Udacity AWS Machine Learning Engineer Nanodegree

Capstone Proposal

INVENTORY MONITORING AT DISTRIBUTION CENTERS

1. DOMAIN BACKGROUND

Robots are used in distribution centers for moving objects as a part of their operations and those objects which are carried in bins can consist of a lot of objects. In this project, we are addressing the problem of inventory monitoring at distribution centers by developing a Deep Learning (DL) model that is capble of automatically counting the number of objects in bins and, thus, providing accurate inventory tracking information to decision makers for optimizing their operations.

To develop this Capstone Project, I am going to use AWS SageMaker to develop an end-to-end solution from obtaining a public dataset from its database to training and deploying a DL model using the dataset.

2. PROBLEM STATEMENT

The main objective of this Capstone Project is to count the number of objects in bins via images. In other words, this is a Computer Vision problem to detect multiple objects in images, and then count the number of objects.

Besides, to complete this Project, we will be using the Amazon Bin Image Dataset [1]. So we need to perform the data ingestion, data processing, model training and deployment based on this public dataset to solve the problem.

3. SOLUTION STATEMENT

To address the problem in this Capstone Project, we will firstly fetch the Amazon Bin Image Dataset from its database, and store the dataset using Amazon Simple Storage Service (Amazon S3). The dataset will be preprocessed and split into training, validation and test sets with a predefined split ratio, which will be ready for training the DL model. In this Project, Amazon SageMaker will be used to build the DL workflow.

Residual Networks, or ResNets [2], are Convolutional Neural Networks (CNNs) that learn residual functions with reference to the layer inputs, instead of learning unreferenced functions. Instead of hoping each few stacked layers directly fit a desired underlying mapping, residual nets let these layers fit a residual mapping. They stack residual blocks ontop of each other to form network. Due to its high performance in image classification, in this Capstone Project, we will employ and fine tune a pretrained ResNet-50 with 50 layers deep (Fig. 1) for detecting objects in the images.

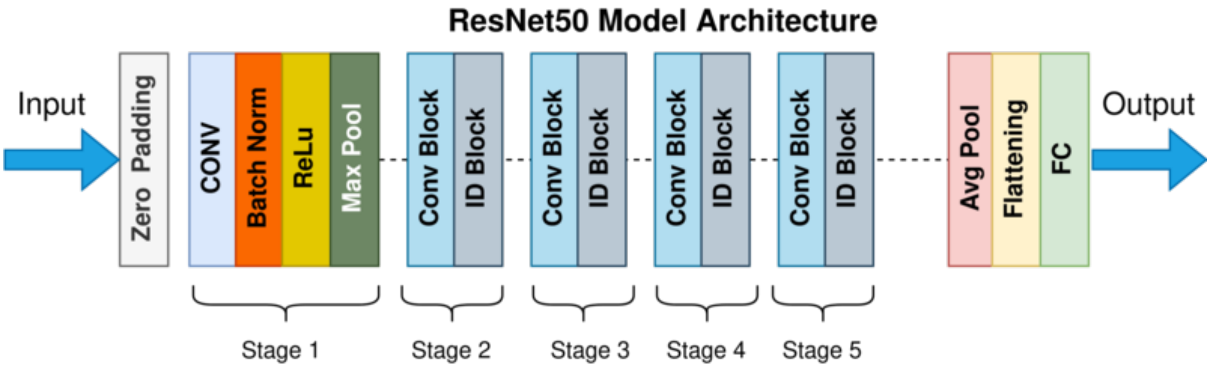

Fig. 1. ResNet-50 Architecture.

4. DATASETS AND INPUTS

As mentioned above, the Amazon Image Bin Dataset will be used in this Capstone Project. This dataset contains more than 500K images (in JPEG format) and their corresponding labels (in JSON format) collected from bins of a pod in an operating Amazon Fulfillment Center. The data can be accessed by using the below command line with no AWS account required:

`aws s3 ls --no-sign-request s3://aft-vbi-pds/`

Images are located in the `bin-images` directory while the corresponding labels are located in the `metadata` directory. For example, the metadata for the image at https://aft-vbi-pds.s3.amazonaws.com/bin-images/523.jpg (Fig. 2) is found at https://aft-vbi-pds.s3.amazonaws.com/metadata/523.json


Fig. 2. An image sample from Amazon Image Bin Dataset.

