

SEOUL

19.09.26

# DEV DAY



© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

모두를 위한 컴퓨터 비전 딥러닝 툴킷, **GluonCV** 따라하기

## 2-1. MXNet / Gluon Overview

김무현 데이터 사이언티스트  
Amazon Machine Learning Solutions Lab



# Apache MXNet Key Benefits



Flexible



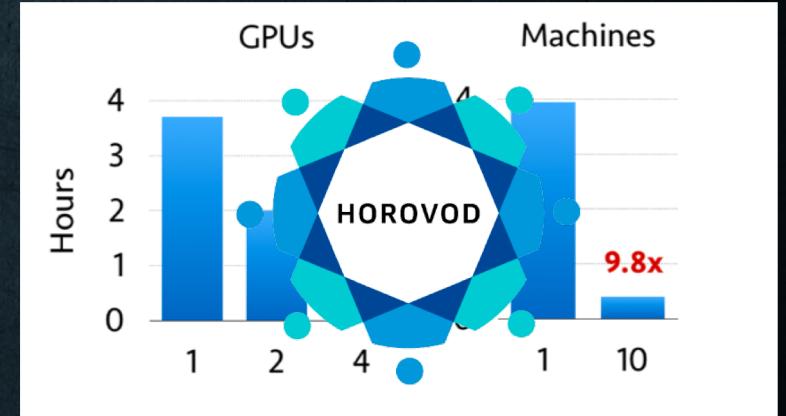
Debuggable



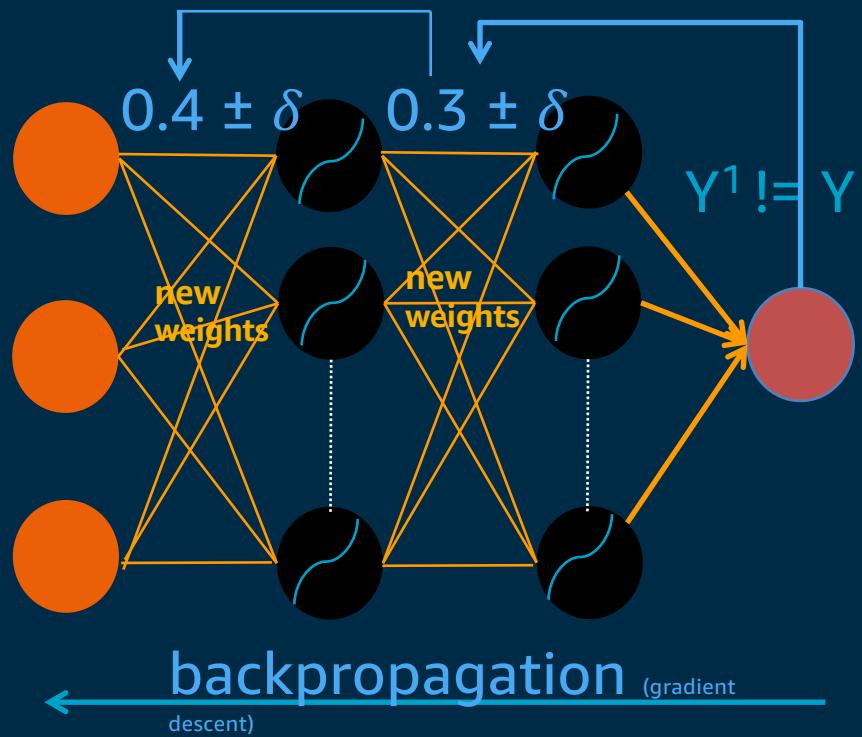
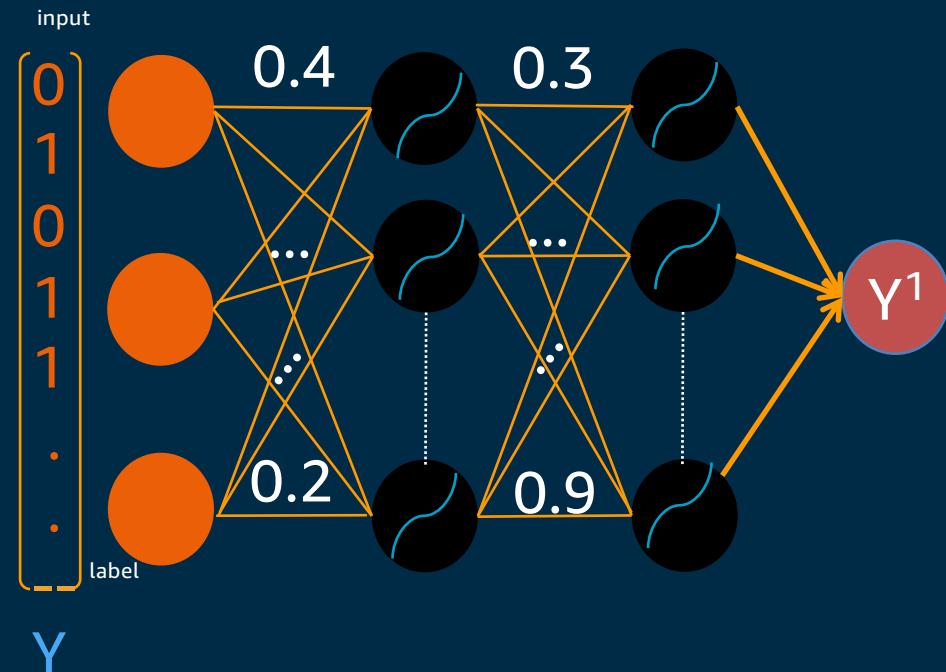
Scalable



