Week 1: Getting Started with Deep Learning

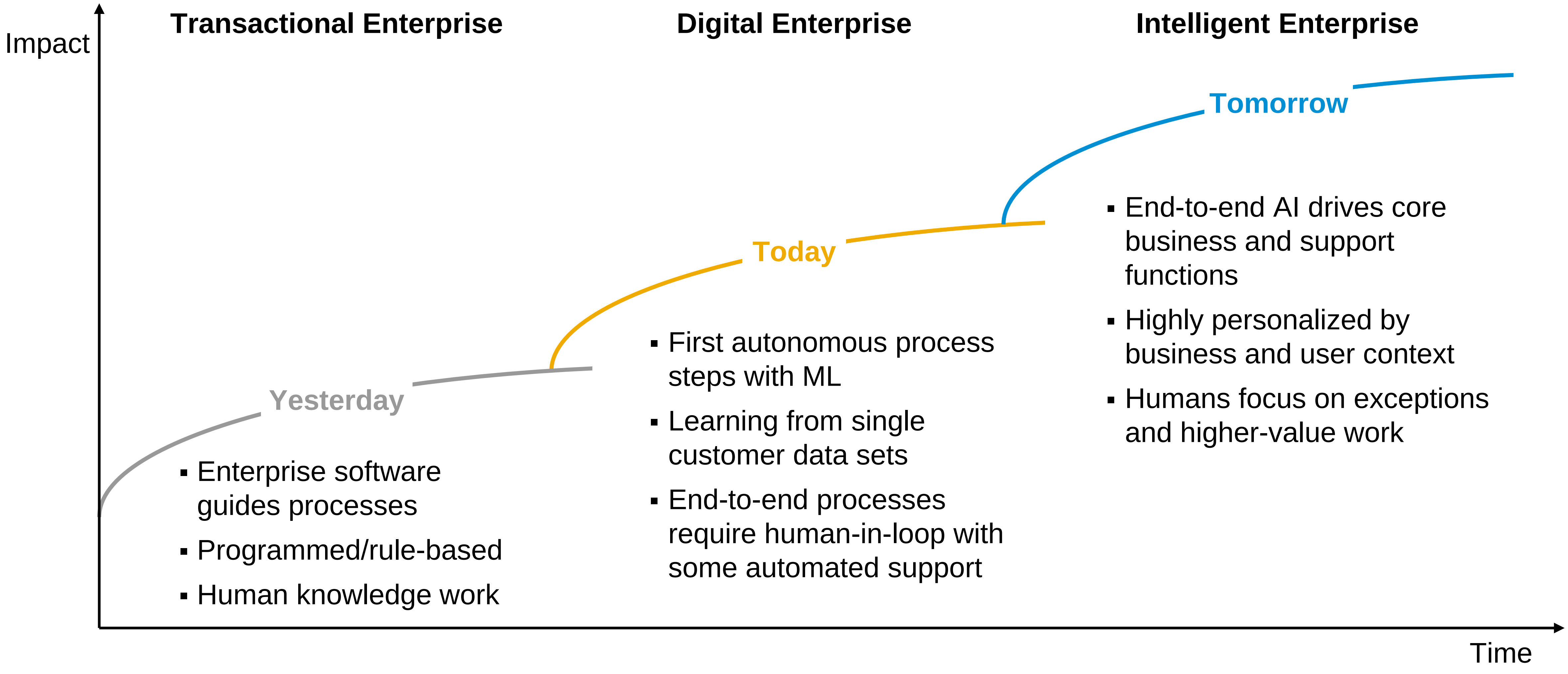
Unit 1: Deep Learning for the Intelligent Enterprise

Run Simple

openSAP
open.sap.com

Deep Learning for the Intelligent Enterprise

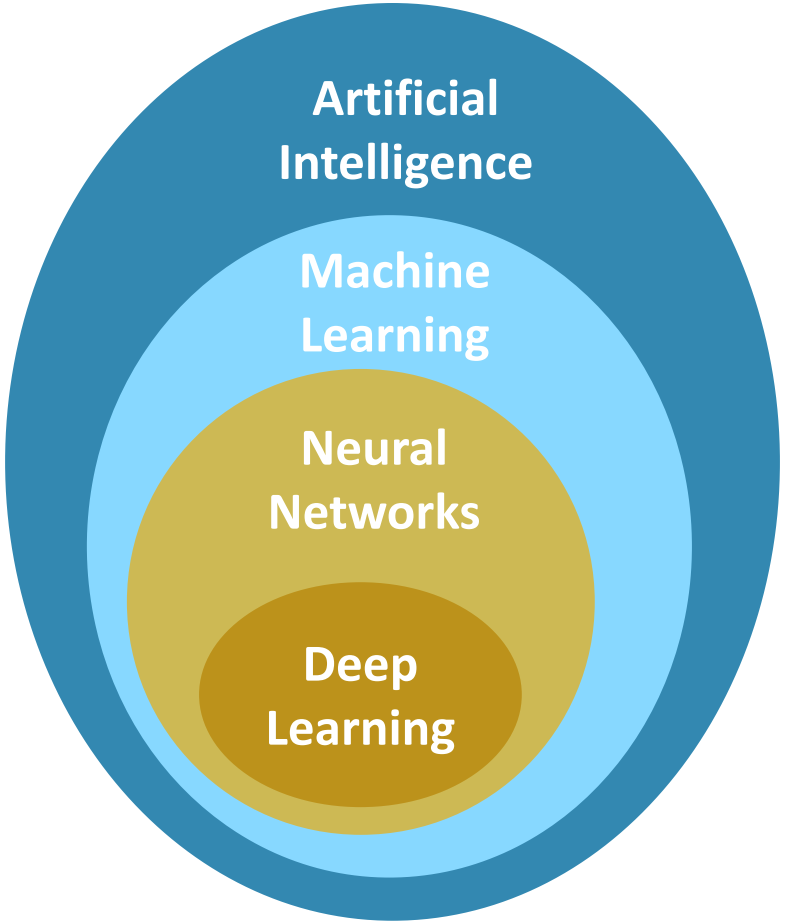
Intelligent Enterprise

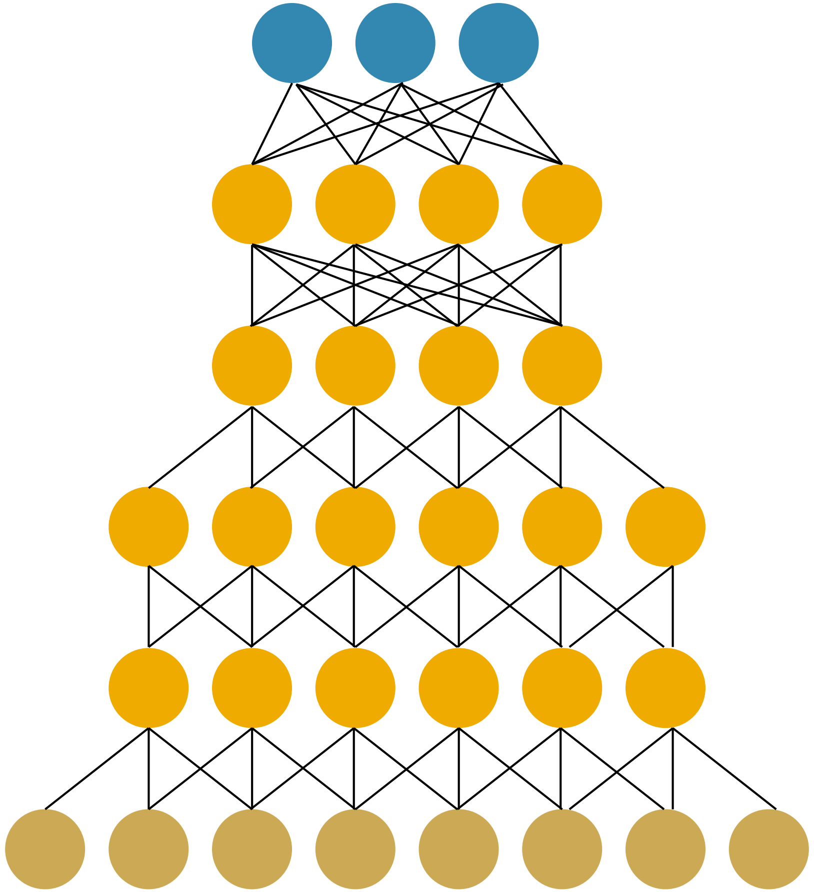


Deep Learning for the Intelligent Enterprise

Deep learning

- Sub-field of neural networks, machine learning, and artificial intelligence
- Deep learning is neural networks with many layers
- Inspired by, but not limited to, the architecture of the human brain
- Deep learning is the reality behind artificial intelligence





Deep Learning for the Intelligent Enterprise

TensorFlow

- Open-source library for deep learning
- Define model structures, library takes care of efficient execution
- Define once, run anywhere: can run on on CPUs and GPUs, many devices
- Can be used in Python and many other languages
- Built for large-scale machine learning development and operations
- Development led by Google

Deep Learning for the Intelligent Enterprise

SAP Leonardo makes machine learning incredibly easy

76% of the world's transaction revenue

25 industries

12 lines of business

The world's largest business-to-business network

Source: SAP Fast Facts

+

SAP Leonardo

Machine Learning

- Applications
- Business services
- Functional services
- Training & Lifecycle management
- SAP Cloud Platform and SAP HANA

=

Business Outcomes

Increased revenue with superior sales targeting and execution

Reimagined business **processes** with digital intelligence

Improved **quality time at work** for employees

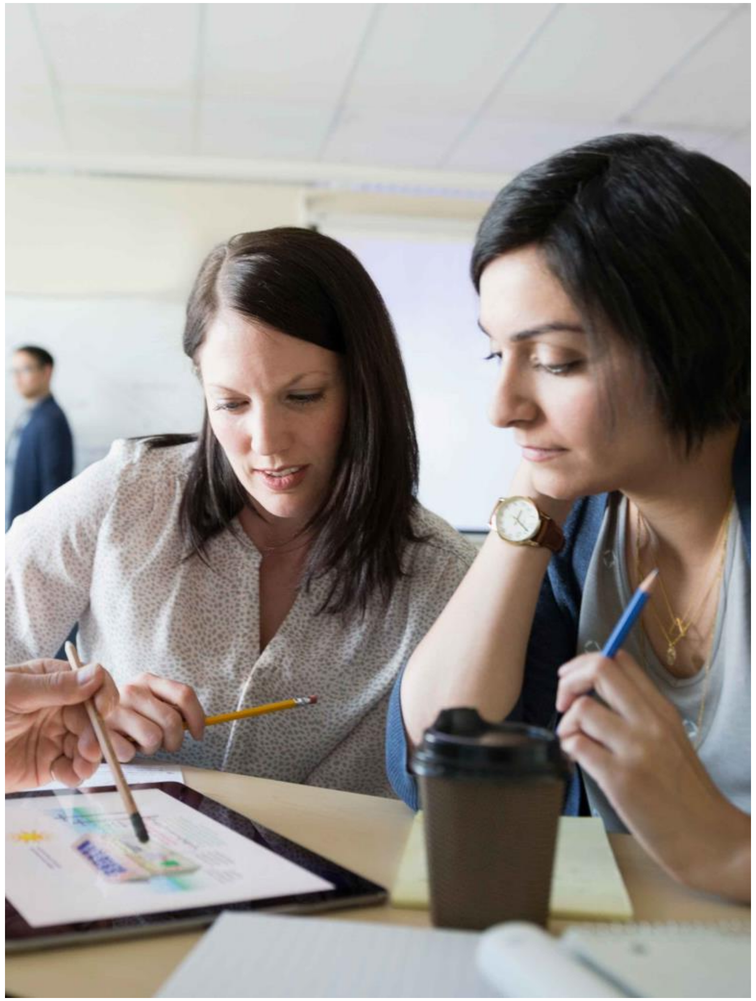
Increased **customer satisfaction** with superior service

Enablement of product, process, and business model **innovations**

Deep Learning for the Intelligent Enterprise

Course objectives

- Know how to identify deep-learning use cases for the Enterprise
- Learn how to build, train, test and deploy deep learning models
- Gain hands-on knowledge of developing deep learning models using TensorFlow
- Get to know best practices from deep learning experts
- Hear about industry applications of deep learning



Deep Learning for the Intelligent Enterprise

Outline of this course

Week 1: Getting started with deep learning

Week 2: Building TensorFlow applications

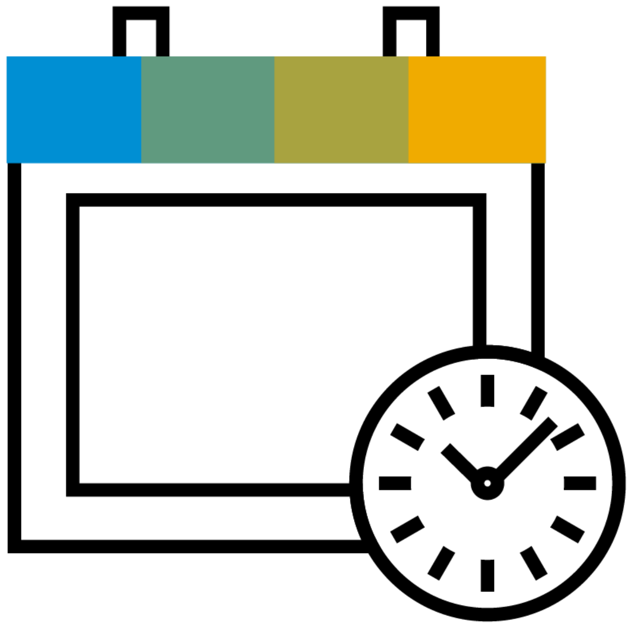
Week 3: Deep networks and sequence models

Week 4: Convolutional networks

Week 5: Industry applications of deep learning

Week 6: Advanced deep learning topics

Week 7: Final exam



Deep Learning for the Intelligent Enterprise

Coming up next

1.1 Deep Learning for the Intelligent Enterprise

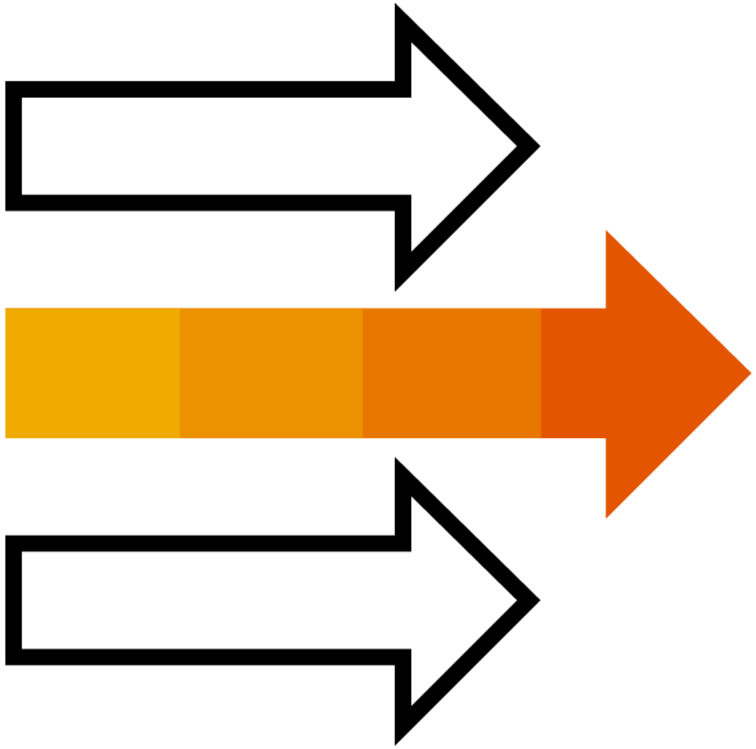
1.2 From Neural Networks to Deep Learning

1.3 Getting Started with Jupyter Notebooks

1.4 Building Your First Neural Network

1.5 Introduction to TensorFlow

1.6 When to Use Deep Learning



Thank you.

Contact information:

open@sap.com

Week 1: Getting Started with Deep Learning

Unit 2: From Neural Networks to Deep Learning

Dr. Damian Borth
German Research Center for Artificial Intelligence (DFKI)

Run Simple

Deutsches Forschungszentrum für Künstliche Intelligenz GmbH

openSAP
open.sap.com

From Neural Networks to Deep Learning

DFKI Deep Learning Competence Center

German Research Center for Artificial Intelligence (DFKI)

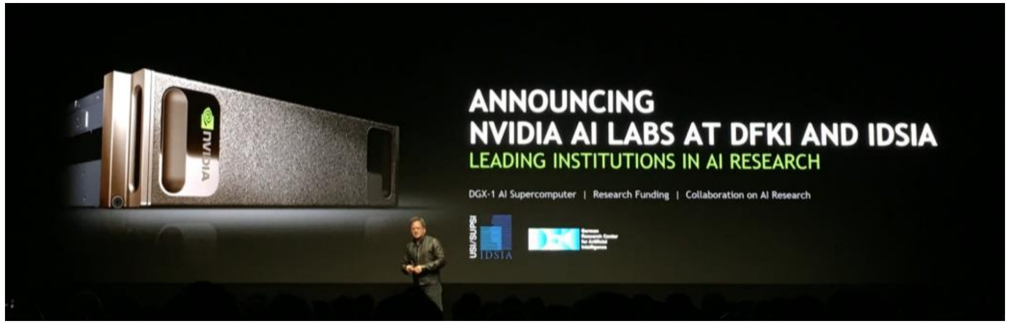
- Founded 1988
- Largest not-for-profit AI center in the world

Deep Learning Competence Center

- Founded Dec 2015 in Kaiserslautern
- Represents researchers in DL from different DFKI sites (Kaiserslautern, Saarbrücken, Bremen, Berlin)


Frankfurter Allgemeine
ZEITUNG FÜR DEUTSCHLAND

Er ist Deutschlands Mister Deep Learning



Collaborations:

Berkeley University of California

Columbia University in the City of New York

DVFA

International Computer Science Institute (ICSI)

Carnegie Mellon

NVIDIA

From Neural Networks to Deep Learning

Mapping between input and output

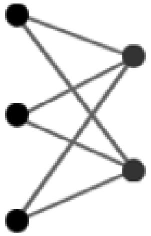
“flower”

Drive carefully due
to the traffic situation

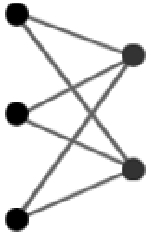
To win this match,
move this piece

From Neural Networks to Deep Learning

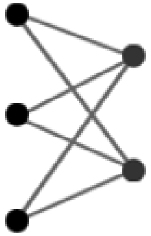
Mapping between input and output



“flower”



Drive carefully due
to the traffic situation



To win this match,
move this piece

From Neural Networks to Deep Learning

What is Deep Learning?

