Introduction to ML Productionization

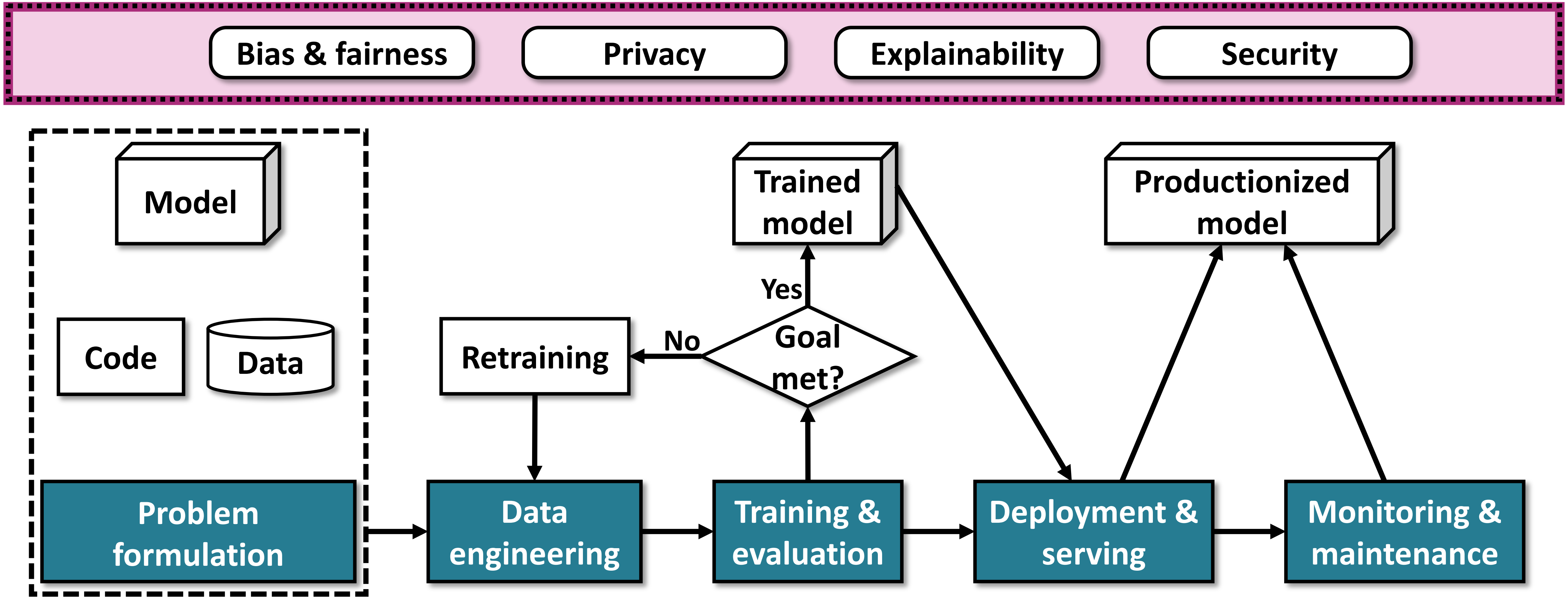
Rachel Hu
Applied Scientist @ Amazon MLU

Amazon Machine Learning University (MLU) - MLOps Lecture 1

Schedule

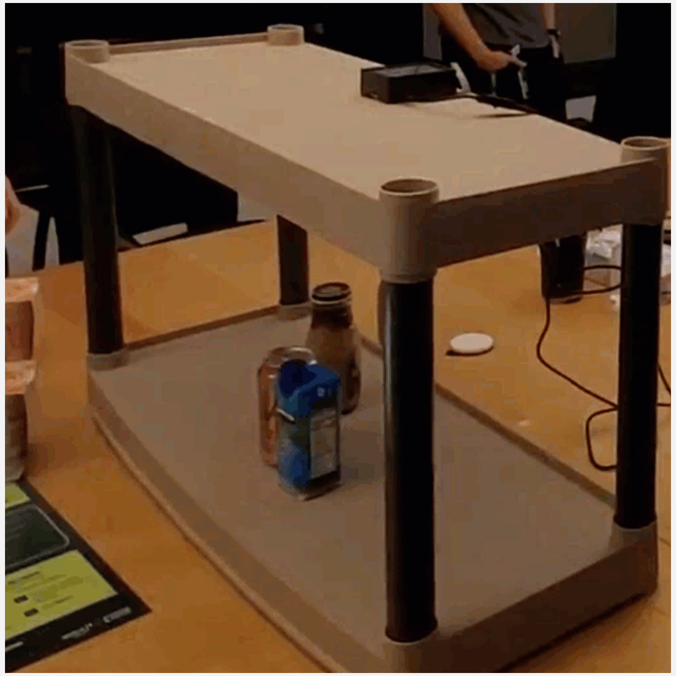
- [Lecture] Intro to ML Lifecycle – 15 min
- [Lab] Build an end-to-end ML system – 30 min
  - » https://github.com/goldmermaid/MLU-MLOps-Lab
- Q&A - 15 min

