Machine Learning in Engineering: Panacea or Deep Trouble ?

Kostas Plataniotis

ECE Department

www.dsp.utoronto.ca

Concordia University
Montreal, QC,
Thursday, August 9, 2018

The Edward S. Rogers Sr. Department of Electrical & Computer Engineering
UNIVERSITY OF TORONTO

FACULTY OF APPLIED SCIENCE & ENGINEERING

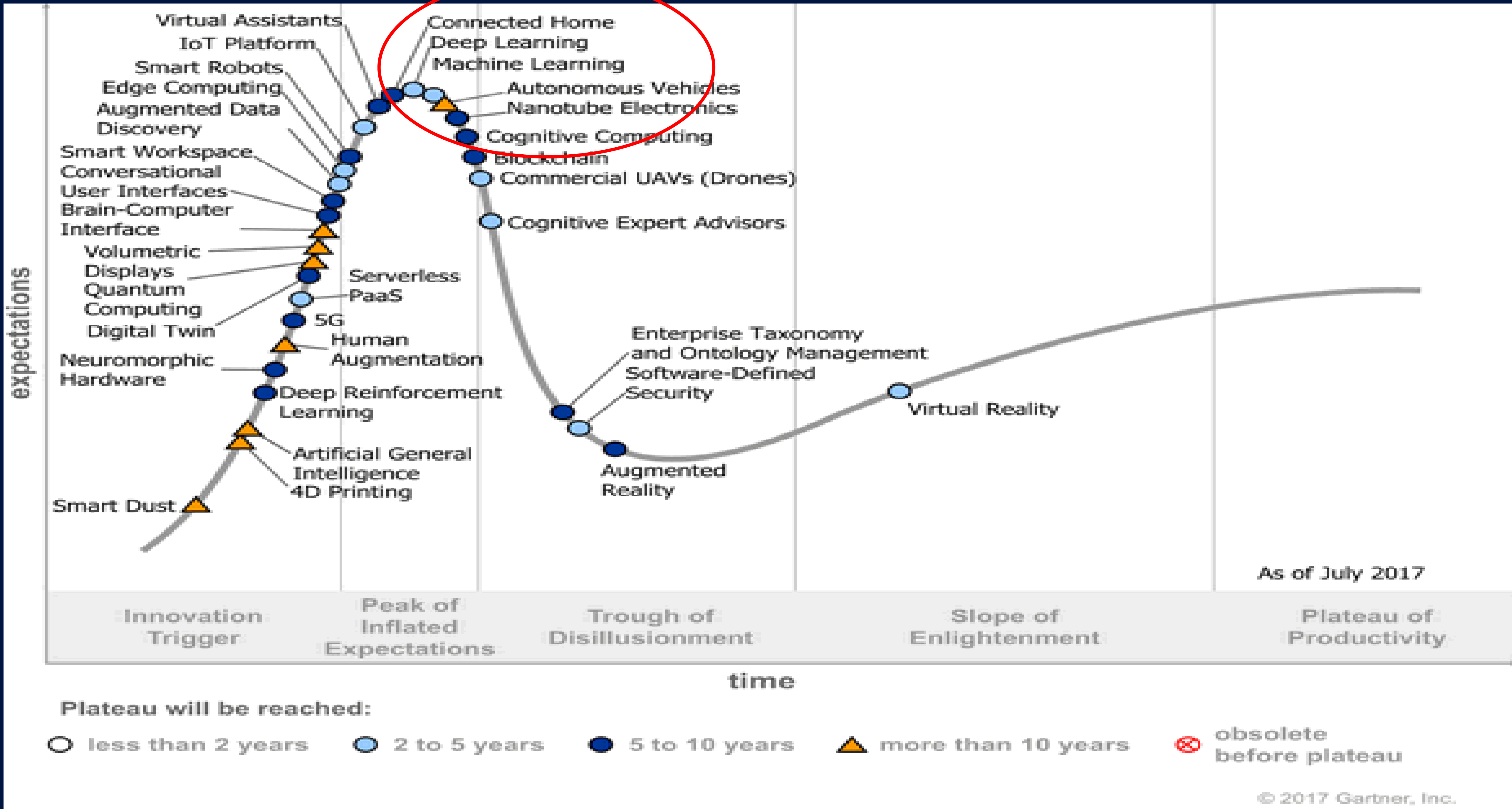
What this presentation is all about ?

A personal account of (some) key issues in the emerging field of machine learning
(relevant to our engineering practice)

The Edward S. Rogers Sr. Department of Electrical & Computer Engineering
UNIVERSITY OF TORONTO
FACULTY OF APPLIED SCIENCE & ENGINEERING

August 2018

The "hype cycle" (2017-Gartner) (in emerging technologies)



"Priority Matrix" in data science and machine learning (2017-Gartner)

benefit	years to mainstream adoption: less than 2 years	2 to 5 years	5 to 10 years	more than 10 years
transformational		Augmented Data Discovery Deep Learning Event Stream Processing Machine Learning	Algorithm Marketplaces Citizen Data Science Cognitive Computing Conversational Analytics	Artificial General Intelligence Human-in-the-Loop Crowdsourcing
high	Ensemble Learning Model Management Video/Image Analytics	AutoML Guided Analytics Predictive Analytics Self-Service Data Preparation	Graph Analytics IoT Edge Analytics Optimization Prescriptive Analytics Speech Analytics	
moderate		Notebooks Spark Text Analytics	Advanced Anomaly Detection Data Lakes Embedded Analytics Python Simulation	
low				

As of August 2017

“Priority Matrix” in emerging technologies (2017-Gartner)

benefit	years to mainstream adoption: less than 2 years	2 to 5 years	5 to 10 years	more than 10 years
transformational		Augmented Data Discovery Cognitive Expert Advisors Deep Learning Edge Computing IoT Platform Machine Learning Software-Defined Security	Blockchain Cognitive Computing Conversational User Interfaces Deep Reinforcement Learning Digital Twin Nanotube Electronics Smart Workspace Virtual Assistants	4D Printing Artificial General Intelligence Autonomous Vehicles Brain-Computer Interface Human Augmentation Smart Dust
high		Commercial UAVs (Drones)	5G Augmented Reality Connected Home Neuromorphic Hardware Smart Robots	Quantum Computing
moderate		Serverless PaaS Virtual Reality	Enterprise Taxonomy and Ontology Management	Volumetric Displays
low				

As of July 2017

The Edward S. Rogers Sr. Department of Electrical & Computer Engineering
UNIVERSITY OF TORONTO
FACULTY OF APPLIED SCIENCE & ENGINEERING

Outline

- **A definition (or two)**
- Altum Visum on deep learning networks
- Machine Learning: Myths & Realities
- Machine Learning as a process
- Explainable Artificial Intelligence
- Epilogue

The Edward S. Rogers Sr. Department of Electrical & Computer Engineering
UNIVERSITY OF TORONTO
FACULTY OF APPLIED SCIENCE & ENGINEERING

It’s all Greek to me

How we learn / know something:

- **Techné** (skill) - Knowing by doing. A carpenter learns to build by building, a potter by making pots.
- **Epistemé** (science) - Knowing by demonstration. Scientific facts are capable of being repeatedly demonstrated.
- **Nous** (intuition) - Knowing without the demonstration of invariable facts.

Nicomachean Ethics - Aristotle

It’s still Greek to me

The pertinent questions :

what are we learning and why?

The Aristotelian answer:

The goal of **episteme´** is to know truth from falsehood. The goal of **phronesis (nous)** is to know good from bad, and the goal of **techné** is to know how to express and appreciate beauty.

The Aristotelian view:

Each of these kinds of knowledge is a uniquely human capacity, thus the aim of learning is to help human beings become more fully human.

Nicomachean Ethics - Aristotle

The Edward S. Rogers Sr. Department of Electrical & Computer Engineering
UNIVERSITY OF TORONTO

FACULTY OF APPLIED SCIENCE & ENGINEERING

(Lay) Definitions – I

Learning: The activity or process of gaining knowledge or skill by studying, practicing, being taught, or experiencing something. (Merriam Webster Dictionary).

Machine: a mechanically, electrically, or electronically operated device for performing a task. Archaic : a constructed thing whether material or immaterial. (Merriam Webster Dictionary).

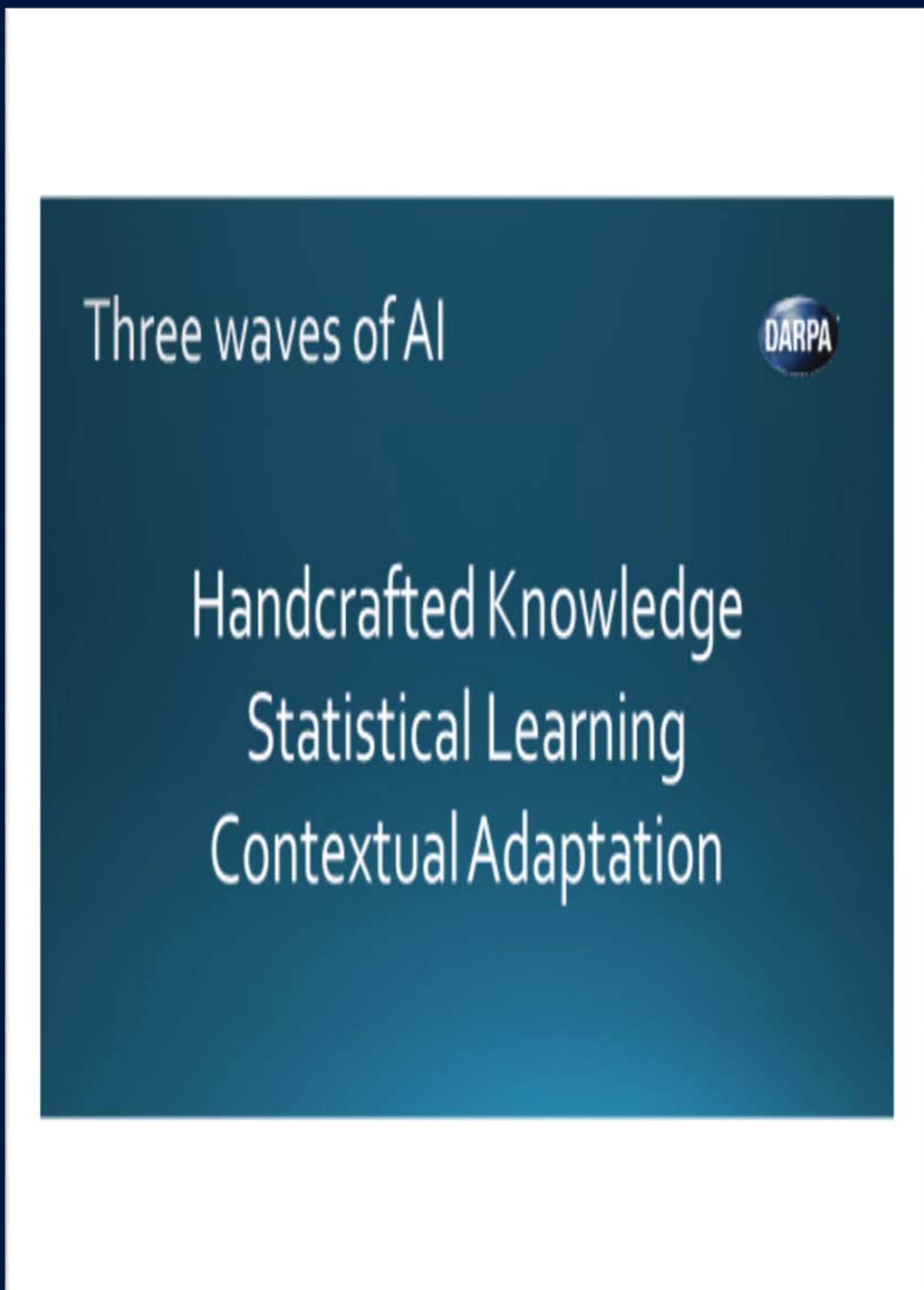
The Edward S. Rogers Sr. Department of Electrical & Computer Engineering
UNIVERSITY OF TORONTO
FACULTY OF APPLIED SCIENCE & ENGINEERING

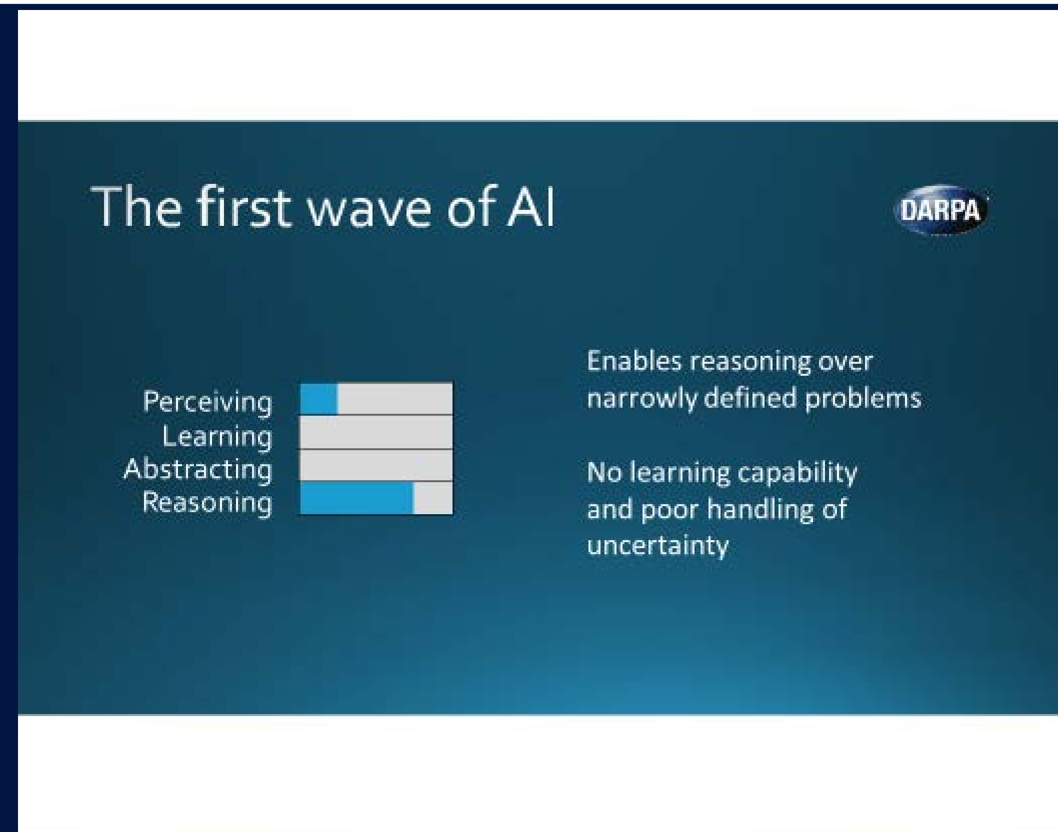
(Lay) Definitions - II

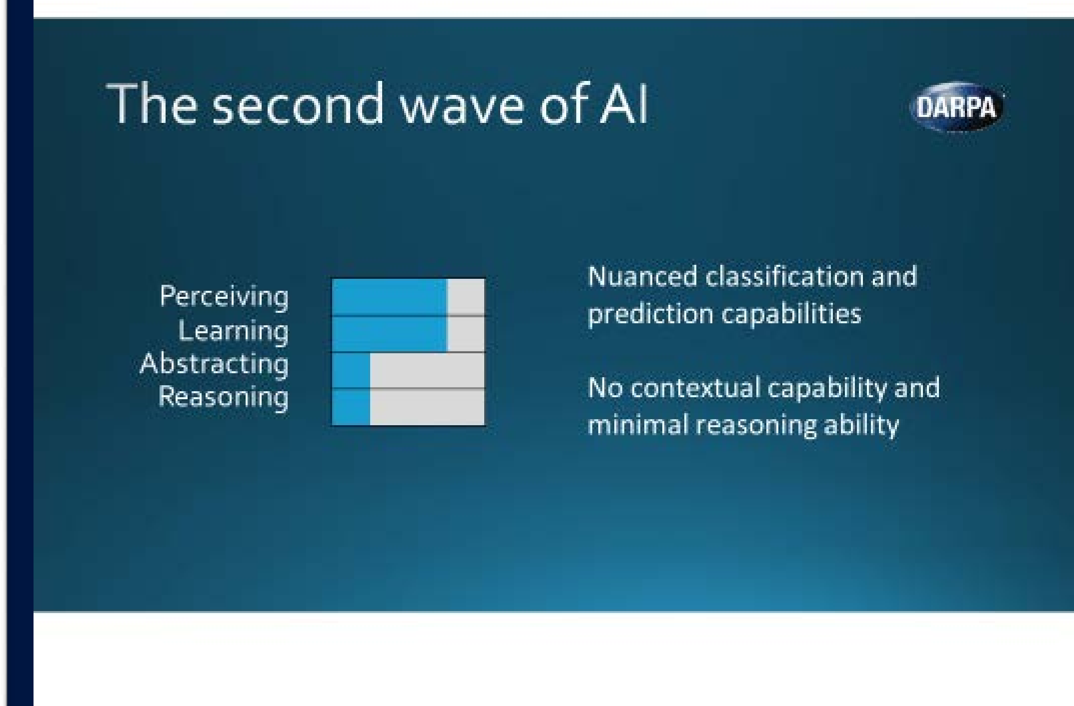
- **Artificial Intelligence (AI)**: the broader concept of machines being able to carry out tasks in a way that we would consider "smart". [1]
- Machine Learning (ML): a current application of AI based around the idea that we should really just be able to give machines access to data and let them learn for themselves. [1]