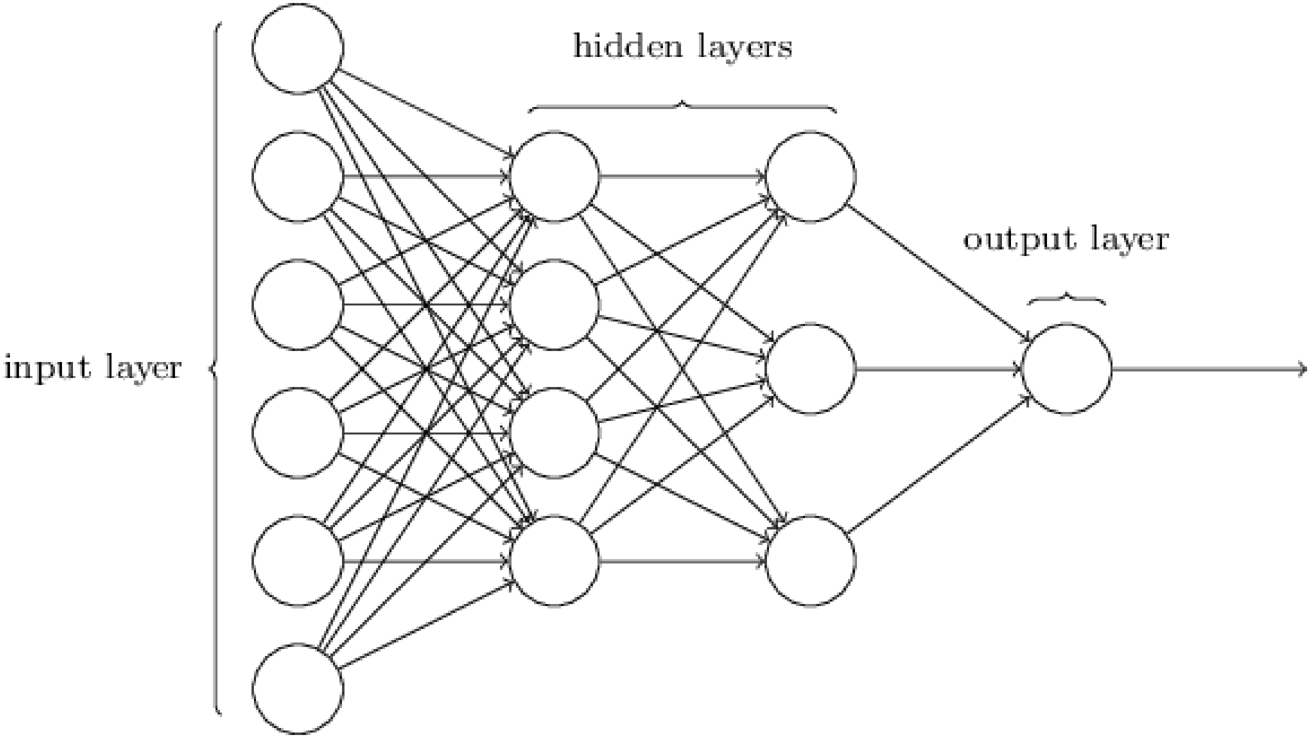
Deep Learning is an umbrella term for machine learning approaches based on (deep) neural networks.

Deep Learning is an “end-to-end” learning approach where features are learned together with the classification function.

Deep Learning defines a cascade of multiple layers of nonlinear processing units for feature extraction and transformation. Each successive layer uses the output from the previous layer as input.

From Neural Networks to Deep Learning

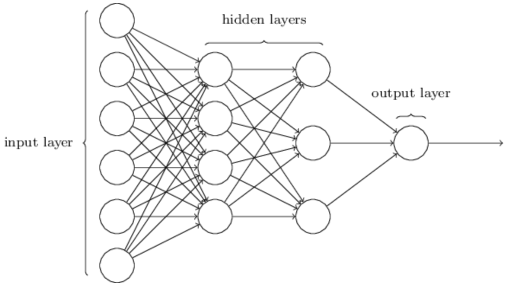
Neural Networks (NN)



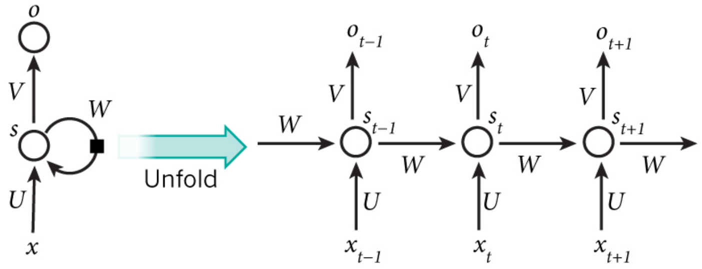
From Neural Networks to Deep Learning

Deep learning – Different types of architectures

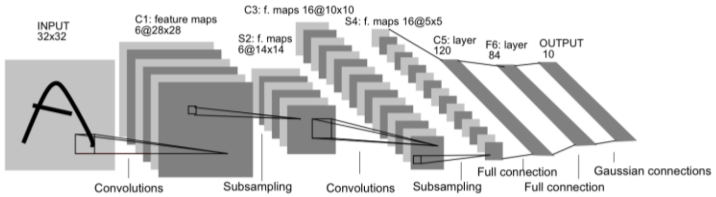
Neural Networks (NN)

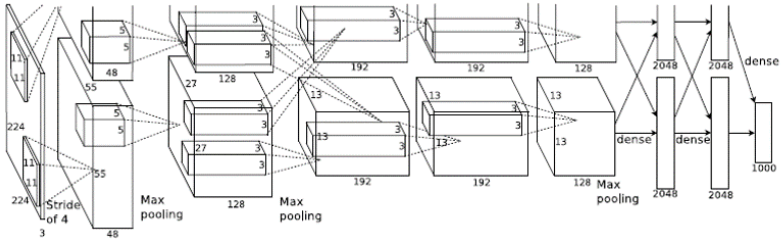


Recurrent Neural Networks (RNN) & Long-Short Term Memory (LSTM)

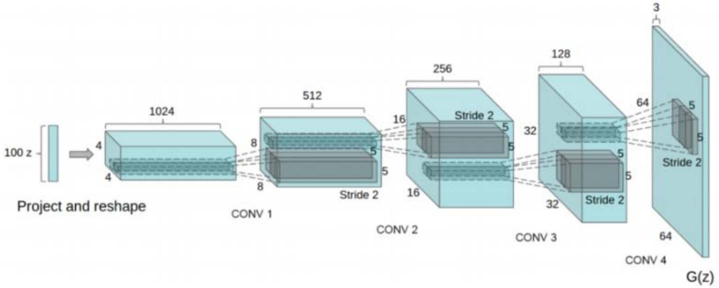


Convolutional Neural Networks (CNN)





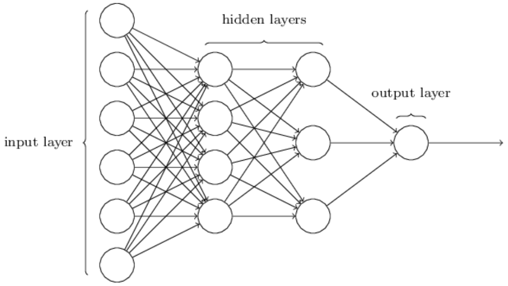
Generative Adversarial Networks (GAN)



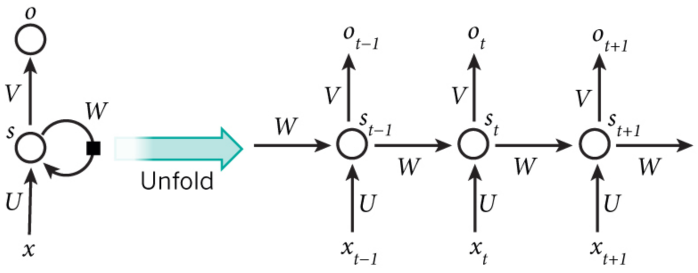
From Neural Networks to Deep Learning

Deep learning – Different types of architectures

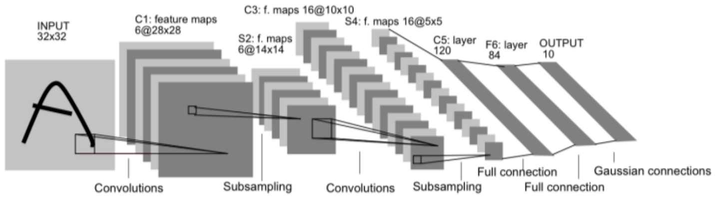
Neural Networks (NN)

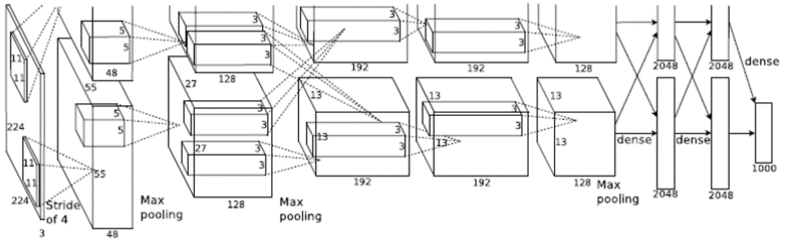


Recurrent Neural Networks (RNN) & Long-Short Term Memory (LSTM)

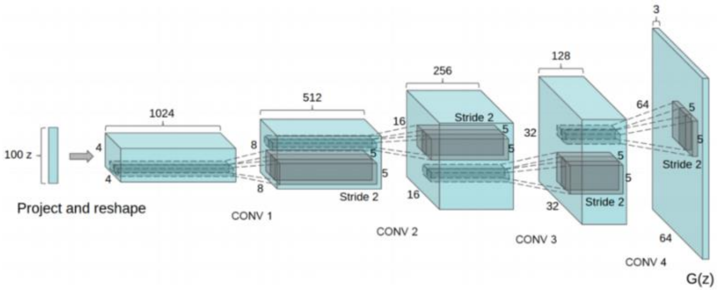


Convolutional Neural Networks (CNN)





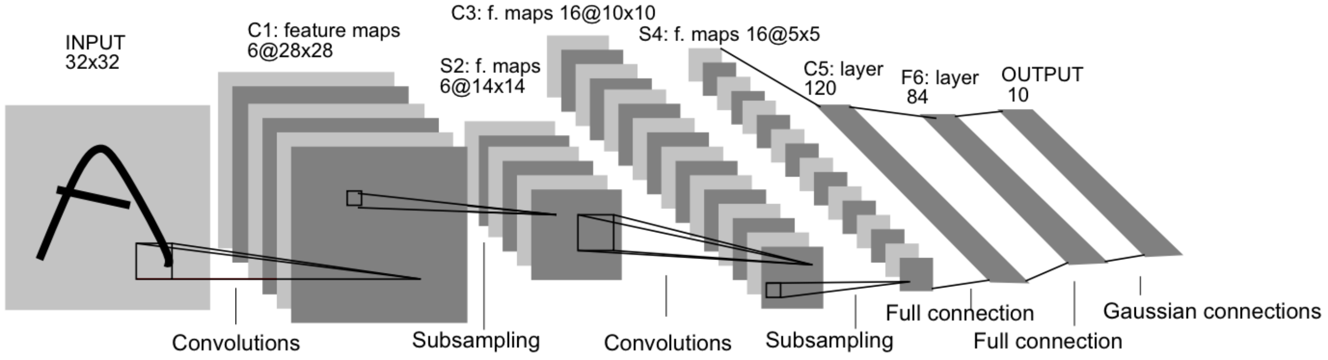
Generative Adversarial Networks (GAN)



From Neural Networks to Deep Learning

Deep learning – What made the difference in 2012

LeNet 1989: Handwriting Recognition



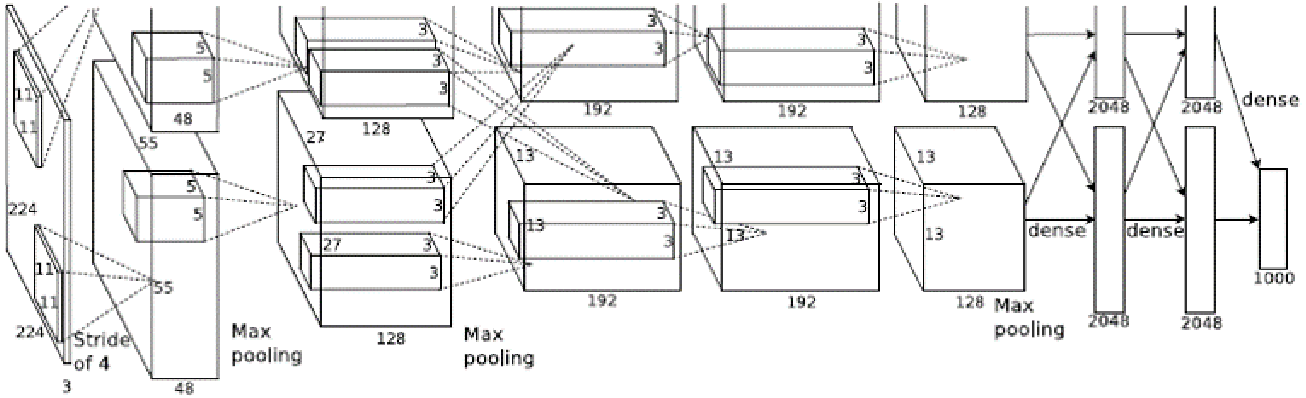
# of transistors

10^6

# of pixels used in training

10^7 NIST

AlexNet 2012: Image Understanding (**Data + GPU**)



# of transistors GPUs

10^9

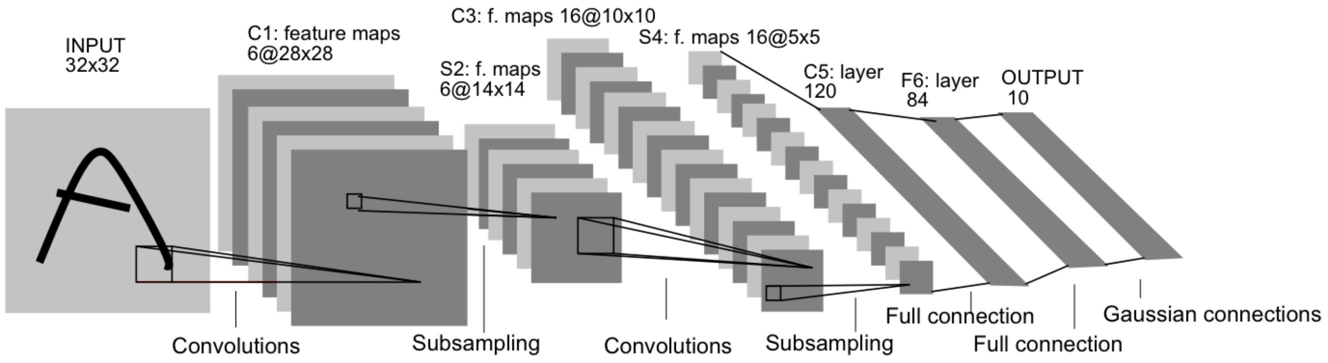
# of pixels used in training

10^{14} IMAGENET

From Neural Networks to Deep Learning

Deep learning – What made the difference in 2012

LeNet 1989: Handwriting Recognition



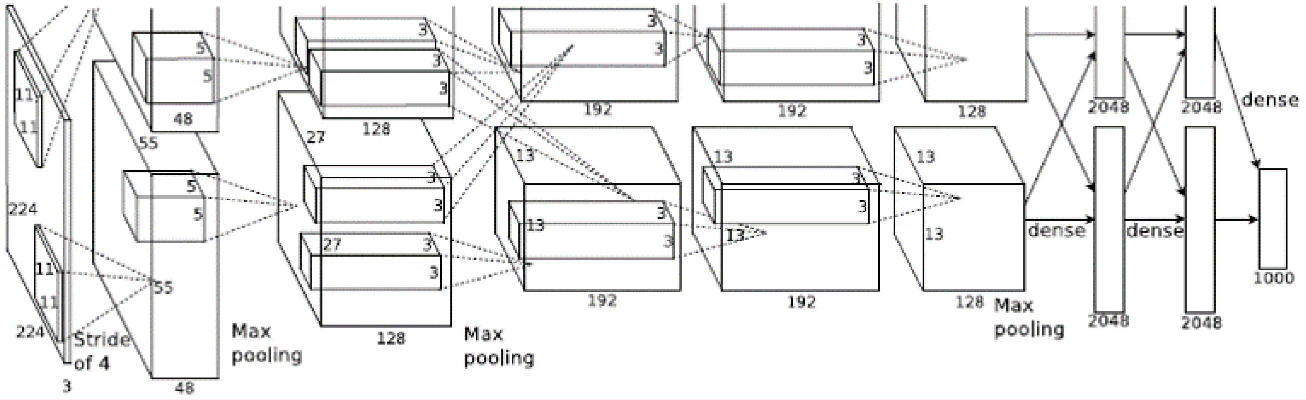
# of transistors

10^6

# of pixels used in training

10^7 NIST

AlexNet 2012: Image Understanding (**Data + GPU**)



