Deep Learning

0 Introduction and Course logistics

Dr. Konda Reddy Mopuri
kmopuri@ai.iith.ac.in
Dept. of AI, IIT Hyderabad
Jan-May 2023

Time slot

- B slot

Time slot

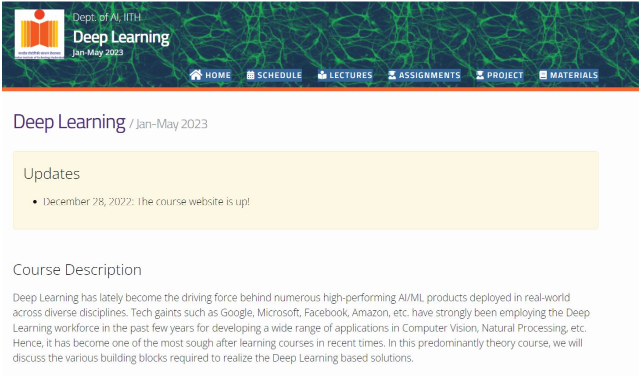
- B slot
- Monday 10 - 10:55 AM
- Wednesday 9 - 9:55 AM
- Thursday 11 - 11:55 AM

Time slot

- B slot
- Monday 10 - 10:55 AM
- Wednesday 9 - 9:55 AM
- Thursday 11 - 11:55 AM
- A-LH-1 (02.01.2023 to 15.01.2023 & 18.02.2023 to 02.05.2023), Auditorium (16.01.2023 to 17.02.2023)

Logistics

- Course website: `https://krmopuri.github.io/dl/`



Dept. of AI, IITH

Deep Learning

Jan-May 2023

HOME SCHEDULE LECTURES ASSIGNMENTS PROJECT MATERIALS

Deep Learning / Jan-May 2023

Updates

- December 28, 2022: The course website is up!

Course Description

Deep Learning has lately become the driving force behind numerous high-performing AI/ML products deployed in real-world across diverse disciplines. Tech gaints such as Google, Microsoft, Facebook, Amazon, etc. have strongly been employing the Deep Learning workforce in the past few years for developing a wide range of applications in Computer Vision, Natural Processing, etc. Hence, it has become one of the most sough after learning courses in recent times. In this predominantly theory course, we will discuss the various building blocks required to realize the Deep Learning based solutions.

Evaluation

- Assignments - 40% (best 4 of 5; 1 for each of the first 5 segment)
- Mid-1 (First week of Feb; after the 2nd segment) - 15%
- Mid-2 (Last week of March; after the 4th segment) - 15%
- Endsem - 30%

TAs

- Will update soon!

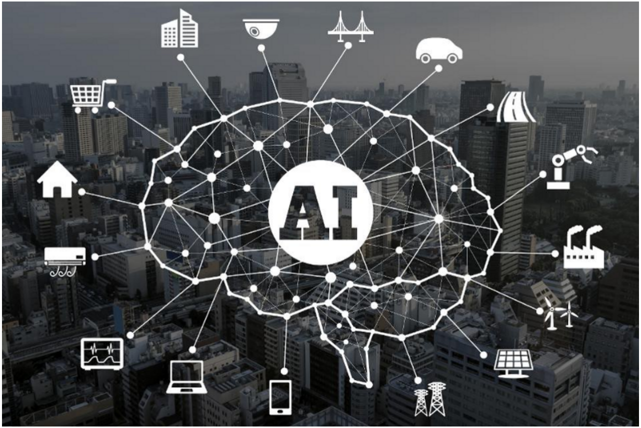
Contents

- Broadly: Building blocks of the Deep Learning based solutions

Contents

- Broadly: Building blocks of the Deep Learning based solutions
- Specifically: Please visit the website!

Why Deep Learning?



