Disclaimer

All information in this session is public

No confidential information has been disclosed from private communication between Linaro and Linaro members

URL’s to the original source are provided in each slide
Why Deep Learning?
End-to-End Learning for Many Tasks

Slide from DIY Deep Learning for Vision: a Hands-On Tutorial with Caffe
It’s complex!!!
From cloud to edge devices
Always online

Uplink bandwidth and traffic

Latency vs real time constraints

Privacy concerns

From cloud to edge devices
From cloud to edge devices
From cloud to edge devices
From cloud to edge devices
AI/ML Frameworks
TensorFlow

Developed in-house by the Google Brain team

- Started as DistBelief in 2011
- Evolved into TensorFlow with its first commit in November 2015
- V1.0.0 released on Feb 11, 2017
TensorFlow

Developed in-house by the Google Brain team

- Started as DistBelief in 2011
- Evolved into TensorFlow with its first commit in November 2015
- V1.0.0 released on Feb 11, 2017

TensorFlow can be built as

- TensorFlow for cloud and datacenters
- TensorFlow Lite for mobile devices
- TensorFlow.js for AI in web browsers

TensorFlow models on tensorflow.github
TensorFlow

Developed in-house by the Google Brain team
- Started as DistBelief in 2011
- Evolved into TensorFlow with its first commit in November 2015
- V1.0.0 released on Feb 11, 2017

TensorFlow can be built as
- TensorFlow for cloud and datacenters
- TensorFlow Lite for mobile devices
- TensorFlow.js for AI in web browsers

Support multiple accelerators
- GPU and TPU
- Android NNAPI and NN HAL
- WebGL

TensorFlow models on tensorflow github
TensorFlow

Developed in-house by the Google Brain team
- Started as DistBelief in 2011
- Evolved into TensorFlow with its first commit in November 2015
- V1.0.0 released on Feb 11, 2017

TensorFlow can be built as
- TensorFlow for cloud and datacenters → GPU and TPU
- TensorFlow Lite for mobile devices → Android NNAPI and NN HAL
- TensorFlow.js for AI in web browsers → WebGL

TensorFlow models on tensorflow github

Support multiple accelerators

31,713 commits
1,624 contributors
1,610,734 lines of code
456 years of effort
1st Commit Nov ‘15
From TensorFlow to TensorFlow Lite

TensorFlow Lite uses FlatBuffers
TensorFlow 1st Commit in November 2015

Commits: Individual Commit

Commit ID f41959ccb2d9d4c722fe8fc3351401d53bcf4900

Contributor: Manjunath Kudlur
Date: 07-November-2015 at 00:27
Repository: git://github.com/tensorflow/tensorflow.git master

Files Modified: 1899
Lines Added: 343903
Lines Removed: 0

Changes by Language

<table>
<thead>
<tr>
<th>Language</th>
<th>Code Added</th>
<th>Code Removed</th>
<th>Comments Added</th>
<th>Comments Removed</th>
<th>Blanks Added</th>
<th>Blanks Removed</th>
</tr>
</thead>
<tbody>
<tr>
<td>C++</td>
<td>180966</td>
<td>0</td>
<td>40104</td>
<td>0</td>
<td>33693</td>
<td>0</td>
</tr>
<tr>
<td>Python</td>
<td>38122</td>
<td>0</td>
<td>15251</td>
<td>0</td>
<td>11904</td>
<td>0</td>
</tr>
<tr>
<td>HTML</td>
<td>16068</td>
<td>0</td>
<td>338</td>
<td>0</td>
<td>706</td>
<td>0</td>
</tr>
</tbody>
</table>
Caffe

- Made with expression, speed, and modularity in mind
- Developed by Berkeley AI Research (BAIR) and by community contributors
  - Yangqing Jia created the project during his PhD at UC Berkeley
  - Caffe is released under the BSD 2-Clause license
- Focus has been vision, but also handles sequences, speech, text
- Tools, reference models, demos, and recipes → Caffe Zoo
- Seamless switch between CPU and GPU

caffe.berkeleyvision.org  github.com/BVLC/caffe

4,137 commits
314 contributors
76,076 lines of code
19 years of effort
1st commit in Sept’13
15,000+ forks
Caffe2

Caffe2 improves Caffe 1.0 in a series of directions

- First-class support for large-scale distributed training
- Mobile deployment
- New hardware support (in addition to CPU and CUDA)
- Flexibility for future directions such as quantized computation
- Stress tested by the vast scale of Facebook applications
- Examples and pre-trained models available from the Caffe2 Zoo
- Running on mobile devices with Android and iOS
  - Step-by-step tutorial with camera demo
- Caffe1 models do not run with Caffe2
  - Converter tool available

3,678 commits
332 contributors
275,560 lines of code
73 years of effort
1st commit in June ’15
## Caffe2 1st commit in June 2015

**Commits**: Individual Commit

**Commit ID**: ac3e6a4d4103706864b336705bd59518f14a5186  
**Contributor**: Yangqing Jia  
**Date**: 25-June-2015 at 23:26  
**Repository**: git://github.com/caffe2/caffe2.git master  
**Commit Comment**: A clean init for Caffe2, removing my earlier hacky commits.

