



DEEP LEARNING

L-T-P-C: 2- 0- 3- 4



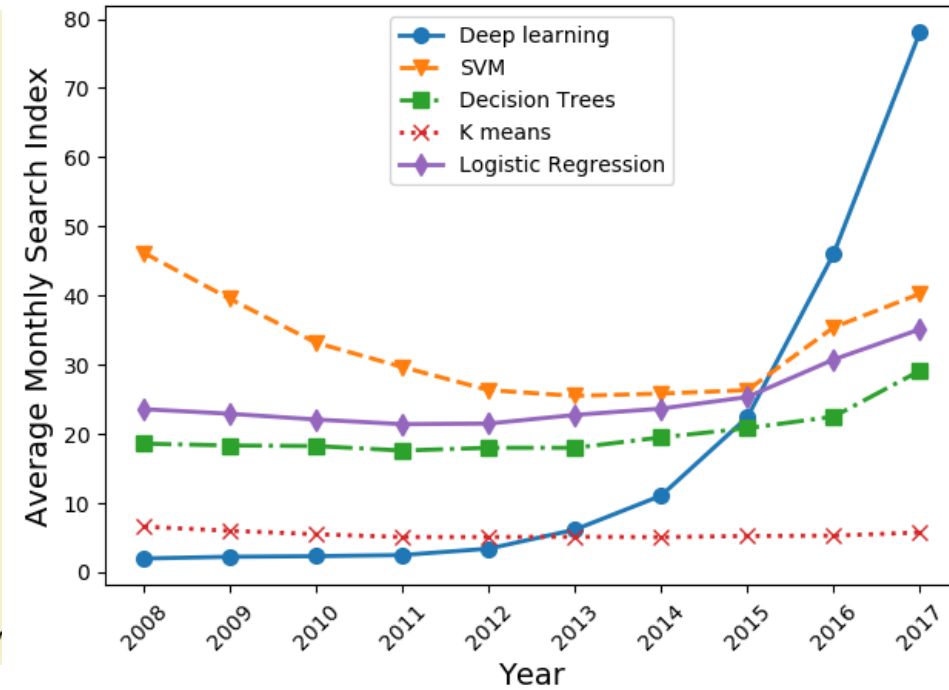
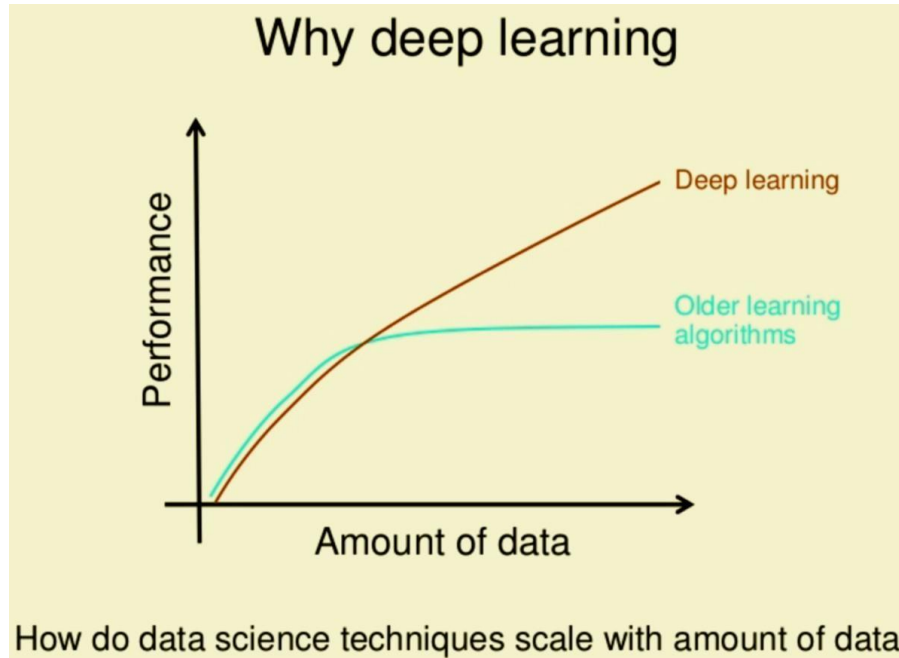
Amrita Vishwa Vidyapeetham
Amritapuri Campus



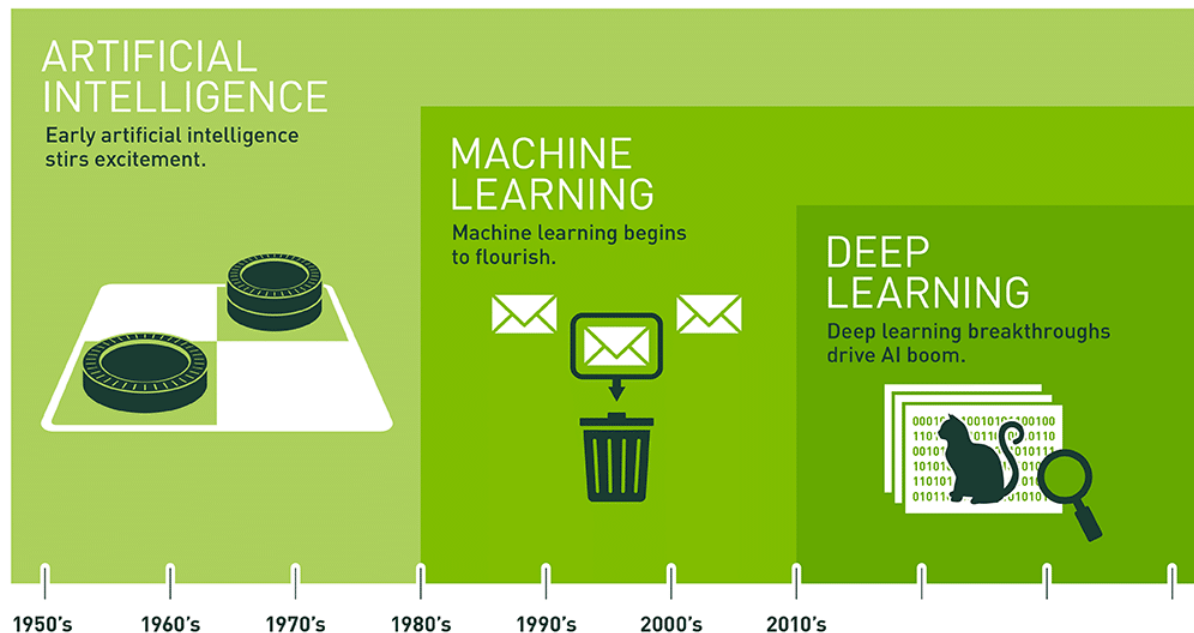
Deep Learning- A breakthrough

Given the Availability of Data, Deep Learning performance has surpassed all traditional algorithms

Artificial Intelligence Paradigmshift!- Machine Learning and Deep Learning?