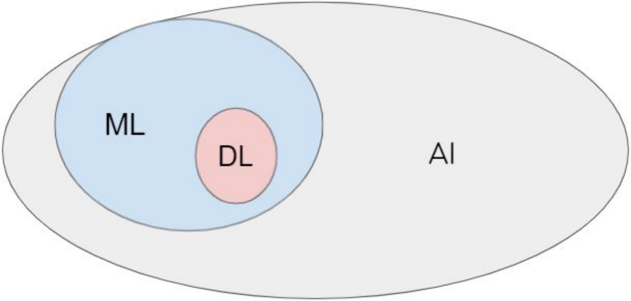
Deep Learning drives the recent AI boom. Image Source: Artificial Intelligence Magazine

Textbooks and References

- Lot of online resources
  - Deep Learning textbook by Ian Goodfellow *et al.*
  - Deep Learning: Methods and Applications, by D. Li and D. Yu
  - NPTEL course on Deep Learning by Prof. Mitesh Khapra, IITM
  - Michael Nielsen's text book on NN & DL
  - DL course by François Fleuret, Uni. of Geneva
  - PyTorch - https://pytorch.org/
  - Many more that I could not list and am not aware of...

What is DL?

What is DL?



Subset of ML that is essentially Artificial Neural Networks with more layers

What is DL?

- Crude attempt to imitate the human brain in learning

Classical ML vs. DL

- Classical ML: Handcrafted features + learnable model
- Need strong domain expertise

Classical ML vs. DL

- Classical ML: Handcrafted features + learnable model
- Need strong domain expertise

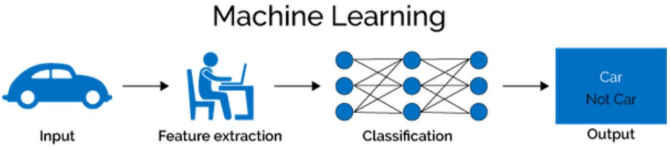


Figure credits: Jay Shaw & Quora

Classical ML vs. DL

- Deep Learning: Deep stack of parameterized processing
- End-to-End learning

Classical ML vs. DL

- Deep Learning: Deep stack of parameterized processing
- End-to-End learning

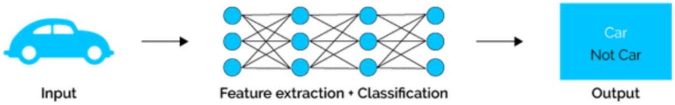


Figure credits: Jay Shaw & Quora

Classical ML vs. DL

भारतीय प्रौद्योगिकी संस्थान हैदराबाद
Indian Institute of Technology Hyderabad

- ANNs predate some of the classical ML techniques
- We are now dealing with a new generation ANNs

Neuron

- About 100 billion neurons in human brain

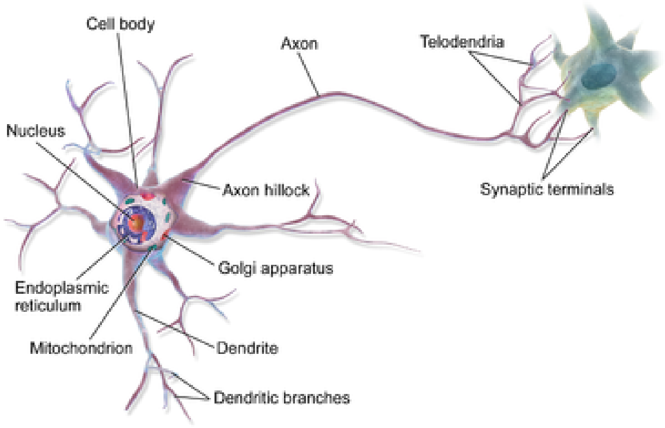


Figure credits: Wikipedia

History of Neural Networks

1. McCulloch Pitts neuron (1943) - Threshold Logic Unit

History of Neural Networks

1. McCulloch Pitts neuron (1943) - Threshold Logic Unit
2. Donald Hebb (1949) - Hebbian Learning Principle

History of Neural Networks

1. McCulloch Pitts neuron (1943) - Threshold Logic Unit
2. Donald Hebb (1949) - Hebbian Learning Principle
3. Marvin Minsky (1951) - created the first ANN (Hebbian Learning, 40 neurons)

