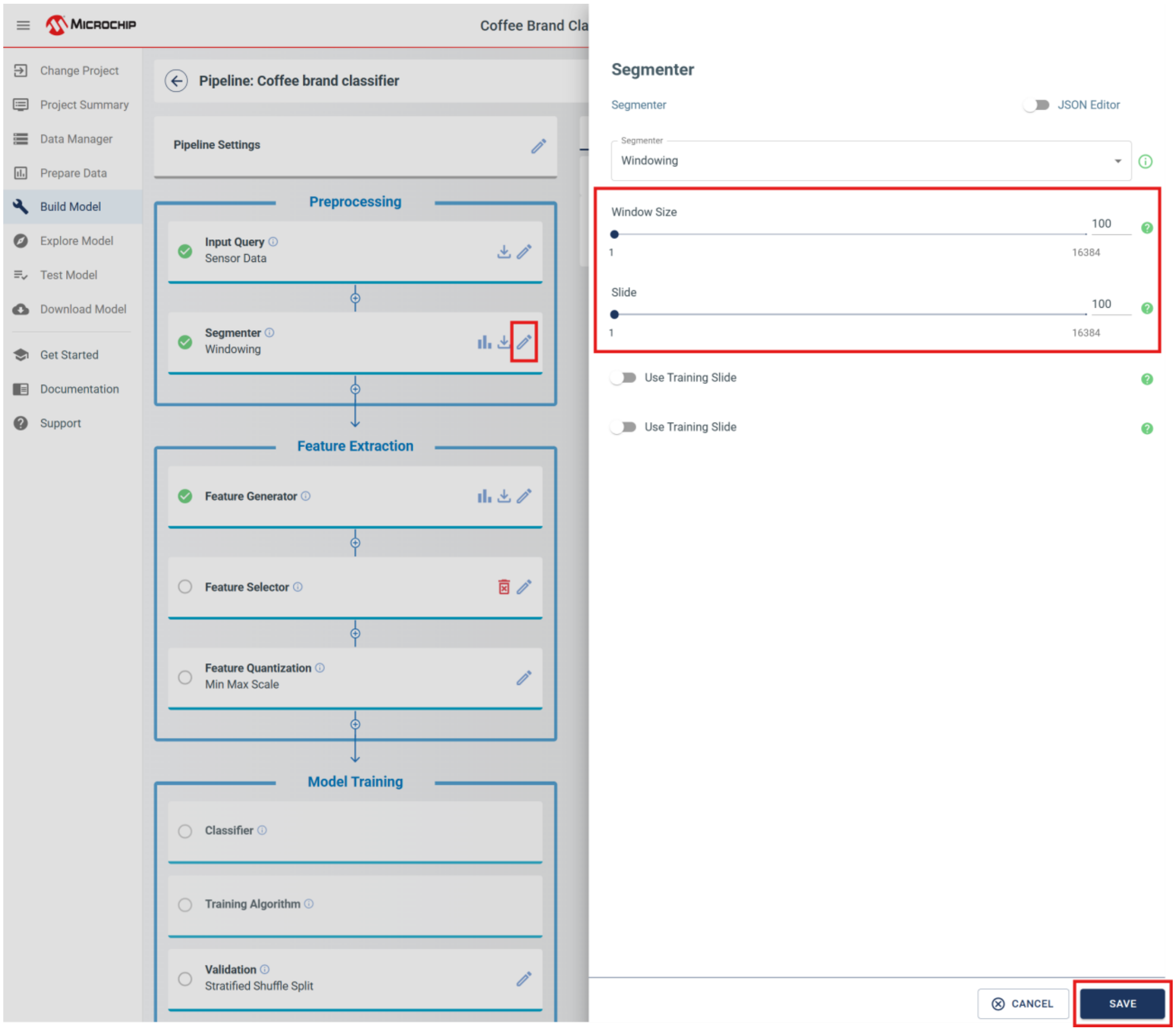
In this step, choose the desired query for constructing the pipeline. In the provided example, "Sensor Data" is chosen as the query and the pipeline has been named "Coffee Brand Classifier". Upon selecting the query, ensure to click the "CREATE PIPELINE" button to preserve your selection.