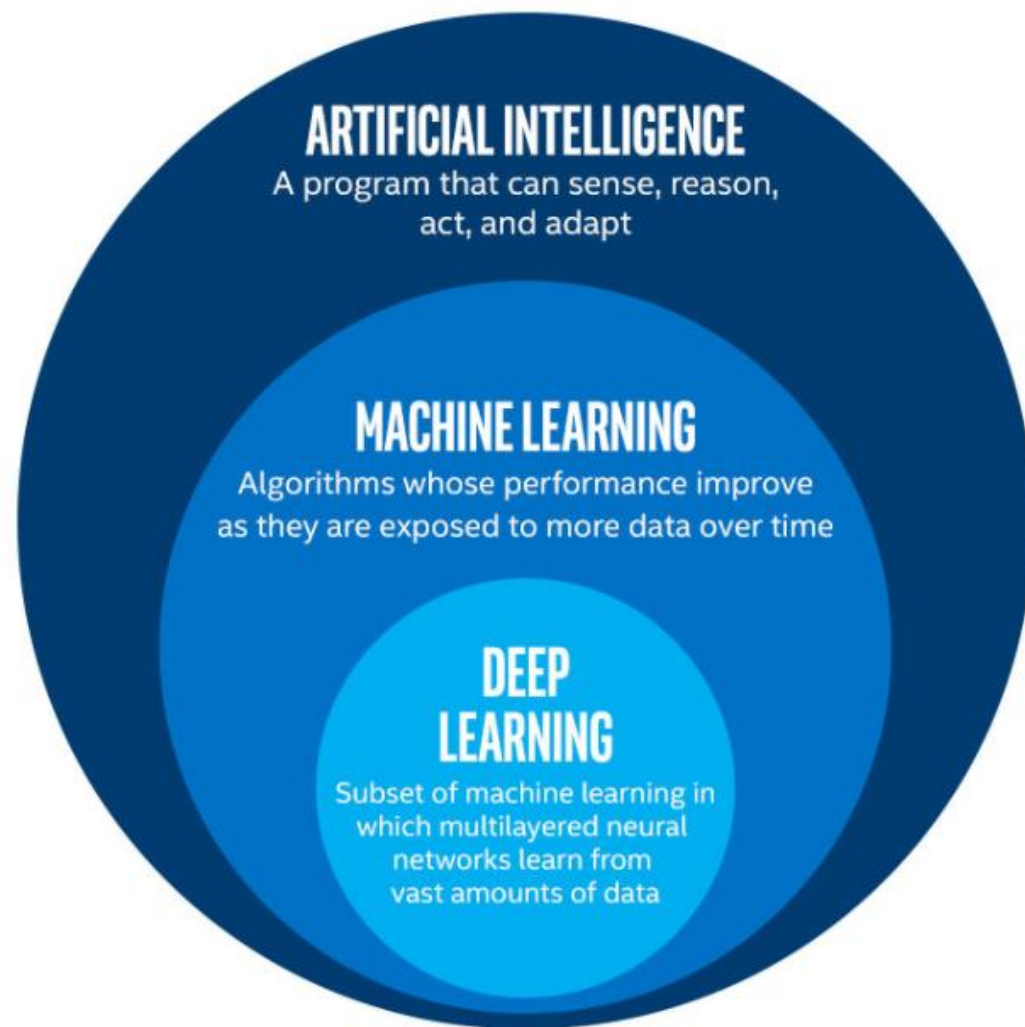
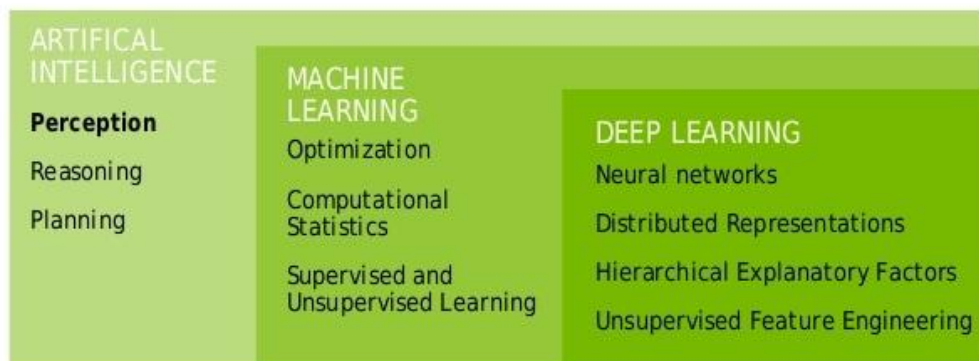


Promising results if trained with lot of data!



Since an early flush of optimism in the 1950s, smaller subsets of artificial intelligence – first machine learning, then deep learning – have emerged.

WHAT IS DEEP LEARNING?