History of Neural Networks

1. McCulloch Pitts neuron (1943) - Threshold Logic Unit
2. Donald Hebb (1949) - Hebbian Learning Principle
3. Marvin Minsky (1951) - created the first ANN (Hebbian Learning, 40 neurons)
4. Frank Rosenblatt (1958) - created perceptron to classify 20X20 images

History of Neural Networks

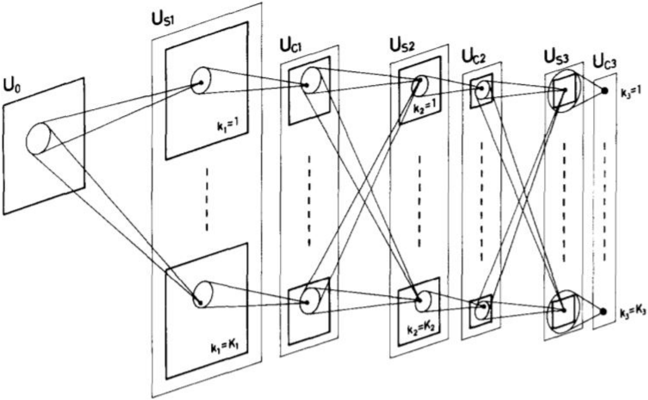
1. McCulloch Pitts neuron (1943) - Threshold Logic Unit
2. Donald Hebb (1949) - Hebbian Learning Principle
3. Marvin Minsky (1951) - created the first ANN (Hebbian Learning, 40 neurons)
4. Frank Rosenblatt (1958) - created perceptron to classify 20X20 images
5. David H Hubel and Torsten Wiesel (1959) demonstrated orientation selectivity and columnar organization in cat's visual cortex

Backpropagation

- Paul Werbos (1982) proposed back-propagation for ANNs

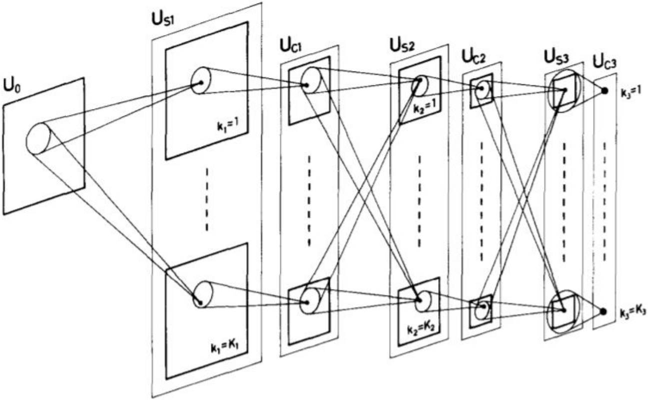
History (contd.)

1. Neocognitron by Fukushima (1980)



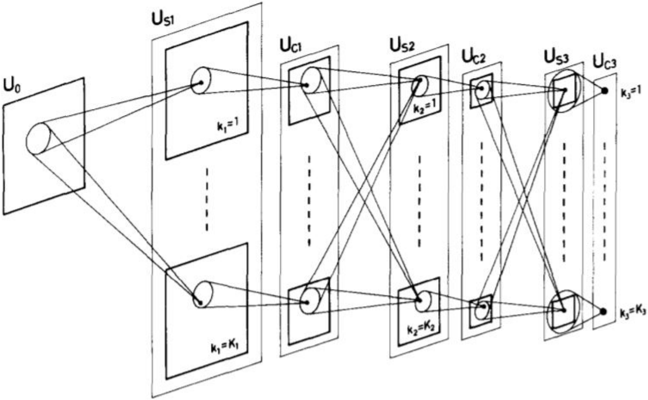
History (contd.)

