AWS re:Invent

AIM343-R

Build computer vision models with Amazon SageMaker

Nathalie Rauschmayr
Applied Scientist
Amazon Web Services

Agenda

Amazon SageMaker

Computer vision toolkit: GluonCV

GluonCV on Amazon SageMaker: A demo

Amazon SageMaker

AWS re:Invent

aws

Amazon SageMaker: Build, train, and deploy ML

LABEL

Amazon SageMaker Ground Truth

BUILD & TRAIN

Notebooks & Jobs

TUNE

Automatic Model Tuner

COMPILE

Amazon SageMaker Neo

CLOUD DEPLOYMENT

Endpoints

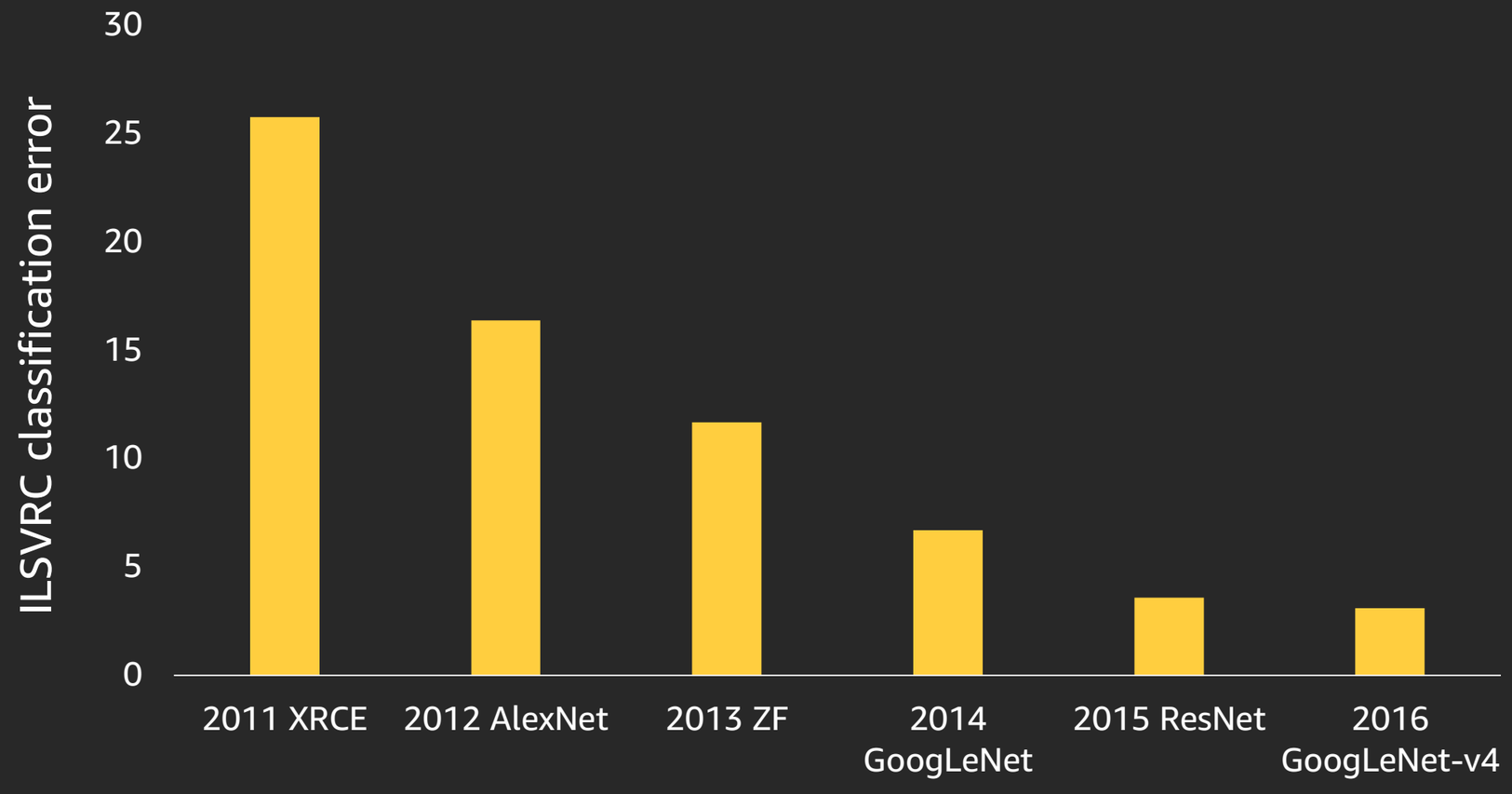
EDGE DEPLOYMENT

AWS IoT Greengrass

Computer vision toolkit: GluonCV

Why GluonCV?

Biggest challenge in deep learning? **Reproducing state of the art.**



Real-world stories

Different transcoding:

In 2016, the same ImageNet models trained by MXNet achieved on average 1% less accuracy than Torch

95 JPEG quality

85 JPEG quality

Source: “GluonCV — Deep Learning Toolkit for Computer Vision.” https://medium.com/apache-mxnet/gluoncv-deep-learning-toolkit-for-computer-vision-9218a907e8da

Real-world stories

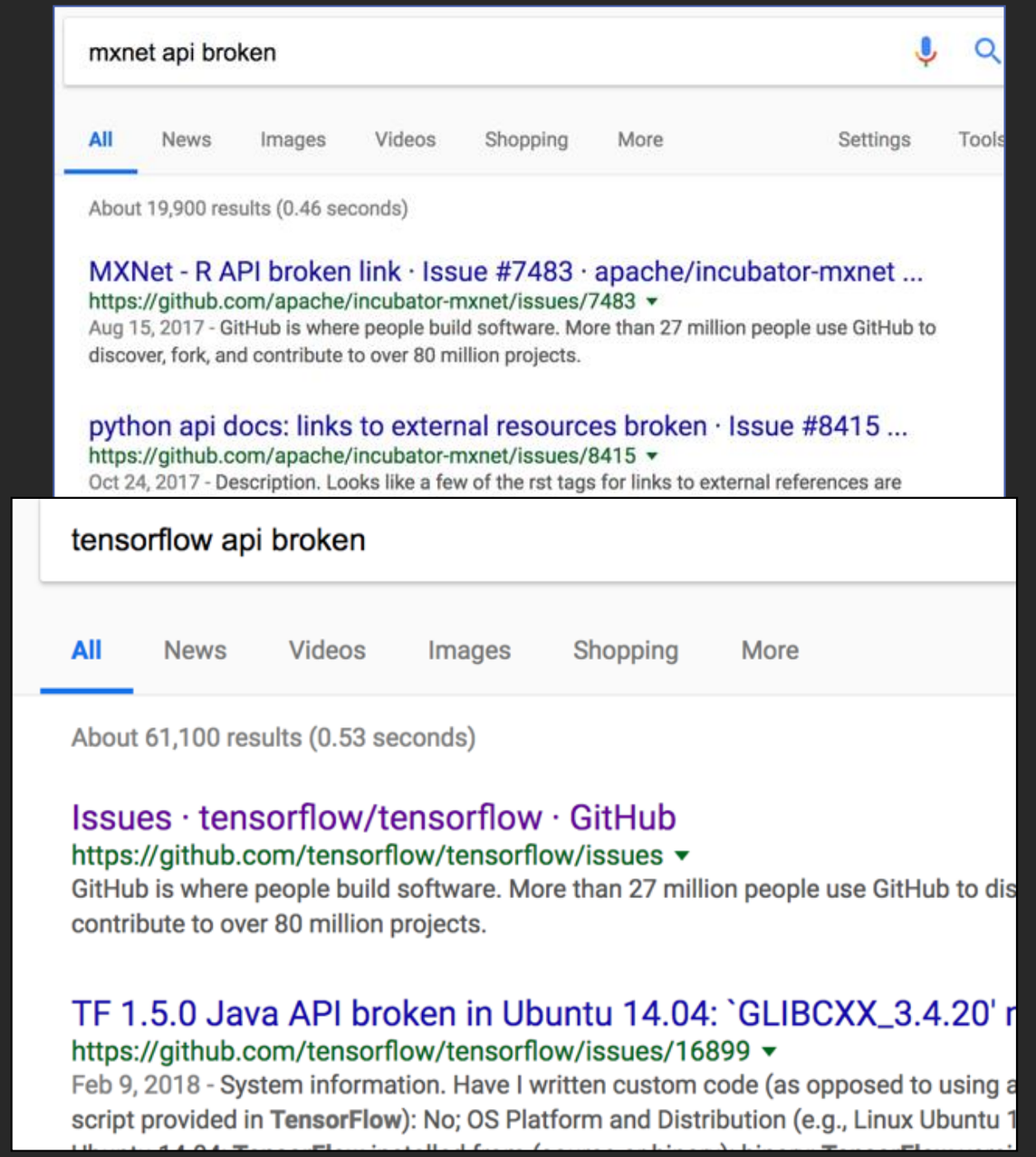
Order of data augmentations:

Using another open-source DL framework: Trained model accuracies cannot match previous internal version.

Reproducibility

My code will still run next year.

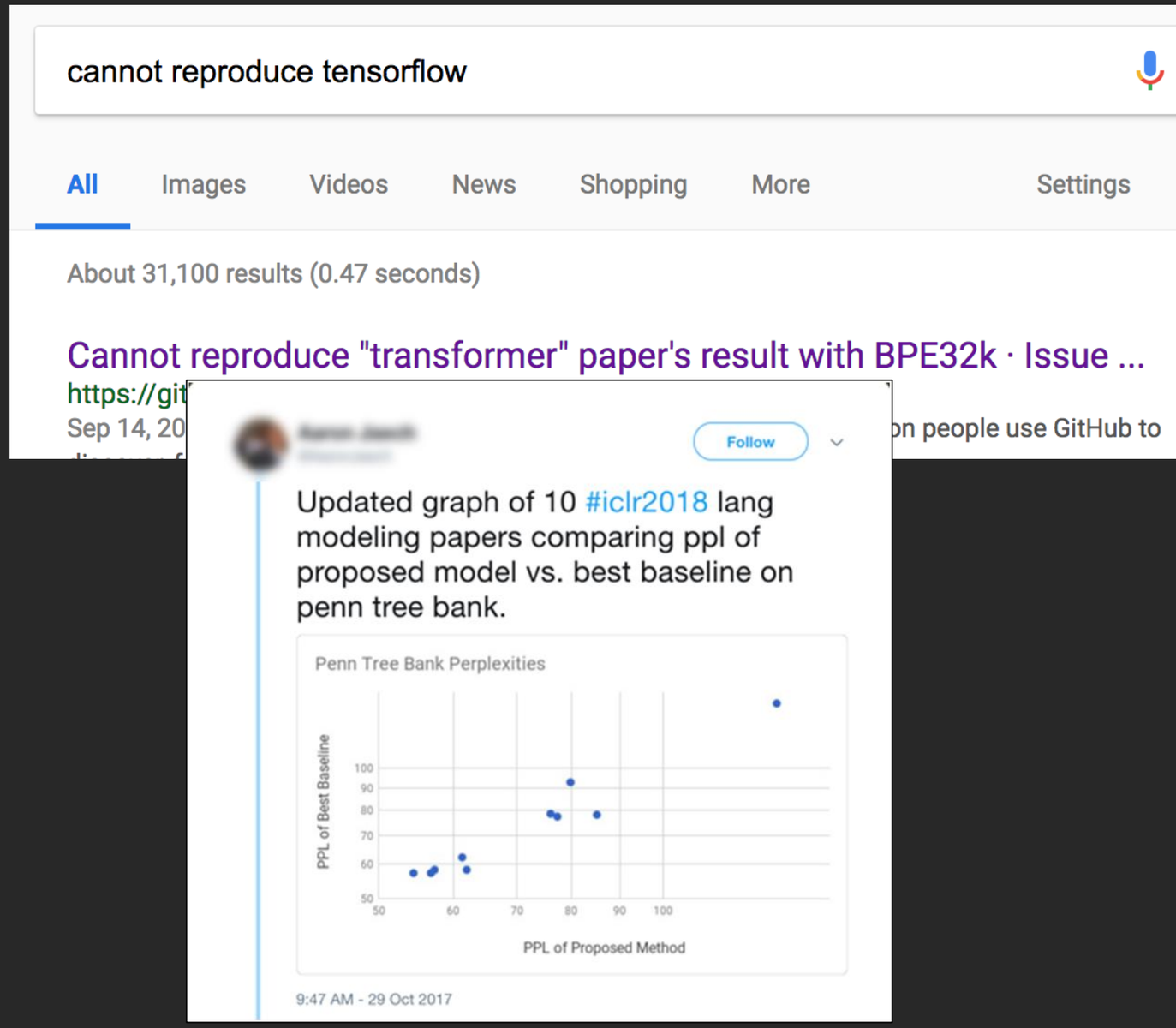
Sometimes, it’s not our fault.



