Warehouse Storage Optimization - Report 1

Dhruvil Dave, Dhatri Kapuriya, Harvish Jariwala, Nisarg Thoriya
School of Engineering and Applied Sciences
Ahmedabad Univerity
Ahmedabad, Gujarat, India - 380009
dhruvil.d@ahduni.edu.in (AU1841003)
dhatri.k@ahduni.edu.in (AU1841129)
harvish.j@ahduni.edu.in (AU1841050)
nisarg.t@ahduni.edu.in (AU1841142)

***Abstract*—This is the first progress report of our group *Gopher - Group 4* for Machine Learning (CSE523) course project.**

***Index Terms*—classification; data preprocessing; clustering; graphical models**

Introduction

For our project, we decided to use the Amazon Bin Images Dataset. This is originally a Computer Vision dataset. The Amazon Bin Image Dataset contains over 530,000 images and metadata from bins of a pod in an operating Amazon Fulfillment Center.

Tasks Performed and Outcomes

Data acquisition

The bin images in this dataset are captured as robot units carry pods as part of normal Amazon Fulfillment Center operations. Along with the images, there were also metadata provided for each particular image in "JSON" format. So our first task was to download the entire dataset that was hosted over Amazon S3 Bucket.

Since the dataset is over 30GiB in size, we had to use Docker loaded with Amazon AWS CLI to download the dataset locally. That process takes about 6-7 hours. The command to reproduce the original dataset is:

```
docker run -it --rm -v "$PWD/data":/aws \
    amazon/aws-cli s3 \
    cp s3://aft-vbi-pds/ . \
    --no-sign-request --recursive
```

Data Preprocessing and Cleaning

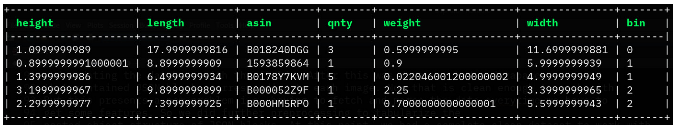
Once the entire dataset was acquired, we took a look at a sample of images and its corresponding metadata to understand how the dataset is structured. A sample image looks like this:

The corresponding metadata block contains information like *ASIN* (Amazon Standard Identification Number), quantity, weight, length, height and width for each product. Since we were not going to use any of the deep learning techniques, we decided to drop the images entirely from our dataset and here we were left with alomst 530,000 JSON files to be cleaned and preprocessed.

The first step we did was to collect all the json blocks and compiled a SQLite database so that all the data can be queried easily. We used SQLite because of two main reasons:

- File Format Database would help us easily distribute our changes amongst the team
- First Class JSON querying support builtin

We wrote a simple python script to automate the process of parsing the json file and updating the database with the fields. After this was done, we had a database that contained JSON blocks corresponding to each image. Now that is clean enough to work with. But this presents a new problem i.e. having to fetch and parse the data everytime we have to get some features. So we did further preprocessing to tabularize the nested object structure of JSON blocks. At this point, we had a complete clean dataset to work with. We also assigned bin numbers so we can use classification algorithms for our purpose as there was no clear response variable that can could be used for classifying as all the other features are continuous in nature. This is a snapshot of the clean data.



```
+--------------------+---------------+------------+------+----------------------+---------------+-------+
| height             | length        | asin       | qnty | weight               | width         | bin   |
|--------------------+---------------+------------+------+----------------------+---------------+-------|
| 1.0999999989       | 17.9999999816 | B018240DGG | 3    | 0.5999999995         | 11.6999999881 | 0     |
| 0.8999999991000001 | 8.8999999909  | 1593859864 | 1    | 0.9                  | 5.9999999939  | 1     |
| 1.3999999986       | 6.4999999934  | B0178Y7KVM | 5    | 0.022046001200000002 | 4.9999999949  | 1     |
| 3.1999999967       | 9.8999999899  | B000052Z9F | 1    | 2.25                 | 3.3999999965  | 2     |
| 2.2999999977       | 7.3999999925  | B000HM5RPO | 1    | 0.7000000000000001   | 5.5999999943  | 2     |
+--------------------+---------------+------------+------+----------------------+---------------+-------+
```

Tasks for Upcoming Week

The main tasks to be performed in the upcoming week are:

1. Exploratory Data Analysis (EDA)
2. Feature Engineering
3. Basic Modeling

Bibliography

1. Seward (2020)
2. AWS (2020)

AWS, Amazon. 2020. "Amazon Bin Images Dataset." https://registry.opendata.aws/amazon-bin-imagery/.

Seward, Calvin. 2020. "Optimizing Warehouse Operations with Machine Learning on Gpus." *NVIDIA Developer Blog*. https://developer.nvidia.com/blog/optimizing-warehouse-operations-machine-learning-gpus/.