intel Software STUDENT DEVELOPER PROGRAM

ARTIFICIAL INTELLIGENCE

HOW TO ACTUALLY BUILD A NEURAL NETWORK FROM BLOCKS?

speaker: Jakub Czakon
moderator: Piotr Migdał

webinar, 6th July 2017

Competition

Objective:

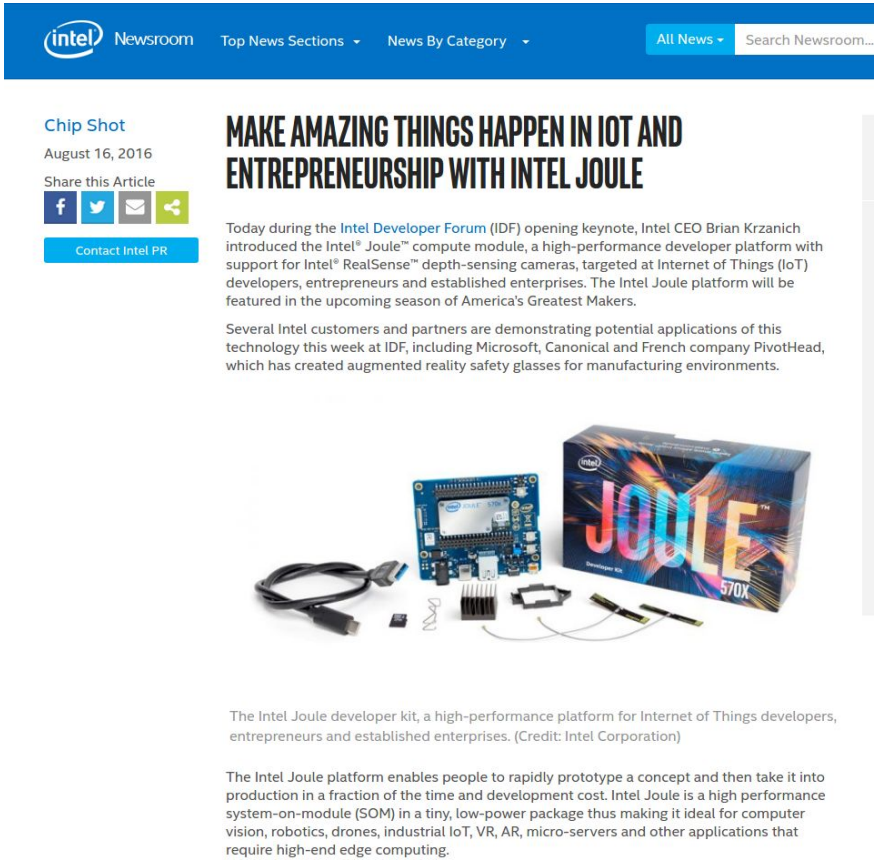
Answer questions

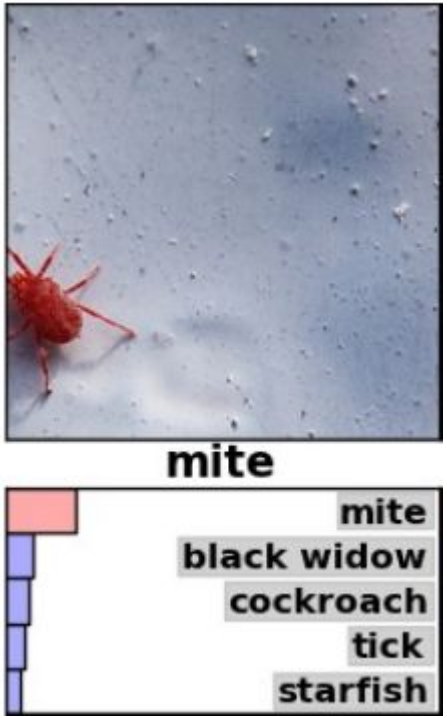
Rules:

