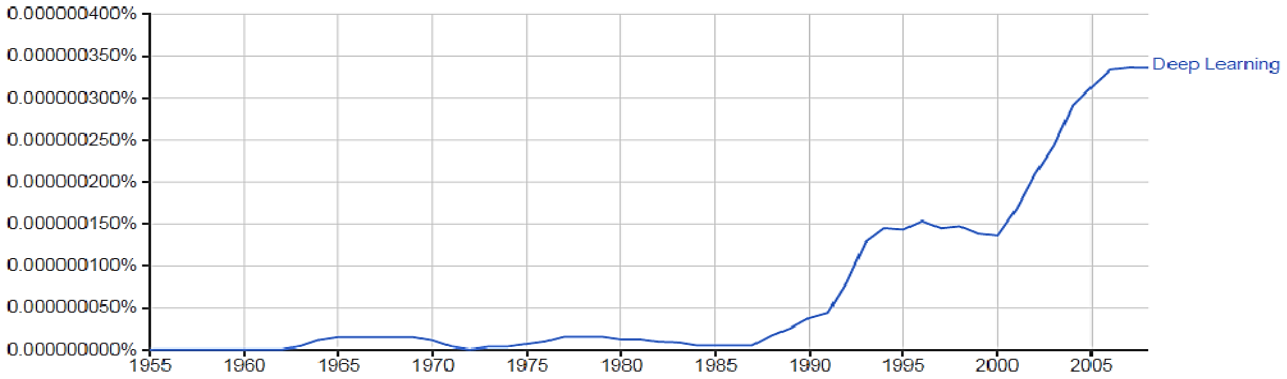
Deep Learning using TensorFlow and TensorFlow-Slim

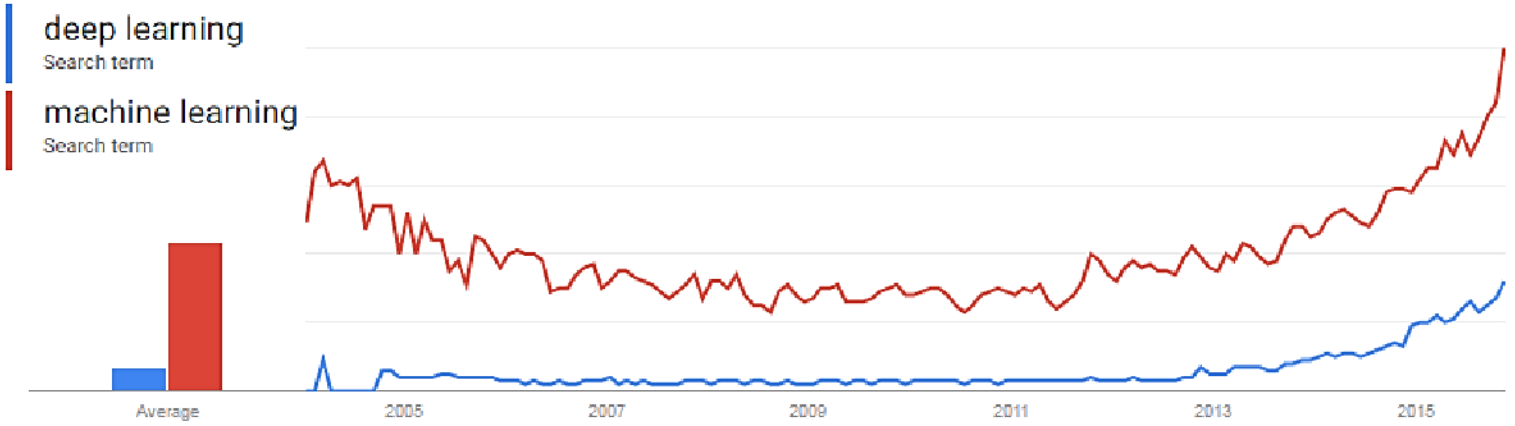
Dipendra Jha
Northwestern University
dipendra009@gmail.com
https://www.linkedin.com/in/dipendra009

Outline

- Deep Learning: Hype or Reality?
- Deep Learning: Why is it happening now?
- Deep Learning: How to define it?
- A quick start with TensorFlow
- Why to use TensorFlow-Slim?
- Training Demonstration using TensorFlow and TF-Slim

Interest: Google Trends

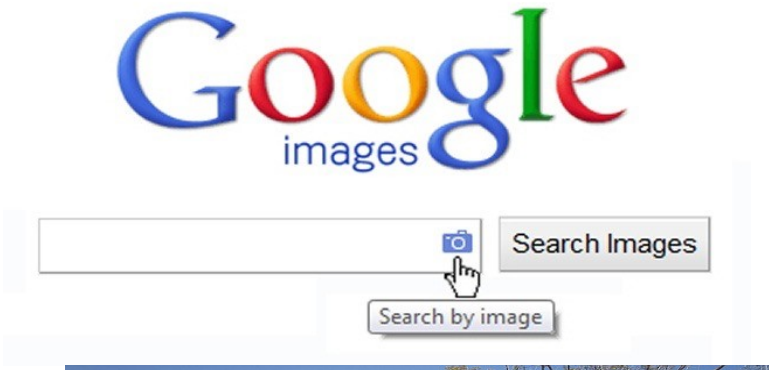


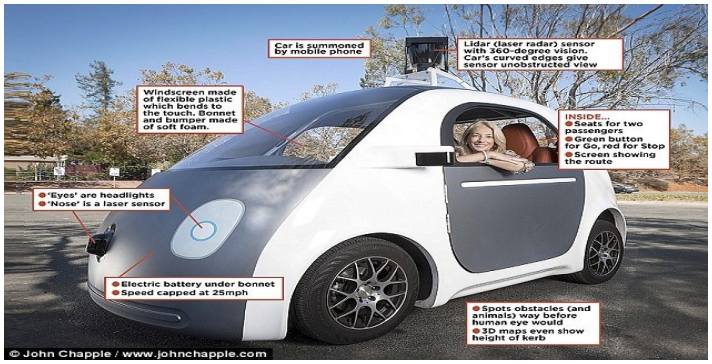


Big Players

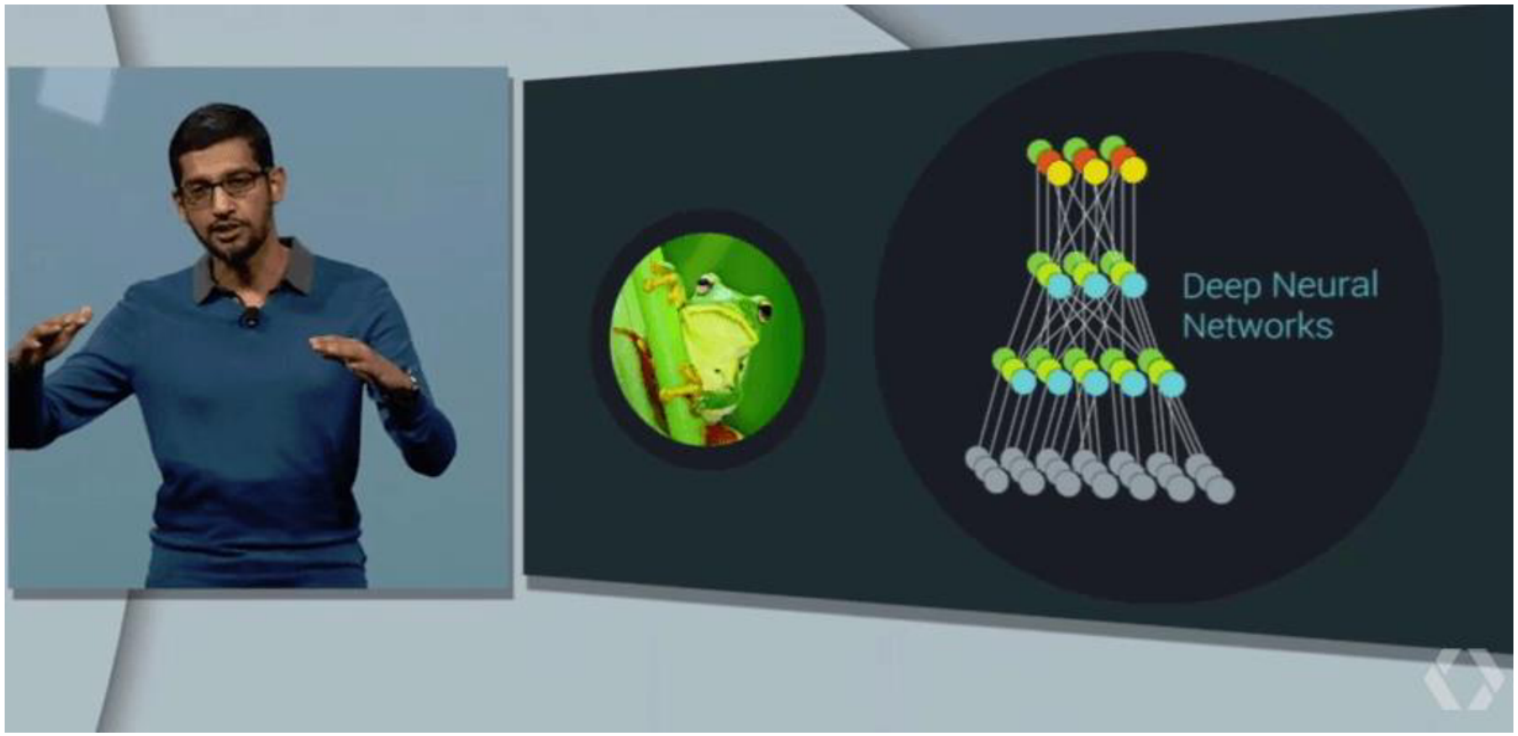
Just with in Google

- Search
- Search by image
- Driveless cars
- Youtube recommendation
  - Videos
  - Thumbnails
- Maps
  - Reading street addresses



