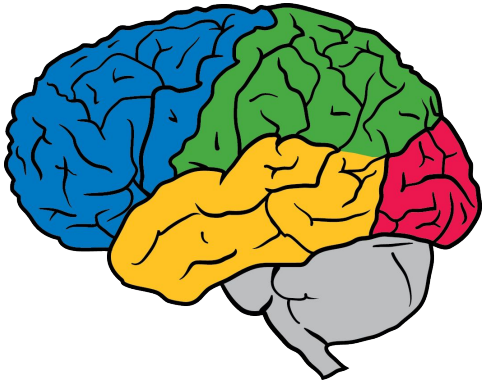


Large-Scale Deep Learning for Intelligent Computer Systems

Jeff Dean
Google Brain team
g.co/brain

In collaboration with **many** other people at Google

Building Intelligent Products

Building Intelligent Products

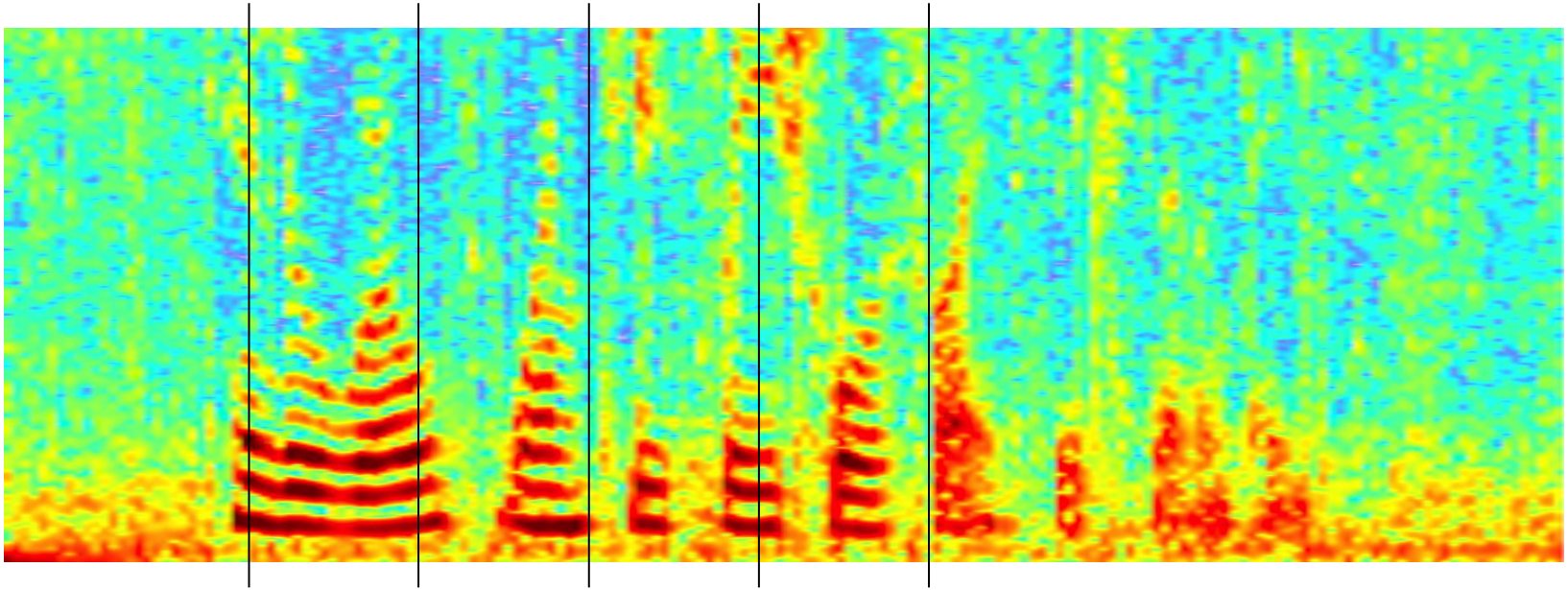
Really hard without **understanding**

Not there yet, but making significant progress

What do I mean by understanding?

What do I mean by understanding?

What do I mean by understanding?



Outline

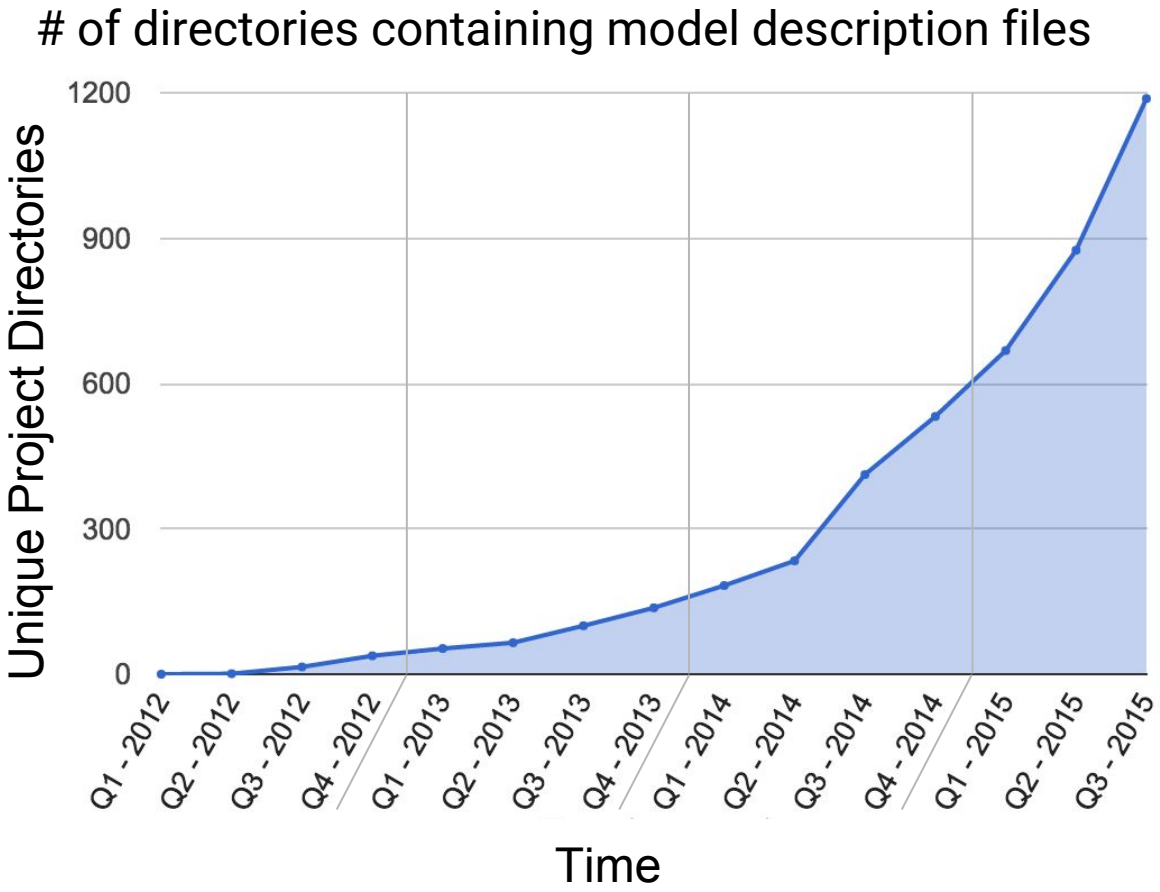
- Why deep neural networks?
- Examples of using these for solving real problems
- TensorFlow: software infrastructure for our work (and yours!)
- What are different ways you can get started using these?

Google Brain project started in 2011, with a focus on pushing state-of-the-art in neural networks. Initial emphasis:

- use large datasets, and
- large amounts of computation

to push boundaries of what is possible in perception and language understanding

Growing Use of Deep Learning at Google



Across many products/areas:

Android
Apps
drug discovery
Gmail
Image understanding
Maps
Natural language understanding
Photos
Robotics research
Speech
Translation
YouTube
… many others ...

The promise (or wishful dream) of Deep Learning

Speech
Text
Search Queries
Images
Videos
Labels
Entities
Words
Audio
Features

Simple,
Reconfigurable,
High Capacity,
Trainable end-to-end
Building Blocks

Speech
Text
Search Queries
Images
Videos
Labels
Entities
Words
Audio
Features

The promise (or wishful dream) of Deep Learning

Common representations across domains.

