Edge_Intelligence_Lab_MACSE604_

25MML0002

Adithyash BC

IN LAB 2

Part 1

To create a lightweight model(CNN/ANN) and train it on MNIST or other image datasets. Save the model using pickle and note the weight of the model

The model chosen was CNN and the dataset chosen was mnist for the following exercise

Code:

```
import tensorflow as tf
from tensorflow.keras.datasets import mnist
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Conv2D, MaxPooling2D, Flatten, Dense
from tensorflow.keras.utils import to_categorical
import pickle
import numpy as np
```

```
(X_train, y_train), (X_test, y_test) = mnist.load_data()

print(f"X_train shape: {X_train.shape}")
print(f"X_test shape: {X_test.shape}")
```

```
X_train shape: (60000, 28, 28)
X_test shape: (10000, 28, 28)
```

```python
X_train = X_train.reshape(X_train.shape[0], 28, 28, 1).astype('float32')
X_test = X_test.reshape(X_test.shape[0], 28, 28, 1).astype('float32')
X_train = X_train / 255.0
X_test = X_test / 255.0

# One-hot encode
y_train = to_categorical(y_train, num_classes=10)
y_test = to_categorical(y_test, num_classes=10)

print(f"X_train reshaped and normalized shape: {X_train.shape}")
print(f"y_train one-hot encoded shape: {y_train.shape}")
```

```
X_train reshaped and normalized shape: (60000, 28, 28, 1)
y_train one-hot encoded shape: (60000, 10)
```

```python
# Build the CNN model
model = Sequential([
    Conv2D(32, (3, 3), activation='relu', input_shape=(28, 28, 1)),
    MaxPooling2D((2, 2)),
    Conv2D(64, (3, 3), activation='relu'),
    MaxPooling2D((2, 2)),
    Flatten(),
    Dense(128, activation='relu'),
    Dense(10, activation='softmax')
])
```

```
/usr/local/lib/python3.12/dist-packages/keras/src/layers/convolutional/base_conv.py:113: UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When using Sequential models, prefer using an `Input(shape)` object as the first layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
```

```
]:
```

```python
model.compile(optimizer='adam',
              loss='categorical_crossentropy',
              metrics=['accuracy'])

model.summary()
```

Model: "sequential_1"

Layer (type)	Output Shape	Param #
conv2d_2 (Conv2D)	(None, 26, 26, 32)	320
max_pooling2d_2 (MaxPooling2D)	(None, 13, 13, 32)	0
conv2d_3 (Conv2D)	(None, 11, 11, 64)	18,496
max_pooling2d_3 (MaxPooling2D)	(None, 5, 5, 64)	0
flatten_1 (Flatten)	(None, 1600)	0
dense_2 (Dense)	(None, 128)	204,928
dense_3 (Dense)	(None, 10)	1,290

Total params: 225,034 (879.04 KB)

Trainable params: 225,034 (879.04 KB)

Non-trainable params: 0 (0.00 B)

```python
#training
history = model.fit(X_train, y_train, epochs=5, batch_size=32, validation_split=0.1)
# Evaluate
loss, accuracy = model.evaluate(X_test, y_test, verbose=0)
print(f"Test Loss: {loss:.4f}")
print(f"Test Accuracy: {accuracy:.4f}")
```

```
Epoch 1/5
1688/1688 ━━━━━━━━━━━━━━━━━━━━ 54s 31ms/step - accuracy: 0.9022 - loss: 0.3230 - val_accuracy: 0.9847 - val_loss: 0.0570
Epoch 2/5
1688/1688 ━━━━━━━━━━━━━━━━━━━━ 51s 30ms/step - accuracy: 0.9852 - loss: 0.0499 - val_accuracy: 0.9858 - val_loss: 0.0502
Epoch 3/5
1688/1688 ━━━━━━━━━━━━━━━━━━━━ 82s 30ms/step - accuracy: 0.9906 - loss: 0.0313 - val_accuracy: 0.9903 - val_loss: 0.0361
Epoch 4/5
1688/1688 ━━━━━━━━━━━━━━━━━━━━ 51s 30ms/step - accuracy: 0.9929 - loss: 0.0226 - val_accuracy: 0.9900 - val_loss: 0.0366
Epoch 5/5
1688/1688 ━━━━━━━━━━━━━━━━━━━━ 50s 29ms/step - accuracy: 0.9950 - loss: 0.0157 - val_accuracy: 0.9895 - val_loss: 0.0368
Test Loss: 0.0360
Test Accuracy: 0.9890
```

```python
model_filename = 'mnist_cnn.pkl'
with open(model_filename, 'wb') as file:
    pickle.dump(model, file)

print(f"Model saved to {model_filename} using pickle.")
```

```
Model saved to mnist_cnn.pkl using pickle.
```

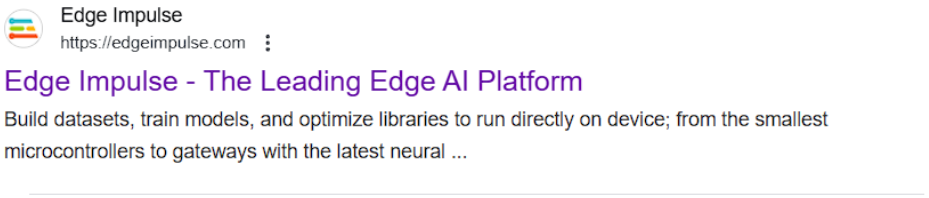
The weight/ size of the model can be found by navigating to the desired model via the Jupyter notebook interface