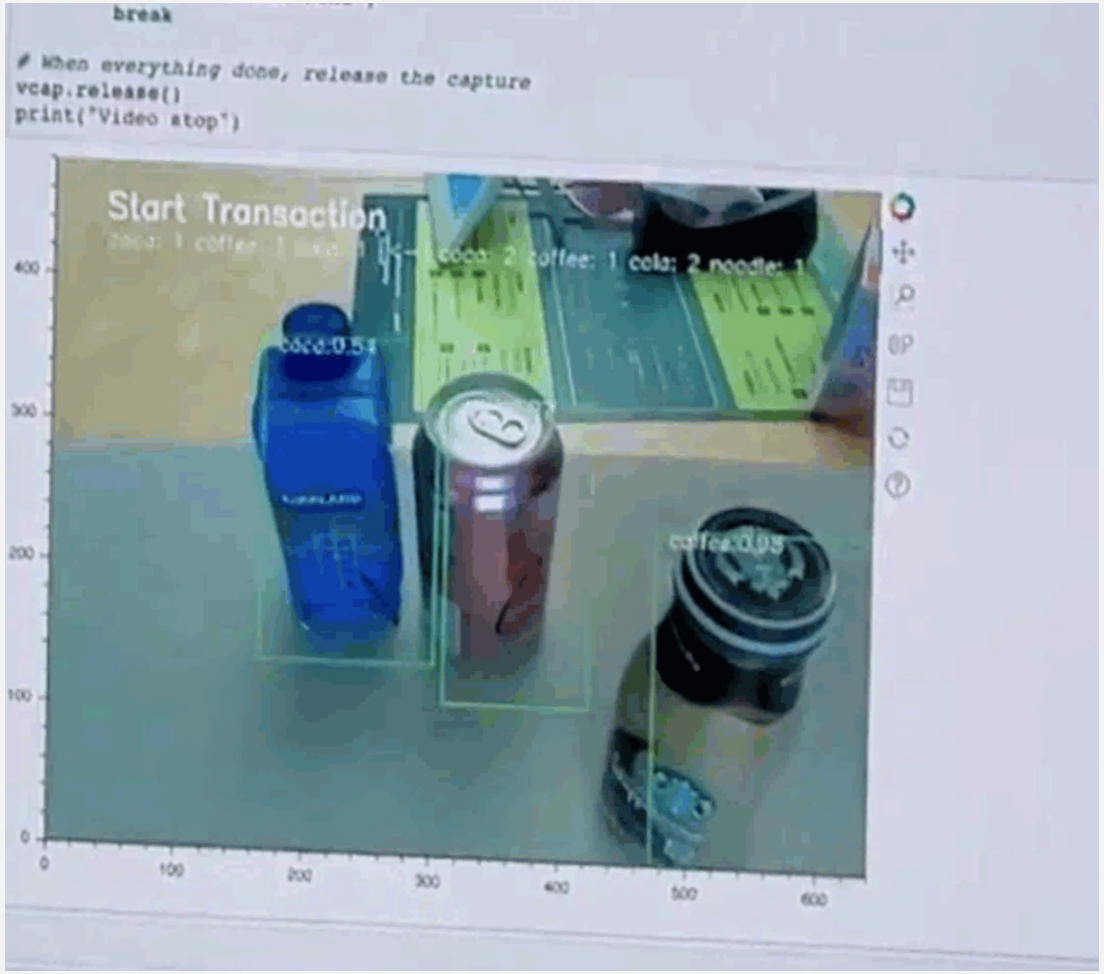
The ML lifecycle



A Simple ML System

Suppose that we are building a visual search system:



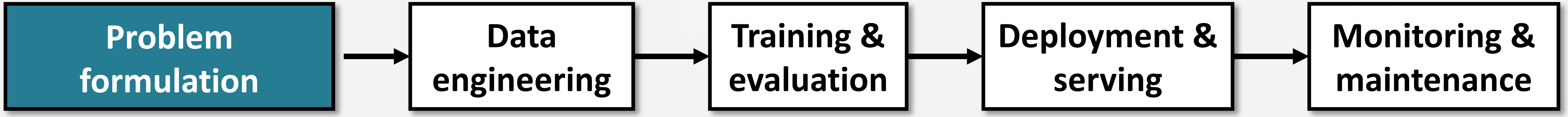


```
        break

# When everything done, release the capture
vcap.release()
print('Video stop')
```