Replacing piles of code with
data and simple learning algorithms.

Would merely be an interesting academic exercise…

…**if it didn't work so well!**

In Research and Industry

Speech Recognition

Speech Recognition with Deep Recurrent Neural Networks
Alex Graves, Abdel-rahman Mohamed, Geoffrey Hinton

Convolutional, Long Short-Term Memory, Fully Connected Deep Neural Networks
Tara N. Sainath, Oriol Vinyals, Andrew Senior, Hasim Sak

Object Recognition and Detection

Going Deeper with Convolutions
Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, Andrew Rabinovich

Scalable Object Detection using Deep Neural Networks
Dumitru Erhan, Christian Szegedy, Alexander Toshev, Dragomir Anguelov

In Research and Industry

Machine Translation

Sequence to Sequence Learning with Neural Networks
Ilya Sutskever, Oriol Vinyals, Quoc V. Le

Neural Machine Translation by Jointly Learning to Align and Translate
Dzmitry Bahdanau, Kyunghyun Cho, Yoshua Bengio

Language Modeling

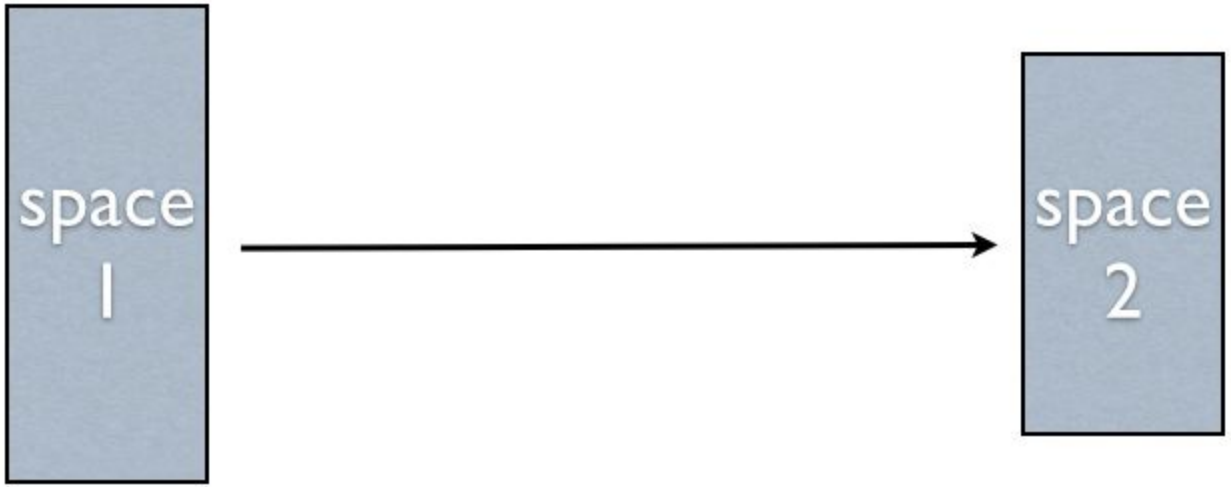
One Billion Word Benchmark for Measuring Progress in Statistical Language Modeling
Ciprian Chelba, Tomas Mikolov, Mike Schuster, Qi Ge, Thorsten Brants, Phillipp Koehn, Tony Robinson

Parsing

Grammar as a Foreign Language
Oriol Vinyals, Lukasz Kaiser, Terry Koo, Slav Petrov, Ilya Sutskever, Geoffrey Hinton

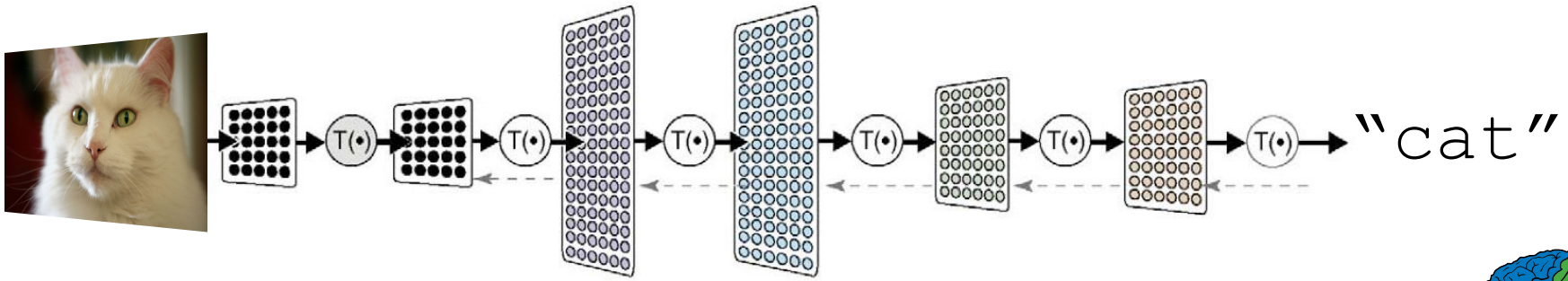
Neural Networks

- Learn a complicated function from data



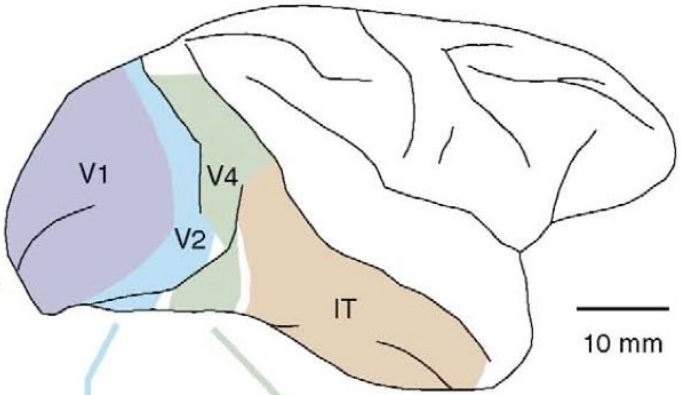
What is Deep Learning?

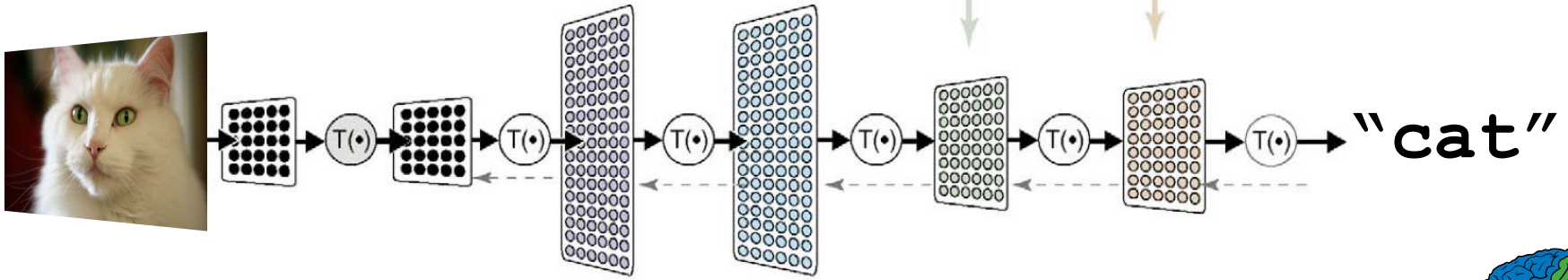
- A powerful class of machine learning model
- Modern reincarnation of artificial neural networks
- Collection of simple, trainable mathematical functions
- Compatible with many variants of machine learning



What is Deep Learning?

- Loosely based on (what little) we know about the brain





IS THIS A

CAT or DOG?

CAT DOG

OUTPUT LAYER

ACTIVATED NEURONS

INPUT LAYER

DEEP *neural* NETWORK

Learning algorithm

While not done:

- Pick a random training example “(input, label)”
- Run neural network on “input”
- Adjust weights on edges to make output closer to “label”

Learning algorithm

While not done:

Pick a random training example “(input, label)”

Run neural network on “input”

Adjust weights on edges to make output closer to “label”

Plenty of raw data

- **Text**: trillions of words of English + other languages
- **Visual data**: billions of images and videos
- **Audio**: tens of thousands of hours of speech per day
- **User activity**: queries, marking messages spam, etc.
- **Knowledge graph**: billions of labelled relation triples
- ...

How can we build systems that truly understand this data?