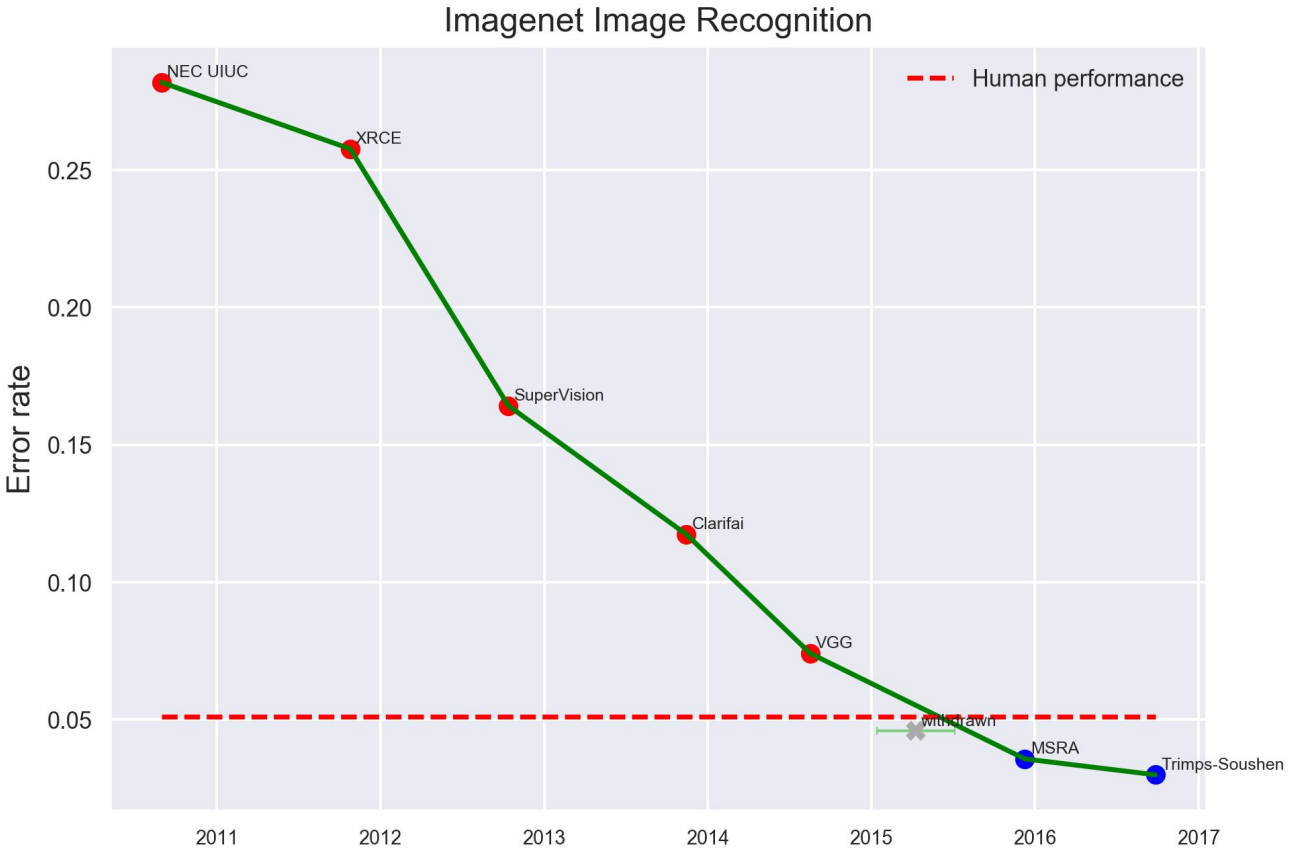
Do it well, do it fast

Prizes:

Intel® Joule™ 570x Developer Kit.







https://www.eff.org/ai/metrics

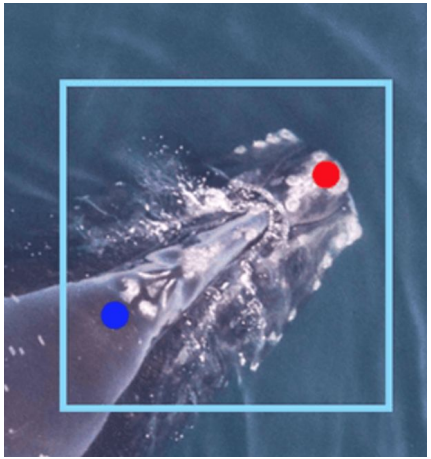
THE POWER OF NEURAL NETWORKS

https://twitter.com/andrewyng/status/788548053745569792

WHALE RECOGNITION

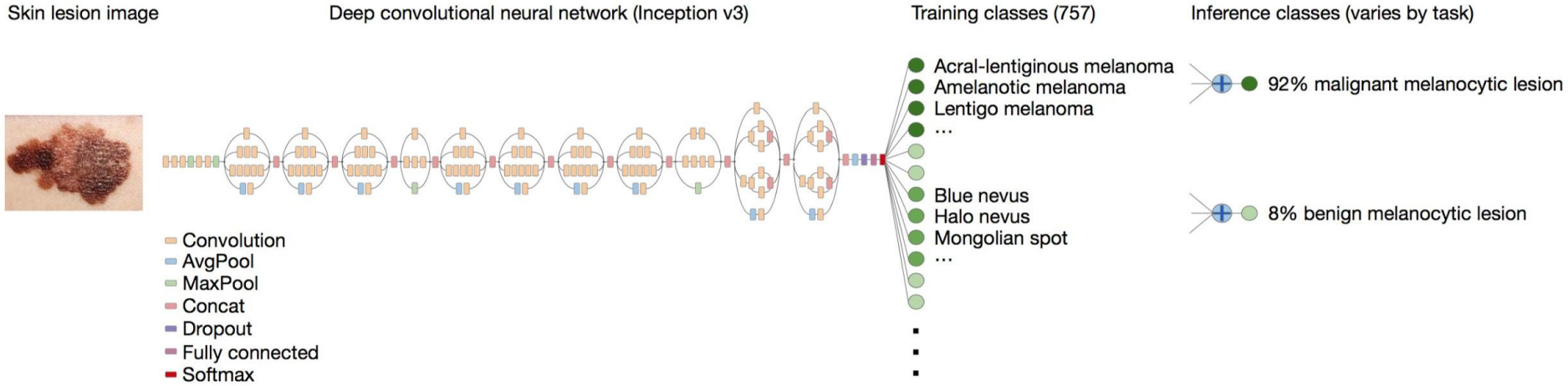
#	Δ1w	Team Name ‡ model uploaded * in the money	Score
1	↑27	deepsense.io ‡ *	0.61614
2	↑1	felixlaumon ‡ *	1.01971

