Hands on Lab: Getting Started with Open Source Datasets and Deep Learning Models

In this lab, you will explore how to access and utilize open source datasets for machine learning projects. You will also get hands-on experience with pre-trained deep learning models to accelerate your development process.

Objective of Exercise 1:

- Explore the open data sets.

Objective of Exercise 2:

- Find ready-to-use deep learning models on the Model Asset Exchange.
- Explore the deep learning model trained to detect objects in an image.

It will take you approximately 15 minutes to complete the lab. Only a web browser is required to complete the tasks.

Exercise 1: Explore open source datasets

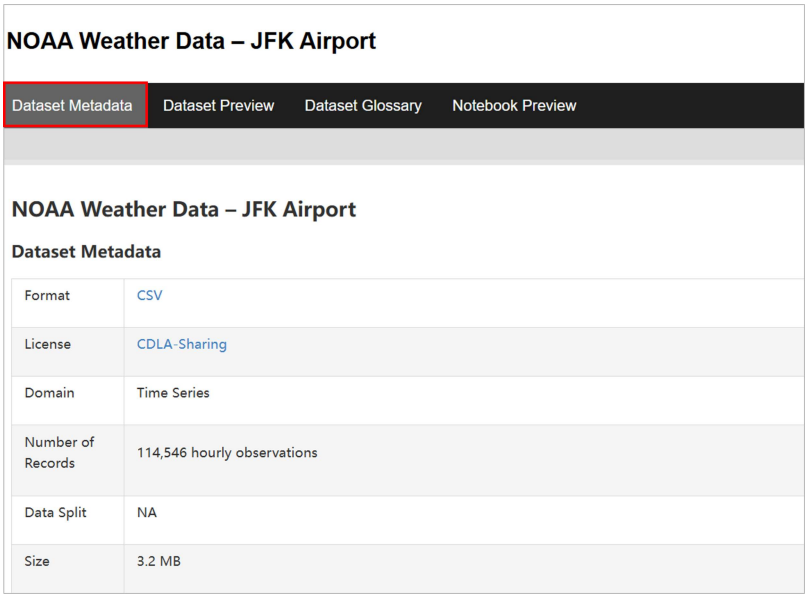
Most datasets are complemented by Python notebooks that you can use to explore, pre-process, and analyze the data. Here we will be exploring a public weather dataset taken from kaggle LINK

Step-by-Step Instructions

1. Click HERE to view the dataset.
2. Once the page loads, click the **Dataset Preview** tab at the top of the page.

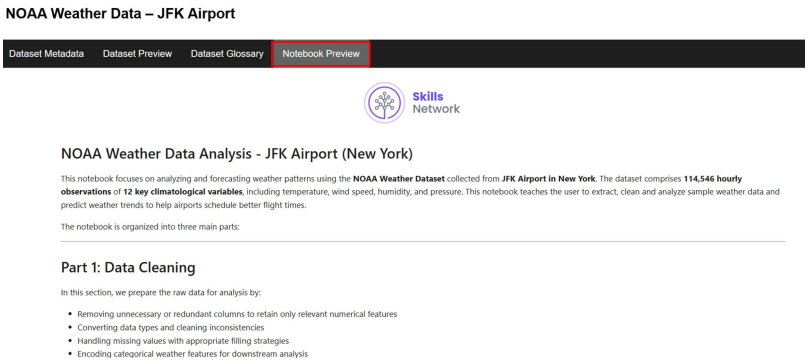
This will allow you to:

- View a sample of the data directly in your browser.
- Understand the structure and fields of the dataset.
- Explore metadata and glossary terms associated with the dataset.



3. Next, click the **Notebook Preview** tab.

- This opens a sample Python notebook associated with the dataset.
- The notebook walks through:
  - Cleaning the weather data.
  - Performing exploratory analysis.
  - Building predictive models to help airports better manage flight schedules.



This concludes Exercise 1 of this lab, which introduced you to the open source dataset. You may proceed to Exercise 2.

4. **[Optional]** If you are comfortable using Jupyter Notebooks, you can download and run the notebook locally by clicking HERE

Exercise 2 - Explore deep learning models

The Model Asset Exchange is a curated repository of Open Source deep learning models for a variety of domains, such as text, image, audio, and video processing.

For more details, please visit - https://github.com/CODAIT/max-central-repo webpage.