# of transistors

10^9

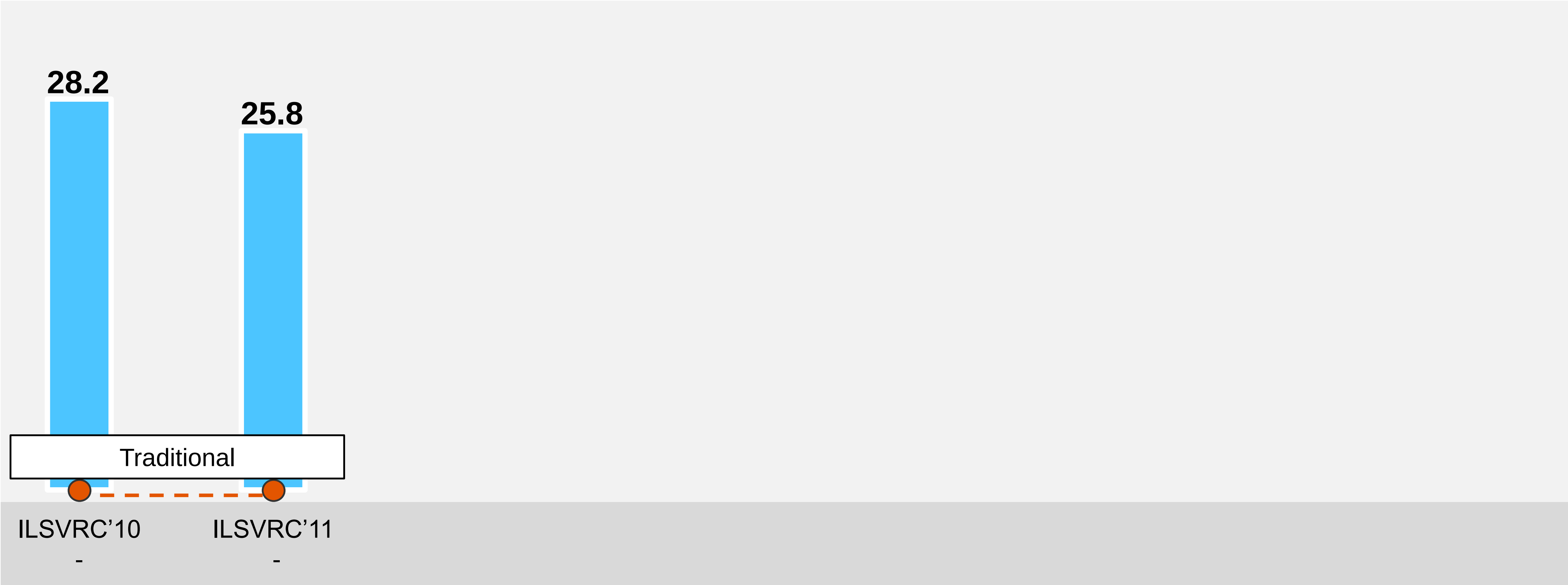
GPUs

# of pixels used in training

10^{14} IMAGENET

From Neural Networks to Deep Learning

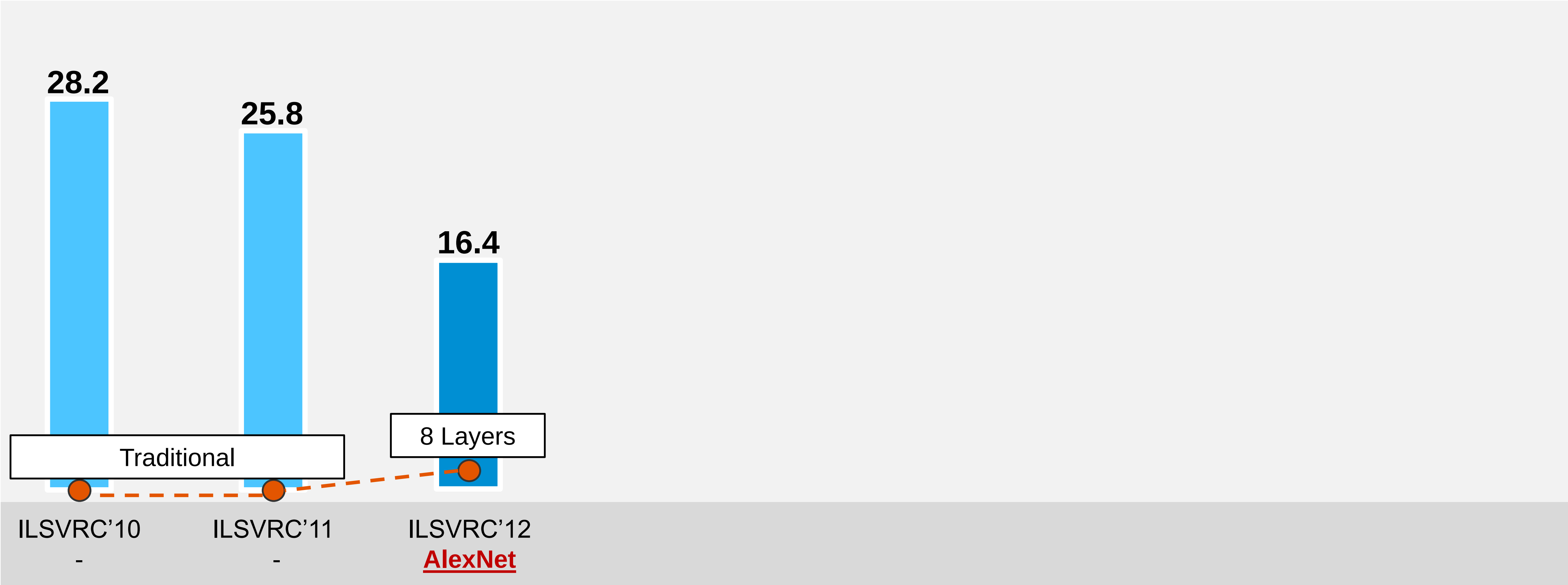
Revolution of depth



IMAGENET : Image Classification Top-5 Error(%)

From Neural Networks to Deep Learning

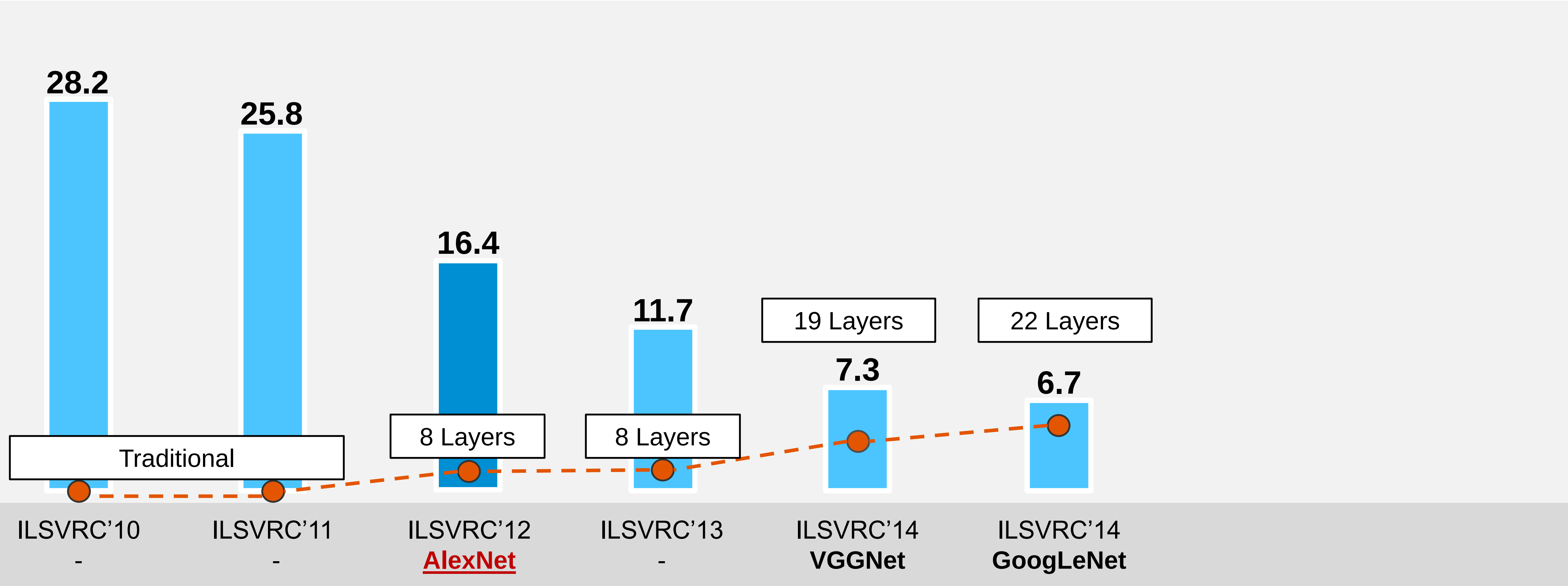
Revolution of depth



IMAGENET : Image Classification Top-5 Error(%)

From Neural Networks to Deep Learning

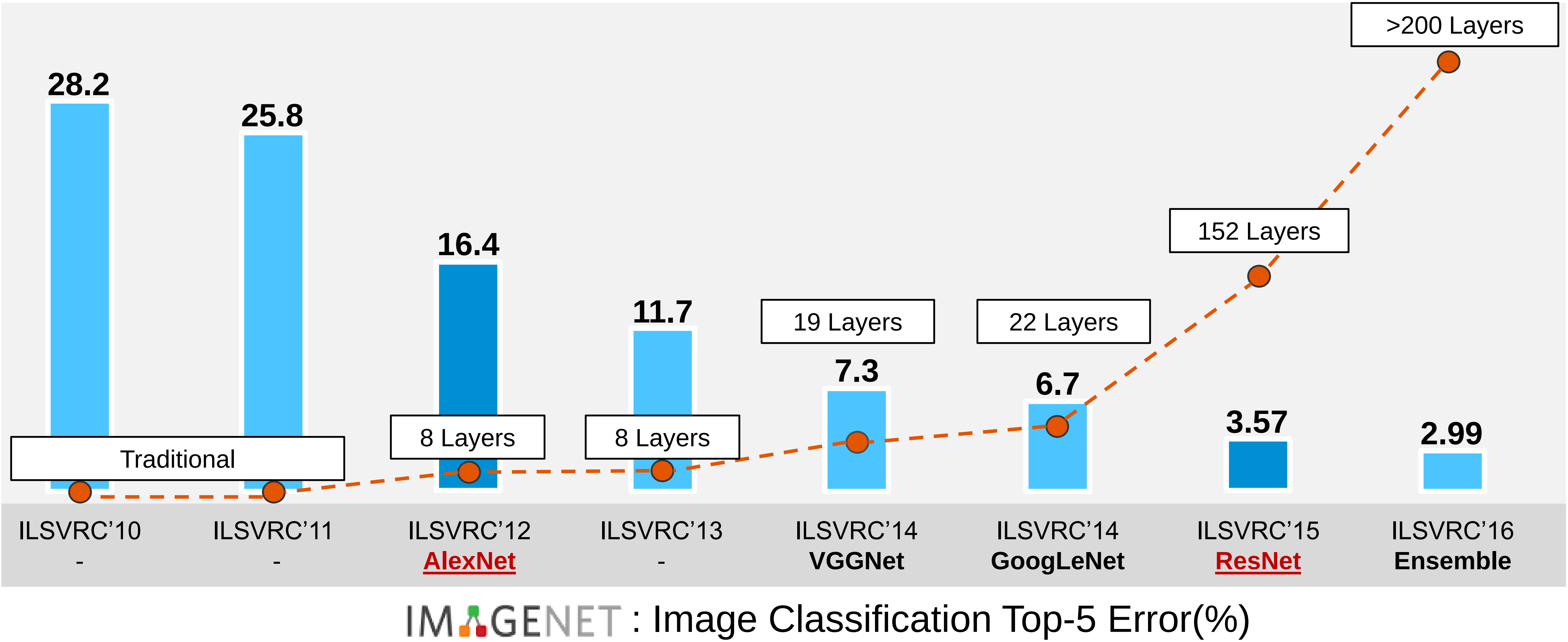
Revolution of depth



IMAGENET : Image Classification Top-5 Error(%)

From Neural Networks to Deep Learning

Revolution of depth



From Neural Networks to Deep Learning

Traditional approaches vs. “End-to-End” approaches

Image Understanding

→ **SIFT / HOG** → ***k*-means / pooling** → **Classification** → “building”

fixed | unsupervised | supervised

“**Deep Learning Tutorial**”,
Y. LeCun, M. Ranzato, ICML, 2013

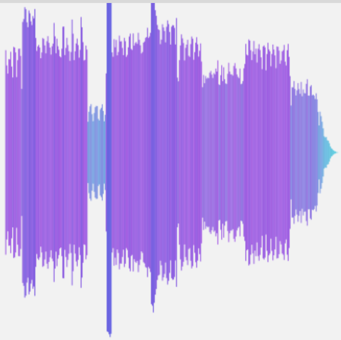
From Neural Networks to Deep Learning

Traditional approaches vs. “End-to-End” approaches

Image Understanding

→	**SIFT / HOG**	→ **_k_-means / pooling**	→ **Classification**	→ “building”
	fixed	unsupervised	supervised	

Speech Recognition



→	**MFCC**	→ **Mixture of Gaussians**	→ **Classification**	→ \teach\
	fixed	unsupervised	supervised	

“**Deep Learning Tutorial**”,
Y. LeCun, M. Ranzato, ICML, 2013

From Neural Networks to Deep Learning

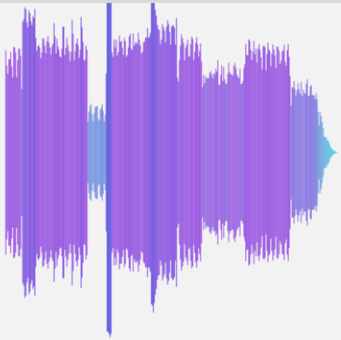
Traditional approaches vs. “End-to-End” approaches

Image Understanding

→ **SIFT / HOG** (fixed) → ***k*-means / pooling** (unsupervised) → **Classification** (supervised) → “building”

Feature Representation

Speech Recognition



→ **MFCC** (fixed) → **Mixture of Gaussians** (unsupervised) → **Classification** (supervised) → \teach\

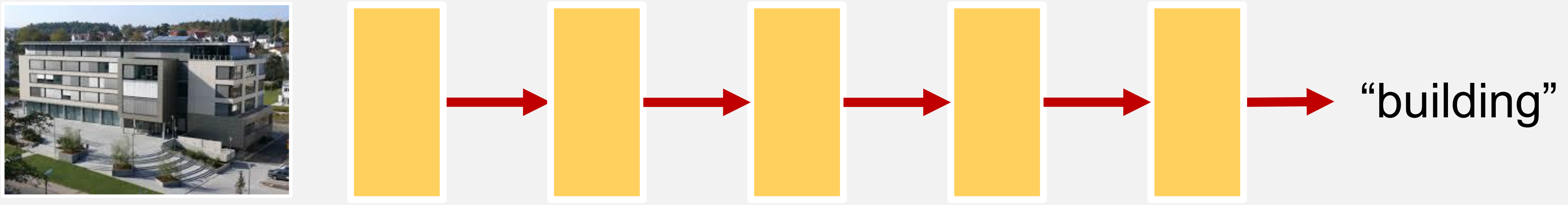
Feature Representation

“**Deep Learning Tutorial**”,
Y. LeCun, M. Ranzato, ICML, 2013

From Neural Networks to Deep Learning

Traditional approaches vs. “End-to-End” approaches

Image Understanding



End-to-End Learning

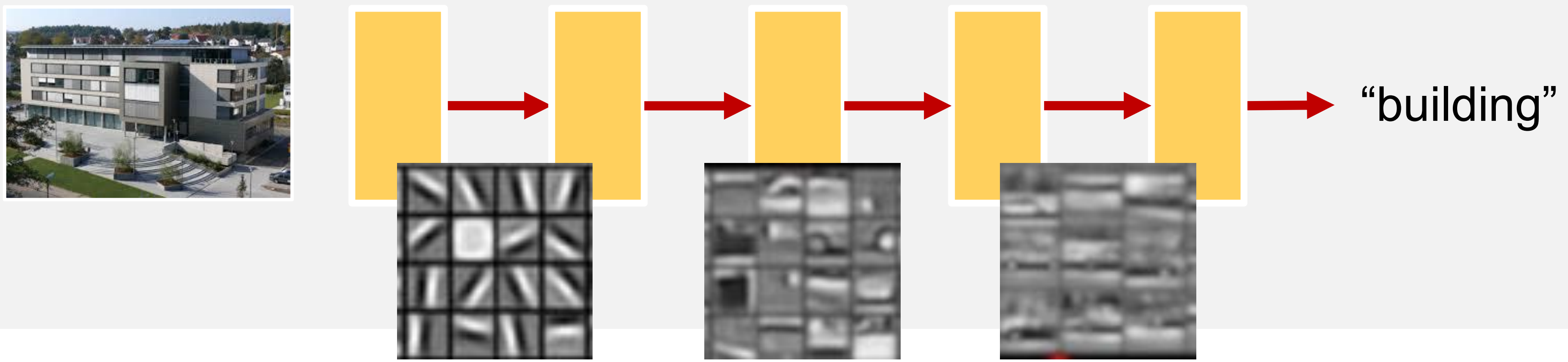
- Cascade of non-linear transformations

“**Deep Learning Tutorial**”,
Y. LeCun, M. Ranzato, ICML, 2013

From Neural Networks to Deep Learning

Traditional approaches vs. “End-to-End” approaches

Image Understanding



End-to-End Learning

- Cascade of non-linear transformations
- Collective training of features and classification creates a shared representation of information
- This representation is internal
- Layers of representations are adaptive

