

DEVELOPERWEEK™

FEBRUARY 17–19, 2021

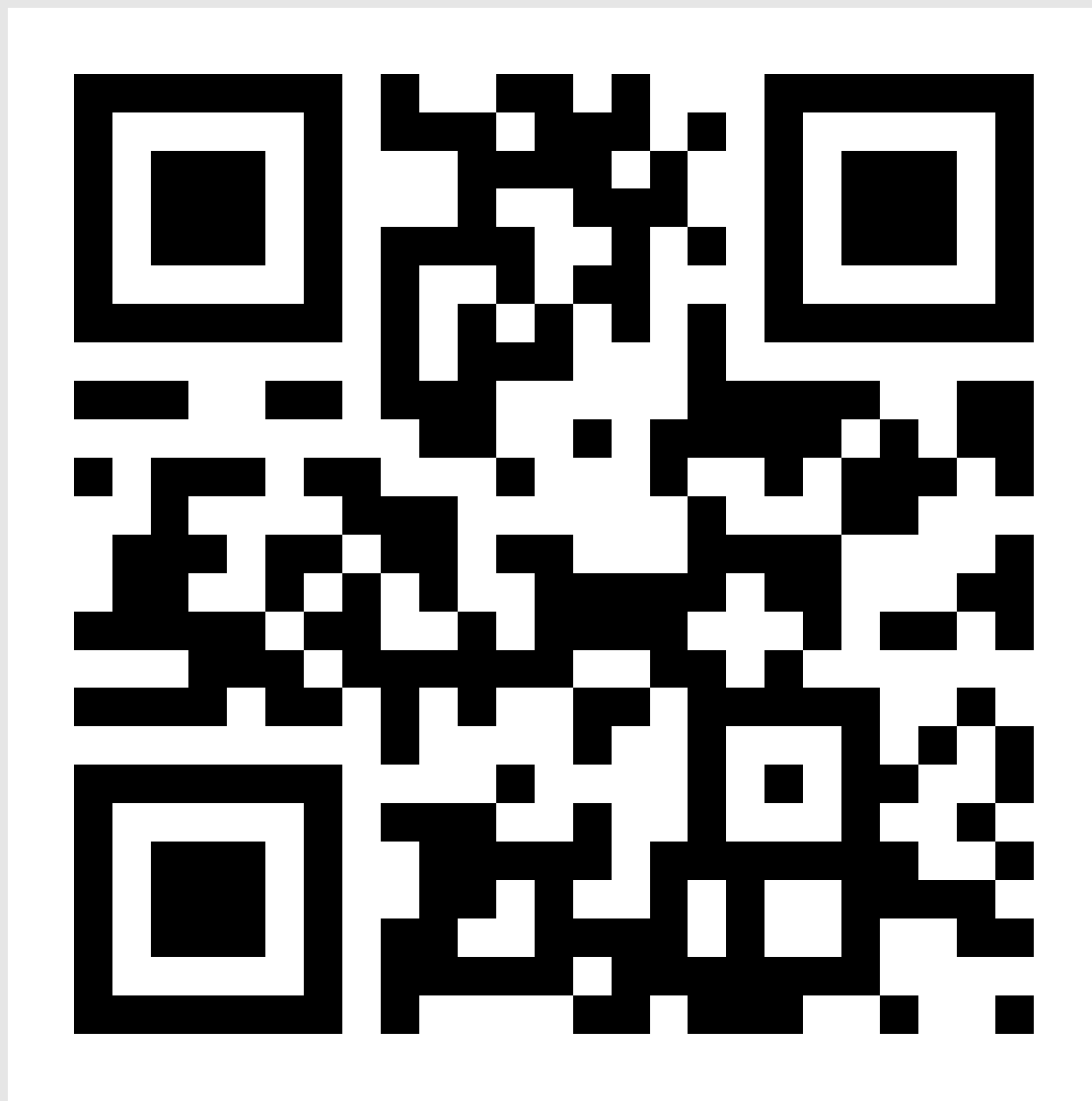
**The World's Largest
Developer & Engineering
Technology Conference & Expo**