[1] Bernard Marr, What Is The Difference Between Artificial Intelligence And Machine Learning?, Forbes Magazine, accessed online, December 6, 2016.

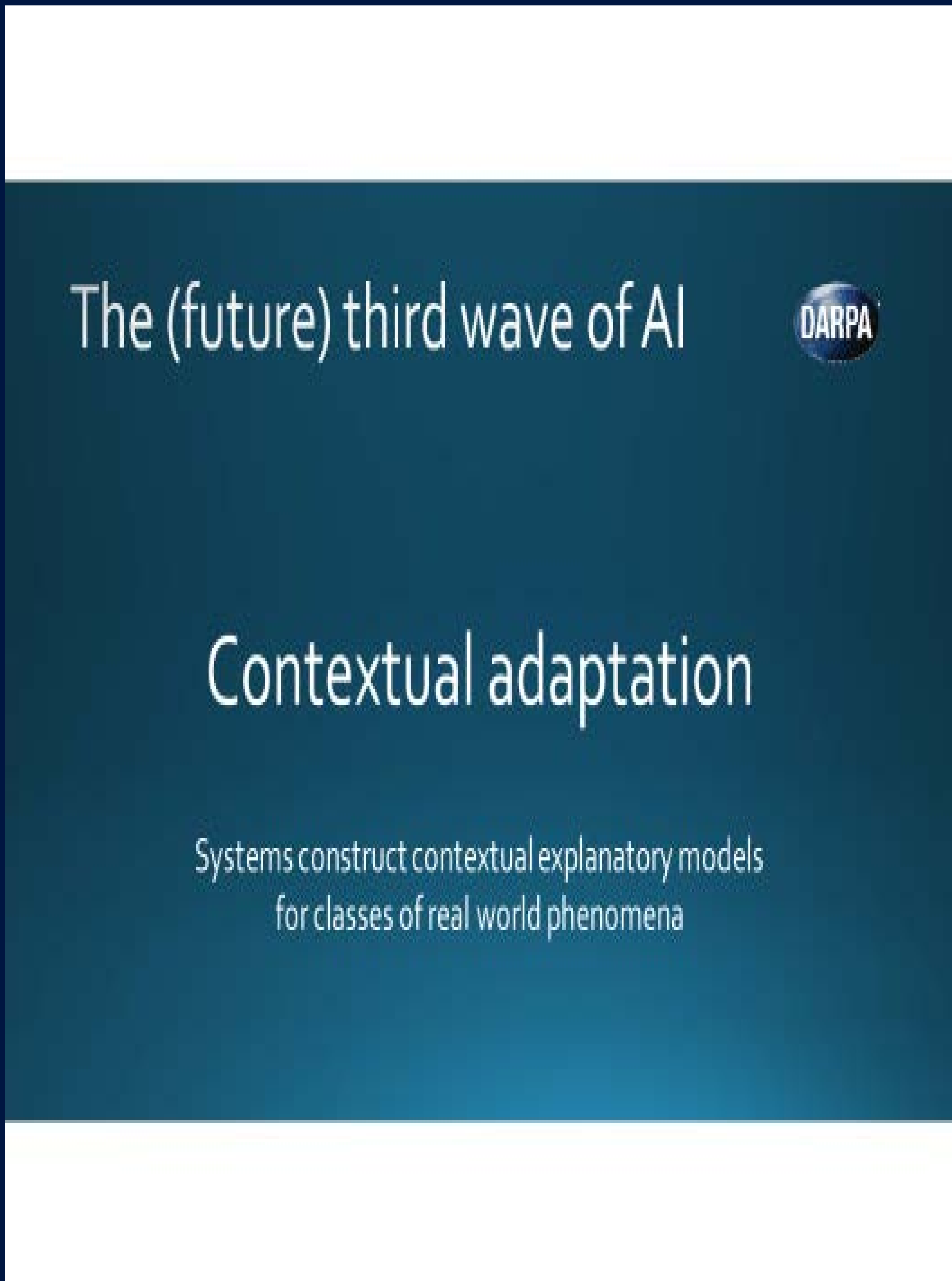
Artificial Intelligence Waves

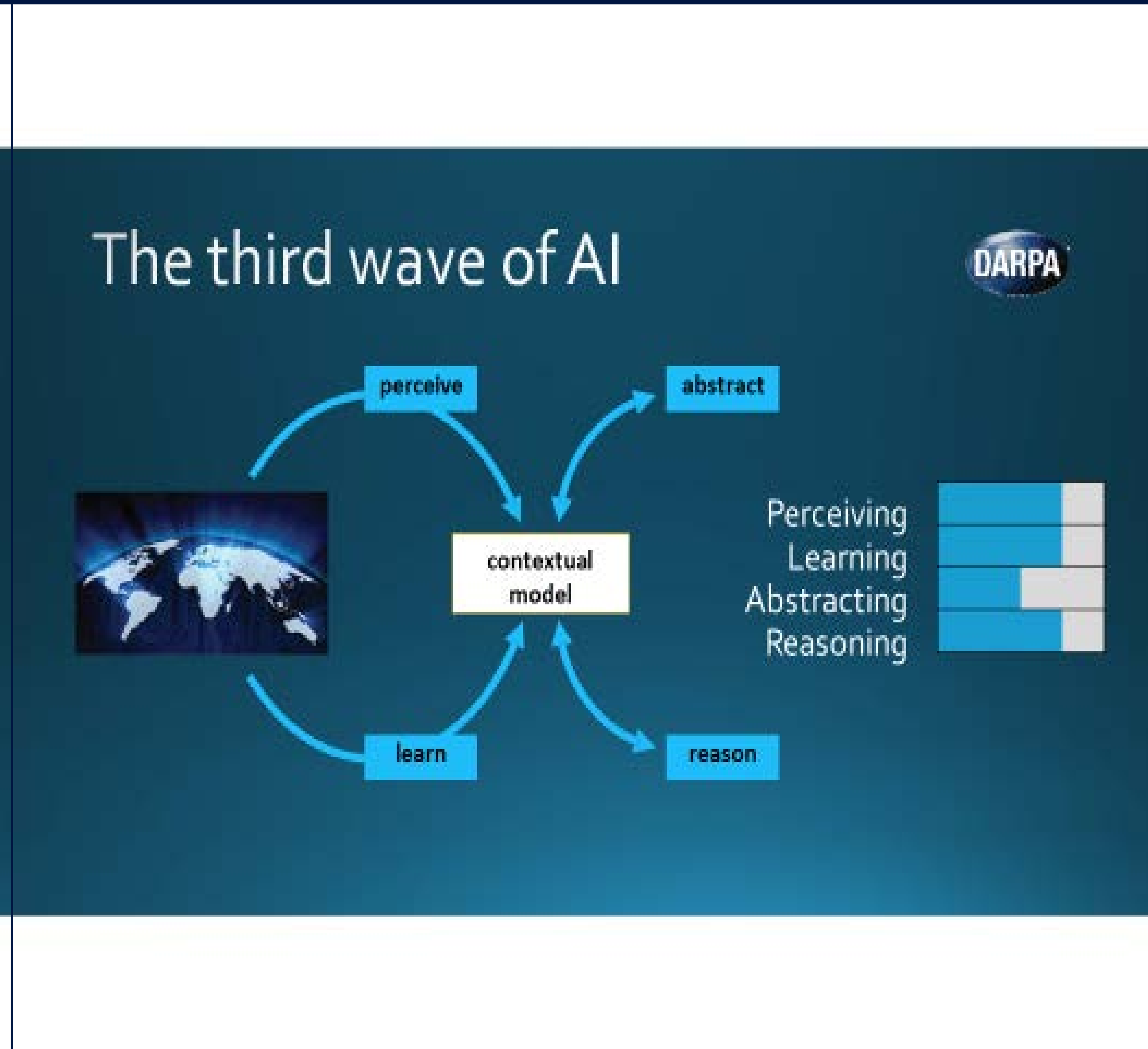






Artificial Intelligence Waves



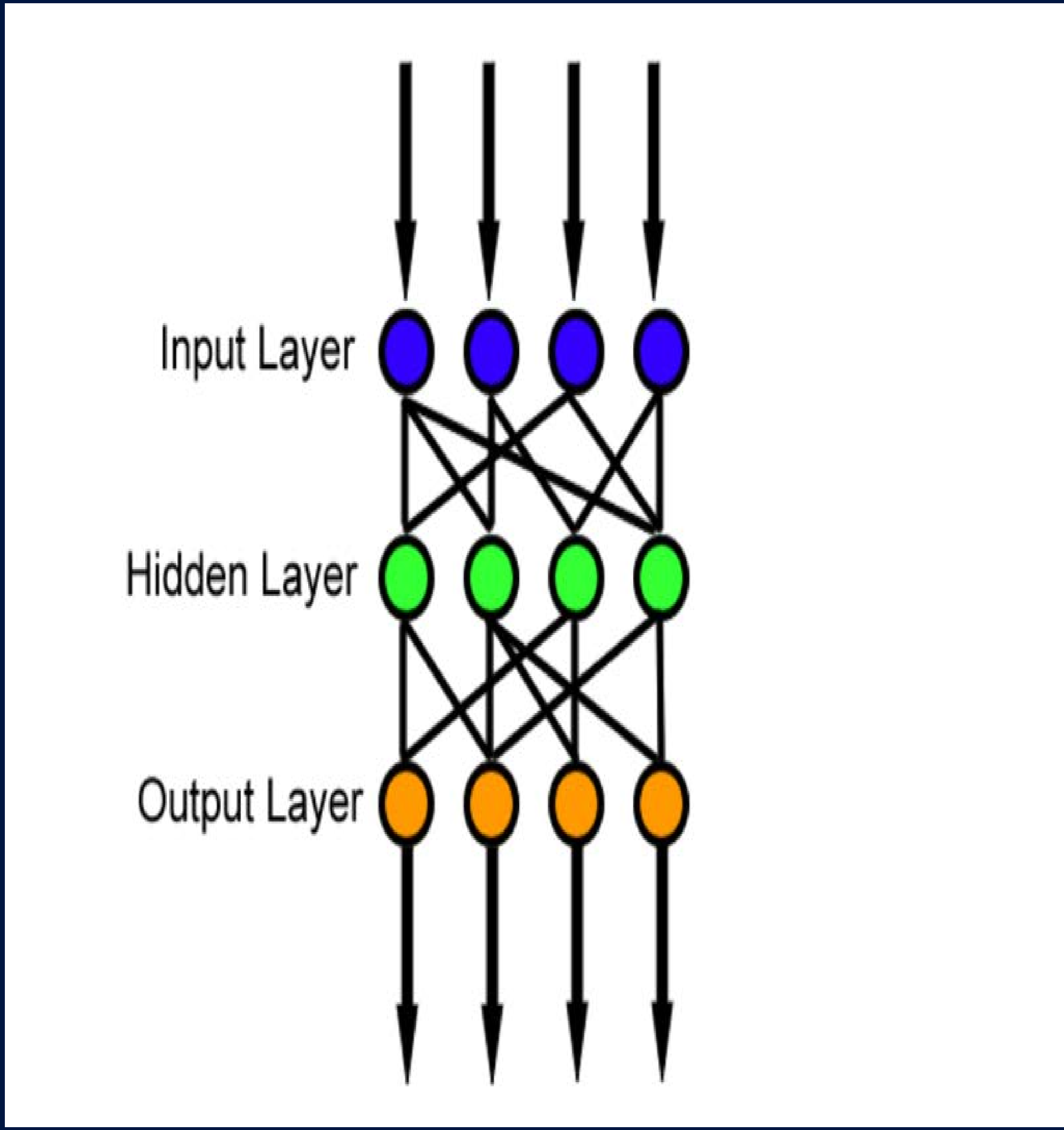


The Edward S. Rogers Sr. Department of Electrical & Computer Engineering
UNIVERSITY OF TORONTO

FACULTY OF APPLIED SCIENCE & ENGINEERING

(Lay) Definitions - III

Deep Learning (a.k.a. deep neural nets): "Deep learning is a particular kind of machine learning that achieves great power and flexibility by learning to represent the world as nested hierarchy of concepts, with each concept defined in relation to simpler concepts, and more abstract representations computed in terms of less abstract ones."