Important Property of Neural Networks

Results get better with

more data +

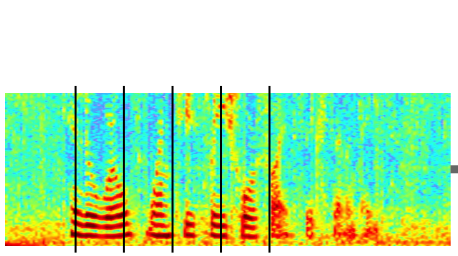
bigger models +

more computation

(Better algorithms, new insights and improved techniques always help, too!)

What are some ways that deep learning is having a significant impact at Google?

Speech Recognition



Acoustic Input

Deep
Recurrent
Neural Network

"How cold is
it outside?"

Text Output

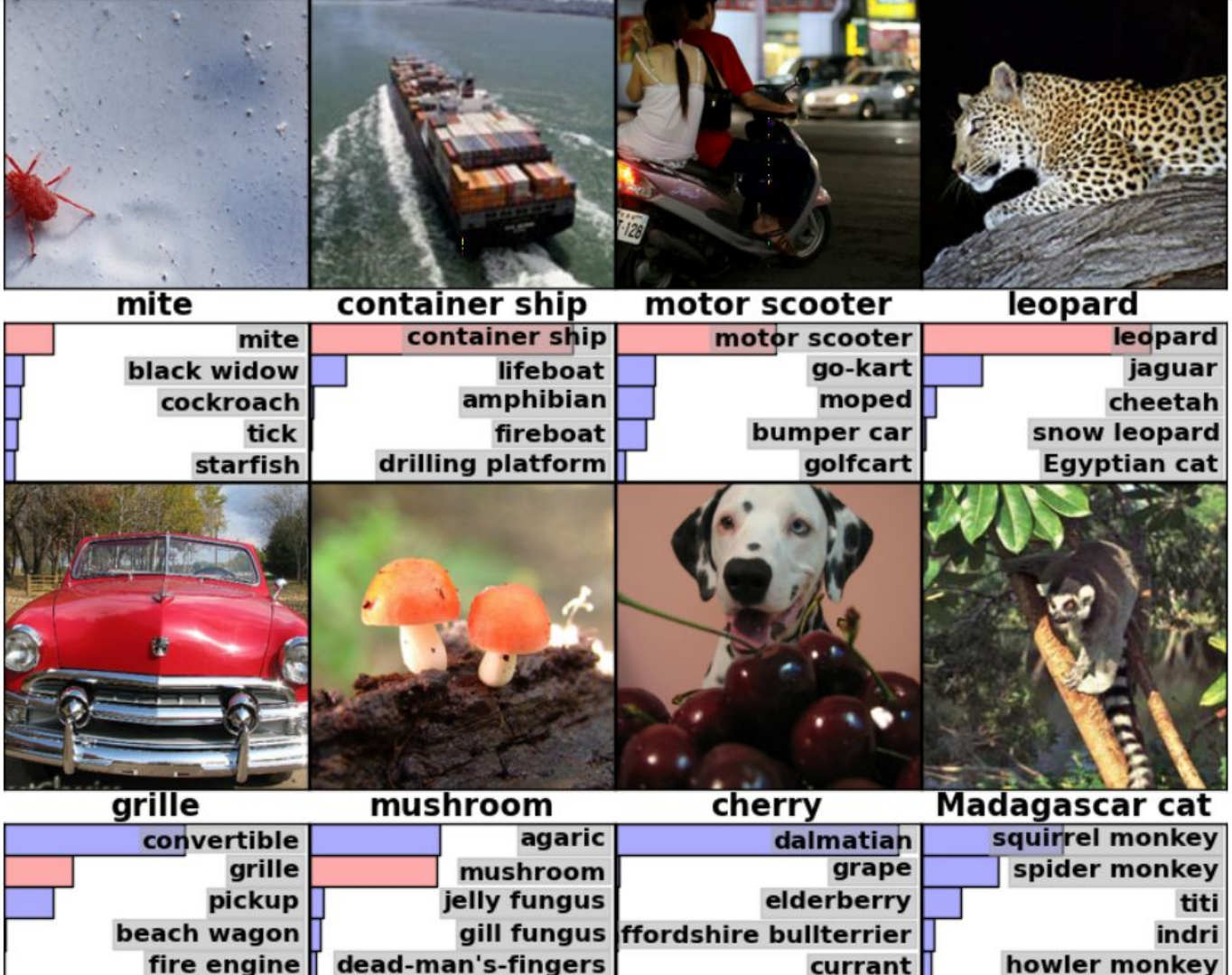
Reduced word errors by more than 30%

Google Research Blog - August 2012, August 2015

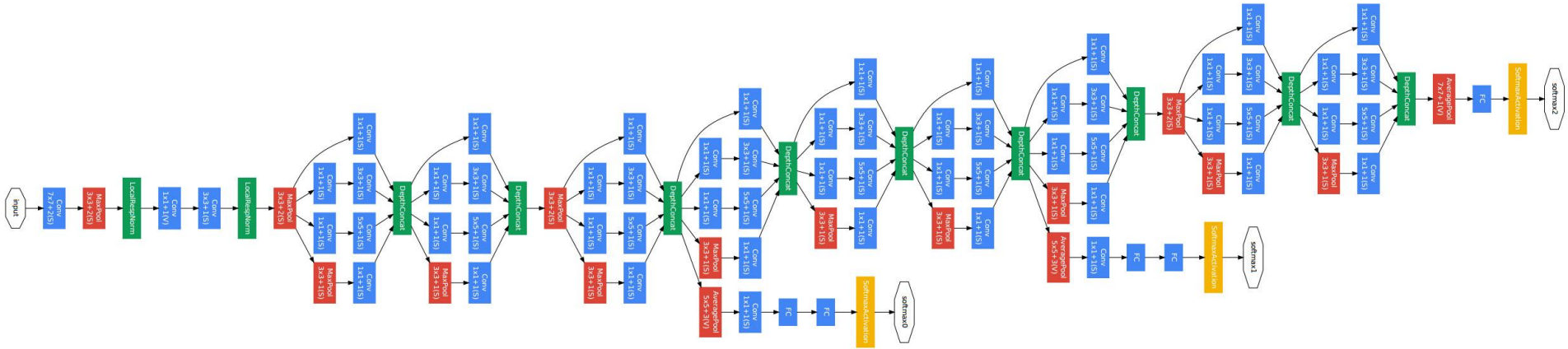
ImageNet Challenge

Given an image, predict one of 1000 different classes

Image credit: www.cs.toronto.edu/~fritz/absps/imagenet.pdf



The Inception Architecture (GoogLeNet, 2014)



Going Deeper with Convolutions

Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, Andrew Rabinovich

ArXiv 2014, CVPR 2015

Neural Nets: Rapid Progress in Image Recognition

Team	Year	Place	Error (top-5)
XRCE (pre-neural-net explosion)	2011	1st	**25.8%**
Supervision (AlexNet)	2012	1st	16.4%
Clarifai	2013	1st	11.7%
GoogLeNet (Inception)	2014	1st	6.66%
Andrej Karpathy (**human**)	2014	N/A	5.1%
BN-Inception (Arxiv)	2015	N/A	4.9%
Inception-v3 (Arxiv)	2015	N/A	**3.46%**

ImageNet challenge classification task

Good Fine-Grained Classification

“hibiscus”

“dahlia”

Good Generalization

Both recognized as “meal”

Sensible Errors

“snake”

“dog”

Google Photos Search

Your Photo

Deep Convolutional Neural Network

"ocean"

Automatic Tag

Search personal photos without tags.