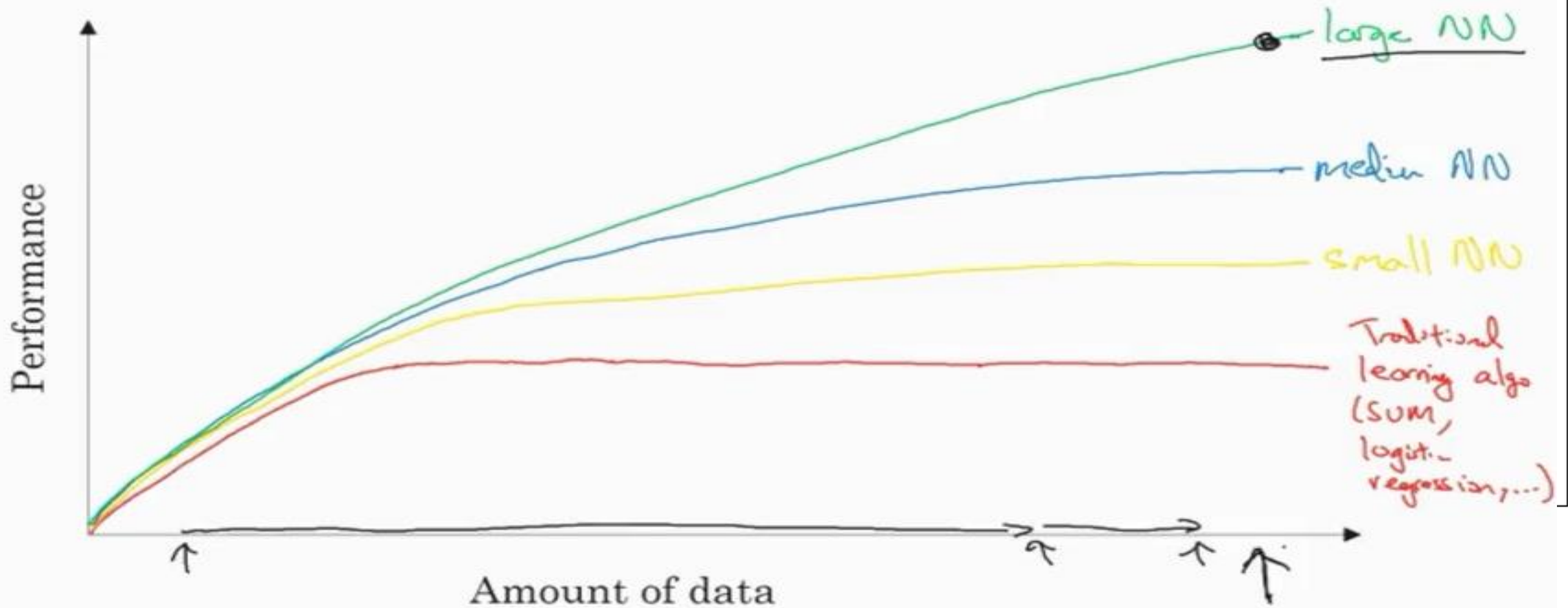
Deep Learning –

- **History**
- **Introduction**

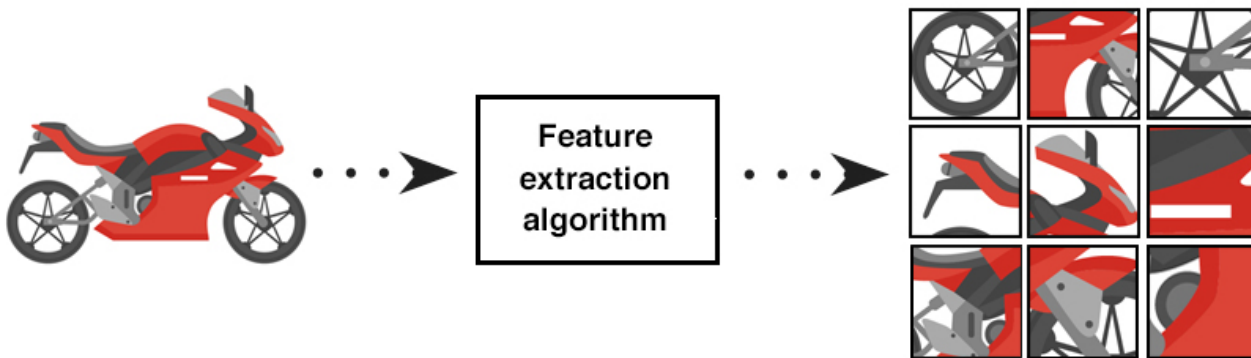
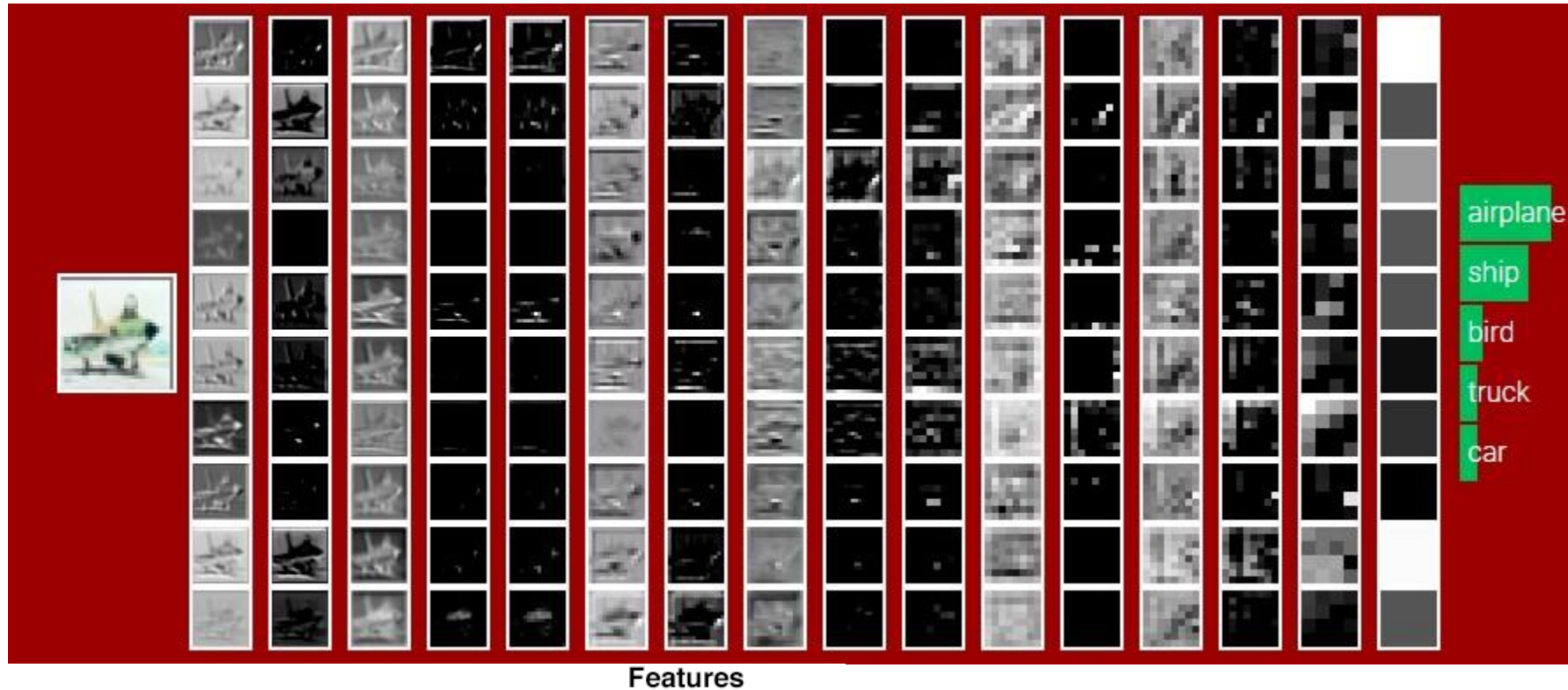


Artificial Intelligence Paradigmshift!- Machine Learning → **Deep** Learning?

Scale drives deep learning progress

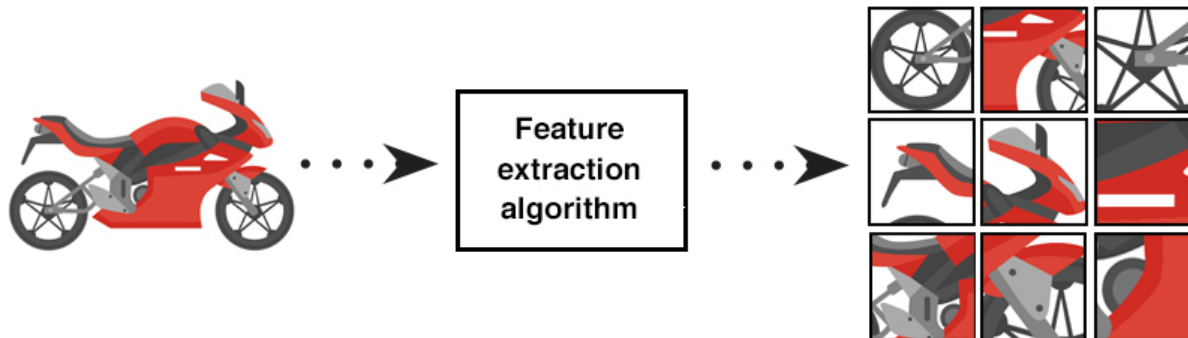
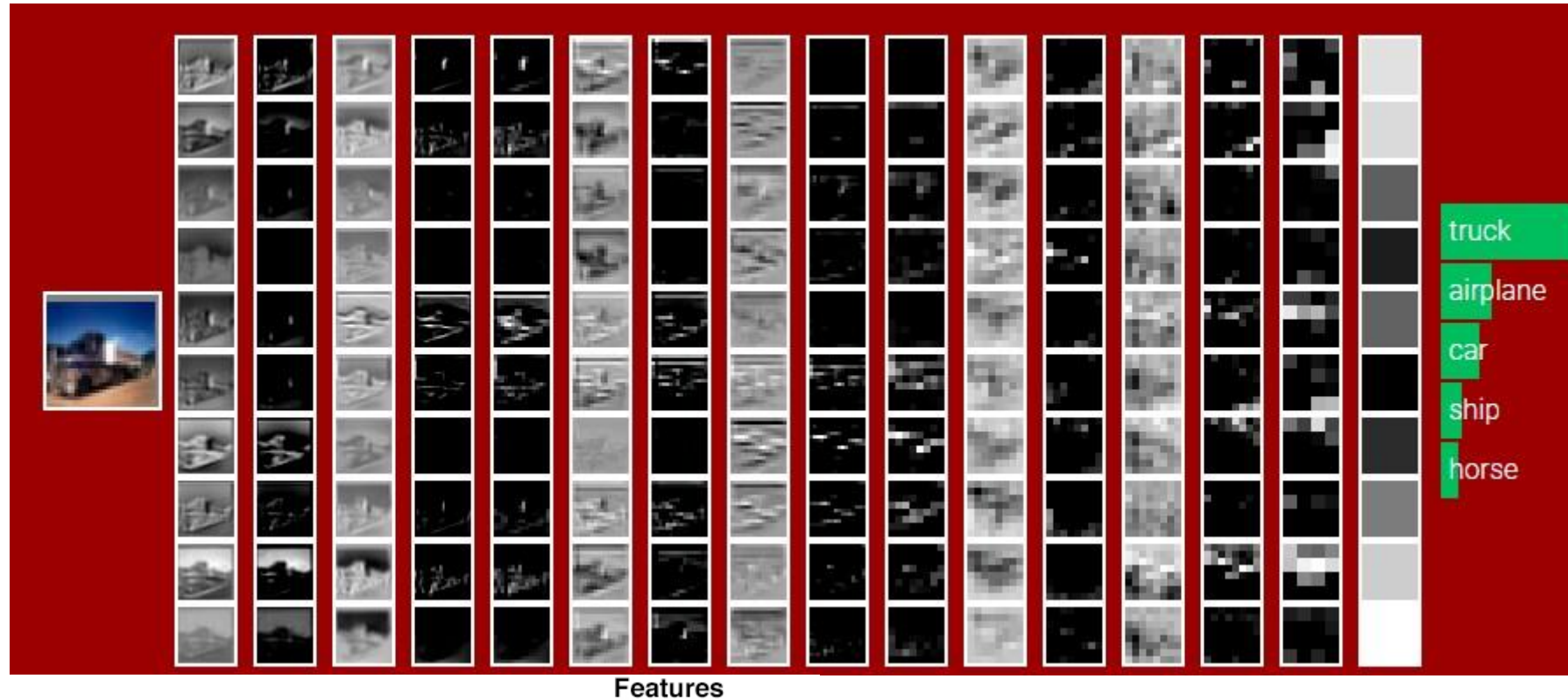