1. Neocognitron by Fukushima (1980)
2. Implements the Hubel and Wiesel's principles



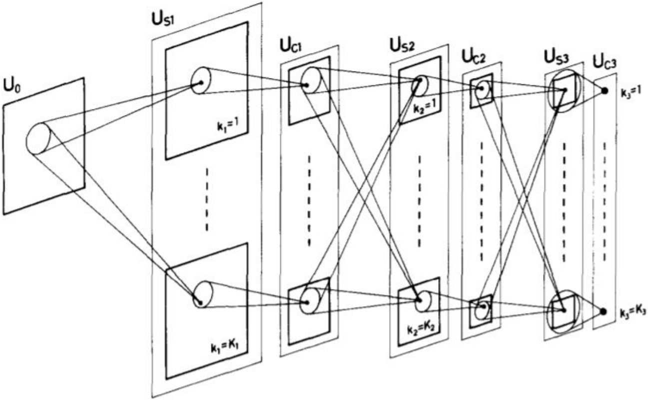
History (contd.)

1. Neocognitron by Fukushima (1980)
2. Implements the Hubel and Wiesel's principles
3. Used for hand-written digit recognition



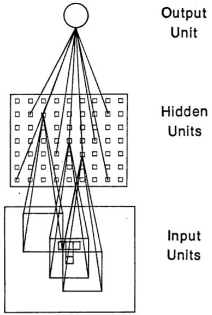
History (contd.)

1. Neocognitron by Fukushima (1980)
2. Implements the Hubel and Wiesel's principles
3. Used for hand-written digit recognition
4. Viewed as precursor for the modern CNNs



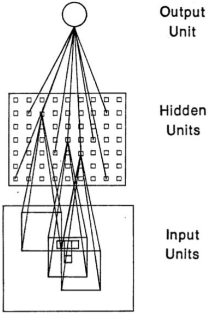
History (contd.)

1. Network for TC problem



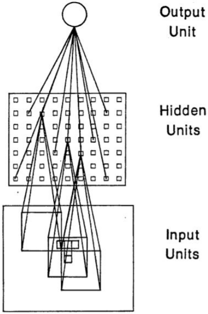
History (contd.)

1. Network for TC problem
2. Rumelhart (1988) trained with backprop



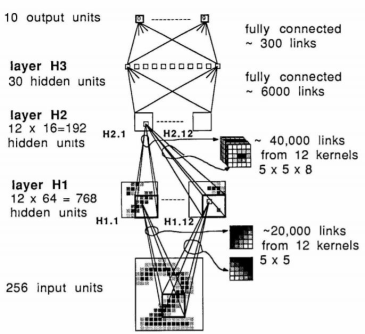
History (contd.)

1. Network for TC problem
2. Rumelhart (1988) trained with backprop
3. Showed that hidden units learn meaningful representations



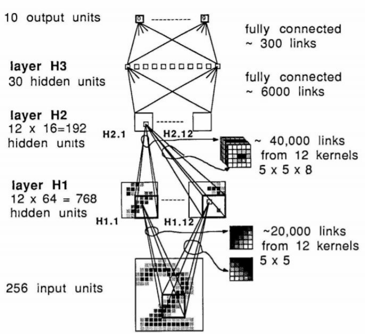
History (contd.)

1. LeNet family (Lecun et al. 1989) is a “convent”



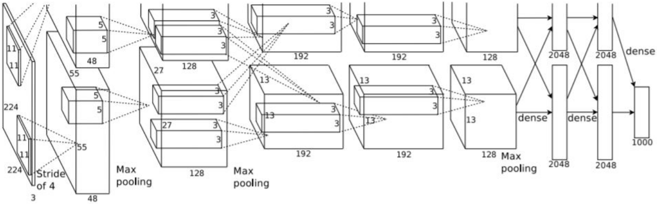
History (contd.)

1. LeNet family (Lecun et al. 1989) is a “convent”
2. Very similar to modern architectures



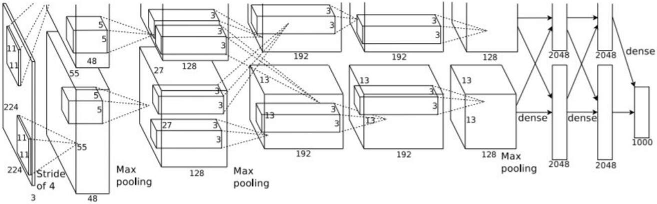
History (contd.)