In particular, each bin image may show only one or more types of products. In this Project, we have utilized a total of 10441 images with the class distributions as below:

- 1 item in bin: 1228 images
- 2 item in bin: 2299 images
- 3 item in bin: 2666 images
- 4 item in bin: 2373 images
- 5 item in bin: 1875 images

More details on Amazon Image Bin Dataset can be found: https://github.com/awslabs/open-data-docs/tree/main/docs/aft-vbi-pds.

5. BENCHMARK MODEL

To benchmark the proposed approach, the input dataset will be split into three subsets: training, validation and test sets with a predefined split ratio: 0.6, 0.2 and 0.2. In other words, 6265 images are used for fine-tuning the pretrained ResNet-50 model, 2088 images are used for validating the fine-tuning process while the remaining 2088 images are used for testing the performance of the trained model.

In this Capstone project, we also employ SageMaker to perform hyperparameter tuning, so that the model can get the optimal hyperparameters during the fine-tuning process. SageMaker Debugger and Profiler are also applied to detect any anomaly while training the model.

6. EVALUATION METRICS

In this Project, the Cross Entropy function is used as the loss function for training and testing the model, defined as below:

$$H(X) = -\Sigma_i P(X = i) log_2 P(X = i)$$

7. PROJECT DESIGN

To summarize, we will develop the ML workflow in the Capstone Project with 3 main steps as below (Fig. 3):

1. **Data Preparations:**
   - Amazon Bin Image Dataset is fetched from the database and stored into Amazon S3.
   - The dataset is split into training, validation and test sets, which are used to fine tune the model in the next step.
2. **Training:**
   - Amazon SageMaker is used to fine tune a pretrained ResNet-50 CNN model.
   - SageMaker Hyperparameter Tuner is used to optimize the model hyperparameters during the fine tuning process.
   - SageMaker Debugger and Profiler are also applied to detect any anomaly while training the model.
3. **Deploy:**
   - The fine tuned model with the optimal hyperparameters is then deployed to a SageMaker Endpoint for further testing.

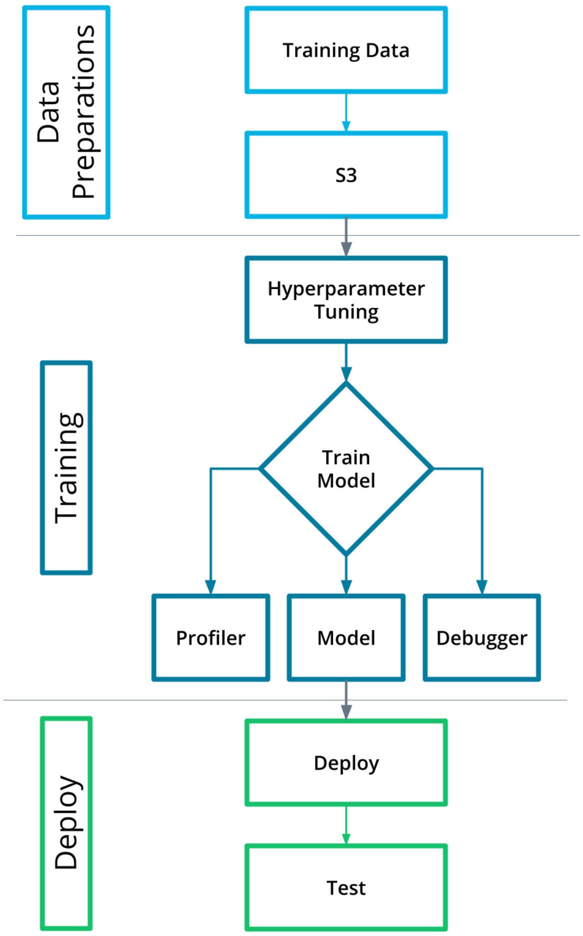

Fig. 3. ML workflow of the Capstone Project.

REFERENCES

[1] Amazon Bin Image Dataset: https://registry.opendata.aws/amazon-bin-imagery/.

[2] He, K., Zhang, X., Ren, S., and Sun, J., "Deep Residual Learning for Image Recognition", <i>arXiv e-prints</i>, 2015. doi:10.48550/arXiv.1512.03385.