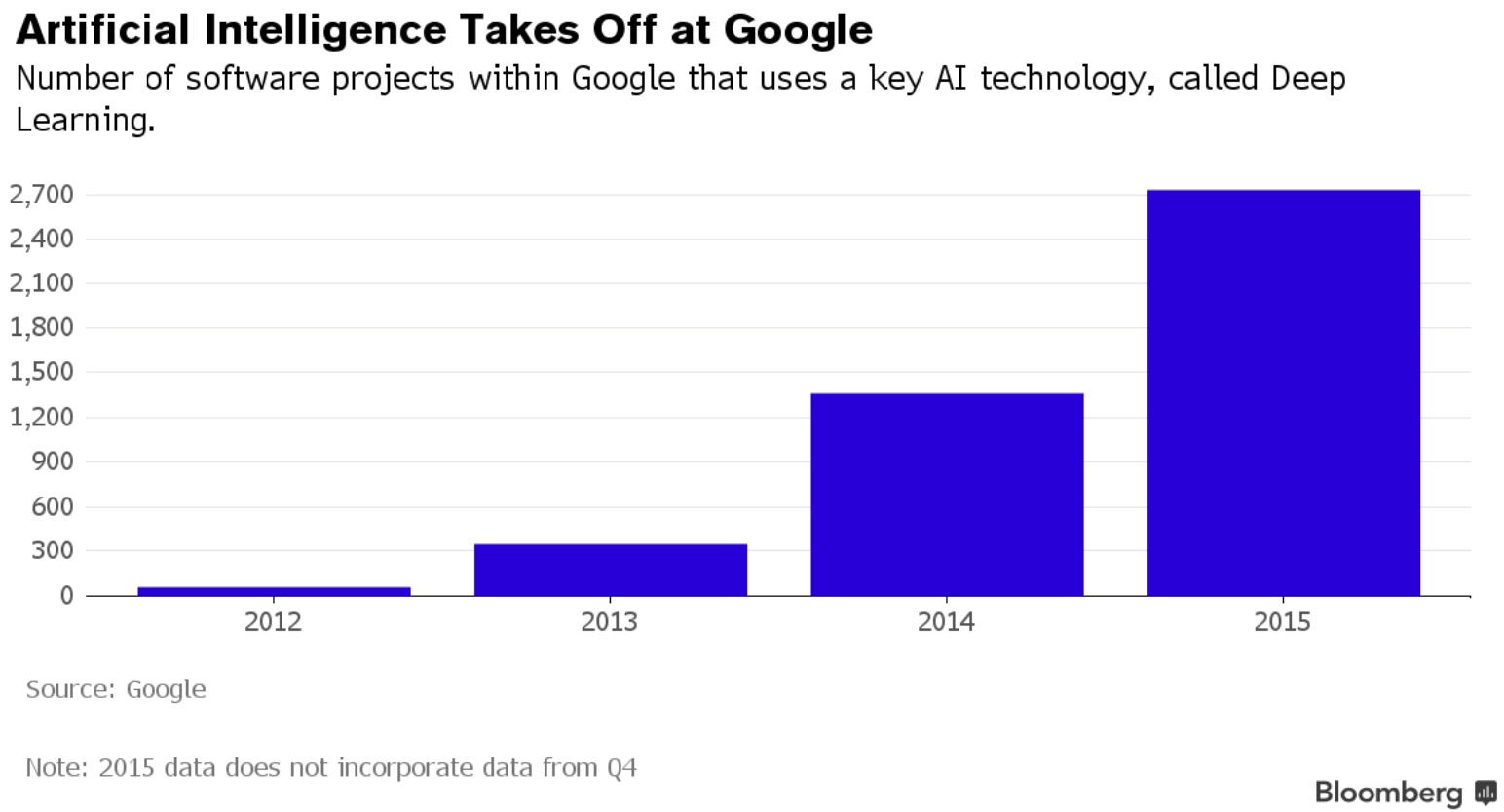


Google: Hype or Reality?



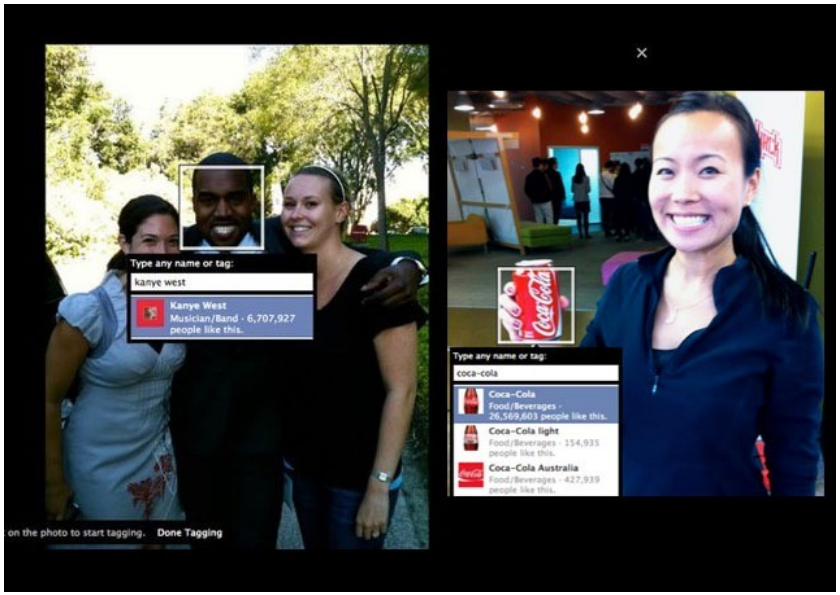
Machine learning is a core transformative way by which we are **rethinking everything** we are doing – *Sundar Pichai (CEO Google)*

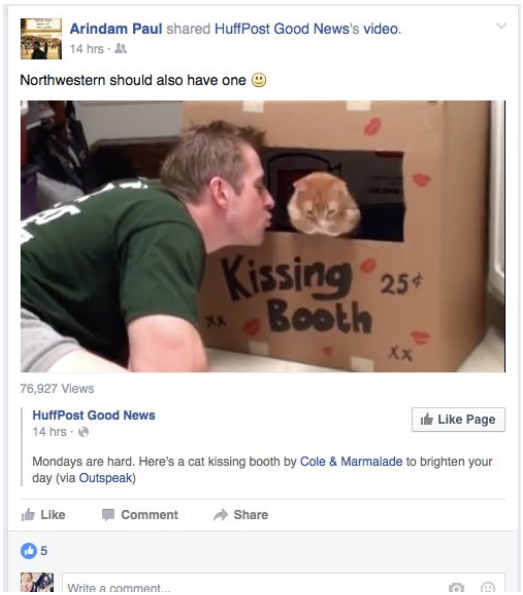
Google: Hype or Reality?



Facebook

- Every image on Facebook goes into two deep networks
  - Tagging
  - Determining close friends





Startups


vicarious
enlitic
clarifai
SKYMIND
SIGNALSENSE
nervana
nnaisense
ersatz labs
cortica
In Every Image
Numenta
sentient
OpenAI
MetaMind
DEEPMIND
AlchemyAPI
An IBM Company
wit.ai
DNNresearch
Acquired

Why Deep Learning? Applications

Speech Recognition

Computer Vision

Natural Language Processing

DNNs better than humans at image recognition

ImageNet

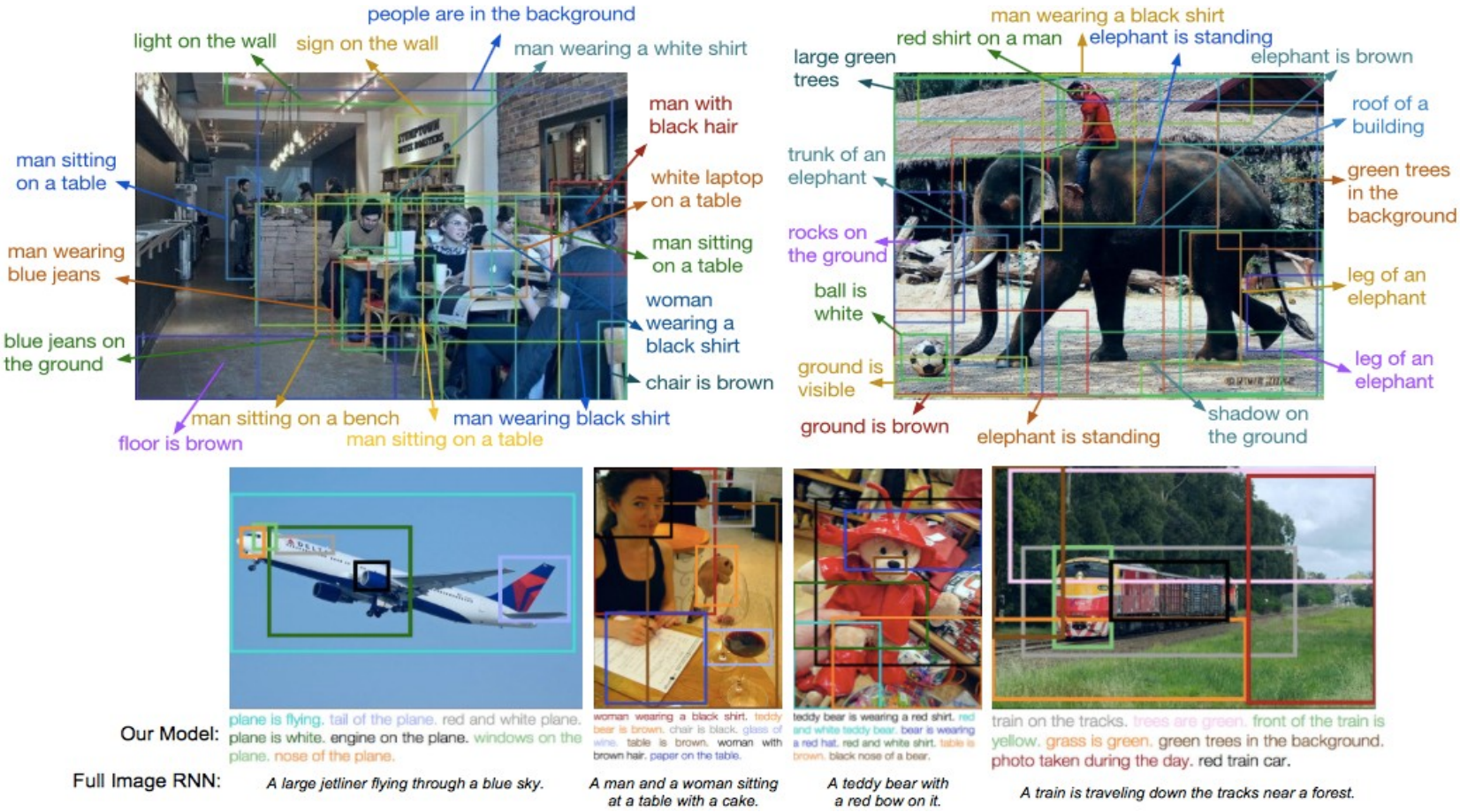
1000 categories

1.3M images

Human error: 5%

DNN: 3%

Image Captioning [DenseCap]