Mix imperative and symbolic paradigms



# 딥러닝에서 “학습”이란?

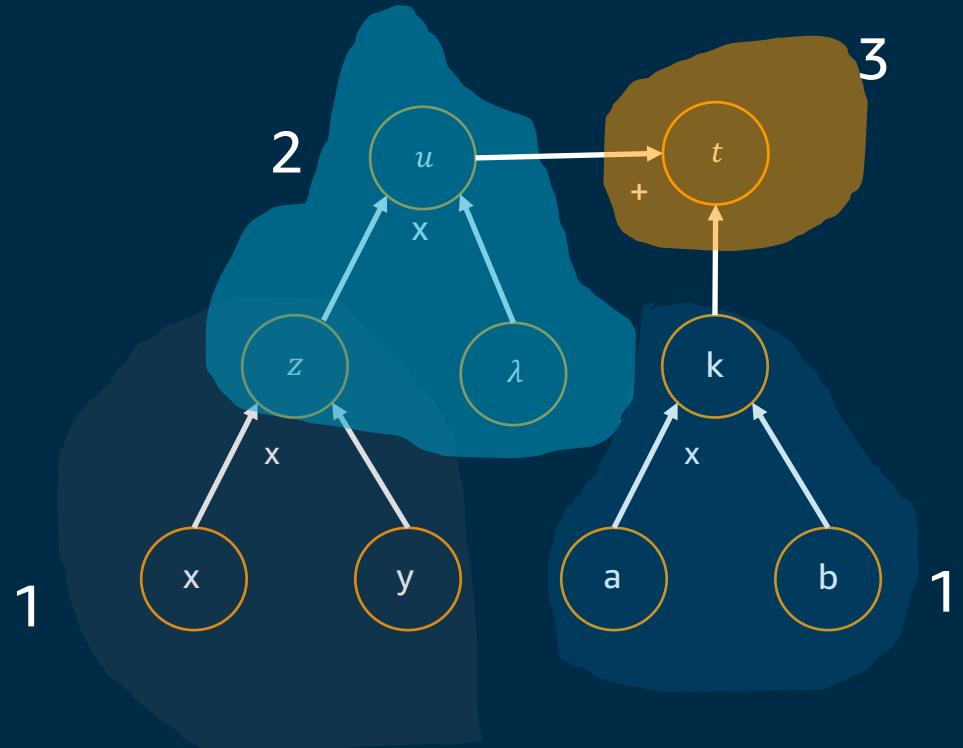


# Computational dependency graph

$$z = x \cdot y$$

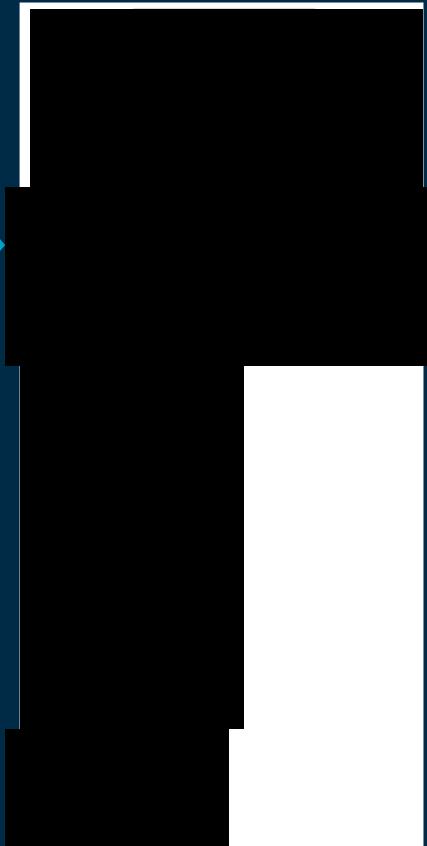
$$k = a \cdot b$$

$$t = \lambda z + k$$



# MXNet computational dependency graph

```
net = mx.sym.Variable('data')  
  
net = mx.sym.FullyConnected(net, name='fc1', num_hidden=128)  
  
net = mx.sym.Activation(net, name='relu1', act_type="relu")  
  
net = mx.sym.FullyConnected(net, name='fc2', num_hidden=10)  
  
net = mx.sym.SoftmaxOutput(net, name='softmax')
```



# Training

```
import logging

logging.getLogger().setLevel(logging.DEBUG) # logging to stdout

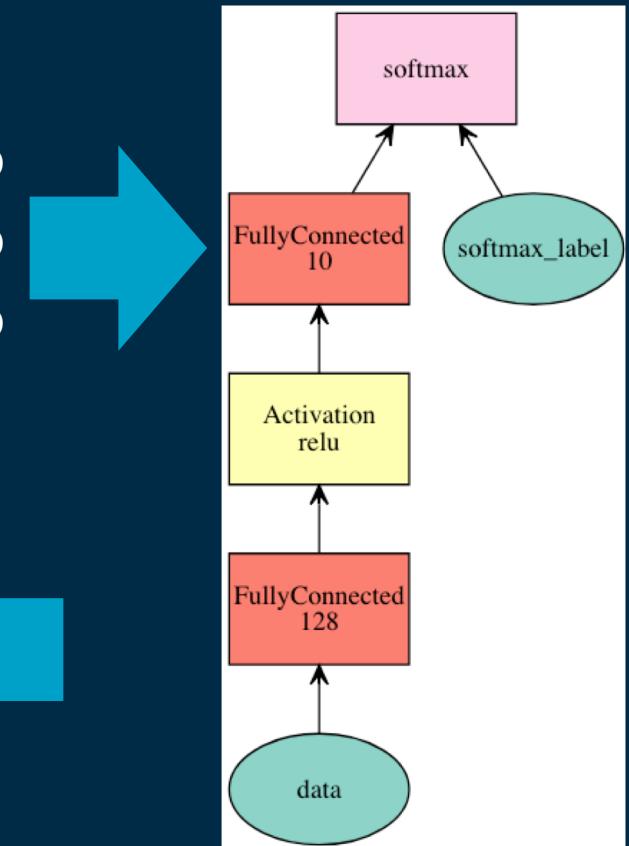
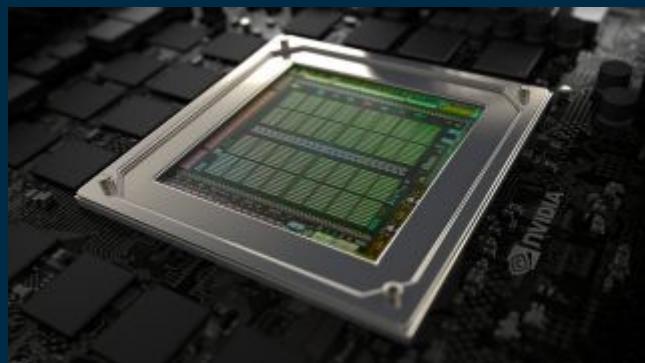
# create a trainable module on compute context

mlp_model = mx.mod.Module(symbol=mlp, context=ctx)

mlp_model.fit(train_iter,
              eval_data=val_iter,
              optimizer='sgd',
              optimizer_params={'learning_rate':0.1},
              eval_metric='acc',
              batch_end_callback = mx.callback.Speedometer(batch_size, 100),
              num_epoch=10)
```

# MXNet computational dependency graph

```
net = mx.sym.Variable('data')  
  
net = mx.sym.FullyConnected(net, name='fc1', num_hidden=64)  
  
net = mx.sym.Activation(net, name='relu1', act_type="relu")  
  
net = mx.sym.FullyConnected(net, name='fc2', num_hidden=10)  
  
net = mx.sym.SoftmaxOutput(net, name='softmax')
```



# Multi-language support

Java

Perl

Julia

Clojure

Python

Scala

C++

R

Frontend

Backend

C++

**DEV DAY**

# Developer Productivity



**Simple, Easy-to-Understand Code**

**Flexible, Imperative Structure**

**Dynamic Graphs**

**High Performance**



# Network definition in Gluon

```
net = gluon.nn.HybridSequential()  
with net.name_scope():  
    net.add(gluon.nn.Dense(units=64, activation='relu'))  
    net.add(gluon.nn.Dense(units=10))  
  
softmax_cross_entropy = gluon.loss.SoftmaxCrossEntropyLoss()  
  
net.initialize(mx.init.Xavier(magnitude=2.24), ctx=ctx, force_rei  
nit=True)  
  
trainer = gluon.Trainer(net.collect_params(), 'sgd', {'learning_r  
ate': 0.02})
```



© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.