- 447 individual whales
- 4.5k images
- 87% accuracy!



http://blog.kaggle.com/2016/01/29/noaa-right-whale-recognition-winners-interview-1st-place-deepsense-io/

MEDICAL-LEVEL SKIN CANCER DETECTION



http://cs.stanford.edu/people/esteva/nature/

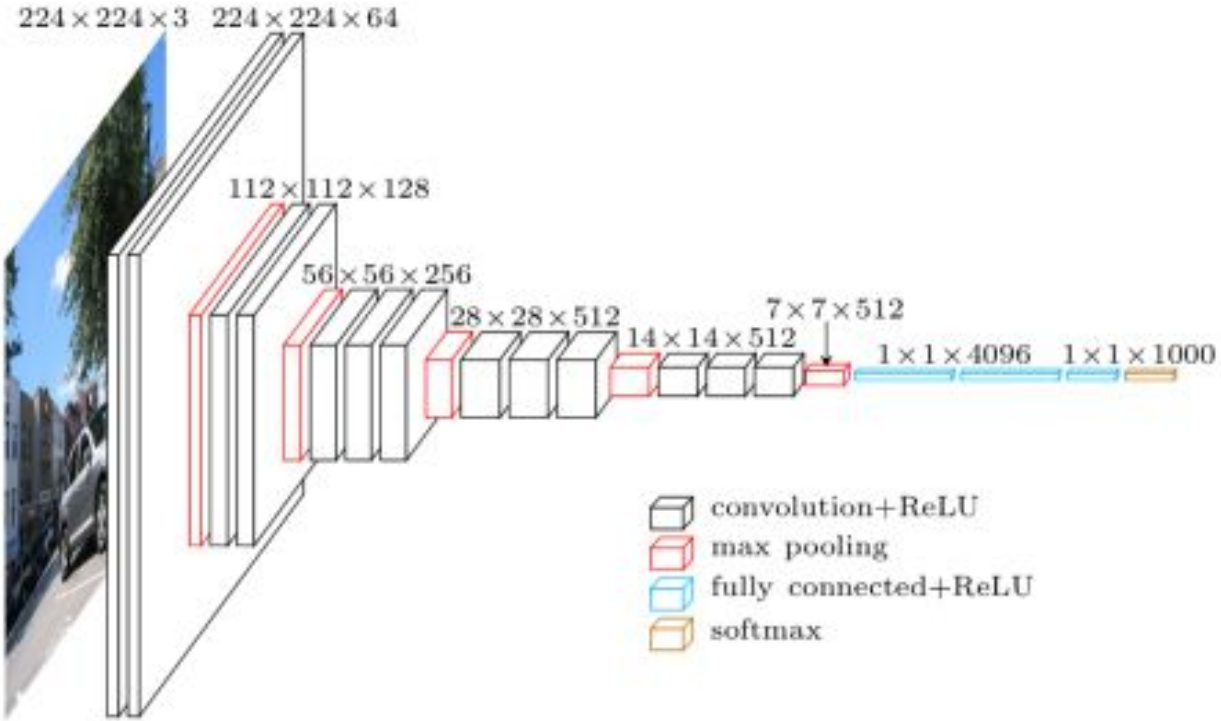
ARTIFICIAL INTELLIGENCE: A CHANGING FRONTIER

The toughest challenge facing AI workers is to answer the question: "What are the letters A and I?"

Douglas R. Hofstadter (1995)