The Edward S. Rogers Sr. Department of Electrical & Computer Engineering
UNIVERSITY OF TORONTO
FACULTY OF APPLIED SCIENCE & ENGINEERING

Outline

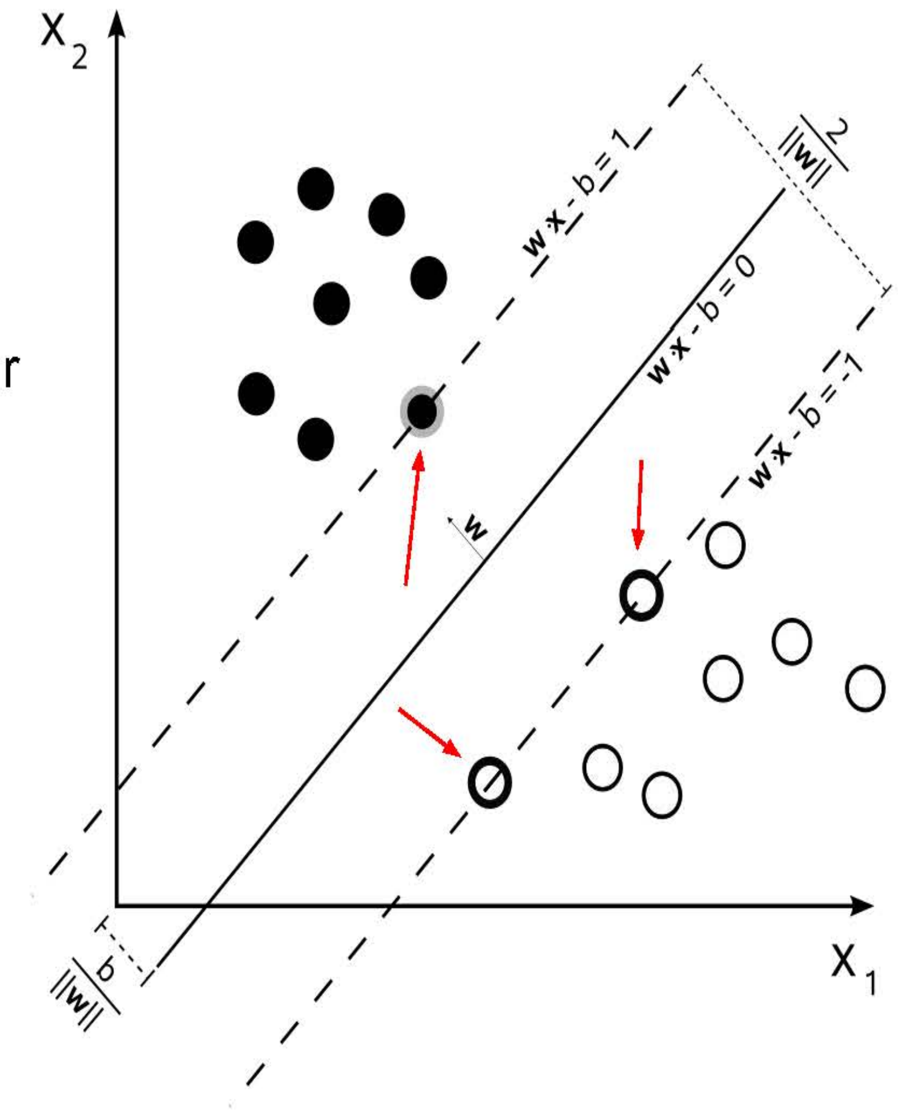
- A definition (or two)
- **Altum Visum on deep learning networks**
- Machine Learning: Myths & Realities
- Machine Learning as a process
- Explainable Artificial Intelligence
- Epilogue

The Edward S. Rogers Sr. Department
of Electrical & Computer Engineering
UNIVERSITY OF TORONTO

FACULTY
OF APPLIED
SCIENCE &
ENGINEERING

The two sides of the debate on DNN

DL will solve all of our problems!

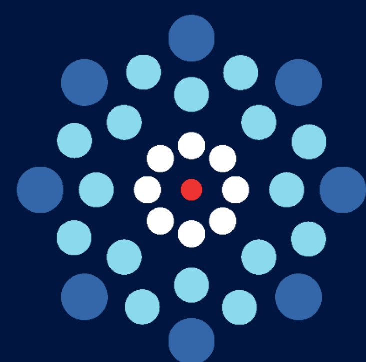


DL is all hype!

Image credit: Diego Almeida (Enlitic), 2016

The Edward S. Rogers Sr. Department of Electrical & Computer Engineering
UNIVERSITY OF TORONTO
FACULTY OF APPLIED SCIENCE & ENGINEERING

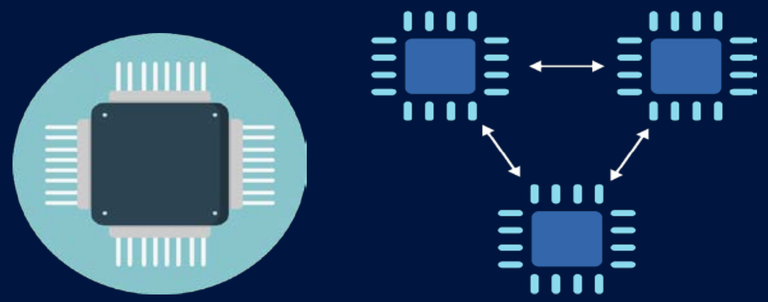
Deep Neural Networks – Where we are



Large data

Large and complex models

Frameworks & Libraries

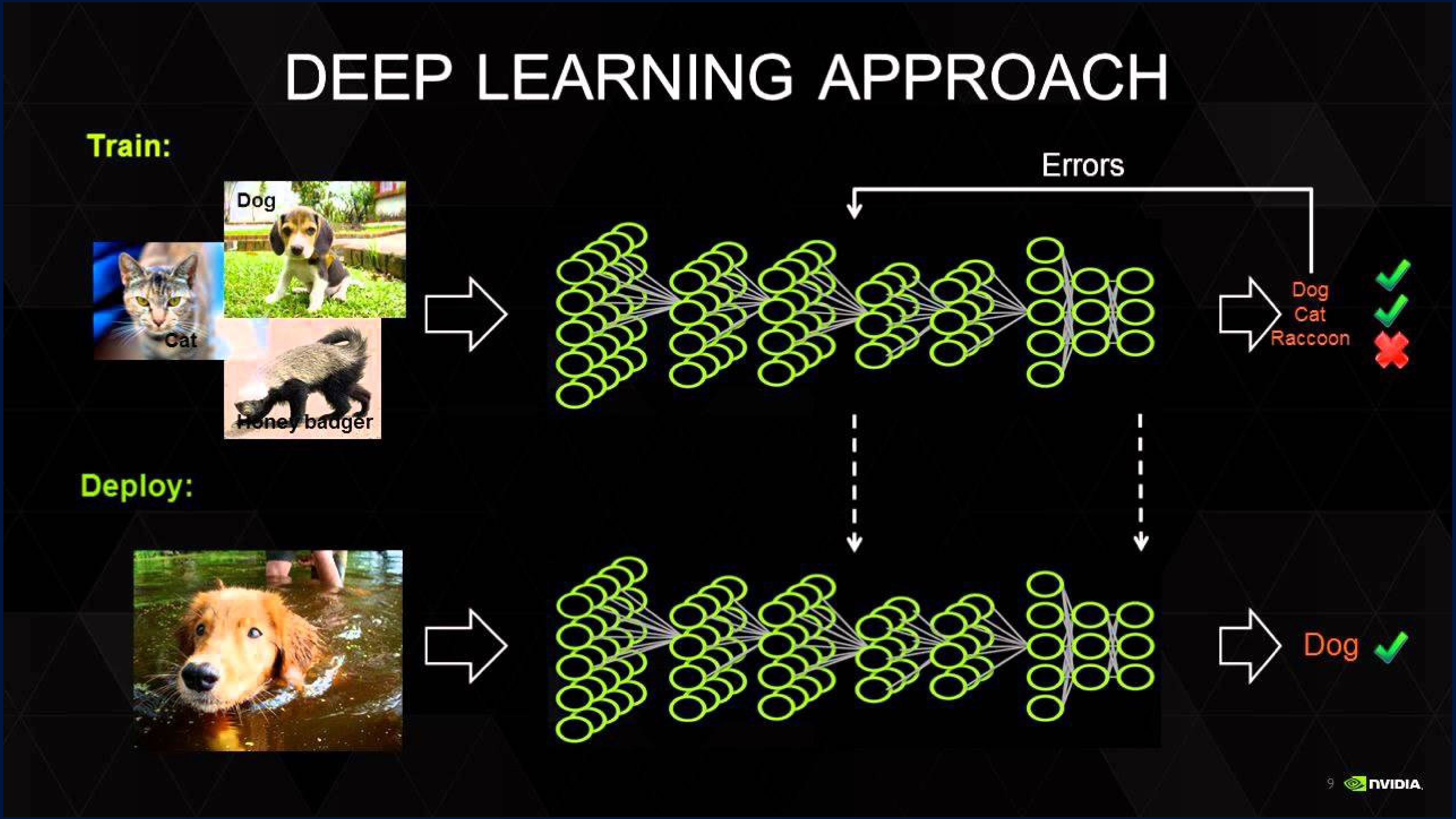


Training Hardware

The Edward S. Rogers Sr. Department of Electrical & Computer Engineering
UNIVERSITY OF TORONTO

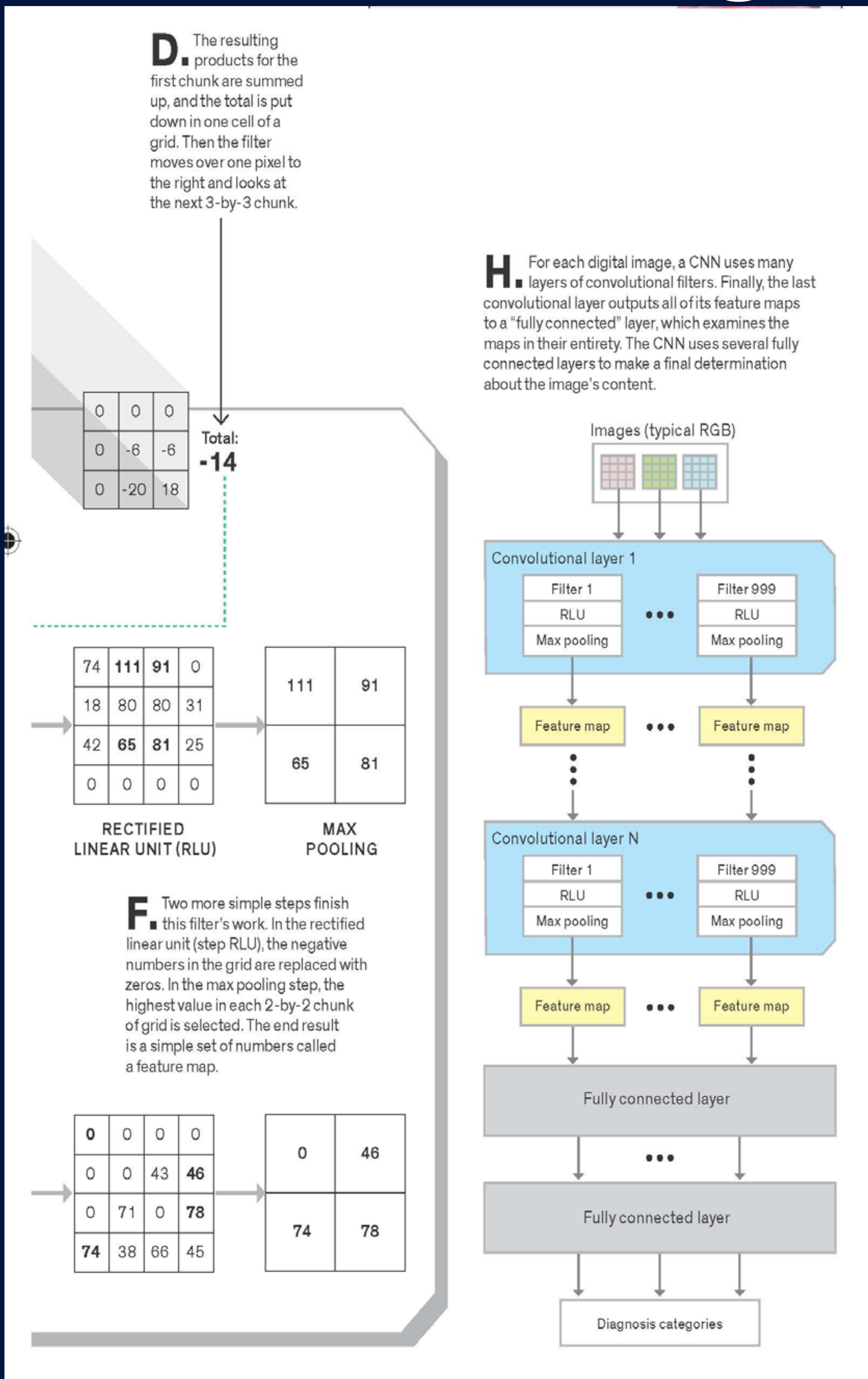
FACULTY OF APPLIED SCIENCE & ENGINEERING

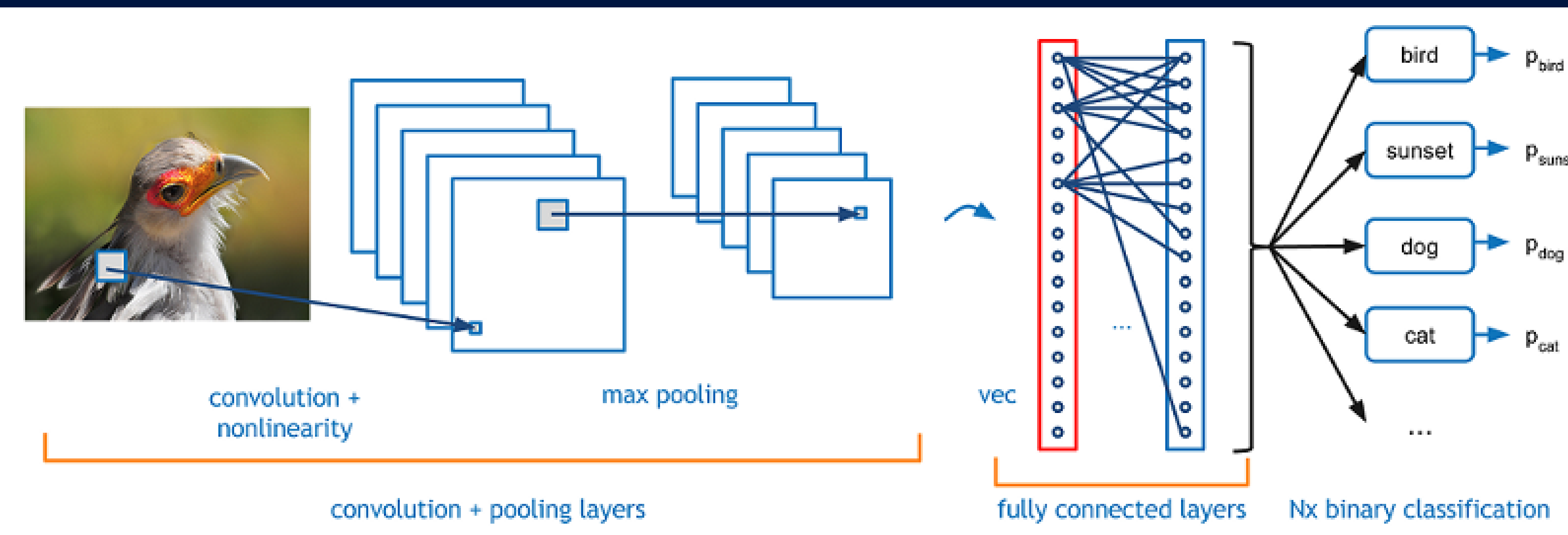
Modern Deep Neural Networks (DNN)