MACHINE LEARNING UNIVERSITY

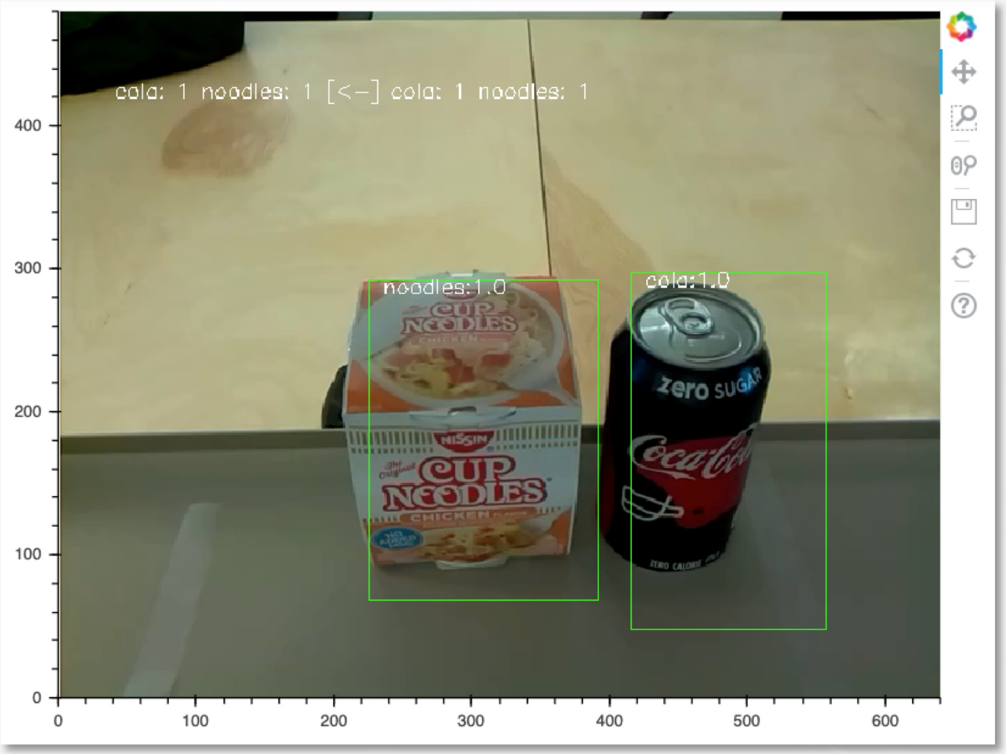
A Simple ML System - Problem Formulation

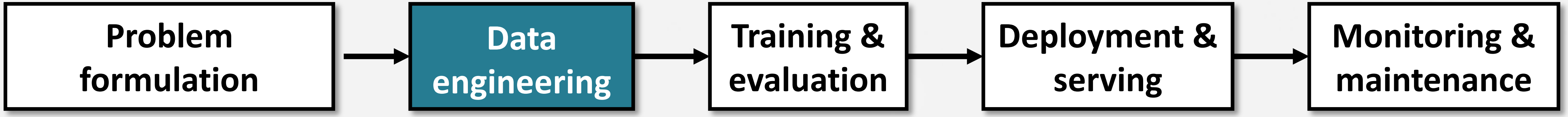
1. What are the inputs and outputs?
2. How to measure this ML system's performance?
3. What are the hardware/software requirements?
   a. Batch inference, online inference or both?
   b. On-premise, cloud based or both?
   c. Which one is more cost efficient?



A Simple ML System - Data Engineering

- Data Ingestion
  - Web sourcing
  - Simulating
  - Manually collecting
- Data Processing
  - Cleaning, labeling, etc.
  - Feature engineering
  - Data augmentation



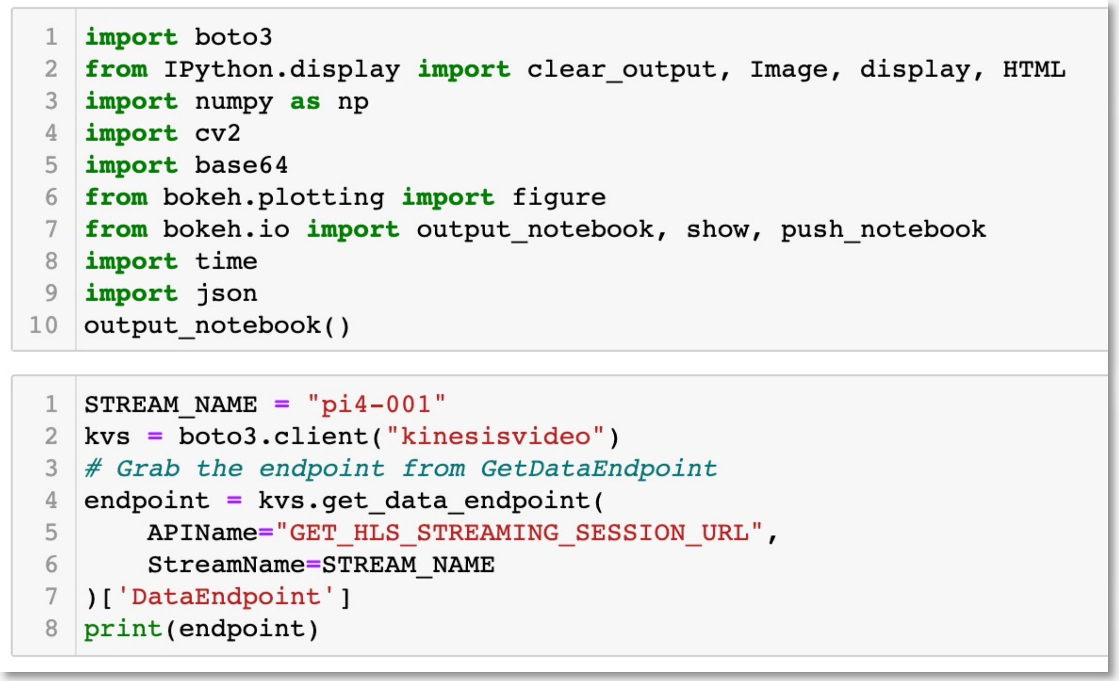


Problem formulation → Data engineering → Training & evaluation → Deployment & serving → Monitoring & maintenance