Leverage Power of Machine
Learning with ONNX

Ron Dagdag
@rondagdag

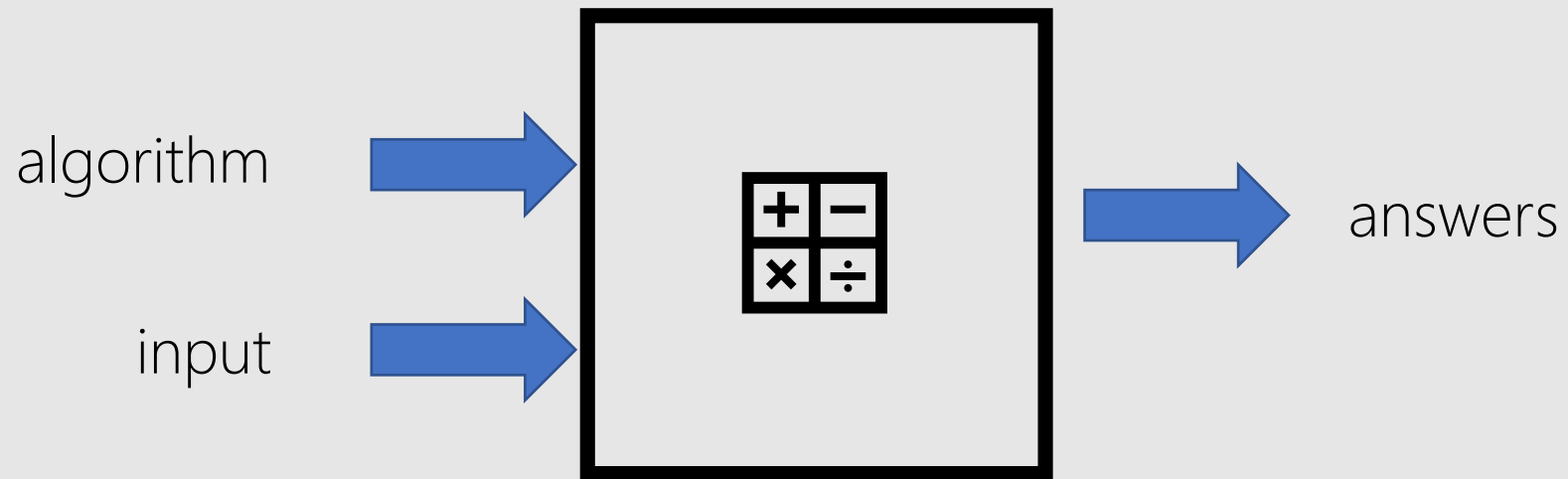
ONNX
Not ONIX
Not ONYX



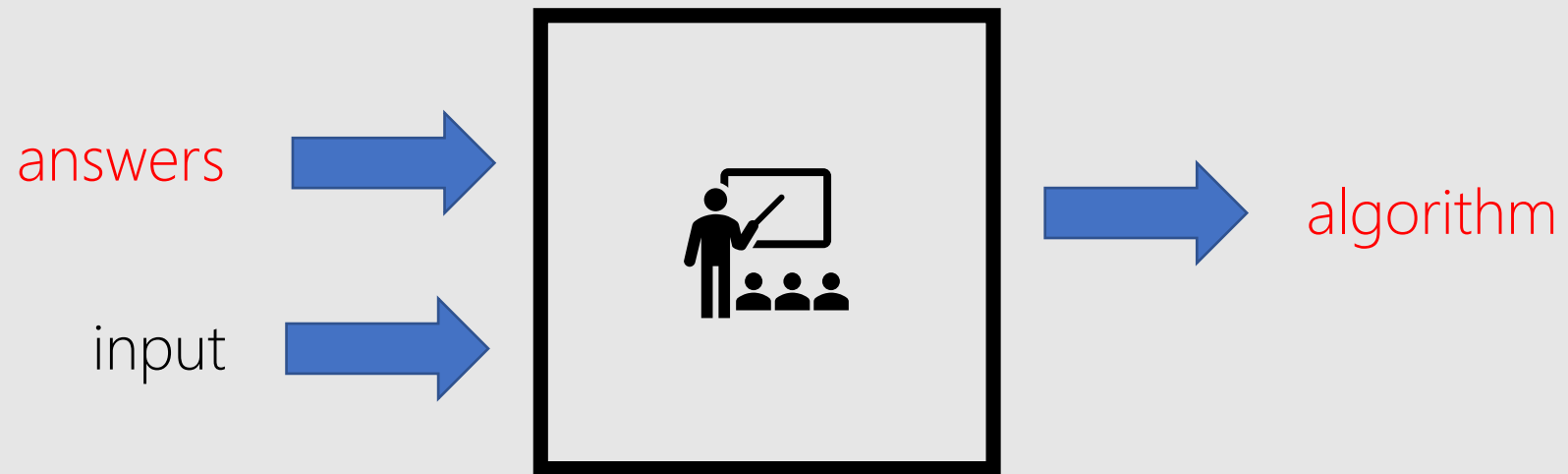


<http://bit.ly/ml-onnx>

programming



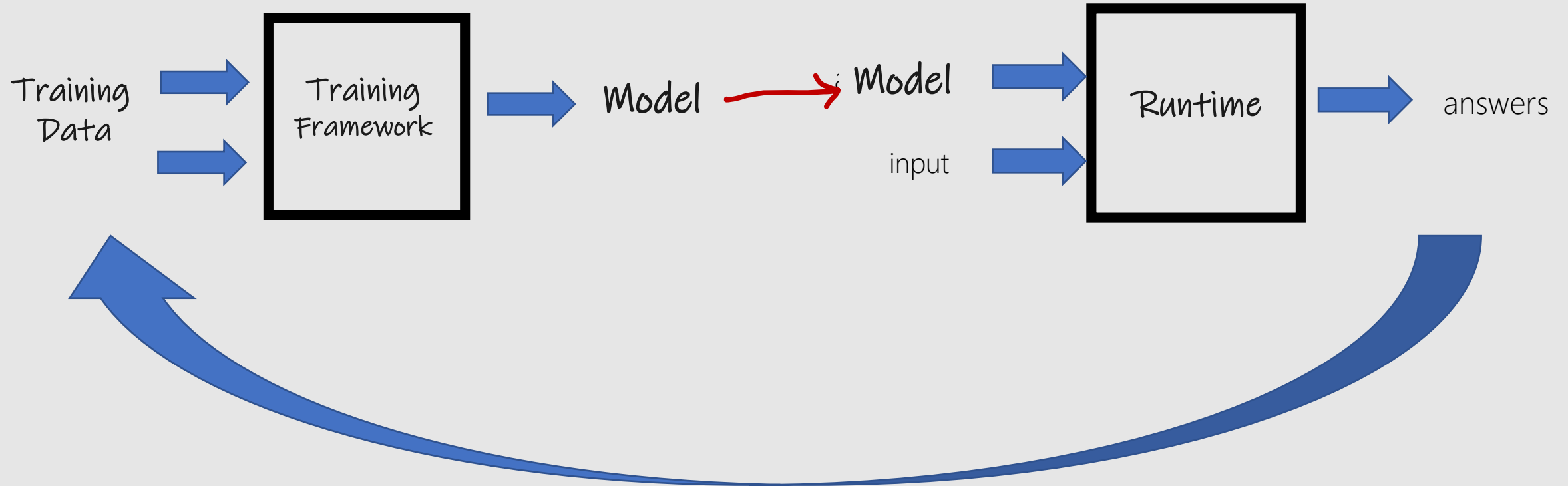
machine learning



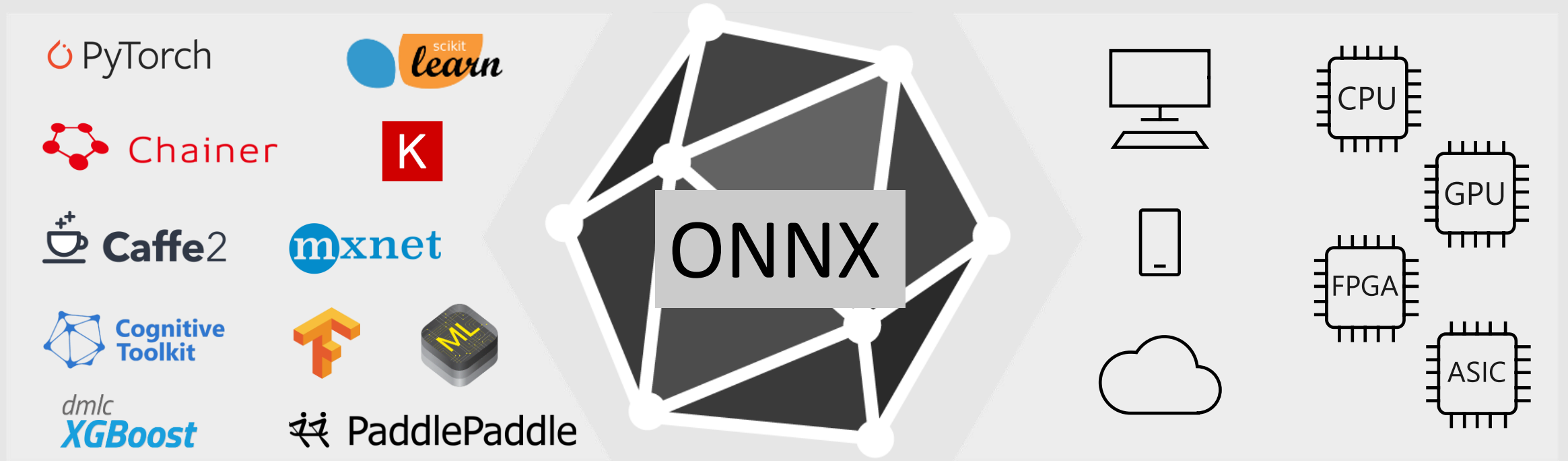
ML Primer

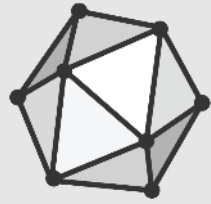
Machine learning

Inferencing



Open and Interoperable AI





ONNX

Open Neural Network Exchange

Open format for ML models

github.com/onnx

onnx.ai/



ONNX Partners

ABBYY®

Alibaba Group
阿里巴巴集团

AMD

arm

aws

Baidu 百度

BECKHOFF

BITMAIN

cadence®

CEVA®

Facebook
Open Source

GRAPHCORE

habana

HAILO

Hewlett Packard
Enterprise

HUAWEI

IBM

Idein Inc

intel AI

MathWorks®

MAXAR

MEDIATEK

MI

Microsoft

NVIDIA

NXP

OctoML

OPEN AI LAB
开放智能

Preferred
Networks

SIEMENS

SONY

Qualcomm

sas

商汤
sensetime

skymizer

SYNOPSYS®

Tencent

unity

verizon
media

WOLFRAM

Yandex

ZETANE



Agenda

- ✓ What is ONNX
- ☐ How to create ONNX models
- ☐ How to deploy ONNX models

Create

Frameworks



Native support

Converters

Services



Native support



Deploy

Cloud Services

Azure Machine Learning services

Ubuntu VM

Windows Server 2019 VM

Windows Devices

IoT Edge Devices

Other Devices (iOS, Android, etc)