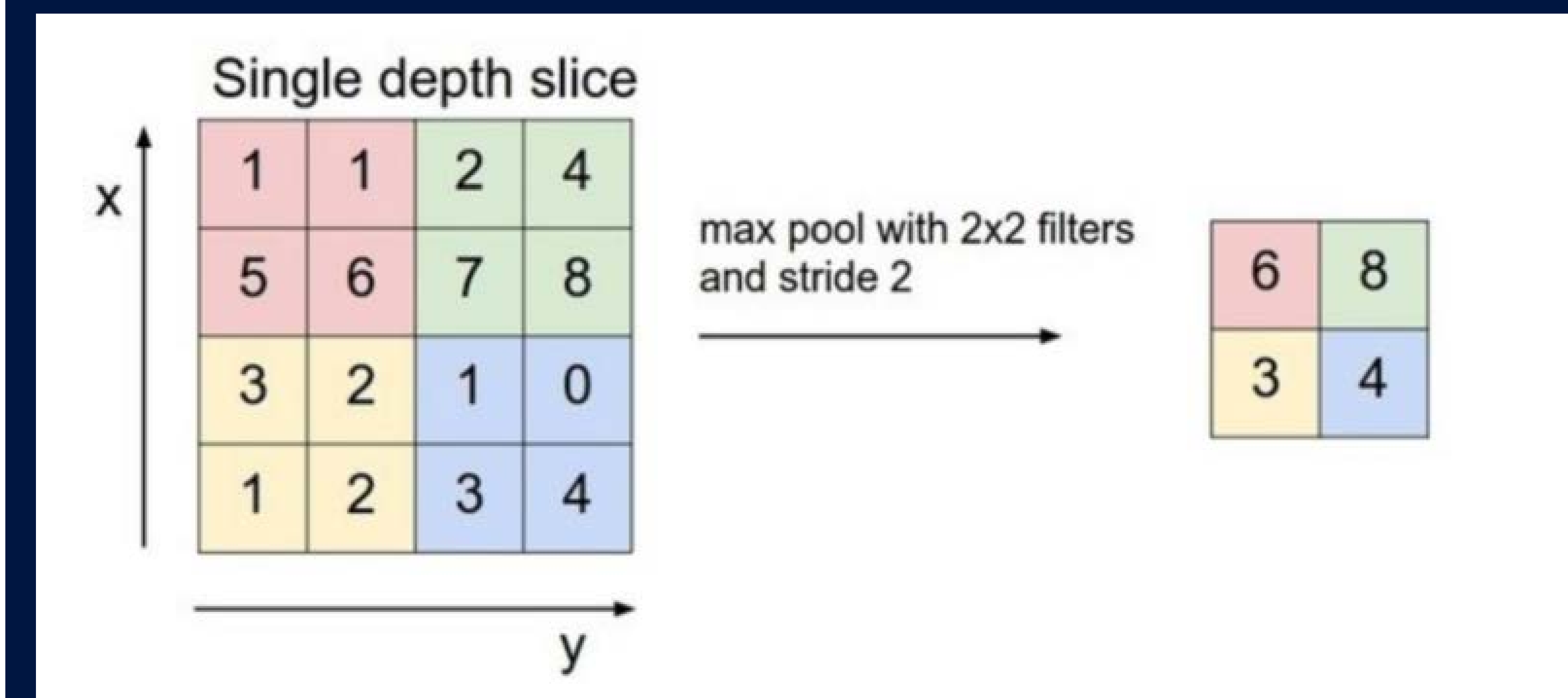


The Edward S. Rogers Sr. Department of Electrical & Computer Engineering
UNIVERSITY OF TORONTO
FACULTY OF APPLIED SCIENCE & ENGINEERING

DNN - Algorithmic Innovation







Convolutional NN (DNN) in popular blogs - I

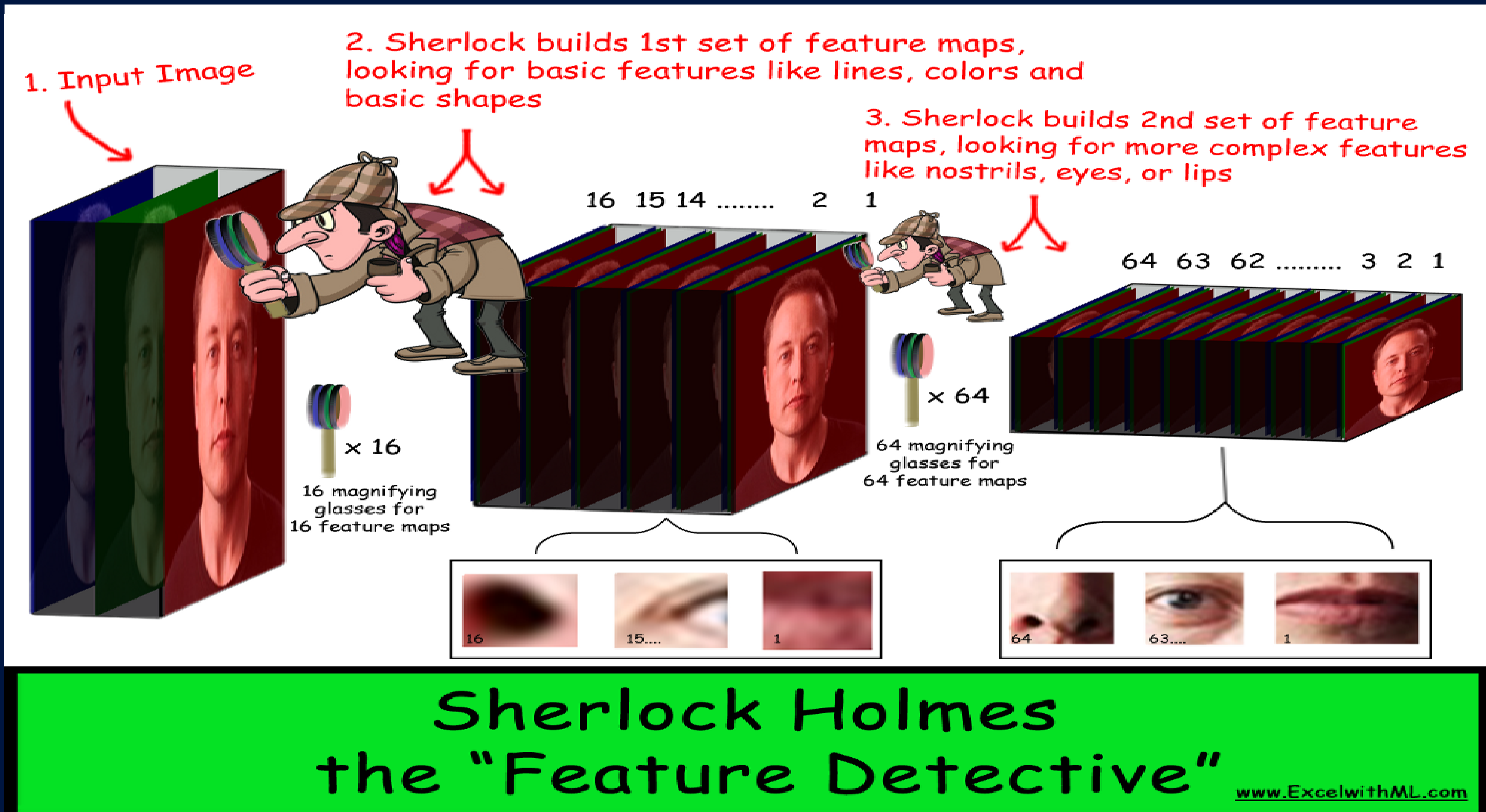


Image Credit: Dave Smith; https://towardsdatascience.com/cutting-edge-face-recognition-is-complicated-these-spreadsheets-make-it-easier-e7864dbf0e1a Accessed; August 7, 2018

The Edward S. Rogers Sr. Department of Electrical & Computer Engineering UNIVERSITY OF TORONTO

FACULTY OF APPLIED SCIENCE & ENGINEERING

Convolutional NN (DNN) in popular blogs - II

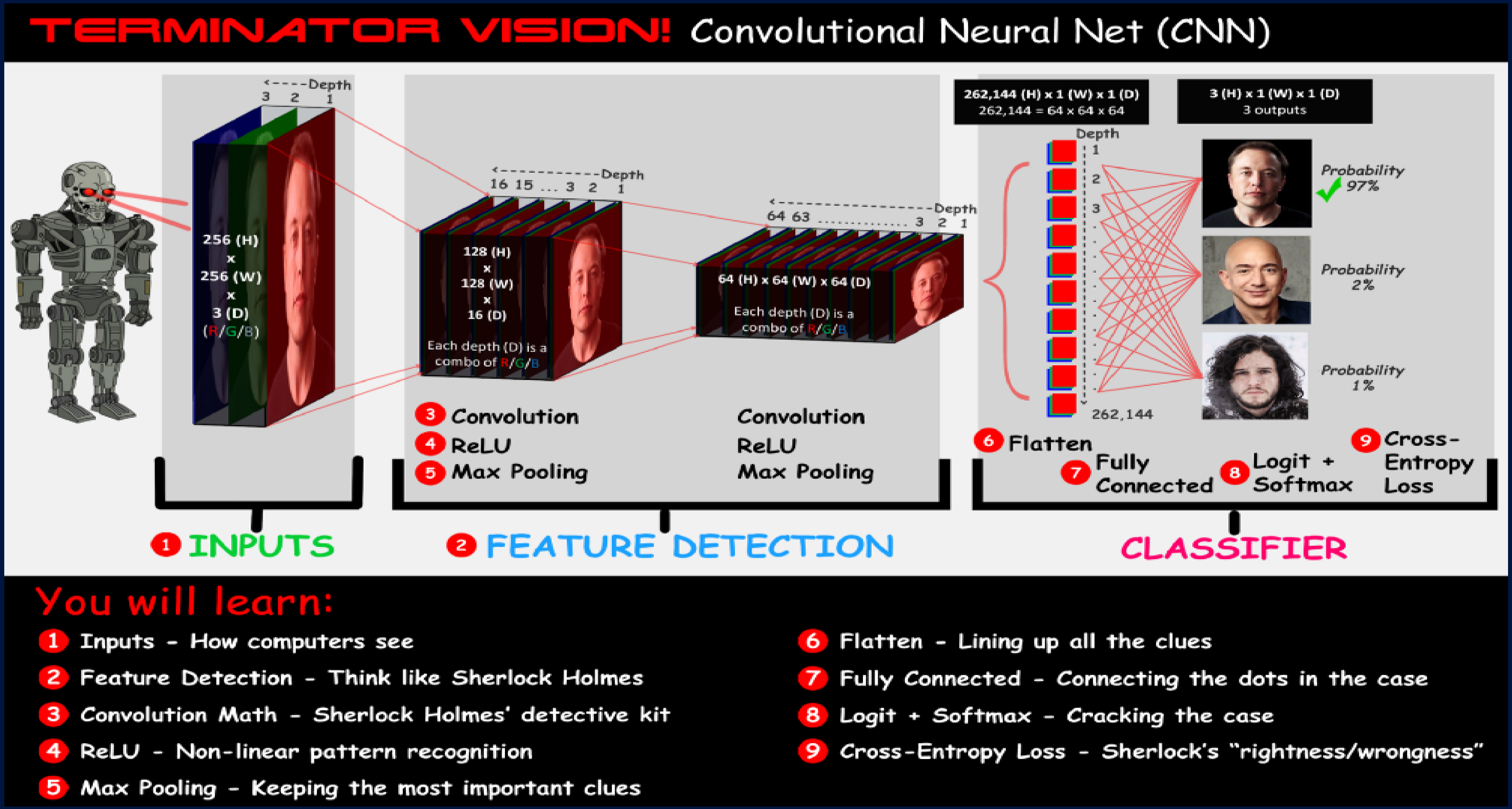
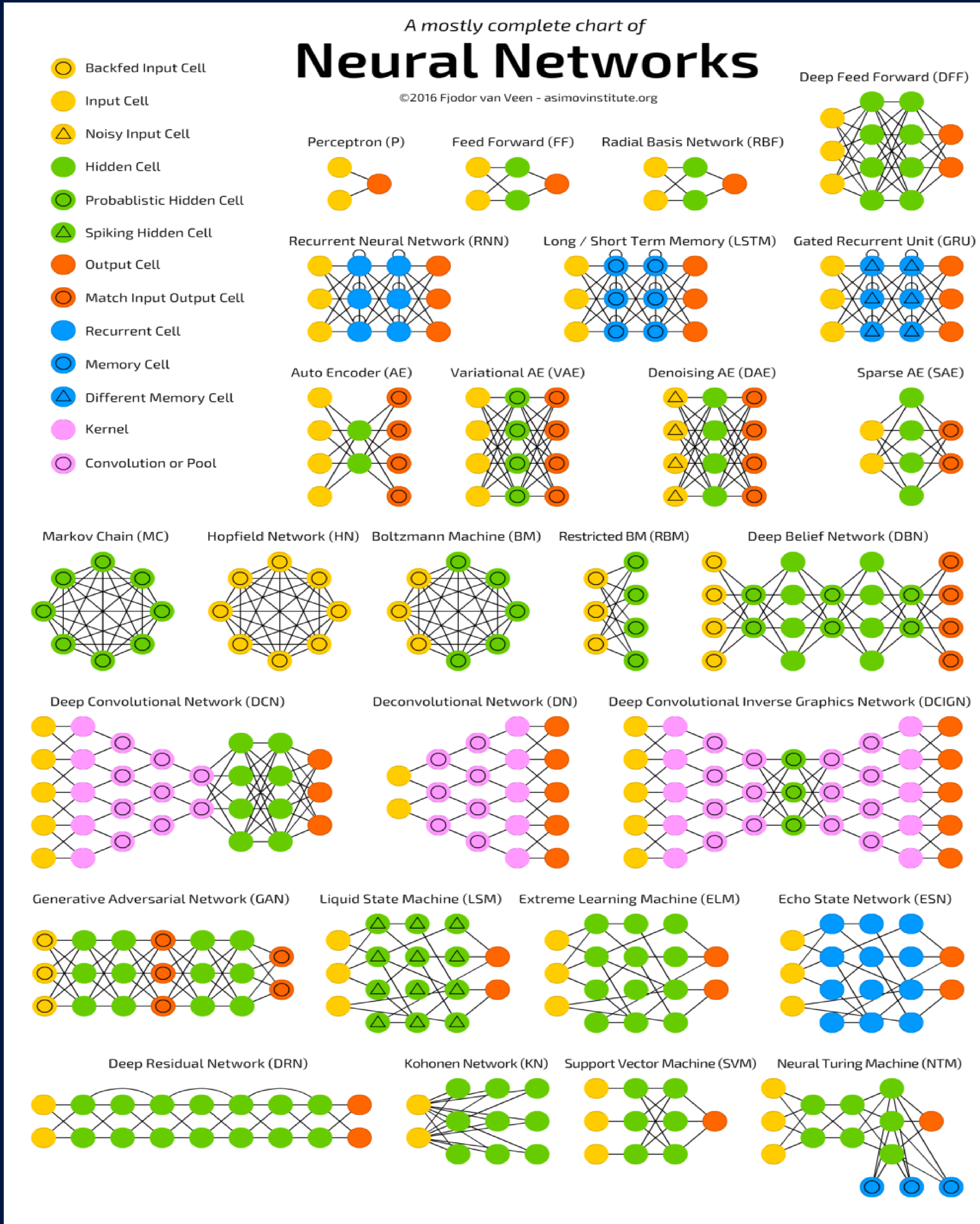


Image Credit: Dave Smith; https://towardsdatascience.com/cutting-edge-face-recognition-is-complicated-these-spreadsheets-make-it-easier-e7864dbf0e1a Accessed; August 7, 2018

The Edward S. Rogers Sr. Department of Electrical & Computer Engineering
UNIVERSITY OF TORONTO
FACULTY OF APPLIED SCIENCE & ENGINEERING



Taxonomy

Source:

http://www.asimovinstitute.org/neural-network-zoo/

Training Hardware

Deep Learning Hardware (2016)

GPUs: Nvidia is dominating

One of the first GPU neural nets was on a NVIDIA GTX 280 up to 9 layers neural network. (2010 Ciresan and Schmidhuber)

- Nvidia chips tend to outperform AMD
- More importantly, all the major frameworks use CUDA as first-class citizen. Poor support for AMD's OpenCL

NVIDIA®

Libraries – A ‘revolution’ in the making ?

