Introduction to Google's TensorFlow

Thomas Quintana

Ft. Lauderdale Machine Learning Meetup

Sponsors

FORT LAUDERDALE
MACHINE LEARNING
MEETUP

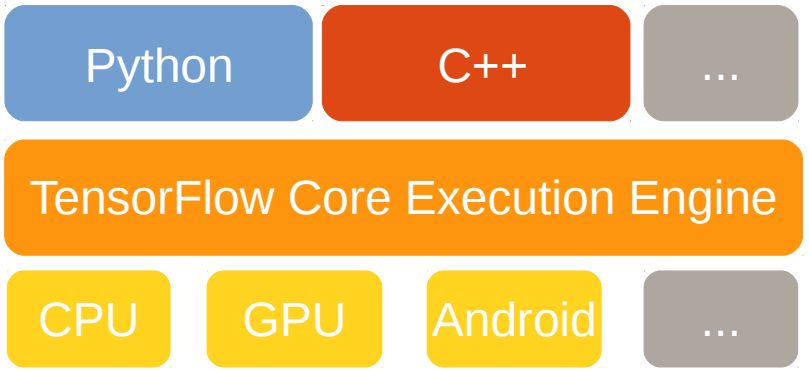
Agenda

- Why Do We Care?
- What is TensorFlow?
- Programming Model
- Underlying Implementations
- Extensions
- Machine Learning in TensorFlow
- TensorBoard
- Resources
- Conclusion

Why Do We Care?

- Has client language support in Python and C++ but many more are popping up in the community (e.g., Java and Rust).
- Has implementations for many platforms from mobile devices to large clusters of high end GPU servers.
- TensorFlow can be used for research and production systems, which greatly simplifies the adoption of machine learning algorithms.
- Support for distributed computation which allows TensorFlow clients to build larger models.
- Google is invested TensorFlow... They have the experience and resources to make it GREAT!

What is TensorFlow?



TensorFlow

/tensər floʊ/

An interface for expressing machine learning algorithms and an implementation for executing such algorithms.

Programming Model

The Data Flow Graph

Data flow graphs describe mathematical computation with a directed acyclic graph of nodes & edges.

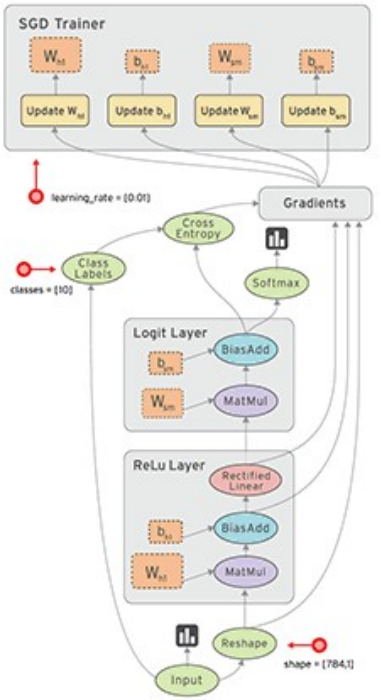


Figure 1: [https://www.tensorflow.org] A computation graph

Programming Model

Basic Concepts

- **Operation** – An operation has a name and represents an abstract computation. (e.g., “add” and “dot product”)
- **Kernel** – An implementation of an operation that can run on a particular type of device. (e.g., “CPU” and “GPU”)
- **Session** – Sessions manage and execute TensorFlow computation graphs. Therefore, client programs interact with the TensorFlow system through sessions.
- **Tensor** – A n-dimensional array that flows along the edges of the computation graph. Hence the name *TensorFlow* ;)
- **Variable** – A special kind of operation that returns a handle to a persistent mutable tensor that survives across executions of a graph.

Programming Model

Jupyter Notebook: Hands-on lab

Underlying Implementations

Components

The main components in a TensorFlow system are the *client*, which uses the Session interface to communicate with the *master* and one or more *worker processes*, with each worker process responsible for arbitrating access to one or more computational *devices* (such as CPU cores and GPU cards).