Reproducibility

I will finish setting up the baseline model this afternoon.

Even though it may not be our fault again.



Starting from scratch can be hard

Even the most talented researchers will get blocked by trivial things.

Experience and instincts can be your enemies in certain circumstances.

Training is time-consuming, initialization and augmentation are randomized, and many implementation details need to be taken care of.

=> Debugging deep learning models is extremely difficult.

What does GluonCV provide?

State-of-the-art models

Official maintenance

Fast development

Reproducibility

Easy deployment

Pre-trained models

Image classification

More than 50+ pre-trained ImageNet models (ResNet, MobileNet…)

Object detection

SSD YOLOv3 Faster-RCNN

RFCN FPN

Pre-trained models

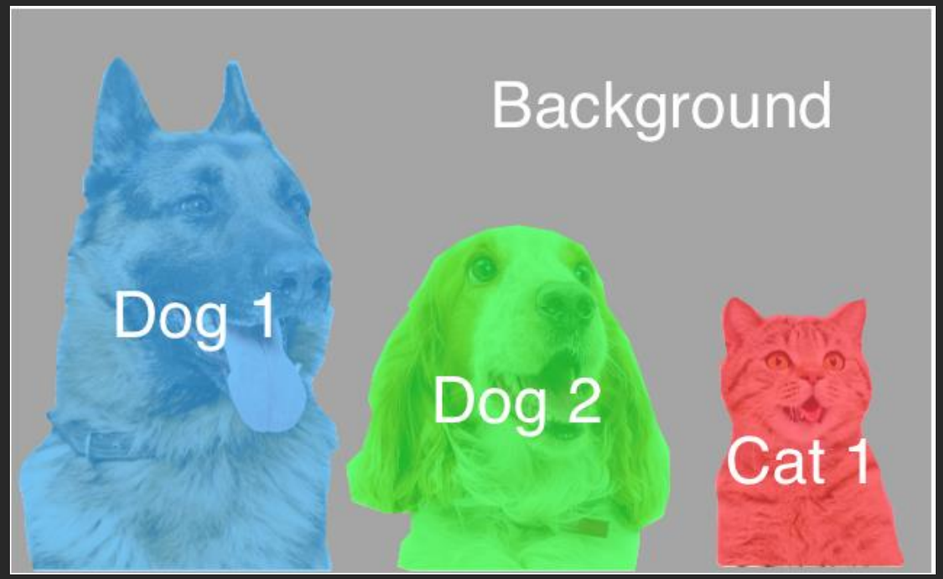
Semantic segmentation

FCN PSPNet

Mask-RCNN DeepLab

Instance segmentation

Mask R-CNN



Key point estimation

SimplePose

Others

Style transfer

MSGNet

GANs

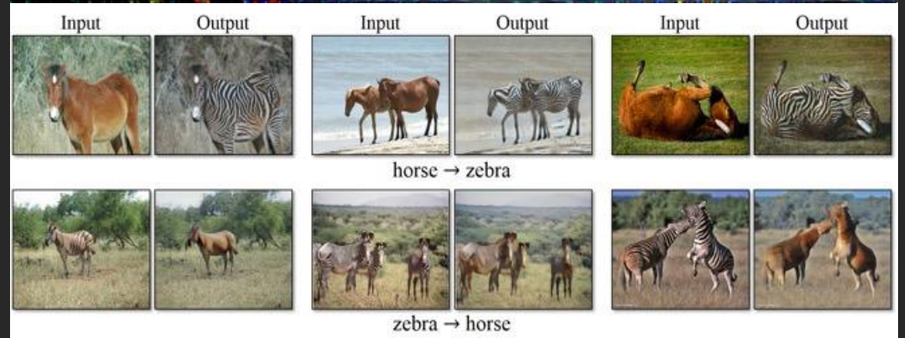
CycleGAN

SRGAN

WGAN

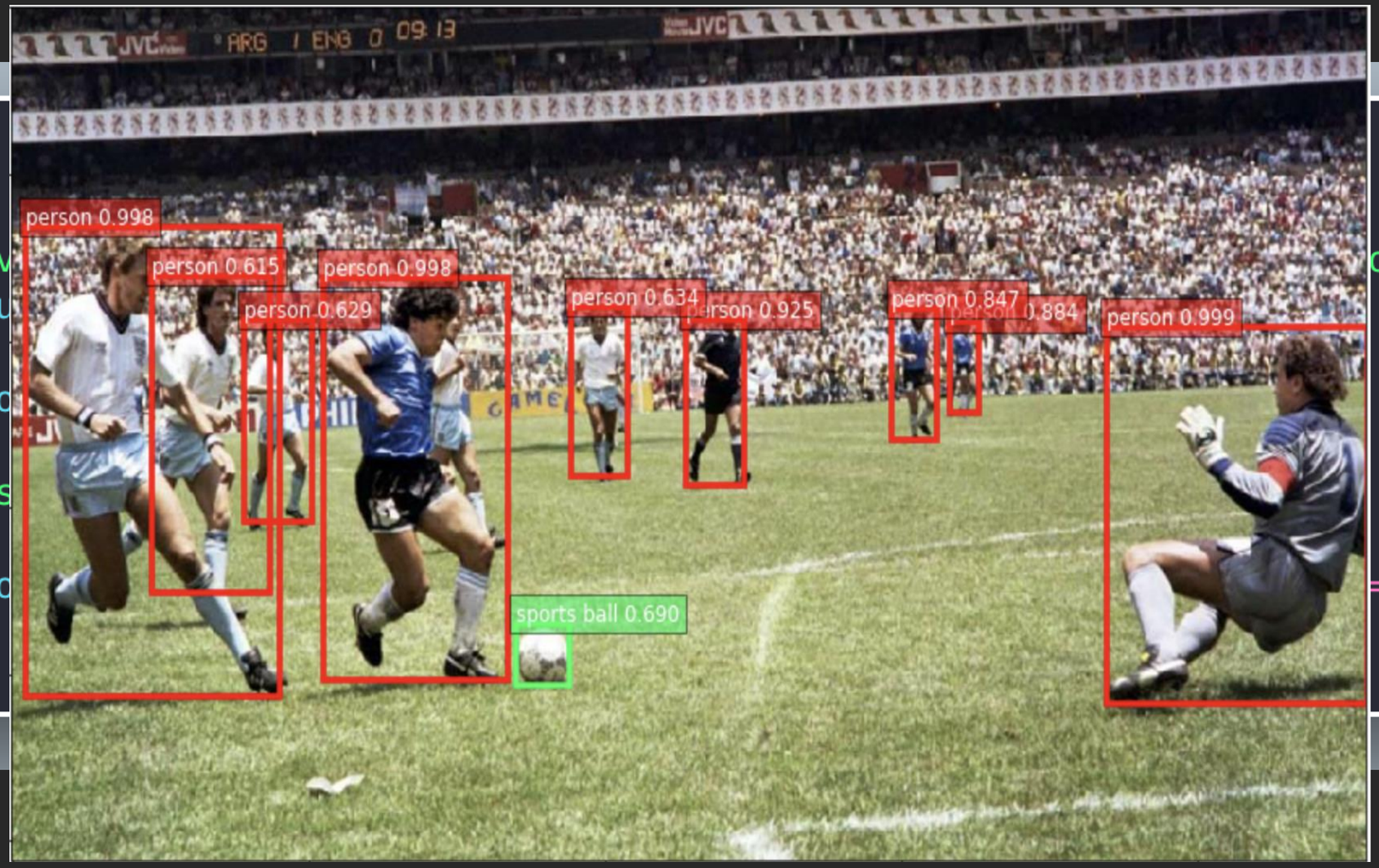
Re-identification

Market1501



GluonCV example

```
x, img = gcv
ctx = mx.gpu

net = gcv.mo

class_IDs, s

viz.plot_bbo
```




```
ort=512)

ctx=ctx)

=net.classes)
```

GluonCV on Amazon SageMaker

AWS re:Invent

aws

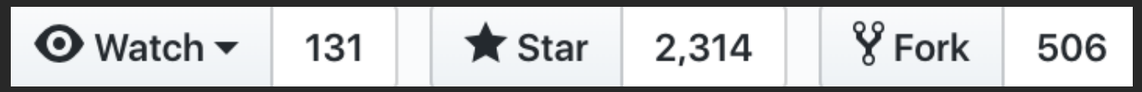
Code walk-through

```
from sagemaker.mxnet import MXNet

mxnet_estimator = MXNet( entry_point='train.py',
                         role='SageMakerRole',
                         train_instance_type='ml.m5.xlarge',
                         train_instance_count=1,
                         framework_version='1.3.0',
                         py_version='py2')

mxnet_estimator.fit({'train': 's3://data/train'})
```

Demo

Go and build!