### Changes by Language

<table>
<thead>
<tr>
<th>Language</th>
<th>Code Added</th>
<th>Code Removed</th>
<th>Comments Added</th>
<th>Comments Removed</th>
<th>Blanks Added</th>
<th>Blanks Removed</th>
</tr>
</thead>
<tbody>
<tr>
<td>C++</td>
<td>26581</td>
<td>0</td>
<td>7938</td>
<td>0</td>
<td>4404</td>
<td>0</td>
</tr>
<tr>
<td>Python</td>
<td>5071</td>
<td>0</td>
<td>2903</td>
<td>0</td>
<td>1243</td>
<td>0</td>
</tr>
<tr>
<td>CUDA</td>
<td>1616</td>
<td>0</td>
<td>127</td>
<td>0</td>
<td>166</td>
<td>0</td>
</tr>
<tr>
<td>C</td>
<td>498</td>
<td>0</td>
<td>58</td>
<td>0</td>
<td>44</td>
<td>0</td>
</tr>
<tr>
<td>HTML</td>
<td>117</td>
<td>0</td>
<td>11</td>
<td>0</td>
<td>6</td>
<td>0</td>
</tr>
<tr>
<td>CSS</td>
<td>96</td>
<td>0</td>
<td>7</td>
<td>0</td>
<td>22</td>
<td>0</td>
</tr>
<tr>
<td>Make</td>
<td>14</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>6</td>
<td>0</td>
</tr>
<tr>
<td>shell script</td>
<td>1</td>
<td>0</td>
<td>6</td>
<td>0</td>
<td>2</td>
<td>0</td>
</tr>
</tbody>
</table>
Caffe2 and PyTorch join forces

Number of Commits who made changes to the project source code each month

https://www.openhub.net/p/_compare?project_0=caffe2&project_1=pytorch
MXNet is a multi-language machine learning (ML) library to ease the development of ML algorithms, especially for deep neural networks. MXNet is computation and memory efficient and runs on various heterogeneous systems, ranging from mobile devices to distributed GPU clusters.

Currently, MXNet is supported by Intel, Dato, Baidu, Microsoft, Wolfram Research, and research institutions such as Carnegie Mellon, MIT, the University of Washington, and the Hong Kong University of Science and Technology.

Gluon API, examples, tutorials and pre-trained models from the Gluon model zoo.
mxnet 1st Commit in April 2015

MXNet

Commits: Individual Commit

Commit ID: ab64fe792f874dddb193c9828fd2cc3898f6bee3

Contributor: Mu Li
Date: 30-April-2015 at 16:21
Repository: git://github.com/dmlc/mxnet.git master
Commit Comment: Initial commit

Files Modified: 3
Lines Added: 0
Lines Removed: 0
mxnet 1st Commit in April 2015

MXNet

Contributors: Mu Li

Activity on MXNet by Mu Li

All-time Commits: 393
12-Month Commits: 93
30-Day Commits: 3

Overall Kudo Rank: 1
First Commit: 30-Apr-2015
Last Commit: 16-Aug-2017

Commit history:
Mu Li • 3rd
Principal Scientist at Amazon
Palo Alto, California

Amazon
Carnegie Mellon University
See contact info
25 connections
# Deep Learning framework comparison

<table>
<thead>
<tr>
<th>General</th>
<th>MXNet</th>
<th>PyTorch</th>
<th>TensorFlow</th>
</tr>
</thead>
<tbody>
<tr>
<td>Project Activity</td>
<td>Very High Activity</td>
<td>Very High Activity</td>
<td>Very High Activity</td>
</tr>
<tr>
<td>Open Hub Data Quality</td>
<td>Updated 6 days ago</td>
<td>Updated 6 days ago</td>
<td>Updated 6 days ago</td>
</tr>
<tr>
<td>Homepage</td>
<td>mxnet.io</td>
<td>pytorch.org</td>
<td>tensorflow.org</td>
</tr>
<tr>
<td>Project License</td>
<td>Apache-2.0</td>
<td>BSD-3-Clause</td>
<td>Apache-2.0</td>
</tr>
<tr>
<td>Estimated Cost</td>
<td>$4,622,328</td>
<td>$9,352,186</td>
<td>$29,702,271</td>
</tr>
</tbody>
</table>