Native support

Converters

Frameworks



**Step 1:
Create**

Services



Azure Custom
Vision Service

Native
support

Converters

Native
support



ONNX Model

Azure

Azure Machine Learning services

Ubuntu VM

Windows Server 2019 VM



Windows Devices

**Step 2:
Deploy**

Other Devices
(iOS, etc)

Native
support

Converters

A still life composition featuring a cardboard egg carton with several brown eggs in the foreground. In the background, a pair of wire-rimmed glasses and an open book are visible. The text "Secret Recipe" is overlaid in white, with a vertical line separating the words.

Secret Recipe

4 ways to get an ONNX model



ONNX Model Zoo



Azure Custom Vision Service



Convert existing models



Train models in Azure Machine Learning

Automated Machine Learning

ONNX Model Zoo: github.com/onnx/models

Image Classification

This collection of models take images as input, then classifies the major objects in the images into a set of predefined classes.

Model Class	Reference	Description
MobileNet	Sandler et al.	Efficient CNN model for mobile and embedded vision applications. Top-5 error from paper - ~10%
ResNet	He et al., He et al.	Very deep CNN model (up to 152 layers), won the ImageNet Challenge in 2015. Top-5 error from paper - ~3.6%
SqueezeNet	Iandola et al.	A lightweight CNN model with fewer parameters and less computation. Top-5 error from paper - ~4.8%
VGG	Simonyan et al.	Deep CNN model, won the ImageNet Challenge in 2014. Top-5 error from paper - ~7.4%

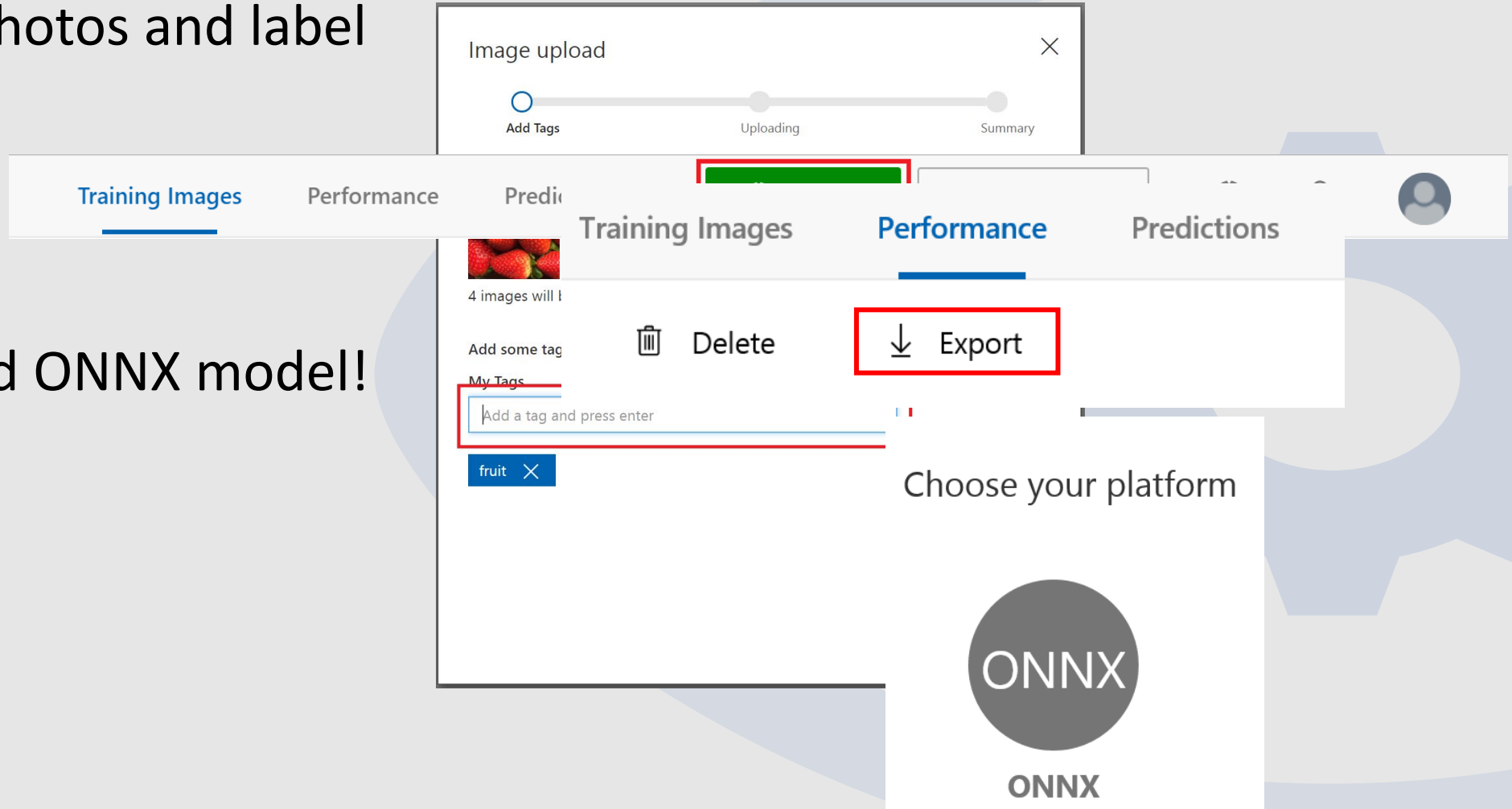
Model	Download	Checksum	Download (with sample test data)	ONNX version	Opset version	Top-1 accuracy (%)	Top-5 accuracy (%)
ResNet-18	44.6 MB	MD5	42.9 MB	1.2.1	7	69.70	89.49
ResNet-34	83.2 MB	MD5	78.6 MB	1.2.1	7	73.36	91.43
ResNet-50	97.7 MB	MD5	92.0 MB	1.2.1	7	75.81	92.82
ResNet-101	170.4 MB	MD5	159.4 MB	1.2.1	7	77.42	93.61
ResNet-152	230.3 MB	MD5	216.0 MB	1.2.1	7	78.20	94.21