Python For Data Science *Cheat Sheet*

SciPy - Linear Algebra

Learn More Python for Data Science Interactively at www.datacamp.com

SciPy

The **SciPy** library is one of the core packages for scientific computing that provides mathematical algorithms and convenience functions built on the NumPy extension of Python.

SciPy

Interacting With NumPy

Also see NumPy

```
>>> import numpy as np
>>> a = np.array([1,2,3])
>>> b = np.array([(1+5j,2j,3j), (4j,5j,6j)])
>>> c = np.array([[(1.5,2,3), (4,5,6)], [(3,2,1), (4,5,6)]])
```

Index Tricks

`>>> np.mgrid[0:5,0:5]`	Create a dense meshgrid
`>>> np.ogrid[0:2,0:2]`	Create an open meshgrid
`>>> np.r_[3,[0]*5,-1:1:10j]`	Stack arrays vertically (row-wise)
`>>> np.c_[b,c]`	Create stacked column-wise arrays

Shape Manipulation

`>>> np.transpose(b)`	Permute array dimensions
`>>> b.flatten()`	Flatten the array
`>>> np.hstack((b,c))`	Stack arrays horizontally (column-wise)
`>>> np.vstack((a,b))`	Stack arrays vertically (row-wise)
`>>> np.hsplit(c,2)`	Split the array horizontally at the 2nd index
`>>> np.vpslit(d,2)`	Split the array vertically at the 2nd index

Polynomials

`>>> from numpy import poly1d`	
`>>> p = poly1d([3,4,5])`	Create a polynomial object

Vectorizing Functions

```
>>> def myfunc(a):
        if a < 0:
          return a*2
        else:
          return a/2
>>> np.vectorize(myfunc)
```

Vectorize functions

Type Handling

`>>> np.real(b)`	Return the real part of the array elements
`>>> np.imag(b)`	Return the imaginary part of the array elements
`>>> np.real_if_close(c,tol=1000)`	Return a real array if complex parts close to 0
`>>> np.cast['f'](np.pi)`	Cast object to a data type

Other Useful Functions

`>>> np.angle(b,deg=True)`	Return the angle of the complex argument
`>>> g = np.linspace(0,np.pi,num=5)`	Create an array of evenly spaced values (number of samples)
`>>> g [3:] += np.pi`	
`>>> np.unwrap(g)`	Unwrap
`>>> np.logspace(0,10,3)`	Create an array of evenly spaced values (log scale)
`>>> np.select([c<4],[c*2])`	Return values from a list of arrays depending on conditions
`>>> misc.factorial(a)`	Factorial
`>>> misc.comb(10,3,exact=True)`	Combine N things taken at k time
`>>> misc.central_diff_weights(3)`	Weights for Np-point central derivative
`>>> misc.derivative(myfunc,1.0)`	Find the n-th derivative of a function at a point

Linear Algebra

Also see NumPy

You'll use the `linalg` and `sparse` modules. Note that `scipy.linalg` contains and expands on `numpy.linalg`.

```
>>> from scipy import linalg, sparse
```

Creating Matrices

```
>>> A = np.matrix(np.random.random((2,2)))
>>> B = np.asmatrix(b)
>>> C = np.mat(np.random.random((10,5)))
>>> D = np.mat([[3,4], [5,6]])
```

Basic Matrix Routines

Inverse	
`>>> A.I`	Inverse
`>>> linalg.inv(A)`	Inverse
Transposition	
`>>> A.T`	Tranpose matrix
`>>> A.H`	Conjugate transposition
Trace	
`>>> np.trace(A)`	Trace
Norm	
`>>> linalg.norm(A)`	Frobenius norm
`>>> linalg.norm(A,1)`	L1 norm (max column sum)
`>>> linalg.norm(A,np.inf)`	L inf norm (max row sum)
Rank	
`>>> np.linalg.matrix_rank(C)`	Matrix rank
Determinant	
`>>> linalg.det(A)`	Determinant
Solving linear problems	
`>>> linalg.solve(A,b)`	Solver for dense matrices
`>>> E = np.mat(a).T`	Solver for dense matrices
`>>> linalg.lstsq(F,E)`	Least-squares solution to linear matrix equation
Generalized inverse	
`>>> linalg.pinv(C)`	Compute the pseudo-inverse of a matrix (least-squares solver)
`>>> linalg.pinv2(C)`	Compute the pseudo-inverse of a matrix (SVD)

Creating Sparse Matrices

`>>> F = np.eye(3, k=1)`	Create a 2X2 identity matrix
`>>> G = np.mat(np.identity(2))`	Create a 2x2 identity matrix
`>>> C[C > 0.5] = 0`	
`>>> H = sparse.csr_matrix(C)`	Compressed Sparse Row matrix
`>>> I = sparse.csc_matrix(D)`	Compressed Sparse Column matrix
`>>> J = sparse.dok_matrix(A)`	Dictionary Of Keys matrix
`>>> E.todense()`	Sparse matrix to full matrix
`>>> sparse.isspmatrix_csc(A)`	Identify sparse matrix

Sparse Matrix Routines

Inverse	
`>>> sparse.linalg.inv(I)`	Inverse
Norm	
`>>> sparse.linalg.norm(I)`	Norm
Solving linear problems	
`>>> sparse.linalg.spsolve(H,I)`	Solver for sparse matrices

Sparse Matrix Functions

`>>> sparse.linalg.expm(I)`	Sparse matrix exponential

Matrix Functions

Addition	
`>>> np.add(A,D)`	Addition
Subtraction	
`>>> np.subtract(A,D)`	Subtraction
Division	
`>>> np.divide(A,D)`	Division
Multiplication	
`>>> A @ D`	Multiplication operator (Python 3)
`>>> np.multiply(D,A)`	Multiplication
`>>> np.dot(A,D)`	Dot product
`>>> np.vdot(A,D)`	Vector dot product
`>>> np.inner(A,D)`	Inner product
`>>> np.outer(A,D)`	Outer product
`>>> np.tensordot(A,D)`	Tensor dot product
`>>> np.kron(A,D)`	Kronecker product
Exponential Functions	
`>>> linalg.expm(A)`	Matrix exponential
`>>> linalg.expm2(A)`	Matrix exponential (Taylor Series)
`>>> linalg.expm3(D)`	Matrix exponential (eigenvalue decomposition)
Logarithm Function	
`>>> linalg.logm(A)`	Matrix logarithm
Trigonometric Functions	
`>>> linalg.sinm(D)`	Matrix sine
`>>> linalg.cosm(D)`	Matrix cosine
`>>> linalg.tanm(A)`	Matrix tangent
Hyperbolic Trigonometric Functions	
`>>> linalg.sinhm(D)`	Hypberbolic matrix sine
`>>> linalg.coshm(D)`	Hyperbolic matrix cosine
`>>> linalg.tanhm(A)`	Hyperbolic matrix tangent
Matrix Sign Function	
`>>> np.signm(A)`	Matrix sign function
Matrix Square Root	
`>>> linalg.sqrtm(A)`	Matrix square root
Arbitrary Functions	
`>>> linalg.funm(A, lambda x: x*x)`	Evaluate matrix function

Decompositions

Eigenvalues and Eigenvectors	
`>>> la, v = linalg.eig(A)`	Solve ordinary or generalized eigenvalue problem for square matrix
`>>> l1, l2 = la`	Unpack eigenvalues
`>>> v[:,0]`	First eigenvector
`>>> v[:,1]`	Second eigenvector
`>>> linalg.eigvals(A)`	Unpack eigenvalues
Singular Value Decomposition	
`>>> U,s,Vh = linalg.svd(B)`	Singular Value Decomposition (SVD)
`>>> M,N = B.shape`	
`>>> Sig = linalg.diagsvd(s,M,N)`	Construct sigma matrix in SVD
LU Decomposition	
`>>> P,L,U = linalg.lu(C)`	LU Decomposition

Sparse Matrix Decompositions

`>>> la, v = sparse.linalg.eigs(F,1)`	Eigenvalues and eigenvectors
`>>> sparse.linalg.svds(H, 2)`	SVD

Asking For Help

```
>>> help(scipy.linalg.diagsvd)
>>> np.info(np.matrix)
```

DataCamp
Learn Python for Data Science Interactively

The Edward S. Rogers Sr. Department of Electrical & Computer Engineering
UNIVERSITY OF TORONTO

FACULTY OF APPLIED SCIENCE & ENGINEERING

Frameworks & Libraries – I

TensorFlow

Created by Google

TensorFlow is written with a Python API over a C/C++ engine

TensorFlow generates a computational graph (e.g. a series of matrix operations) and performs automatic differentiation

TensorFlow

Pros:

- Uses Python + Numpy
- Lots of interest from the community
- Highly parallel, and designed to use various backends (software, gpu, asic)
- Apache License

Cons:

- Slower than other frameworks[1]
- More features, more abstractions than torch
- Not many pretrained models yet

https://arxiv.org/pdf/1511.06435v3.pdf

Some fundamentals

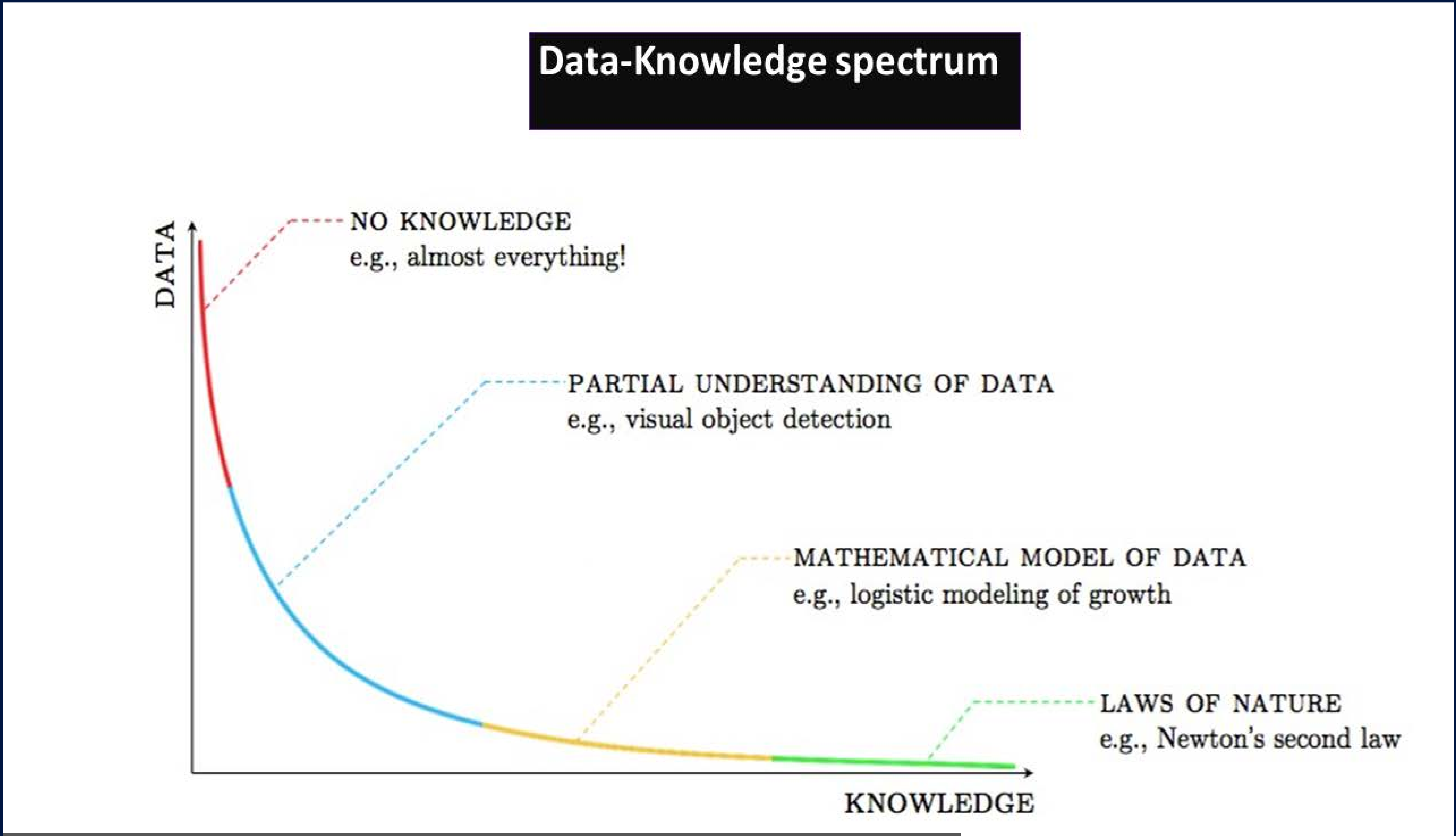


Image taken from Machine Learning Refined, by Watt – Borhani - Katsaggelos

The “Data Sets” (laboratory style)

Image credit: Xuedong Huang

The *ImageNet* Large Scale Visual Recognition Challenge (ILSVRC) evaluates algorithms for object detection and image classification at large scale

Image Credit : Ferenc Huszar

Deep Learning → Real World Problems

What matters: Real-world label distributions; Understanding black box models; Pre-training.