Deep Reinforcement Learning

AlphaGo

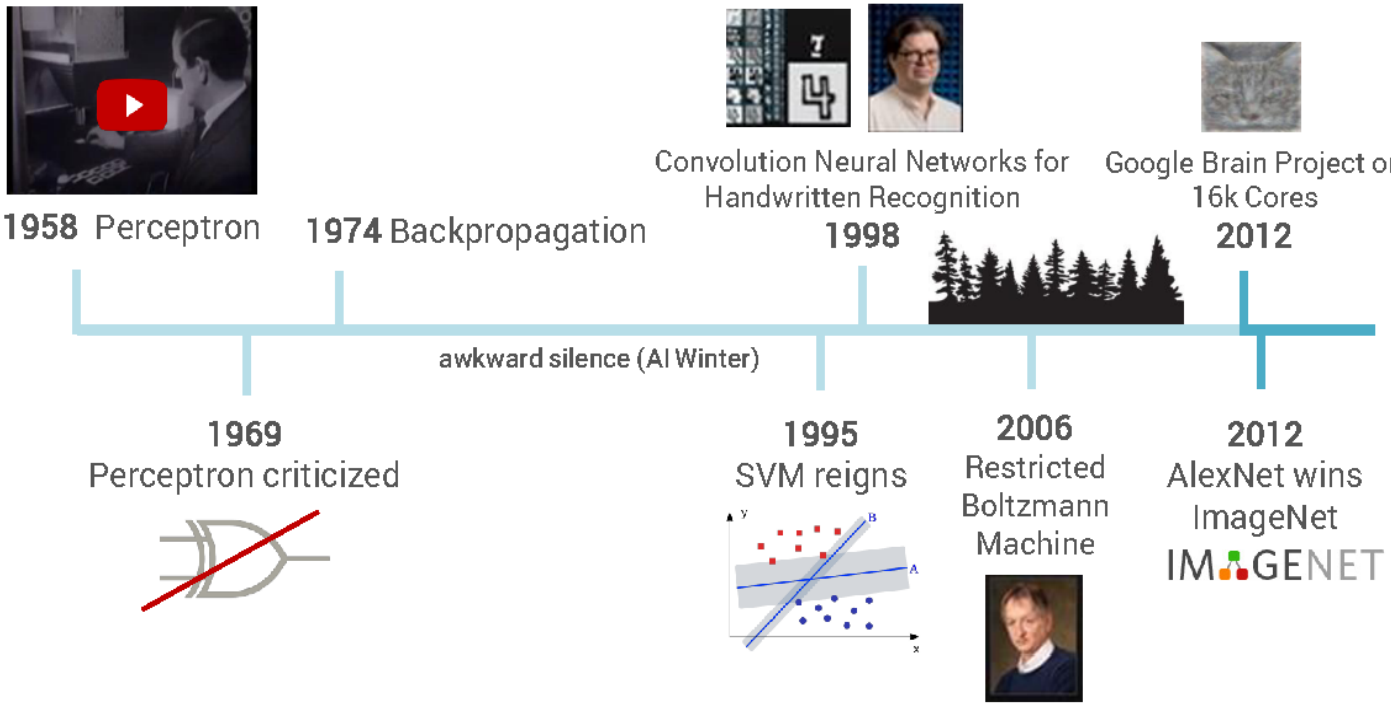
An obvious favorite baby

- Front page stories in NYT, Wired
- Big acquisitions
  - Google acquired DeepMind ($500 million), *Geoff Hinton*
  - Facebook launched FAIR, *Yan LeCun*
  - Baidu Silicon Valley Research Lab, *Andrew Ng*
  - Salesforce acquired MetaMind
- Workshops and sessions in top ML conferences
  - NIPS, ICML, CVPR, KDD, AAAI, ICLR…
- Major universities offer to teach it
- 99% papers appears on arxiv

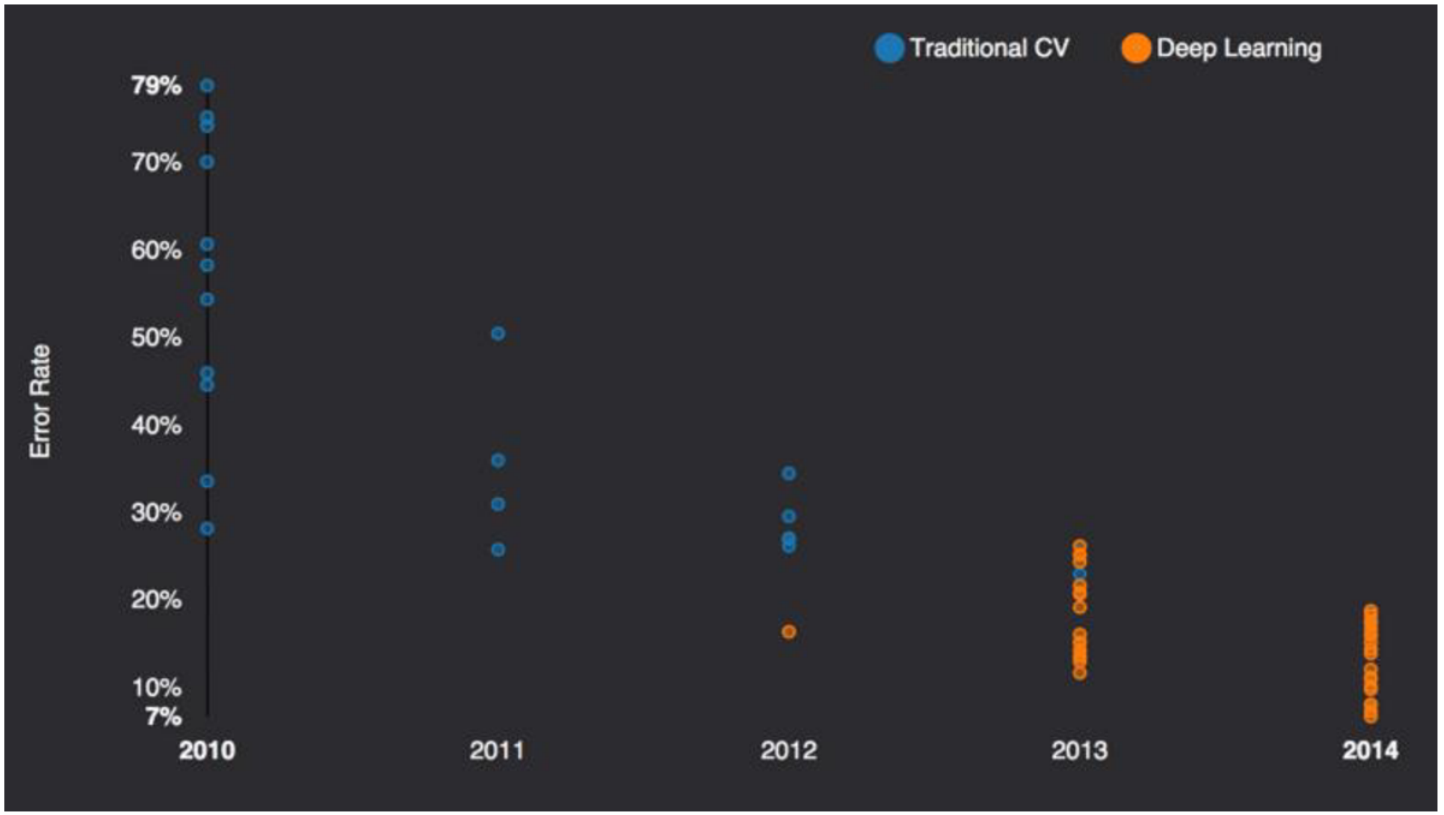
Outline

- Deep Learning: Hype or Reality?
- Deep Learning: Why is it happening now?
- Deep Learning: How to define it?
- A quick start with TensorFlow
- Why to use TensorFlow-Slim?
- Training Demonstration using TensorFlow and TF-Slim