https://github.com/dmlc/gluon-cv



https://gluon-cv.mxnet.io

Backup

Amazon SageMaker: Build, train, and deploy ML

LABEL

Amazon SageMaker Ground Truth

BUILD & TRAIN

Notebooks & Jobs

TUNE

Automatic Model Tuner

COMPILE

Amazon SageMaker Neo

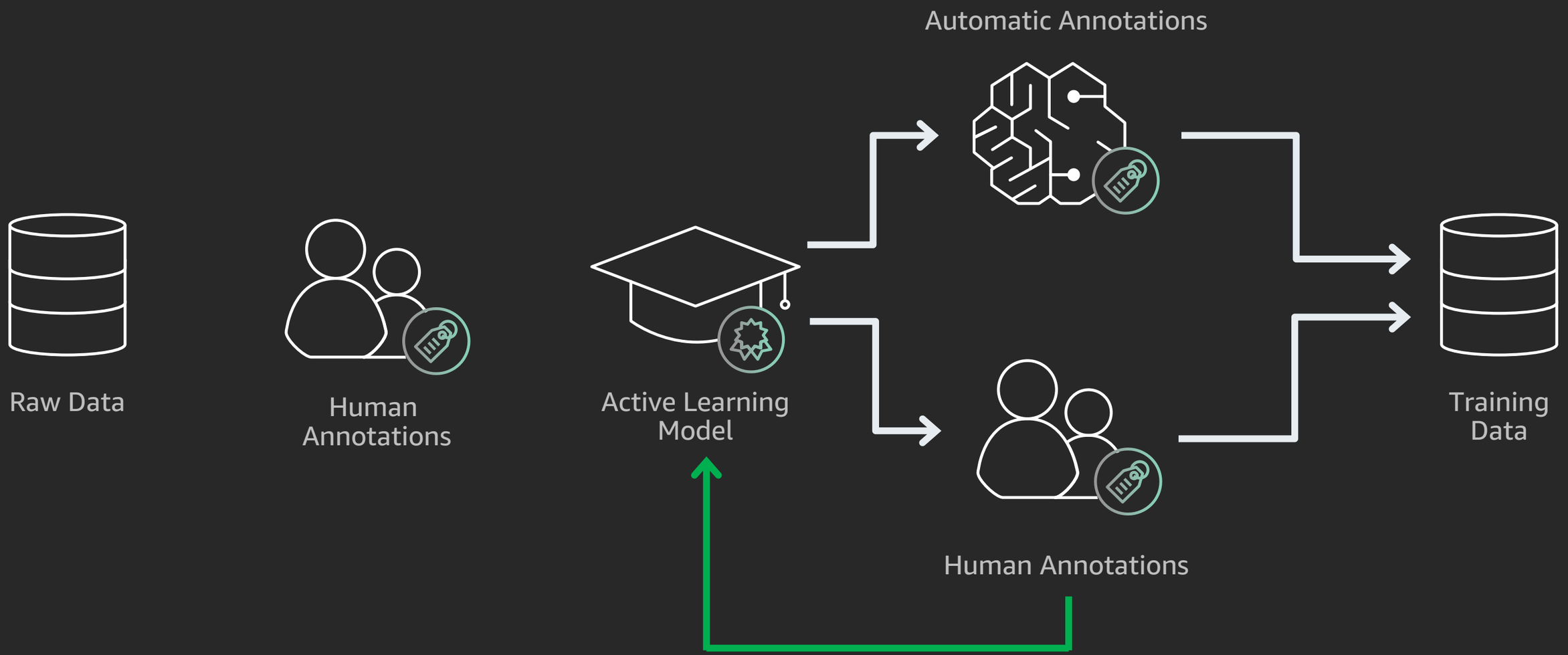
CLOUD DEPLOYMENT

Endpoints

EDGE DEPLOYMENT

AWS IoT Greengrass

Amazon SageMaker Ground Truth



Amazon SageMaker: Build, train, and deploy ML

LABEL

Amazon SageMaker Ground Truth

BUILD & TRAIN

Notebooks & Jobs

TUNE

Automatic Model Tuner

COMPILE

Amazon SageMaker Neo

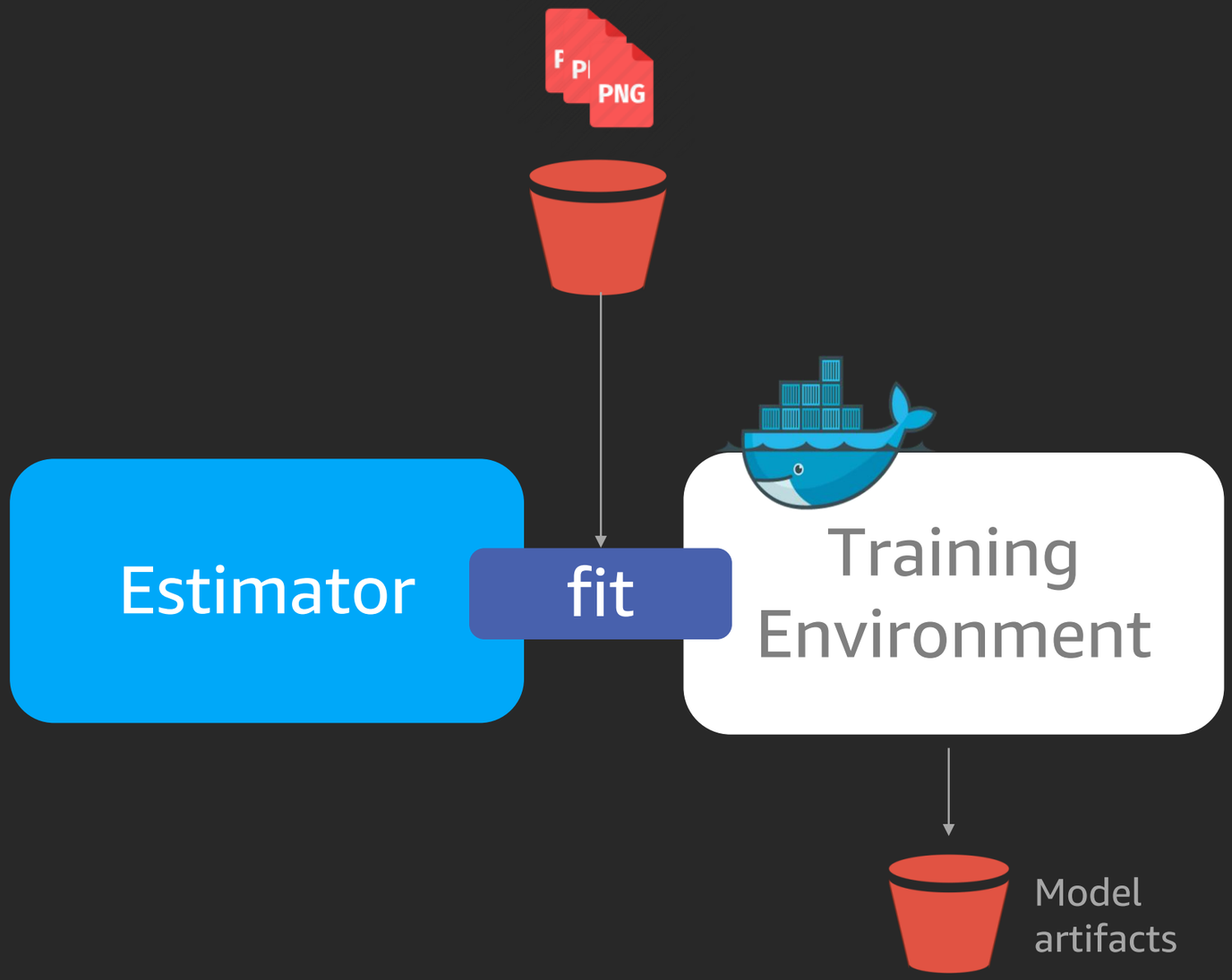
CLOUD DEPLOYMENT

Endpoints

EDGE DEPLOYMENT

AWS IoT Greengrass

Training



Amazon SageMaker: Build, train, and deploy ML

LABEL