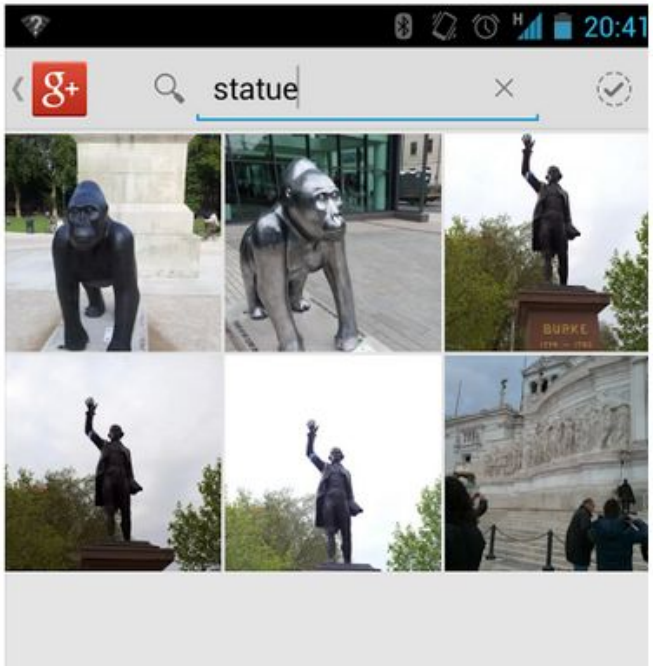
Google Research Blog - June 2013

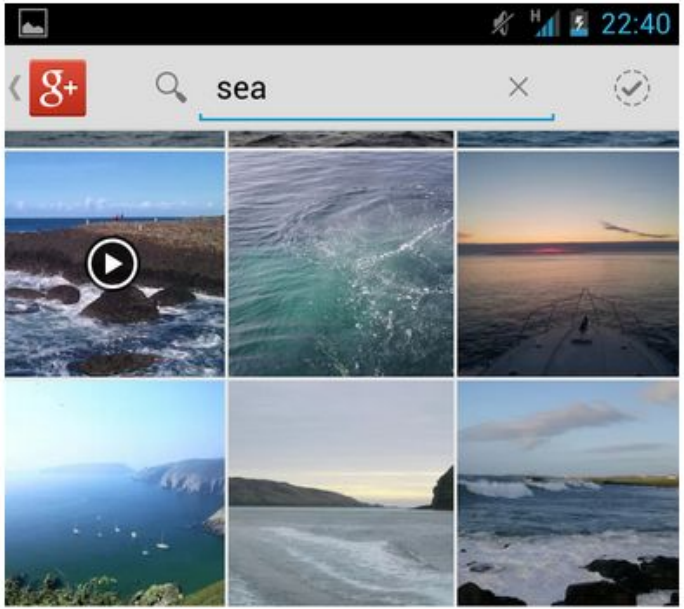
Google Photos Search

Wow.

The new Google plus photo search is a bit insane.

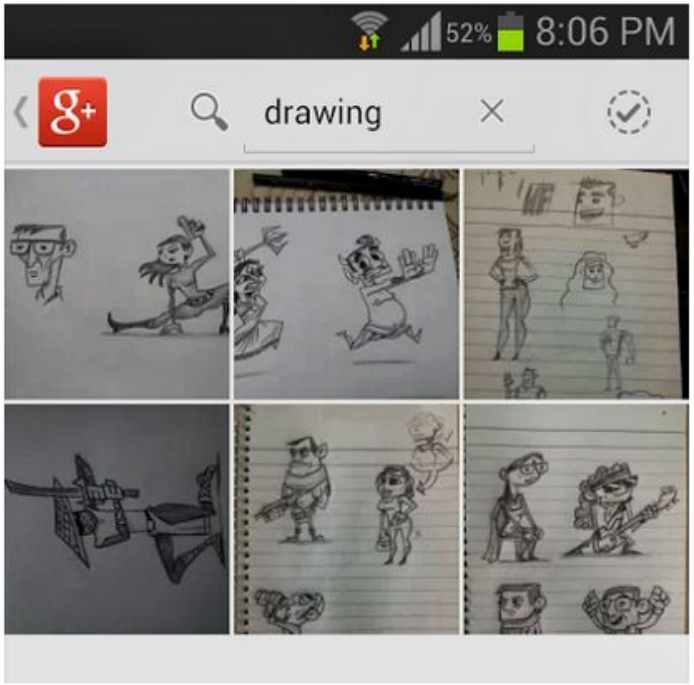
I didn't tag those... :)

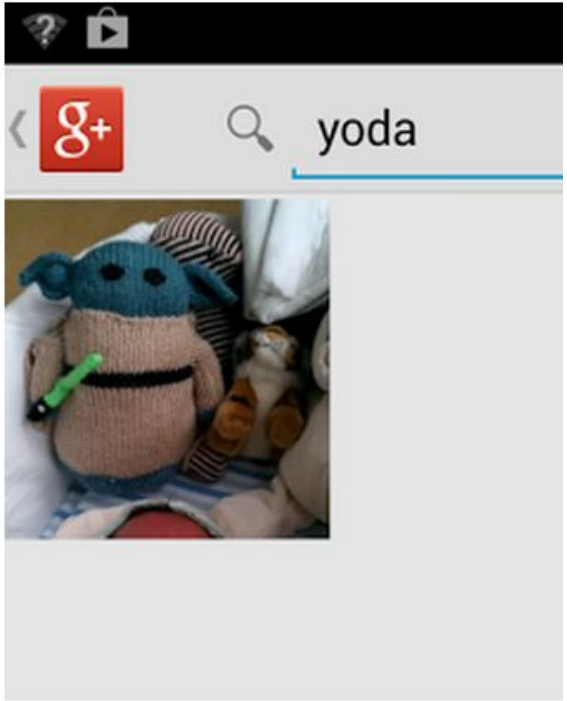




Google Photos Search

Google Plus photo search is awesome. Searched with keyword 'Drawing' to find all my scribbles at once :D



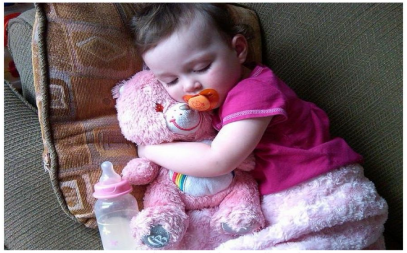



SIAWIDE TRAVEL
9745 3355 1st Floor, 240 BURWOOD RD
aria's Bakery Inn 超羣餅屋
Maria's Bakery Inn 超羣餅屋


Corner
Cubbyhouse
CIANO MOTOR ENGINEERS
MECHANICAL REPAIRS TO ALL MAKES AND MODELS
Specialising in BMW, MINI & TOYOTA
8 REGATTA ROAD FIVE DOCK 9745 3173
88
Factory Trained Technicians

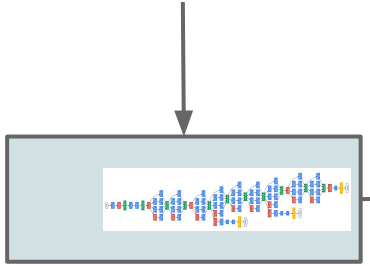
Image Captions

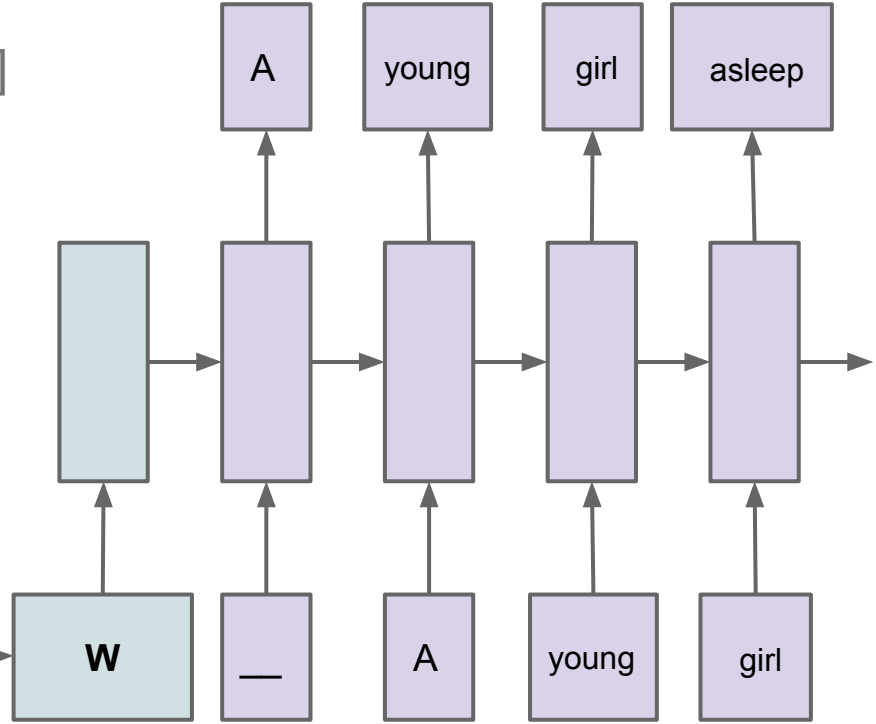
[Vinyals *et al.*, CVPR 2015]



A close up of a child holding a stuffed animal

(GT: A young girl asleep on the sofa cuddling a stuffed bear.)





$$\theta^\star = \arg\max_\theta p(S|I)$$

Human: A young girl asleep on the sofa cuddling a stuffed bear.