1. AlexNet (2012)



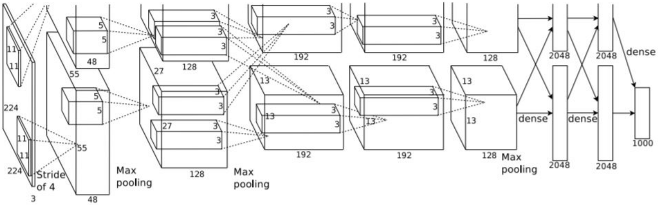
History (contd.)

1. AlexNet (2012)
2. Network similar to LeNet5, but of far greater size



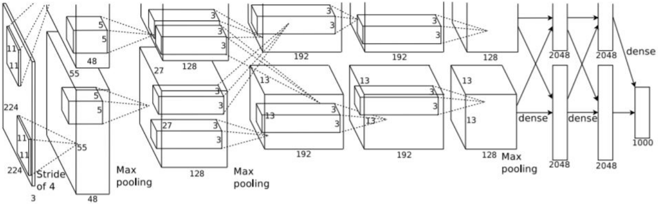
History (contd.)

1. AlexNet (2012)
2. Network similar to LeNet5, but of far greater size
3. Implemented using GPUs



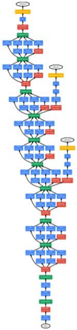
History (contd.)

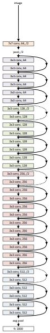
1. AlexNet (2012)
2. Network similar to LeNet5, but of far greater size
3. Implemented using GPUs
4. Could beat the SoTA image classification methods by a large margin



History (contd.)

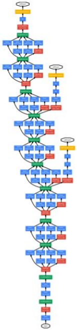
1. AlexNet initiated a trend of more complex and bigger architectures

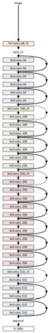




History (contd.)

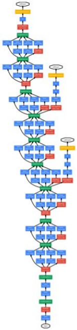
1. AlexNet initiated a trend of more complex and bigger architectures
2. GoogLeNet (2015) contains "inception" modules

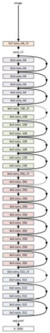




History (contd.)

1. AlexNet initiated a trend of more complex and bigger architectures
2. GoogLeNet (2015) contains “inception” modules
3. ResNet (2015) introduced “skip connections” that facilitate training deeper architectures





History (contd.)

1. Transformers (2017) are attention-based architectures

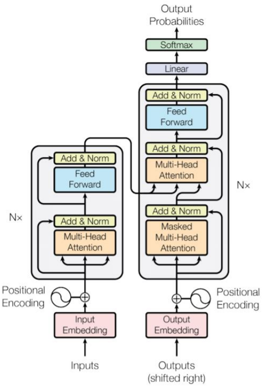


Figure credits: Vaswani et al., 2017

History (contd.)

1. Transformers (2017) are attention-based architectures
2. Very popular in NLP, and CV

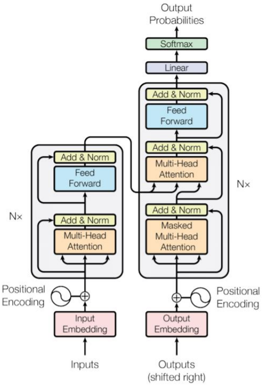


Figure credits: Vaswani et al., 2017

History (contd.)

1. Transformers (2017) are attention-based architectures
2. Very popular in NLP, and CV
3. Some of these models are extremely large. GPT-3 has 3 billion parameters (Brown et al. 2020)

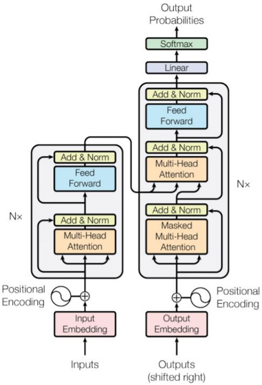


Figure credits: Vaswani et al., 2017

Deep Learning

1. Natural generalization to ANNs - Doesn't differ much from the 90s NNs

Deep Learning

1. Natural generalization to ANNs - Doesn't differ much from the 90s NNs
2. Computational graph of tensor operations that take advantage of
    - Chain rule (back-propagation)
    - SGD
    - GPUs
    - Huge datasets
    - Convolutions, etc.