## All Time Statistics

<table>
<thead>
<tr>
<th>Contributors (All Time)</th>
<th>MXNet (732 developers)</th>
<th>PyTorch (1062 developers)</th>
<th>TensorFlow (1929 developers)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Views as graph</td>
<td>8686 commits</td>
<td>13864 commits</td>
<td>41676 commits</td>
</tr>
<tr>
<td>Initial Commit</td>
<td>over 3 years ago</td>
<td>over 2 years ago</td>
<td></td>
</tr>
</tbody>
</table>
Total lines of project source code, excluding comments and blank lines.

https://www.openhub.net/p/compare?project_0=MXNet&project_1=caffe2&project_2=TensorFlow
Number of Commits who made changes to the project source code each month

https://www.openhub.net/p/ compare?project_0=MXNet&project_1=caffe2&project_2=TensorFlow
Observations

- Each cloud player has its own deep learning framework
- Each AI framework has its own entire ecosystem of formats, tools, model store
- Each AI framework represents a significant investment
- Scaling and acceleration are fundamental to performance
Observations

● Each cloud player has its own deep learning framework
● Each AI framework has its own entire ecosystem of formats, tools, model store
● Each AI framework represents a significant investment
● Scaling and acceleration are fundamental to performance

If you want a really cool job like Manjunath, Yangqing or Mu Li....

INVENT A GREAT NEW AI/ML FRAMEWORK
NN accelerators and software solutions
Google Edge TPU

The Edge TPU is Google's purpose-built ASIC chip designed to run TensorFlow Lite ML inference at the edge

- AIY Edge TPU Dev Board
- AIY Edge TPU Accelerator

https://aiyprojects.withgoogle.com/edge-tpu/
Arm Mali-G72

Arm Mali-G72 is the second generation Bifrost-based GPU for High Performance products. Benefitting from advanced technologies such as claused shaders and full system coherency, Mali-G72 adds increased tile buffer memory supporting up to 16 x Multi-Sample Anti-Aliasing at minimal performance cost. Arithmetic optimizations tailored to complex Machine Learning and High Fidelity Mobile Gaming use cases provide 25% higher energy efficiency, 20% better performance density and 40% greater overall performance than devices based on previous generation Bifrost GPU.

Arm ML processor

The Arm Machine Learning processor is an optimized, ground-up design for machine learning acceleration, targeting mobile and adjacent markets:

- optimized fixed-function engines for best-in-class performance
- additional programmable layer engines support the execution of non-convolution layers, and the implementation of selected primitives and operators

The network control unit manages the overall execution and traversal of the network and the DMA moves data in and out of the main memory. Onboard memory allows central storage for weights and feature maps.

Arm OD processor

- Detects object in real time with Full HD at 60fps.
- Object sizes from 50x60 pixels to full screen.
- Virtually unlimited objects detected per frame.
- Detailed people model provides rich metadata and allows detection of direction, trajectory, pose and gesture.
- Advanced software running on accompanying application processor allows for higher-level behaviour to be determined, including sophisticated inter-frame tracking.
- Additional software libraries enable higher-level, on-device features, such as face recognition.

Arm NN

Arm NN SDK is a set of open-source Linux software and tools that enables machine learning workloads on power-efficient devices. It provides a bridge between existing neural network frameworks and power-efficient Arm Cortex CPUs, Arm Mali GPUs or the Arm Machine Learning processor.

Arm NN SDK utilizes the Compute Library to target programmable cores, such as Cortex-A CPUs and Mali GPUs, as efficiently as possible. It includes support for the Arm Machine Learning processor and, via CMSIS-NN, support for Cortex-M CPUs.

Arm NN

NN Frameworks

Arm NN

CMSIS-NN
Compute Library
Cortex-M CPU
Partner IP Driver & SW Functions
Third-party IP

Compute Library
Cortex-A CPU
Mali GPU

Compute Library
Arm ML Processor

Snapdragon NPE SW Diagram

https://connect.linaro.org/resources/hkg18/hkg18-306/
- 99 operators
- Caffe, TensorFlow, TensorFlow Lite, Huawei HiAI SDK, Android NN
- Converter tools from AI models to serialized offline model

https://connect.linaro.org/resources/hkg18/hkg18-302/
An ecosystem of 3rd parties providing NN IP and tools
Observations

- Complete offload vs heterogenous computing
- Shared memory vs sub-system memories and DMA
- Fixed operators and software fallback
- Graph split vs cost of context switch
- Serialized models and converter tools
Observations

- Complete offload vs heterogenous computing
- Shared memory vs sub-system memories and DMA
- Fixed operators and software fallback
- Graph split vs cost of context switch
- Serialized models and converter tools