Amazon SageMaker Ground Truth

BUILD & TRAIN

Notebooks & Jobs

TUNE

Automatic Model Tuner

COMPILE

Amazon SageMaker Neo

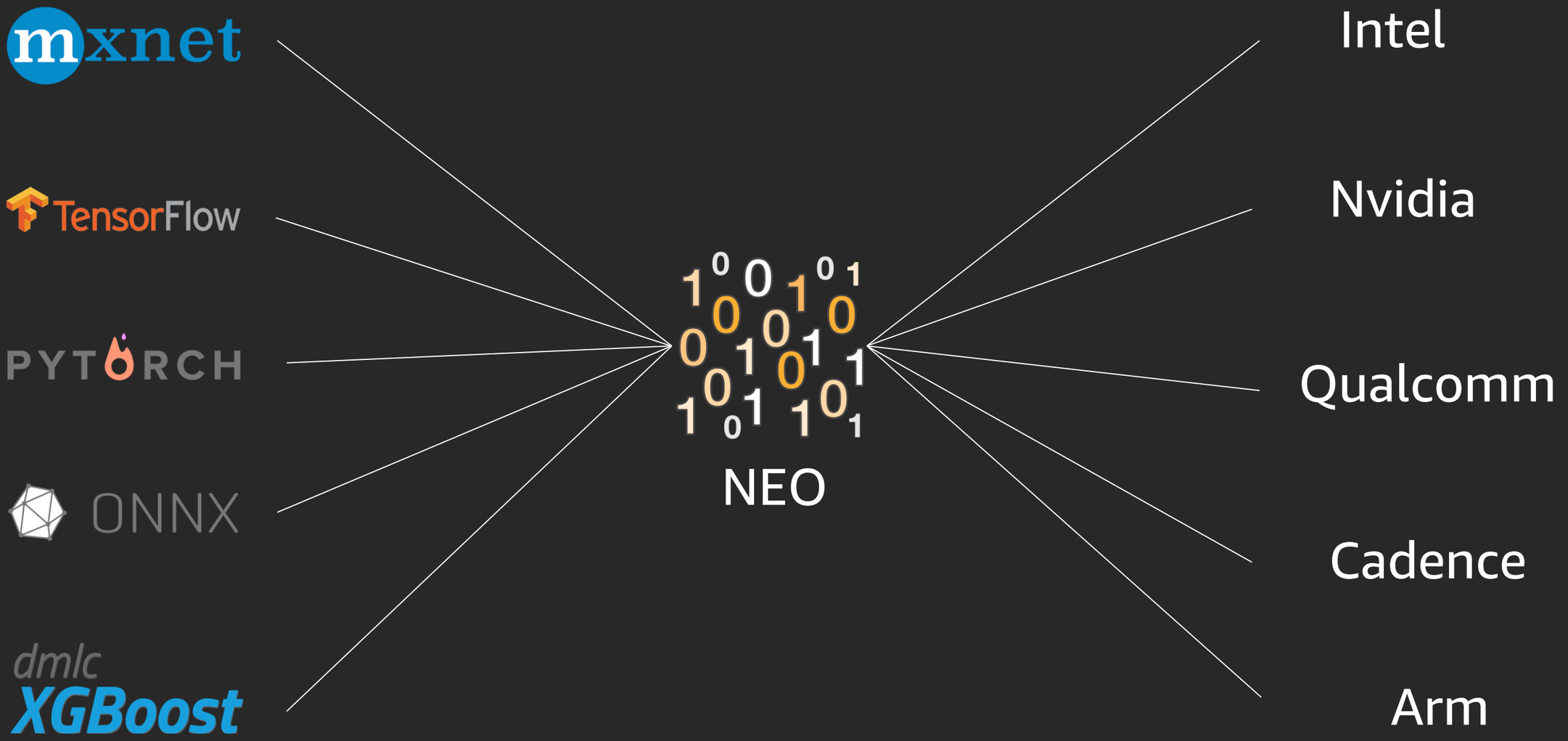
CLOUD DEPLOYMENT

Endpoints

EDGE DEPLOYMENT

AWS IoT Greengrass

Amazon SageMaker Neo: Train once, run anywhere



Amazon SageMaker: Build, train, and deploy ML

LABEL

Amazon SageMaker Ground Truth

BUILD & TRAIN

Notebooks & Jobs

TUNE

Automatic Model Tuner

COMPILE

Amazon SageMaker Neo

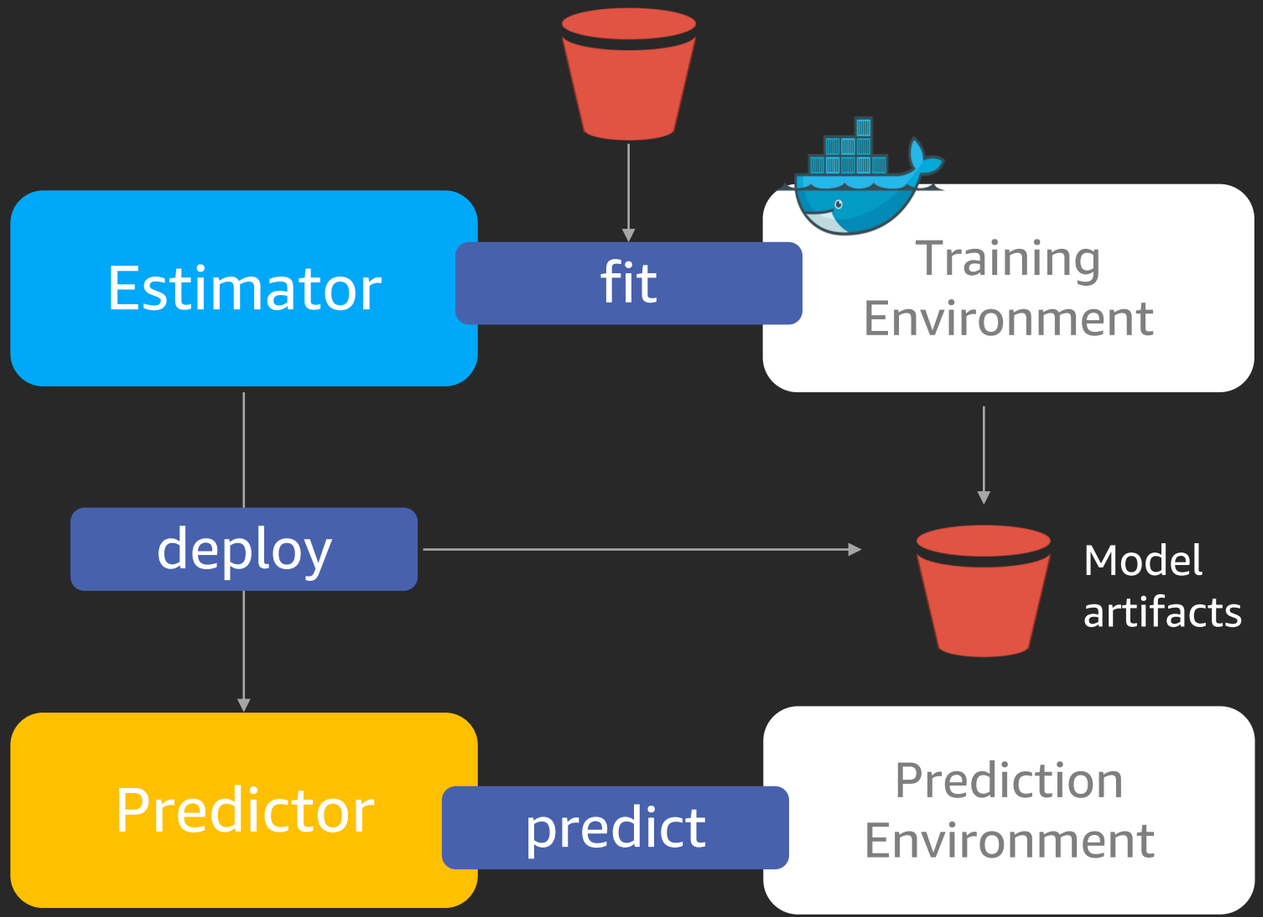
CLOUD DEPLOYMENT

Endpoints

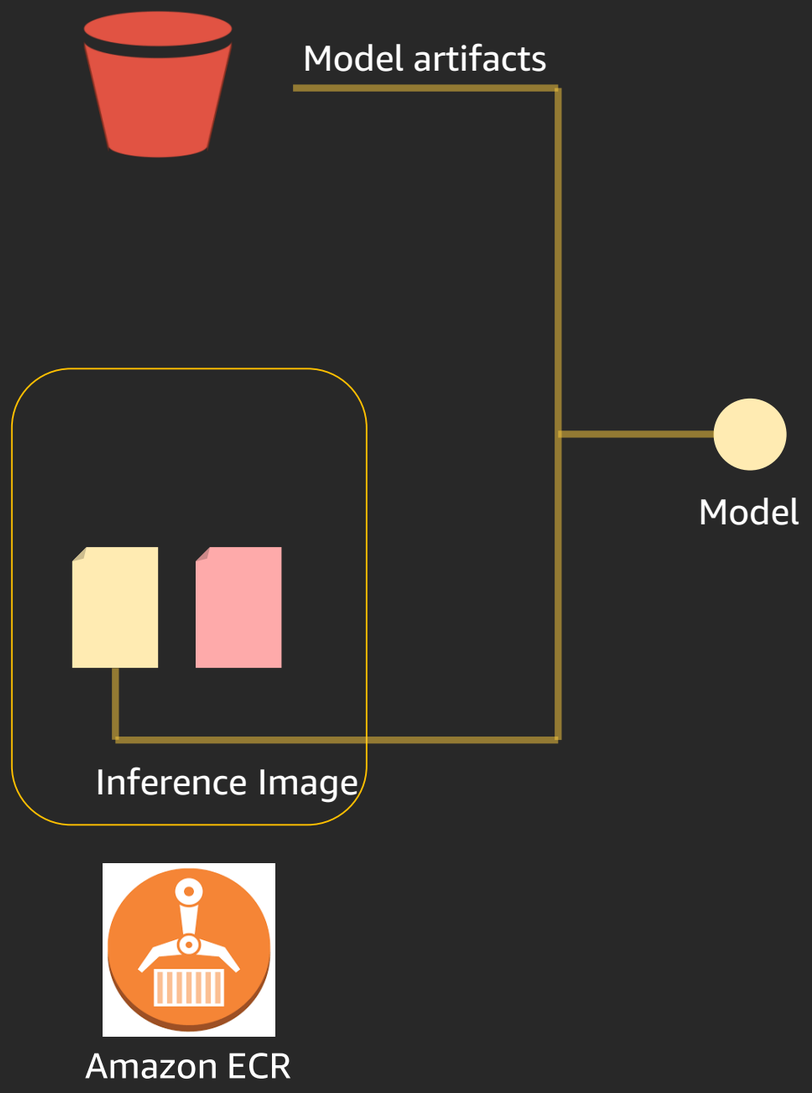
EDGE DEPLOYMENT

AWS IoT Greengrass

Hosting

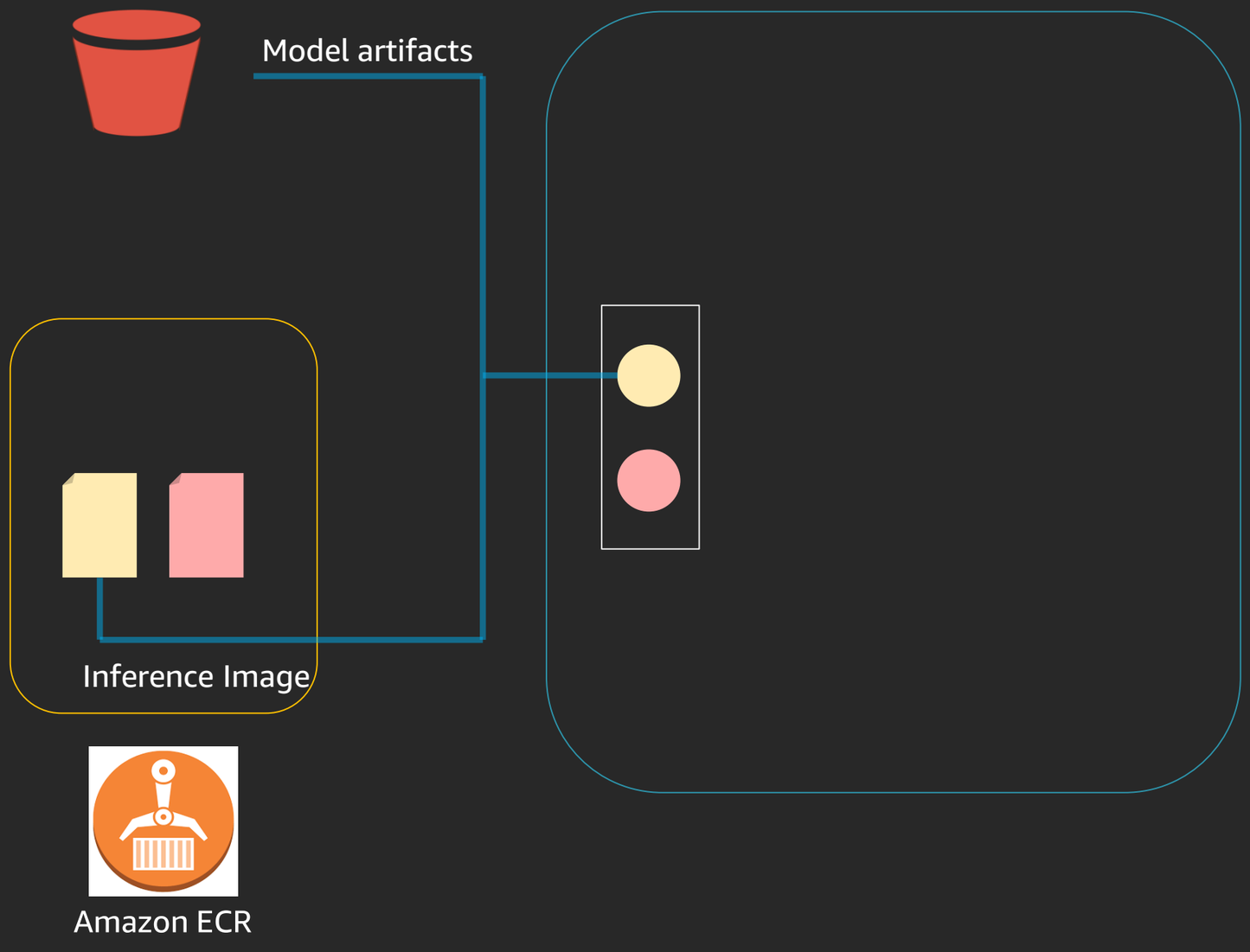

Estimator
fit
Training Environment
deploy
Model artifacts
Predictor
predict
Prediction Environment