Custom Vision Service: customvision.ai

1. Upload photos and label

2. Train

3. Download ONNX model!



Convert
models



Convert models

1. Load existing model
2. (Convert to ONNX)
3. Save ONNX model



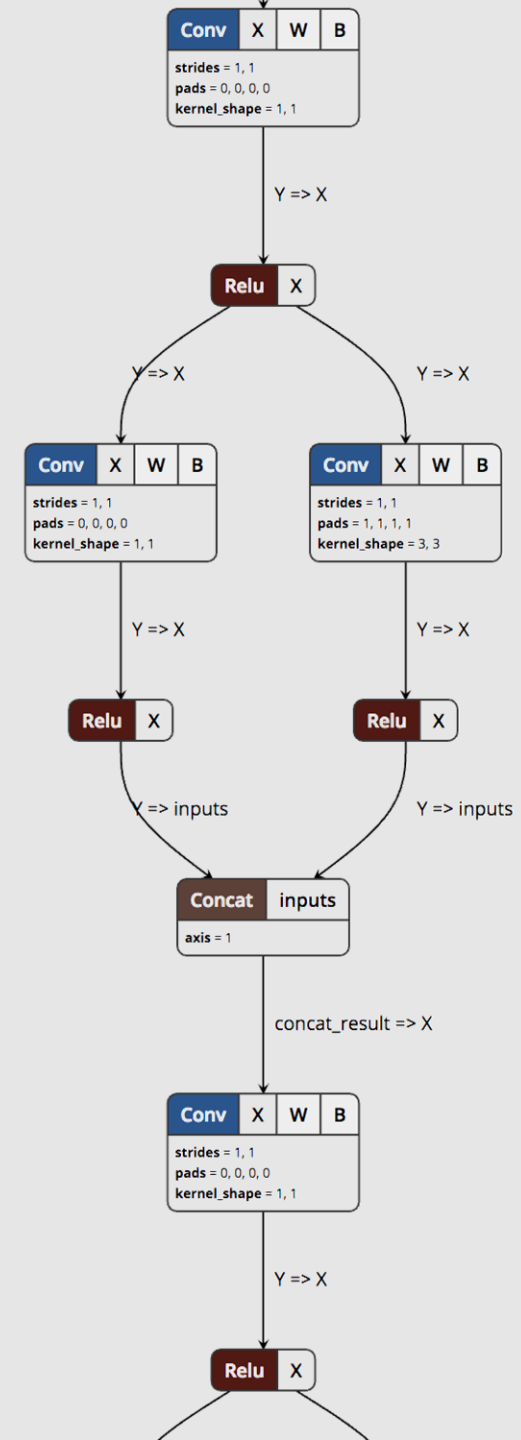
ONNX Models

Graph of operations

Netron

<https://netron.app/>

<https://lutzroeder.github.io/netron/>



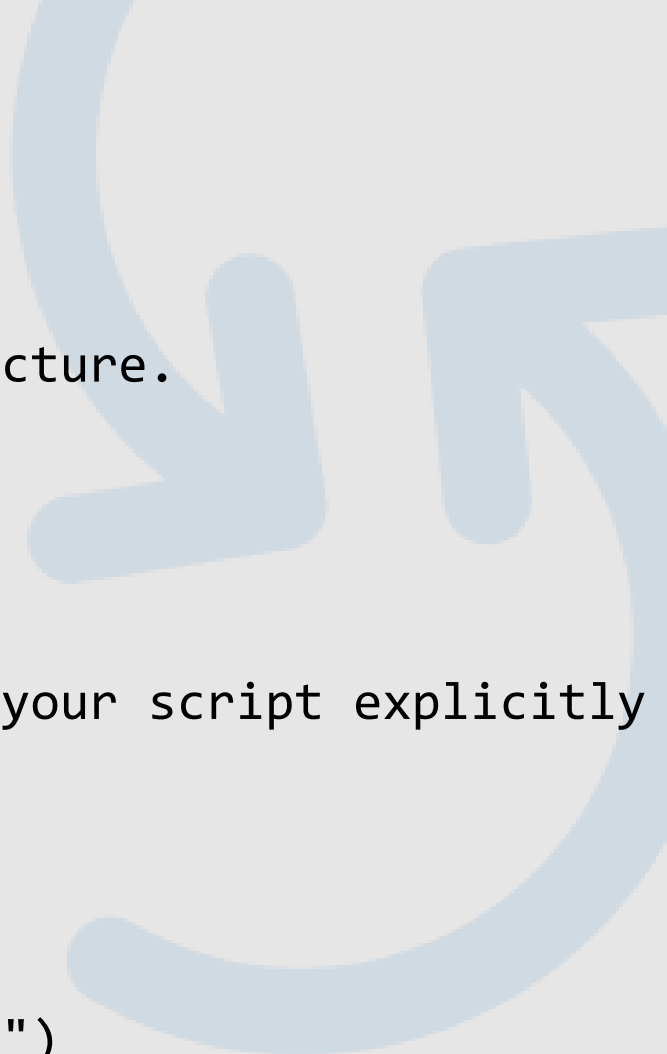
Convert models: Pytorch

```
import torch.onnx
import torchvision

# Standard ImageNet input - 3 channels, 224x224,
# values don't matter as we care about network structure.
# But they can also be real inputs.
dummy_input = torch.randn(1, 3, 224, 224)

# Obtain your model, it can be also constructed in your script explicitly
model = torchvision.models.alexnet(pretrained=True)

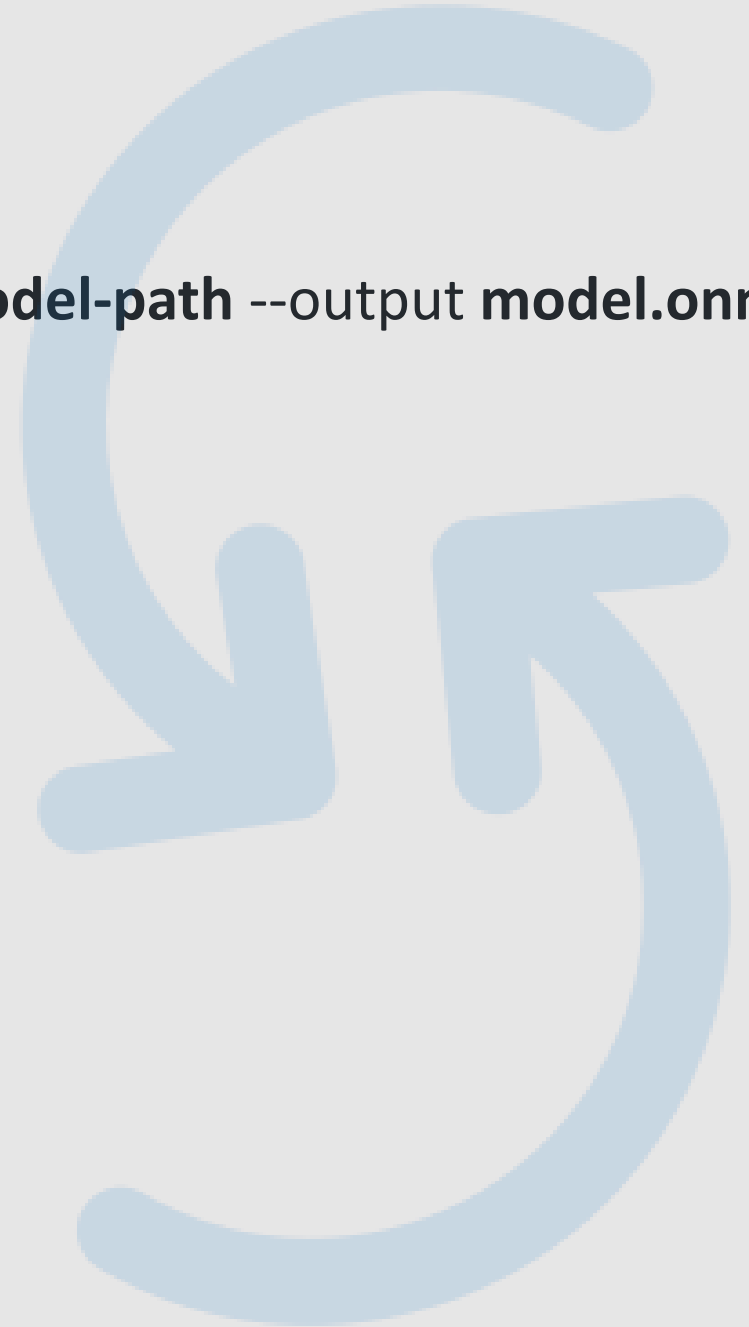
# Invoke export
torch.onnx.export(model, dummy_input, "alexnet.onnx")
```