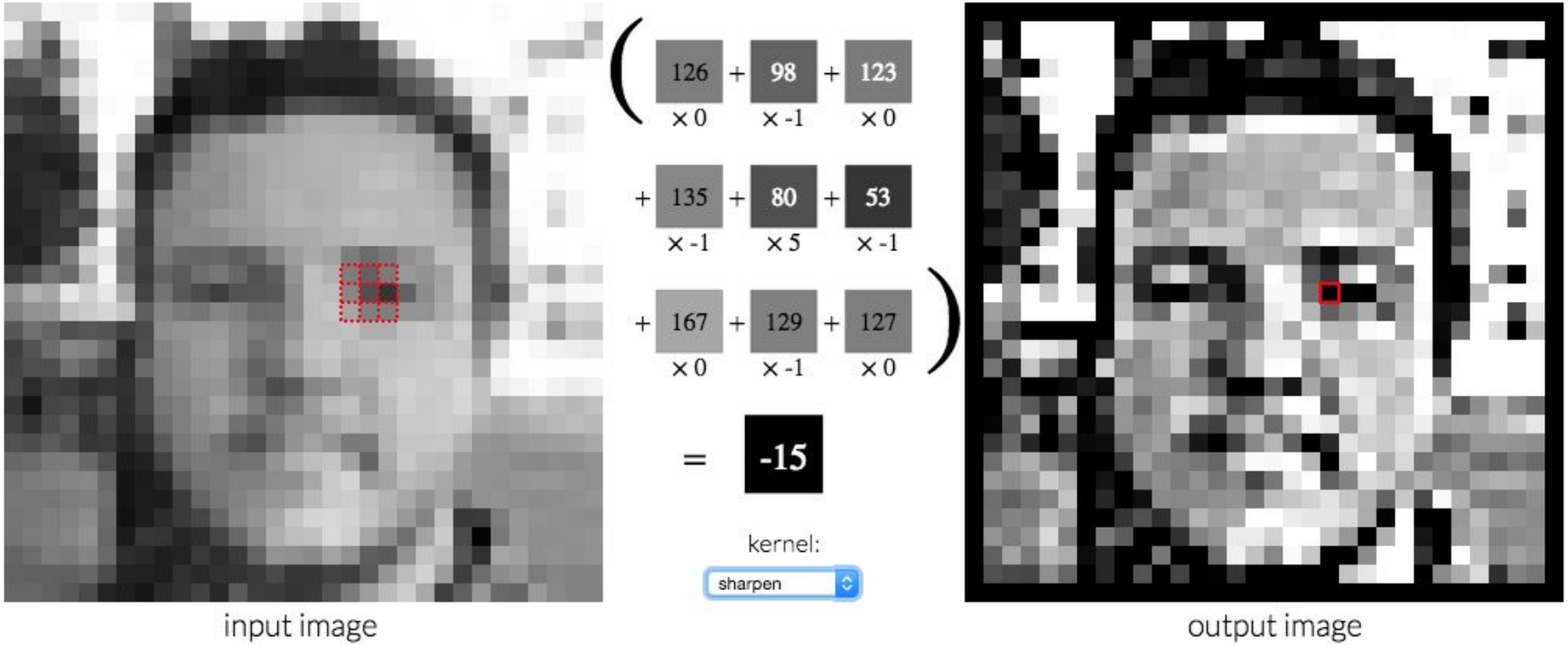
CONVOLUTIONAL NEURAL NETWORKS

CONVNET EXAMPLE: VGG16



https://www.cs.toronto.edu/~frossard/post/vgg16/

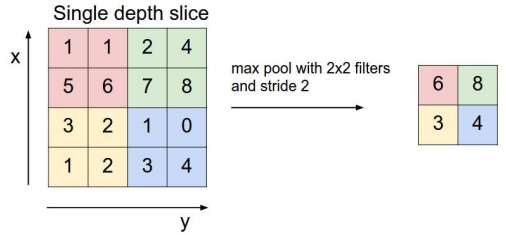
CONVOLUTIONS

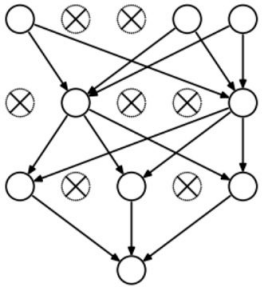


http://setosa.io/ev/image-kernels/

TYPICAL BUILDING BLOCKS

- convolutional layers
- activation function (tanh or ReLU)
- max pooling
- batch normalization
- dense layer
- softmax
- log-loss cost function
- optimizer