“**Deep Learning Tutorial**”,
Y. LeCun, M. Ranzato, ICML, 2013

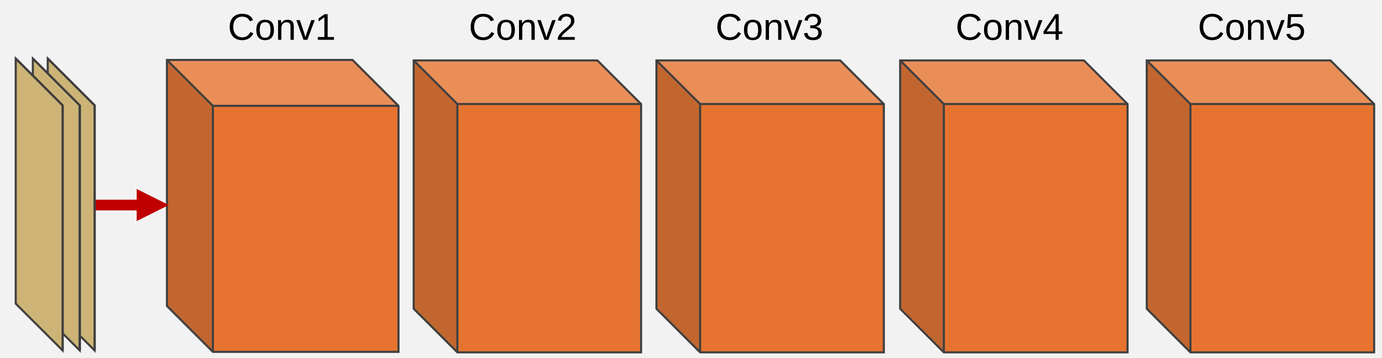
From Neural Networks to Deep Learning
Reference architecture: AlexNet

[**“ImageNet Classification with Deep Convolutional Neural Networks”**, A Krizhevsky, I Sutskever, GE Hinton, NIPS, 2012]
[“**A Taxonomy of Deep Convolutional Neural Nets for Computer Vision**”, S. Srinivas, et al, 2016]

From Neural Networks to Deep Learning

Reference architecture: AlexNet

[**"ImageNet Classification with Deep Convolutional Neural Networks"**, A Krizhevsky, I Sutskever, GE Hinton, NIPS, 2012]
["**A Taxonomy of Deep Convolutional Neural Nets for Computer Vision**", S. Srinivas, et al, 2016]



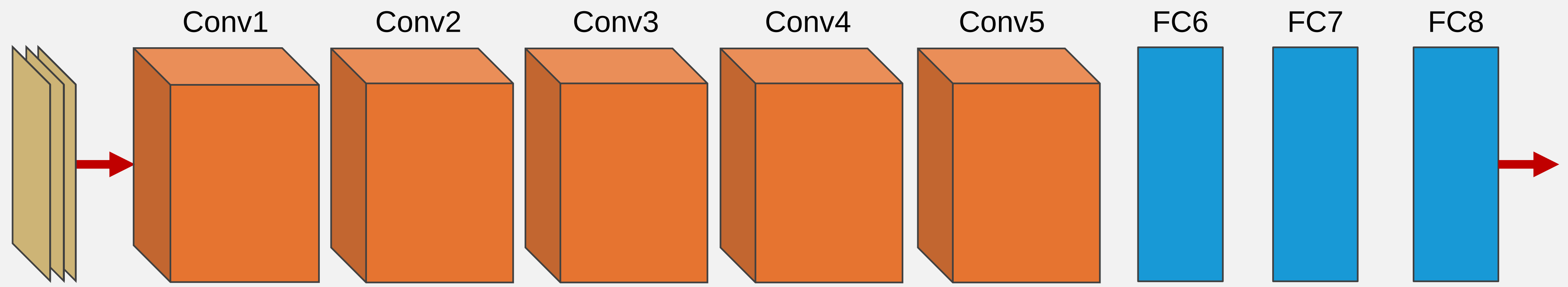
General architecture: 8 layers

- 5 layers to learn the feature representation (Conv)

From Neural Networks to Deep Learning

Reference architecture: AlexNet

[**"ImageNet Classification with Deep Convolutional Neural Networks"**, A Krizhevsky, I Sutskever, GE Hinton, NIPS, 2012]
["**A Taxonomy of Deep Convolutional Neural Nets for Computer Vision**", S. Srinivas, et al, 2016]



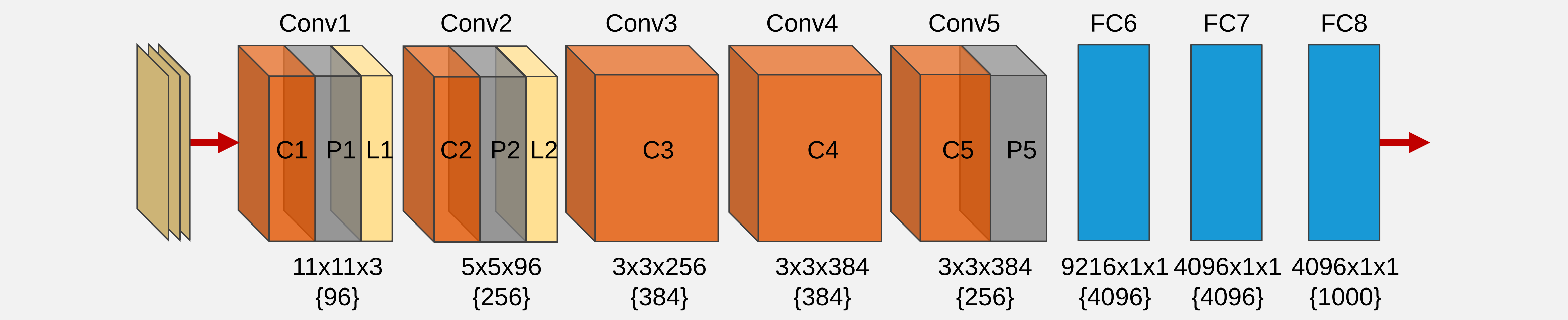
General architecture: 8 layers

- 5 layers to learn the feature representation (Conv)
- 3 layers for classification (FC)

From Neural Networks to Deep Learning

Reference architecture: AlexNet

[**"ImageNet Classification with Deep Convolutional Neural Networks"**, A Krizhevsky, I Sutskever, GE Hinton, NIPS, 2012]
["**A Taxonomy of Deep Convolutional Neural Nets for Computer Vision**", S. Srinivas, et al, 2016]



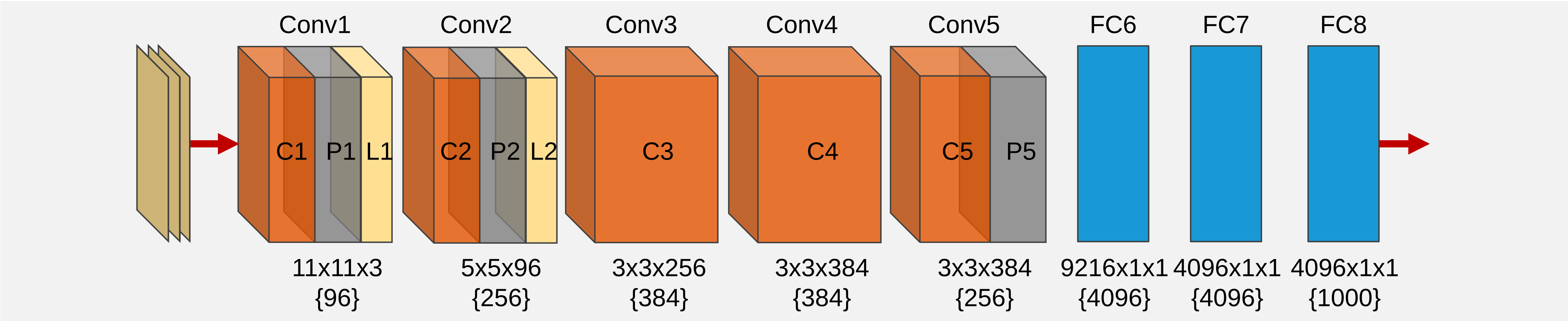
General architecture: 8 layers

- 5 layers to learn the feature representation (Conv)
  - "Convolution" (C)
  - "Max. Pooling" (P)
  - "Local Response Normalization" (L)
- 3 layers for classification (FC)

From Neural Networks to Deep Learning

Reference architecture: AlexNet

[**“ImageNet Classification with Deep Convolutional Neural Networks”**, A Krizhevsky, I Sutskever, GE Hinton, NIPS, 2012]
[“**A Taxonomy of Deep Convolutional Neural Nets for Computer Vision**”, S. Srinivas, et al, 2016]



Convolution (C1, …, C5)

- These operations calculate convolutions over a local image region and the learned feature map (weights)
- Parameter Constraints
  – Local Connectivity / Receptive Field / Shared Weights

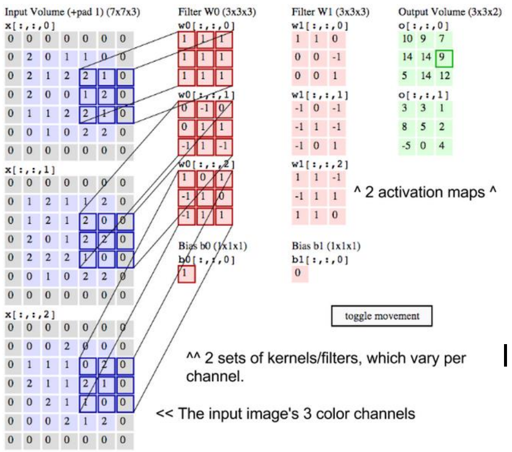
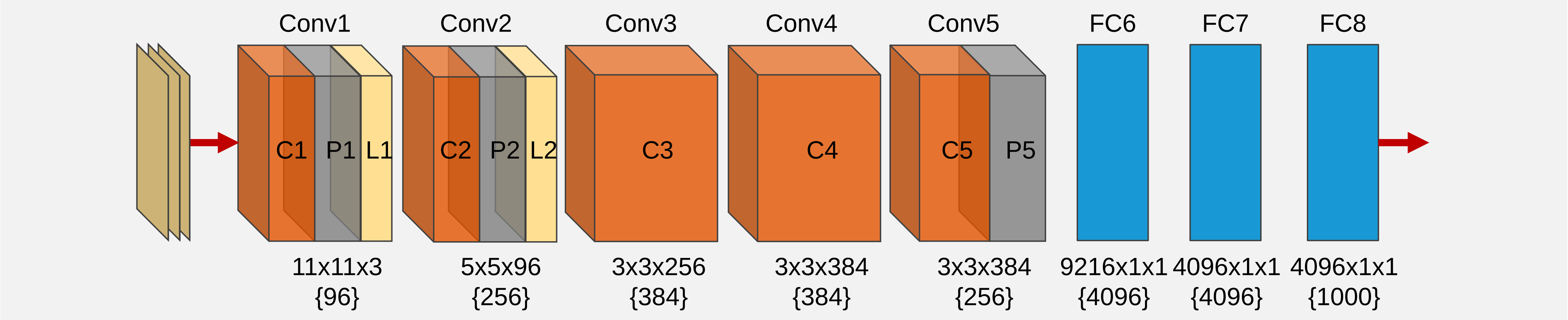


Image from Andrej Karpathy

From Neural Networks to Deep Learning

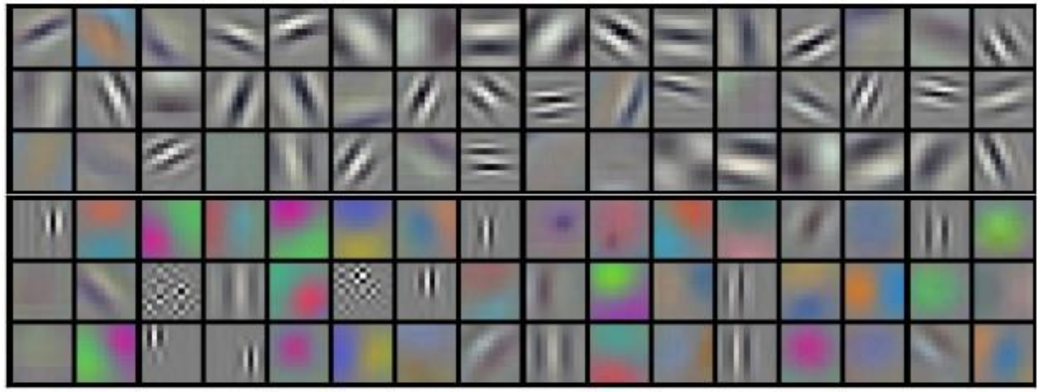
Reference architecture: AlexNet

[**"ImageNet Classification with Deep Convolutional Neural Networks"**, A Krizhevsky, I Sutskever, GE Hinton, NIPS, 2012]
["**A Taxonomy of Deep Convolutional Neural Nets for Computer Vision**", S. Srinivas, et al, 2016]



Convolution

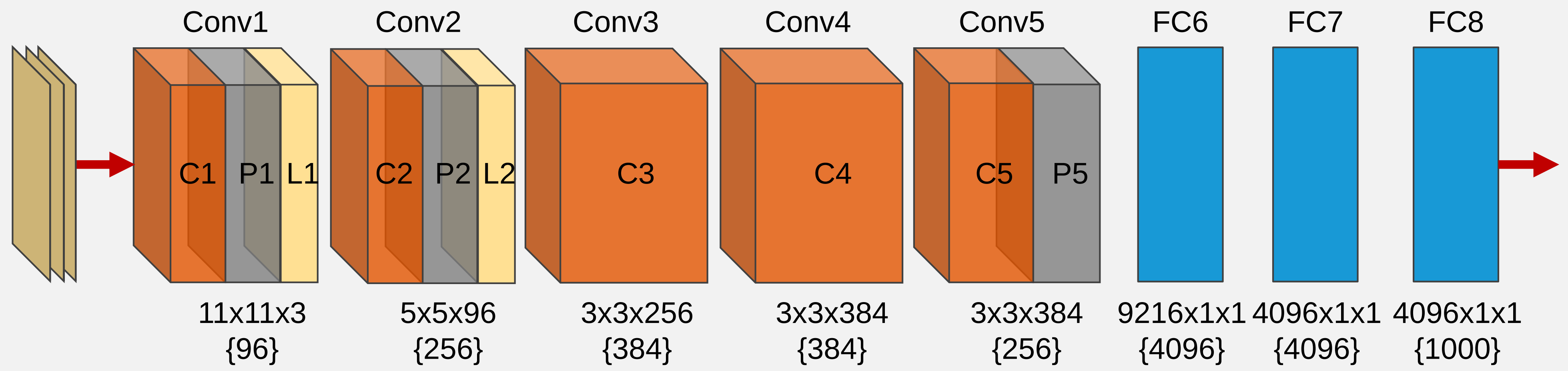
- 96 learned feature maps of the first layer of AlexNet
- Size: 11x11x3



From Neural Networks to Deep Learning

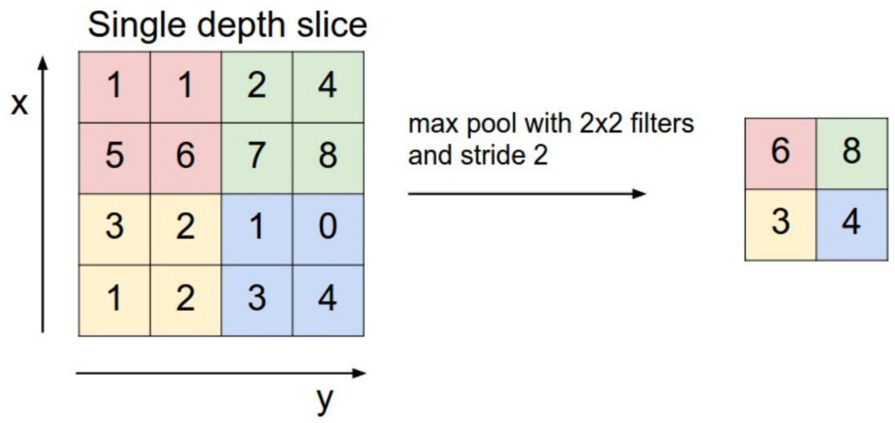
Reference architecture: AlexNet

[**"ImageNet Classification with Deep Convolutional Neural Networks"**, A Krizhevsky, I Sutskever, GE Hinton, NIPS, 2012]
["**A Taxonomy of Deep Convolutional Neural Nets for Computer Vision**", S. Srinivas, et al, 2016]