The pipeline will be created showing parameters that will be used while building the Model. Click **NEXT**.

To begin, access the pipeline settings by selecting the edit icon adjacent to it. By default, the system employs AutoML mode. However, users have the option to utilize custom training mode if necessary. For this demonstration, we will proceed with AutoML. Adjustments will be made solely to the search settings, specifically the number of iterations and population size, as indicated below.

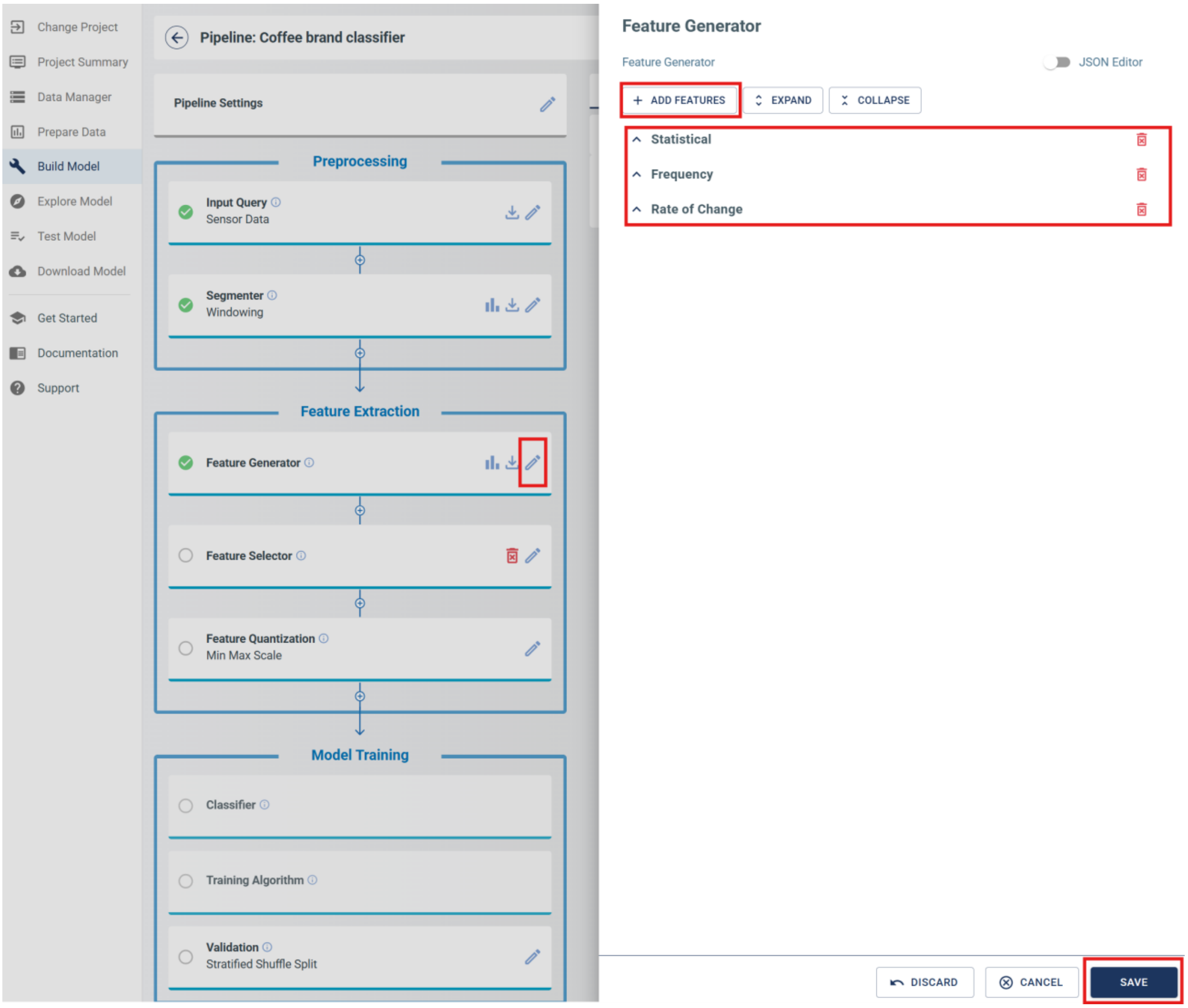


Access the Segmenter module, which performs segmentation of the data. Set the parameters as depicted in the figure below to ensure the appropriate segmentation of the data. After setting the parameters, click the **SAVE** button to save the changes made.