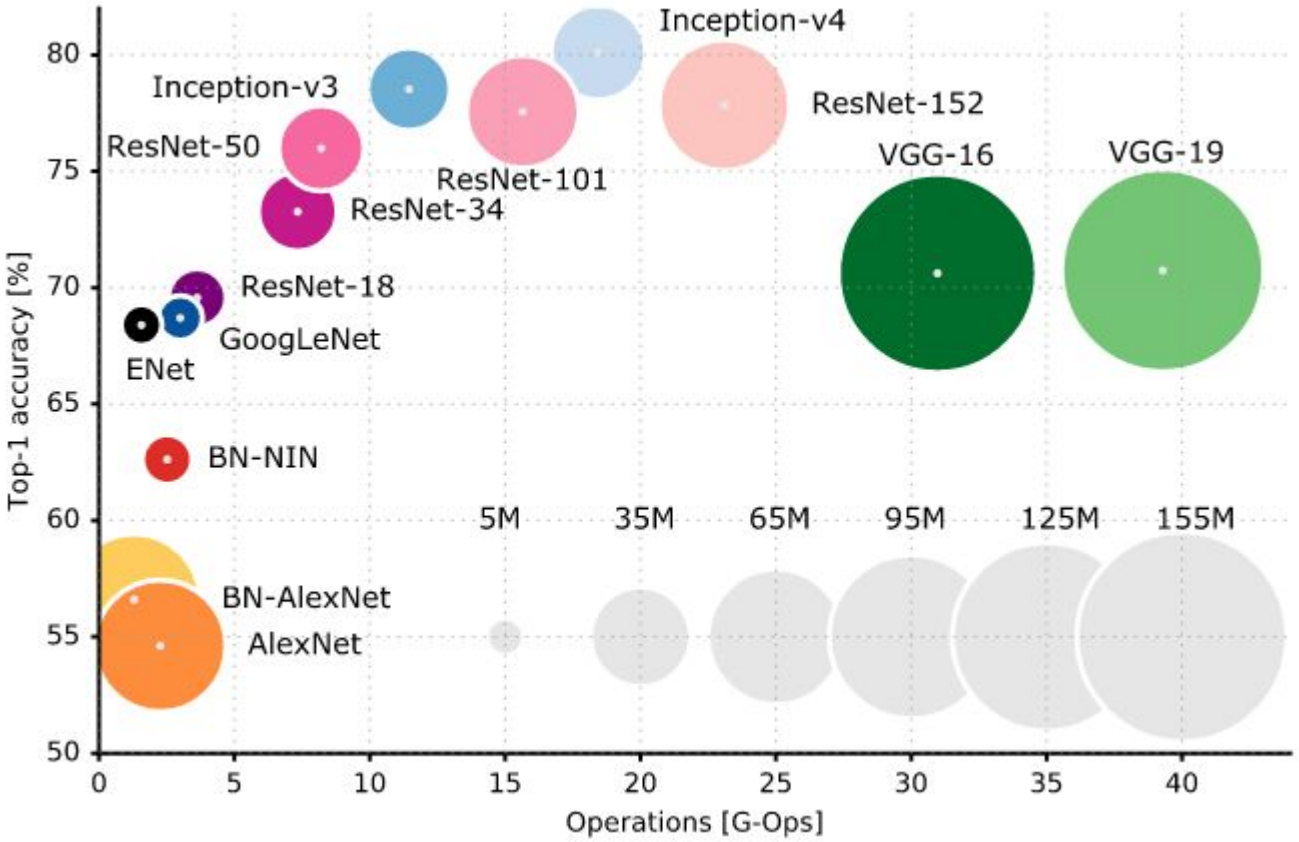




```
z = array([0.4, -0.9, 2.1])
```

```
# soft-max
exp(z) / sum(exp(z))
```

```
array([ 0.14822546,  0.04039615,  0.81137839])
```



https://medium.com/towards-data-science/neural-network-architectures-156e5bad51ba

TYPICAL APPROACHES

Training from scratch

- Full control over network architecture
- Requires a lot of data and time

Fine-tuning

- Using a pre-trained network (e.g. VGG16 or ResNet) and training it a bit
- Great for detecting similar objects

Feature generation

- Using a neural pre-trained network for features for other ML methods

Question 1

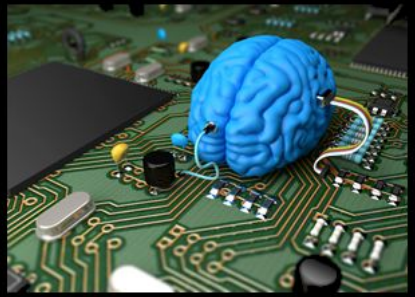
What is the correct order (largest to lowest) of the following neural networks in terms of the computation complexity of the forward pass?