Deep Learning

- This generalization enables us to build complex networks that work with Images, text, speech and sequences and train end-to-end

ILSVRC Error

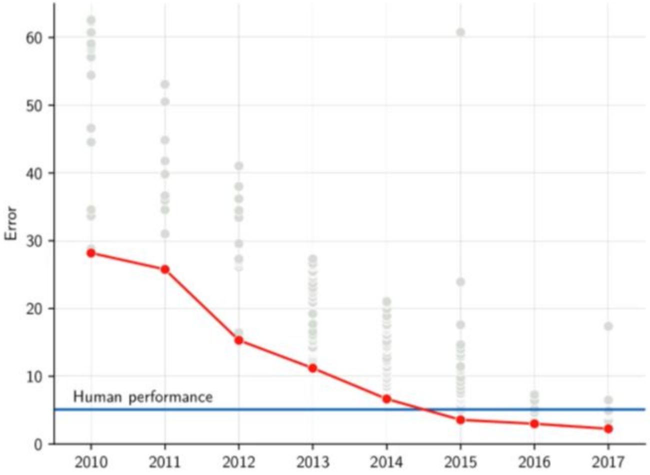


Figure credits: Gershgorn, 2017

What makes it work now?

What makes it work now?

1. Huge research and progress in ML

What makes it work now?

1. Huge research and progress in ML
2. Hardware developments - CPUs/GPUs/Storage technologies

What makes it work now?

1. Huge research and progress in ML
2. Hardware developments - CPUs/GPUs/Storage technologies
3. Piles of data over the Internet

What makes it work now?

1. Huge research and progress in ML
2. Hardware developments - CPUs/GPUs/Storage technologies
3. Piles of data over the Internet
4. Collaborative development (open source tools and forums for sharing/discussions, etc)

What makes it work now?

1. Huge research and progress in ML
2. Hardware developments - CPUs/GPUs/Storage technologies
3. Piles of data over the Internet
4. Collaborative development (open source tools and forums for sharing/discussions, etc)
5. Collective efforts from large institutions/corporations

What makes it work now?

1. Huge research and progress in ML
2. Hardware developments - CPUs/GPUs/Storage technologies
3. Piles of data over the Internet
4. Collaborative development (open source tools and forums for sharing/discussions, etc)
5. Collective efforts from large institutions/corporations
6. ...

What makes it work now?

- We have been doing a lot of ML already
  - Taxonomy of ML concepts: Classification, regression, generative models, clustering, etc.
  - Rich statistical formalizations: Bayesian estimation, PAC, etc.
  - Understood fundamentals: Bias-Variance, VC dimension, etc.
  - Good understanding of optimization
  - Efficient large-scale algorithms

Deep Learning - practical perspective

1. Doesn't require a deep mathematical grasp

Deep Learning - practical perspective

1. Doesn't require a deep mathematical grasp
2. Makes the design of large models a system/software development task

Deep Learning - practical perspective

1. Doesn't require a deep mathematical grasp
2. Makes the design of large models a system/software development task
3. Leverages modern hardware

Deep Learning - practical perspective

1. Doesn't require a deep mathematical grasp
2. Makes the design of large models a system/software development task
3. Leverages modern hardware
4. Doesn't seem to plateau with more data

Deep Learning - practical perspective

1. Doesn't require a deep mathematical grasp
2. Makes the design of large models a system/software development task
3. Leverages modern hardware
4. Doesn't seem to plateau with more data
5. Makes the trained models a commodity