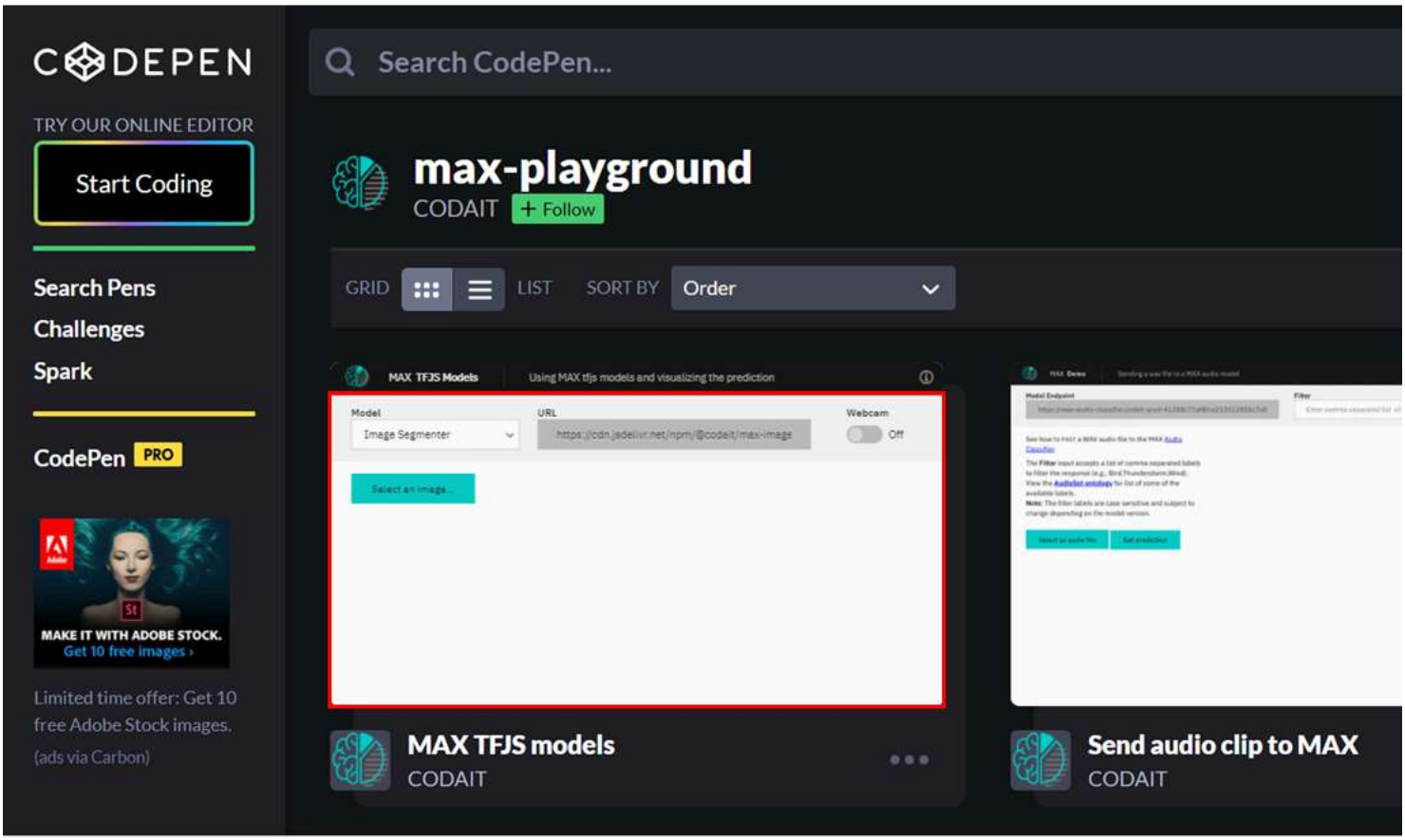
The curated list includes deployable models, which you can run as a microservice locally or in the cloud on Docker or Kubernetes, and trainable models where you can use your own data to train the models. Some of the models are already built for you to test. Let's test one of the models.

In this exercise, explore the **Object Detector** model hosted on CodePen platform. This model recognizes the objects present in an image. The model consists of a deep convolutional net base model for image feature extraction, together with additional convolutional layers specialized for the task of object detection, trained on the COCO data set. The input to the model is an image, and the output are extracted objects from the image, appropriately labeled.