AlexNet, VGG19, InceptionV3, ResNet152

LET’S GET PRACTICAL

Deep Learning

What society thinks I do



What my friends think I do

What other computer scientists think I do

What mathematicians think I do

What I think I do

```
In [1]:
import keras
Using TensorFlow backend.
```

What I actually do

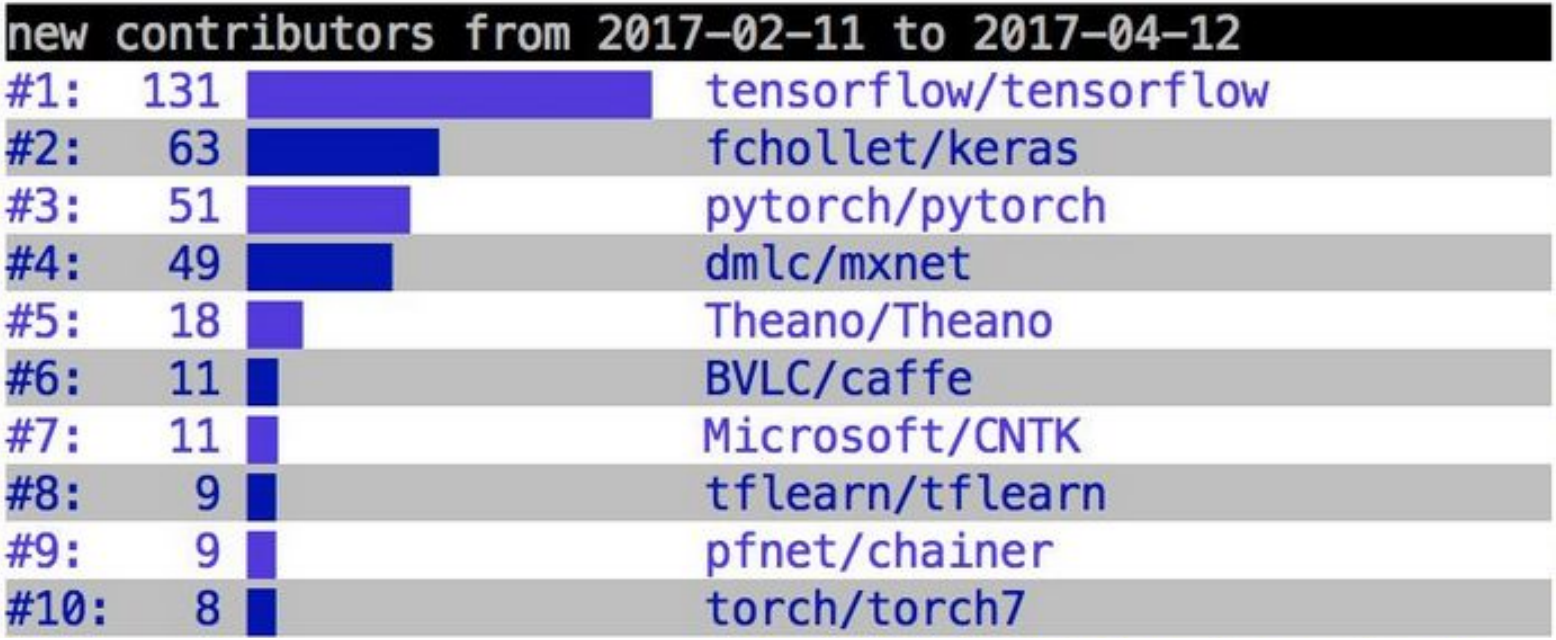
SOFTWARE FOR DEEP LEARNING

Question 2

What is the correct order (first commit on GitHub) the following frameworks were created in?

Keras, Theano, Caffe, MXnet, Torch

KERAS: HIGH-LEVEL FRAMEWORK



LIVE DEMONSTRATION

In [18]:

```
model = Sequential()

model.add(Conv2D(16, (3, 3), activation='relu',
                 input_shape=(resolution, resolution, 1)))
model.add(MaxPool2D())

model.add(Conv2D(32, (3, 3), activation='relu'))
model.add(MaxPool2D())

model.add(Flatten())
model.add(Dense(classes, activation='softmax'))

model.compile(loss='categorical_crossentropy',
              optimizer='adam',
              metrics=['accuracy'])
```

Question 3

Who is the owner of the following avatar on GitHub?

INTEL DEEP LEARNING SDK

Machine Learning: Your Path to Deeper Insight

Driving increasing innovation and competitive advantage across industries

intel strategy provides the foundation for success using AI

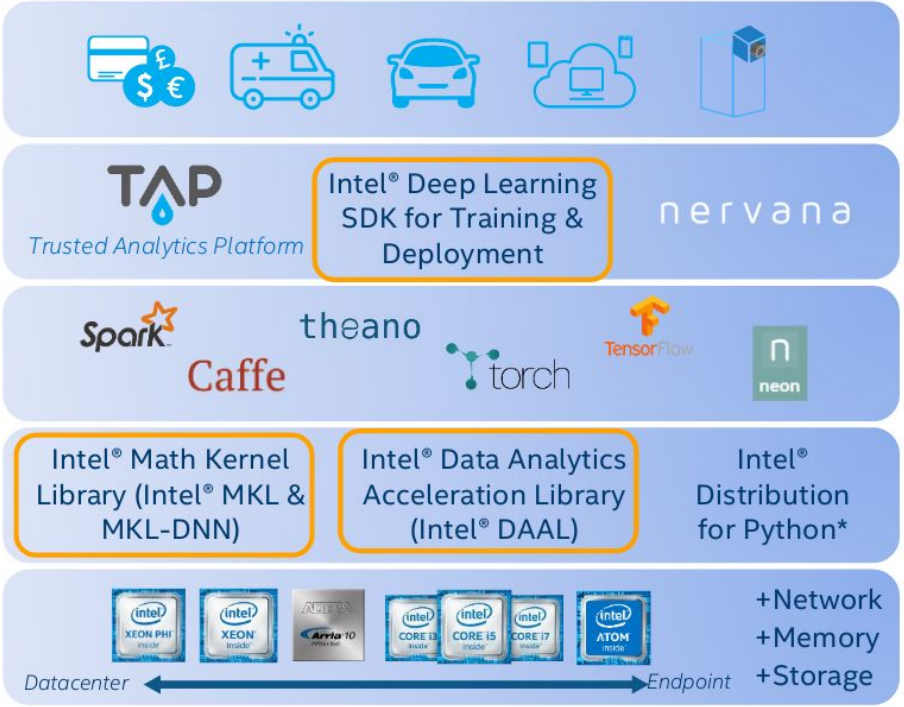
Solutions
for reference across industries