Deep Learning



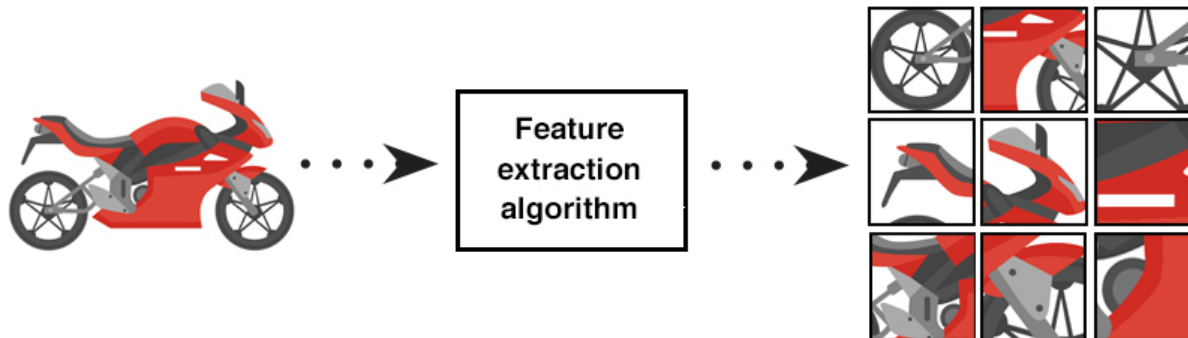
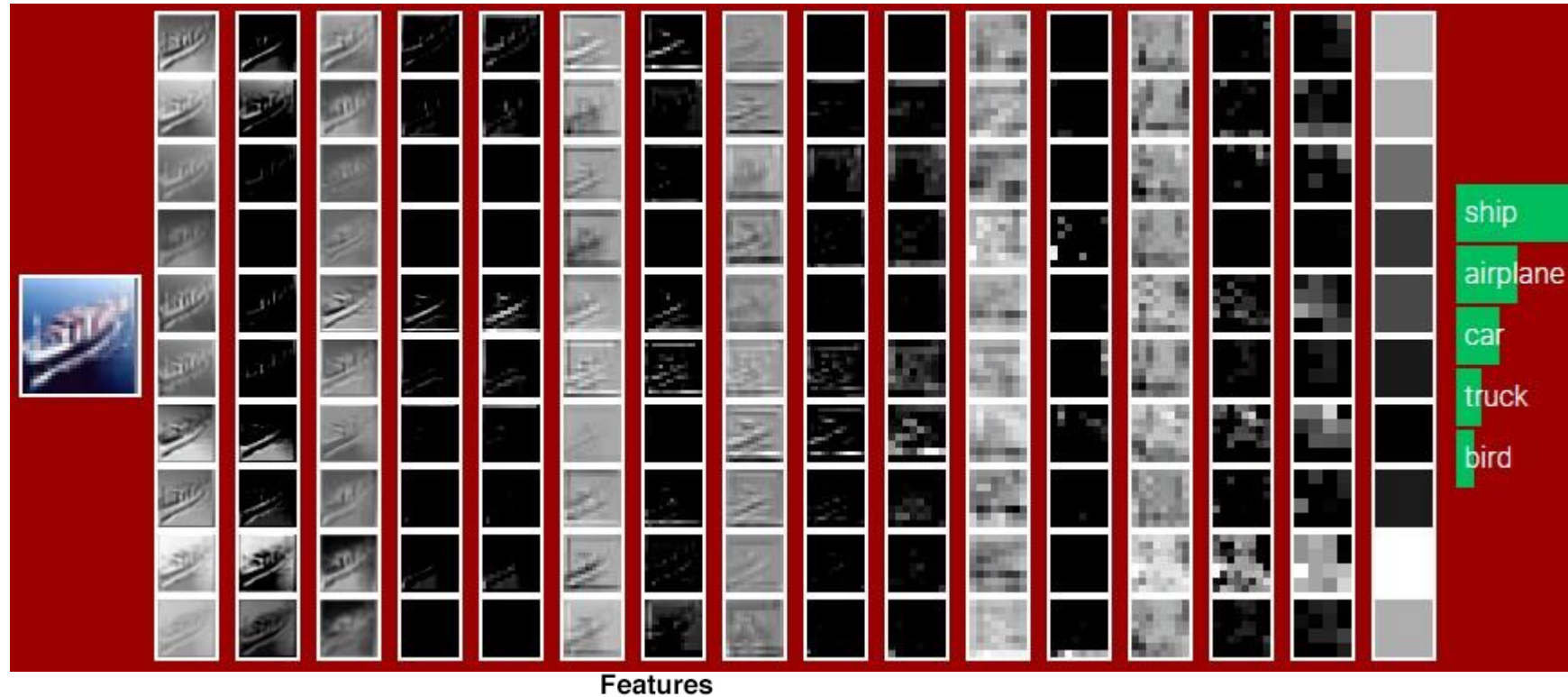
Deep Learning (DL) uses layers of algorithms to process data, understand human speech, and visually recognize objects.

Deep Learning-



Information is passed through each layer, with the output of the previous layer providing input for the next layer. The first layer in a network is called the **input layer**, while the last is called an **output layer**. All the layers between the two are referred to as **hidden layers**.

Deep Learning-

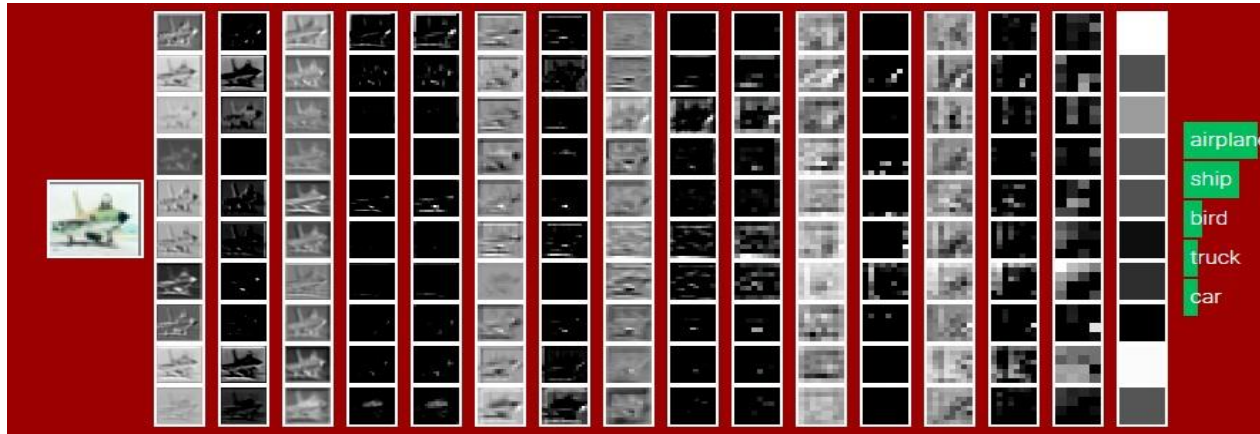


Information is passed through each layer, with the output of the previous layer providing input for the next layer. The first layer in a network is called the **input layer**, while the last is called an **output layer**. All the layers between the two are referred to as **hidden layers**.