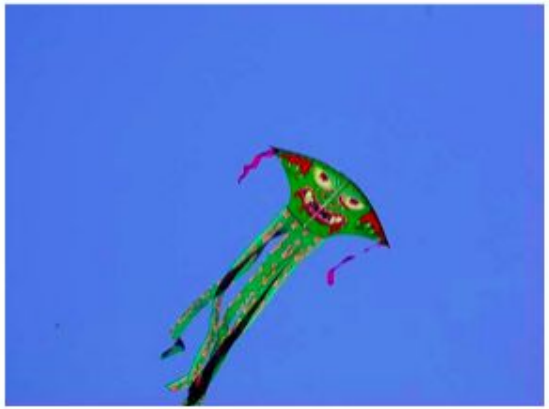
Model: A close up of a child holding a stuffed animal.

Model: A baby is asleep next to a teddy bear.

A man holding a tennis racquet on a tennis court.

Two pizzas sitting on top of a stove top oven

A group of young people playing a game of Frisbee



A man flying through the air while riding a snowboard

Combined Vision + Translation

What do you want in a machine learning system?

- **Ease of expression**: for lots of crazy ML ideas/algorithms
- **Scalability**: can run experiments quickly
- **Portability**: can run on wide variety of platforms
- **Reproducibility**: easy to share and reproduce research
- **Production readiness**: go from research to real products

TensorFlow:
Second Generation Deep Learning System

TensorFlow

If we like it, wouldn't the rest of the world like it, too?

Open sourced single-machine TensorFlow on Monday, Nov. 9th, 2015

- Flexible Apache 2.0 open source licensing
- Updates for distributed implementation coming soon

http://tensorflow.org/

and

https://github.com/tensorflow/tensorflow

Version: master

TensorFlow Mechanics 101

This is a technical tutorial, where we walk you through the details of using TensorFlow infrastructure to train models at scale. We use again MNIST as the example.

View Tutorial

Convolutional Neural Networks

An introduction to convolutional neural networks using the CIFAR-10 data set. Convolutional neural nets are particularly tailored to images, since they exploit translation invariance to yield more compact and effective representations of visual content.

View Tutorial

Vector Representations of Words

This tutorial motivates why it is useful to learn to represent words as vectors (called word embeddings). It introduces the word2vec model as an efficient method for learning embeddings. It also covers the high-level details behind noise-contrastive training methods (the biggest recent advance in training embeddings).

View Tutorial

Recurrent Neural Networks

An introduction to RNNs, wherein we train an LSTM network to predict the next word in an English sentence. (A task sometimes called language modeling.)

View Tutorial

Sequence-to-Sequence Models

A follow on to the RNN tutorial, where we assemble a sequence-to-sequence model for machine translation. You will learn to build your own English-to-French translator, entirely machine learned, end-to-end.

View Tutorial

TensorFlow:
Large-Scale Machine Learning on Heterogeneous Distributed Systems

(Preliminary White Paper, November 9, 2015)

Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng
Google Research*

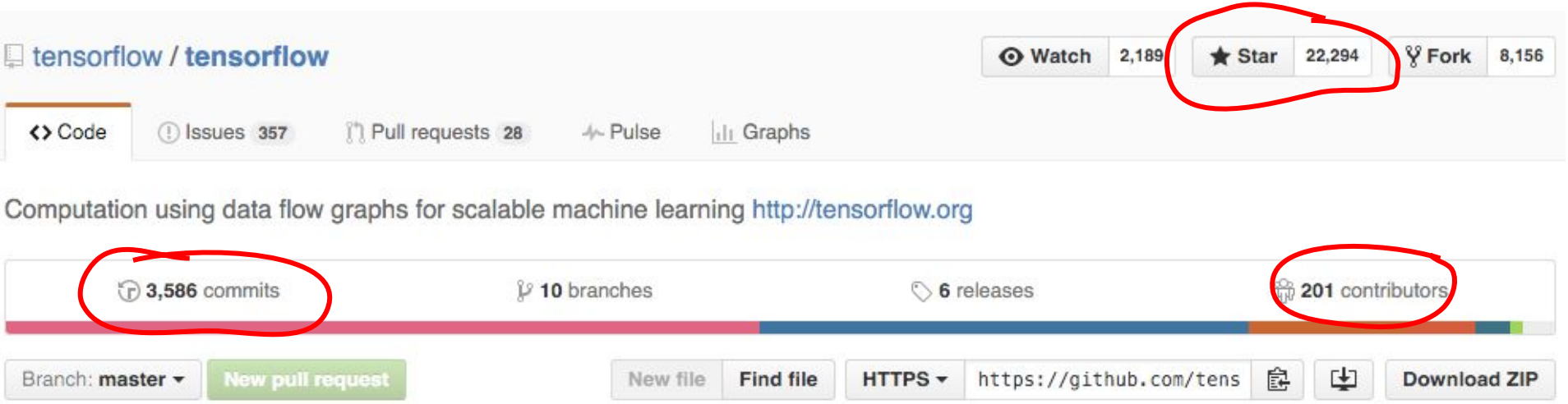
Abstract

TensorFlow [1] is an interface for expressing machine learning algorithms, and an implementation for executing such algorithms. A computation expressed using TensorFlow can be executed with little or no change on a wide variety of heterogeneous systems, ranging from mobile devices such as phones

sequence prediction [47], move selection for Go [34], pedestrian detection [2], reinforcement learning [38], and other areas [17, 5]. In addition, often in close collaboration with the Google Brain team, more than 50 teams at Google and other Alphabet companies have deployed deep neural networks using DistBelief in a wide variety

http://tensorflow.org/whitepaper2015.pdf

Source on GitHub



https://github.com/tensorflow/tensorflow

Motivations

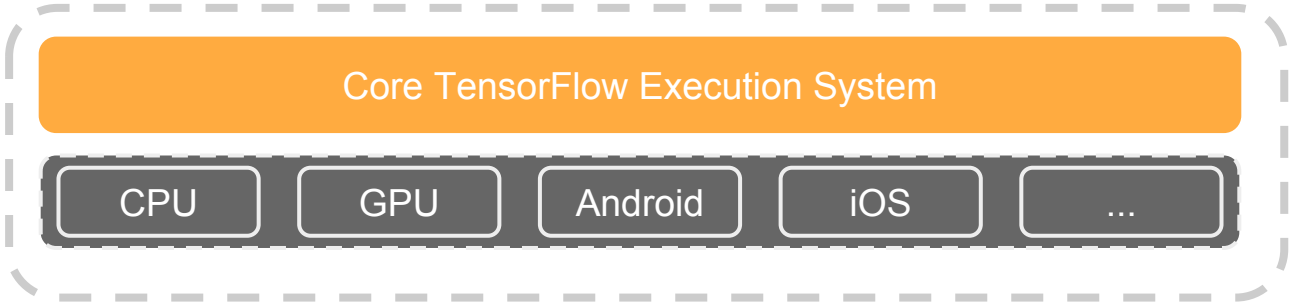
DistBelief (1st system) was great for scalability, and production training of basic kinds of models

Not as flexible as we wanted for research purposes

Better understanding of problem space allowed us to make some dramatic simplifications

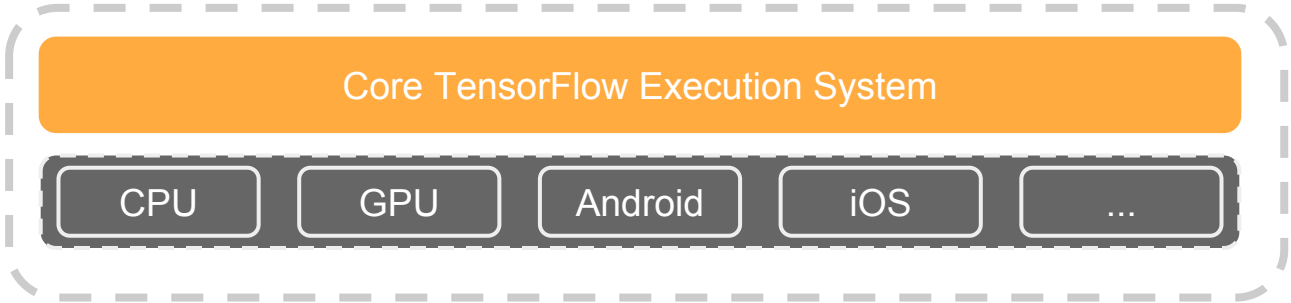
TensorFlow: Expressing High-Level ML Computations