Hosting



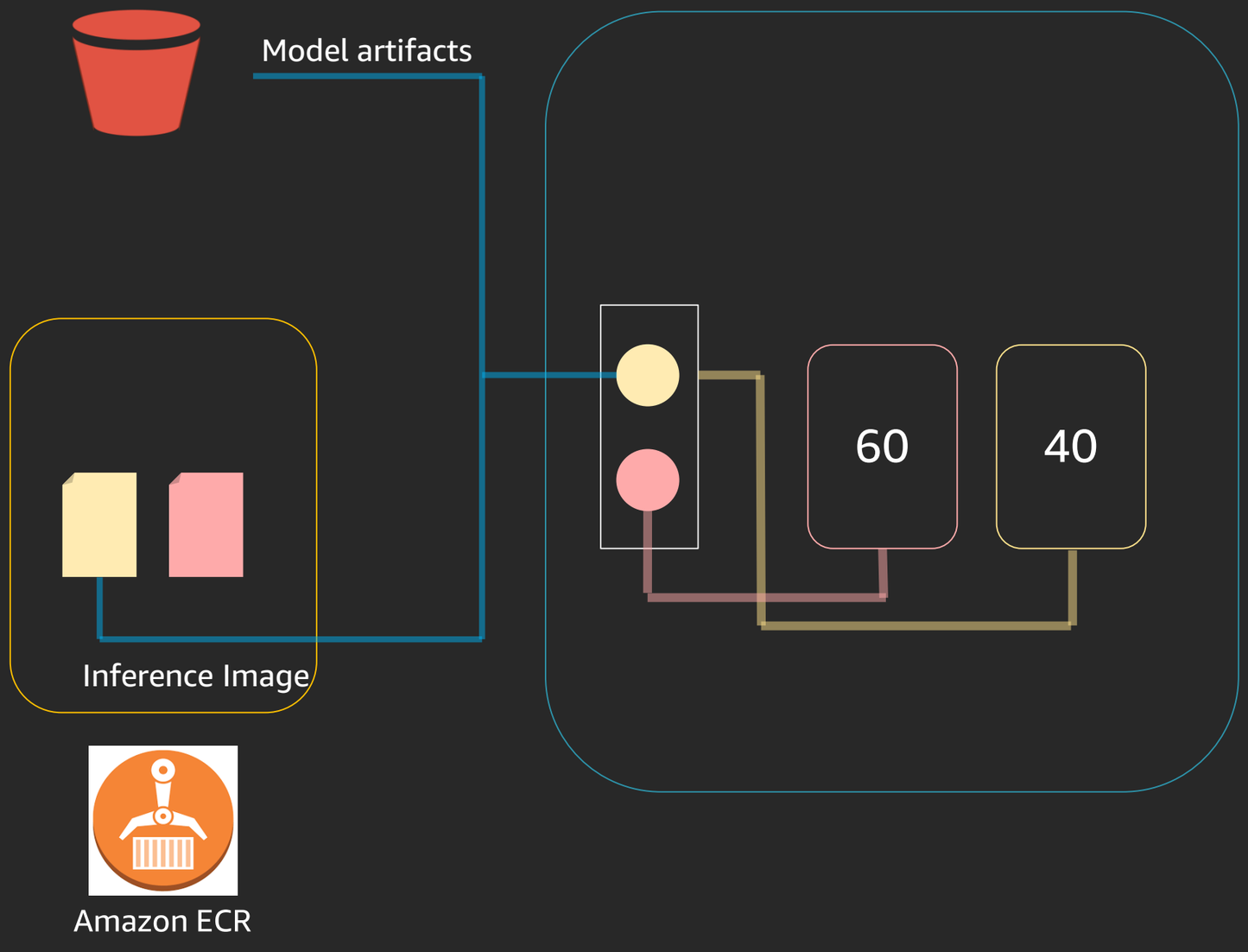
Create a model

Hosting



Create versions of a model

Hosting



Create weighted production variants

Production Variant

InstanceType

InitialInstanceCount

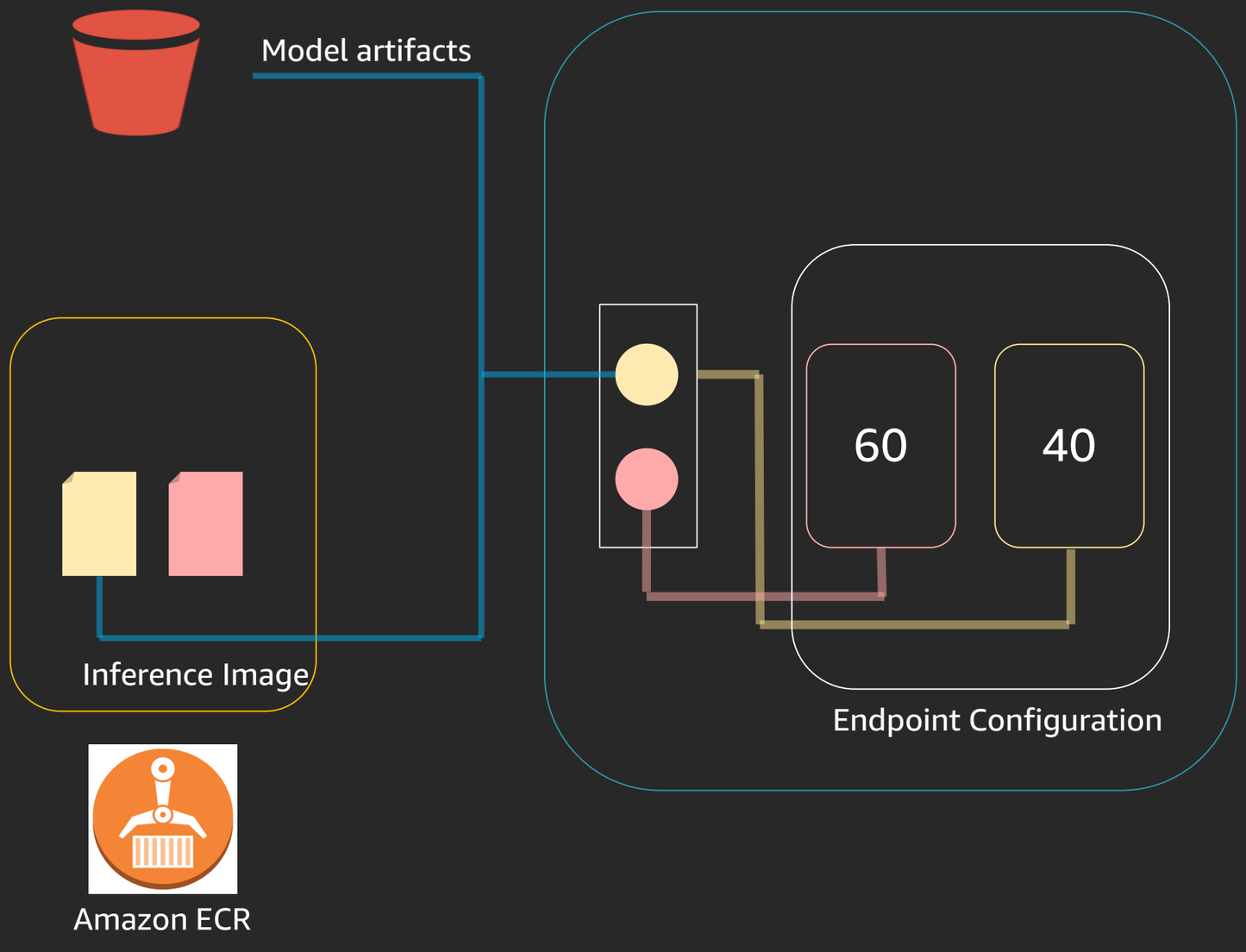
MaxInstanceCount

ModelName

VariantName

InitialVariantWeight

Hosting



Create and Endpoint from Endpoint Configuration

Production Variant

InstanceType

InitialInstanceCount

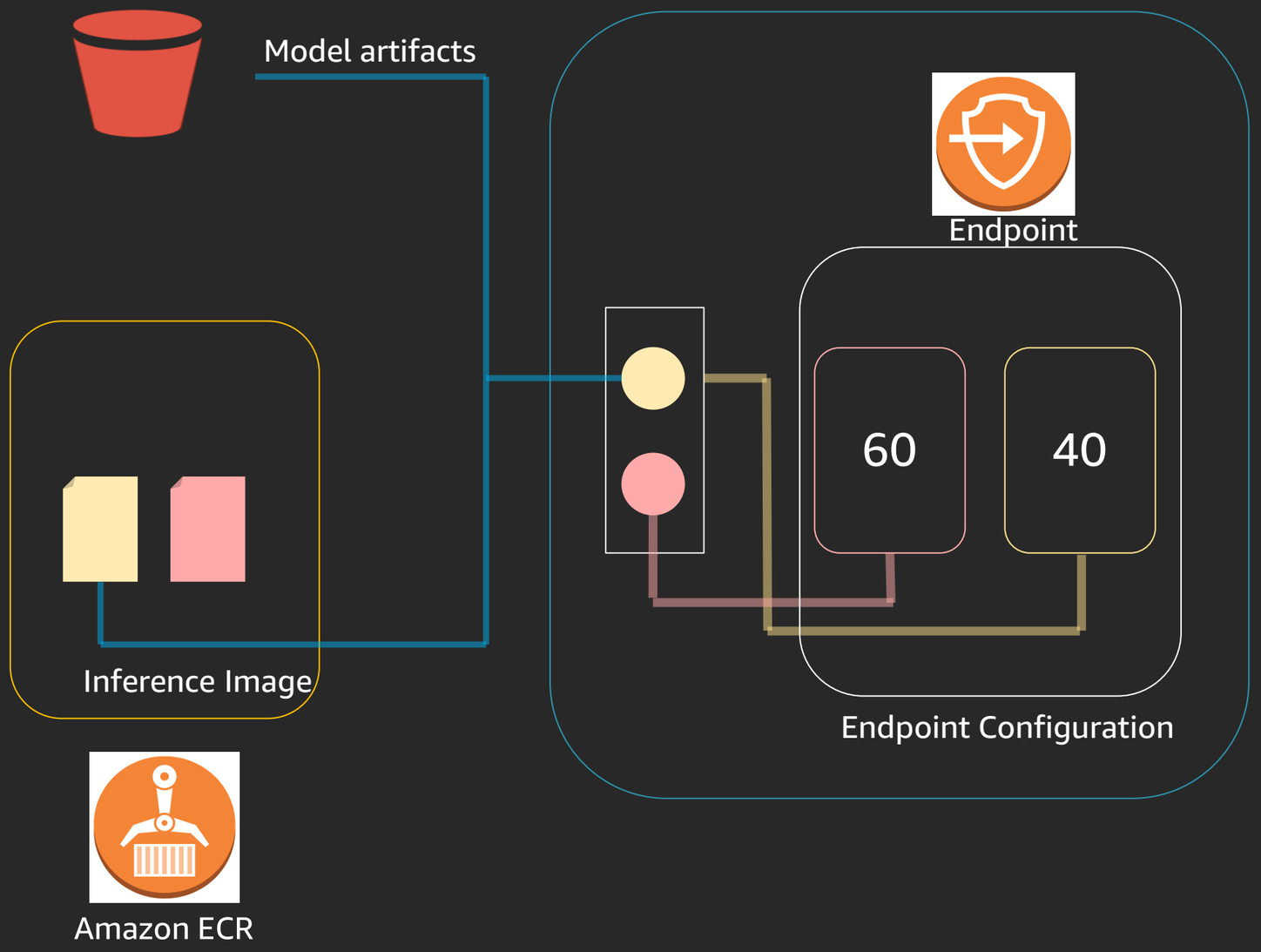