Tools/Platforms
to accelerate deployment

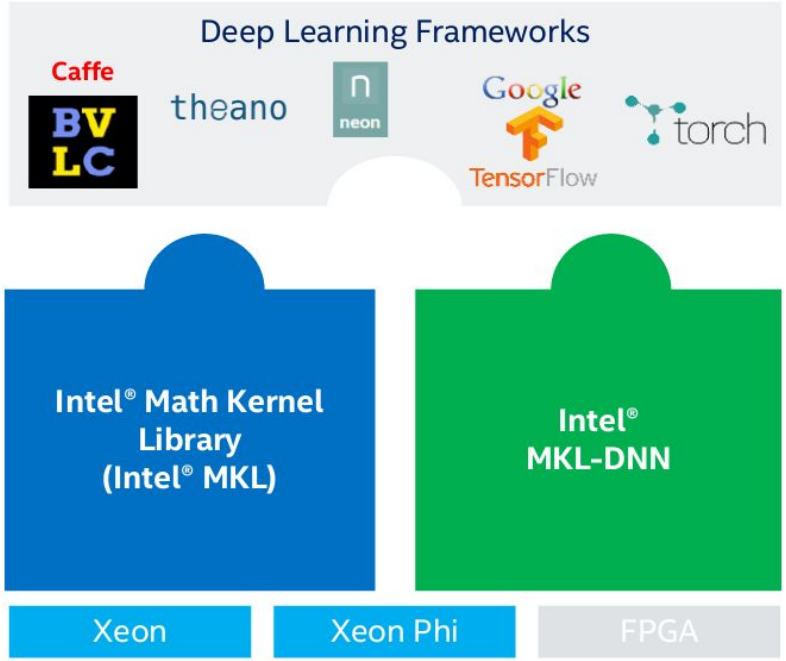
Optimized Frameworks
to simplify development

Libraries/Languages
featuring optimized building blocks

Hardware Technology
portfolio that is broad and cross-compatible



Intel® Math Kernel Library and Intel® MKL-DNN for Deep Learning Framework Optimization



Intel® MKL	Intel® MKL-DNN
DNN primitives + wide variety of other math functions	DNN primitives
C DNN APIs	C/C++ DNN APIs
Binary distribution	Open source DNN code*
Free community license. Premium support available as part of Parallel Studio XE	Apache 2.0 license
Broad usage DNN primitives; not specific to individual frameworks	Multiple variants of DNN primitives as required for framework integrations
Quarterly update releases	Rapid development ahead of Intel MKL releases

* GEMM matrix multiply building blocks are binary

CLOSING REMARKS

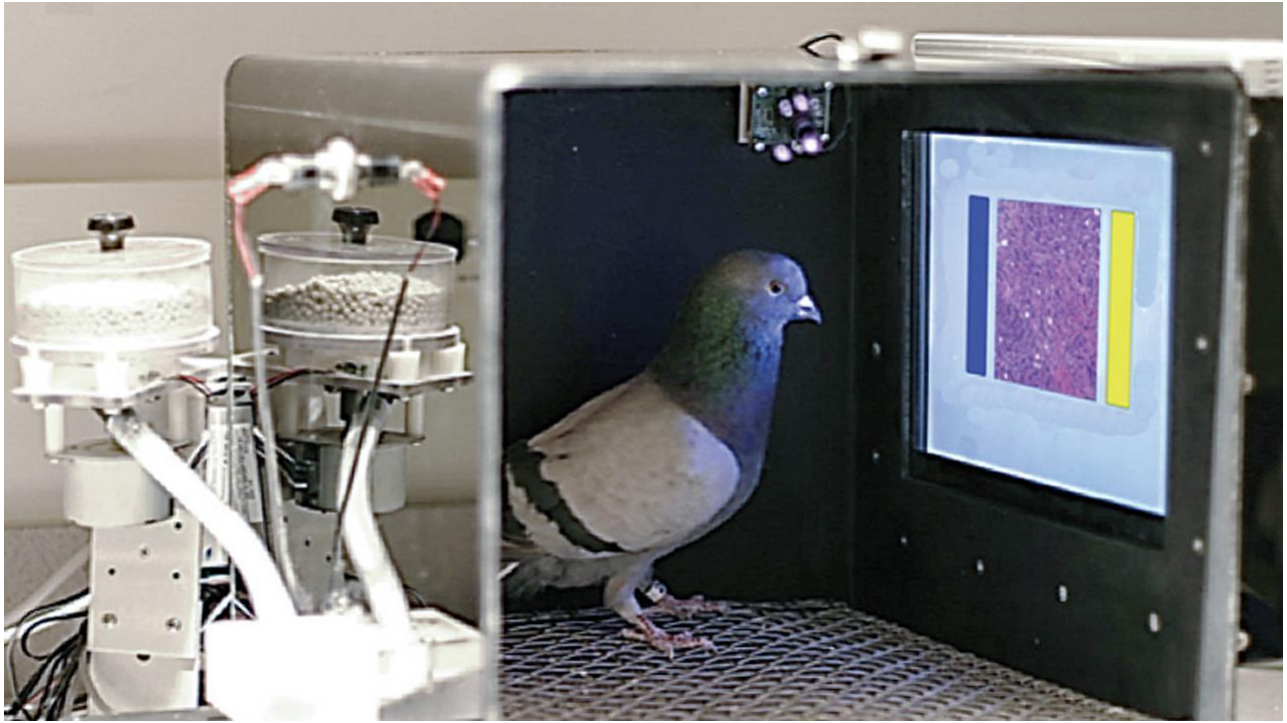
INSPIRATION: CAN I HUG THAT?