- Core in C++
  - Very low overhead



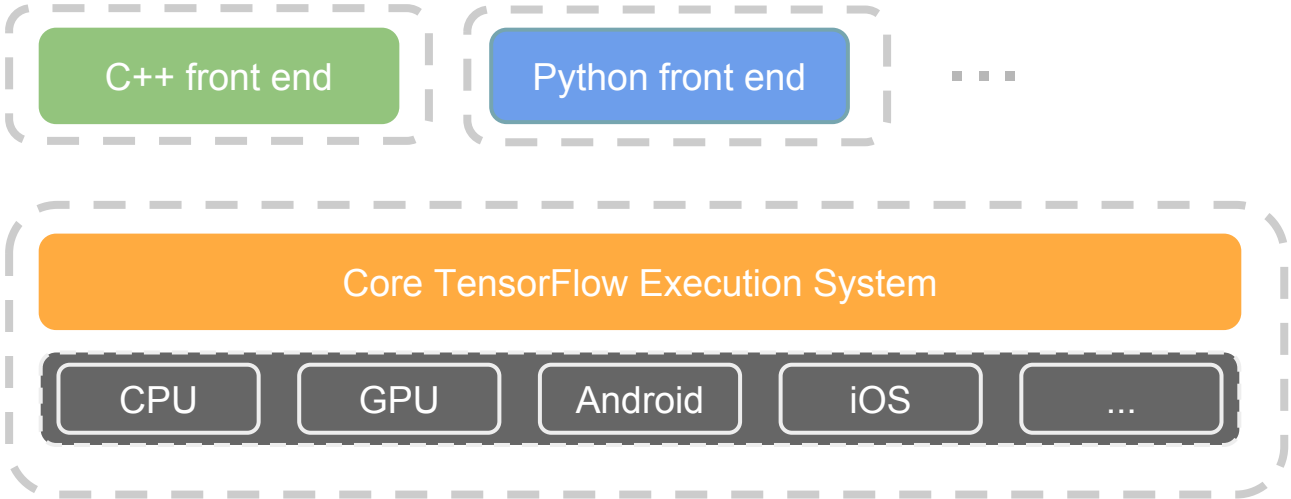
TensorFlow: Expressing High-Level ML Computations

- Core in C++
  - Very low overhead
- Different front ends for specifying/driving the computation
  - Python and C++ today, easy to add more

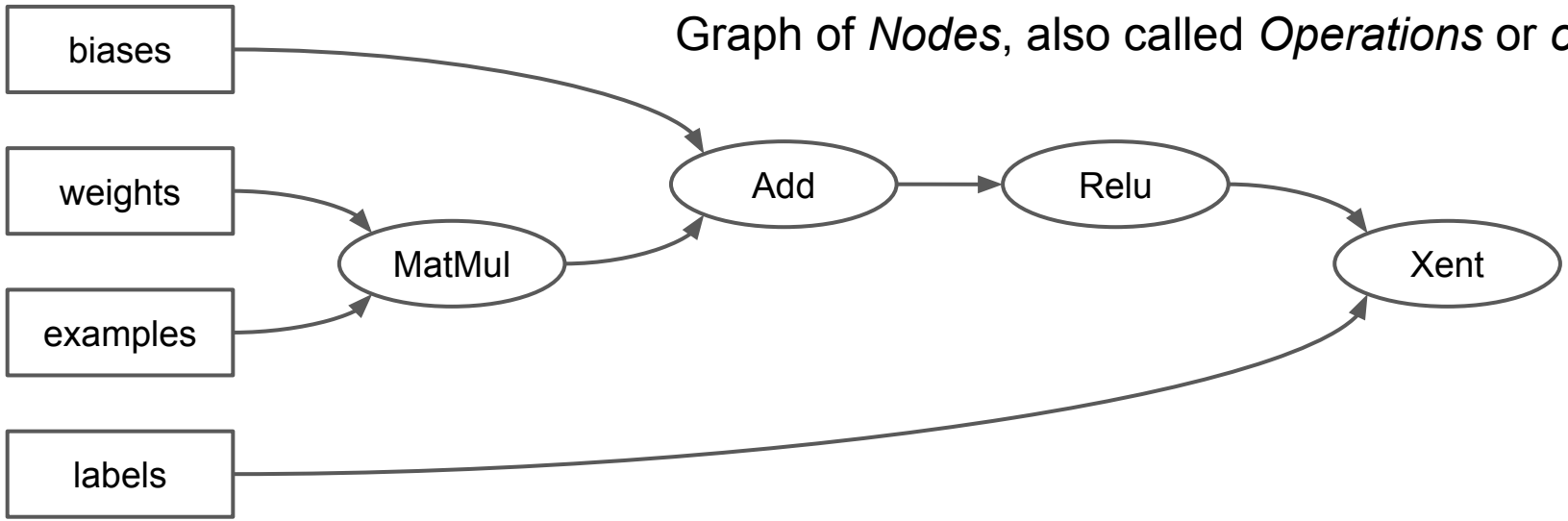


TensorFlow: Expressing High-Level ML Computations

- Core in C++
  - Very low overhead
- Different front ends for specifying/driving the computation
  - Python and C++ today, easy to add more