Pooling (P1, P2, P5)

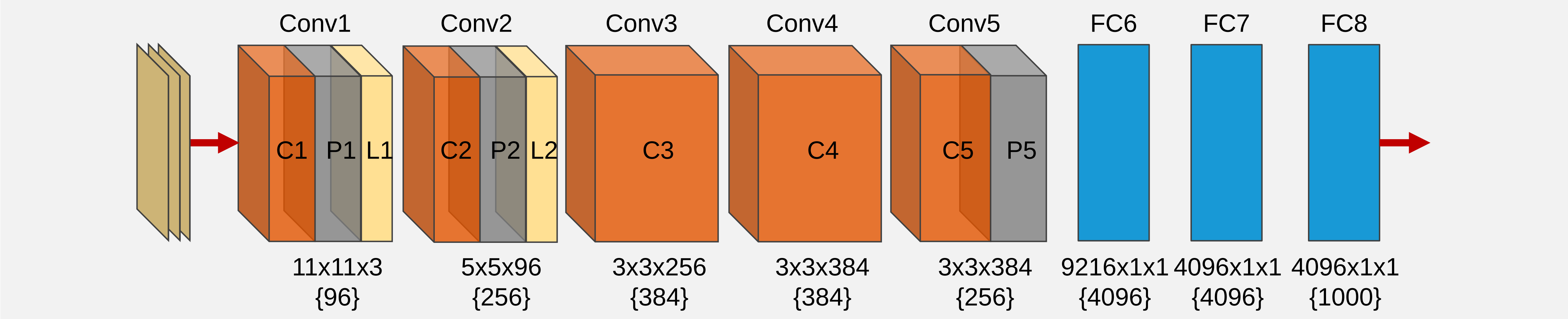
- An operation which fuses multiple inputs into an aggregated output
- Increases robustness
- Often used: **Max Pooling**



From Neural Networks to Deep Learning

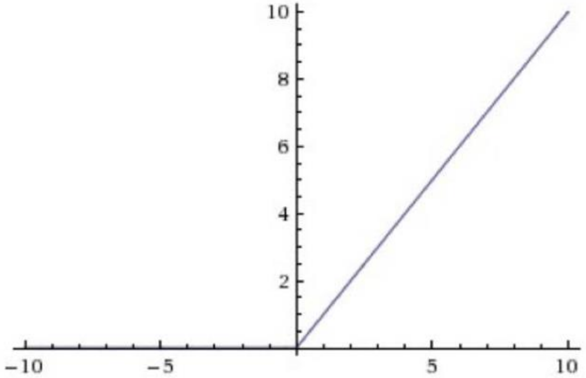
Reference architecture: AlexNet

[“**ImageNet Classification with Deep Convolutional Neural Networks”**, A Krizhevsky, I Sutskever, GE Hinton, NIPS, 2012]
[“**A Taxonomy of Deep Convolutional Neural Nets for Computer Vision**”, S. Srinivas, et al, 2016]



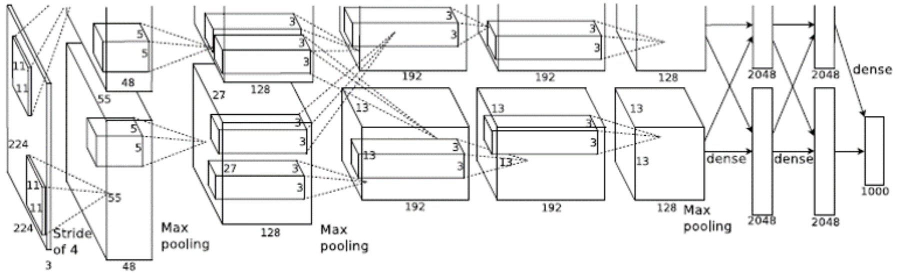
Activation function

- Sigmoid Function
- Rectified Linear Units (ReLU)
  – $f(x) = \max(0,x)$
- Responsible for the non-linear behavior of CNNs



From Neural Networks to Deep Learning

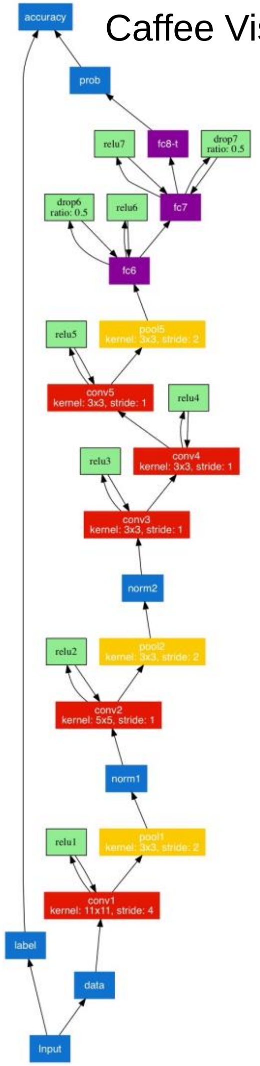
Visualization of deep learning CNN architectures



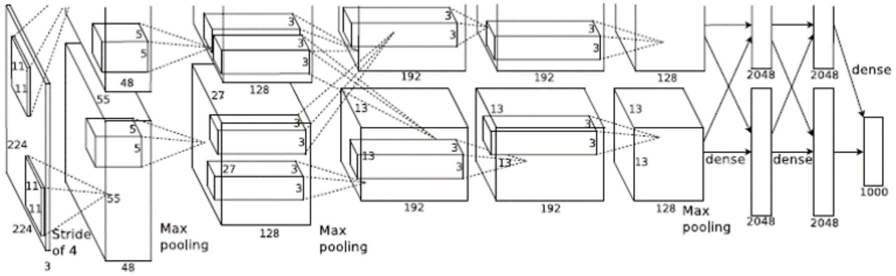
Original AlexNet Visualization

From Neural Networks to Deep Learning

Visualization of deep learning CNN architectures



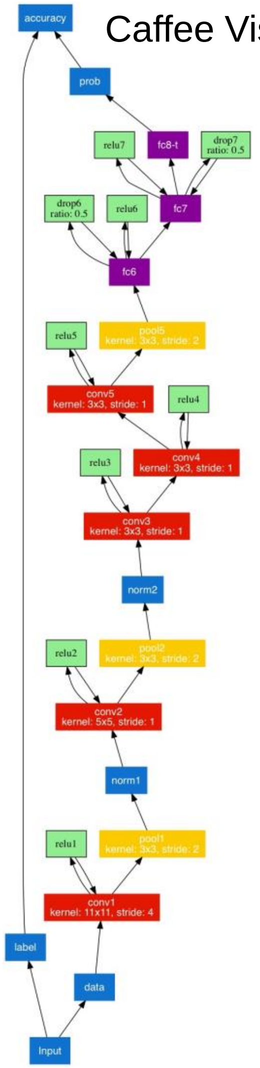
Caffee Visualization



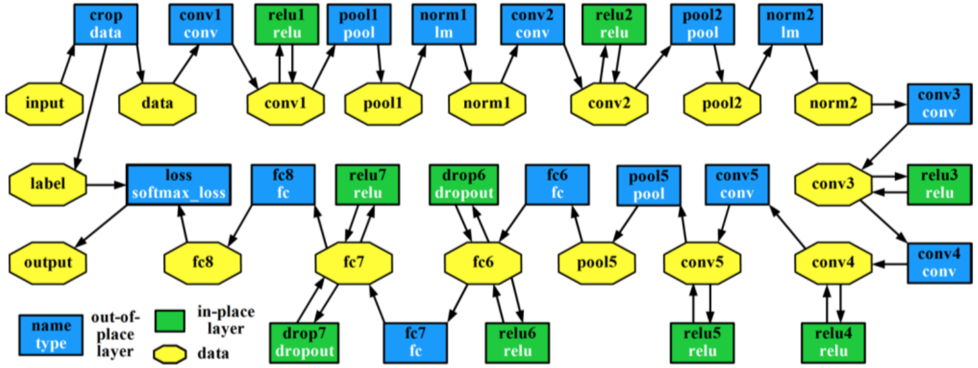
Original AlexNet Visualization

From Neural Networks to Deep Learning

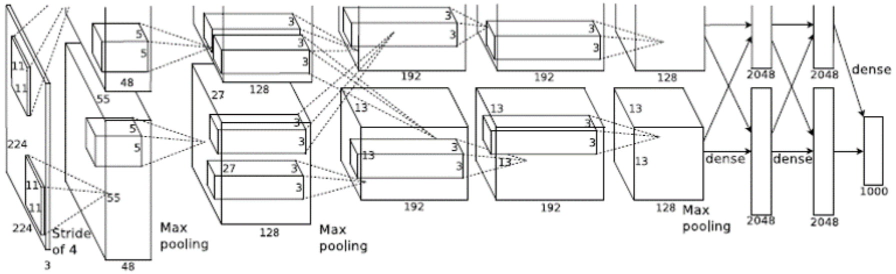
Visualization of deep learning CNN architectures



Caffee Visualization



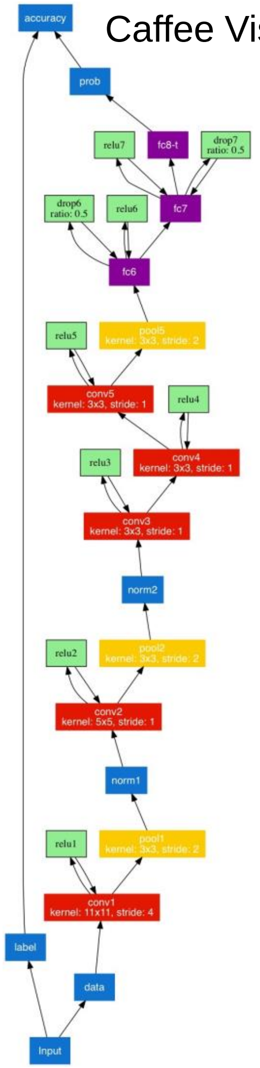
Layer / Blob
Caffee Visualization



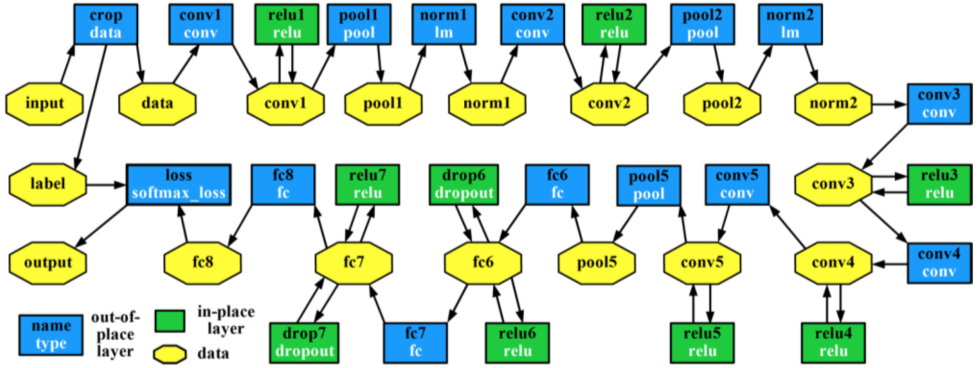
Original AlexNet
Visualization

From Neural Networks to Deep Learning

Visualization of deep learning CNN architectures

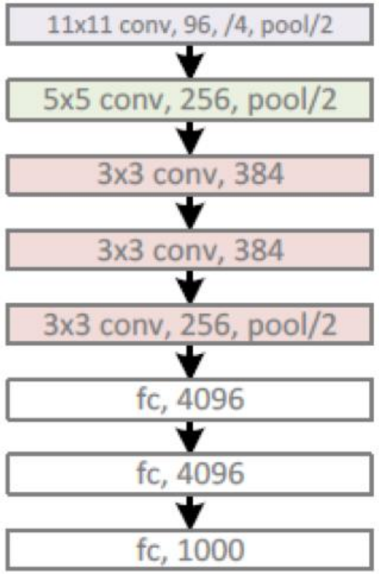


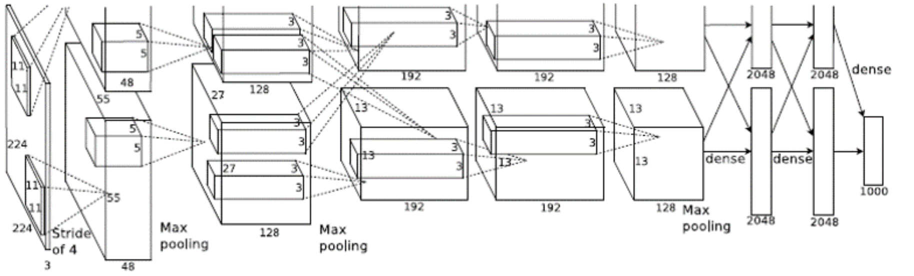
Caffee Visualization



Layer / Blob
Caffee Visualization

Stacked Layer
Visualization

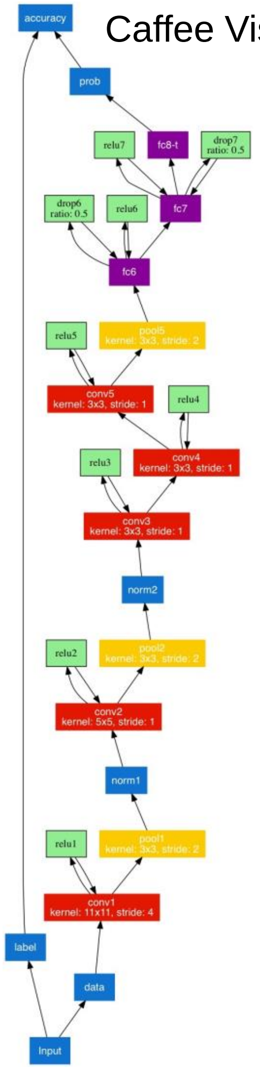




Original AlexNet
Visualization

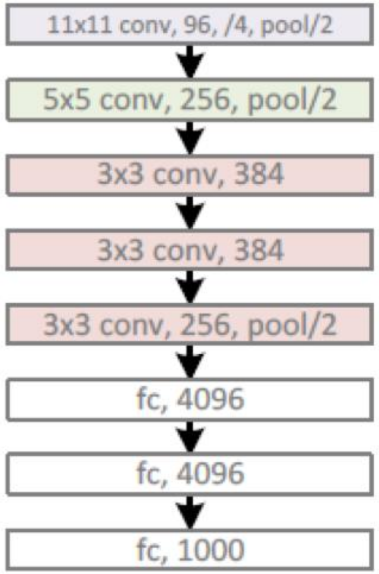
From Neural Networks to Deep Learning

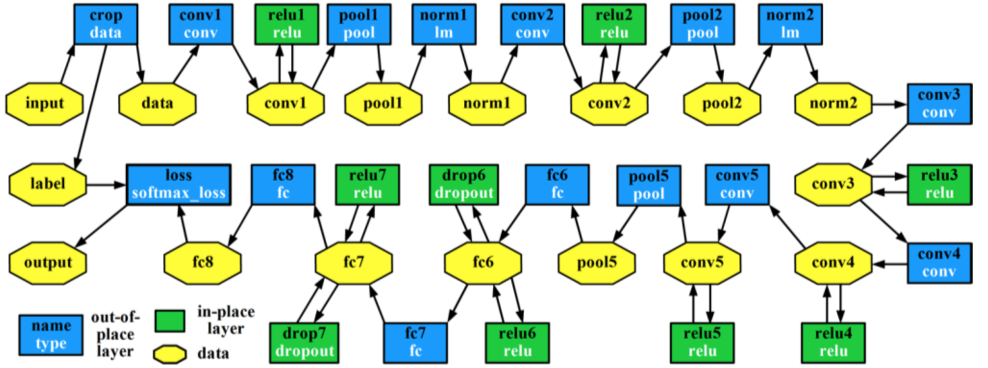
Visualization of deep learning CNN architectures