Then, open the Feature Generator module. Select the features used to extract the information from the input sensor data. In this form, you can specify the types of feature generators you would like to use, the inputs into the feature generators and any parameters the feature generators take. Select the features that are shown below and click **NEXT** after selecting the required features.

Figure 1-40. Selecting the Features for model building



Upon configuring all settings, activate the optimization process by selecting the "Run Pipeline" button located at the top.

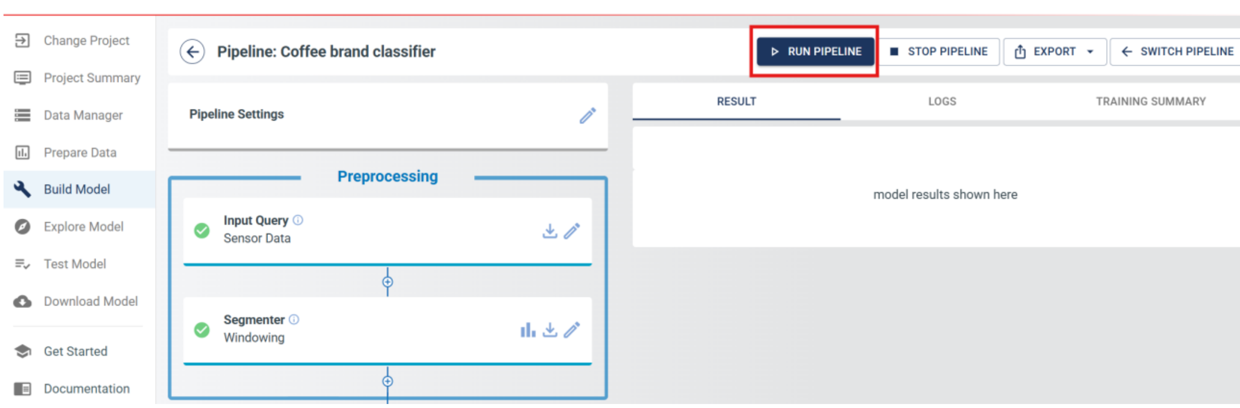
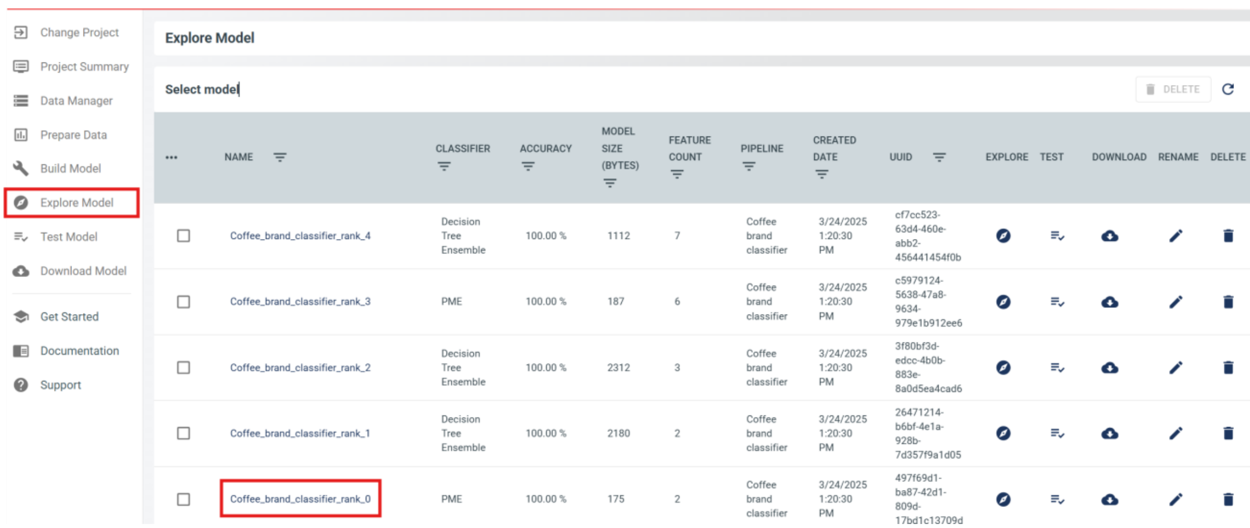