A Brief History

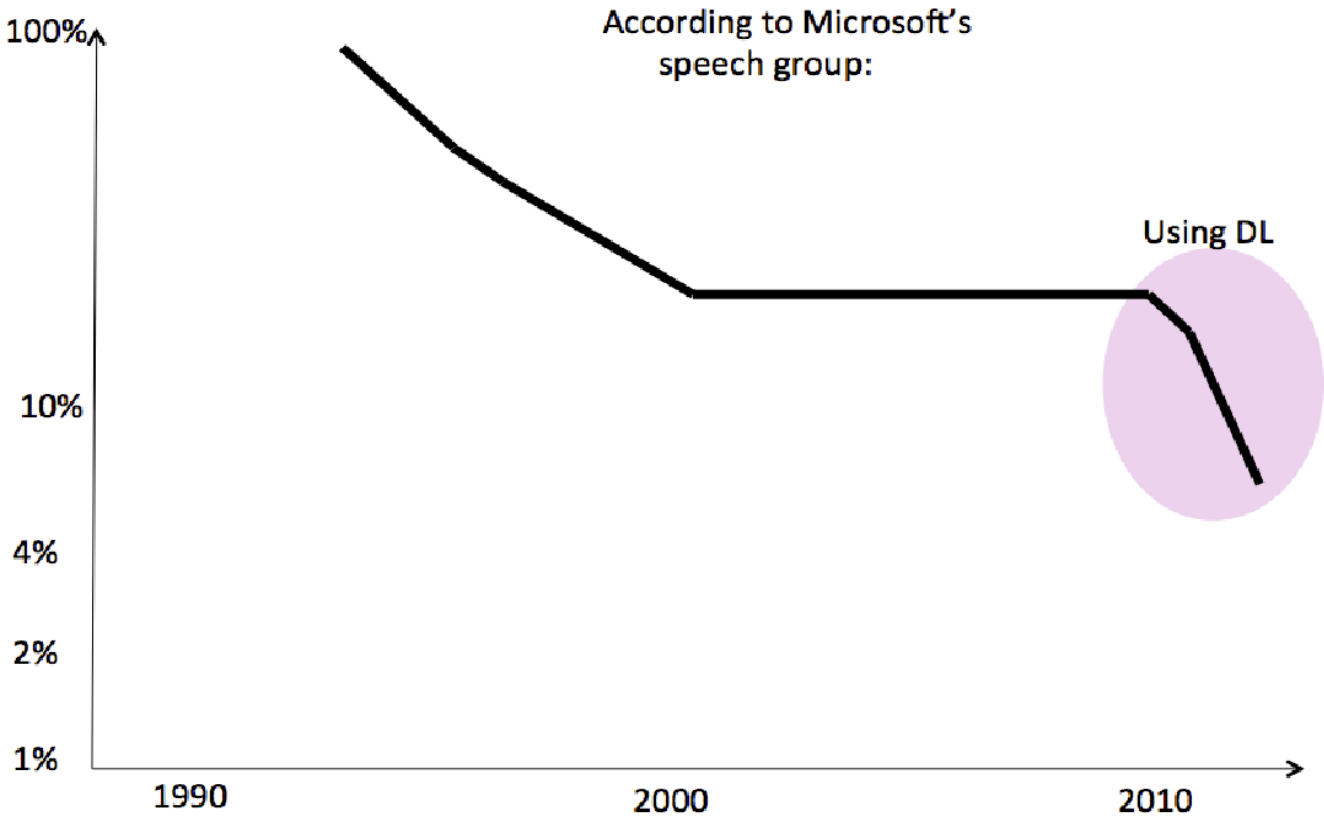


The Big Bang



ImageNet: The "computer vision World Cup"

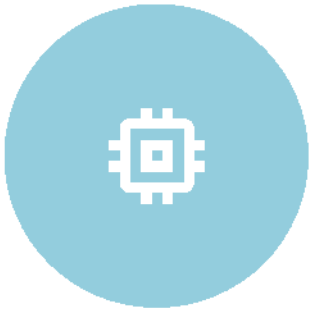
The Big Bang



Deep Learning in **Speech Recognition**

What Changed?
Old wine in new bottles

Big Data
(Digitalization)



Computation
(Moore's Law, GPUs)

Algorithmic
Progress

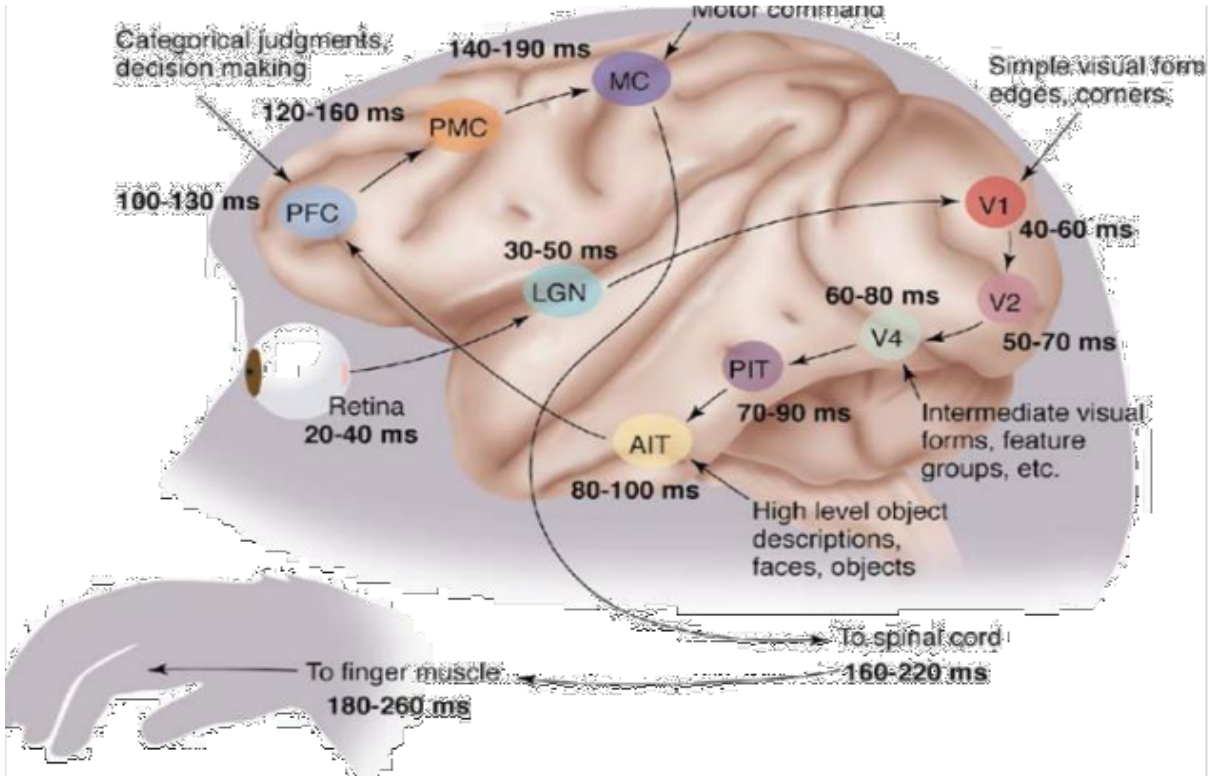
Outline

- Deep Learning: Hype or Reality?
- Deep Learning: Why is it happening now?
- Deep Learning: How to define it?
- A quick start with TensorFlow
- Why to use TensorFlow-Slim?
- Training Demonstration using TensorFlow and TF-Slim

Neural Networks

- A branch of machine learning inspired by how the brain works.
- “Build learning algorithms that mimic the brain”.
- “A computer system modeled on the human brain and nervous system”

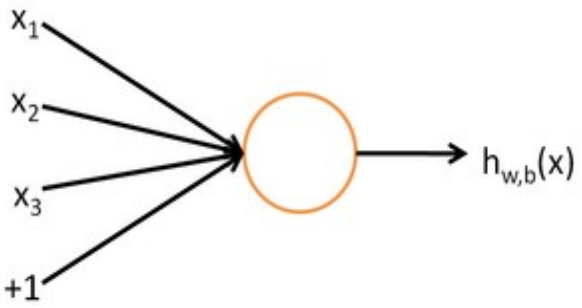
Inspired by the Brain



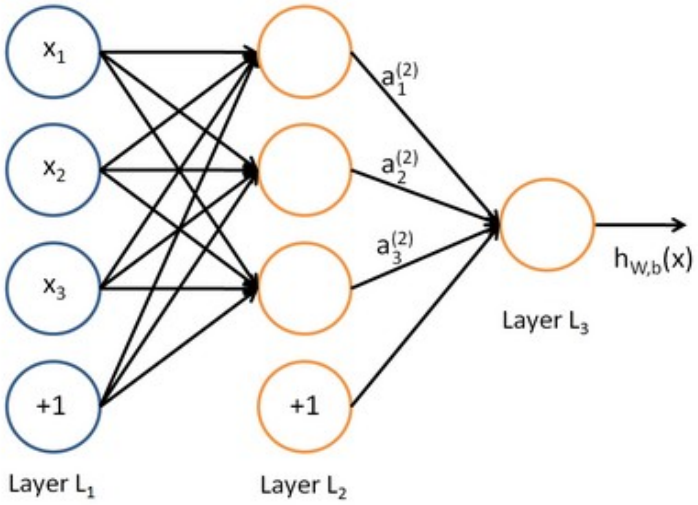
The first **hierarcy of neurons** that receives information in the visual cortex are sensitive to specific edges while brain regions further down the visul pipeline are sensitive to more complex structures such as faces.

(Artificial) Neural Networks and training

A "neuron"



A small neural net



Training of a NN

Loop until tired:

1. **Sample** a batch of data.
2. **Forward** it through the network to get predictions.
3. **Backprop** the errors.
4. **Update** the weights.

What is Deep Learning?

- **Part of the machine learning** field of learning representation of data. Exceptional effective at learning patterns.

- Utilizes learning algorithms that derive meaning out of data by using a **hierarchy** of multiple layers that **mimic the neural networks of our brain**.

- If you provide the system tons of information, it begins to understand it and respond in useful ways.

How can we define it?

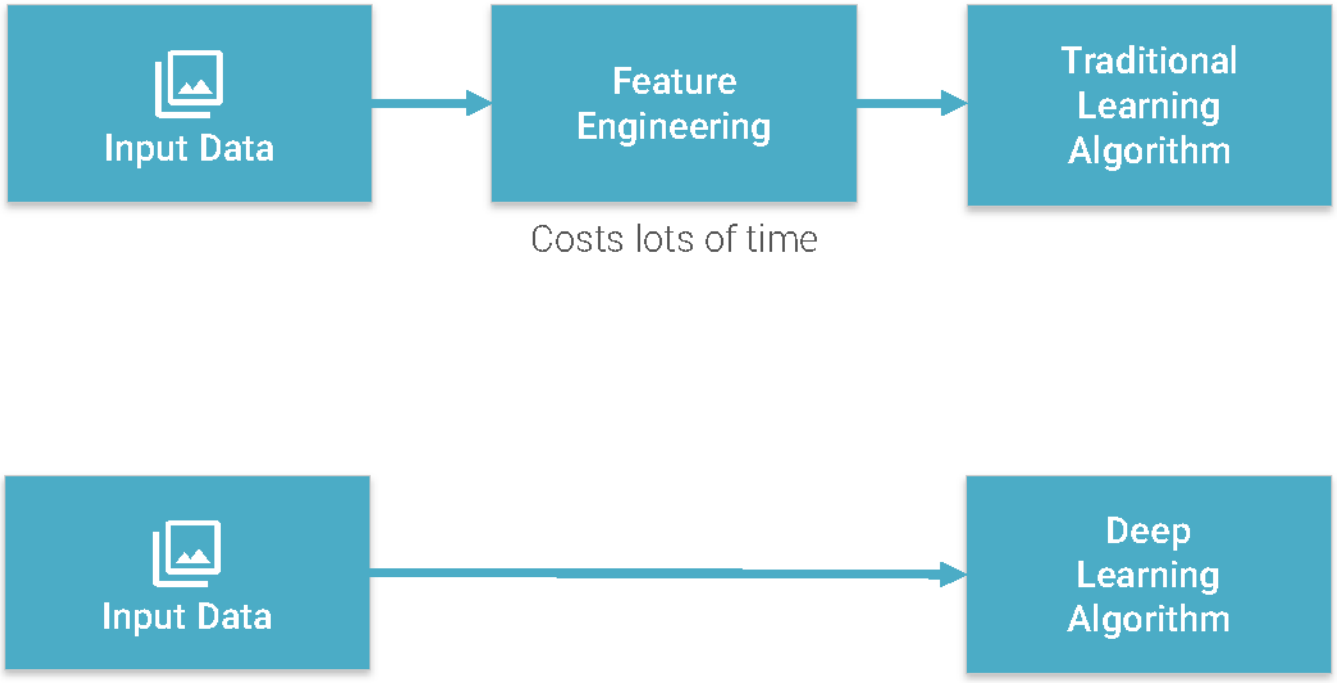
- Deep learning: the most cutting edge ML/AI research

