

Amazon Bin Image Dataset

Definition

Project Overview

Amazon Fulfillment Centers use robotic and computer vision technology to deliver millions of goods to customers in over 100 countries around the world. The Amazon Bin Image Dataset comprises photos and metadata from bins in an Amazon Fulfillment Center that is currently operational. The bin photos in this dataset were collected while robot units were transporting pods as part of typical Amazon Fulfillment Center operations.

Problem Statement

Each picture has a metadata file that contains information about the image such as the number of objects, the dimensions, and the kind of item. We will attempt to categorize the number of items in each bucket for this challenge. We will use a model to accomplish the categorization. We can utilize a pre-trained convolutional neural network or our own neural network design, and SageMaker to train our model.

Metrics

The Evaluating Metrics I used for this project are the accuracy and the RMSE and compared them to another project accuracy and RMSE.

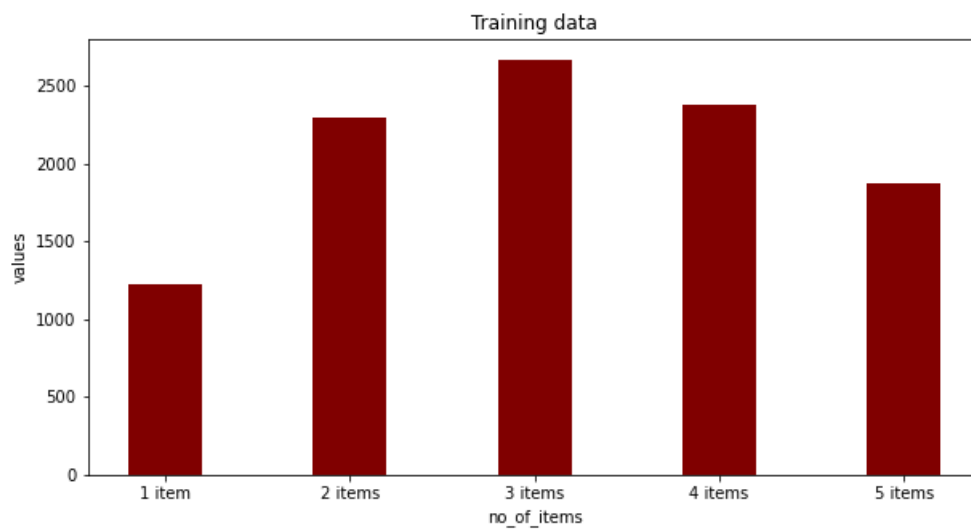
Analysis

Data Exploration

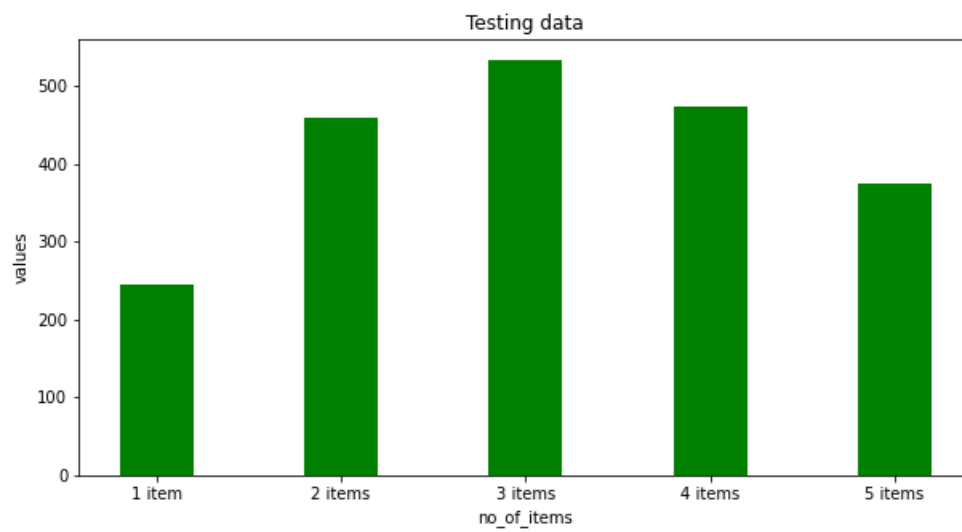
The Amazon Bin Image Dataset comprises over 500,000 photos and metadata from bins in an Amazon Fulfillment Center that is currently operational. The bin photos in this dataset were collected while robot units were transporting pods as part of typical Amazon Fulfillment Center operations.

Exploratory Visualization

I have split the data into train, test and split. Also splited the pictures according to the number of the objects in it



```
plt.show()
```





Algorithms and Techniques

I made a Deep Learning model that would assist in counting the items in each image by utilizing a pre-trained model like Resnet. This model, which is pretrained and can be tweaked to categorize photos from various use cases, is commonly used for image classification. Several training tasks will be run in AWS SageMaker notebook to adjust this pretrained model to our use case.

Benchmark

I will compare my results with this person, he is using the same data set and he got 55.67 accuracy

https://github.com/silverbottle/abid_challenge

Methodology

Data Pre-processing

I have first downloaded the dataset and then put the photos and metadata in the same directory. In the python script you will find the data pre-processing and data transformation like normalisation and resizing of the data which helps in reducing the number of duplicates in the database also we seize the images because deep learning models train faster on smaller images.

Implementation & Refinement

Firstly I downloaded the data and then I uploaded it to the s3 bucket. Then I made an estimator for training. It is a simple classification network for counting tasks. The image will be classified as one of the categories by the deep CNN (0-5). It is trained using the resnet 50 layer architecture. After that I made a training job and got the hyperparameter tuning values. Then I made another training job with the new hyperparameters. As shown in the next figure this is the accuracy that the model got, which is around 30% and this is not a good one. Also the root mean squared error is not a good one ,After trying another Resnet model I got Accuracy: 34.21903052064632%

. I would consider choosing another model or making more pre-processing to the data to get better results. Because this changed my results when I played around on those factors but for sure it can be better with more sophisticated techniques. But also we have to take care of the training cost.

```
Accuracy: 30.72866730584851%, Testing Loss: 1.5076169905932273
Downloading: "https://download.pytorch.org/models/resnet18-5c106cde.pth" to /root/.cache/torch/hub/checkpoints/resnet18-5c106cde.pth
#015  0% | 0.00/44.7M [00:00<?, ?B/s]#015 60%[██████████] | 26.7M/44.7M [00:00<00:00, 280MB/s]#015100%[██████████] 44.7M/44.7M [00:00<00:00, 289MB/
2022-11-08 21:15:35,543 sagemaker-training-toolkit INFO      Reporting training SUCCESS

2022-11-08 21:16:27 Uploading - Uploading generated training model
2022-11-08 21:16:27 Completed - Training job completed
Training seconds: 1943
Billable seconds: 1943
```

Results

Model Evaluation and Validation & Justification

I have made hyperparameter tuning and it gave better results at the end. There are multiple more ways to get better results like using bigger fine tuning data sets (I used RESNET 50). I faced some issues and Complications regarding the training time of the model. I used a small instance to keep the budget low, but it kept giving me error so I decided to change it with a bigger more expensive one. Regarding (silverbottlep Eunbyung Park) the person I was comparing my results(30.72 accuracy and 1.5 RMSE) with, he had better results than me (55.67 accuracy and 0.930 RMSE). I inspected his code and I found that he used resnet 34 and splitted the dataset into 3 categories, train validate and test. Also there were some differences in the data pre-processing between me and him.