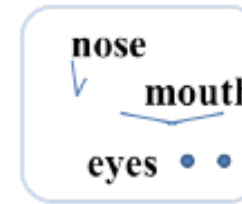
Neurons are arranged in a hierarchical fashion

Neurons are arranged in a hierarchical fashion and each layer has its own role and responsibility.

To detect a face, the brain could be relying on the entire network and not on a single layer.



Layer 1: detect edges & corners



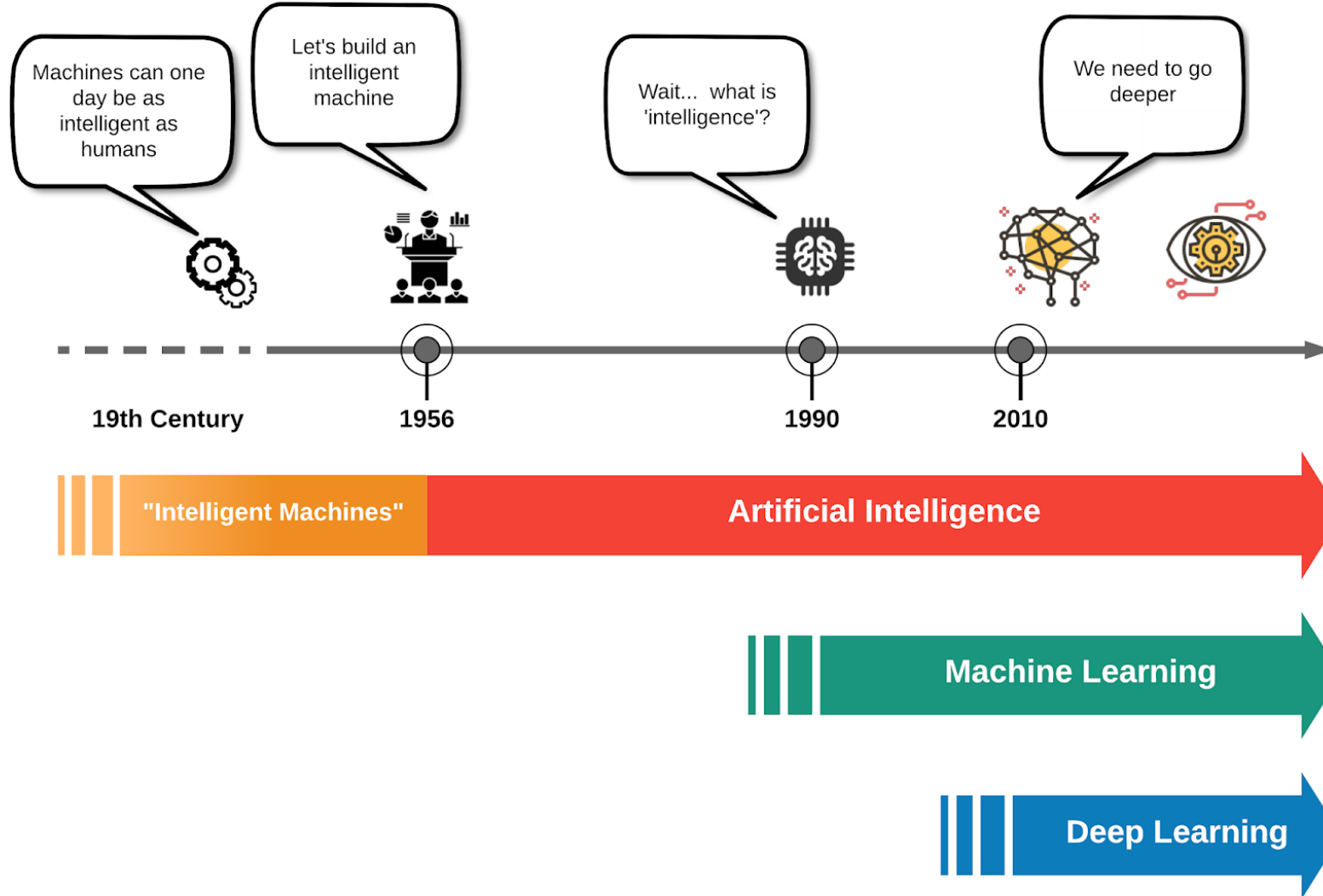
Layer 2: form feature groups



Layer 3: detect high level objects, faces, etc.

Courtesy :Sample illustration of hierarchical processing. Credits: Mitesh M. Khapra's lecture slides

Lets have a walk through History



Courtesy: medium



Warren McCulloch & Walter Pitts, wrote a paper on how neurons might work; they modeled a simple neural network with electrical circuits.

Nathanial Rochester from the IBM research laboratories led the first effort to simulate a neural network.

John von Neumann suggested imitating simple neuron functions by using telegraph relays or vacuum tubes.

STORY BY DATA

1943

1949

1950s

1956

1957

1958

HISTORY OF NEURAL NETWORKS

1943-2019

Donald Hebb reinforced the concept of neurons in his book, *The Organization of Behavior*. It pointed out that neural pathways are strengthened each time they are used.

The **Dartmouth Summer Research Project** on Artificial Intelligence provided a boost to both artificial intelligence and neural networks.

Frank Rosenblatt began work on the Perceptron; the oldest neural network still in use today.

1982

1981

1969

1959

1982

John Hopfield presented a paper to the national Academy of Sciences. His approach to create useful devices; he was likeable, articulate, and charismatic.

Progress on neural network research halted due fear, unfulfilled claims, etc.

Marvin Minsky & Seymour Papert proved the Perceptron to be limited in their book, *Perceptrons*.

Bernard Widrow & Marcian Hoff of Stanford developed models they called ADALINE and MADALINE; the first neural network to be applied to a real world problem.

1982

1985

1997

1998

NOW

US-Japan Joint Conference on Cooperative/Competitive Neural Networks; Japan announced their Fifth-Generation effort resulted in US worrying about being left behind and restarted the funding in US.

American Institute of Physics began what has become an annual meeting - **Neural Networks for Computing**.

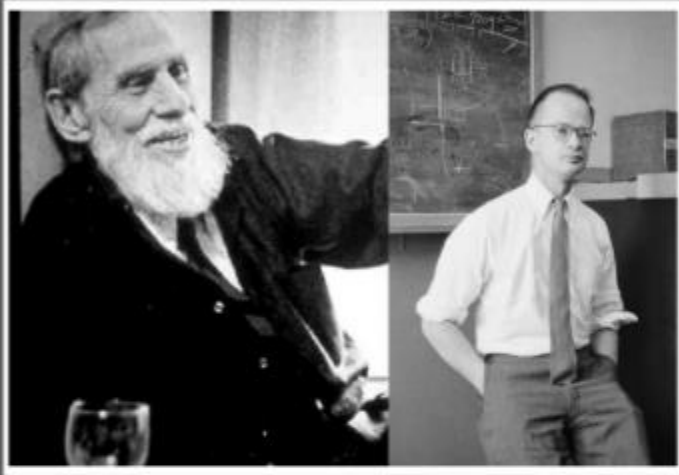
A recurrent neural network framework, LSTM was proposed by Schmidhuber & Hochreiter.

Yann LeCun published *Gradient-Based Learning Applied to Document Recognition*.

Neural networks discussions are prevalent; the future is here!