MACHINE LEARNING UNIVERSITY

A Simple ML System - Data Engineering

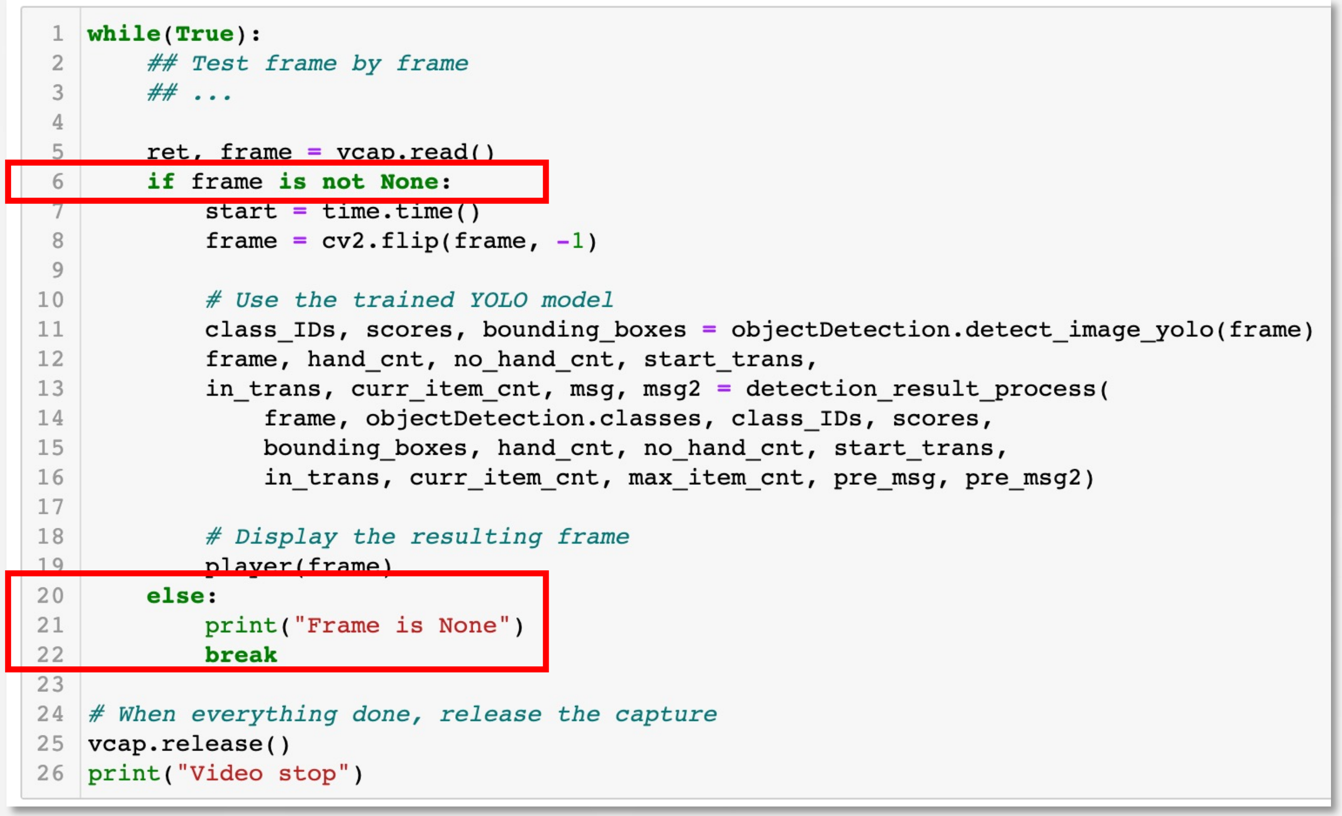
Data Ingestion, Processing, and Transforming



```python
import boto3
from IPython.display import clear_output, Image, display, HTML
import numpy as np
import cv2
import base64
from bokeh.plotting import figure
from bokeh.io import output_notebook, show, push_notebook
import time
import json
output_notebook()
```

```python
STREAM_NAME = "pi4-001"
kvs = boto3.client("kinesisvideo")
# Grab the endpoint from GetDataEndpoint
endpoint = kvs.get_data_endpoint(
    APIName="GET_HLS_STREAMING_SESSION_URL",
    StreamName=STREAM_NAME
)['DataEndpoint']
print(endpoint)
```

... ...



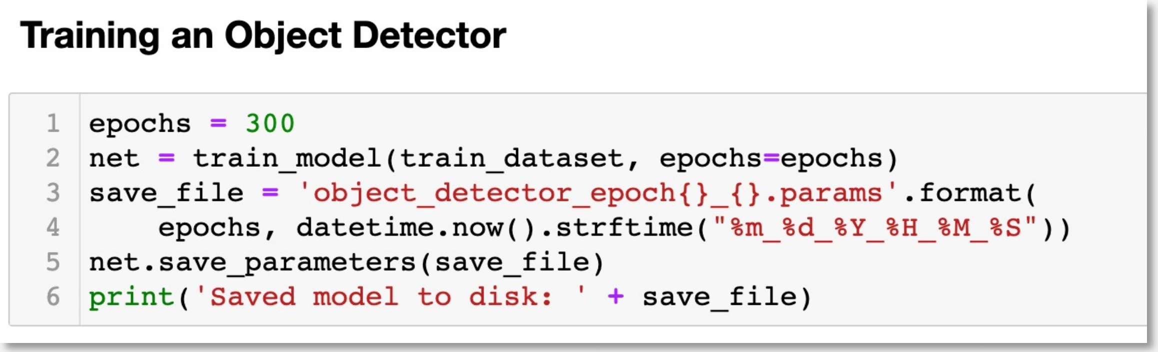
```python
while(True):
    ## Test frame by frame
    ## ...

    ret, frame = vcap.read()
    if frame is not None:
        start = time.time()
        frame = cv2.flip(frame, -1)

        # Use the trained YOLO model
        class_IDs, scores, bounding_boxes = objectDetection.detect_image_yolo(frame)
        frame, hand_cnt, no_hand_cnt, start_trans,
        in_trans, curr_item_cnt, msg, msg2 = detection_result_process(
            frame, objectDetection.classes, class_IDs, scores,
            bounding_boxes, hand_cnt, no_hand_cnt, start_trans,
            in_trans, curr_item_cnt, max_item_cnt, pre_msg, pre_msg2)

        # Display the resulting frame
        player(frame)
    else:
        print("Frame is None")
        break

# When everything done, release the capture
vcap.release()
print("Video stop")
```