- ”A method which makes predictions by using **a sequence of non-linear processing stages**. The resulting intermediate representations can be interpreted as feature hierarchies and the whole system is jointly learned from data.” -Facebook Research

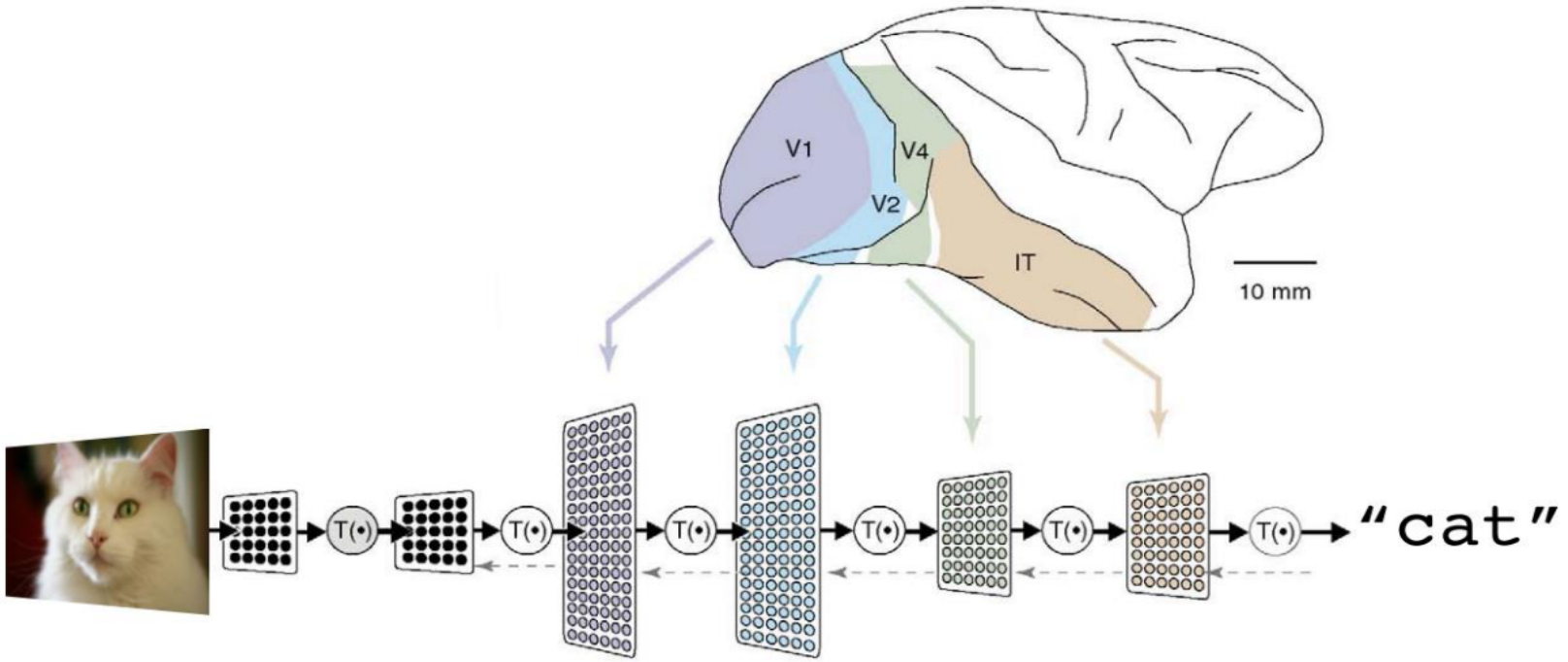
- “Machines that learn to represent the world”

- “End-to-end machine learning.” - Yann LeCun

No more feature engineering

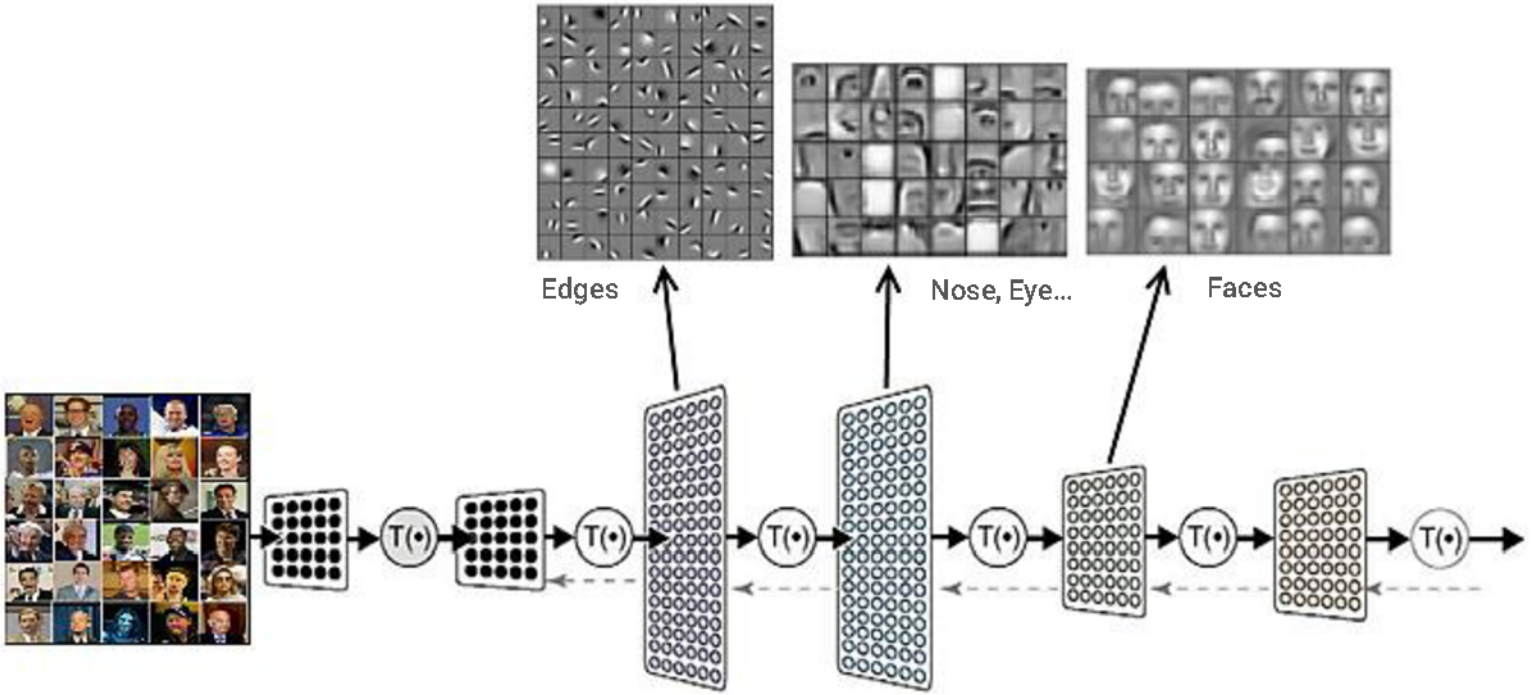


Architecture



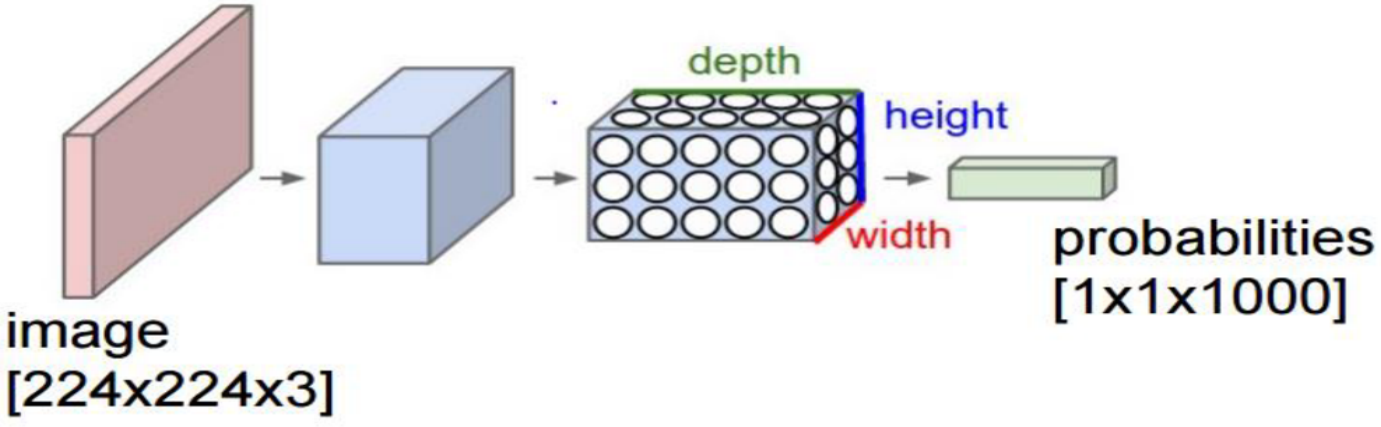
A deep neural Network consists of a **hierarchy of layers**, whereby each layer **transforms the input data** into more abstract representations (e.g.) edge -> nose-> face. The output layer combines those features to make predictions.