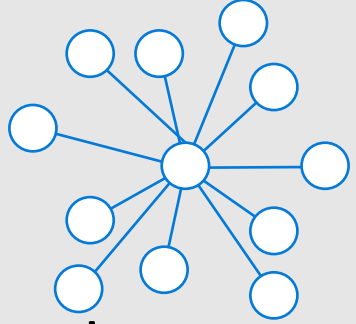


Convert models: TensorFlow

```
python -m tf2onnx.convert --saved-model tensorflow-model-path --output model.onnx
```

<https://github.com/onnx/tensorflow-onnx>





Train models in Azure Machine Learning

- Experiment locally then quickly scale with GPU clusters in the cloud
- Use automated machine learning and hyper-parameter tuning.
- Keeping Track of experiments, manage models, and easily deploy with integrated CI/CD tooling

Machine Learning

Typical E2E Process

Prepare



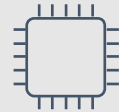
Prepare
Data



Experiment



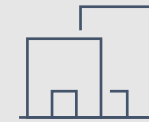
Build model
(your favorite IDE)



Train &
Test Model



Register and
Manage Model



Build
Image



Deploy Service
Monitor Model

Deploy



Frameworks

Caffe2 Chainer Cognitive Toolkit

mxnet PyTorch PaddlePaddle

ML.NET MathWorks dmlc XGBoost

**Step 1:
Create**

Services



Azure Custom
Vision Service

Native
support

Converters

Native
support



ONNX Model

Azure

Azure Machine Learning services

Ubuntu VM

Windows Server 2019 VM




**Step 2:
Deploy**

Other Devices
(iOS, etc)

Native
support

Converters

A baker in a white shirt and apron is shown from the waist down, working on a wooden table. The table is covered with a thick layer of white flour. The baker is shaping a large, round loaf of bread. The background is a plain, light-colored wall.

Baker vs Starting a Bakery

Create

Frameworks



Native support

Converters

Services



Native support



Deploy

Azure

Azure Machine Learning services

Ubuntu VM

Windows Server 2019 VM

Windows/Linux Devices

IoT Edge Devices

Other Devices
(iOS, etc)

Native support

Converters

A person's hands are visible, holding a large, round, rustic loaf of bread. The bread has a thick, golden-brown crust with some darker, caramelized spots. It is wrapped in a blue and white striped cloth, which is tied around the middle. The background is a blurred wooden surface, possibly a table or a wall.

Cloud or Edge

Deploy with Azure Machine Learning

- Model management services
- Deploy as web service to ACI or AKS
- Capture model telemetry



Azure
Machine Learning

Machine Learning

Typical E2E Process

Prepare



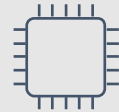
Prepare
Data



Experiment



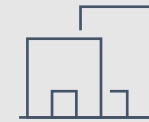
Build model
(your favorite IDE)



Train &
Test Model



Register and
Manage Model



Build
Image



Deploy Service
Monitor Model

Deploy





ONNX

ONNX Docker Image

[onnx-base](#): Use published ONNX package from PyPi with minimal dependencies.

[onnx-dev](#): Build ONNX from source with minimal dependencies.

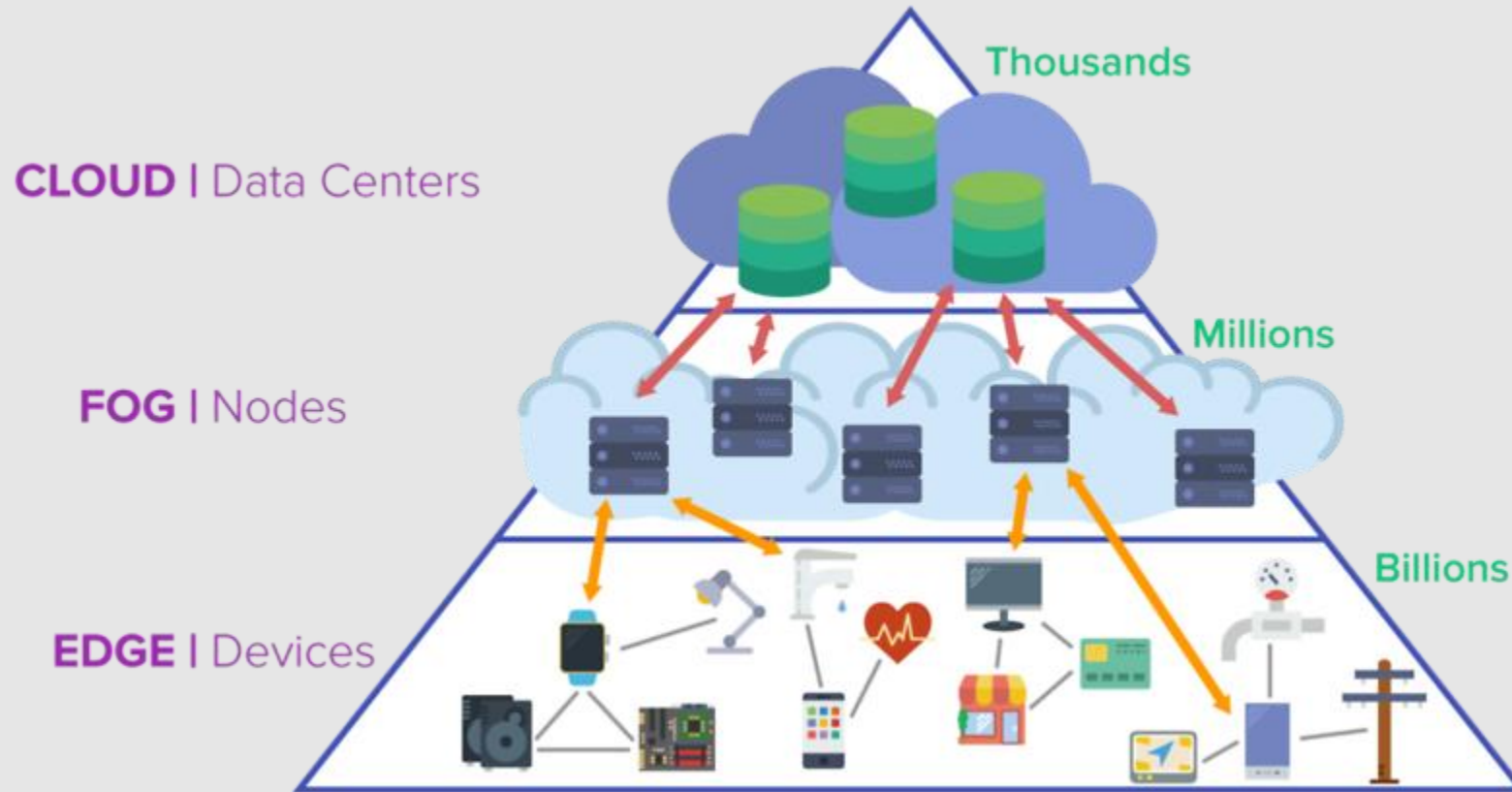
[onnx-ecosystem](#): Jupyter notebook environment