VideoStream.ipydb

MACHINE LEARNING UNIVERSITY

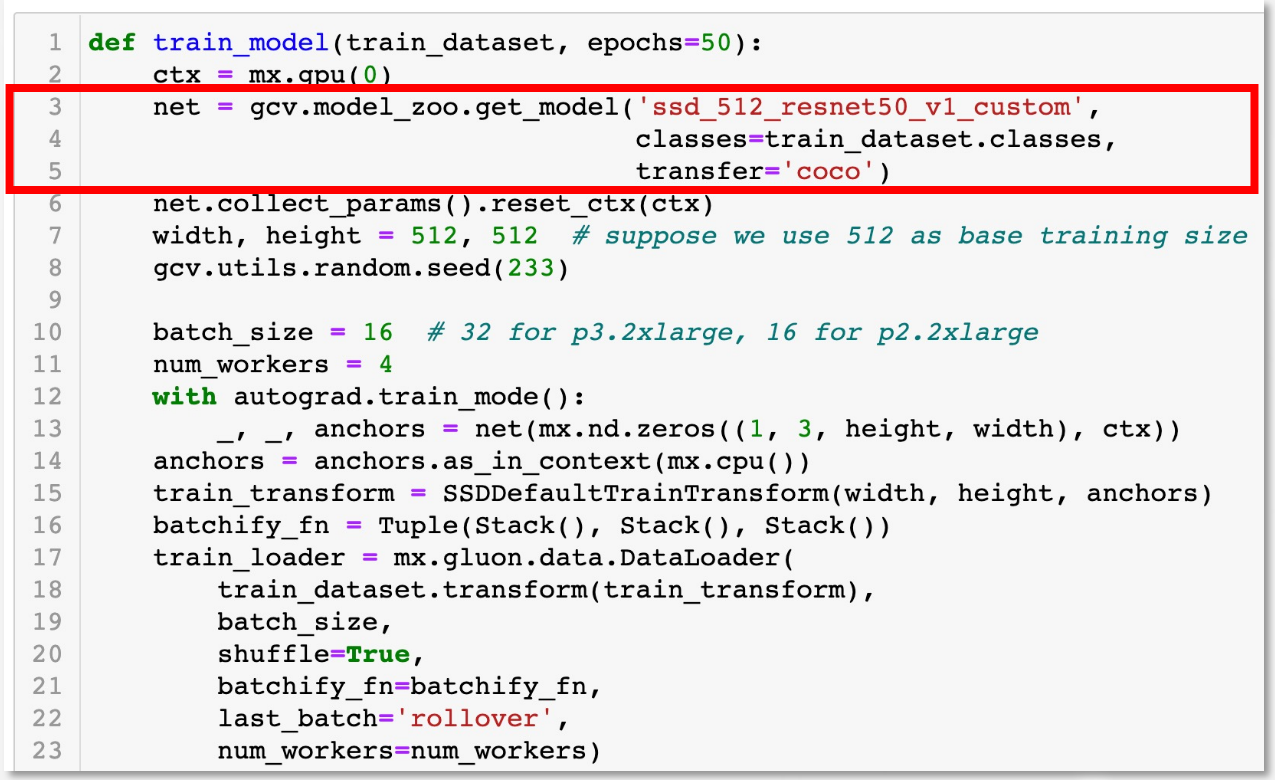
A Simple ML System - Modeling

training_object_detector.ipynb



Training an Object Detector

```
epochs = 300
net = train_model(train_dataset, epochs=epochs)
save_file = 'object_detector_epoch{}_{}.params'.format(
    epochs, datetime.now().strftime("%m_%d_%Y_%H_%M_%S"))
net.save_parameters(save_file)
print('Saved model to disk: ' + save_file)
```