What did it learn?

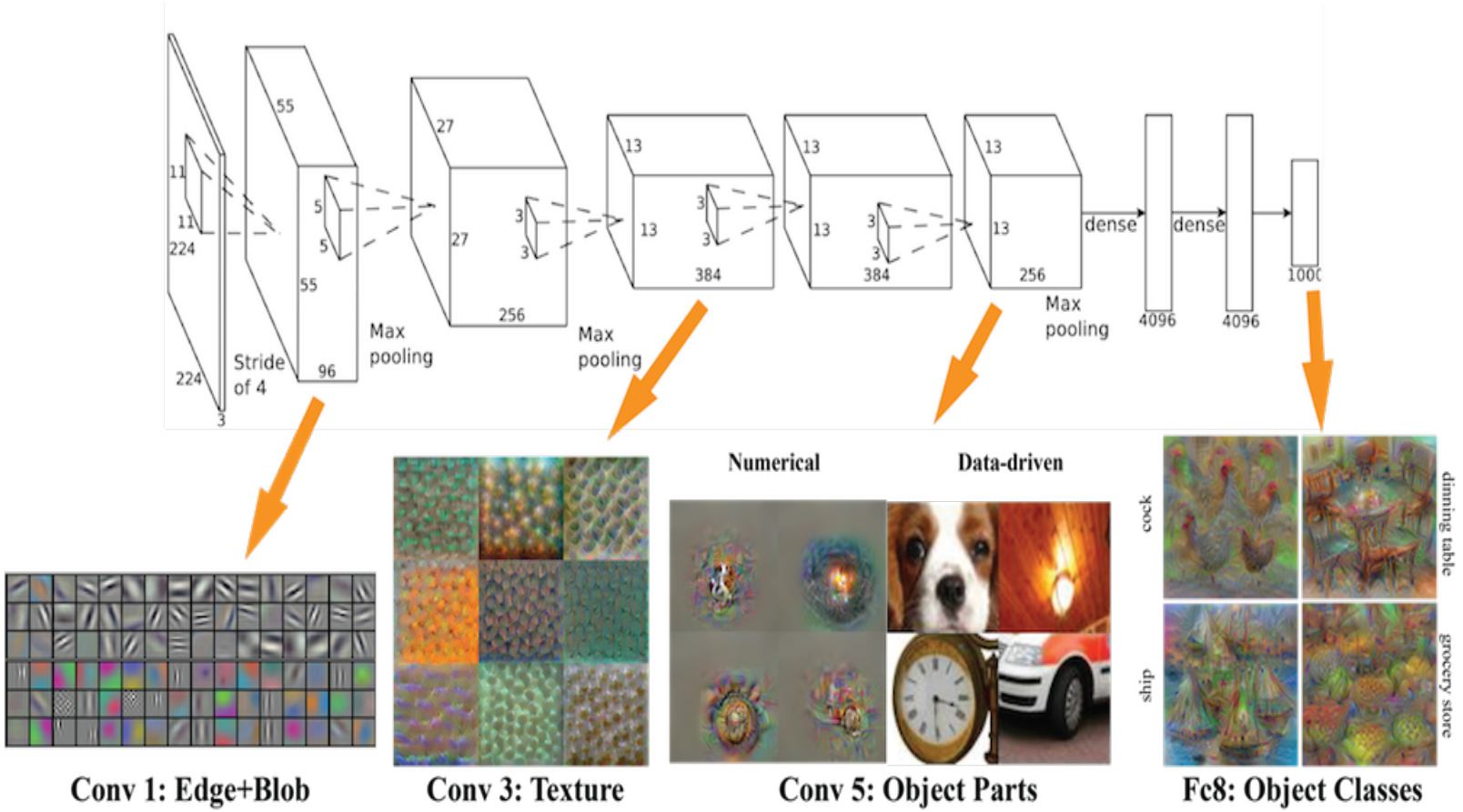


Convolutional Neural Networks (CNNs)

- Inspired by the human visual system
- Emulate the visual processing system of visual cortex
- Each layer takes a 3D volume of numbers and outputs a 3D volume of numbers.



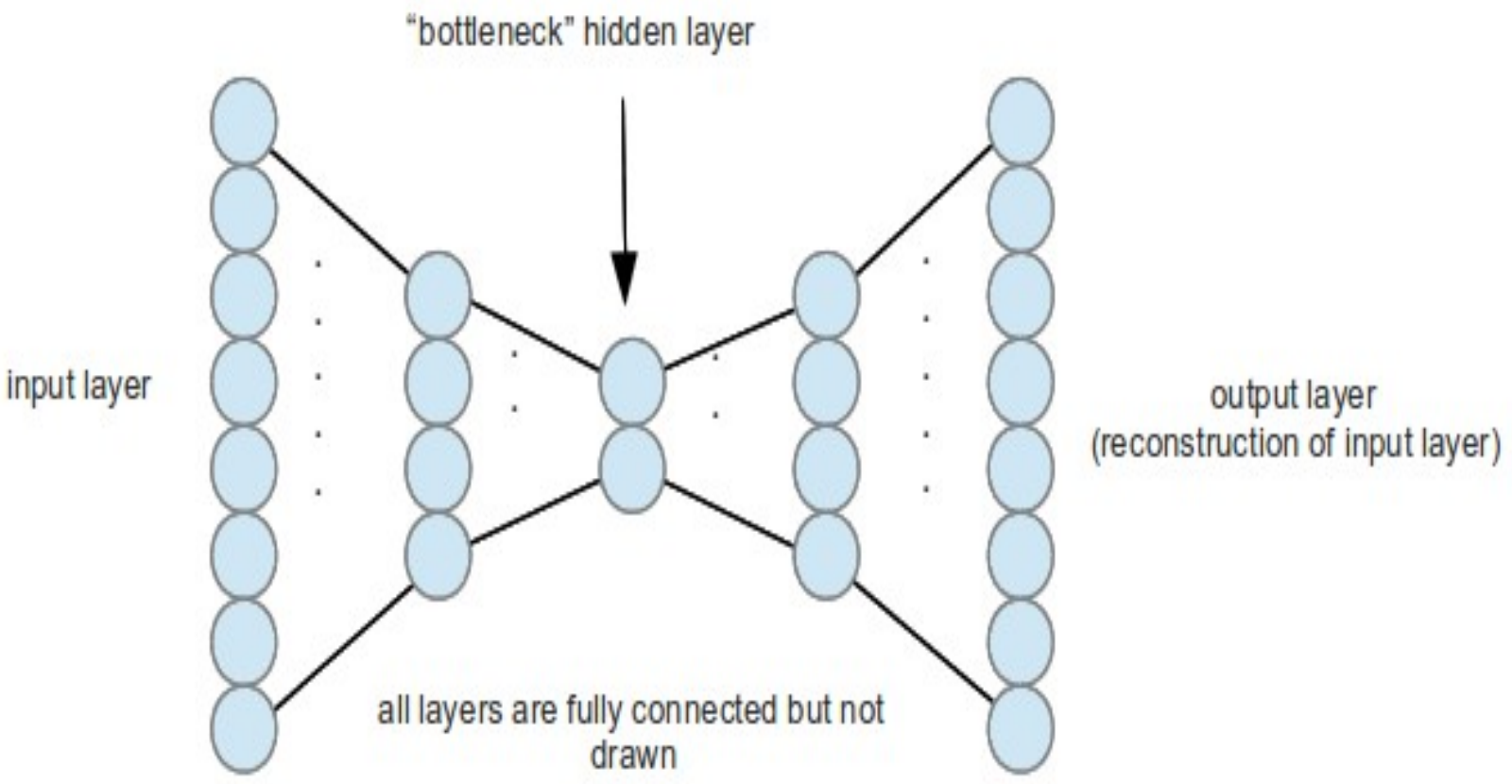
Convolutional Neural Networks



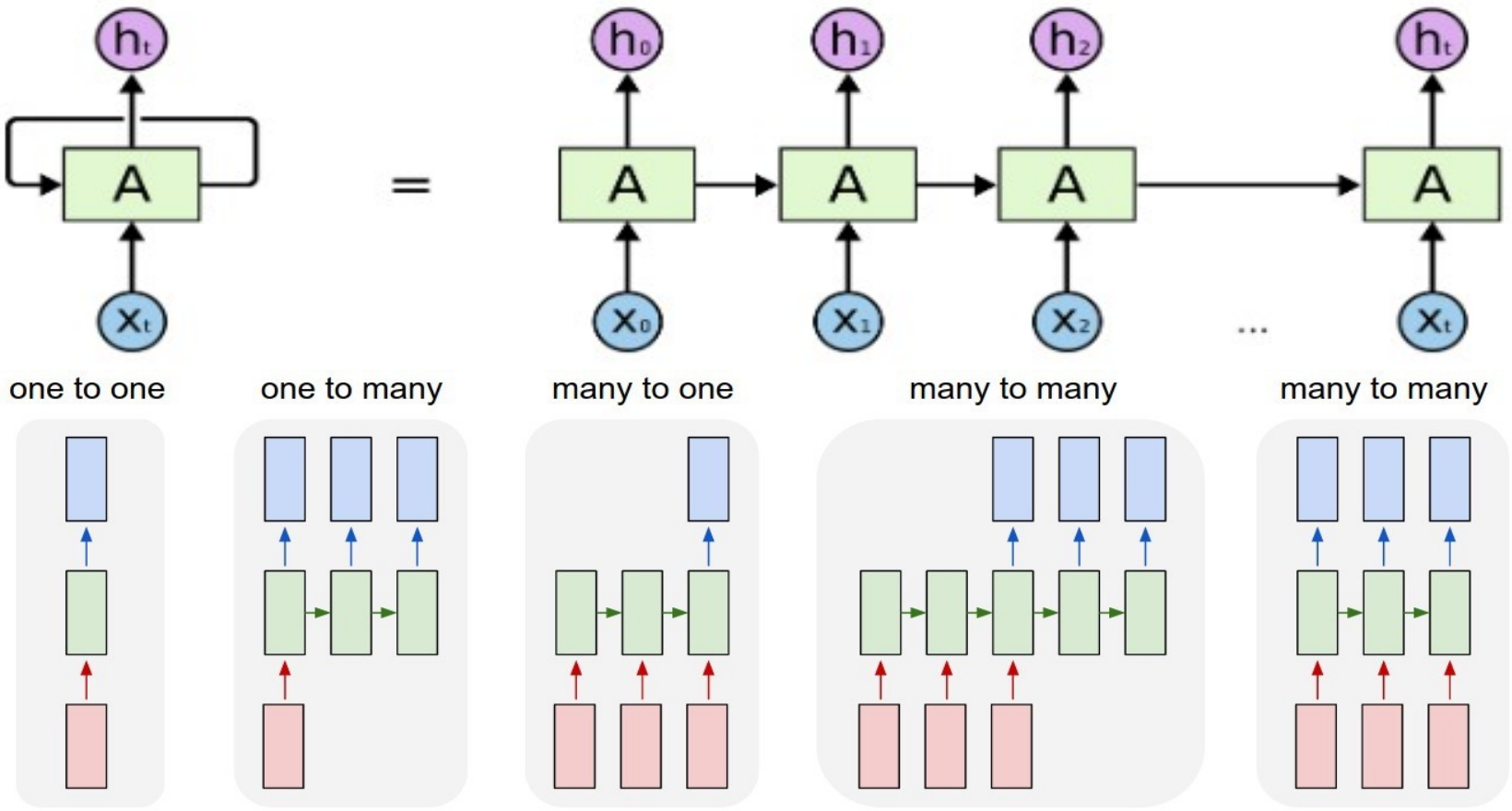
Convolutional Neural Networks (CNNs)