# Training in Gluon

```
smoothing_constant = .01
for e in range(10):
    cumulative_loss = 0
    for i, (data, label) in enumerate(train_data):
        data = data.as_in_context(model_ctx).reshape((-1, 784))
        label = label.as_in_context(model_ctx)
        with autograd.record():
            output = net(data)
            loss = softmax_cross_entropy(output, label)
        loss.backward()
        trainer.step(data.shape[0])
```

# Imperative API



Debuggable

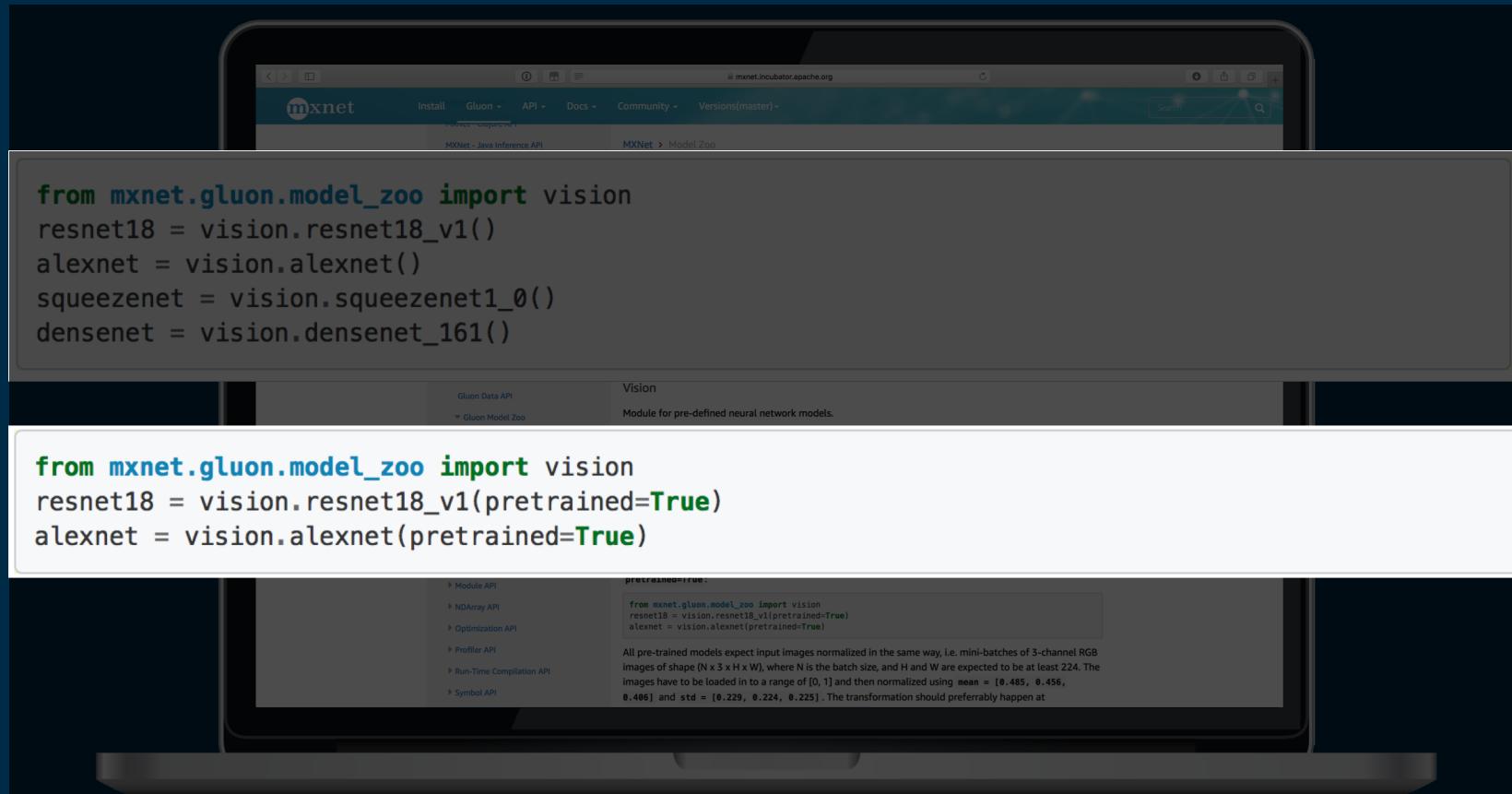


Flexible



Scalable

# Gluon Model Zoo



# GluonCV, GluonNLP and GluonTS Toolkits

- Provides reference implementation of state-of-the-art work as described in research literature
- Over 300 pre-trained models allow you to begin immediately
- Provides common public dataset, dataset APIs, and training/benchmarking script
- GluonCV toolkit: <https://gluon-cv.mxnet.io/>
- GluonNLP toolkit: <https://gluon-nlp.mxnet.io/>
- GluonTS toolkit: <https://gluon-ts.mxnet.io/>