Figure 1-46. Starting the AutoML optimization process.

The AutoML techniques will automatically select the best features and machine learning algorithm for the gesture classification task based on the input data you provided. Note that this optimization process typically takes several minutes to complete.

After completing the process of Model Optimization, proceed to the Explore Model view. Within this view, the top five models are ranked and displayed based on their ability to produce optimal outcomes. To access additional information regarding each individual model, click the open model icon that is situated on the left-hand side of its respective name.

Figure 1-47. Exploring the model.



Users can access different output results via the options available at the top. These results include confusion matrices, graphs, and flowcharts, which can be explored for detailed analysis through modules such as confusion matrix, feature embedding, feature visualization, and different summaries.

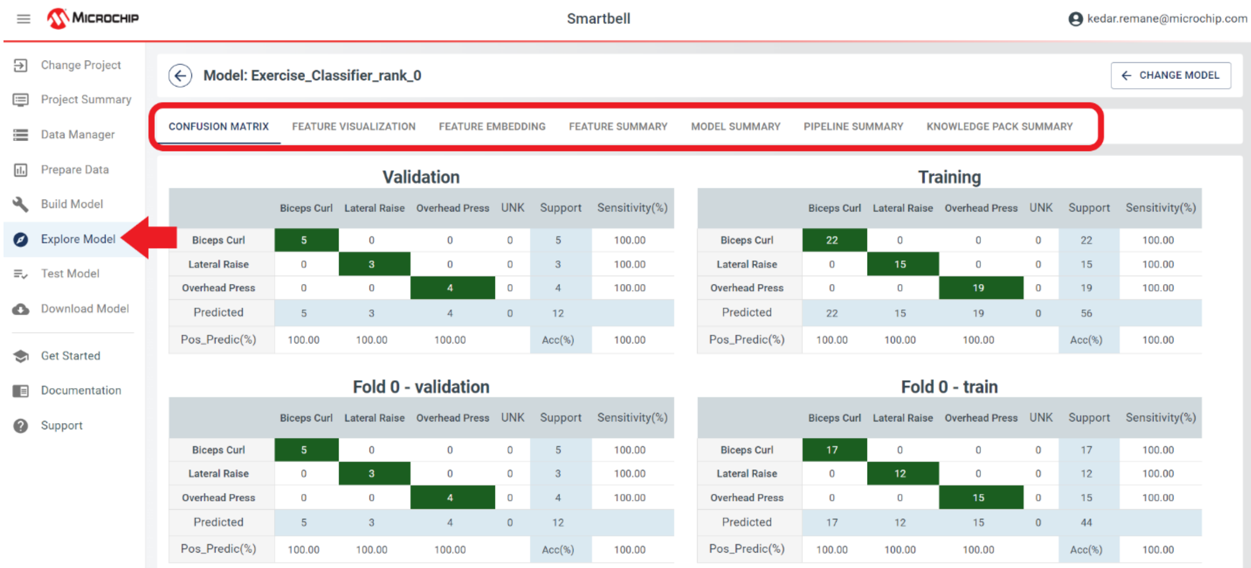


Figure 1-48. Different modules under explore model.