Courtesy: medium

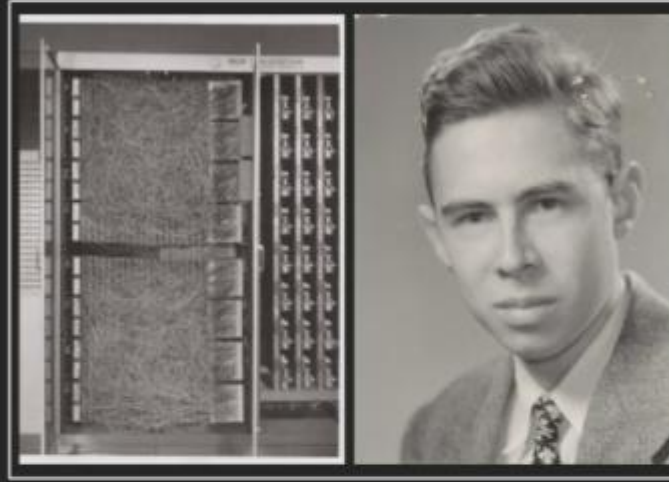
McCulloch Pitts Neuron – Beginning



Walter Pitts and Warren McCulloch in their paper, "*A Logical Calculus of the Ideas Immanent in Nervous Activity*" shows the mathematical model of biological neuron. This McCulloch Pitts Neuron has very limited capability and has no learning mechanism. Yet it will lay the foundation for artificial neural network & deep learning.

1943

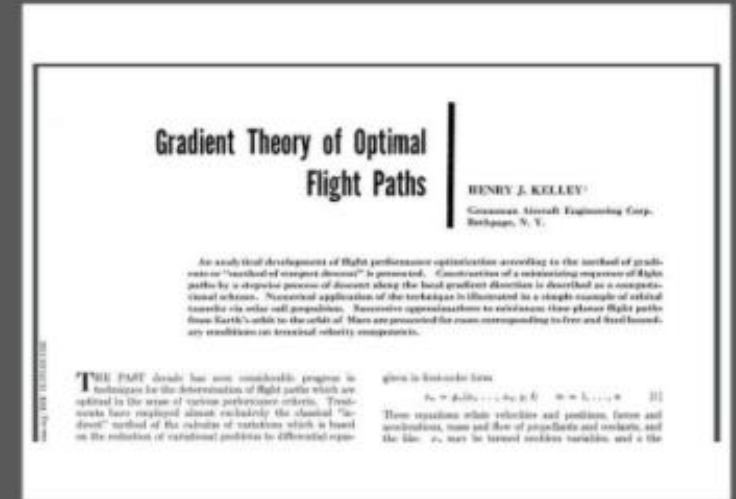
Frank Rosenblatt Creates Perceptron



In his paper "The Perceptron: A Perceiving and Recognizing Automaton", Rosenblatt shows the new avatar of McCulloch-Pitts neuron – 'Perceptron' that had true learning capabilities to do binary classification on it's own. This inspires the revolution in research of shallow neural network for years to come, till first AI winter.

1957

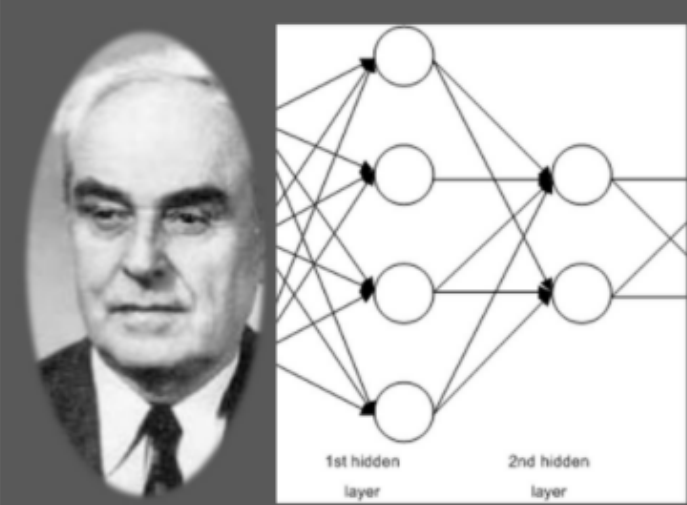
The First Backpropagation Model



Henry J. Kelley in his paper, "Gradient Theory of Optimal Flight Paths" shows the first ever version of continuous backpropagation model. His model is in context to Control Theory, yet it lays the foundation for further refinement in the model and would be used in ANN in future years.

1960

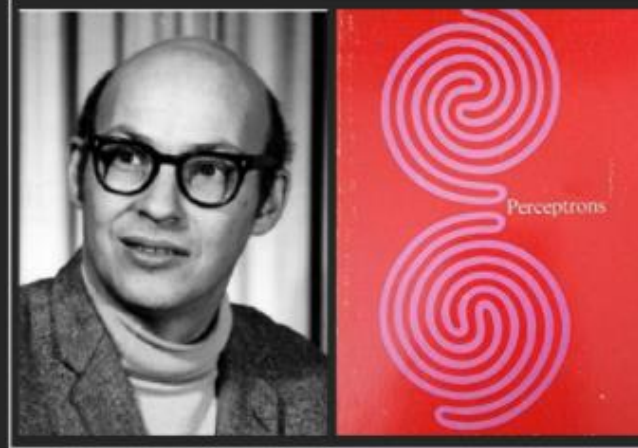
Birth Of Multilayer Neural Network



Alexey Grigoryevich Ivakhnenko along with Valentin Grigor'evich Lapa, creates hierarchical representation of neural network that uses polynomial activation function and are trained using Group Method of Data Handling (GMDH). It is now considered as the first ever multi-layer perceptron and Ivakhnenko is often considered as father of deep learning.

1965

The Fall Of Perceptron



Marvin Minsky and Seymour Papert publishes the book "Perceptrons" in which they show that Rosenblatt's perceptron cannot solve complicated functions like XOR. For such function perceptrons should be placed in multiple hidden layers which compromises perceptron learning algorithm. This setback triggers the winter of neural network research.

1969

Backpropagation Is Computer Coded



Seppo Linnainmaa publishes general method for automatic differentiation for backpropagation and also implements backpropagation in computer code. The research in backpropagation has now come very far, yet it would not be implemented in neural network till next decade.

1970

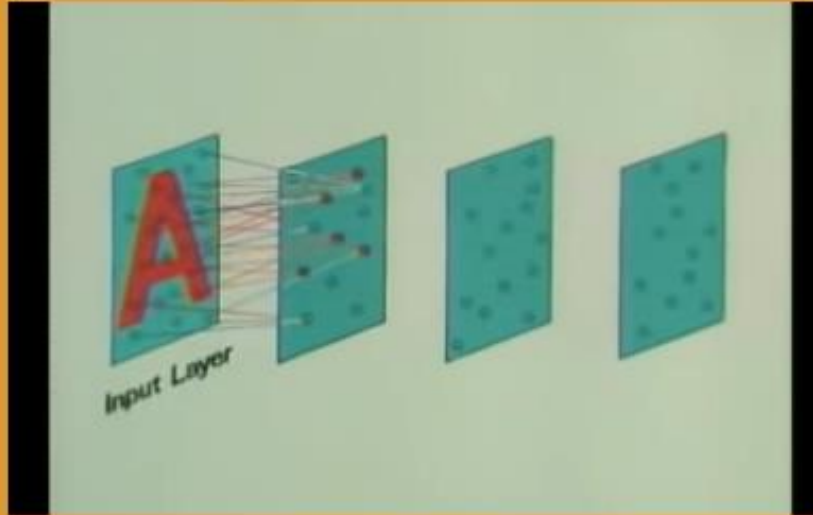
Neural Network Goes Deep



Alexey Grigoryevich Ivakhnenko continues his research in Neural Network. He creates 8-layer Deep neural network using Group Method of Data Handling (GMDH).