# GluonCV: a deep learning toolkit for computer vision

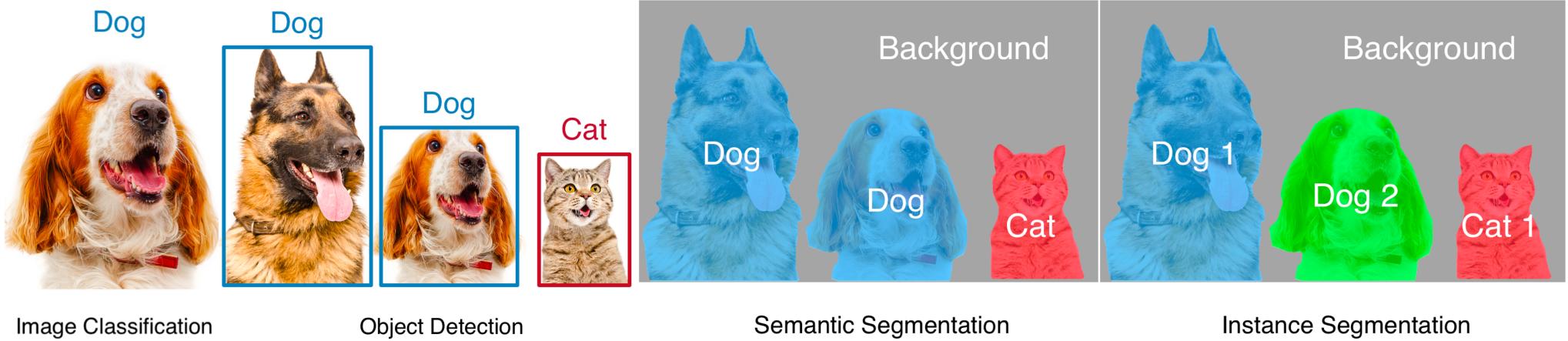


Image Classification

Object Detection

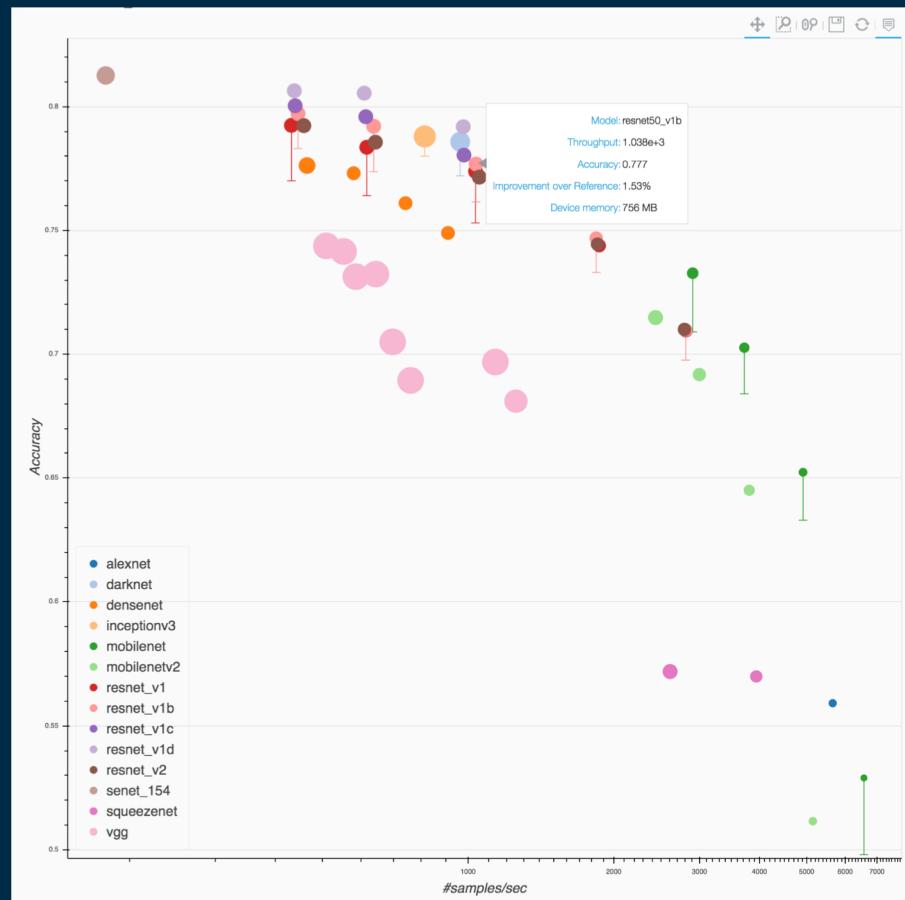
Semantic Segmentation

Instance Segmentation

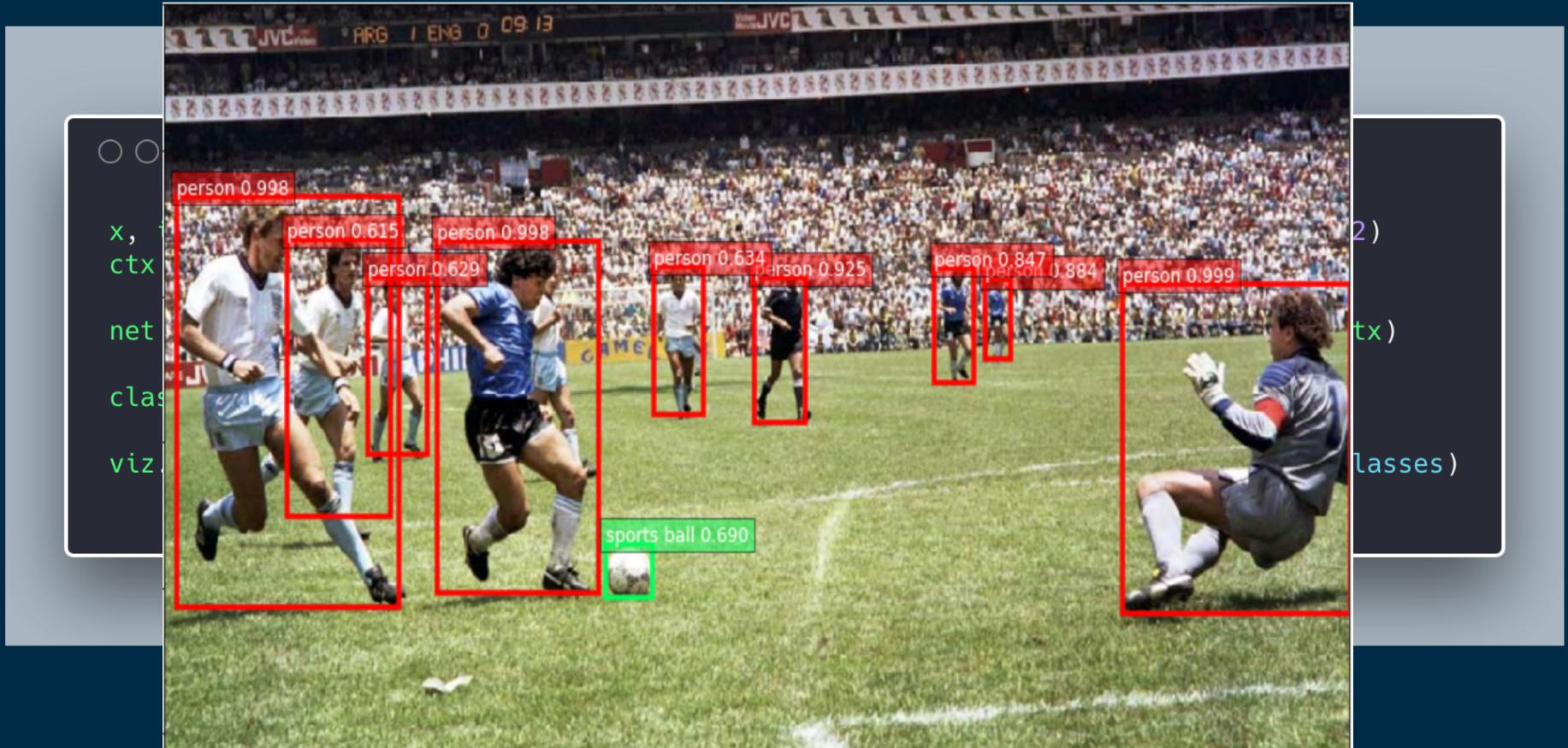
50+ Pre-trained models, with training scripts, datasets, tutorials

<https://gluon-cv.mxnet.io>

# GluonCV pre-trained models



# GluonCV example code



# GluonNLP: a deep learning toolkit for natural language processing



- 300+ word embedding pre-trained models
- 5 language models
- Neural Machine Translation (Google NMT, Transformer)
- Flexible data pipeline tools
- Public datasets
- NLP examples, e.g. sentiment analysis

# GluonTS

Gluon toolkit for probabilistic time series modeling, focusing on deep learning-based models.

- utilities for loading and iterating over time series datasets,
  - state of the art models ready to be trained,
  - building blocks to define your own models and quickly experiment with different solutions.
- 
- <https://gluon-ts.mxnet.io/>
  - <https://github.com/awslabs/gluon-ts>