"Medical vision: if you want to build a system which detects lymph nodes in the human body in Computed Tomography (CT) images, you need annotated images where the lymph node is labeled. This is a rather time consuming task, as the images are in 3D and it is required to recognize very small structures. Assuming that a radiologist earns 100$/h and can carefully annotate 4 images per hour, this implies that you incur costs of 25$ per image or 250k$ for 10000 labeled images. Considering that you require several physicians to label the same image to ensure close to 100% diagnosis correctness, acquiring a dataset for the given medical task would easily exceed those 250k$." [1]

"Credit scoring: if you want to build a system that makes credit decisions, you need to know who is likely to default so you can train a machine learning system to recognize them beforehand. Unfortunately, you only know for sure if somebody defaults when it happens. Thus, a naive strategy would be to give loans of say 10k$ to everyone. However, this means that every person that defaults will cost you 10k$. This puts a very expensive price tag on each labeled data point." [1]

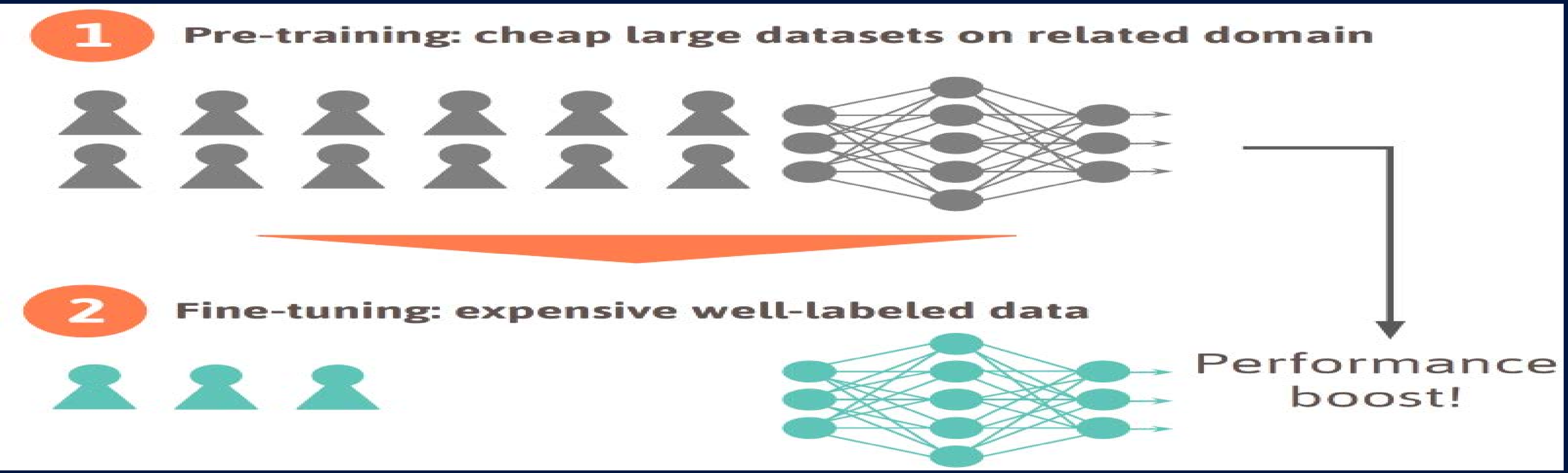
[1] Credit: Rasmus Rothe, MERANDIX

The Edward S. Rogers Sr. Department of Electrical & Computer Engineering
UNIVERSITY OF TORONTO
FACULTY OF APPLIED SCIENCE & ENGINEERING

Deep Learning → Real World Problems

What matters: Real-world label distributions; Understanding black box models; Pre -training.

" Medical Care: Researchers at the University of Pittsburg in the late 1990s evaluated machine learning algorithms for predicting mortality rates from pneumonia. The algorithms recommended that hospitals send home pneumonia patients who were also asthma sufferers, estimating their risk of death from pneumonia to be lower. It turned out that the dataset fed into the algorithms did not account for the fact that asthma sufferers had been immediately sent to intensive care, and had fared better only due to the additional attention." [1]



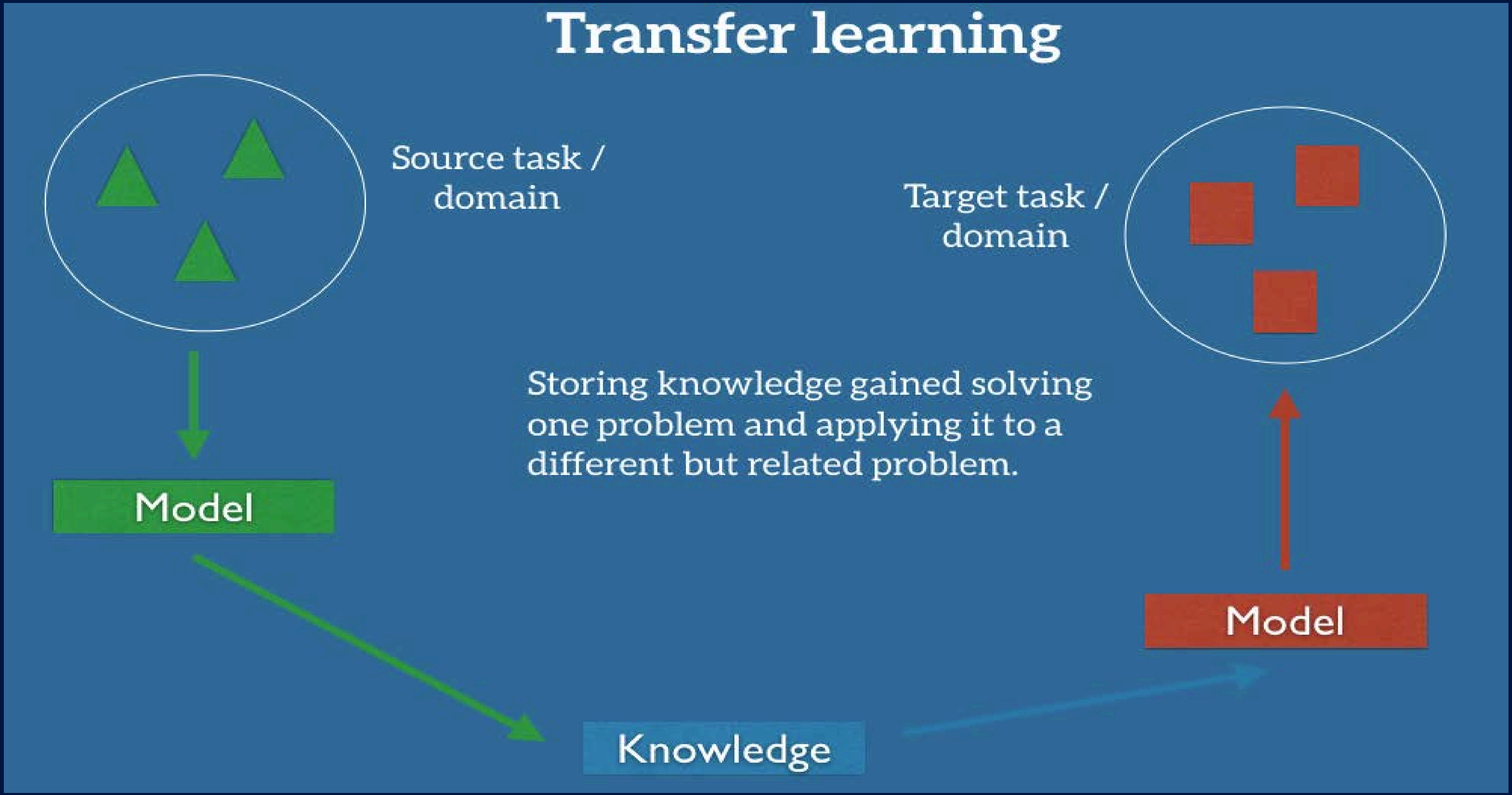
[1] Credit: Anastassia Fendyk, Harvard

Image Credit: Rasmus Rothe, MERANDIX

The Edward S. Rogers Sr. Department of Electrical & Computer Engineering
UNIVERSITY OF TORONTO

FACULTY OF APPLIED SCIENCE & ENGINEERING

Deep Learning → Real World Problems



The Edward S. Rogers Sr. Department of Electrical & Computer Engineering
UNIVERSITY OF TORONTO
FACULTY OF APPLIED SCIENCE & ENGINEERING

Neural Nets - The Achilles Heel

- Great empirical achievements (in certain application areas) were obtained with hardly any theoretical understanding of the underlying paradigm.
- The optimization employed in the learning process is highly non-convex and intractable from a theoretical viewpoint.
- Proponents offer very little interpretability of the found solution or understanding of the underlying phenomena.

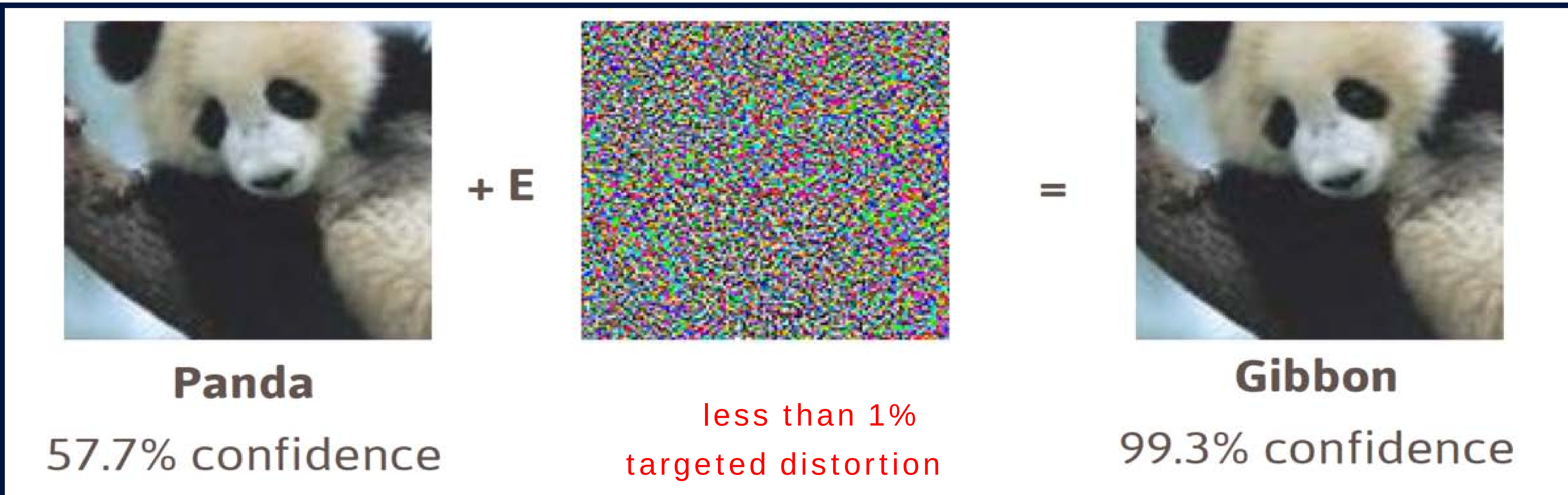
Neural Nets - Challenges

a young boy is holding a baseball bat

Statistically impressive, but individually unreliable

"Deep Visual-Semantic Alignments for Generating Image Descriptions" by Andrej Karpathy, Li Fei-Fei (CVPR 2015).

Neural Nets - Challenges



Conclusion: Inherent flaws can be exploited

Neural Nets - Challenges

Internet trolls cause the AI bot, Tay, to act offensively

Skewed training data creates maladaptation

Outline

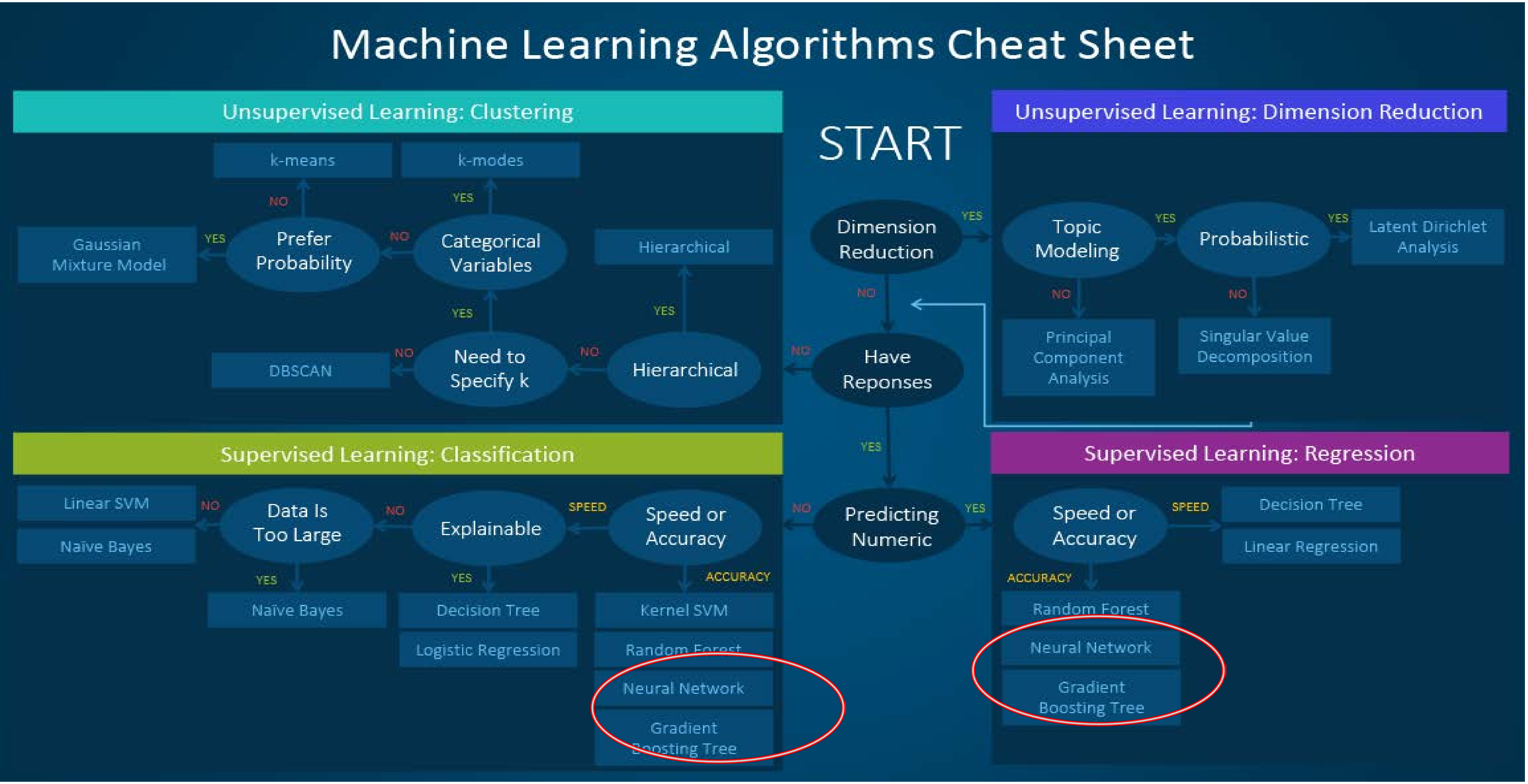
- A definition (or two)
- Altum Visum on deep learning networks
- **Machine Learning: Myths & Realities**
- Machine Learning as a process
- Explainable Artificial Intelligence
- Epilogue