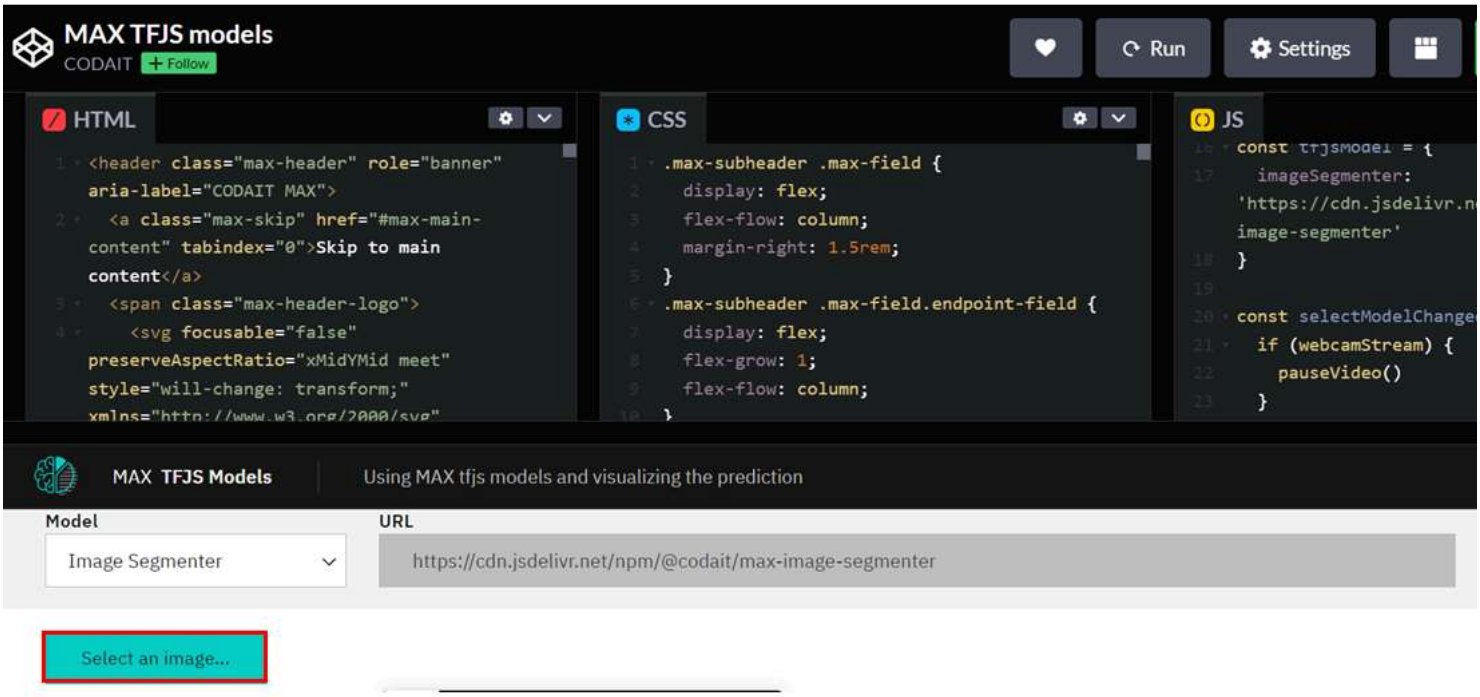
CodePen is a social development environment. At its heart, it allows to write code in the browser and see the results of it as you build. It is a useful and liberating online code editor for developers of any skill and is particularly empowering for people learning to code.

1. Navigate to **CodePen** webpage.

2. Select **MAX TFJS models** as shown in the screenshot below. Here the **Image Segmenter**, divides an image into regions or categories that correspond to different objects or parts of objects. Every pixel in an image is allocated to one of a number of these categories.



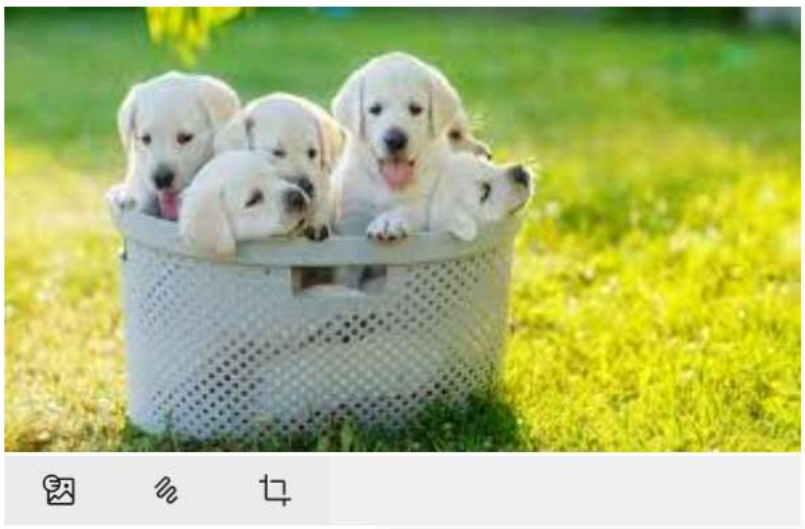
3. Click on **Select Image** and upload an image. You may choose images with a person, dog, cat, truck, car, and so on, which are labels the model has been trained on.



▶ Click here for all the labels the model is trained on

4. Click the icon **Extract prediction** as shown below:

You can see the output of the prediction on the basis of the uploaded image.



Here the background and the dog image are separated, showing two different parts of the image.

You can also try the webcam option, which will show the real-time prediction by the toggle-on webcam option.

This concludes Exercise 2 of this lab, which introduced the Model Asset Exchange (MAX).

You can also watch a demo of the object detector model here.

Author(s)

Joseph Santarcangelo

Other Contributor(s)

Lavanya