- **Input**: 4D tensor of shape [mini-batch size, number of input feature maps, image heights, image width]
- **Weights**: 4D tensor of shape[number of feature maps at layer m, number of feature maps at layer m-1, filter height, filter width]
- **MaxPooling**: outputs maximum value for each overlapping subregion
- **Depth**: Number of filters we would like to use
- **Padding**: Pad the input volume with zeroes around the border
- **Stride:** Number of pixels with which we slide the filter
- https://cs231n.github.io/assets/conv-demo/index.html

Auto-encoder

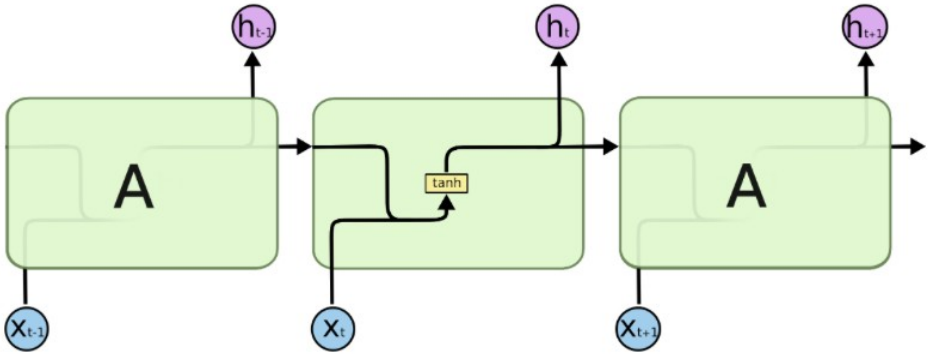


Recurrent Neural Networks

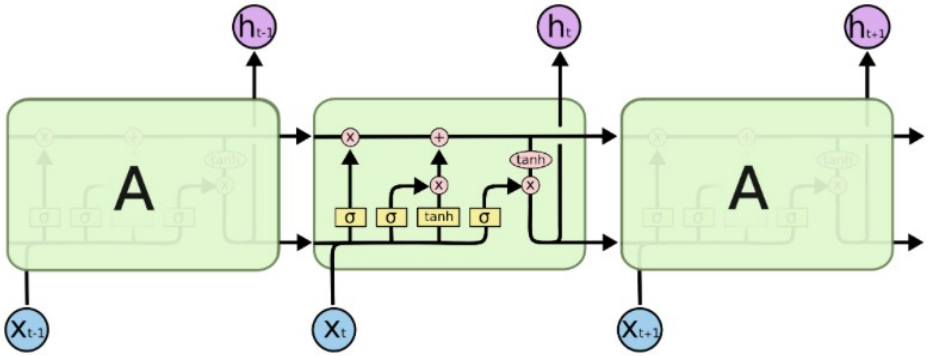


Long short Term Memory

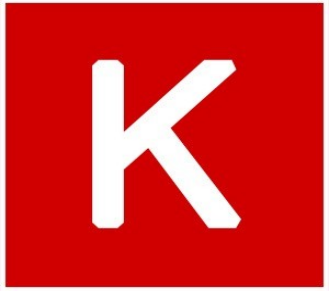
Standard RNN

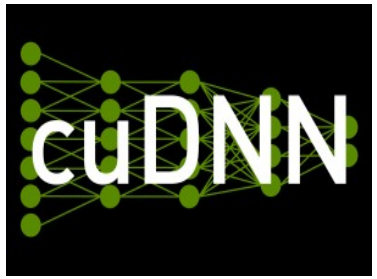


LSTM



Deep Learning Package Zoo

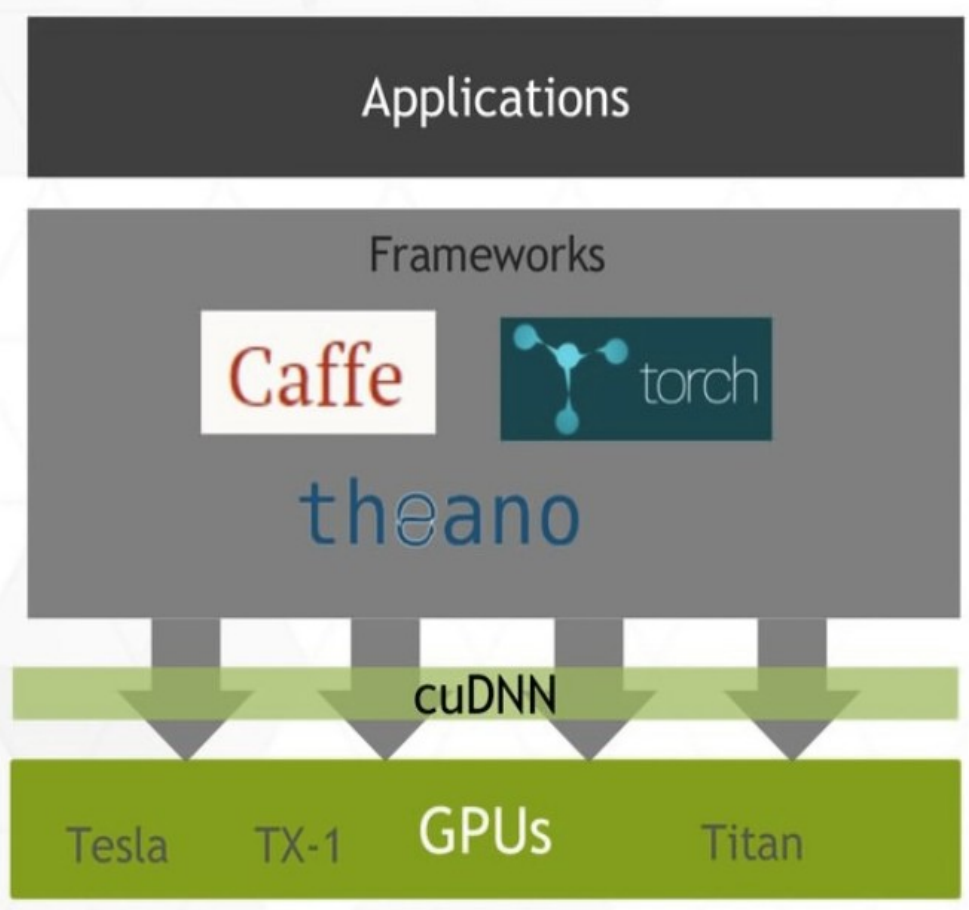




What they all have in common?

- Tensor
- Symbolic differentiation
- Open source
- GPU support (through backend like cuDNN)