Go for it! Hug it out. With a score of **0.695779**, we're somewhat sure.

Don't hug that. Please. With a score of **0.778686**, we're pretty sure.

https://www.reddit.com/r/MachineLearning/comments/4casci/can_i_hug_that_i_trained_a_classifier_to_tell_you/

PIGEONS RECOGNIZING CANCER TISSUE



http://www.sciencemag.org/news/2015/11/pigeons-spot-cancer-well-human-experts

WHERE TO LEARN MORE?

Tutorial

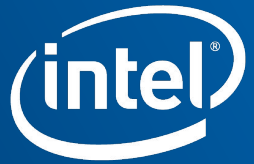
- Intel Nervana AI Academy
  https://software.intel.com/en-us/ai/academy

A course in image processing

- CS231n: Convolutional Neural Networks for Visual Recognition
  http://cs231n.github.io/

An easy introductory blog post

- Learning Deep Learning with Keras
  http://p.migdal.pl/2017/04/30/teaching-deep-learning.html



Software

STUDENT DEVELOPER PROGRAM

INTEL SUPPORT

FIND MORE KNOWLEDGE

Intel Nervana AI Academy https://software.intel.com/en-us/ai/academy

BUILD YOUR OWN PROJECT

Devmash - Find and follow world-class developers, share your own projects and collaborate with others https://devmesh.intel.com/

FIND EVEN MORE SUPPORT FOR YOUR ML IDEAS

- Interested graduate students can apply to become a student ambassadors https://software.intel.com/en-us/experts/ambassadors/apply
- Any & All interested students can join Intel-sponsored University Clubs

  https://software.intel.com/en-us/experts/ambassadors/club-sponsorship

That’s all folks, thank you!

DEEP LEARNING (VS 'SHALLOW' ML)

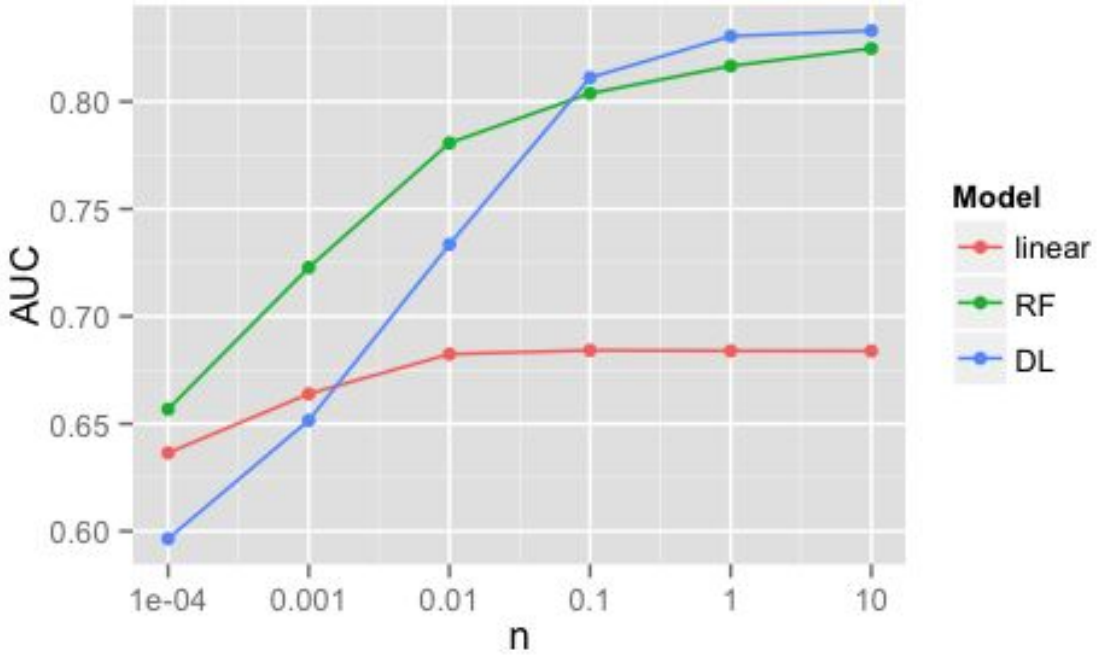
PROS

- By far the best method for image classification and segmentation
- Great progress in natural language processing
- Easy to base on a pre-trained model

CONS

- Require a lot of data
- Training takes a lot of time
- Tweaking architecture and parameters is difficult
- For many business applications tree-based techniques are better (e.g. Random Forest and XGBoost)

MORE DATA FOR MORE COMPLEX MODELS



https://github.com/szilard/benchm-ml/tree/master/x1-data-higgs
see also: https://github.com/szilard/benchm-ml

PERFORMANCE

CPU

- Suitable for small neural network models
- Check out Intel-optimized TensorFlow! (Intel Deep Learning SDK)

GPU

- Speed up networks a lot
- Expensive to buy, but convenient to rent ($0.5-2/h) at AWS

TPU?

- Architecture optimized for neural networks