1970

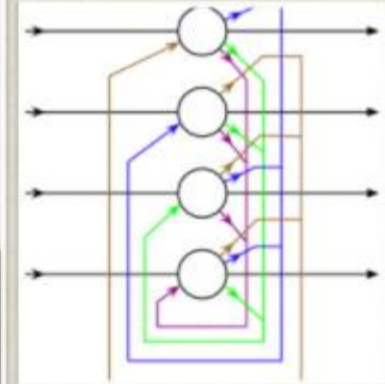
Neocognitron – First CNN Architecture



Kunihiko Fukushima comes up with Neocognitron, the first convolutional neural network architecture which could recognize visual patterns such as handwritten characters.

1980

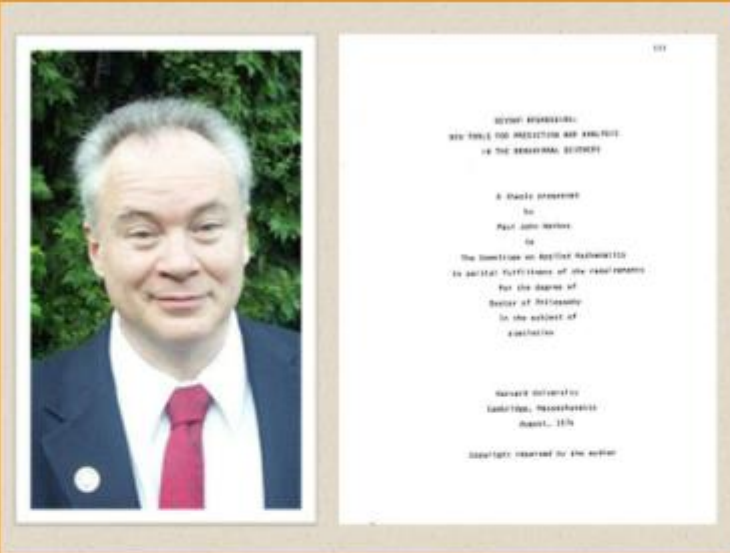
Hopfield Network – Early RNN



John Hopfield creates Hopfield Network, which is nothing but a recurrent neural network. It serves as a content-addressable memory system, and would be instrumental for further RNN models of modern deep learning era.

1982

Proposal For Backpropagation In ANN



Paul Werbos, based on his 1974 Ph.D. thesis, publicly proposes the use of Backpropagation for propagating errors during the training of Neural Networks. His results of the Ph.D. thesis will eventually lead to the practical adoption of backpropagation by the neural network community in the future.

1982

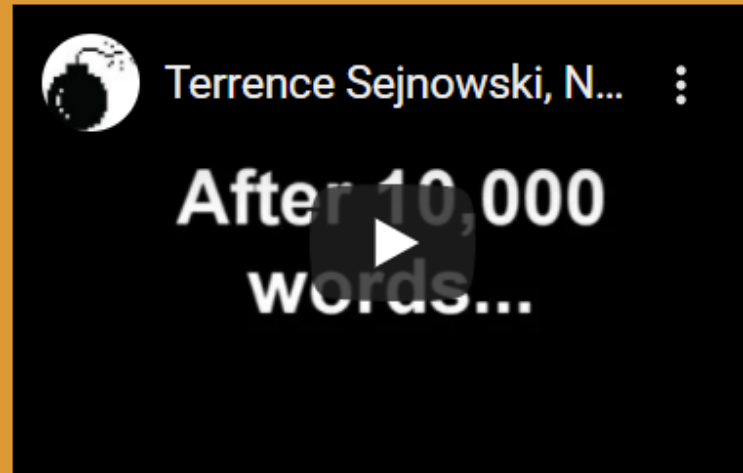
Boltzmann Machine



David H. Ackley, Geoffrey Hinton and Terrence Sejnowski create Boltzmann Machine that is a stochastic recurrent neural network. This neural network has only input layer and hidden layer but no output layer.

1985

NetTalk – ANN Learns Speech



Terry Sejnowski creates NeTalk, a neural network which learns to pronounce written English text by being shown text as input and matching phonetic transcriptions for comparison.

1986

Implementation Of Backpropagation



Learning representations by back-propagating errors

David E. Rumelhart*, Geoffrey E. Hinton*
& Ronald J. Williams*

*Institute for Cognitive Science, © 1986, University of California, San Diego, La Jolla, California 92037, USA
†Department of Computer Science, Carnegie-Mellon University, Pittsburgh, Pennsylvania 15213, USA

We describe a new learning procedure, back-propagation, for networks of neurone-like units. The procedure repeatedly adjusts the weights of the connections in the network so as to minimize a measure of the difference between the actual output vector of the net and the desired output vector. As a result of the weight adjustments, internal 'hidden' units which are not part of the input or output come to represent important features of the task domain, and the regularities in the task are captured by the interactions of these units. The ability to create useful new features distinguishes back-propagation from earlier, simpler methods such as the perceptron-convergence procedure.

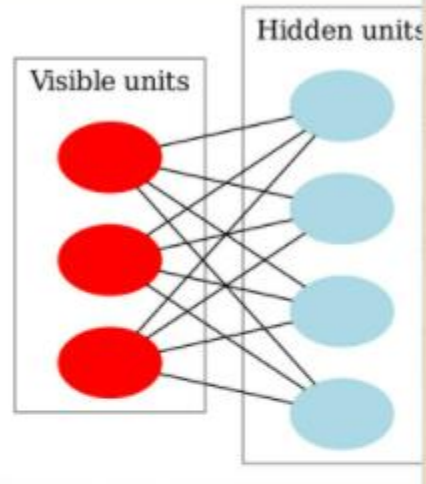
There have been many attempts to design self-organizing neural networks. The aim is to find a powerful synaptic-modification rule that will allow an arbitrarily-connected neural network to develop an internal structure that is appropriate for a particular task domain. The task is specified by giving the desired state vector of the output units for each state vector of the input units. If the input units are directly connected to the output units it is relatively easy to find learning rules that iteratively adjust the relative strengths of the connections so as to progressively reduce the difference between the actual and desired output vectors. Learning becomes more interesting but

*By other correspondence should be addressed

Geoffrey Hinton, Rumelhart, and Williams in their paper "Learning Representations by back-propagating errors" show the successful implementation of backpropagation in the neural network. It opened gates for training complex deep neural network easily which was the main obstruction in earlier days of research in this area.

1986

Restricted Boltzmann Machine



Paul Smolensky comes up with a variation of Boltzmann Machine where there is not intra layer connection in input and hidden layer. It is known as Restricted Boltzmann Machine (RBM). It would become popular in years to come especially for building recommender systems.

1986

CNN Using Backpropagation



Yann LeCun uses backpropagation to train convolutional neural network to recognize handwritten digits. This is a breakthrough moment as it lays the foundation of modern computer vision using deep learning.

1989

Universal Approximators Theorem



George Cybenko publishes earliest version of the Universal Approximation Theorem in his paper "*Approximation by superpositions of a sigmoidal function*". He proves that feed forward neural network with single hidden layer containing finite number of neurons can approximate any continuous function. It further adds credibility to Deep Learning.

1989

Vanishing Gradient Problem Appears



Sepp Hochreiter identifies the problem of vanishing gradient which can make the learning of deep neural network extremely slow and almost impractical. This problem will continue to annoy deep learning community for many more years to come.

1990

The Milestone Of LSTM



Sepp Hochreiter and Jürgen Schmidhuber publishes a milestone paper on "Long Short-Term Memory" (LSTM). It is a type of recurrent neural network architecture which will go on to revolutionize deep learning in decades to come.

1997

Deep Belief Network



A fast learning algorithm for deep belief nets

Geoffrey Hinton and Simon Osindero
University of Toronto
270 College Street
Toronto, Canada M5S 3G4
hinton@cs.toronto.edu

Yee-Whye Teh
Department of Computer Science
National University of Singapore
3 Science Drive 3, Singapore
tehyw@comp.nus.edu.sg

The authors propose a fast learning algorithm for deep belief networks. The algorithm consists of three stages: (1) pre-training each layer of the network as a Restricted Boltzmann Machine (RBM), (2) initializing the weights of the layers, and (3) fine-tuning the network using a slow learning algorithm. The authors claim that this algorithm is much faster than other methods for training deep belief networks.

remaining hidden layers form converts the representations in observable variables such as the hidden model has some attractive

1. There is a fast, greedy learning algorithm for training a fairly good set of parameters for networks with millions of layers.
2. The learning algorithm is applied to labeled data by both the label and the data.
3. There is a fine-tuning algorithm for training a network with three layers.