Myth

Machine Learning = Deep Neural Networks

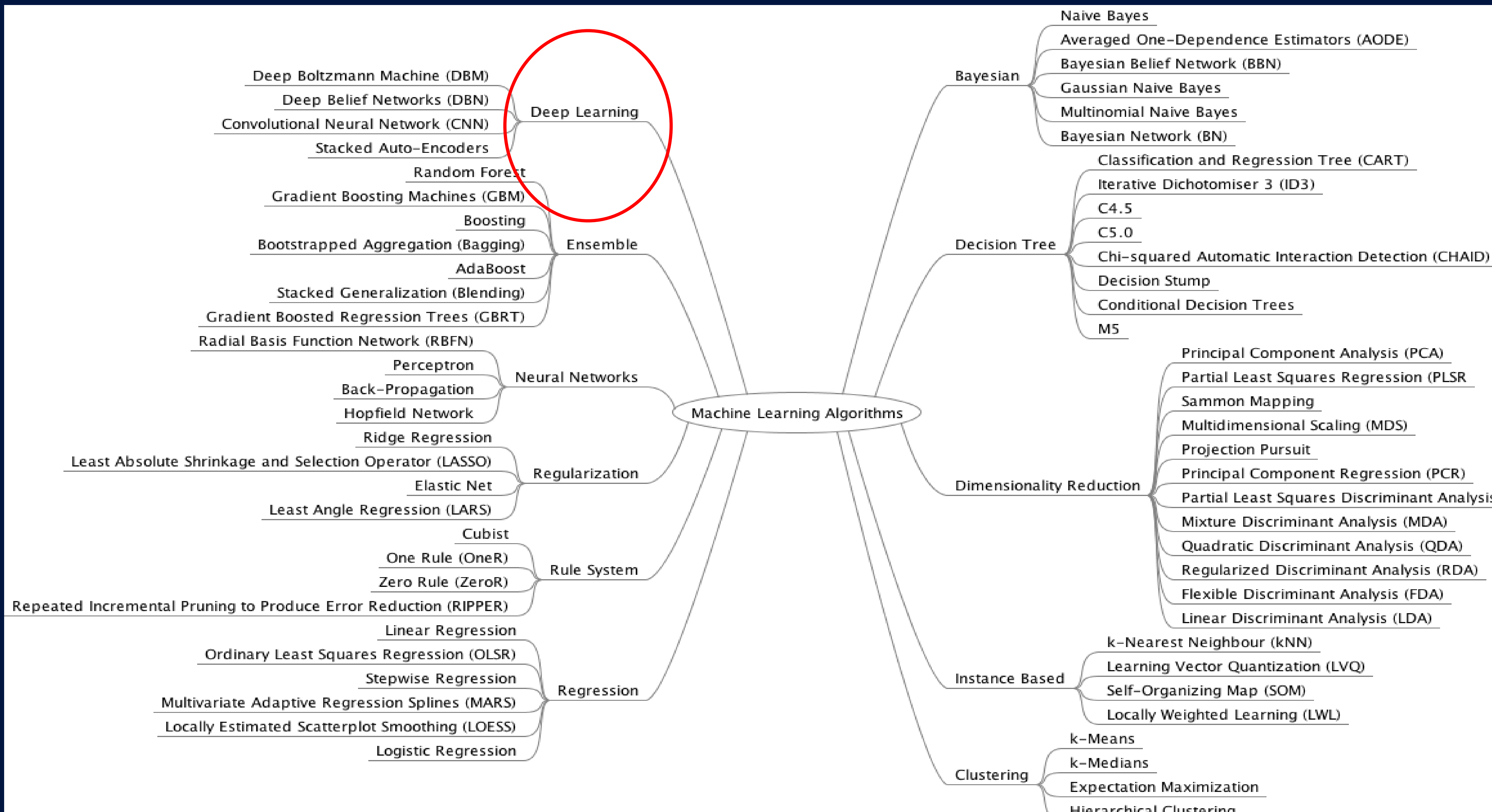
Quiz question: When the term “A.I. Winter” was invented and why ?

Reality

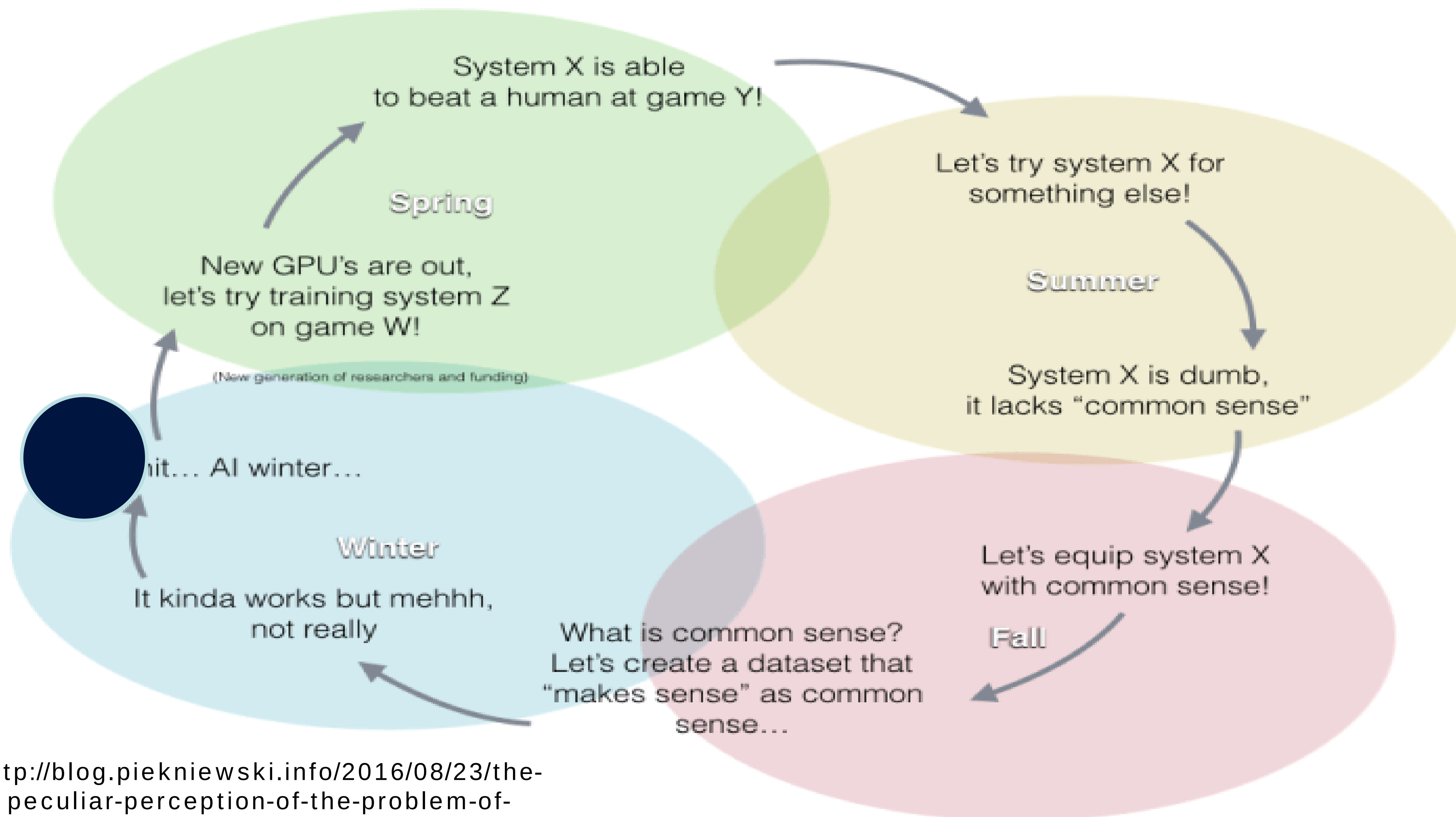


Credit: Hui Li, SAS Analytics, April 2017

Reality


Machine Learning Algorithms
Deep Learning
Deep Boltzmann Machine (DBM)
Deep Belief Networks (DBN)
Convolutional Neural Network (CNN)
Stacked Auto-Encoders
Ensemble
Random Forest
Gradient Boosting Machines (GBM)
Boosting
Bootstrapped Aggregation (Bagging)
AdaBoost
Stacked Generalization (Blending)
Gradient Boosted Regression Trees (GBRT)
Neural Networks
Radial Basis Function Network (RBFN)
Perceptron
Back-Propagation
Hopfield Network
Regularization
Ridge Regression
Least Absolute Shrinkage and Selection Operator (LASSO)
Elastic Net
Least Angle Regression (LARS)
Rule System
Cubist
One Rule (OneR)
Zero Rule (ZeroR)
Repeated Incremental Pruning to Produce Error Reduction (RIPPER)
Regression
Linear Regression
Ordinary Least Squares Regression (OLSR)
Stepwise Regression
Multivariate Adaptive Regression Splines (MARS)
Locally Estimated Scatterplot Smoothing (LOESS)
Logistic Regression
Bayesian
Naive Bayes
Averaged One-Dependence Estimators (AODE)
Bayesian Belief Network (BBN)
Gaussian Naive Bayes
Multinomial Naive Bayes
Bayesian Network (BN)
Decision Tree
Classification and Regression Tree (CART)
Iterative Dichotomiser 3 (ID3)
C4.5
C5.0
Chi-squared Automatic Interaction Detection (CHAID)
Decision Stump
Conditional Decision Trees
M5
Dimensionality Reduction
Principal Component Analysis (PCA)
Partial Least Squares Regression (PLSR
Sammon Mapping
Multidimensional Scaling (MDS)
Projection Pursuit
Principal Component Regression (PCR)
Partial Least Squares Discriminant Analysis
Mixture Discriminant Analysis (MDA)
Quadratic Discriminant Analysis (QDA)
Regularized Discriminant Analysis (RDA)
Flexible Discriminant Analysis (FDA)
Linear Discriminant Analysis (LDA)
Instance Based
k-Nearest Neighbour (kNN)
Learning Vector Quantization (LVQ)
Self-Organizing Map (SOM)
Locally Weighted Learning (LWL)
Clustering
k-Means
k-Medians
Expectation Maximization
Hierarchical Clustering

Reality or Myth: The four Seasons



http://blog.piekniewski.info/2016/08/23/the-peculiar-perception-of-the-problem-of-perception/

Consider

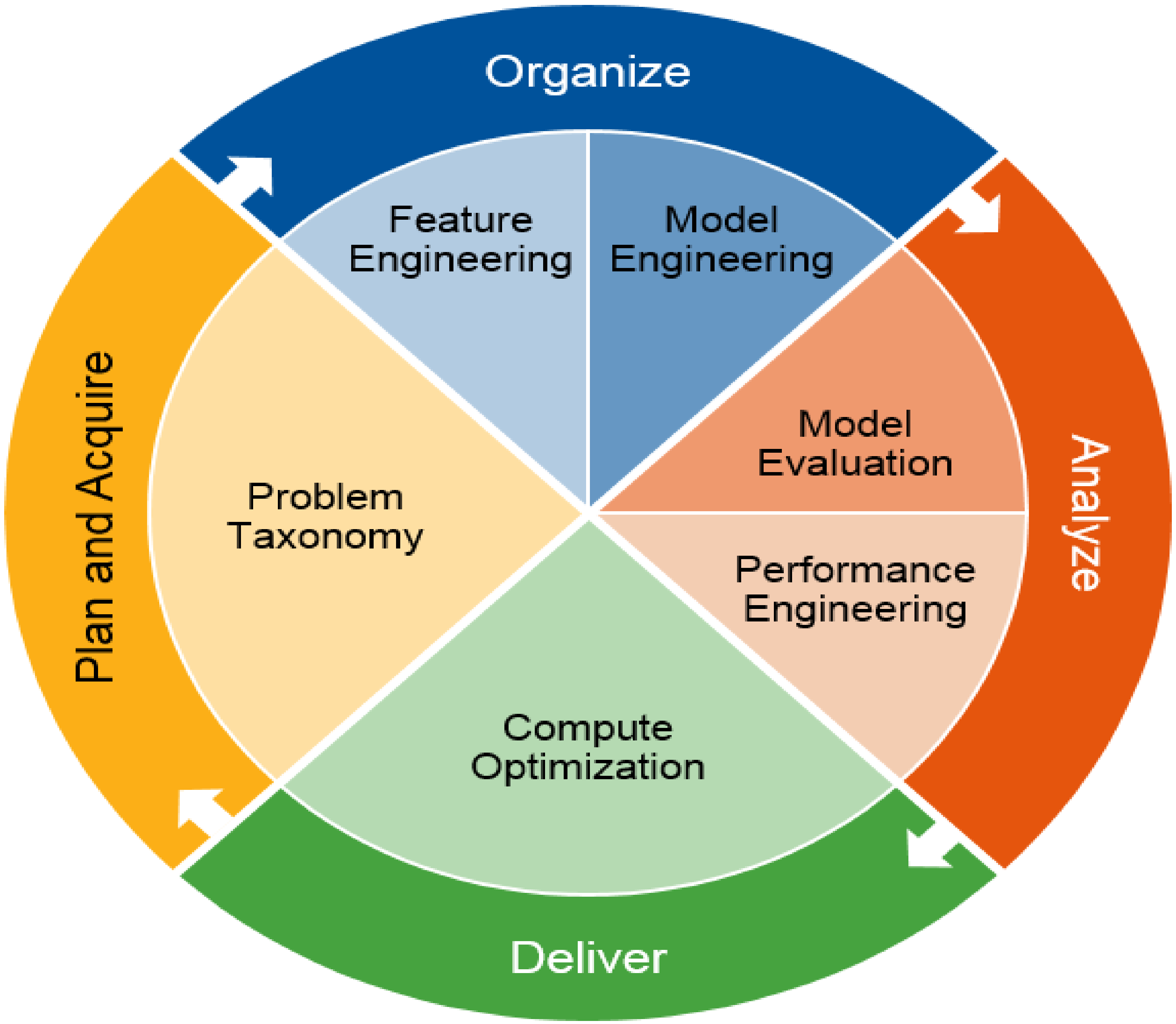
"If your only tool is a hammer, then all of the problems look like nails".

Abraham H. Maslow (1962) **via** S. (Pas) Pasupathy (1999).

Outline

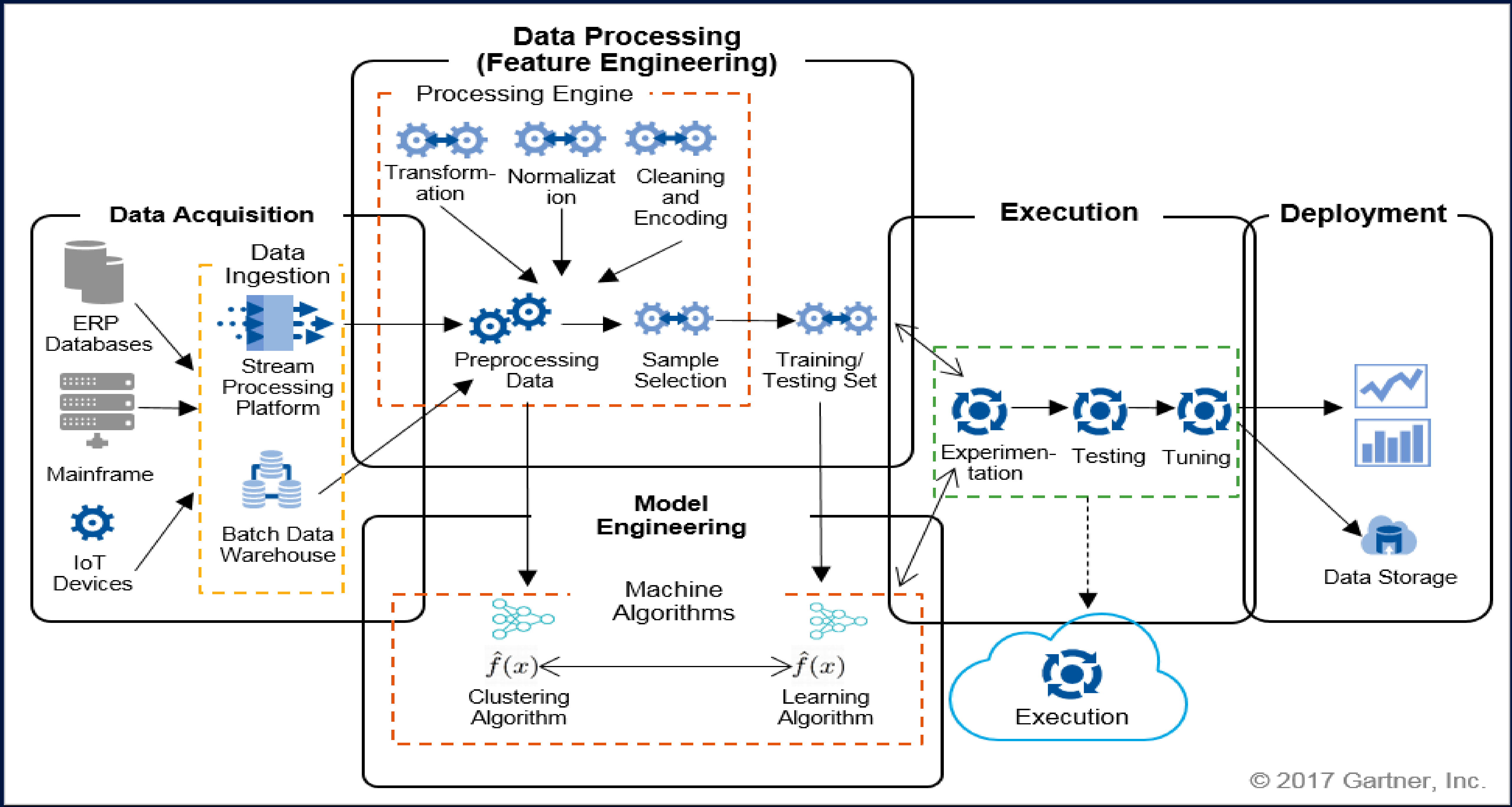
- A definition (or two)
- Altum Visum on deep learning networks
- Machine Learning: Myths & Realities
- **Machine Learning as a process**
- Explainable Artificial Intelligence
- Epilogue