MaxInStanceCount

ModelName

VariantName

InitialVariantWeight

Hosting



Create and Endpoint Configuration one or many Production Variants

Production Variant

InstanceType

InitialInstanceCount

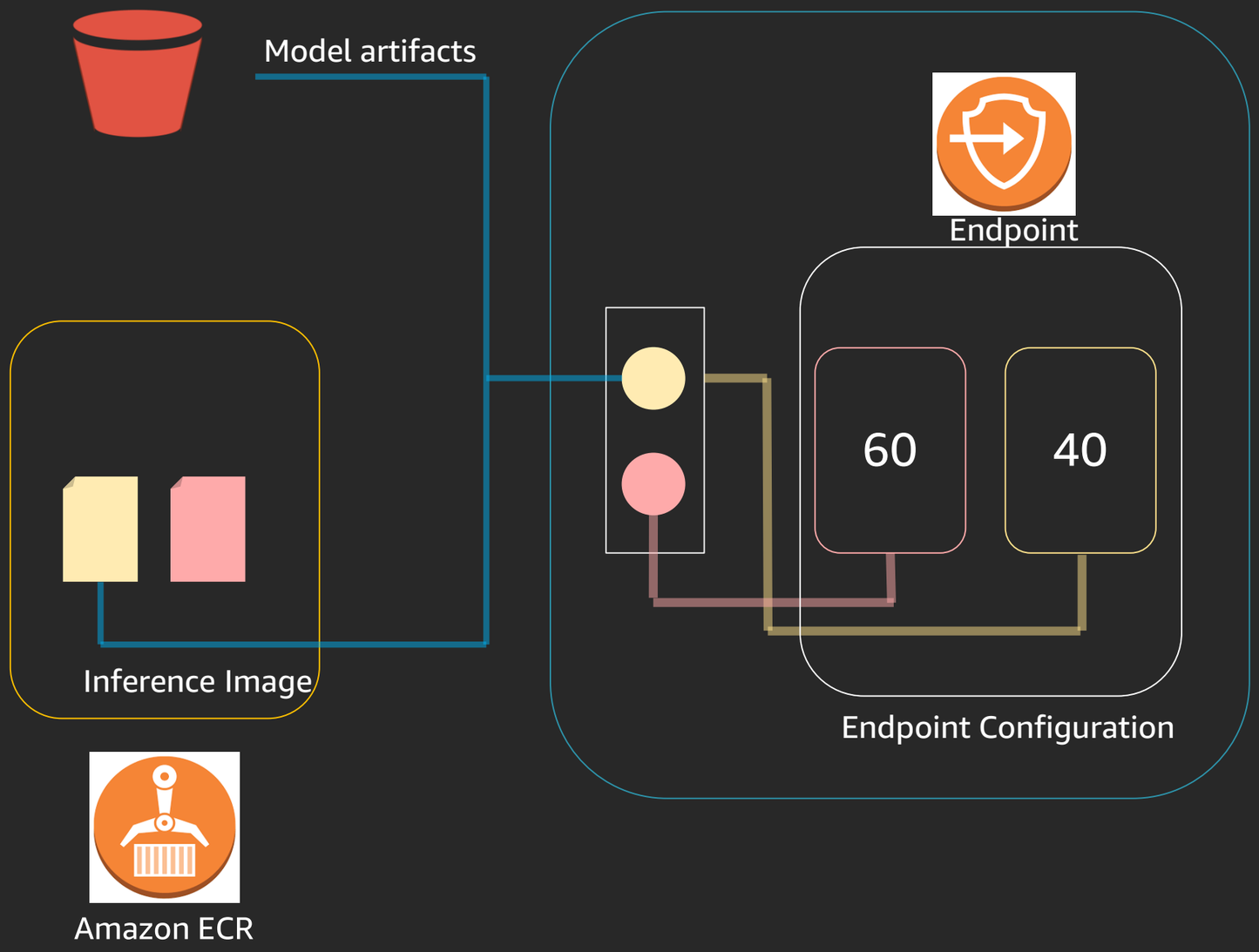
MaxInstanceCount

ModelName

VariantName

InitialVariantWeight

Hosting



One-click deployment for built-in algorithms and containers

Production Variant

InstanceType

InitialInstanceCount

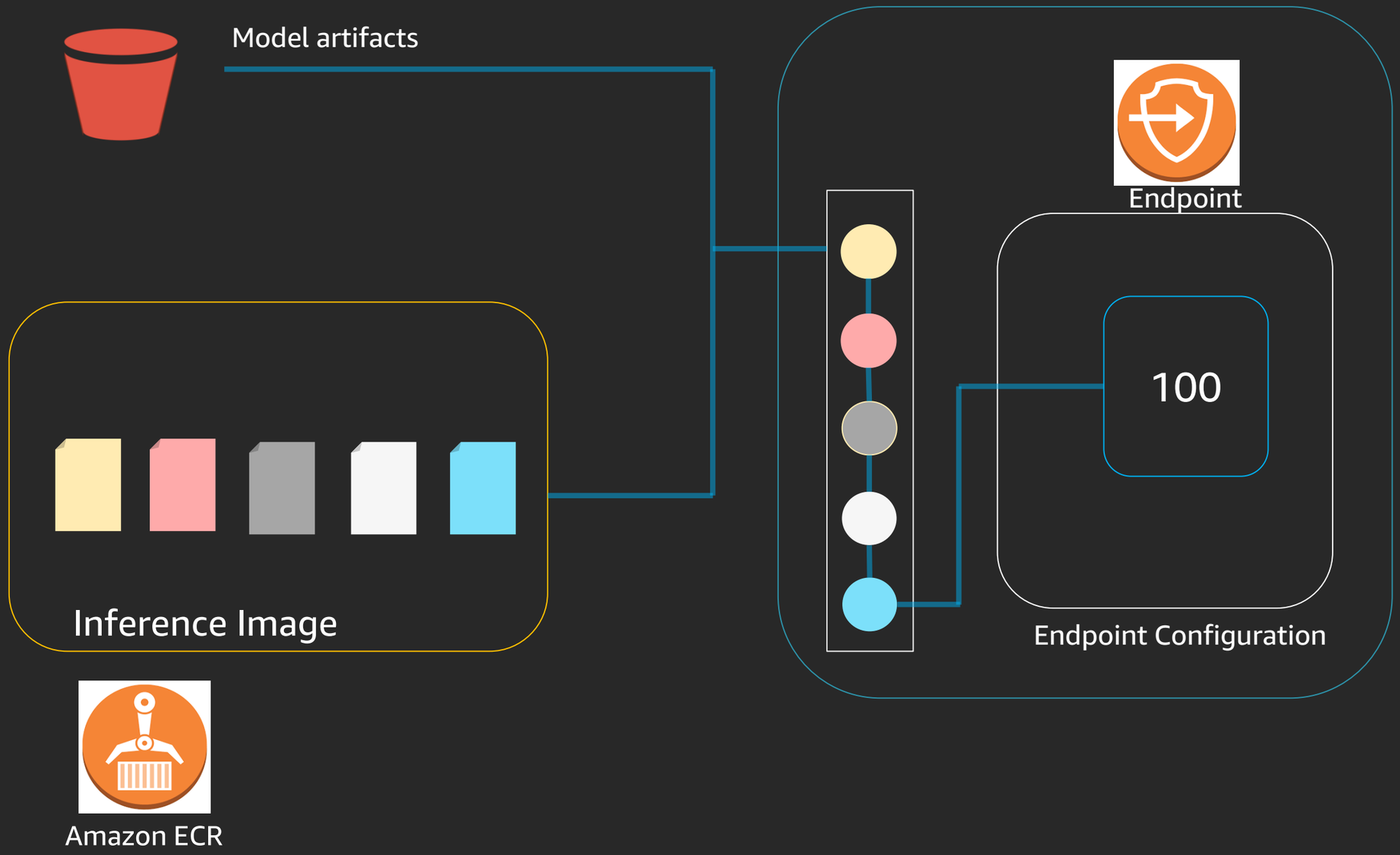