After confirming satisfactory model results, users can navigate to the Test Model view. To filter data and isolate test samples, follow these steps:

1. Choose the desired Session.
2. Click the upside-down triangle icon in the Fold column.
3. Select "Test" from the dropdown menu to filter and display only test samples.

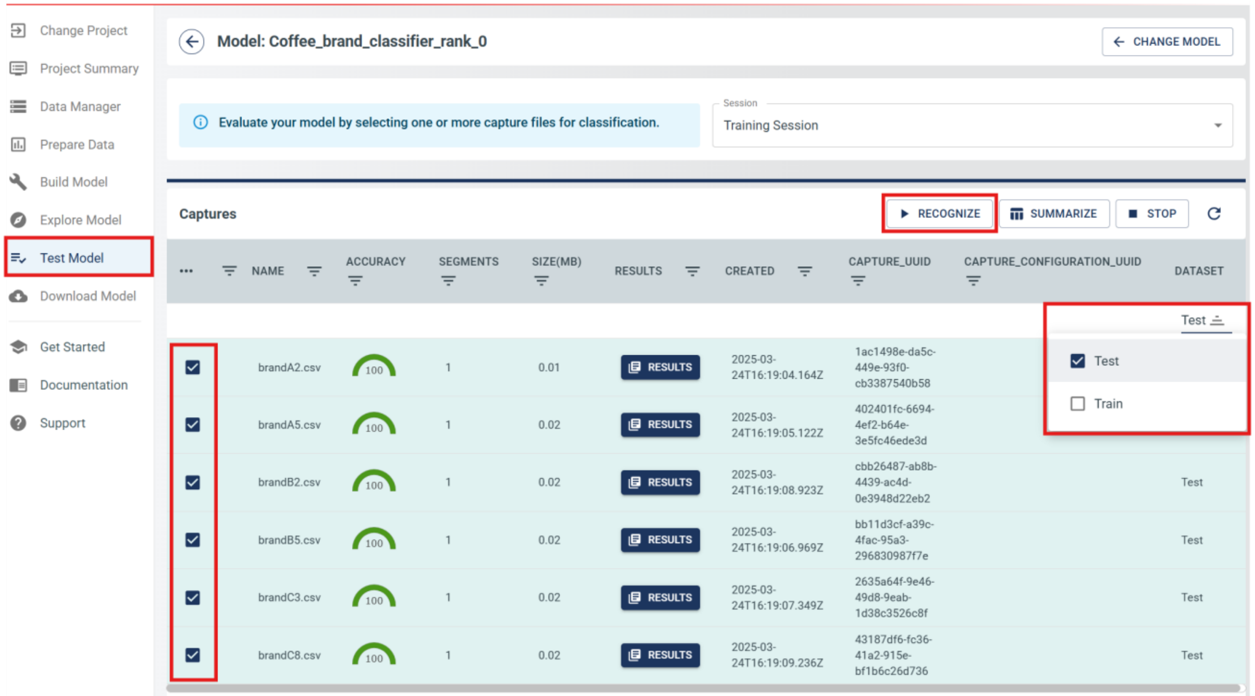


Figure 1-49. Testing the Model.