- getting started quickly with ONNX models
- ONNX converters
- inference using ONNX Runtime.

Caffe2/PyTorch Docker

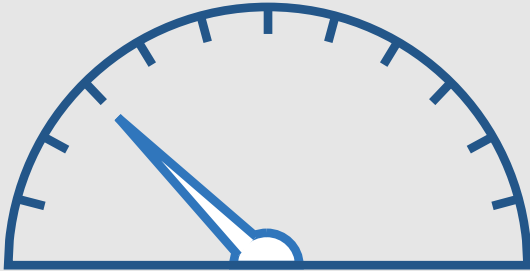
```
docker run -it --rm onnx/onnx-docker:cpu /bin/bash
```


What is the Edge?

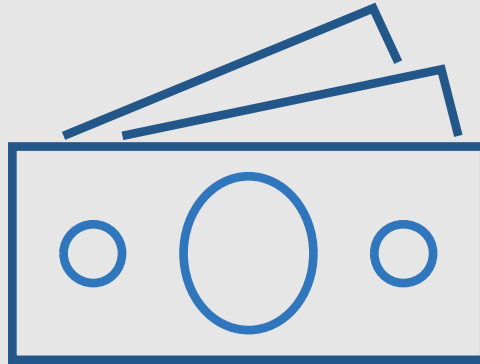


[Imagimob AB](#)

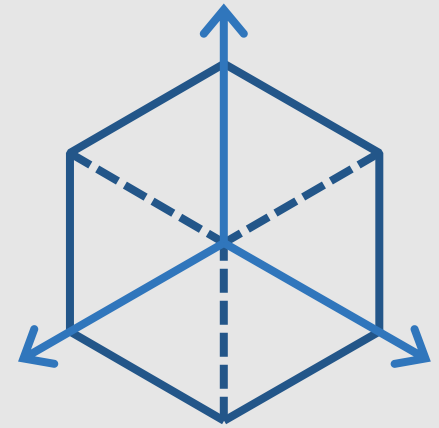
AI on the edge



Low latency



Scalability



Flexibility

ONNX as an intermediary format

- **Convert to Tensorflow for Android**
 - [Convert a PyTorch model to Tensorflow using ONNX](#)
- **Convert to CoreML for iOS**
 - <https://github.com/onnx/onnx-coreml>
- **Fine-tuning an ONNX model with MXNet/Gluon**
 - https://mxnet.apache.org/versions/1.3.1/tutorials/onnx/fine_tuning_gluon.html

<https://github.com/onnx/tutorials>

ONNX Runtime

- High performance runtime for ONNX models
- Supports full ONNX-ML spec
- Extensible architecture to plug-in hardware accelerators
- API Support
- onnxruntime.ai



ONNX

ONNX Runtime

Optimize Inferencing

Optimize Training (Preview)

OS

Windows

Linux

Mac

Android
(Preview)

iOS (Preview)

API

Python
(3.5-3.7)

C++

C#

C

Java

Javascript
(Node.js)

WinRT

Architecture

X64

X86

ARM64

ARM32

Hardware Acceleration

Default CPU

ACL (Preview)

ArmNN
(Preview)

CUDA

DirectML

DNNL

MIGraphX
(Preview)

NNAPI (Preview)

NUPHAR
(Preview)

OpenVINO

Rockchip NPU
(Preview)

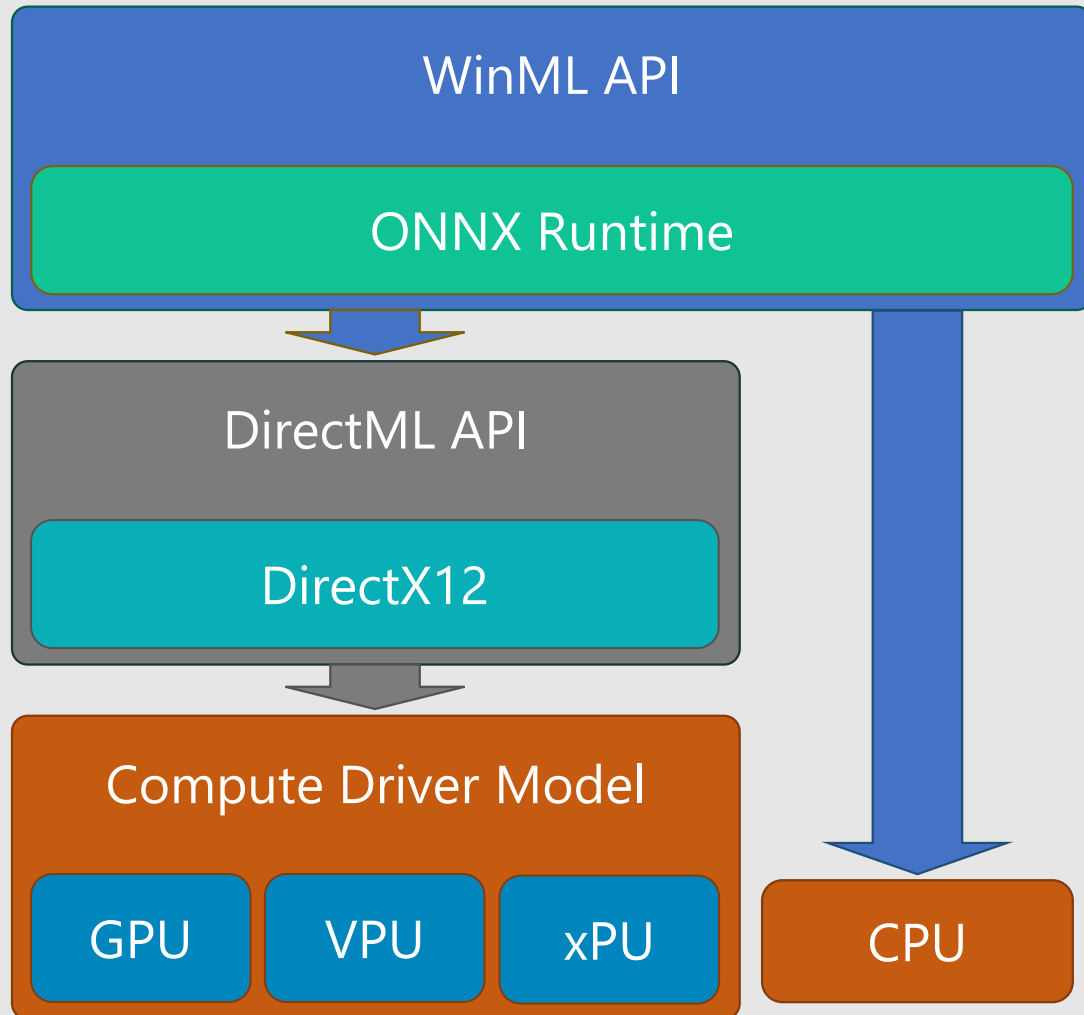
TensorRT

Vitis AI (Preview)