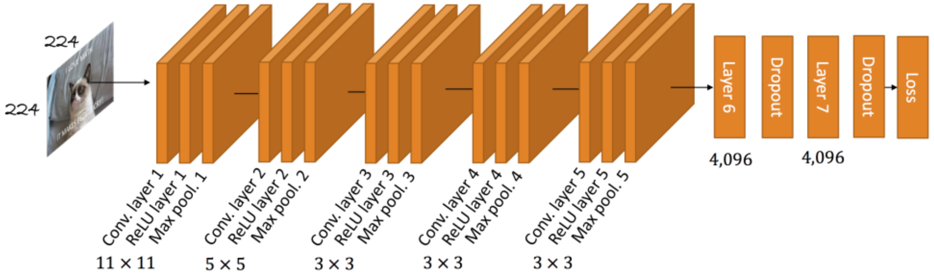
Caffee Visualization

Stacked Layer Visualization

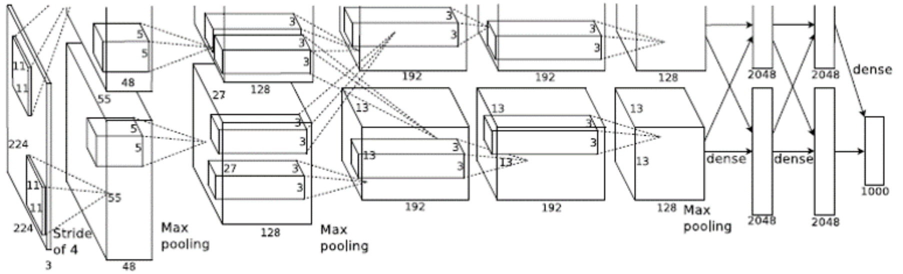




Layer / Blob
Caffee Visualization



Volume Layer Visualization

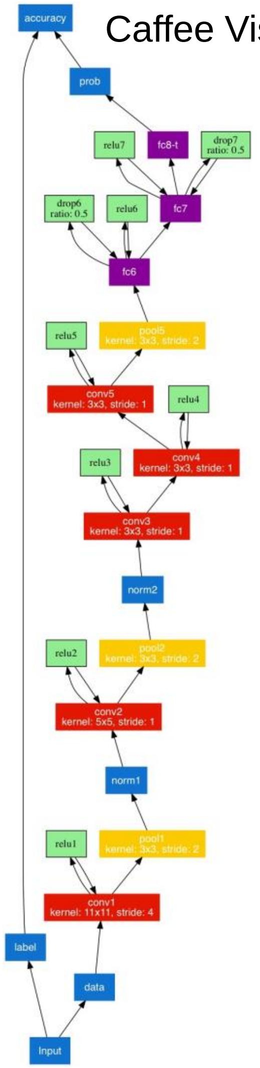


Original AlexNet Visualization

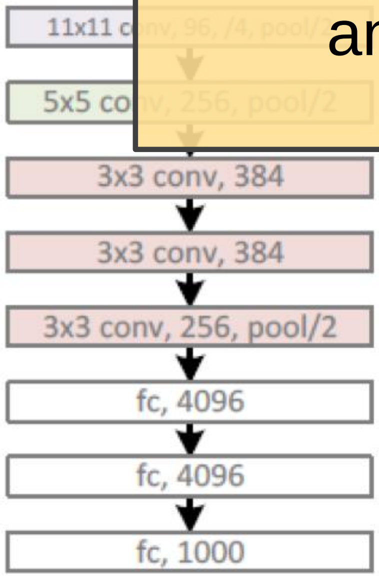
From Neural Networks to Deep Learning

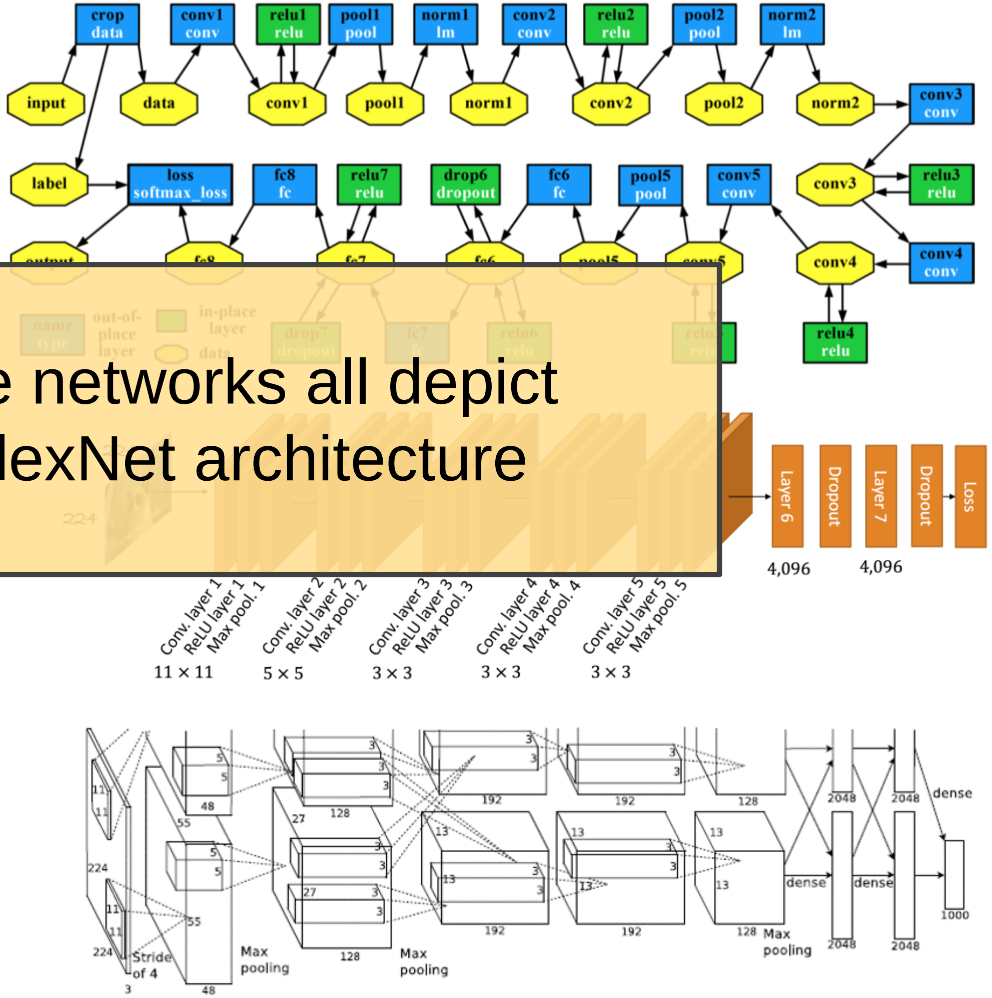
Visualization of deep learning CNN architectures

Caffee Visualization



Stacked Layer Visualization





Layer / Blob Caffee Visualization

Volume Layer Visualization

Original AlexNet Visualization

These networks all depict an AlexNet architecture

From Neural Networks to Deep Learning

Available deep learning frameworks

PyTorch	Caffe	TensorFlow	Torch
http://pytorch.org	http://caffe.berkeleyvision.org	https://www.tensorflow.org	http://torch.ch
MXNet	Microsoft CNTK	Theano	Keras
http://mxnet.incubator.apache.org	https://github.com/Microsoft/CNTK	http://deeplearning.net/software/theano/	https://keras.io

Thank you.

Contact information:

Dr. Damian Borth
Deep Learning Competence Center

German Research Center for Artificial Intelligence (DFKI)

http://www.dfki.uni-kl.de/~borth/

Week 1: Getting Started with Deep Learning

Unit 3: Getting Started with Jupyter Notebooks

Getting Started with Jupyter Notebooks

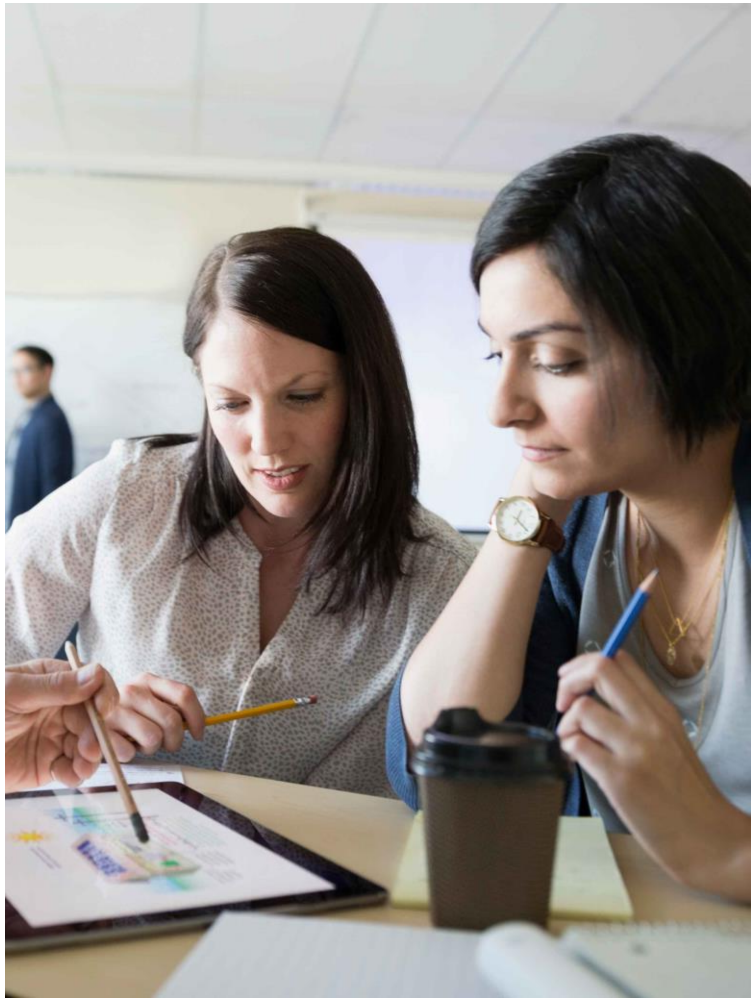
What we covered in the last unit

A brief history of machine learning and neural networks

Early works on ANNs

Common terminology used in this course

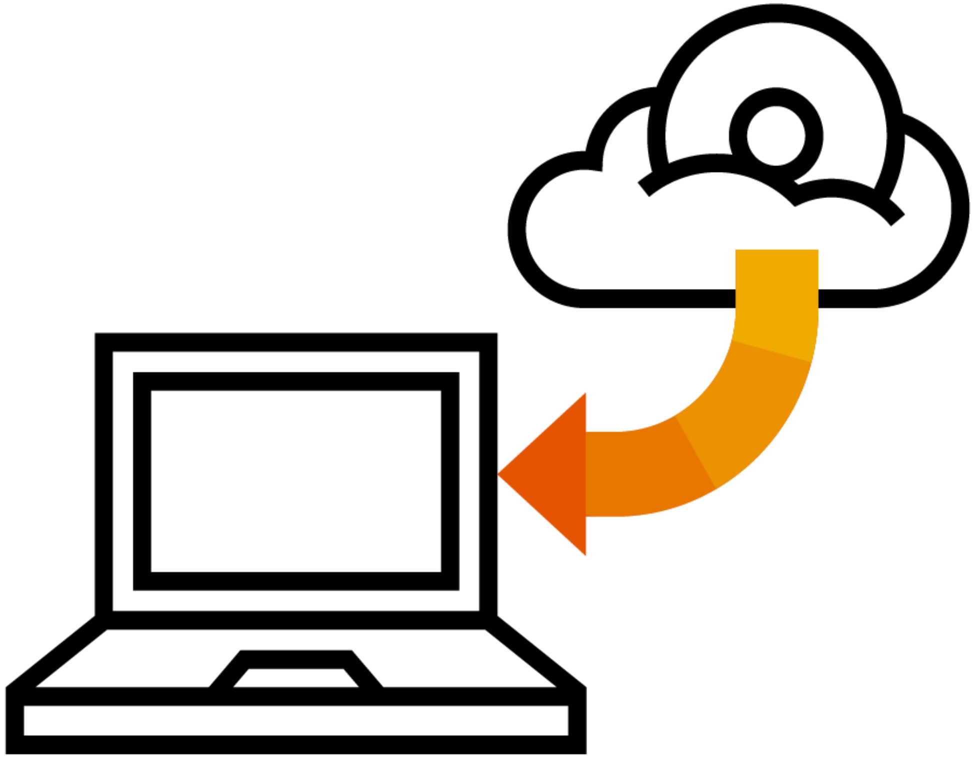
Relationship between various terms



Getting Started with Jupyter Notebooks

Software and libraries

Name	Version	Remarks
Anaconda Python	4.3.24 3.5 / 3.6	Package and environment manager for Python with a collection of open source packages
TensorFlow	1.3.0	Machine learning toolkit for developing and deploying ML frameworks in a large scale
NumPy	1.13.1	Python library for scientific computing
Jupyter	4.3.0	Web application for live coding, data exploration, visualization, numerical computation, and more
Git	2.11.0	Software version control



Getting Started with Jupyter Notebooks

Anaconda

What is Anaconda?

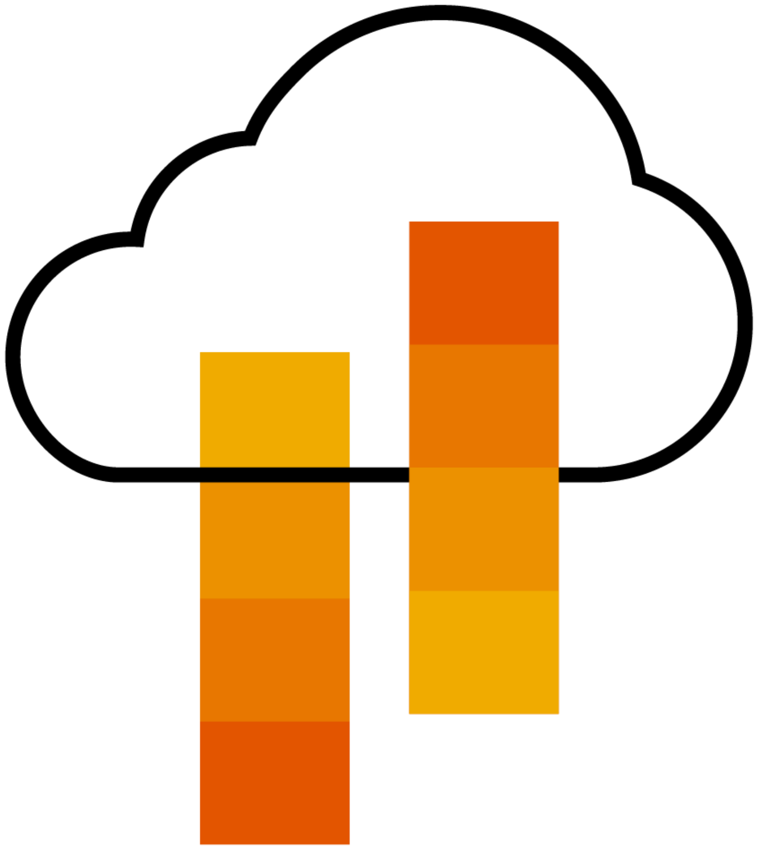
- Python package manager
- Easy installation of Python packages across platforms
- Allows sandboxed virtual environments

What operating systems are supported by Anaconda?

- Supports Linux, Mac OS, and Windows OS

Can I install packages on demand?

- Yes, you can run lighter versions – Miniconda



Getting Started with Jupyter Notebooks

Jupyter

What is Jupyter?

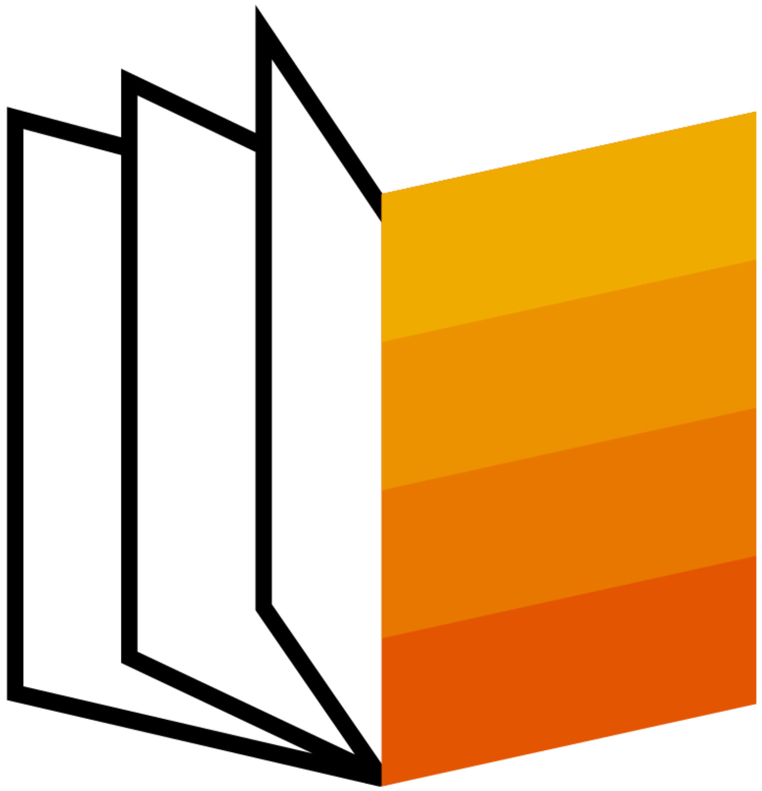
- Web application for live coding
- Separates language-agnostic layers from kernels
- Interactive data science and scientific computing

Why is it named Jupyter?

- Supports **Ju**lia, **Py**thon and **R**

How do we use Jupyter? What can we do with it?