To compute accuracy and generate a confusion matrix for the test samples, follow these steps:

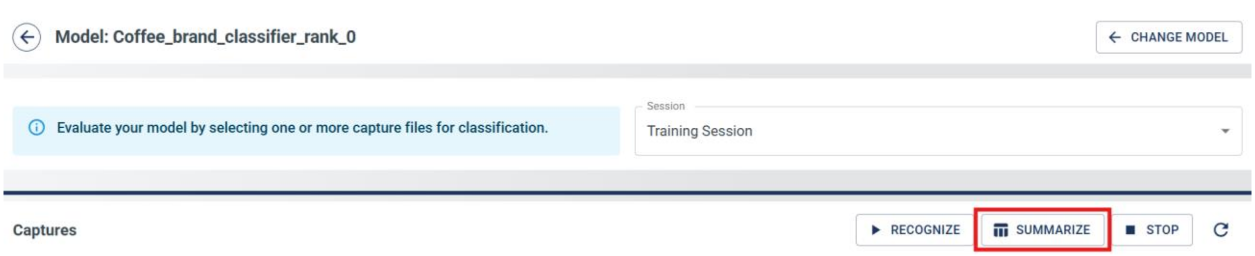
1. Click the ellipsis (...) icon in the left-most column of the table.

2. Select "Select All" from the dropdown menu to include all test samples.

3. Click the "Recognize" button to compute accuracy metrics.

4. Once accuracy is computed, click "Summarize" to generate the confusion matrix.

5. Note that this process may take a few minutes. Upon completion, a table summarizing classification results will be presented.



SUMMARY

	Coffee Brand A	Coffee Brand B	Coffee Brand C	Unknown	UNK	Ground Truth	Support	Sensitivity(%)
Coffee Brand A	17	0	0	0		2	17	100.00
Coffee Brand B	0	28	0	0		2	28	100.00
Coffee Brand C	0	0	25	0		2	25	100.00
Unknown	0	0	0	0		0	0	
Predicted	17	28	25	0		6	70	
Pos_Predic(%)	100.00	100.00	100.00				Acc(%)	100.00

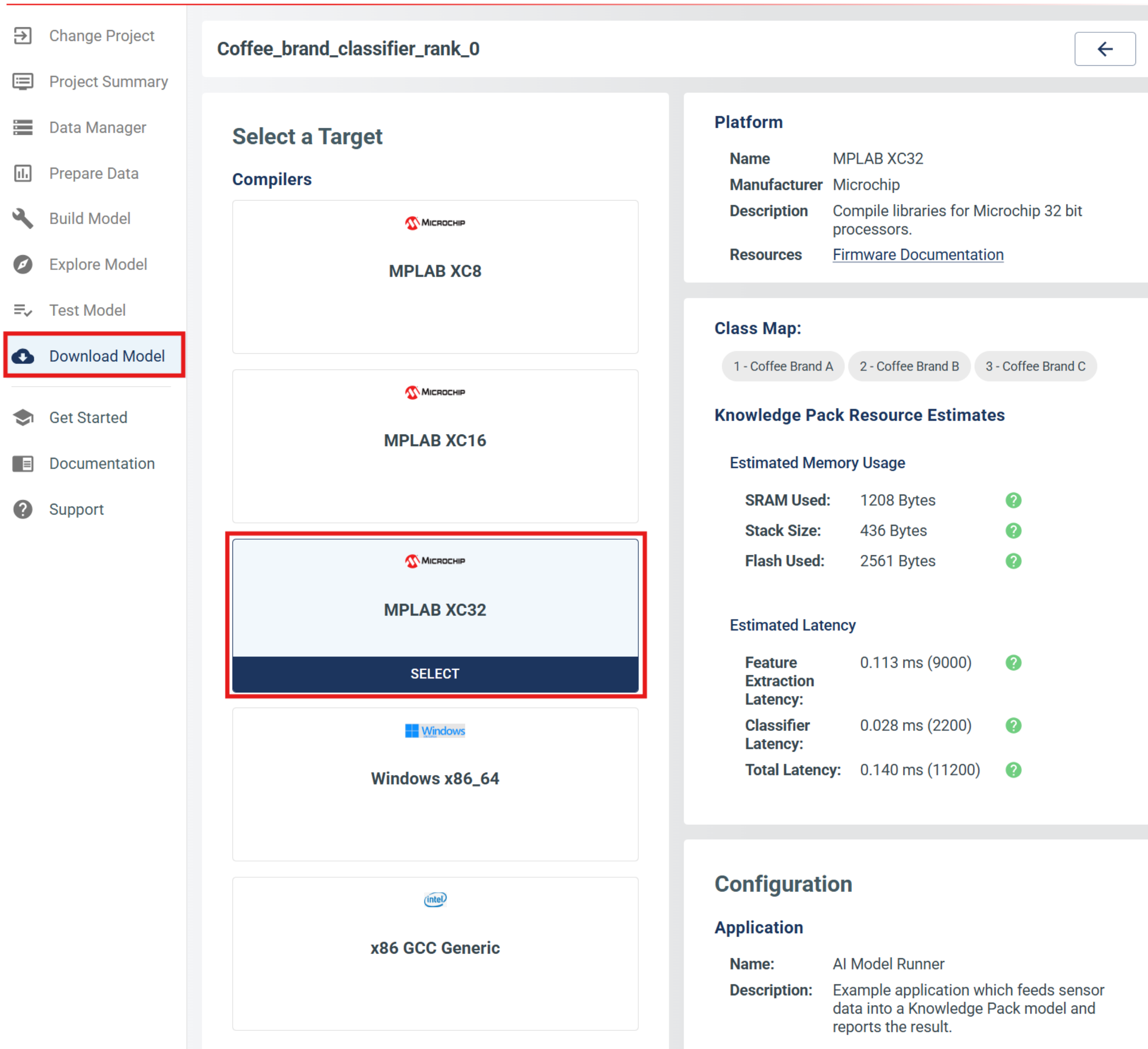
Figure 1-50. Confusion matrix summary.