- Forked and accelerated inference engine for each NN IP and each framework
  → high total cost of ownership
  → delayed rebases and updates
  → delayed security fixes
Call to Action
Linaro Collaboration

Members fund Linaro and drive work through engineering steering committees

Member and Linaro engineers collaborate to develop work once, for all

Linaro delivers output to members, into open source projects, and into the community

Now ~25 members, up from 6 in 2010

Over 300 OSS engineers globally, including 140 Linaro staff
Linaro works Upstream

Delivering high value collaboration
Top 5 company contributor to Linux and Zephyr kernels
Contributor to >70 open source projects; many maintained by Linaro engineers

<table>
<thead>
<tr>
<th>Company</th>
<th>4.8-4.13 Changesets</th>
<th>%</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Intel</td>
<td>10,833</td>
<td>13.1%</td>
</tr>
<tr>
<td>2 Red Hat</td>
<td>5,965</td>
<td>7.2%</td>
</tr>
<tr>
<td>3 Linaro</td>
<td>4,636</td>
<td>5.6%</td>
</tr>
</tbody>
</table>

Source: 2017 Linux Kernel Development Report, Linux Foundation
Linaro Machine Intelligence Initiative

- Common model description format and APIs to the runtime
- Common optimized runtime inference engine for Arm-based SoC
- Plug-in framework to support multiple 3rd party NPU, CPU, GPU, DSP
- CI loops on reference development boards to measure accuracy, performance speed up and regression testing
Arm NN becomes an open source project

Arm: Accelerating ML Collaboration with Arm NN and Linaro
Arm and Linaro announce launch of Machine Intelligence Initiative

Open Neural Network Exchange (ONNX)
An open source format for AI models
An extensible computation graph model
Definitions of built-in operators and standard data types
Initial focus on inference

ONNX Interface for Framework Integration (ONNXIFI)
Standardized interface for neural network inference on special-purpose accelerators, CPUs, GPUs, DSPs, and FPGAs
Discussions started last March

AI/ML Resources from HKG18
HKG18-417 - OpenCL support by NNVM & TVM
HKG18-413 - AI and Machine Learning BoF
HKG18-405 - Accelerating Neural Networks with...
HKG18-312 - CMSIS-NN
HKG18-306 - Overview of Qualcomm SNPE
HKG18-304 - Scalable AI server
HKG18-302 - Huawei HiAI : Unlock The Future
HKG18-200K2 - Keynote: Accelerating AI from Cloud to Edge
Join us at the

AI and Neural Networks on Arm Summit

At Linaro Connect Vancouver 2018
Wednesday 19 September - Hyatt Regency Vancouver,
655 Burrard Street, V6C 2R7
$45 to attend the summit only

<table>
<thead>
<tr>
<th>Speaker</th>
<th>Company</th>
<th>ID</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jem Davies</td>
<td>Arm</td>
<td>YVR18-300K1</td>
<td>Keynote: Enabling Machine Learning to Explode with Open Standards and Collaboration</td>
</tr>
<tr>
<td>Robert Elliott</td>
<td>Arm</td>
<td>YVR18-329</td>
<td>Arm NN intro</td>
</tr>
<tr>
<td>Pete Warden</td>
<td>Google Tensorflow</td>
<td>YVR18-338</td>
<td>Tensorflow for Arm devices</td>
</tr>
<tr>
<td>Mark Charlebois</td>
<td>Qualcomm</td>
<td>YVR18-330</td>
<td>Qualcomm Snapdragon AI Software</td>
</tr>
<tr>
<td>Thom Lane</td>
<td>Amazon AWS AI</td>
<td>YVR18-331</td>
<td>ONNX and Edge Deployments</td>
</tr>
<tr>
<td>Jammy Zhou</td>
<td>Linaro</td>
<td>YVR18-332</td>
<td>TVM compiler stack and ONNX support</td>
</tr>
<tr>
<td>Luba Tang</td>
<td>Skymizer</td>
<td>YVR18-333</td>
<td>ONNC (Open Neural Network Compiler) for ARM Cortex-M</td>
</tr>
<tr>
<td>Shouyong Liu</td>
<td>Thundersoft</td>
<td>YVR18-334</td>
<td>AI Alive: On Device and In-App</td>
</tr>
<tr>
<td>Ralph Wittig</td>
<td>Xilinx</td>
<td>YVR18-335</td>
<td>Xilinx: AI on FPGA and ACAP Roadmap</td>
</tr>
<tr>
<td>Andrea Gallo and others</td>
<td>Linaro, Arm, Qualcomm, Skymizer, Xilinx</td>
<td>YVR18-337</td>
<td>BoF: JIT vs offline compilers vs deploying at the Edge</td>
</tr>
</tbody>
</table>
Jem Davies, Arm Fellow and GM of the ML Group
Stay in touch!