Installation Instructions

Install Nuget package [Microsoft.ML.OnnxRuntime.Gpu](#)

Windows AI platform



- WinML
 - **Practical**, simple model-based API for ML inferencing on Windows
- DirectML
 - **Realtime, high control** ML operator API; part of DirectX family
- Compute Driver Model
 - Robust **hardware reach**/abstraction layer for compute and graphics silicon

ONNX.js

- ONNX.js is a JavaScript library for running ONNX models on browsers and on Node.js.
- ONNX.js has adopted Web Assembly and WebGL technologies
- optimized ONNX model inference runtime for both CPUs and GPUs.

<https://github.com/microsoft/onnxjs>



ONNX

ONNX.js

Compatibility

Desktop Platforms

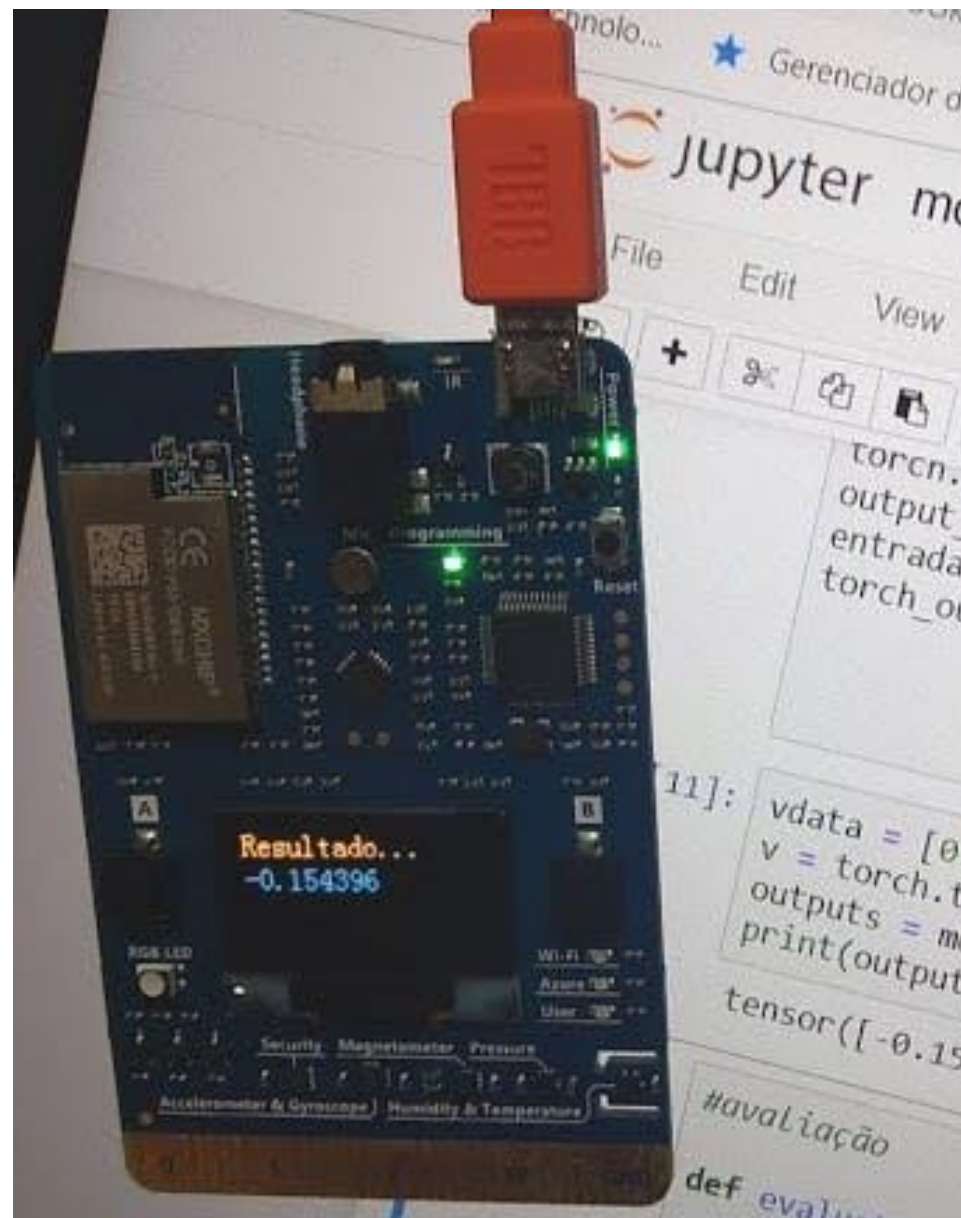
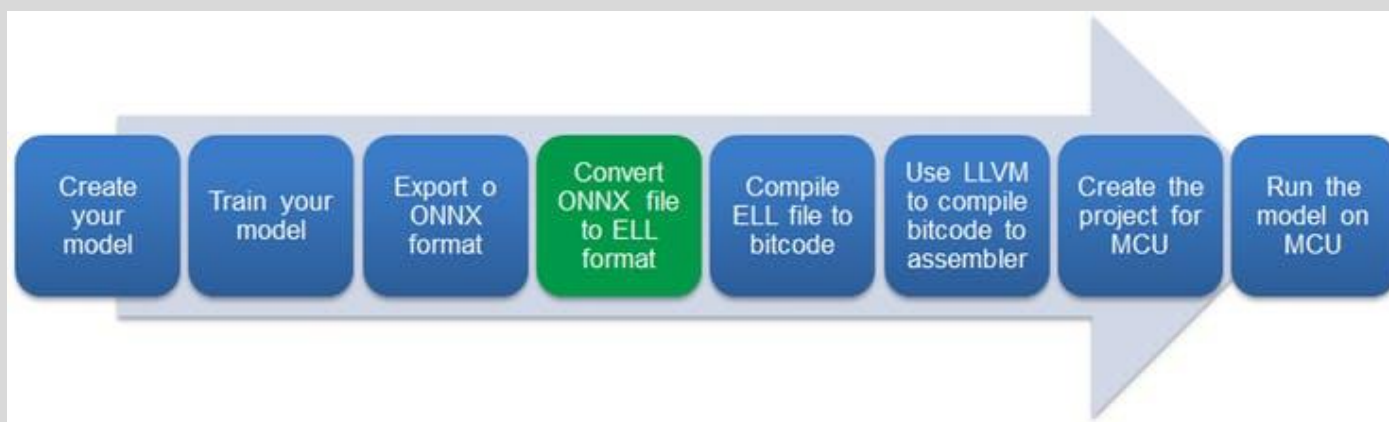
OS/Browser	Chrome	Edge	FireFox	Safari	Opera	Electron	Node.js
Windows 10	✓	✓	✓	-	✓	✓	✓
macOS	✓	-	✓	✓	✓	✓	✓
Ubuntu LTS 18.04	✓	-	✓	-	✓	✓	✓

Mobile Platforms

OS/Browser	Chrome	Edge	FireFox	Safari	Opera
iOS	✓	✓	✓	✓	✓
Android	✓	✓	Coming soon	-	✓

Wait... there's more

- Embedded Learning Library
 - <https://github.com/microsoft/ELL>
- Machine Learning Model Running on Azure IoT Starter Kit
 - <https://www.hackster.io/waltercoan/machine-learning-model-running-on-azure-iot-starter-kit-f9608b>



When to use ONNX?