To deploy your model, follow these steps:

1. Navigate to the Download Model tab.
2. Fill out the Knowledge Pack settings using the Pipeline, Model and Data Source you created in the previous steps.
3. Select the library output format.
4. Click the **Download** button.

Figure 1-51. Downloading the model.

By completing these steps, you will be able to deploy your model and make it available for use. It is important to ensure that all necessary settings are correctly entered before proceeding with the download.



Change Project
Project Summary
Data Manager
Prepare Data
Build Model
Explore Model
Test Model
Download Model
Get Started
Documentation
Support
Coffee_brand_classifier_rank_0
Select a Target
Compilers
MICROCHIP
MPLAB XC8
MICROCHIP
MPLAB XC16
MICROCHIP
MPLAB XC32
SELECT
Windows
Windows x86_64
intel
x86 GCC Generic
Platform
Name MPLAB XC32
Manufacturer Microchip
Description Compile libraries for Microchip 32 bit processors.
Resources Firmware Documentation
Class Map:
1 - Coffee Brand A
2 - Coffee Brand B
3 - Coffee Brand C
Knowledge Pack Resource Estimates
Estimated Memory Usage
SRAM Used: 1208 Bytes
Stack Size: 436 Bytes
Flash Used: 2561 Bytes
Estimated Latency
Feature Extraction Latency: 0.113 ms (9000)
Classifier Latency: 0.028 ms (2200)
Total Latency: 0.140 ms (11200)
Configuration
Application
Name: AI Model Runner
Description: Example application which feeds sensor data into a Knowledge Pack model and reports the result.