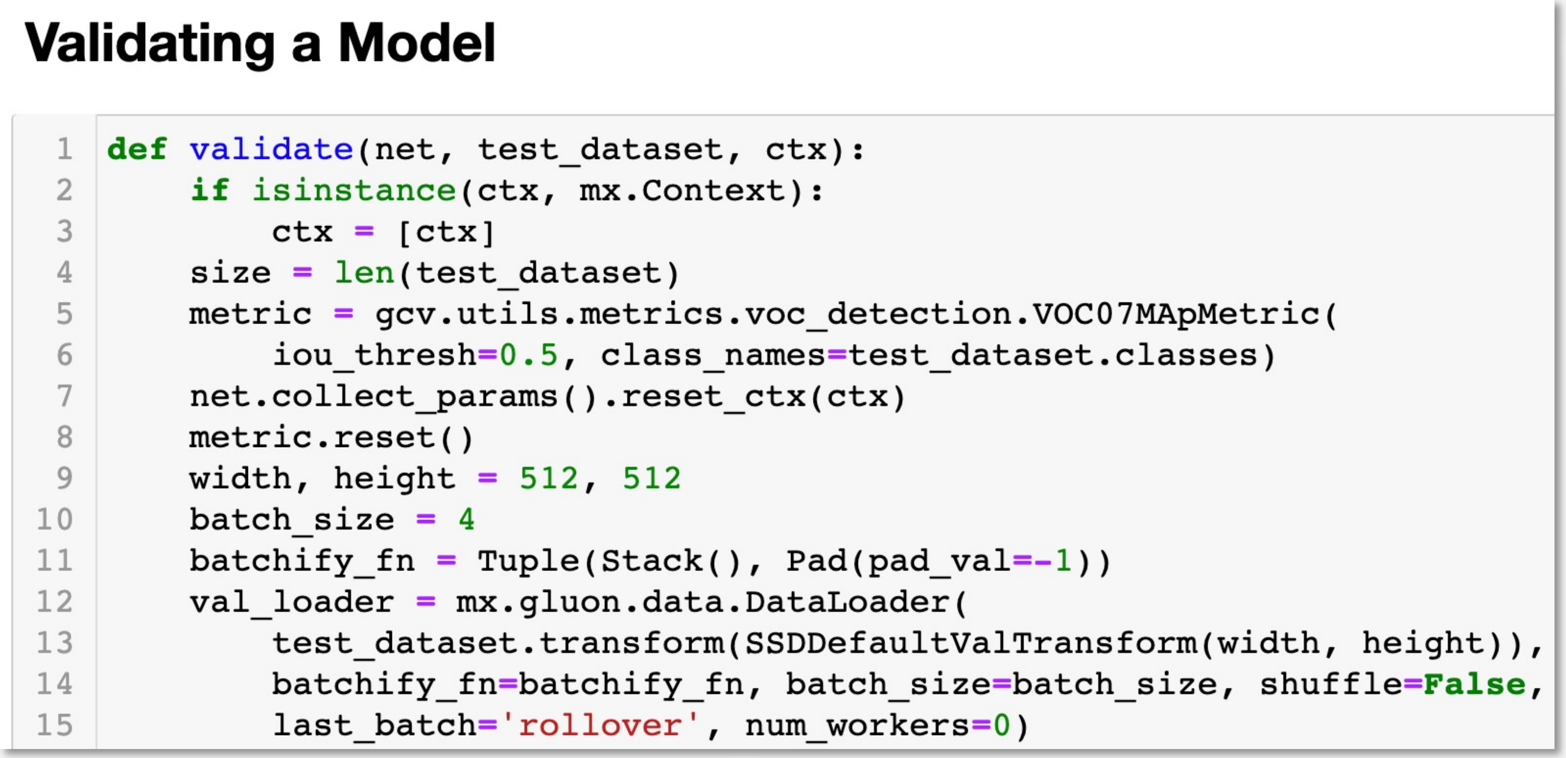
```
def train_model(train_dataset, epochs=50):
    ctx = mx.gpu(0)
    net = gcv.model_zoo.get_model('ssd_512_resnet50_v1_custom',
                                  classes=train_dataset.classes,
                                  transfer='coco')
    net.collect_params().reset_ctx(ctx)
    width, height = 512, 512  # suppose we use 512 as base training size
    gcv.utils.random.seed(233)

    batch_size = 16  # 32 for p3.2xlarge, 16 for p2.2xlarge
    num_workers = 4
    with autograd.train_mode():
        _, _, anchors = net(mx.nd.zeros((1, 3, height, width), ctx))
    anchors = anchors.as_in_context(mx.cpu())
    train_transform = SSDDefaultTrainTransform(width, height, anchors)
    batchify_fn = Tuple(Stack(), Stack(), Stack())
    train_loader = mx.gluon.data.DataLoader(
        train_dataset.transform(train_transform),
        batch_size,
        shuffle=True,
        batchify_fn=batchify_fn,
        last_batch='rollover',
        num_workers=num_workers)
```

... ...

