CAPSTONE PROJECT REPORT

Nguyen Phung Hung

Inventory Monitoring at Distribution Centers Capstone Project

1. Domain Background:

Distribution centers often use robots to move objects as a part of their operations. Objects are carried in bins which can contain multiple objects. In this project, you will have to build a model that can count the number of objects in each bin. A system like this can be used to track inventory and make sure that delivery consignments have the correct number of items.This is where robots come in to help in Inventory Monitoring. They can be trained with Machine Learning Models, to perform tasks like Object Detection, Outlier & Anomaly Detection and much more. Once trained, these models are scalable, and can be deployed at a low cost for usage in actual warehouses and distribution centres on industry level robots.

To build this project i will use AWS SageMaker and good machine learning engineering practices to fetch data from a database, preprocess it, and then train a machine learning model. This project will serve as a demonstration of end-to-end machine learning engineering skills that you have learned as a part of this nanodegree.

2. Problem Statement:

Distribution centres often have robots which carry objects. These objects are present in bins, each bin contain 1-5 objects. The problem in this project is to count the number of items in the bin. We will create a model, which can take in a picture of a bin, and accurately return the number of objects present in that, we could solve & thus fully automate one crucial step in the Inventory Management process!

3. Solution Statement:

The identified problem can be solved using Image Classification . By used data to train the machine learning model, a model can be produced.

Components:

- Dataset Image Amazon Bin

- Deep learning Model

- Amazon Web Service

Pllatform:

1.. Sage Maker Studio – to train, tune and deploy the model.

2. S3 – Storage Bucket

4. Benchmark Model:

The result of the Amazon Bin Image Dataset from this Repository is considered as a Benchmark. We has achieved the accuracy of 40 percent..

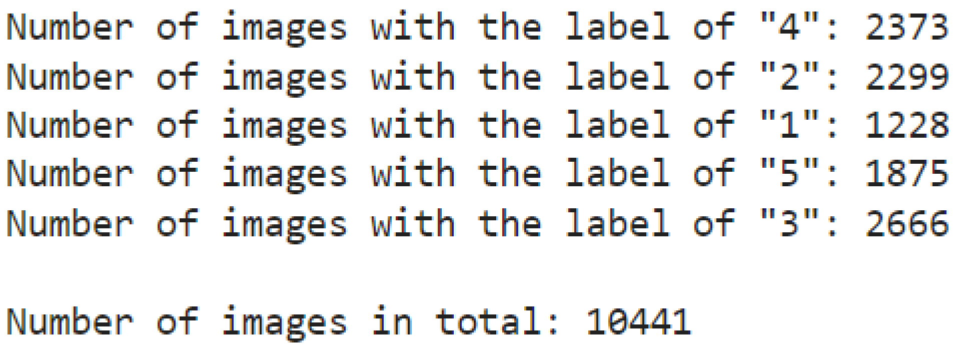
5. Evaluation Metrics:

There is a classification problem, the overall accuracy of the classification can be used to evaluate the performance of the trained model.

6. Dataset or the Image Source:

The Amazon Bin Image Dataset is to be used to train the model.

The dataset used in this problem is the Amazon Bin Image Dataset. This dataset have 500,000 images of bins and the data have one or more objects present in it. One of each image contains information about the image, like the number of objects it has, the dimensions and type of objects. For our problem statement, we only need the total count of objects in the imageThese are some typical images in the dataset.



```
Number of images with the label of "4": 2373
Number of images with the label of "2": 2299
Number of images with the label of "1": 1228
Number of images with the label of "5": 1875
Number of images with the label of "3": 2666

Number of images in total: 10441
```

The, the data loaders are created at a batch size of 32 for training, testing and validation.

7. Algorithm:

1. The images is used to train the model, that could learn the parameters to perform classification based on the number of items in the bin.

2. Deep Learning Framework – PyTorch

3. A corresponding Sage Maker instance will be created, and data will be fed from the S3 bucket.

4. The model is also tuned to find out the best hyper-parameters as a model improvement method.

5. A pre-trained model ResNet18 is used in this project.

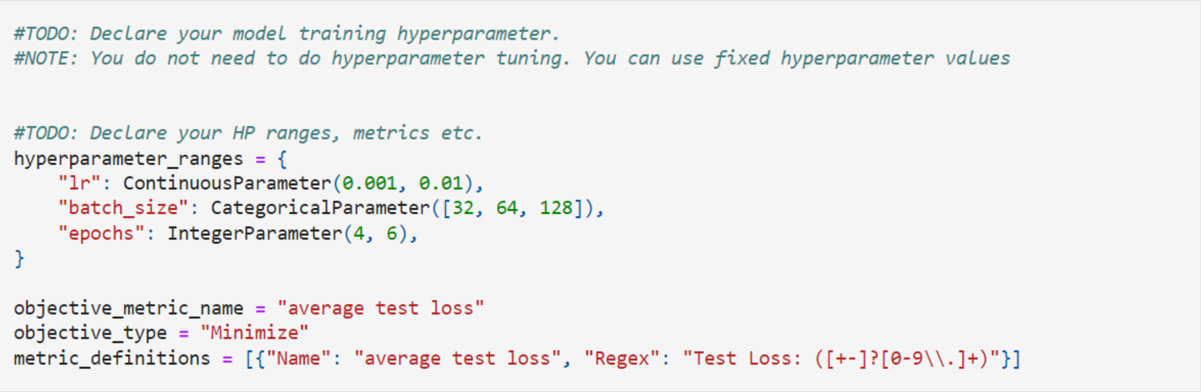
8. Hyper-parameter Tunning:

The following hyper-parameters are identified for tunning.

- Learning Rate

- Batch Size

- Epoch The Learning rate for the Adam optimizer is identified as one of the hyper-parameters which could help improve the training process rather than the pre-defined default.



```
#TODO: Declare your model training hyperparameter.
#NOTE: You do not need to do hyperparameter tuning. You can use fixed hyperparameter values


#TODO: Declare your HP ranges, metrics etc.
hyperparameter_ranges = {
    "lr": ContinuousParameter(0.001, 0.01),
    "batch_size": CategoricalParameter([32, 64, 128]),
    "epochs": IntegerParameter(4, 6),
}

objective_metric_name = "average test loss"
objective_type = "Minimize"
metric_definitions = [{"Name": "average test loss", "Regex": "Test Loss: ([+-]?[0-9\\.]+)"}]
```

9. Model Evaluation:

Finnaly we can used the model is deployed like an endpoint to predict the Image to see the result and so that we can evaluation the accuracy

```
In [68]:
# TODO: Run an prediction on the endpoint
from PIL import Image
import io
import os
import numpy as np

import requests
request_dict={ "url": "https://aft-vbi-pds.s3.amazonaws.com/bin-images/777.jpg"}

img_bytes = requests.get(request_dict['url']).content
# type(img_bytes)

response=predictor.predict(img_bytes, initial_args={"ContentType": "image/jpeg"})
import numpy as np
np.argmax(response)
```

```
Out[68]: 2
```