The size of the model is 2.74MB

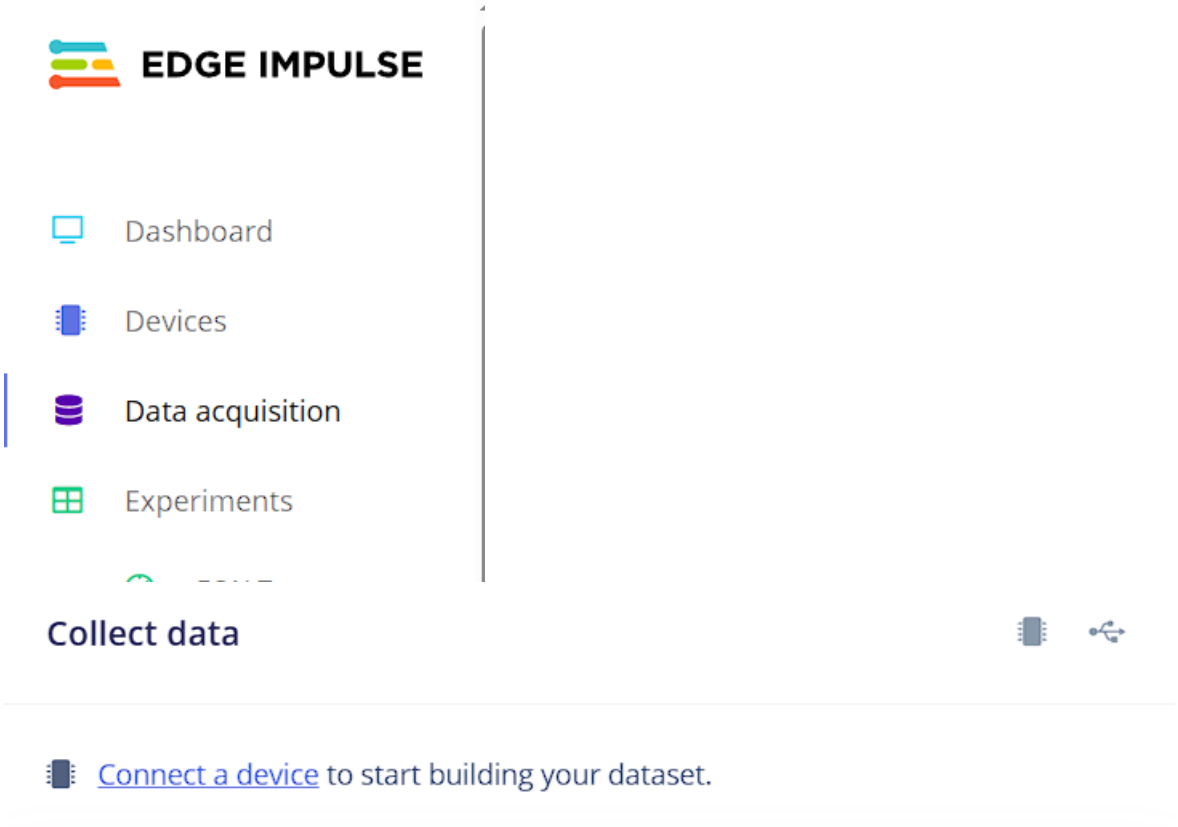
Part 2

Create an edge impulse account and create an image dataset by capturing images via your cellphone

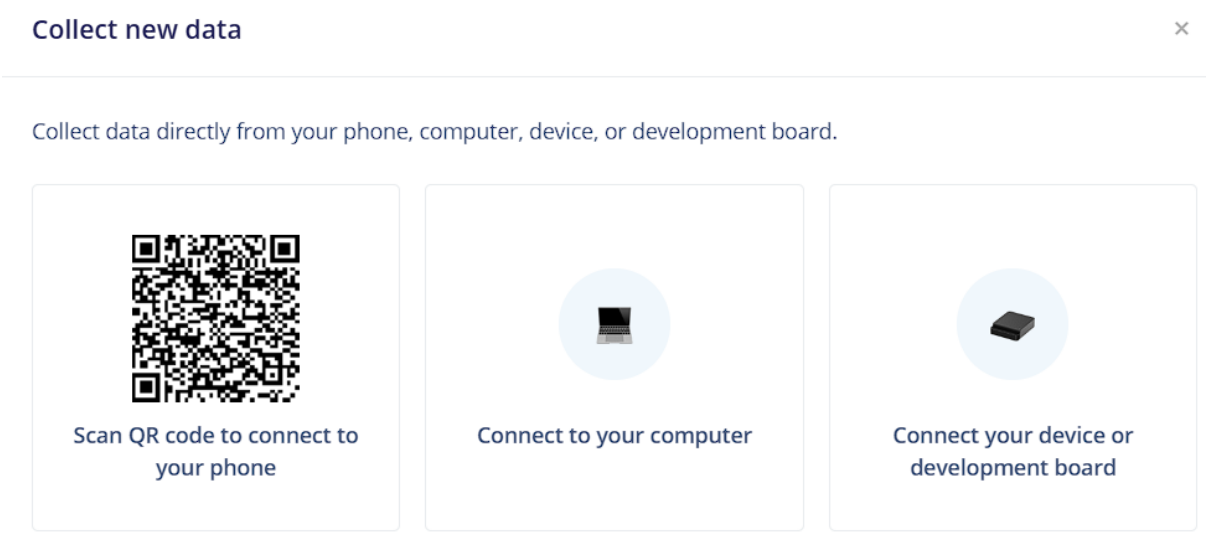
We must first navigate to edgeimpulse.com and sign up for an account



After filling in the required details we navigate to the data acquisition tab and click on “Connect a Device” which will prompt us with three options.



We pick the QR code option/ Mobile and scan the displayed QR code to connect our mobile device to the platform via an API



After that we may freely take photos of objects we want in our dataset and label them. The final view should look a little something like this