MACHINE LEARNING
UNIVERSITY

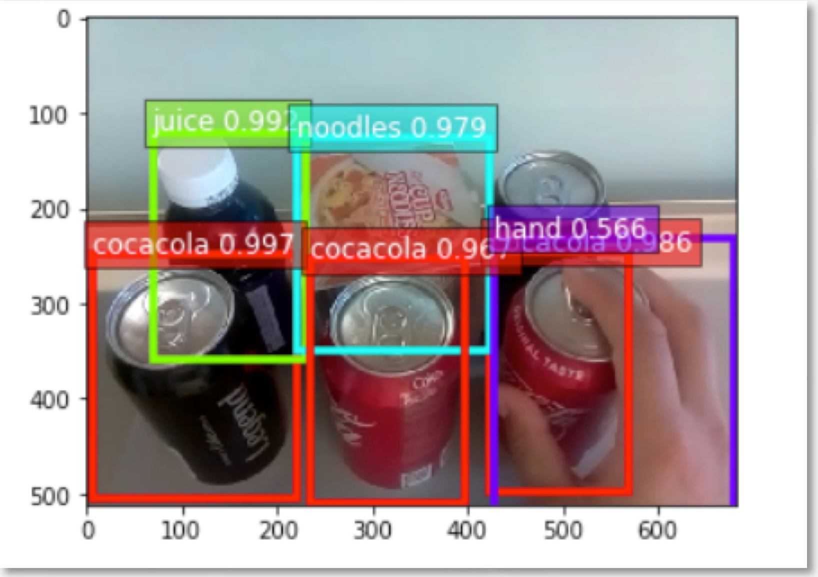
A Simple ML System - Testing

test_object_detector.ipynb



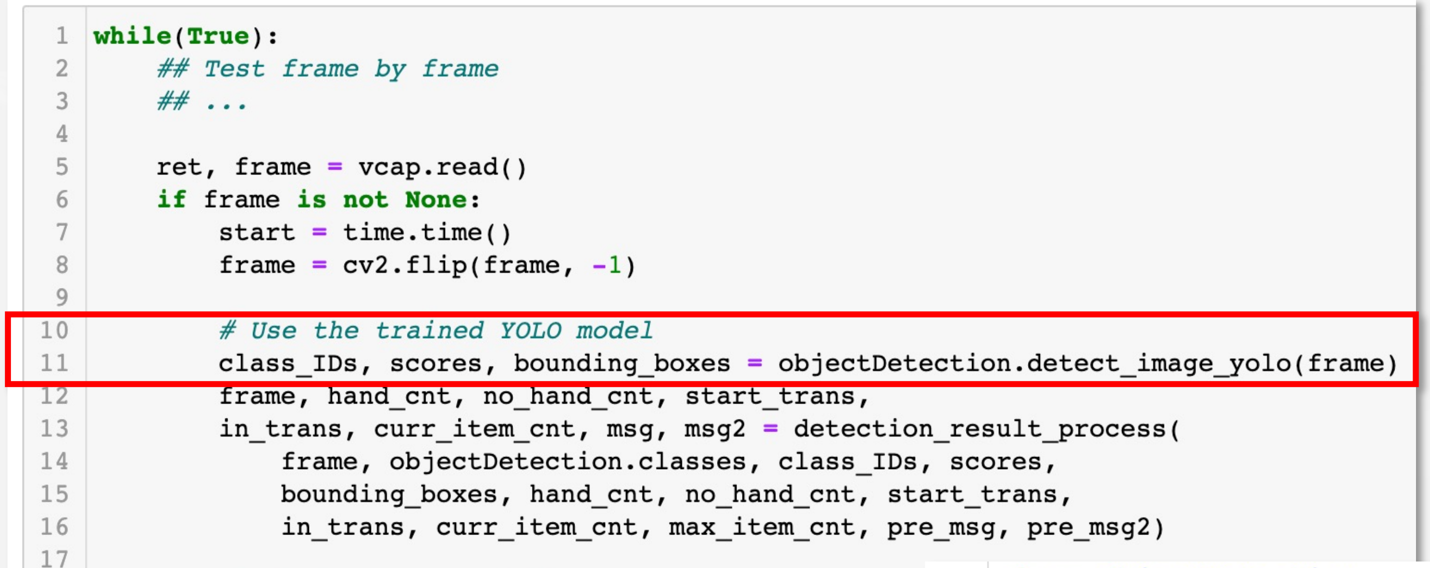
Validating a Model

```
def validate(net, test_dataset, ctx):
    if isinstance(ctx, mx.Context):
        ctx = [ctx]
    size = len(test_dataset)
    metric = gcv.utils.metrics.voc_detection.VOC07MApMetric(
        iou_thresh=0.5, class_names=test_dataset.classes)
    net.collect_params().reset_ctx(ctx)
    metric.reset()
    width, height = 512, 512
    batch_size = 4
    batchify_fn = Tuple(Stack(), Pad(pad_val=-1))
    val_loader = mx.gluon.data.DataLoader(
        test_dataset.transform(SSDDefaultValTransform(width, height)),
        batchify_fn=batchify_fn, batch_size=batch_size, shuffle=False,
        last_batch='rollover', num_workers=0)
```



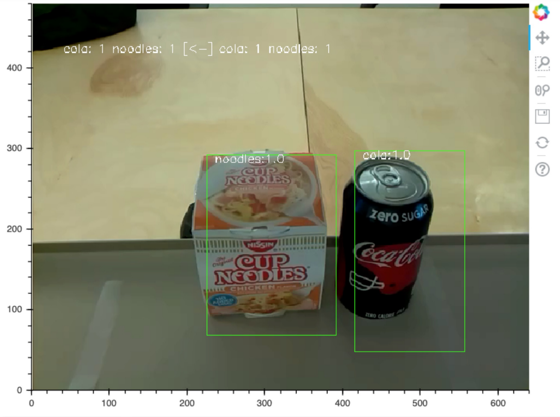
... ...