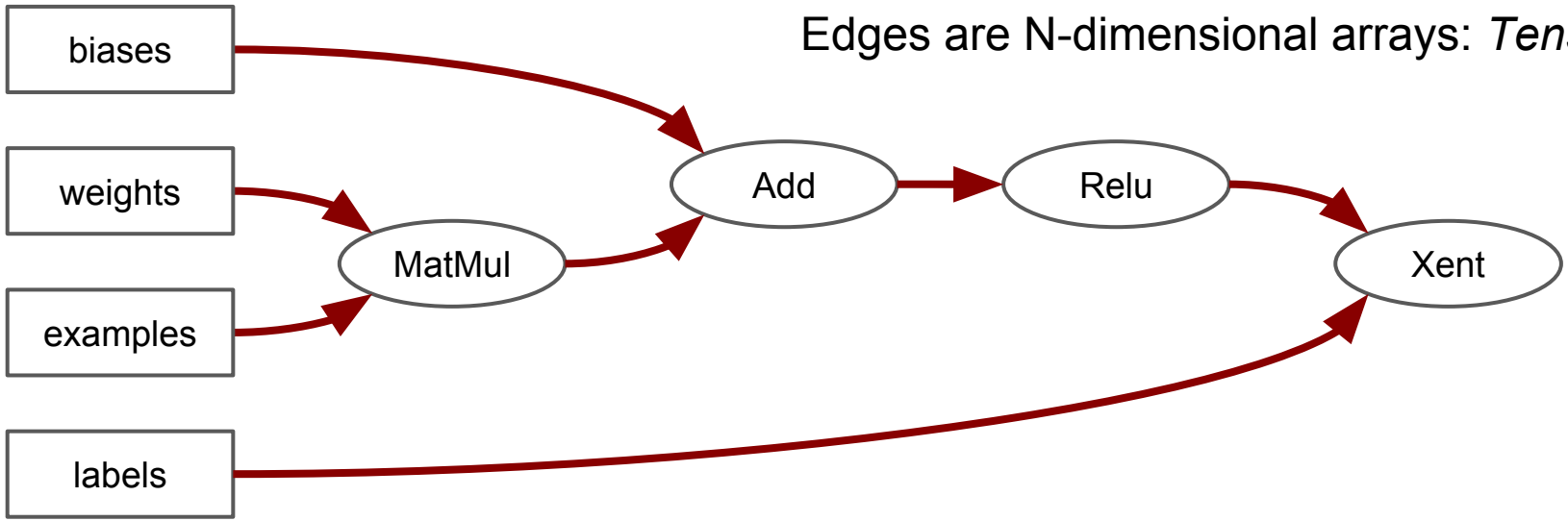
Computation is a dataflow graph



Graph of *Nodes*, also called *Operations* or *ops.*

Computation is a dataflow graph

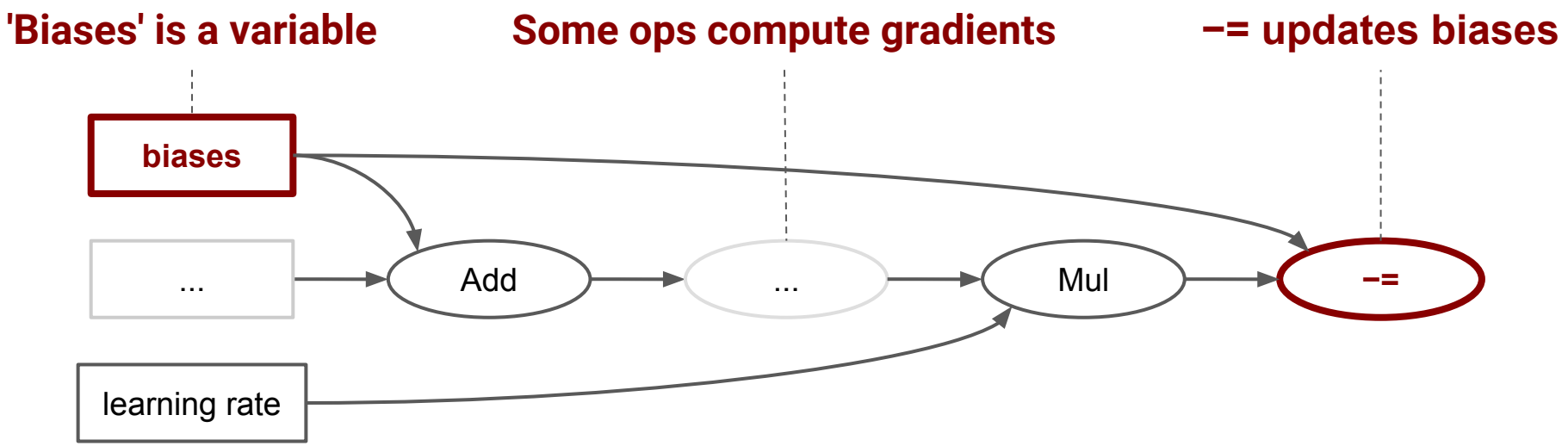
with tensors



Edges are N-dimensional arrays: *Tensors*

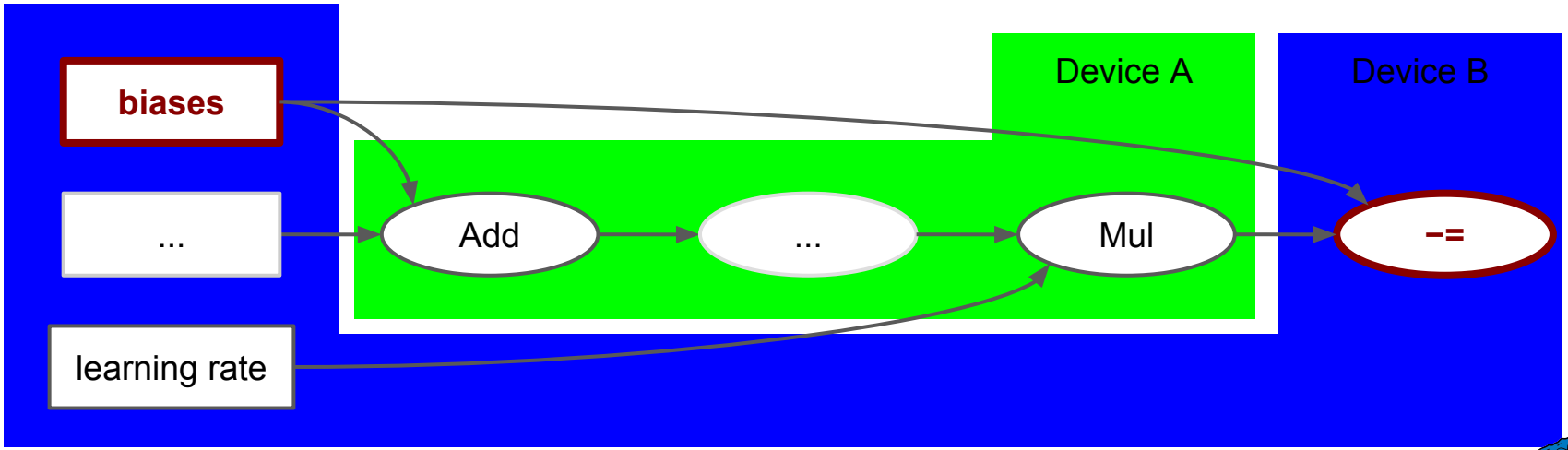
Computation is a dataflow graph

with state

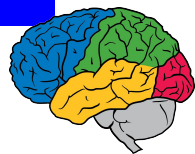


Computation is a dataflow graph

distributed



Devices: Processes, Machines, GPUs, etc



TensorFlow: Expressing High-Level ML Computations

Automatically runs models on range of platforms:

from **phones** ...

to **single machines** (CPU and/or GPUs) …

to **distributed systems** of many 100s of GPU cards

Mobile and Embedded Deployment Desires

- Execution efficiency
- Low power consumption
- Modest size requirements

Quantization: Using Low Precision Integer Math

- Train using 32-bit floats, and after training, convert parameters to quantized 8-bit integer representations
- We have used this in many different applications:
  - Very minor losses in overall model accuracy. E.g.:
    - **~77%** top-1 accuracy for image model with 32-bit float,
      ~76% top-1 accuracy with 8-bit quantized integers
  - 8-bit math gives close to 4X speedup and 4X reduction in model size
  - Saves considerable power, as well

gemmlowp

Support for this open-sourced in `gemmlowp` package:

https://github.com/google/gemmlowp

Efficient GEMM implementations for ARM and x86

Ongoing performance work to make it even better

TensorFlow and Quantized Models

Support for quantized integer kernels

Automated tool coming shortly to

`tensorflow/contrib/quantization/quantize_graph`

Converts TensorFlow model/graph trained using 32-bit floats

- Emits new graph and associated parameter checkpoint
- Uses quantized ops where equivalent ops exist
- Falls back to float when no equivalent quantized op exists

TensorFlow and Mobile Execution

- Android support already there
  - Example app in TensorFlow GitHub repository under:
    - tensorflow/examples/android/...
- iOS support coming shortly:
  - https://github.com/tensorflow/tensorflow/issues/16
  - https://github.com/tensorflow/tensorflow/pull/1631

To Learn More

- Attend Pete Warden's talk tomorrow (Tuesday, May 3), 10:30 to 11:15 AM

  "*TensorFlow: Enabling Mobile and Embedded Machine Intelligence*"

http://www.embedded-vision.com/summit/tensorflow-enabling-mobile-and-embedded-machine-intelligence

How Can You Get Started with Machine Learning?

Four ways, with varying complexity:

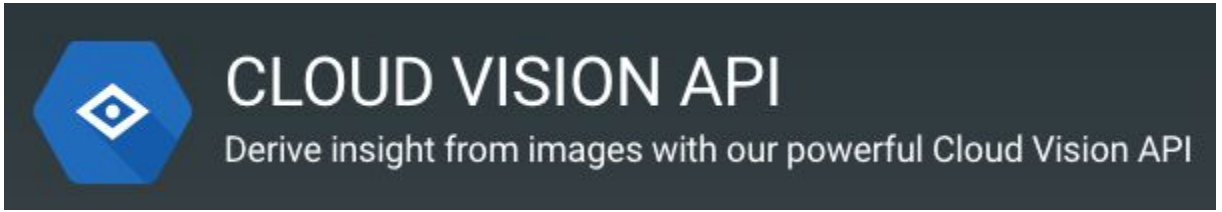
(1) Use a Cloud-based API (Vision, Speech, etc.)
(2) Run your own pretrained model
(3) Use an existing model architecture, and retrain it or fine tune on your dataset
(4) Develop your own machine learning models for new problems