- High Inferencing latency for production use
- Trained in Python - deploy into a C#/Java/JavaScript app
- Model to run resource constraint device (e.g. IoT/edge devices)
- Model to run on different OS or Hardware
- Combine running models created from different frameworks
- Training takes too long (transformer models)



Recap

- ✓ What is ONNX

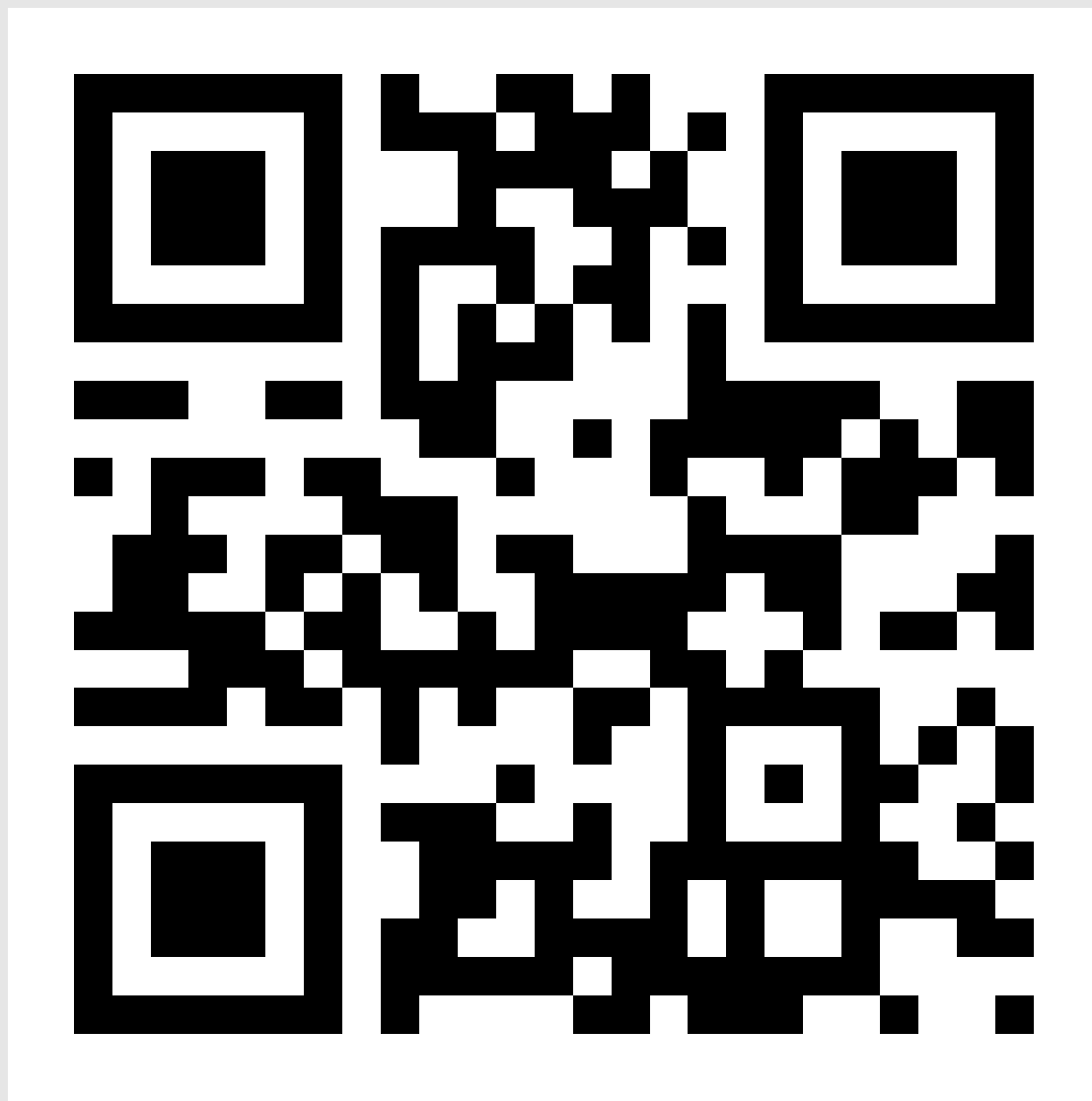
ONNX is an open standard so you can use the right tools for the job and be confident your models will run efficiently on your target platforms

- ✓ How to create ONNX models

ONNX models can be created from many frameworks

- ✓ How to deploy ONNX models

ONNX models can be deployed with Windows ML, .NET/Javascript/Python and to the cloud with Azure ML and the high performance ONNX Runtime



<http://bit.ly/ml-onnx>

About Me

Ron Dagdag



Ron Lyle Dagdag

Immersive Experience Developer

Cell: 682-560-3988

ron@dagdag.net



Experience AR

www.dagdag.net

[@rondagdag](https://twitter.com/rondagdag)

<http://ron.dagdag.net>

Lead Software Engineer at Spacee

4th year Microsoft MVP awardee

Personal Projects
www.dagdag.net

Email: ron@dagdag.net
Twitter [@rondagdag](https://twitter.com/rondagdag)


Connect me via Linked In
www.linkedin.com/in/rondagdag/

Thanks for geeking out with me about ONNX

Hackster Portfolio

www.dagdag.net

@rondagdag



Ron Dagdag
Dad / Lead Software Engineer / 3D Developer / Tax Return Preparer.
Passionate to learn about Robotics, VR, AR, Artificial Intelligence, IOT
@rondagdag
FORT WORTH, United States
Team [Augmented Reality](#)
Team [Virtual Reality](#)

INTERNET OF "KINECT" 2,443 16 44
IoT Gateway Azure Event Hub Stream Analytics Production Base
Posture Recognition using K...
Ron Dagdag

Easy 60 0
Littlebits Arduino Keyboard ...
Ron Dagdag

Intermediate 701 9
Alexa, tell Echobot to fly
Ron Dagdag

Advanced 1,345 12
Control your "Earth Rover" i...
Ron Dagdag

Advanced 2,256 30
ConstructAR - The Holograp...
TEAM ConstructAR

Intermediate 449 4
Color Changing Fireworks in...
Ron Dagdag