- Let’s look at what can be done with Jupyter notebooks



Getting Started with Jupyter Notebooks

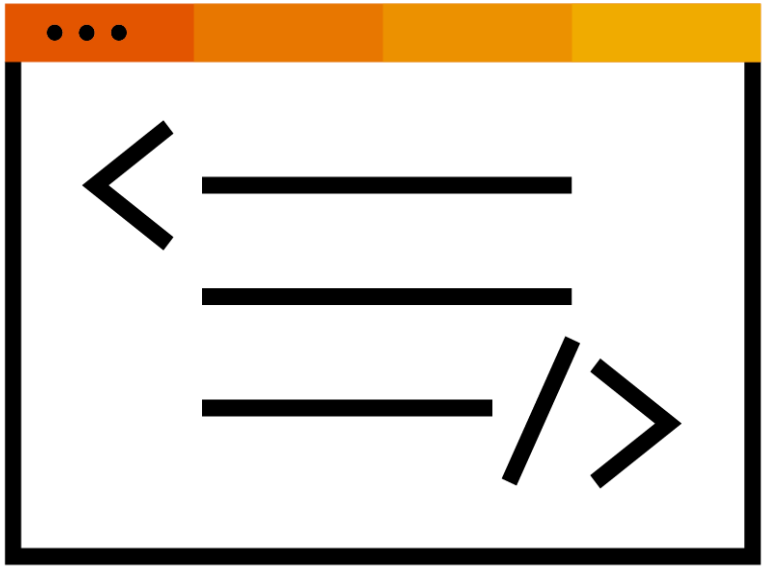
GitHub

What is GitHub?

- A version control repository

What is Git?

- Version control system
- Supports collaboration among developers
- Reduces code collision between collaborators
- Supports local operations to repository



Getting Started with Jupyter Notebooks

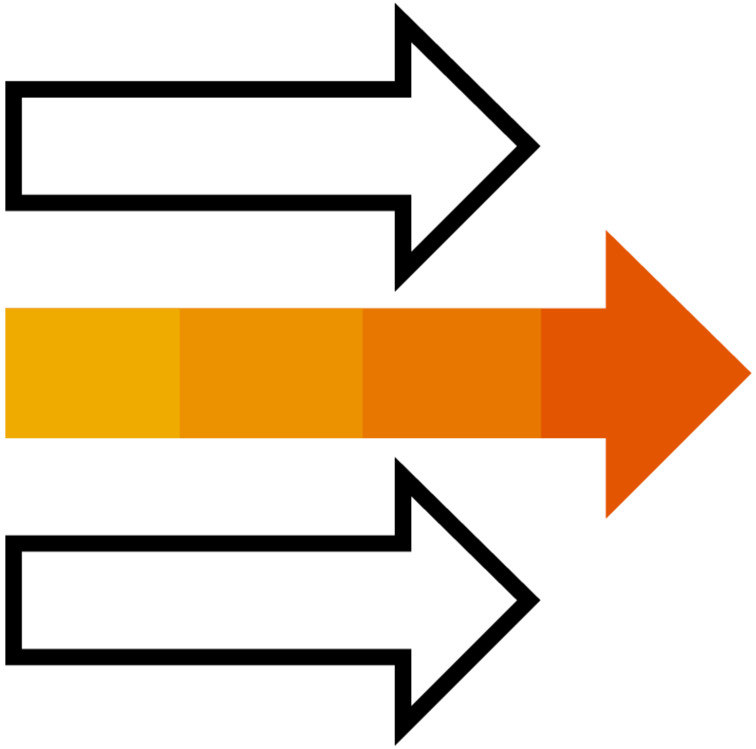
Coming up next

Understand neural networks

Use NumPy to build your first neural network

Explore math behind neural networks learning

Understand some key terms used in ML



Thank you.

Contact information:

open@sap.com

Week 1: Getting Started with Deep Learning

Unit 4: Building Your First Neural Network

Run Simple

openSAP
open.sap.com

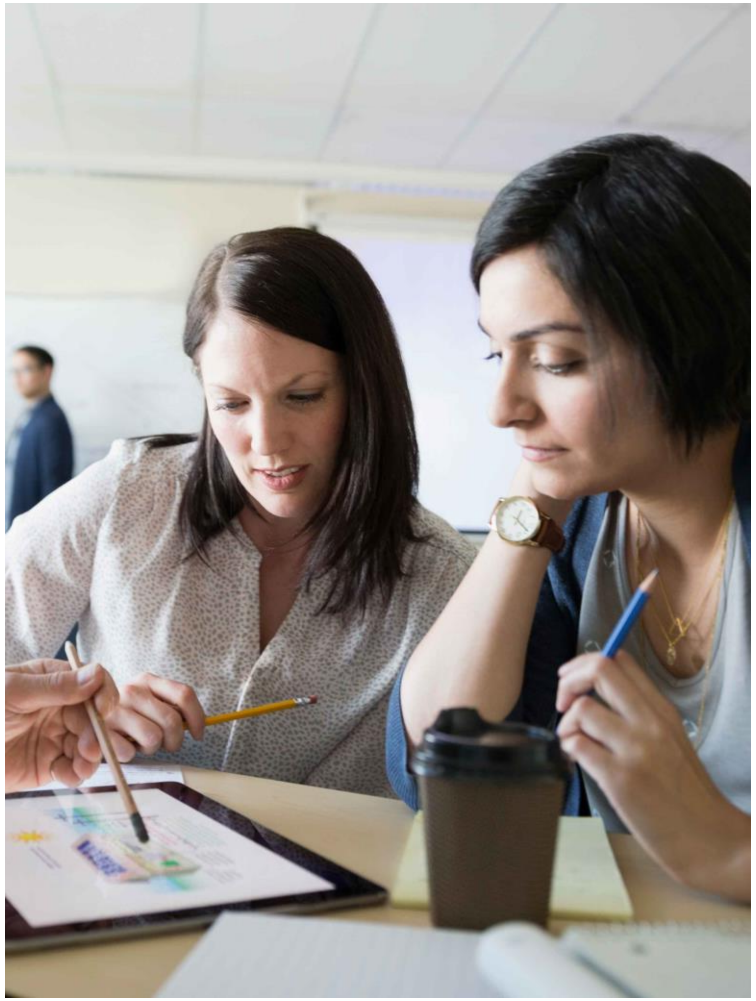
Building Your First Neural Network

What we covered in the last unit

Introduced software and libraries used in this course

Learnt about code versioning using GitHub

Explored Jupyter and ran Python code in a browser



Building Your First Neural Network

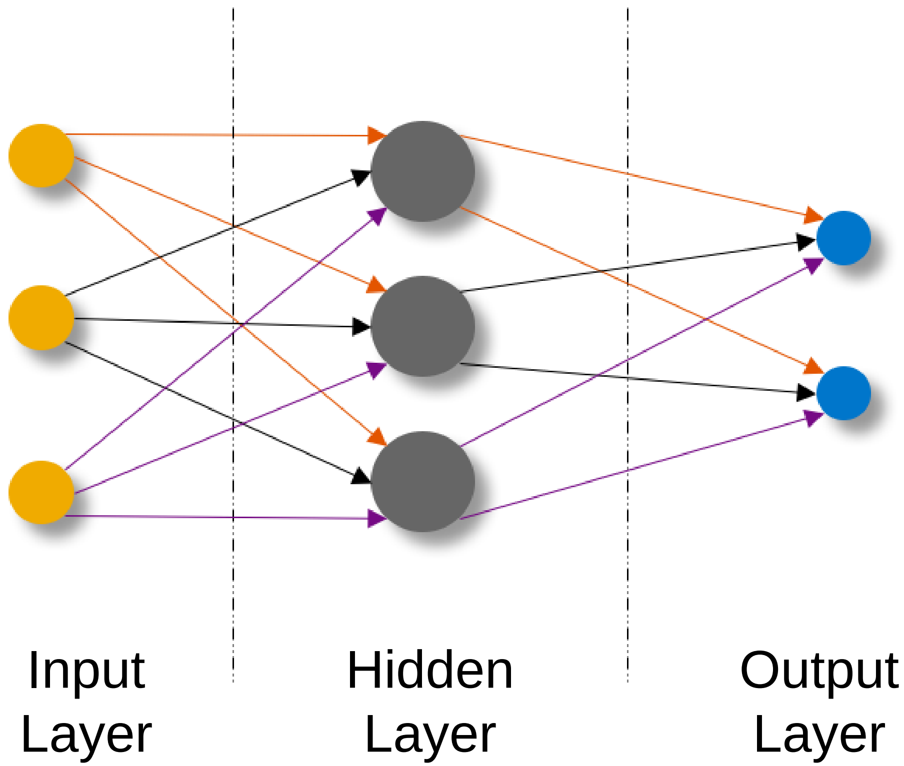
Neural networks

What is a neural network?

- Computer program that learns from input data
- Idea inspired by behavior of neurons in human brain

Why are neural networks so widely used?

- No need for explicit rules or features
- Exceptionally good at recognizing patterns in data



Building Your First Neural Network

TensorFlow playground

What is TensorFlow playground?

- Lightweight neural network capable of running on a browser
- Teaches about hyperparameters and their impact on output

What types of tasks can you experiment with on TensorFlow playground?

- Allows experimentation with network designs for classification and regression tasks

How do you run an experiment on TensorFlow playground?

- Let’s solve a simple classification problem on playground.tensorflow.org

Building Your First Neural Network

Neural networks and learning

How does a neural network learn?

- Neuron weights are updated with the objective of reducing error

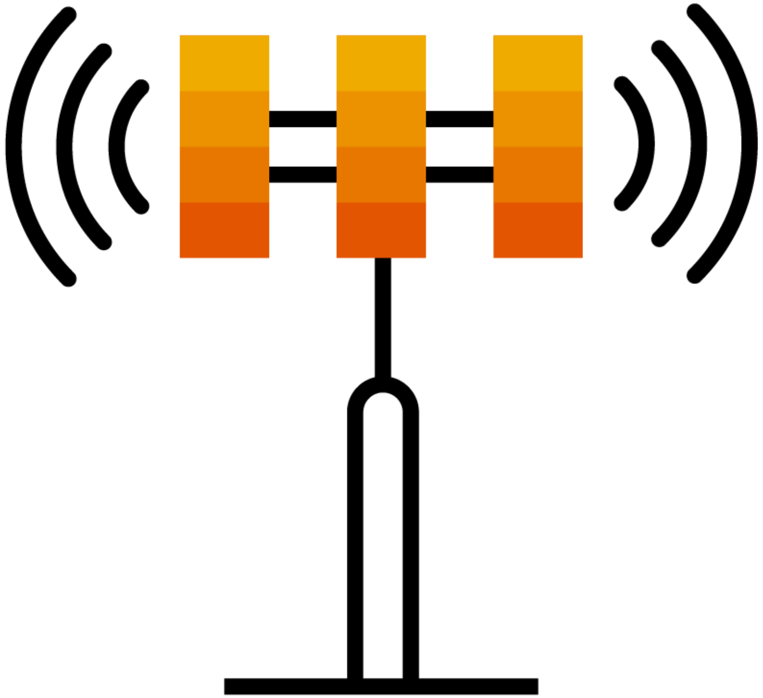
How is error calculated?

- Using output labels corresponding to each input

What happens with the error?

- Error is propagated back to the network to alter weights of neurons – backpropagation

Task: Implement backpropagation using NumPy for a simple problem



Building Your First Neural Network

Coming up next

Introduction to TensorFlow, a versatile machine learning library

Understand how TensorFlow represents data

Explore concepts such as compute graphs

Solve a simple ML problem using TensorFlow

Understand capabilities of TensorFlow

Thank you.

Contact information:

open@sap.com

Run Simple

openSAP
open.sap.com

Week 1: Getting Started with Deep Learning

Unit 5: Introduction to TensorFlow

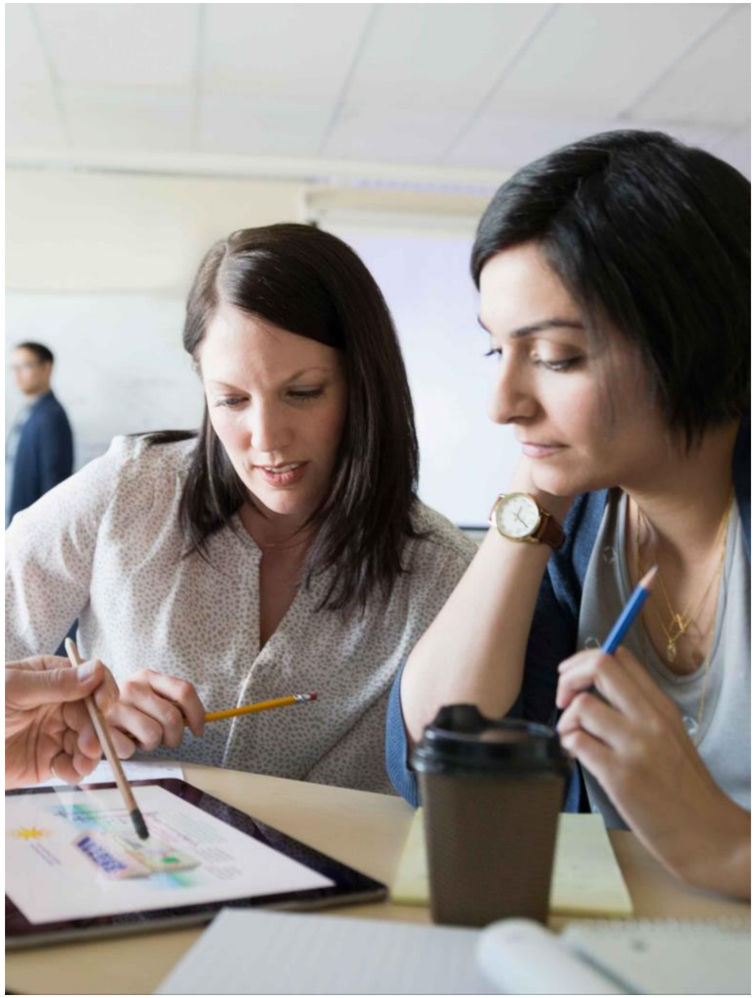
Run Simple

openSAP
open.sap.com

Introduction to TensorFlow

What we covered in the last unit

- Used NumPy to build your first neural network
- Explored some math behind neural networks
- Experimented with TensorFlow playground
- Learned some key terms used in ML



Introduction to TensorFlow

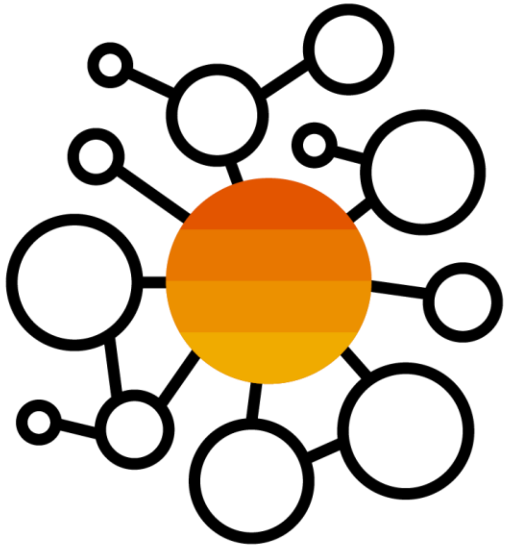
A versatile machine learning library

What is TensorFlow?

- An open-source library for numerical computation
- Released by Google in 2015
- Apache 2.0 license

Why TensorFlow?

- A highly flexible library for developing machine learning frameworks
- An end-to-end library for development and deployment
- Easy to move from prototyping to production

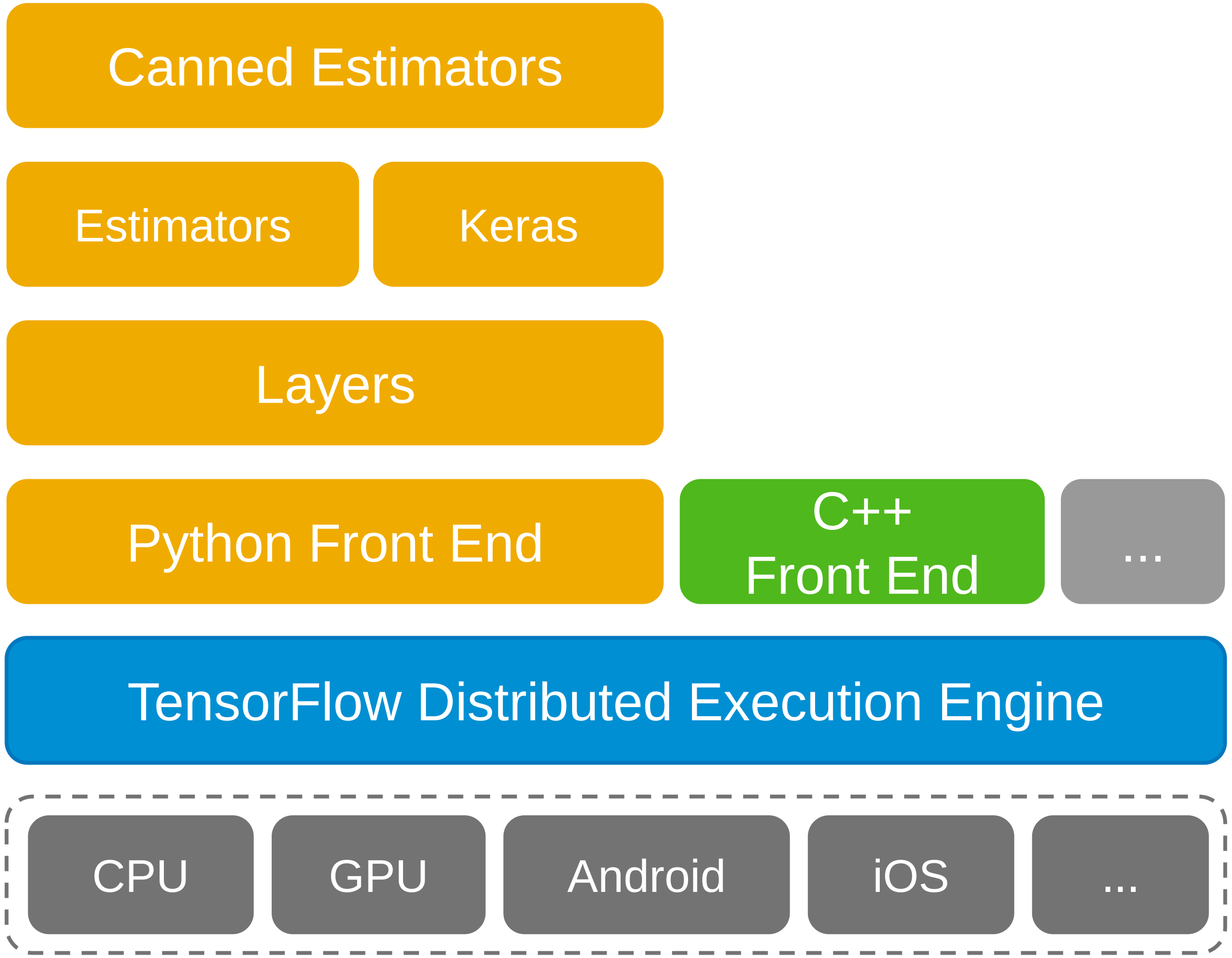


Introduction to TensorFlow

TensorFlow architecture

How does TensorFlow go from development to deployment?