DEV DAY

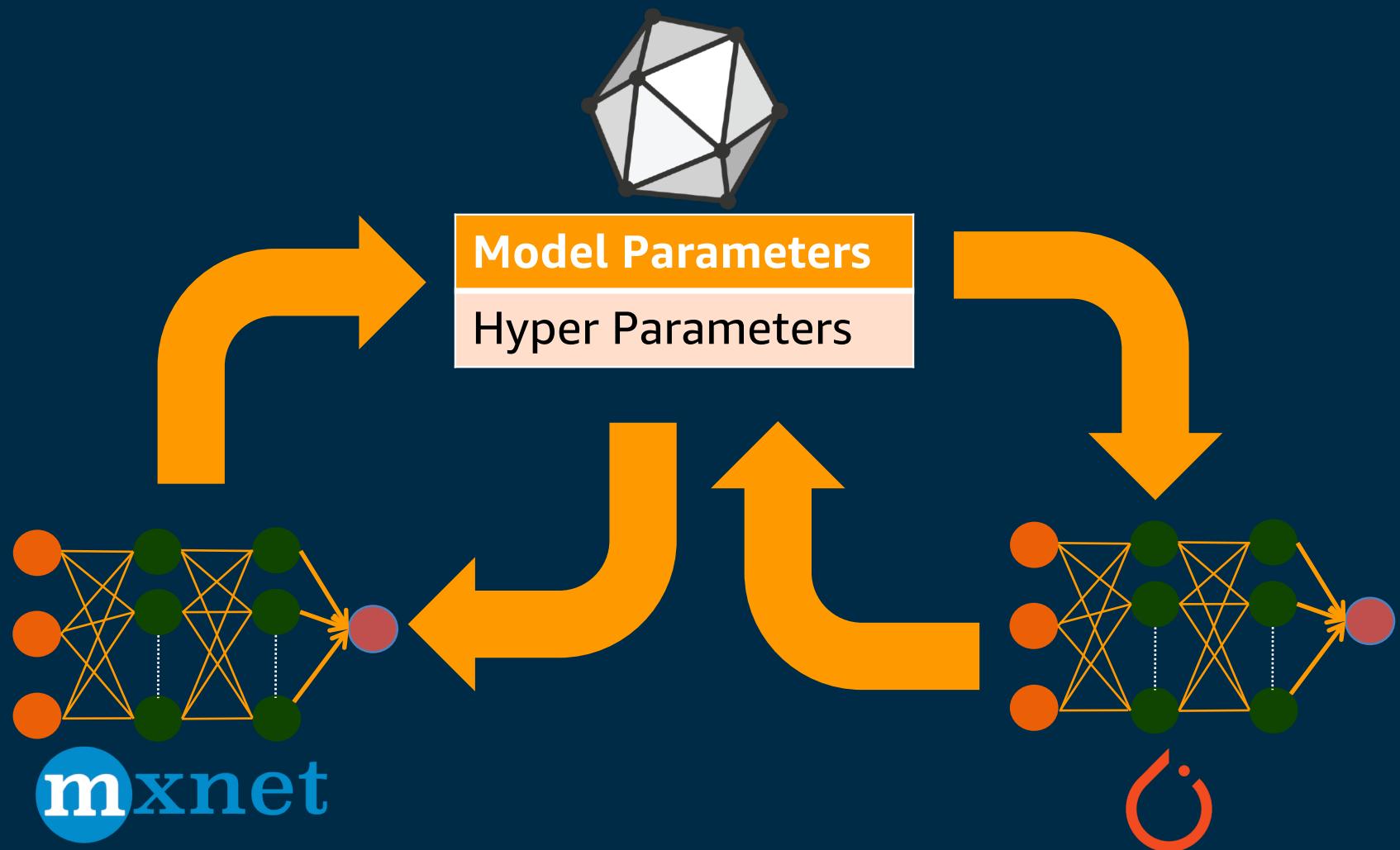
# Interoperability





# ONNX

# Portability with ONNX



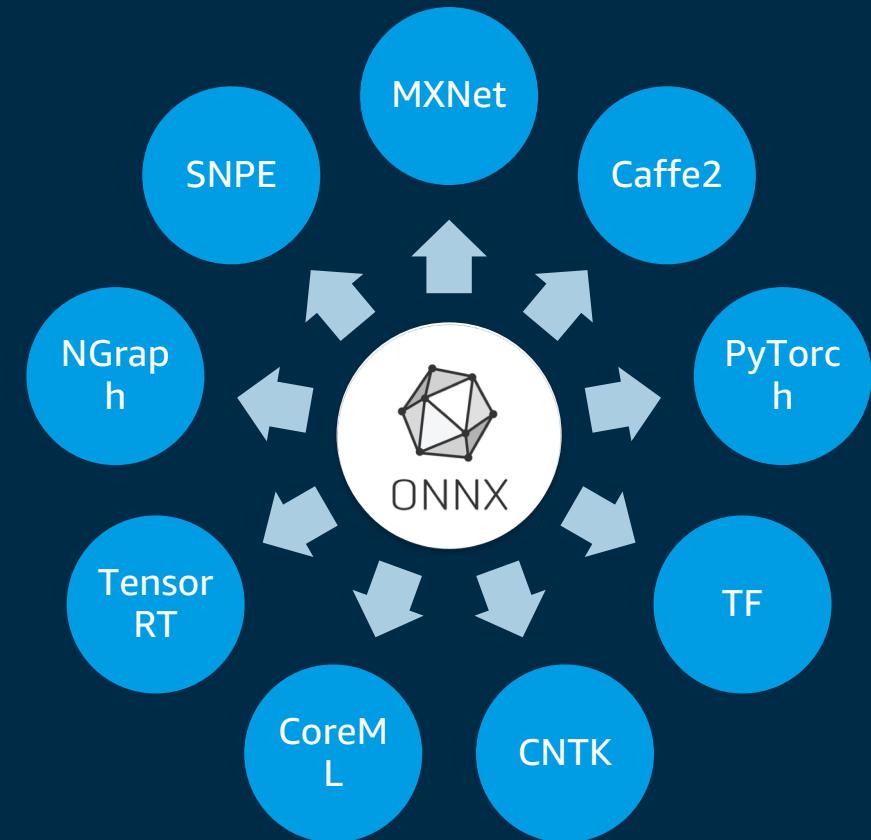
# Open Neural Network eXchange

Many Frameworks

Many Platforms

ONNX: Common  
Intermediate Representation (IR)

- Open source
- Community driven
- Simple



# Supported tools

Frameworks:



Converters:



# Supported runtimes



Qualcomm

BITMAIN

Tencent



SYNOPSYS®



CEVA®



habana

# Supported compilers and visualizers



**DEV DAY**

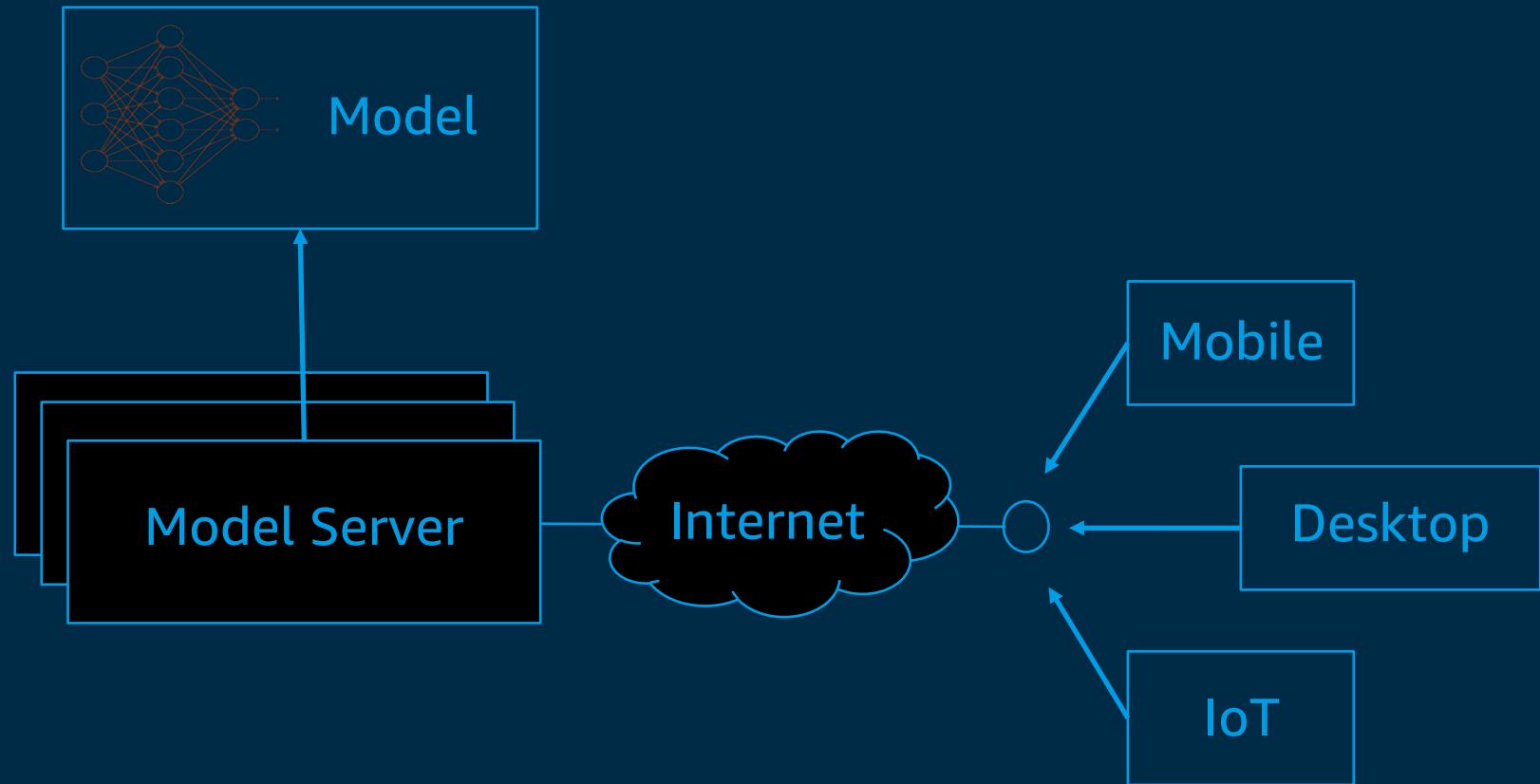
# Inference



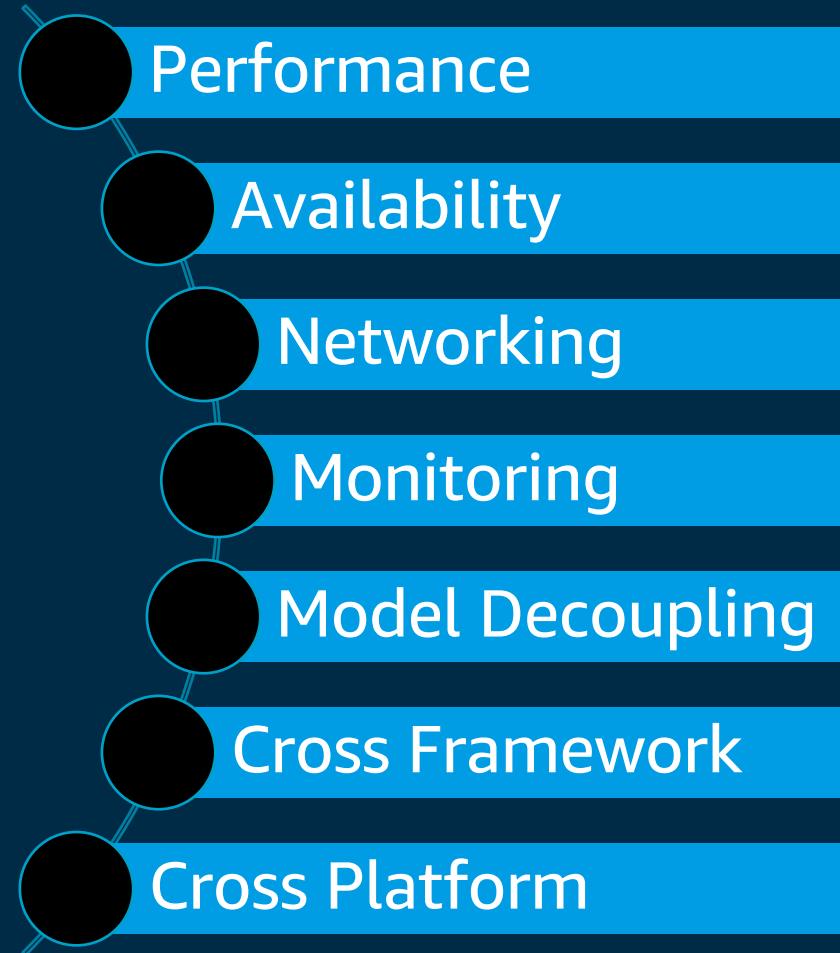


# Model Server

# What does a model server look like?

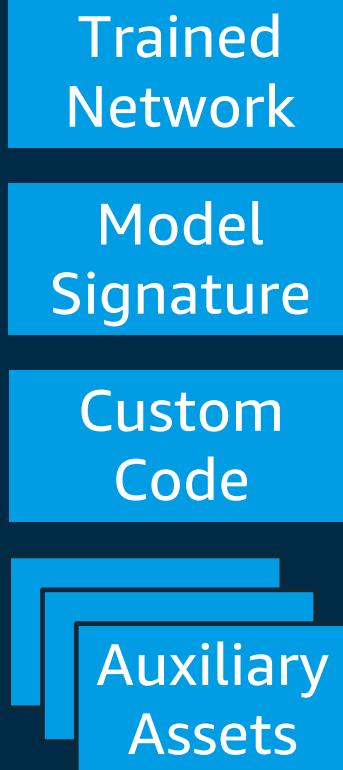


# The Undifferentiated Heavy Lifting of Model Serving

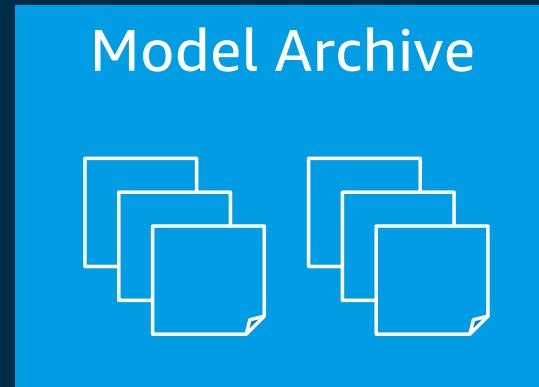




# Model Archive



Model Export CLI



# Amazon SageMaker Neo

Train once, run anywhere with 2x the performance



Get accuracy  
and performance



Automatic  
optimization



Broad framework  
support



Broad hardware  
support

## KEY FEATURES

Open-source Neo-AI device runtime and compiler under the Apache software license;  
1/10<sup>th</sup> the size of original frameworks