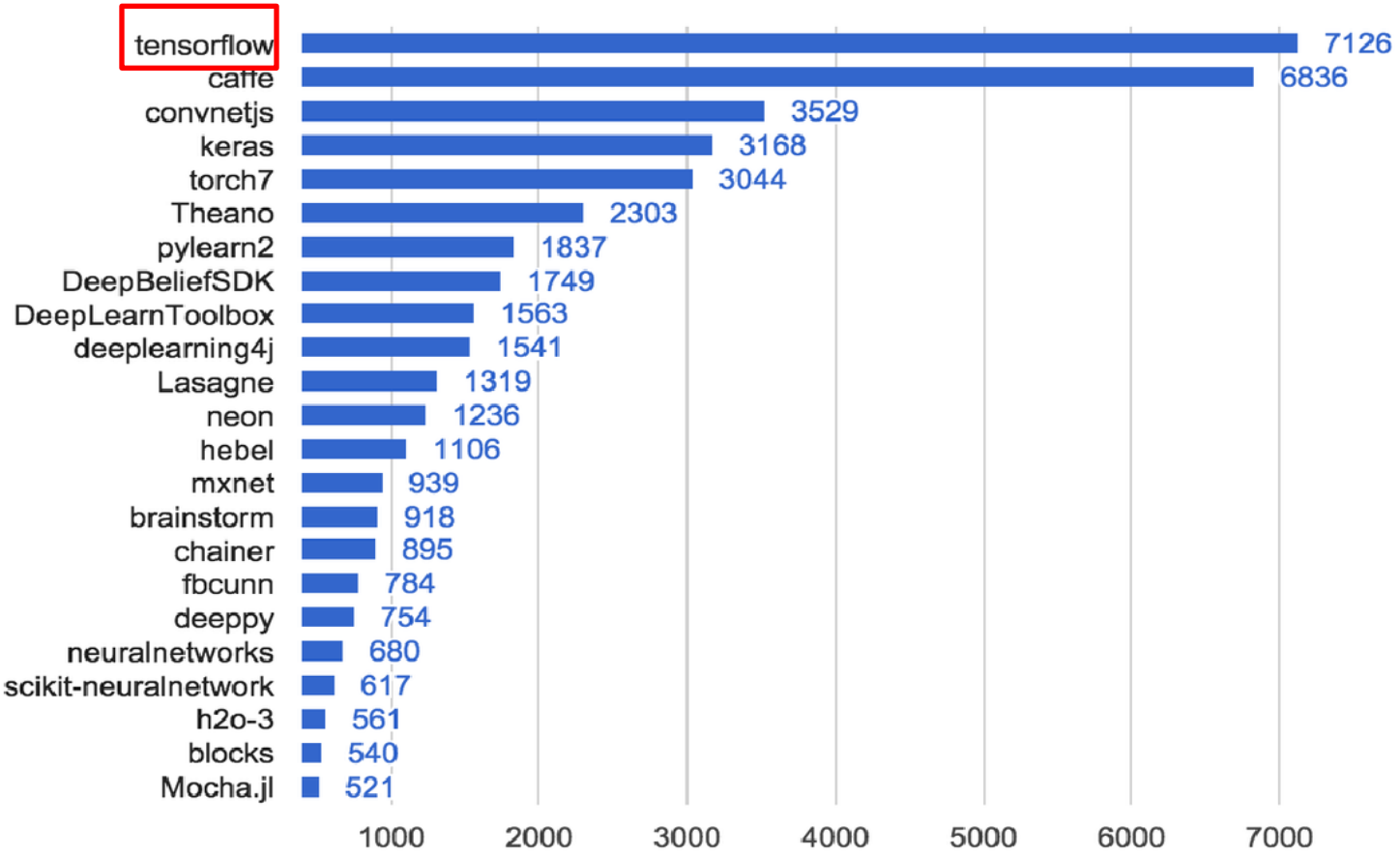
CuDNN is behind every package



CuDNN

- GPU-accelerated Deep Learning subroutines
- High performance neural network training
- Accelerates major deep learning frameworks: Caffe, Theano, Torch, TensorFlow

Deep Learning Tools



Outline

- Deep Learning: Hype or Reality?
- Deep Learning: Why is it happening now?
- Deep Learning: How to define it?
- A quick start with TensorFlow
- Why to use TensorFlow-Slim?
- Training Demonstration using TensorFlow and TF-Slim

TensorFlow

- A machine learning library created by Google
- Started as a research project
- Generates a computational graph like Theano
- Everything about TensorFlow is here:

  https://www.tensorflow.org

TensorFlow vs. Theano

- Both are deep learning libraries with Python wrapper
- Theano came out first (was inspiration for TensorFlow)
- TensorFlow has better support for distributed systems
- TensorFlow has development funded by Google, while Theano is an academic project

TensorFlow Recipe

- Recipe for a TensorFlow application:
  - Define a series of expressions
  - Initialize variables
  - Start a session (launch a graph)
  - Run the graph, feed some data, fetch some values

TensorFlow Essentials

- Four types of objects make TensorFlow unique from other frameworks
  - Session
  - Computational graph
  - Variables
  - Placeholder

TensorFlow Session

- "A Session object encapsulates the environment in which Tensor objects are evaluated." — TensorFlow Docs

```
In [4]: a = tf.constant(5.0)
        b = tf.constant(6.0)
        c = a*b

In [5]: print c

        Tensor("mul_2:0", shape=TensorShape([]), dtype=float32)

In [6]: with tf.Session() as sess:
            print (sess.run(c))

        30.0
```

TensorFlow Computational 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." — [TensorFlow Docs]

```
In [4]: a = tf.constant(5.0)
        b = tf.constant(6.0)
        c = a*b

In [5]: print c

        Tensor("mul_2:0", shape=TensorShape([]), dtype=float32)

In [6]: with tf.Session() as sess:
            print (sess.run(c))

        30.0
```

TensorFlow Variables

- TensorFlow Variables: hold and update **parameters**

```
In [9]: W1 = tf.ones((2,2))
        W2 = tf.Variable(tf.zeros((2,2)))

In [10]: init = tf.initialize_all_variables()

In [11]: with tf.Session() as sess:
             print sess.run(W1)
             sess.run(init)
             print sess.run(W2)

[[ 1.  1.]
 [ 1.  1.]]
[[ 0.  0.]
 [ 0.  0.]]
```

TensorFlow Placeholders

- TensorFlow placeholders: dummy nodes that provide entry points for data to computational graph

```
In [15]: input1 = tf.placeholder(tf.float32)
         input2 = tf.placeholder(tf.float32)

In [16]: output = tf.mul(input1, input2)

In [20]: with tf.Session()as sess:
             print(sess.run(output, feed_dict={input1:7., input2:2.}))

14.0
```

Why TensorFlow?

- Python + Numpy
- Graph based, easy to model
- Faster compile times than Theano
- Tensorboard for Visualization
- Open Sourced
- Data and Model Paralllelism
- Distributed supported

Large scale Deep Learning using Distributed TensorFlow

- **Provides support for execution on multiple GPUs:**

  bazel-bin/inception/imagenet_train --train_dir=/raid/ImageNet/train6/ --data_dir=/raid/ImageNet/small_person_data/ --pretrained_model_checkpoint_path=/raid/ImageNet/pretrained_model/model.ckpt-236500 --fine_tune=False --initial_learning_rate=0.001 --input_queue_memory_factor=1 **--num_gpus=3** --batch_size=96

- **Can be run on a cluster of CPUs:**

  bazel-bin/inception/imagenet_distributed_train --batch_size=96 --data_dir=/home/data/ImageNet/cluster_data **--job_name='ps'** **--task_id=0** **--ps_hosts='192.168.1.30:2225'** **--worker_hosts='192.168.1.31:2225,192.168.1.32:2225,192.168.1.33:2225'** --train_dir=$HOME/tmp/ImageNet/ps0

TensorFlow in Google Products

- **RankBrain** (Search)
- **Speech Recognition** (Google now)
- **Spam Filter** (Gmail)
- **Photo Search** (Google Photos)
- **Visual Translation** (Google Translate)
- **Video Thumbnails** (YouTube)
- **Voicemail transcriptions** (Google Voice)

Meta-frameworks for TensorFlow

- Keras
- SkFlow
- PrettyTensor
- **TF-Slim**

Outline

- Deep Learning: Hype or Reality?
- Deep Learning: Why is it happening now?
- Deep Learning: How to define it?
- A quick start with TensorFlow
- Why to use TensorFlow-Slim?
- Training Demonstration using TensorFlow and TF-Slim

TensorFlow-Slim

- A light-weight library for defining, training and evaluating complex models
- Can be mixed with native TensorFlow
- Simple to use:

```
import tensorflow.contrib.slim as slim
```

Why TensorFlow-Slim?

- Simple and compact way to define model architectures
- Provides commonly used regularizers
- Contains model architectures for models such as VGG, AlexNet and Inception
- Easy to extend complex models
- Supported by Google

Defining a model using TF-Slim

- **Layers**
- **Tensorflow:**

```
with tf.name_scope('conv1_1') as scope:
    kernel = tf.Variable(tf.truncated_normal([3, 3, 64, 128], dtype=tf.float32, stddev=1e-1), name='weights')
    conv = tf.nn.conv2d(input, kernel, [1, 1, 1, 1], padding='SAME')
    biases = tf.Variable(tf.constant(0.0, shape=[128], dtype=tf.float32), trainable=True, name='biases')
    bias = tf.nn.bias_add(conv, biases)
    conv1 = tf.nn.relu(bias, name=scope)
```

- **TF-Slim**:

```
net = slim.conv2d(input, 128, [3, 3], scope='conv1_1')
```

Defining a model using TF-Slim

- **Layers**
  - **Normal way:**

```
net = slim.conv2d(net, 256, [3, 3], scope='conv3_1')
net = slim.conv2d(net, 256, [3, 3], scope='conv3_2')
net = slim.conv2d(net, 256, [3, 3], scope='conv3_3')
net = slim.max_pool2d(net, [2, 2], scope='pool3')
```

- **Using Repeat:**

```
net = slim.repeat(net, 3, slim.conv2d, 256, [3, 3], scope='conv3')
net = slim.max_pool(net, [2, 2], scope='pool2')
```

Defining a model using TF-Slim

- **Normal way:**

```
net = slim.conv2d(inputs, 64, [11, 11], 4, padding='SAME',
weights_initializer=tf.truncated_normal_initializer(stddev=0.01),
weights_regularizer=slim.l2_regularizer(0.0005), scope='conv1')

net = slim.conv2d(net, 128, [11, 11], padding='VALID',
weights_initializer=tf.truncated_normal_initializer(stddev=0.01),
weights_regularizer=slim.l2_regularizer(0.0005), scope='conv2')

net = slim.conv2d(net, 256, [11, 11], padding='SAME',
weights_initializer=tf.truncated_normal_initializer(stddev=0.01),
weights_regularizer=slim.l2_regularizer(0.0005), scope='conv3')
```

Defining a model using TF-Slim

- **Using Scope:**

```
with slim.arg_scope([slim.conv2d, slim.fully_connected],
   activation_fn=tf.nn.relu,
   weights_initializer=tf.truncated_normal_initializer(stddev=0.01),
   weights_regularizer=slim.l2_regularizer(0.0005)):

with arg_scope([slim.conv2d], stride=1, padding='SAME'):

    net = slim.conv2d(inputs, 64, [11, 11], 4, padding='VALID', scope='conv1')
    net = slim.conv2d(net, 256, [5, 5],
    weights_initializer=tf.truncated_normal_initializer(stddev=0.03),
    scope='conv2')
    net = slim.fully_connected(net, 1000, activation_fn=None, scope='fc')
```

Components of TensorFlow-Slim

- arg_scope
- data
- evaluation
- layers
- learning
- losses
- metrics
- nets
- preprocess
- regularizers
- queues
- variables

Outline

- Deep Learning: Hype or Reality?
- Deep Learning: Why is it happening now?
- Deep Learning: How to define it?
- A quick start with TensorFlow
- Why to use TensorFlow-Slim?
- Training Demonstration using TensorFlow and TF-Slim

Training Demonstration

- MNIST Image Classification using LeNet Architecture with TensorFlow and TF-Slim

https://github.com/dipendra009/MNIST_TF-Slim

Conclusion

- Deep learning is no magic!
- Learnt about how CNNs work.
- Use of TensorFlow and TF-Slim
- Training Demonstration using TF and TF-Slim

Thank You

Queries???