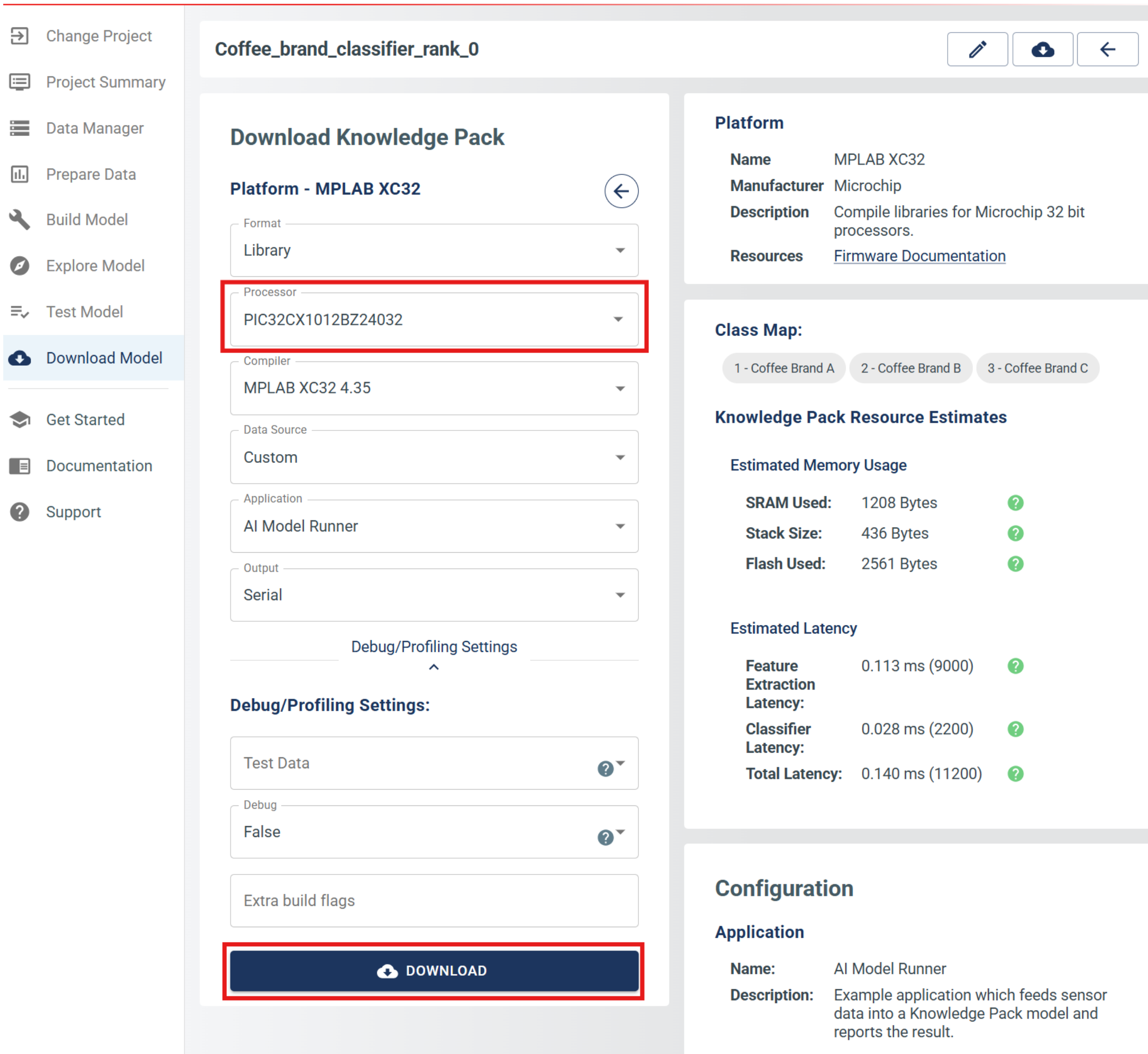


Figure 1-52. Selecting the parameters for downloading the model.

Knowledge Pack Integration

Integrate the Knowledge Pack APIs into your embedded device firmware by following the Knowledge Pack Integration.

Coffee Bean Classifier Firmware Overview

For a description of the demo firmware included with this project including operation, usage and benchmarks, see the README file in the GitHub Repository.

Final Remarks

Upon completing this guide, you will possess a foundational knowledge of developing a Coffee Bean Classifier application utilizing MPLAB ML Development Suite in conjunction with the PIC32CX-BZ2 / WBZ451 and BMI160 6DOF IMU sensor ML evaluation kit. For additional insights into machine learning capabilities at Microchip, please refer to the resources available on Artificial Learning and Machine Learning.