MaxInstanceCount

ModelName

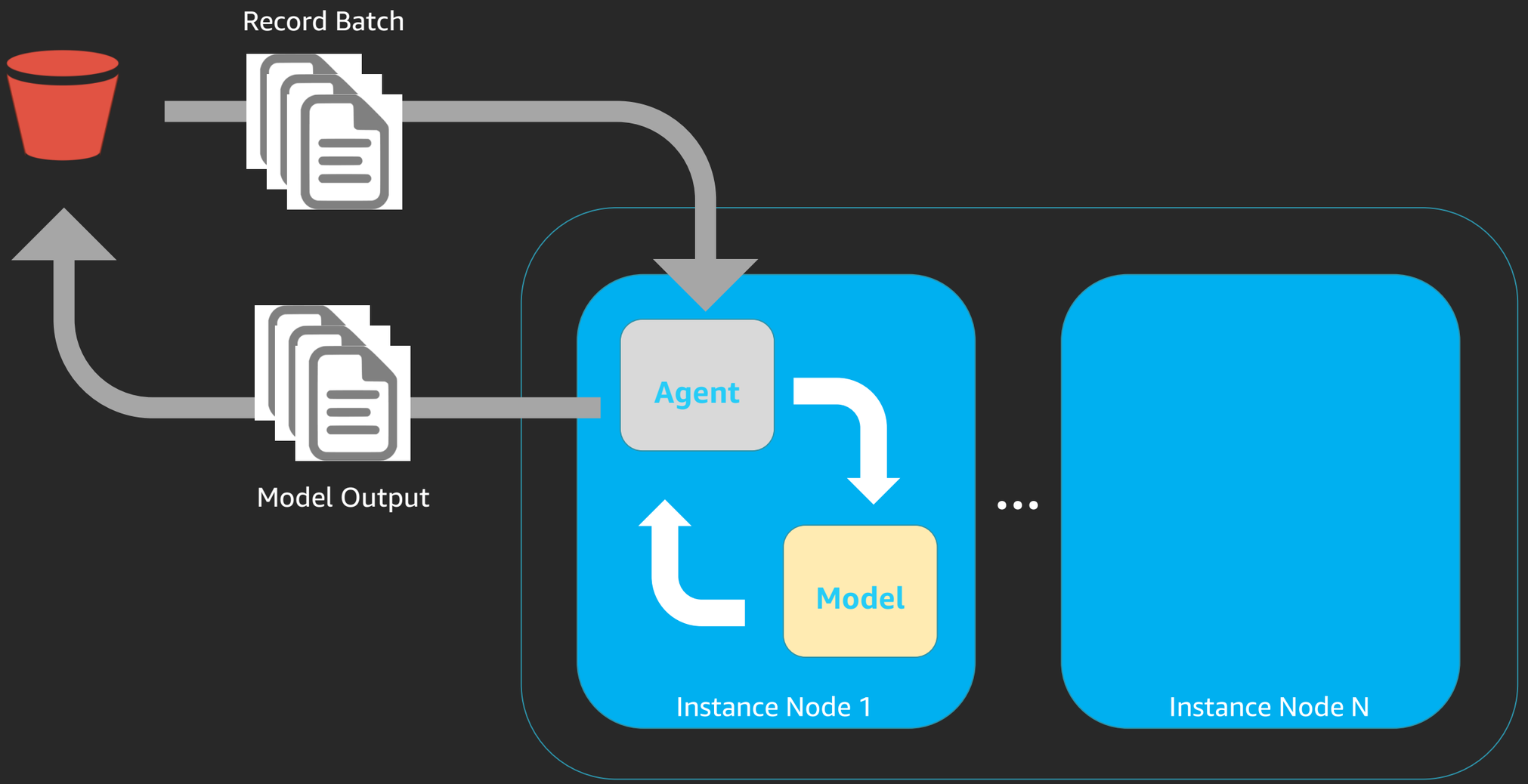
VariantName

InitialVariantWeight

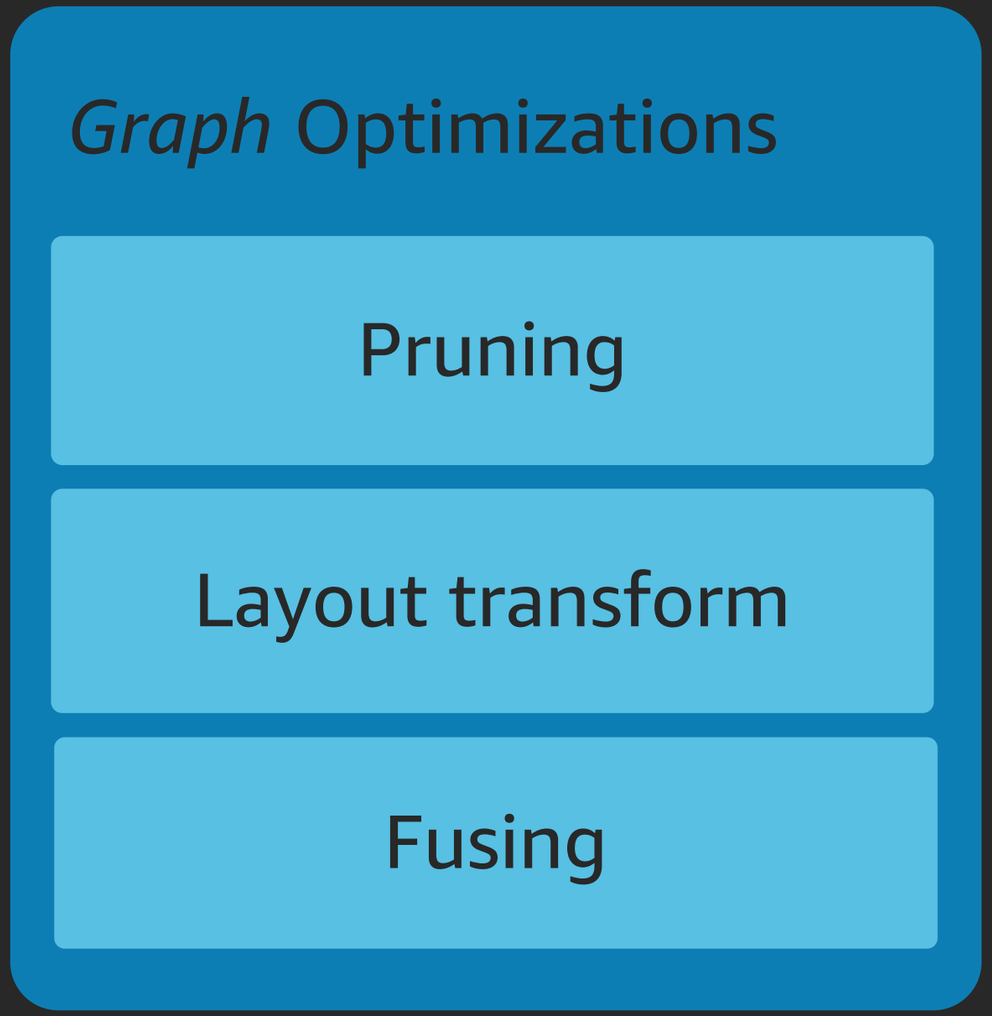
Inference pipeline

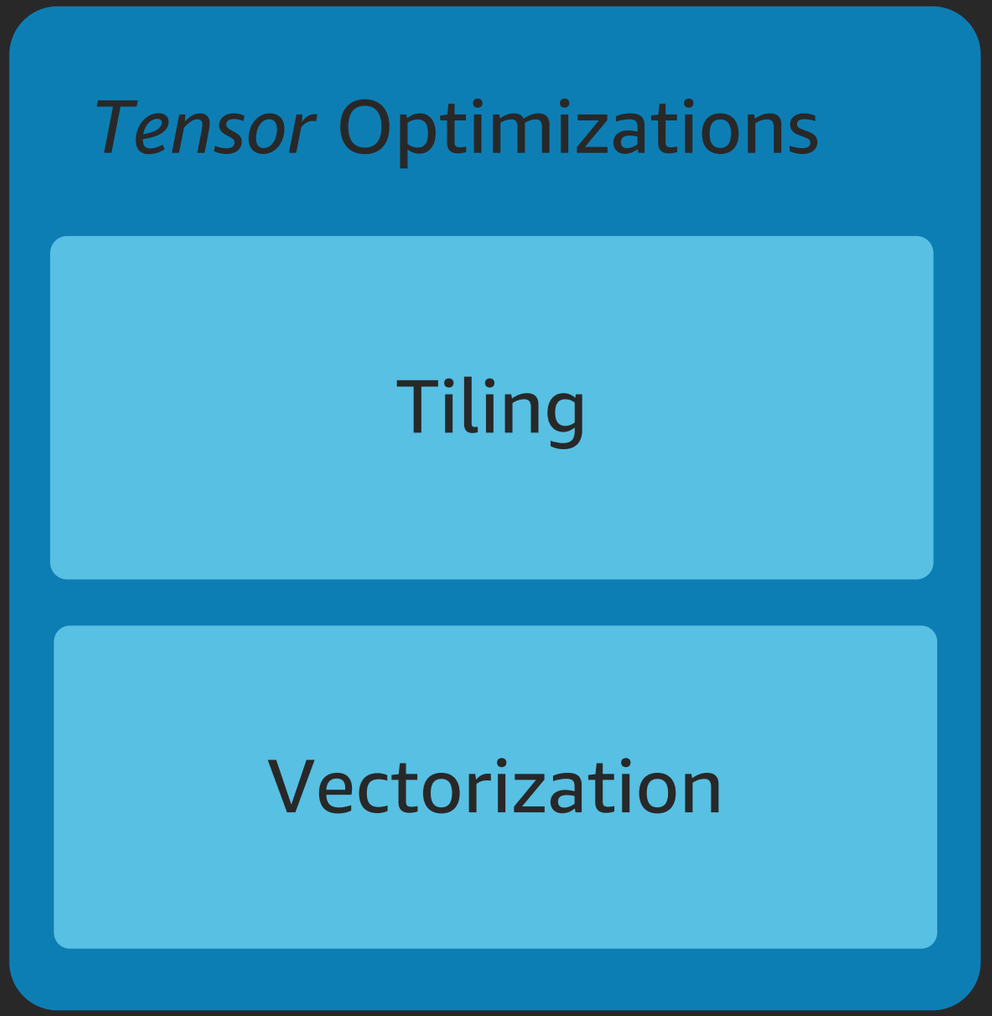


Batch Transform

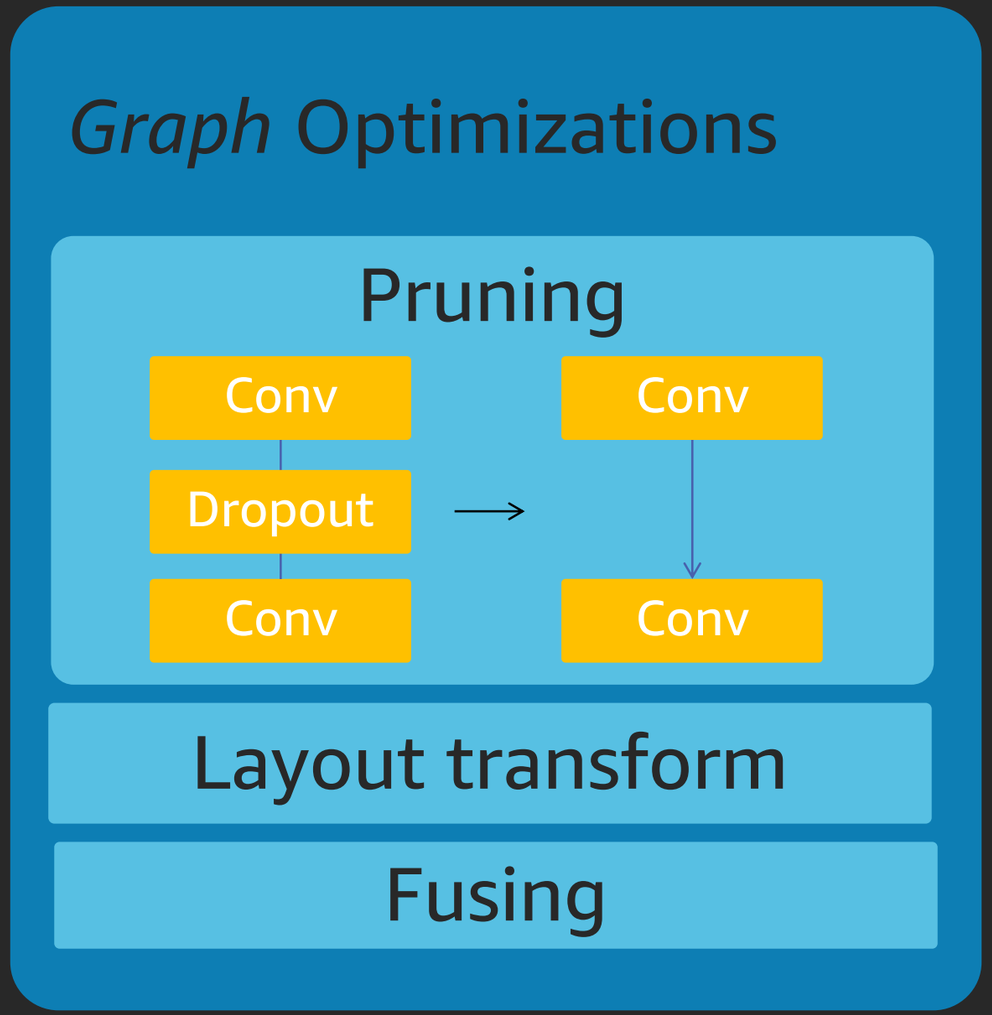


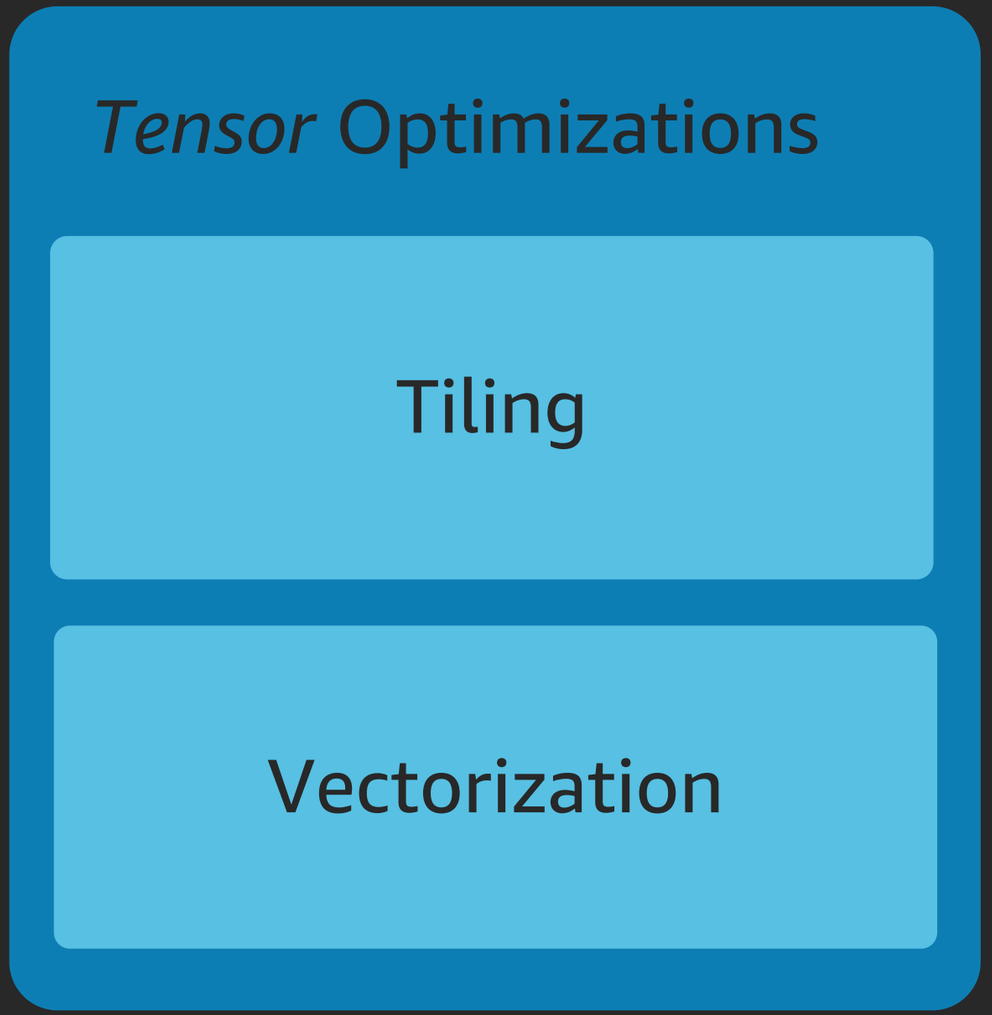
Amazon SageMaker Neo



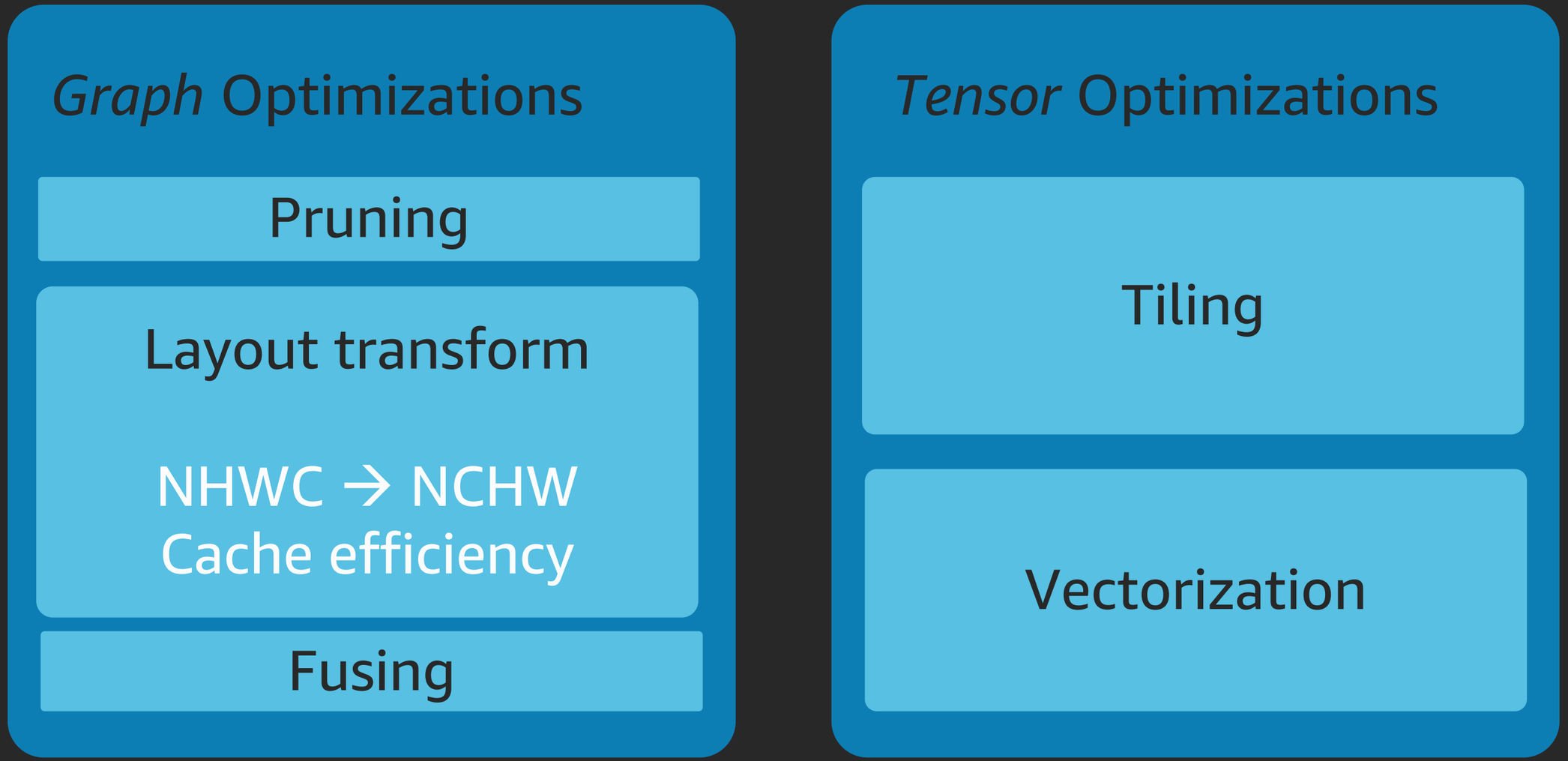


Amazon SageMaker Neo

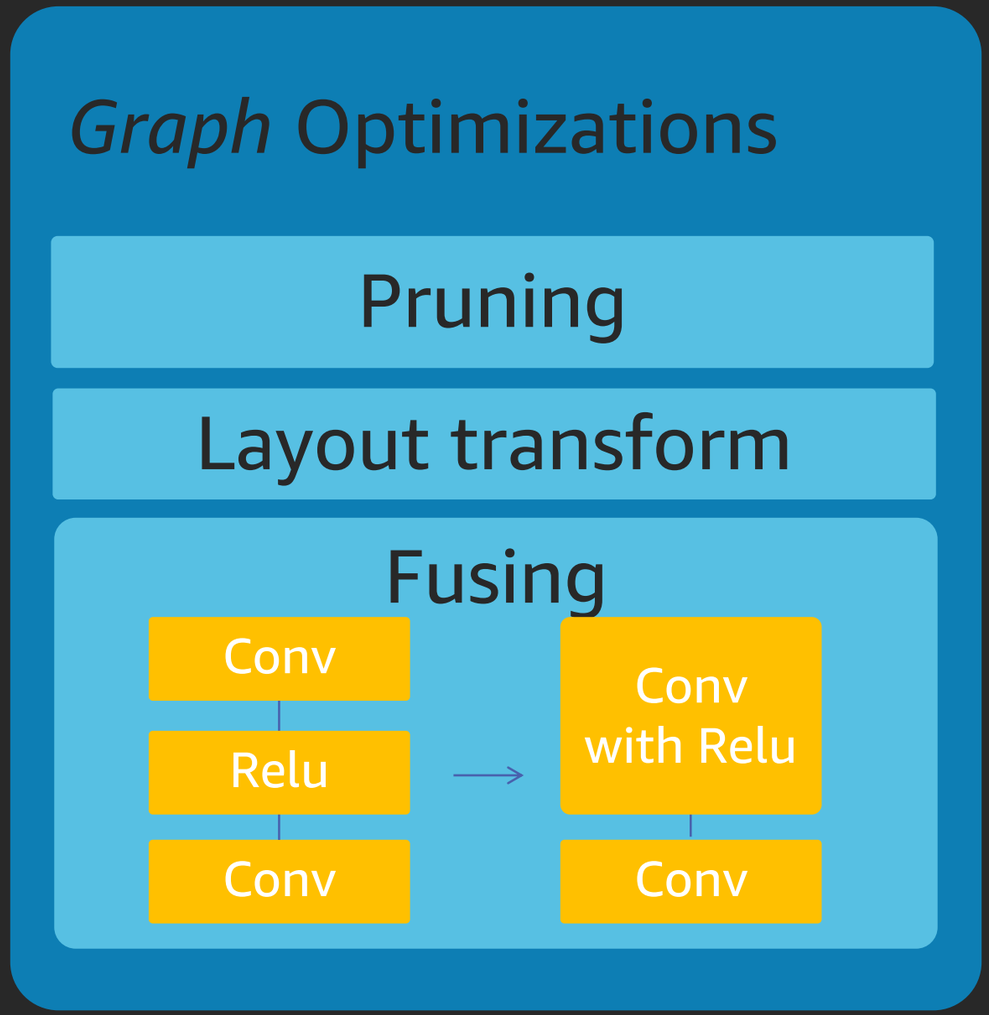


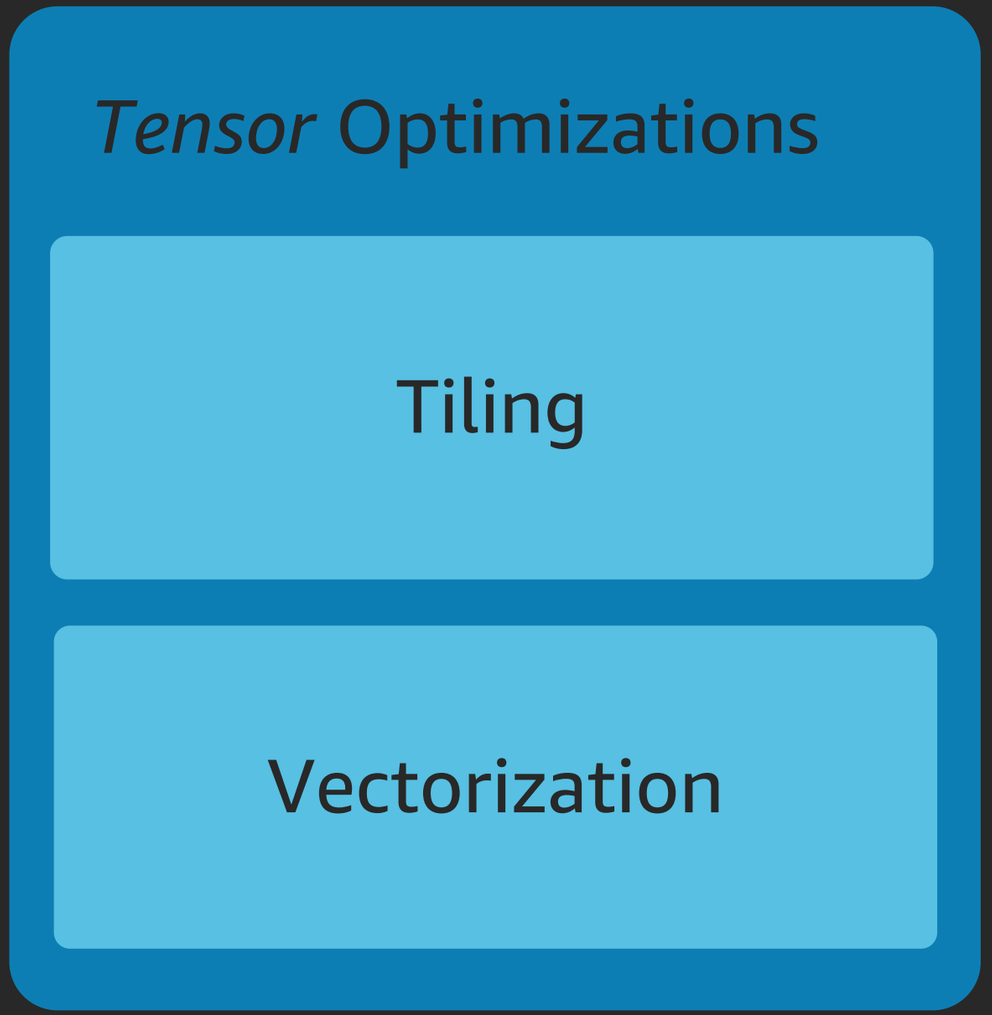