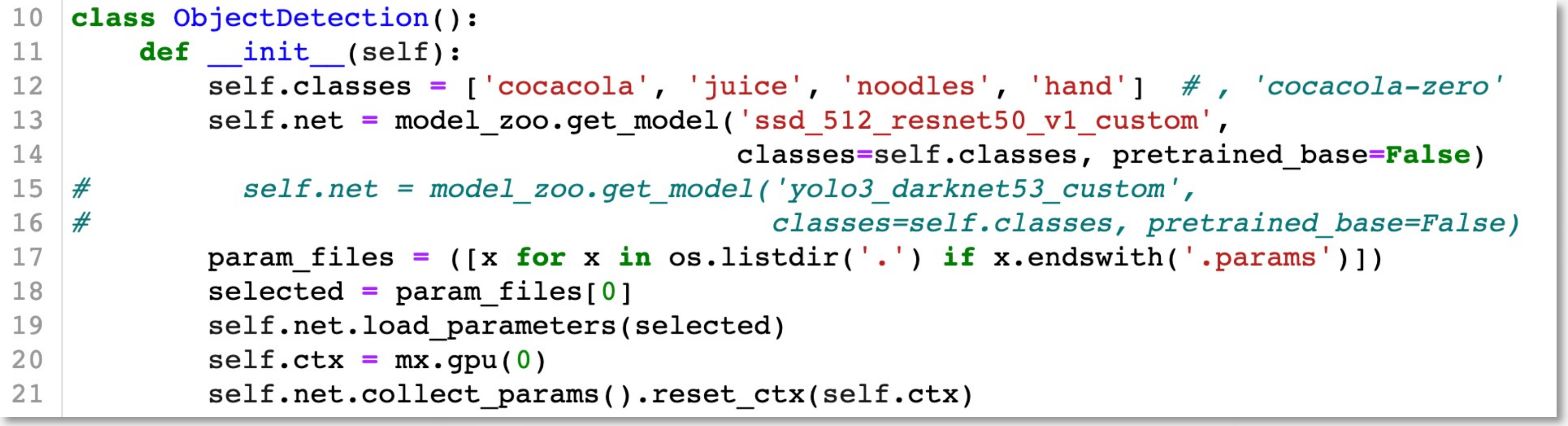
Deployment & Serving



```python
while(True):
    ## Test frame by frame
    ## ...

    ret, frame = vcap.read()
    if frame is not None:
        start = time.time()
        frame = cv2.flip(frame, -1)

        # Use the trained YOLO model
        class_IDs, scores, bounding_boxes = objectDetection.detect_image_yolo(frame)
        frame, hand_cnt, no_hand_cnt, start_trans,
        in_trans, curr_item_cnt, msg, msg2 = detection_result_process(
            frame, objectDetection.classes, class_IDs, scores,
            bounding_boxes, hand_cnt, no_hand_cnt, start_trans,
            in_trans, curr_item_cnt, max_item_cnt, pre_msg, pre_msg2)

        # Display the resulting frame
        player(frame)
    else:
        print("Frame is None")
        break

# When everything done, release the capture
vcap.release()
print("Video stop")
```

VideoStream.ipydb





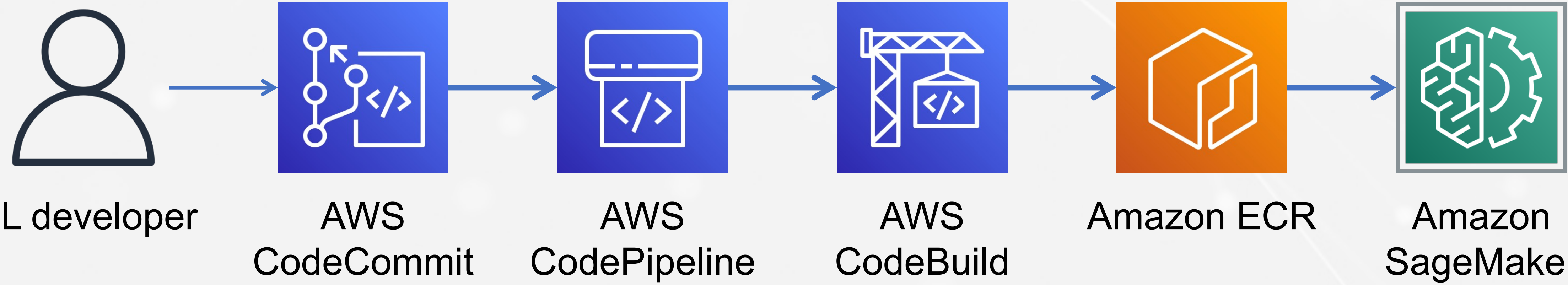
```python
class ObjectDetection():
    def __init__(self):
        self.classes = ['cocacola', 'juice', 'noodles', 'hand']  # , 'cocacola-zero'
        self.net = model_zoo.get_model('ssd_512_resnet50_v1_custom',
                                       classes=self.classes, pretrained_base=False)
#         self.net = model_zoo.get_model('yolo3_darknet53_custom',
#                                        classes=self.classes, pretrained_base=False)
        param_files = ([x for x in os.listdir('.') if x.endswith('.params')])
        selected = param_files[0]
        self.net.load_parameters(selected)
        self.ctx = mx.gpu(0)
        self.net.collect_params().reset_ctx(self.ctx)
```

object_detection.py

Lab

Lab Exercise

In this exercise, we will create an MLOps pipeline using CloudFormation



Lab instructions: https://github.com/goldmermaid/MLU-MLOps-Lab

- Email: rlhu@amazon.com
- LinkedIn: https://www.linkedin.com/in/rachelsonghu

Thank you!