Underlying Implementations

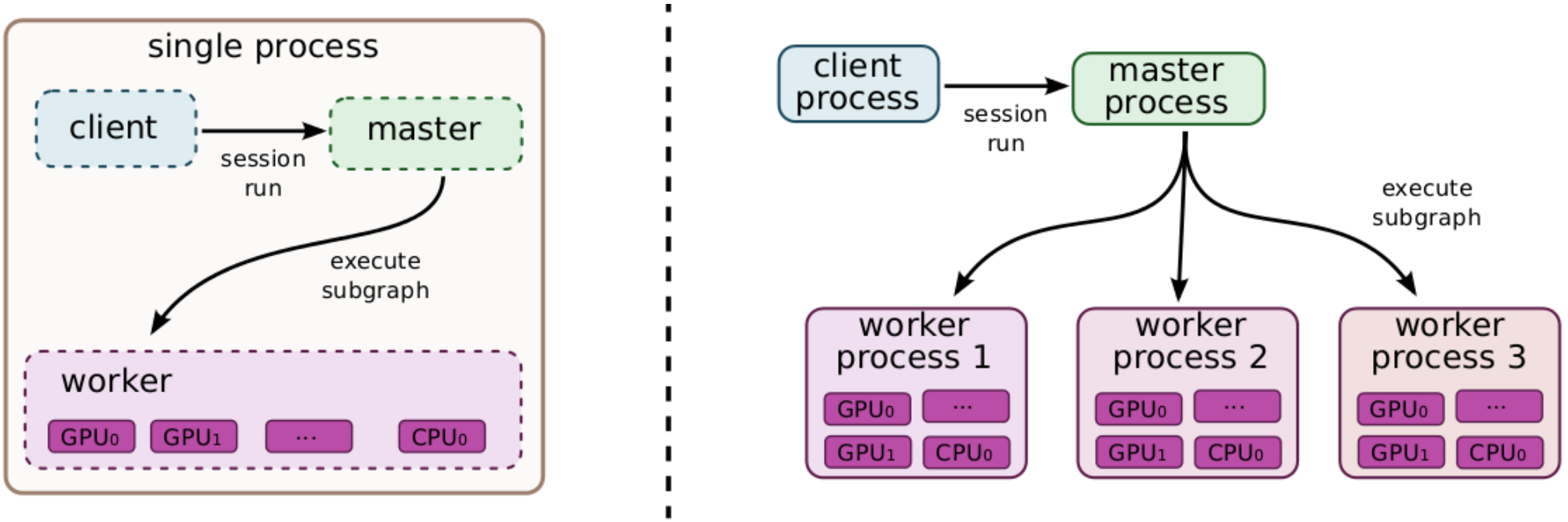


Figure 2: [TensorFlow: Large-scale machine learning on heterogeneous systems.] Single and Distributed System Structure

Extensions

- **Automatic Differentiation** – Automatically computes gradients for data flow graphs.
- **Partial Execution** – Allows TensorFlow clients to execute a subgraph of the entire execution graph.
- **Device Constraints** – Allows TensorFlow clients to control the placement of nodes on a device.
- **Control Flow** – Enables support for conditionals and loops in data flow graphs.
- **Input Operations** – Facilitate efficient loading of data into large scale models from the storage system.
- **Queues** – Allow different portions of the graph to execute asynchronously and to hand off data through Enqueue and Dequeue operation. Enqueue and Dequeue operations are **blocking**.
- **Containers** – The mechanism within TensorFlow for managing longer-lived mutable state.

Extensions

Automatic Differentiation

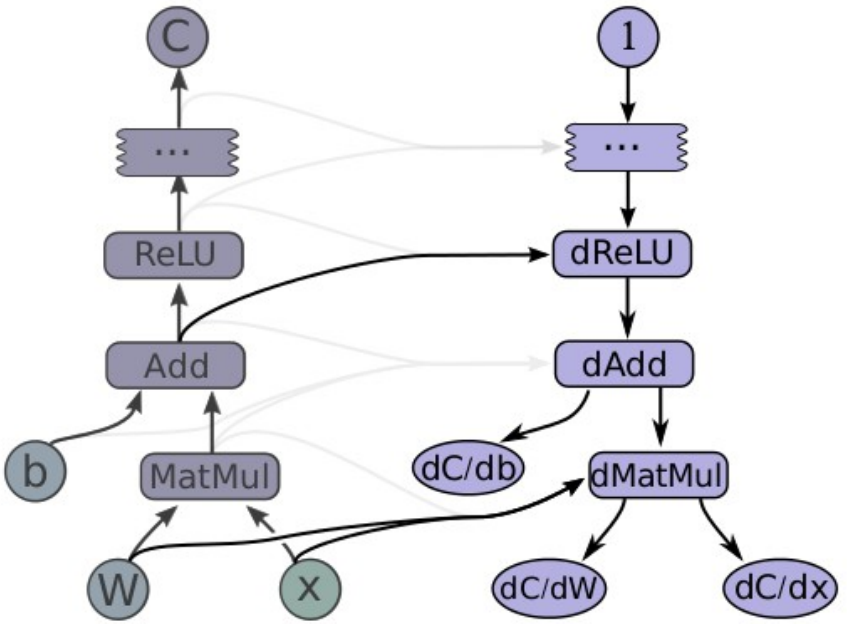


Figure 2: [TensorFlow: Large-scale machine learning on heterogeneous systems.] Gradients computed for graph.

Machine Learning with TensorFlow

Jupyter Notebook: Hands-on lab

FORT LAUDERDALE
MACHINE LEARNING
MEETUP

Resources

Papers

[1] TensorFlow: Large-scale machine learning on heterogeneous systems.2015

Websites

TensorFlow

Deep Learning | Udacity

GitHub

Conclusions

E-Mail: quintana.thomas@gmail.com

Twitter: @thomasquintana