- Device-agnostic execution framework
- Supports multiple language front ends
- Layers allows easy building of models
- Estimators provide a standard interface
- TF Keras is an API spec built on core TensorFlow features
- Canned estimators provide efficient implementations of standard models



Introduction to TensorFlow

In-depth look into TensorFlow

What is a tensor?

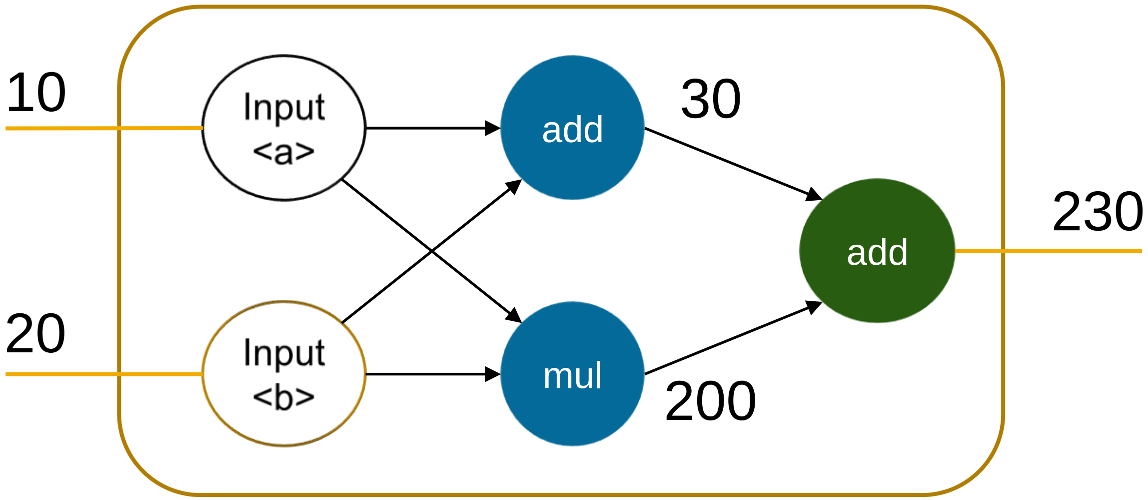
- An n-dimensional matrix used to represent data

What are TensorFlow operations?

- TensorFlow operations are nodes that perform computations on or with tensor objects

What is a computation graph?

- A graph used for defining the computation structure



$$c = (a + b) + (a \times b)$$

Introduction to TensorFlow
TensorFlow sessions

What do TensorFlow sessions do?
- Sessions are responsible for graph execution

What do fetches do?
- Allow you to evaluate either a tensor or an operation
- Tensor objects return NumPy arrays

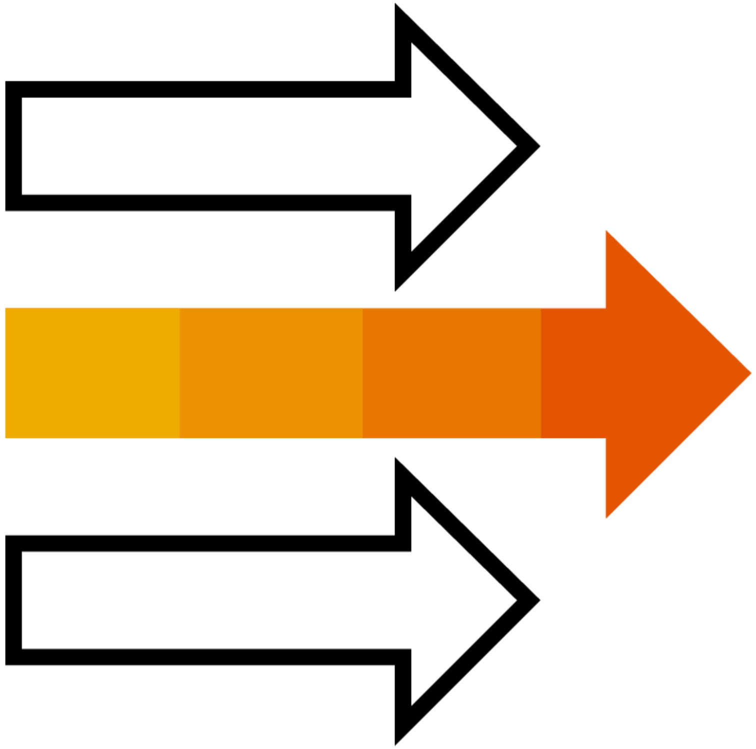
What does a feed dictionary do?
- Allows you to override tensor objects in a graph
- Useful when inputs are not known a priori

Putting it all together – TensorFlow program

Introduction to TensorFlow

Coming up next

- Learn when to use deep learning
- Learn when not to use deep learning
- Understand when classical machine learning techniques or rule-based approaches are preferable



Thank you.

Contact information:

open@sap.com

Week 1: Getting Started with Deep Learning

Unit 6: When to Use Deep Learning

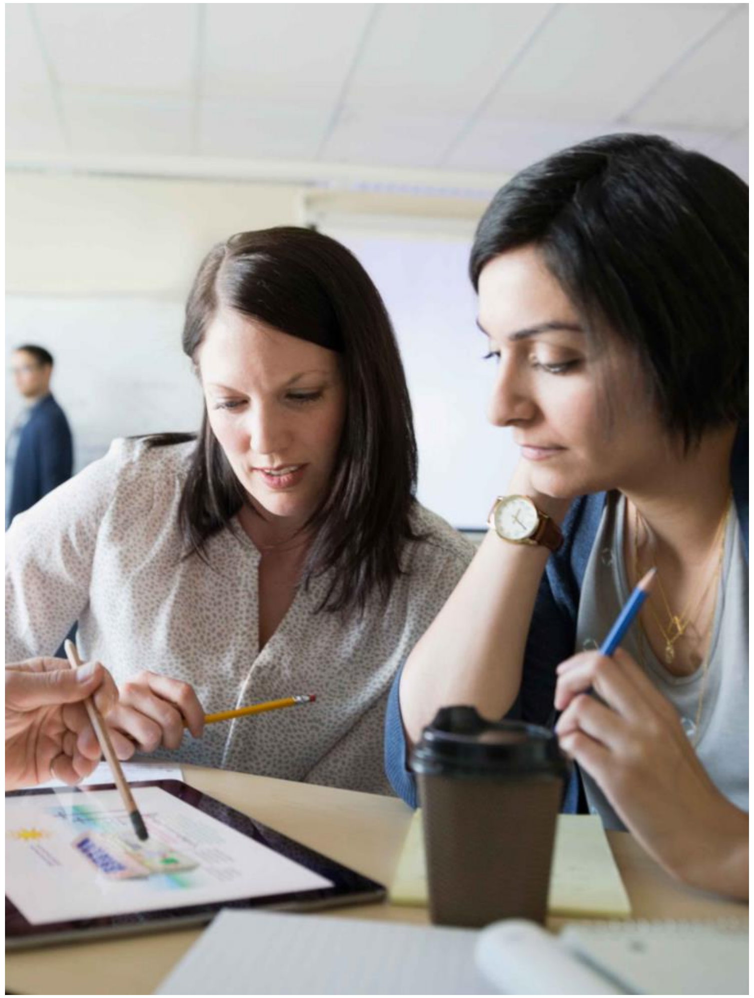
Run Simple

open.sap.com

When to Use Deep Learning

What we covered in the last unit

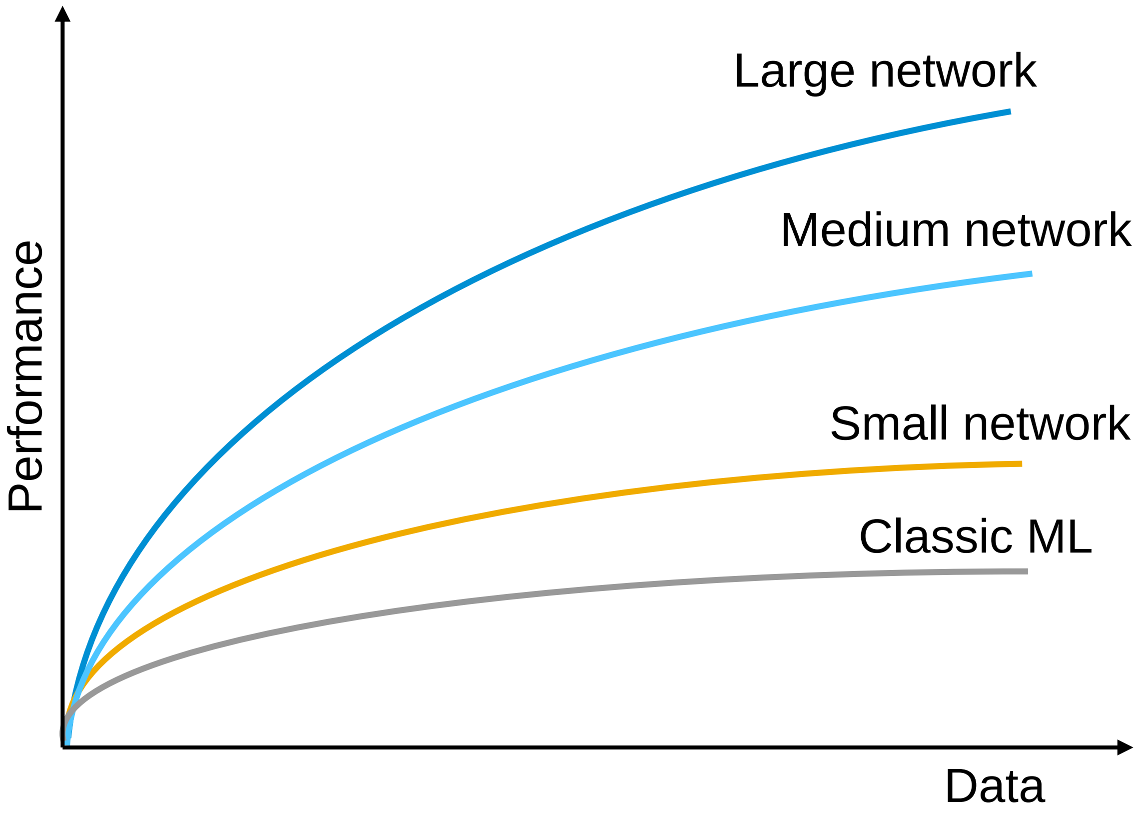
- Basic concepts of TensorFlow
- How TensorFlow makes neural network development easier and more scalable



When to Use Deep Learning

Why deep learning?

- Performance of classic machine learning algorithms often plateaus with more data
- Deep learning networks can take advantage of large data sets
- Larger networks scale to larger data sets



When to Use Deep Learning

When to use deep learning

Deep learning is a promising approach when…

- you have a large amount of training data
- you are solving an image/audio/natural language problem
- the raw input data has little structure and you need the model to learn meaningful representations (e.g., pixels in an image)

When to Use Deep Learning

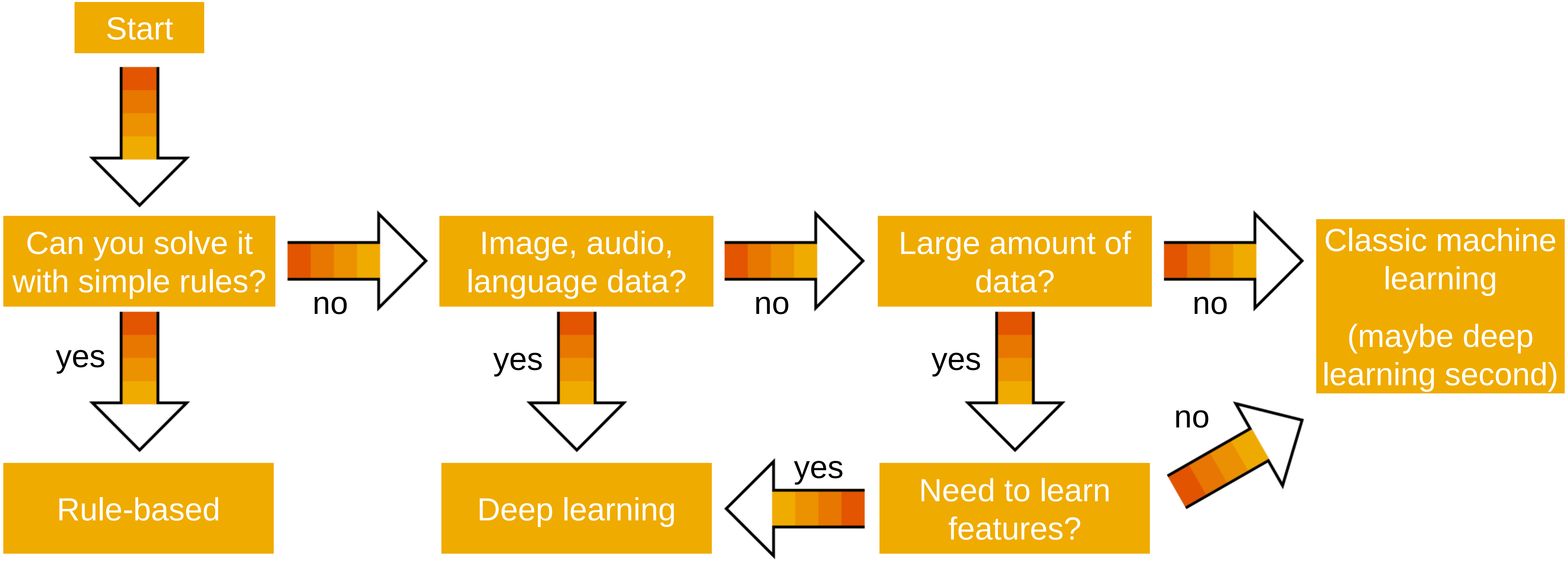
When NOT to use deep learning

Deep learning is a less promising approach when…

- you can solve the problem with simple hand-written rules (then just use the rules)
- you have a small amount of training data
- you have structured data

When to Use Deep Learning

When to use the deep learning “cheat sheet”



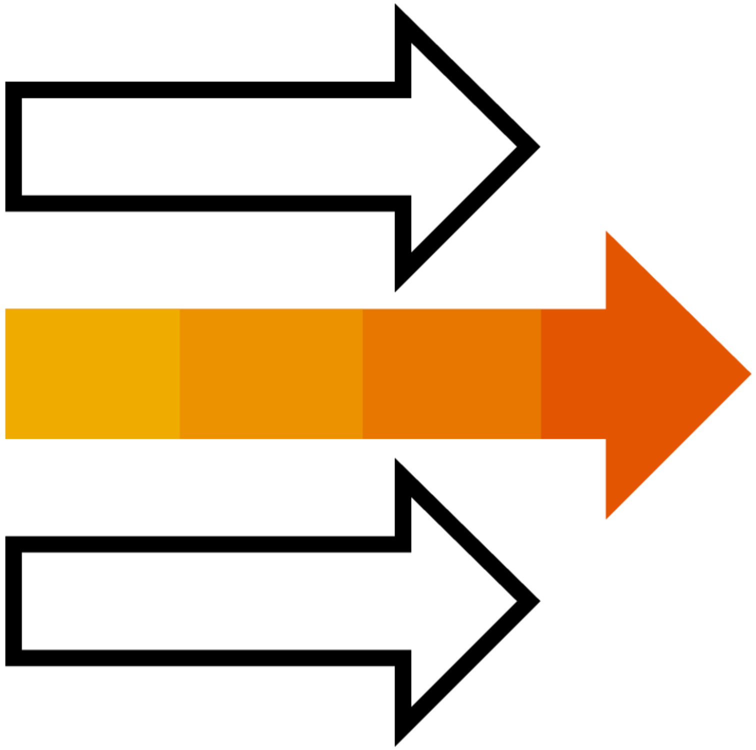
Build a simple baseline and iterate quickly

When to Use Deep Learning

Coming up next

Week 2: Shallow Neural Networks

- Nuts and bolts of machine learning experiments
- Download, visualize, and prepare data
- Build a model using TF estimators
- Serve a model using TF serving
- Understand architectures for deep learning



Thank you.

Contact information:

open@sap.com

Run Simple

open.sap.com