# Deep Learning acceleration



CUDA & CuDNN

`pip install mxnet-cu92`

TensorRT

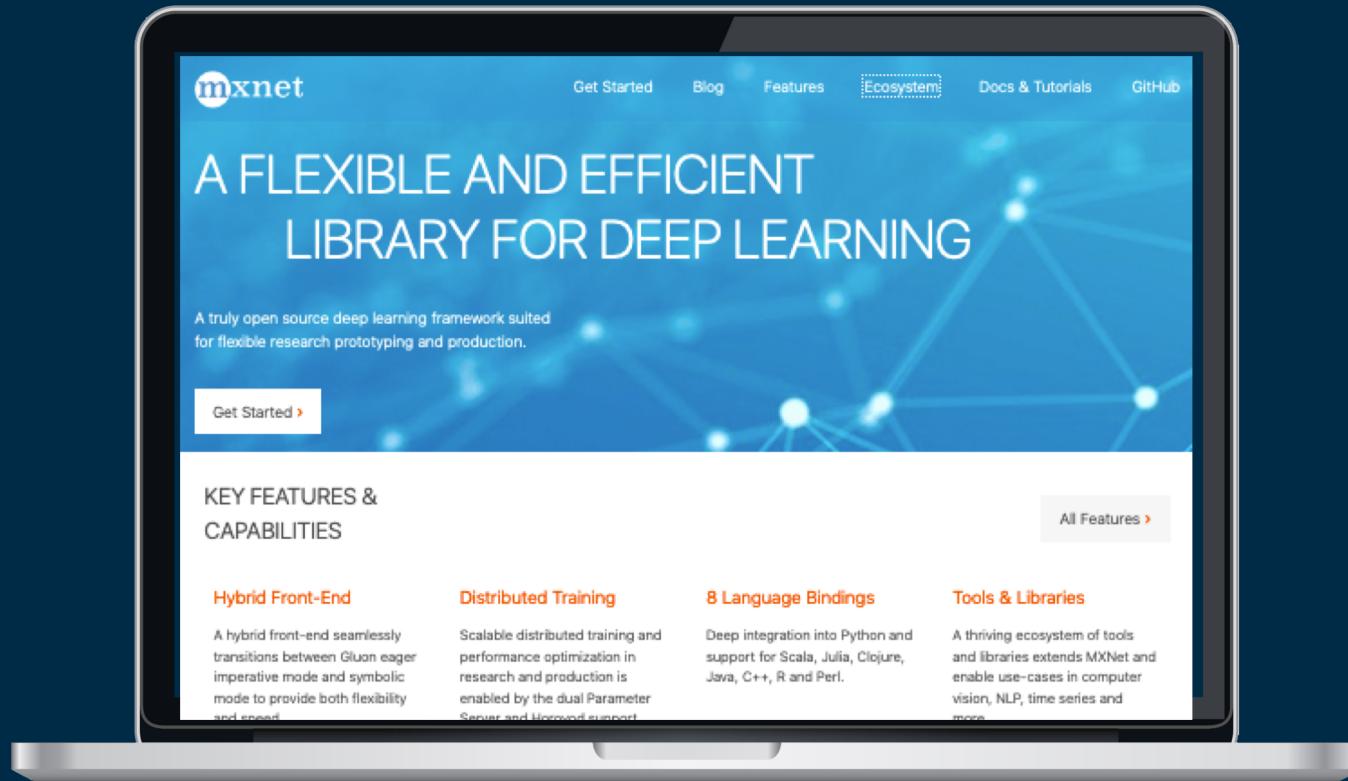
`pip install mxnet-tensorrt-cu92`

MKL, MKLML & MKLDNN

e.g. `pip install mxnet-mkl`



# Project home page



<https://mxnet.apache.org>

# Ecosystem

## Coach RL



Coach is a python reinforcement learning framework containing implementation of many state-of-the-art algorithms, it supports MXNet as a back-end

## Keras-MXNet



Keras-MXNet provides a backend support for the widely used high level API Keras.

## Sockeye

Sockeye is a sequence-to-sequence framework for Neural Machine Translation based on Apache MXNet Incubating. It implements state-of-the-art encoder-decoder architectures.

## Deep Graph Library

DGL is a Python package dedicated to deep learning on graphs supporting MXNet as a backend.

## MXBoard

MXBoard provides a set of APIs for logging MXNet data for visualization in TensorBoard.

## TensorLy



TensorLy is a high level API for tensor methods and deep tensorized neural networks in Python that aims to make tensor learning simple.

## GluonFR

Community-driven toolkit for Face Recognition and Face Detection

## MXFusion



MXFusion is a modular deep probabilistic programming library. It lets you use state-of-the-art inference techniques for specialized probabilistic models.

## TVM



TVM is an open deep learning compiler stack for CPUs, GPUs, and specialized accelerators. It supports a number of framework including MXNet.

## InsightFace

State-of-the-art face detection and face recognition repository, including ArcFace loss and RetinaFace implementation

## MXNet Model Server

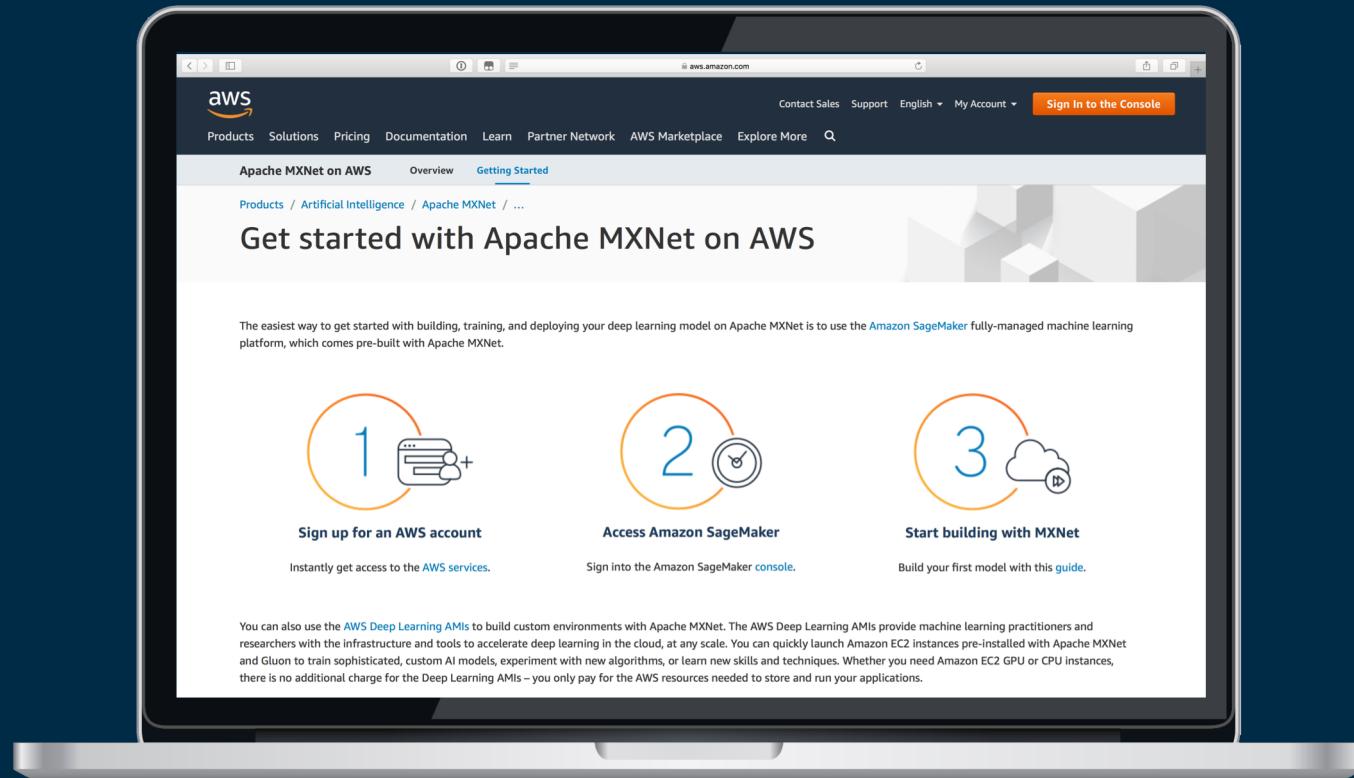
Model Server for Apache MXNet (MMS) is a flexible and easy to use tool for serving deep learning models exported from MXNet or the Open Neural Network Exchange (ONNX).



