The status of Tensorflow on Arm-based HPC

LTD20-104 presented by Paul Isaac’s, Director (LDCG, HPC-SIG)

25 March 2020
Welcome!

1. We are looking to actively develop software at two levels. One to enable a library/framework. One to make use of the library or variation of the library.

2. Currently, TensorFlow does compile and run on Aarch64 hardware. Problem solved!

3. Beyond compilation what is it used for...
What is TensorFlow?

   a. Great for Data Scientists?
      i. If they're also software engineers
      
         ii. With the addition of tools some of the heavy-lifting code is reduced.
         1. Keras preprocessing layers help prepare the dataset. (Available for TF2.0)

2. Models can be efficiently identified using AutoML principles
   a. Use KerasTuner for hyper-parameterisation tuning
      i. However, the number of iterations grows significantly during training

https://www.youtube.com/watch?v=_lsjCH3fd00&list=PLQY2H8rRoyvzuJw20FG82Lgm2SZjTdlXU&index=2&t=0s
Versions

1. Currently we are pursuing in parallel TensorFlow v1.15 and v2.2.0 with a resource emphasis on v2.2.0.
   a. Note: Python 2 support ended 01JAN2020. v2.1.0 is last to ship for Python 2.7.
   b. Starting on December 19, 2020, you can no longer create Google Cloud training jobs, batch prediction jobs, or model versions that use v1.15 runtime version.

2. TF Community Supported Builds
### Official Builds

<table>
<thead>
<tr>
<th>Build Type</th>
<th>Status</th>
<th>Artifacts</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linux CPU</td>
<td><strong>Ubuntu CC</strong> passing</td>
<td>PyPI</td>
</tr>
<tr>
<td>Linux GPU</td>
<td><strong>Ubuntu GPU PY3</strong> passing</td>
<td>PyPI</td>
</tr>
<tr>
<td>Linux XLA</td>
<td><strong>Ubuntu XLA</strong> failing</td>
<td>TBA</td>
</tr>
<tr>
<td>macOS</td>
<td><strong>MacOS PY2 CC</strong> passing</td>
<td>PyPI</td>
</tr>
<tr>
<td>Windows CPU</td>
<td><strong>Windows CPU</strong> passing</td>
<td>PyPI</td>
</tr>
<tr>
<td>Windows GPU</td>
<td><strong>Windows GPU</strong> passing</td>
<td>PyPI</td>
</tr>
<tr>
<td>Android</td>
<td><strong>Android</strong> passing</td>
<td>Download 1.15.0</td>
</tr>
<tr>
<td>Raspberry Pi 0 and 1</td>
<td><strong>Rpi01 py2</strong> failing <strong>Rpi01 py3</strong> passing</td>
<td>Py2 Py3</td>
</tr>
<tr>
<td>Raspberry Pi 2 and 3</td>
<td><strong>Rpi23 py2</strong> failing <strong>Rpi23 py3</strong> passing</td>
<td>Py2 Py3</td>
</tr>
</tbody>
</table>
# Build Status

## Community Supported Builds

<table>
<thead>
<tr>
<th>Build Type</th>
<th>Status</th>
<th>Artifacts</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linux AMD ROCm GPU Nightly</td>
<td>build failing</td>
<td>Nightly</td>
</tr>
<tr>
<td>Linux AMD ROCm GPU Stable Release</td>
<td>build passing</td>
<td>Release 1.15 / 2.x</td>
</tr>
<tr>
<td>Linux s390x Nightly</td>
<td>build running</td>
<td>Nightly</td>
</tr>
<tr>
<td>Linux s390x CPU Stable Release</td>
<td>build passing</td>
<td>Release</td>
</tr>
<tr>
<td>Linux ppc64le CPU Nightly</td>
<td>build running</td>
<td>Nightly</td>
</tr>
<tr>
<td>Linux ppc64le CPU Stable Release</td>
<td>build passing</td>
<td>Release 1.15 / 2.x</td>
</tr>
<tr>
<td>Linux ppc64le GPU Nightly</td>
<td>build running</td>
<td>Nightly</td>
</tr>
<tr>
<td>Linux ppc64le GPU Stable Release</td>
<td>build passing</td>
<td>Release 1.15 / 2.x</td>
</tr>
<tr>
<td>Linux CPU with Intel® MKL-DNN Nightly</td>
<td>build passing</td>
<td>Nightly</td>
</tr>
<tr>
<td>Linux CPU with Intel® MKL-DNN Stable Release</td>
<td>build passing</td>
<td>Release 1.15 / 2.x</td>
</tr>
<tr>
<td>Red Hat® Enterprise Linux® 7.6 CPU &amp; GPU Python 2.7, 3.6</td>
<td>build passing</td>
<td>1.13.1 PyPI</td>
</tr>
</tbody>
</table>
Bazel

Required to build TF.

- Bazel is a Google tool that automates software builds and tests. Supported build tasks include running compilers and linkers to produce executable programs and libraries, and assembling deployable packages for Linux, Android, iOS and other target environments.

- Latest release Mar.2020: r2.2.0

- Request for Aarch64 support -
  https://github.com/bazelbuild/bazel/issues/8833#issuecomment-591355107
AI on Servers Direction

Datacenter
- Training
- Inference
- Reinforcement training
- HPC AI (Online SNN)
- Hardware Accelerators
- SVE and SVE2

AI on Servers Team
- TensorFlow, PyTorch
- DNNL on ARM64
- Neoverse (N1SDP)
- Optimize for SVE (and SVE2)

Learning
- Batch/Datasets
- Online/Dynamic

Middleware
- OpenCV
- OpenCL
- OpenMP
Accelerating existing training methods

Use Cases Scenarios

**Image Classification**

**Object Detection**
Voice Recognition
Language Translation
Recommendation Engines
Sentiment Analysis in Customer Relationship Management

Examples required:

https://github.com/Linaro/aionservers

====

**Perception, Understanding, Discovering, Prediction and Creation.**
Test Environment

In addition to CI build status...

Need: Provide an environment to run application tests against and for pursuit of secondary goal towards AI.

- **DeepMind Lab**: [https://github.com/deepmind/lab](https://github.com/deepmind/lab)

- **Google Research**: [https://github.com/google-research/football](https://github.com/google-research/football)

- **Facebook Habitat**: [https://aihabitat.org/](https://aihabitat.org/)
  - Github: [https://github.com/facebookresearch/habitat-sim](https://github.com/facebookresearch/habitat-sim)


- **Cyberbotics Webots**: [https://github.com/cyberbotics/webots/wiki](https://github.com/cyberbotics/webots/wiki)
Accelerate Deployment of ARM into Data Centers

The Linaro Developer Cloud provides open source developers and commercial ISVs access to the latest Arm-based, server-class hardware running mainstream environments, to enable the IoT, Edge and Cloud ecosystem to develop, port, test and enable CI/CD for the Arm architecture.
Technology Progress

- Linaro HPC hardware being reconfigured towards a scalable environment.

- ARM providing their experience with building recipes for TF & MLperf

- Review of TensorFlowCI has found some build issues related to Python3 - ongoing.
  - https://github.com/Linaro/hpc_tensorflowci

- OpenCV on Aarch64 built. Needs automating.

- TODO:
  - Include additional Libraries OpenCL and OpenMP into the builds.
  - Test software builds on N1SDP platform.
  - Test containerisation.
Thank you

Continuing to accelerate deployment of your Arm-based solutions through collaboration

aionservers@linaro.org