More flexible, but more effort required

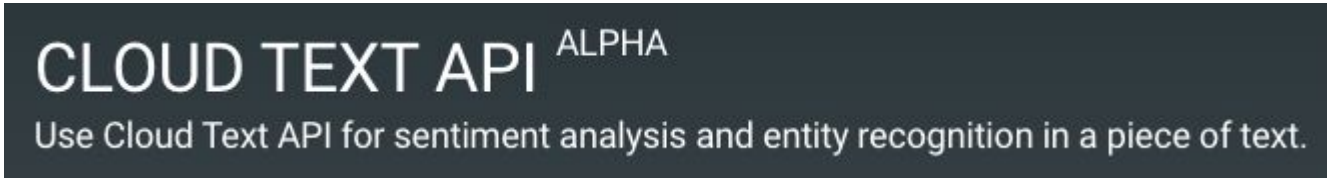
(1) Use Cloud-based APIs

cloud.google.com/translate

cloud.google.com/speech



cloud.google.com/vision

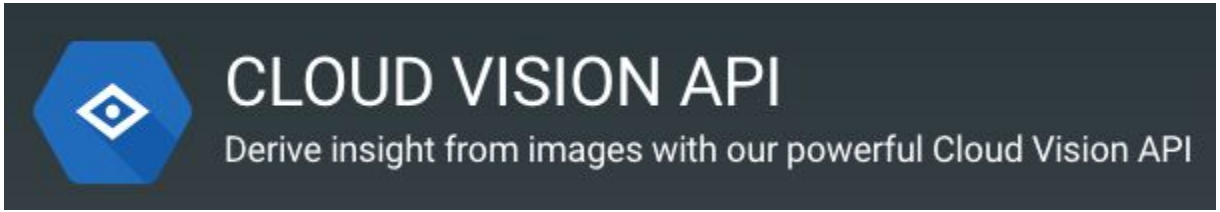


cloud.google.com/text

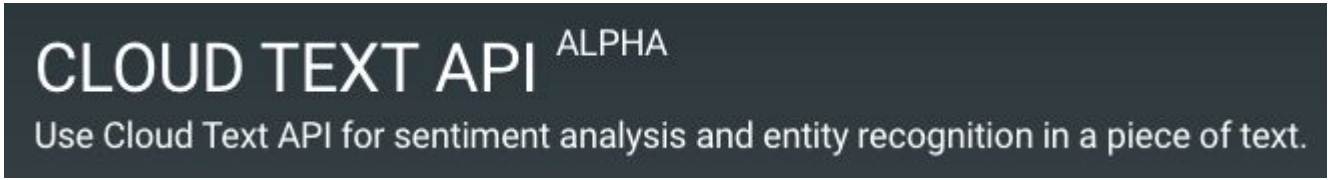
(1) Use Cloud-based APIs

cloud.google.com/translate

cloud.google.com/speech



cloud.google.com/vision



cloud.google.com/text

Google Cloud Vision API

https://cloud.google.com/vision/

"running", "score": 0.99803412,
"marathon", "score": 0.99482006

"joyLikelihood": "VERY_LIKELY"

,"description": "ABIERTO\n",
"local": "es"

Google Cloud Vision API demo

(2) Using a Pre-trained Image Model with TensorFlow

www.tensorflow.org/tutorials/image_recognition/index.html

Usage with Python API

`classify_image.py` downloads the trained model from `tensorflow.org` when the program is run for the first time. You'll need about 200M of free space available on your hard disk.

The following instructions assume you installed TensorFlow from a PIP package and that your terminal resides in the TensorFlow root directory.

```
cd tensorflow/models/image/imagenet
python classify_image.py
```

The above command will classify a supplied image of a panda bear.

If the model runs correctly, the script will produce the following output:

```
giant panda, panda, panda bear, coon bear, Ailuropoda melanoleuca (score = 0.88493)
indri, indris, Indri indri, Indri brevicaudatus (score = 0.00878)
lesser panda, red panda, panda, bear cat, cat bear, Ailurus fulgens (score = 0.00317)
custard apple (score = 0.00149)
earthstar (score = 0.00127)
```

Using a Pre-trained Image Model with TensorFlow on Android

https://github.com/tensorflow/tensorflow/tree/master/tensorflow/examples/android

README.md

Tensorflow Android Camera Demo

This folder contains a simple camera-based demo application utilizing Tensorflow.

Description

This demo uses a Google Inception model to classify camera frames in real-time, displaying the top results in an overlay on the camera image.

To build/install/run

As a prerequisite, Bazel, the Android NDK, and the Android SDK must all be installed on your system.

1. Get the recommended Bazel version listed at:
   https://www.tensorflow.org/versions/master/get_started/os_setup.html#source
2. The Android NDK may be obtained from: http://developer.android.com/tools/sdk/ndk/index.html
3. The Android SDK and build tools may be obtained from: https://developer.android.com/tools/revisions/build-tools.html

The Android entries in `<workspace_root>/WORKSPACE` must be uncommented with the paths filled in appropriately depending on where you installed the NDK and SDK. Otherwise an error such as: "The external label '//external:android/sdk' is not bound to anything" will be reported.

The TensorFlow `GraphDef` that contains the model definition and weights is not packaged in the repo because of its size. Instead, you must first download the file to the `assets` directory in the source tree:

```
$ wget https://storage.googleapis.com/download.tensorflow.org/models/inception5h.zip -O /tmp/inception5h.z
```

(3) Training a Model on Your Own Image Data

www.tensorflow.org/versions/master/how_tos/image_retraining/index.html

(4) Develop your own machine learning models

https://www.tensorflow.org/versions/master/get_started/basic_usage.html

TensorFlow ™ GET STARTED

Overview

TensorFlow is a programming system in which you represent computations as graphs. Nodes in the graph are called ops (short for operations). An op takes zero or more `Tensors`, performs some computation, and produces zero or more `Tensors`. A `Tensor` is a typed multi-dimensional array. For example, you can represent a mini-batch of images as a 4-D array of floating point numbers with dimensions `[batch, height, width, channels]`.

A TensorFlow graph is a description of computations. To compute anything, a graph must be launched in a `Session`. A `Session` places the graph ops onto `Devices`, such as CPUs or GPUs, and provides methods to execute them. These methods return tensors produced by ops as numpy `ndarray` objects in Python, and as `tensorflow::Tensor` instances in C and C++.

The computation graph

TensorFlow programs are usually structured into a construction phase, that assembles a graph, and an execution phase that uses a session to execute ops in the graph.

What Does the Future Hold?

Deep learning usage will continue to grow and accelerate:

- Across more and more fields and problems:
  - robotics, self-driving vehicles, ...
  - health care
  - video understanding
  - dialogue systems
  - personal assistance
  - ...

Combining Vision with Robotics

“Deep Learning for Robots: Learning from Large-Scale Interaction”, Google Research Blog, March, 2016

“Learning Hand-Eye Coordination for Robotic Grasping with Deep Learning and Large-Scale Data Collection”, Sergey Levine, Peter Pastor, Alex Krizhevsky, & Deirdre Quillen, Arxiv, arxiv.org/abs/1603.02199

Conclusions

Deep neural networks are making significant strides in understanding:
In speech, vision, language, search, …

If you're not considering how to use deep neural nets to solve your vision or understanding problems, **you almost certainly should be**

Pre-trained models or pre-trained APIs are a low overhead way of starting to explore

TensorFlow makes it easy for everyone to experiment with these techniques

- Highly scalable design allows faster experiments, accelerates research
- Easy to share models and to publish code to give reproducible results
- Ability to go from research to production within same system

Further Reading

- Le, Ranzato, Monga, Devin, Chen, Corrado, Dean, & Ng. *Building High-Level Features Using Large Scale Unsupervised Learning*, ICML 2012. research.google.com/archive/unsupervised_icml2012.html
- Dean, *et al.*, *Large Scale Distributed Deep Networks,* NIPS 2012, research.google.com/archive/large_deep_networks_nips2012.html.
- Sutskever, Vinyals, & Le, *Sequence to Sequence Learning with Neural Networks*, NIPS, 2014, arxiv.org/abs/1409.3215.
- Vinyals, Toshev, Bengio, & Erhan. *Show and Tell: A Neural Image Caption Generator*. CVPR 2015. arxiv.org/abs/1411.4555
- Szegedy, Vanhoucke, Ioffe, Shlens, & Wojna. Rethinking the Inception Architecture for Computer Vision. arxiv.org/abs/1512.00567
- TensorFlow white paper, tensorflow.org/whitepaper2015.pdf (clickable links in bibliography)

research.google.com/people/jeff

research.google.com/pubs/MachineIntelligence.html

Questions?