## Xfer

Xfer is a library that allows quick and easy transfer of knowledge stored in deep neural networks implemented in MXNet.

# Getting started with Apache MXNet on AWS



<https://aws.amazon.com/mxnet/get-started/>

# Using Apache MXNet with AWS ML services

- Amazon SageMaker: [aws.amazon.com/sagemaker](https://aws.amazon.com/sagemaker)
- Amazon SageMaker Neo: [aws.amazon.com/sagemaker/neo](https://aws.amazon.com/sagemaker/neo)
- Amazon SageMaker Reinforcement Learning:  
[aws.amazon.com/about-aws/whats-new/2018/11/amazon-sagemaker-announces-support-for-reinforcement-learning/](https://aws.amazon.com/about-aws/whats-new/2018/11/amazon-sagemaker-announces-support-for-reinforcement-learning/)
- Amazon Elastic Inference: [aws.amazon.com/machine-learning/elastic-inference](https://aws.amazon.com/machine-learning/elastic-inference)
- AWS IoT Greengrass ML Inference: [aws.amazon.com/greengrass/ml](https://aws.amazon.com/greengrass/ml)
- Dynamic Training with Apache MXNet on AWS:  
<https://aws.amazon.com/about-aws/whats-new/2018/11/introducing-dynamic-training-with-apache-mxnet/>

# Staying in touch

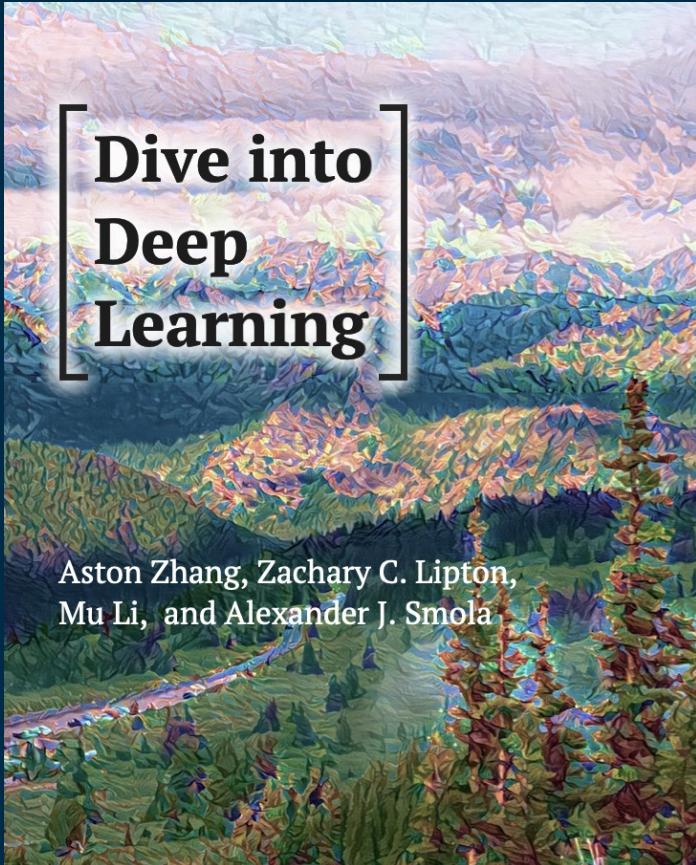
- GitHub: [github.com/apache/incubator-mxnet](https://github.com/apache/incubator-mxnet)
- Discussion forum: [discuss.mxnet.io](https://discuss.mxnet.io)
- Blog: [medium.com/apache-mxnet](https://medium.com/apache-mxnet)
- SlideShare: [slideshare.net/apachemxnet](https://www.slideshare.net/apachemxnet)
- Twitter: [@ApacheMXNet](https://twitter.com/@ApacheMXNet)
- YouTube : [youtube.com/apachemxnet](https://www.youtube.com/apachemxnet)
- Reddit: [r/mxnet/](https://www.reddit.com/r/mxnet/)
- Meetup: [meetup.com/pro/deep-learning-with-apache-mxnet](https://www.meetup.com/pro/deep-learning-with-apache-mxnet)

# ML course in coursera

The screenshot shows a web browser window with multiple tabs open. The active tab is for the 'Getting Started with AWS Machine Learning' course on Coursera. The page header includes the Coursera logo, a search bar, and links for 'For Enterprise', 'Log In', and 'Join for Free'. The main content area displays the course title 'Getting Started with AWS Machine Learning' offered by 'aws'. It features a large blue banner with an 'Enroll for Free' button (starts Sep 26), financial aid information, and a count of 5,139 already enrolled. Below the banner, there's a navigation bar with links for 'About', 'Syllabus', 'Instructors', 'Enrollment Options', and 'FAQ'. The 'About this Course' section contains a brief description of machine learning, a 'SHOW ALL' link, and two boxes: 'WHAT YOU WILL LEARN' and 'SKILLS YOU WILL GAIN'. The 'WHAT YOU WILL LEARN' box lists several learning objectives, and the 'SKILLS YOU WILL GAIN' box lists various AI-related skills. At the bottom, there's a section for 'Learners taking this Course are' with categories like 'Associate Directors', 'Data Scientists', 'IT Managers', 'Technical Leads', and 'Chief Technology Officers (CTOs)'. A URL at the bottom left is https://www.coursera.org/learn/aws-machine-learning?#about.

- **Introduction to Machine Learning**
- **Machine Learning Pipeline**
- **Amazon AI Services: Computer Vision**
- **Amazon AI Services: NLP**
- **Introduction to Amazon SageMaker**

# Dive into Deep Learning



An interactive deep learning book with code, math, and discussions

<http://d2l.ai/>  
<http://ko.d2l.ai/>

[STAT 157 Course at UC Berkeley, Spring 2019](#)

한국어 version of the first 4 chapters is available NOW.

- GitHub Pull Request for any correction is welcome
- Raise issue at <https://github.com/d2l-ai/d2l-ko/issues>

# Contact us!

[mxnet-info@amazon.com](mailto:mxnet-info@amazon.com)

DEV DAY

# Thank you!



© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

# 여러분의 피드백을 기다립니다!



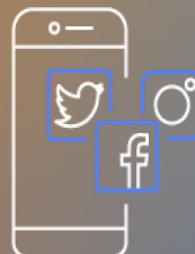
## 강연 평가 및 설문 조사

QR 코드를 통해 AWS DEV DAY SEOUL에 대한 여러분의 의견을 공유해주세요.  
강연 평가 및 설문 조사에 참여해 주신 분께는 등록데스크에서 특별한 기념품을 드립니다.



## 강연 영상

AWS DEV DAY SEOUL 강연 영상은 행사 종료 후 메일로 공유드릴 예정입니다.



## #AWSDEVDAYSEOUL

소셜미디어에 행사 참여 소감을 공유해주세요!