Compute getting cheaper

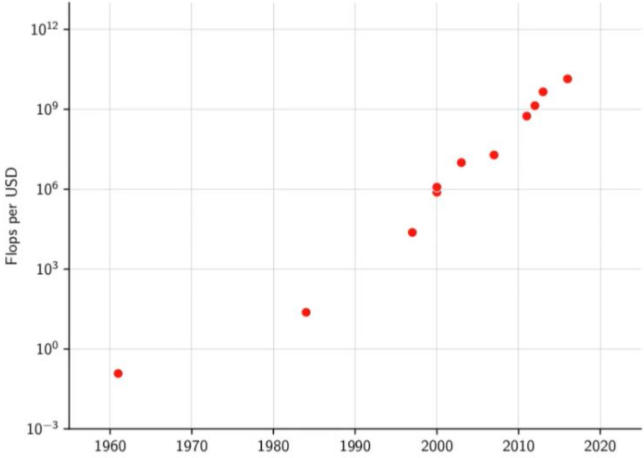


Figure Credits: Wikipedia

Storage getting cheaper

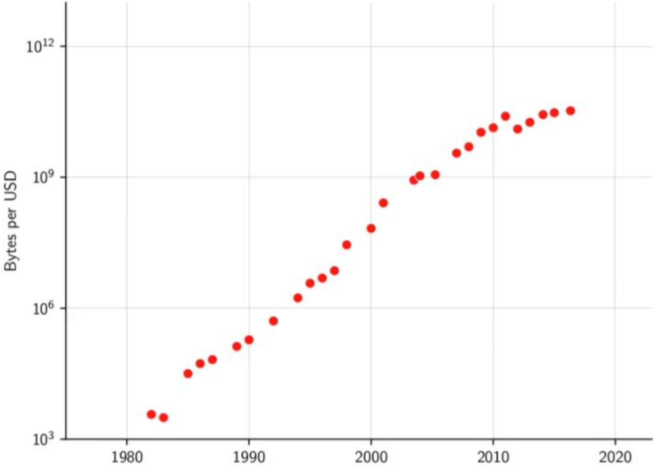


Figure Credits: John C Mccallum

AlexNet to AlphaGo: 300000X increase in compute

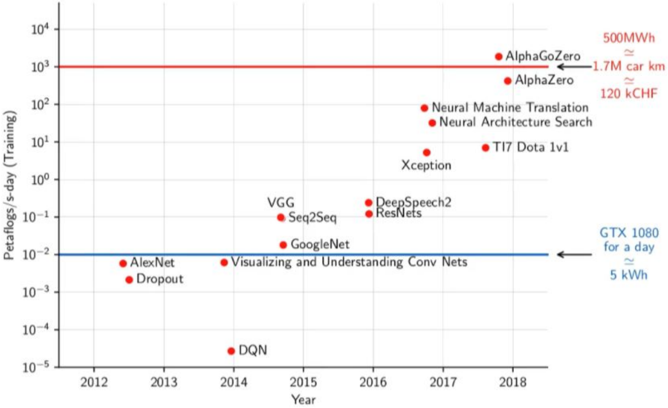


Figure Credits: Radford, 2018. 1 petaflop/s-day $\approx$ 100 GTX 1080 GPUs for a day, $\approx$ 500kwh

Datasets

Data-set		Year	Nb. images	Size
MNIST	(classification)	1998	60K	12Mb
Caltech 101	(classification)	2003	9.1K	130Mb
Caltech 256	(classification)	2007	30K	1.2Gb
CIFAR10	(classification)	2009	60K	160Mb
ImageNet	(classification)	2012	1.2M	150Gb
MS-COCO	(segmentation)	2015	200K	32Gb
Cityscape	(segmentation)	2016	25K	60Gb

Data-set		Year	Size
SST2	(sentiment analysis)	2013	20Mb
WMT-18	(translation)	2018	7Gb
OSCAR	(language model)	2020	6Tb

Figure Credits: François Fleuret

Implementation

भारतीय प्रौद्योगिकी संस्थान हैदराबाद
Indian Institute of Technology Hyderabad

	Language(s)	License	Main backer
PyTorch	**Python**, C++	BSD	Facebook
TensorFlow	Python, C++	Apache	Google
JAX	Python	Apache	Google
MXNet	Python, C++, R, Scala	Apache	Amazon
CNTK	Python, C++	MIT	Microsoft
Torch	Lua	BSD	Facebook
Theano	Python	BSD	U. of Montreal
Caffe	C++	BSD 2 clauses	U. of CA, Berkeley

Figure Credits: François Fleuret

We use PyTroch for this course

PyTorch

http://pytorch.org