Engineering Cycle of Machine Learning

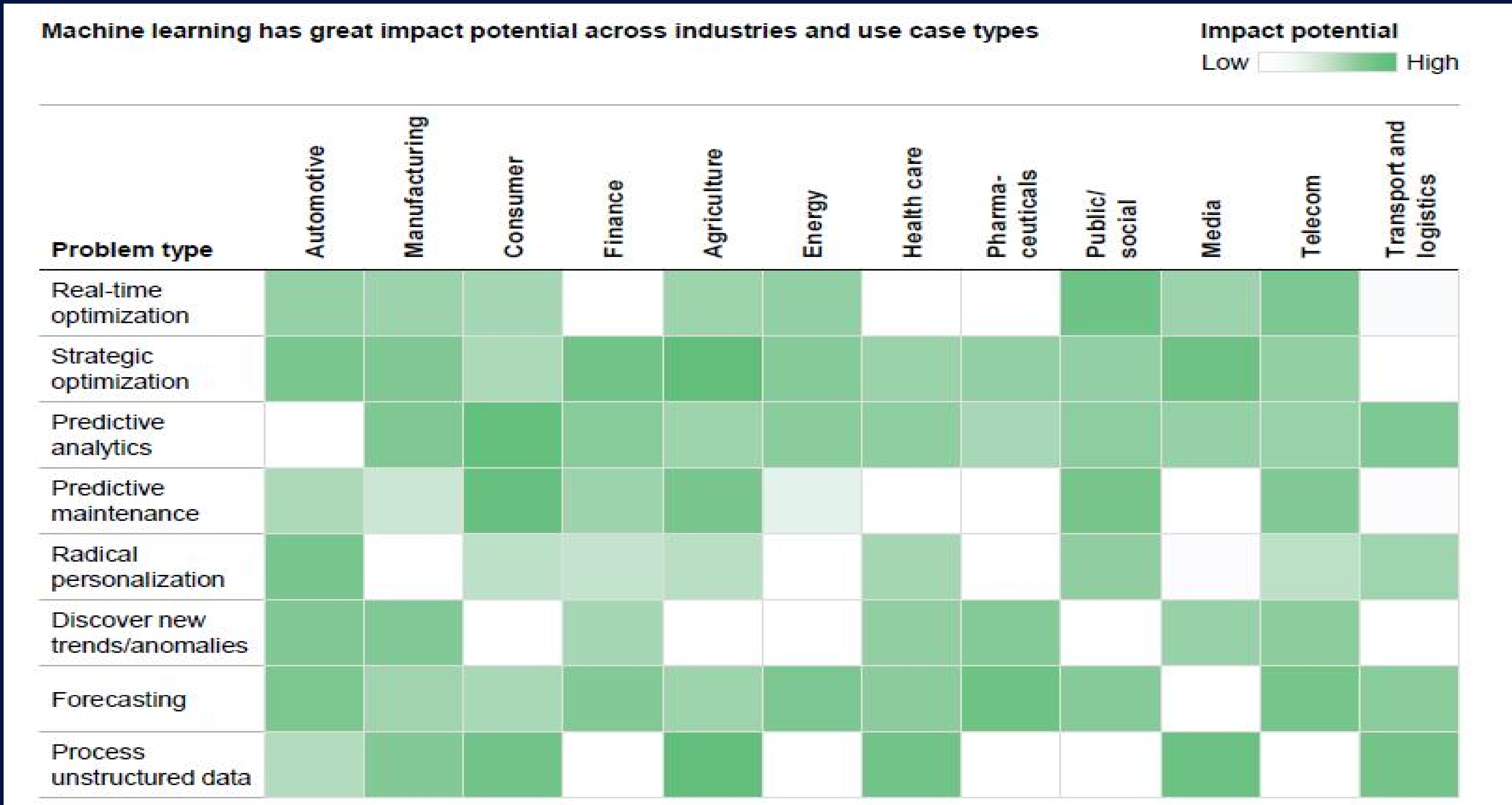


The Edward S. Rogers Sr. Department of Electrical & Computer Engineering
UNIVERSITY OF TORONTO
FACULTY OF APPLIED SCIENCE & ENGINEERING

Machine Learning: Engineering Architecture



What is the expected impact & where



SOURCE: McKinsey Global Institute analysis

Outline

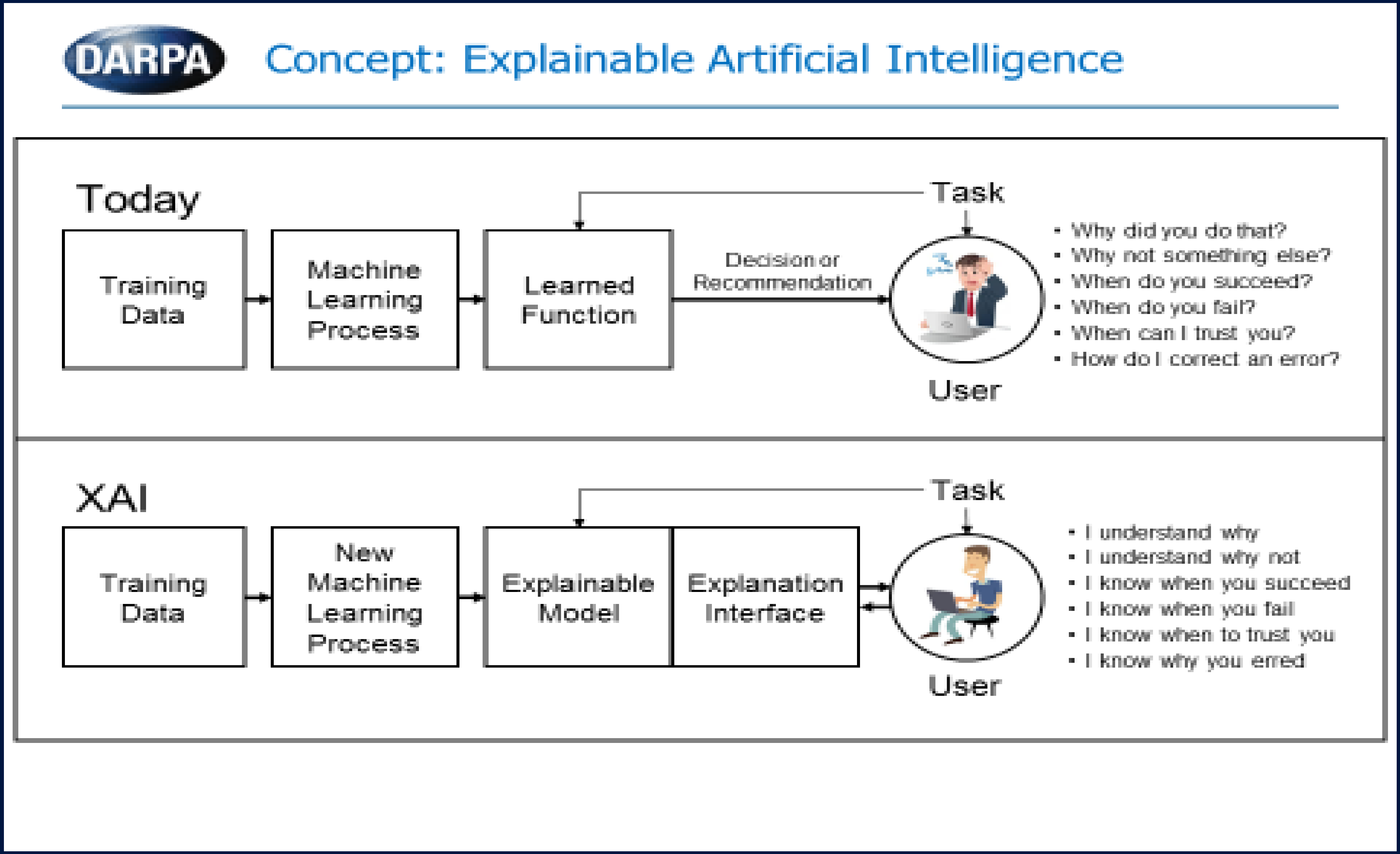
- A definition (or two)
- Altum Visum on deep learning networks
- Machine Learning: Myths & Realities
- Machine Learning as a process
- **Explainable Artificial Intelligence**
- Epilogue

A definition revised:

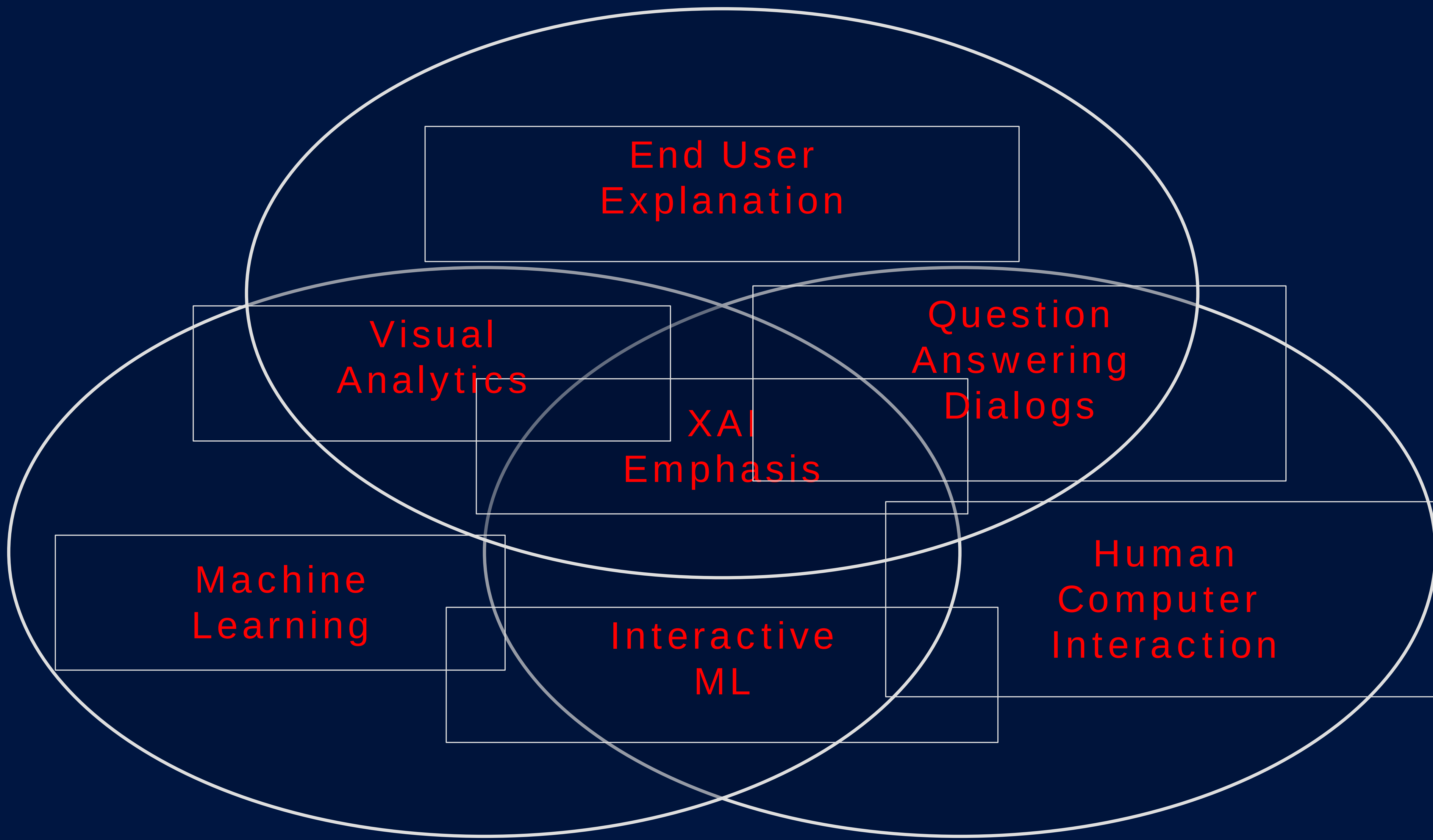
“**Machine learning (ML):** a subset of artificial intelligence (AI) is more than a technique for analyzing data. It's a system that is fueled by data, with the ability to learn and improve by using algorithms that provide new insights without being explicitly programmed to do so.”

Gartner, “Preparing and Architecting for Machine Learning”, Technical Professional Advice, published January 17, 2017.

Explainable Artificial Intelligence



Explainable Artificial Intelligence (XAI)



The Edward S. Rogers Sr. Department of Electrical & Computer Engineering
UNIVERSITY OF TORONTO

FACULTY OF APPLIED SCIENCE & ENGINEERING

Outline

- A definition (or two)
- Altum Visum on deep learning networks
- Machine Learning: Myths & Realities
- Machine Learning as a process
- Explainable Artificial Intelligence
- **Epilogue**

An old (?) paradox

The Moravec's Paradox (1988): "it is comparatively easy to make computers exhibit adult level performance on intelligent tests or playing checkers, and difficult or impossible to give them the skills of a one-year-old when it comes to perception and mobility" . [1]

[1] Hans Moravec, ***Mind Children:*** **The Future of Robot and Human Intelligence, Harvard University Press, 1988,** (ISBN 0674576187).

The Edward S. Rogers Sr. Department of Electrical & Computer Engineering
UNIVERSITY OF TORONTO
FACULTY OF APPLIED SCIENCE & ENGINEERING

Big Picture

Is DNN (or ML in general) a "Deus ex Machina Moment" ?

Epilogue

- Machine learning is best-suited for dealing with big, **albeit curated,** data.
- Supervised networks (DNN) can learn semantically relevant representations useful in areas such as (image) classification, content-aware advertising, content filtering, social networks.
- Preparing data for Machine Learning pipelines is challenging.
- Machine Learning implies “learning” – the ability to generalize from experience – not yet there.

Thank you!

kostas@ece.utoronto.ca

www.dsp.utoronto.ca