Amazon SageMaker Neo



Amazon SageMaker Neo





Amazon SageMaker Neo

Graph Optimizations

Pruning

Layout transform

Fusing

Tensor Optimizations

Tiling

N*C*H*W

↓

N*(C/16)*H*W*16

Vectorization

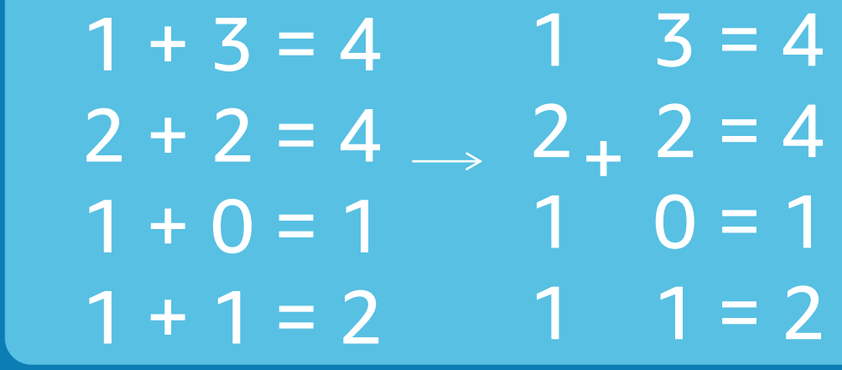
Amazon SageMaker Neo

Graph Optimizations

- Pruning
- Layout transform
- Fusing

Tensor Optimizations

- Tiling
- Vectorization



Thank you!

AWS re:Invent

aws

Please complete the session survey in the mobile app.