Geoffrey Hinton, Ruslan Salakhutdinov, Osindero and Teh publishes the paper "A fast learning algorithm for deep belief nets" in which they stacked multiple RBMs together in layers and called them Deep Belief Networks. The training process is much more efficient for large amount of data.

2006

GPU Revolution Begins



Andrew NG's group in Stanford starts advocating for the use of GPUs for training Deep Neural Networks to speed up the training time by many folds. This could bring practicality in the field of Deep Learning for training on huge volume of data efficiently.

2008

ImageNet Is Launched



Finding enough labeled data has always been a challenge for Deep Learning community. In 2009 Fei-Fei Li, a professor at Stanford, launches ImageNet which is a database of 14 million labeled images. It would serve as a benchmark for the deep learning researchers who would participate in ImageNet competitions (ILSVRC) every year.

2010

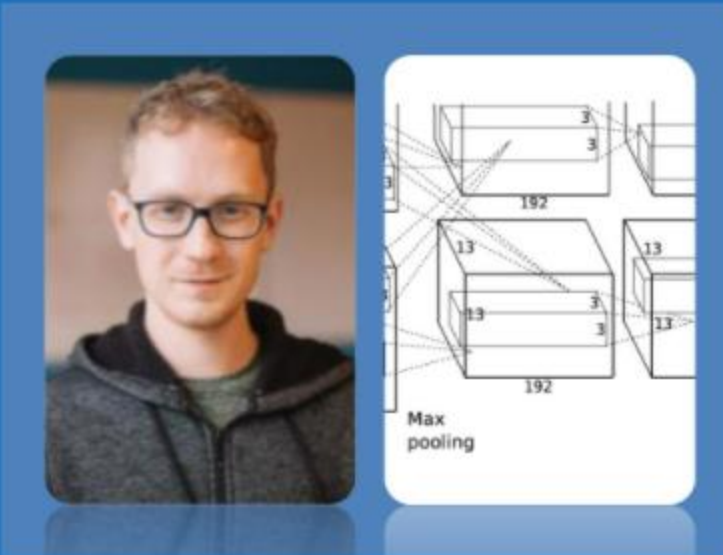
Combat For Vanishing Gradient



Yoshua Bengio, Antoine Bordes, Xavier Glorot in their paper "Deep Sparse Rectifier Neural Networks" shows that ReLU activation function can avoid vanishing gradient problem. This means that now, apart from GPU, deep learning community has another tool to avoid issues of longer and impractical training times of deep neural network.

2011

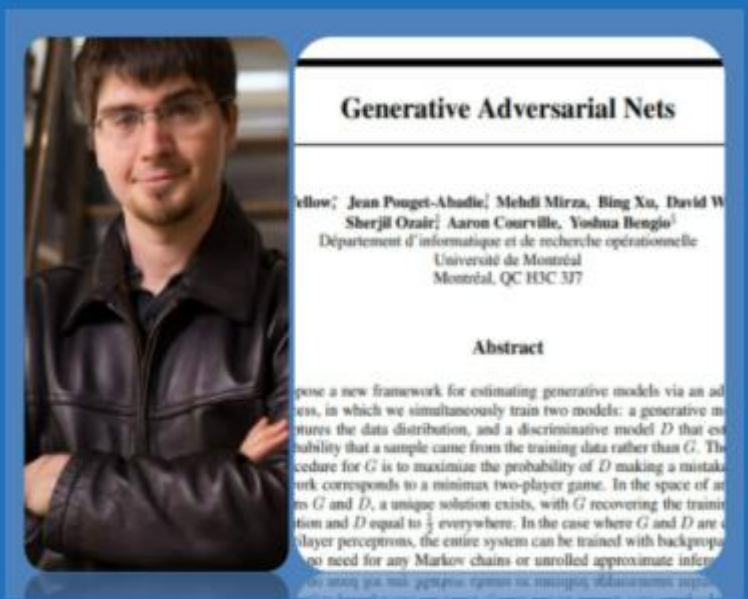
AlexNet Starts Deep Learning Boom



AlexNet, a GPU implemented CNN model designed by Alex Krizhevsky, wins Imagenet's image classification contest with accuracy of 84%. It is a huge jump over 75% accuracy that earlier models had achieved. This win triggers a new deep learning boom globally.

2012

The Birth Of GANs



Generative Adversarial Neural Network also known as GAN is created by Ian Goodfellow. GANs open a whole new doors of application of deep learning in fashion, art, science due it's ability to synthesize real like data.

2014

AlphaGo Beats Human



Deepmind's deep reinforcement learning model beats human champion in the complex game of Go. The game is much more complex than chess, so this feat captures the imagination of everyone and takes the promise of deep learning to whole new level.

2016

Geoffrey Hinton

- long known as the "**Godfather of Deep Learning**"
- now a Google researcher

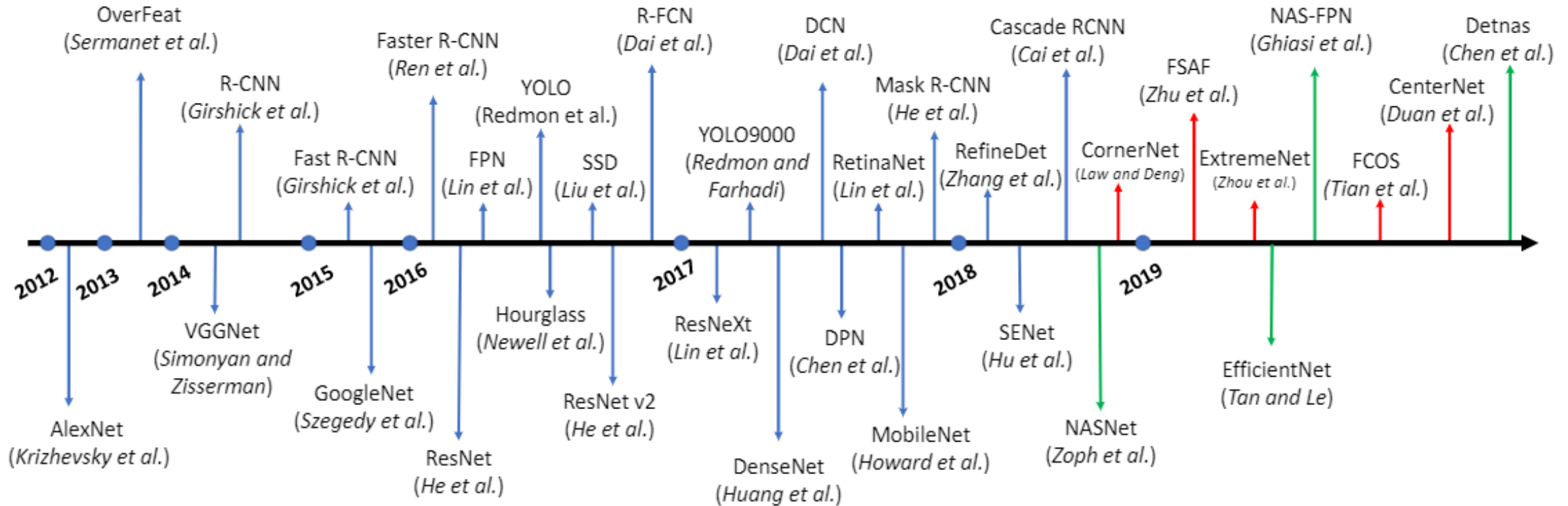
Trio Win Turing Award



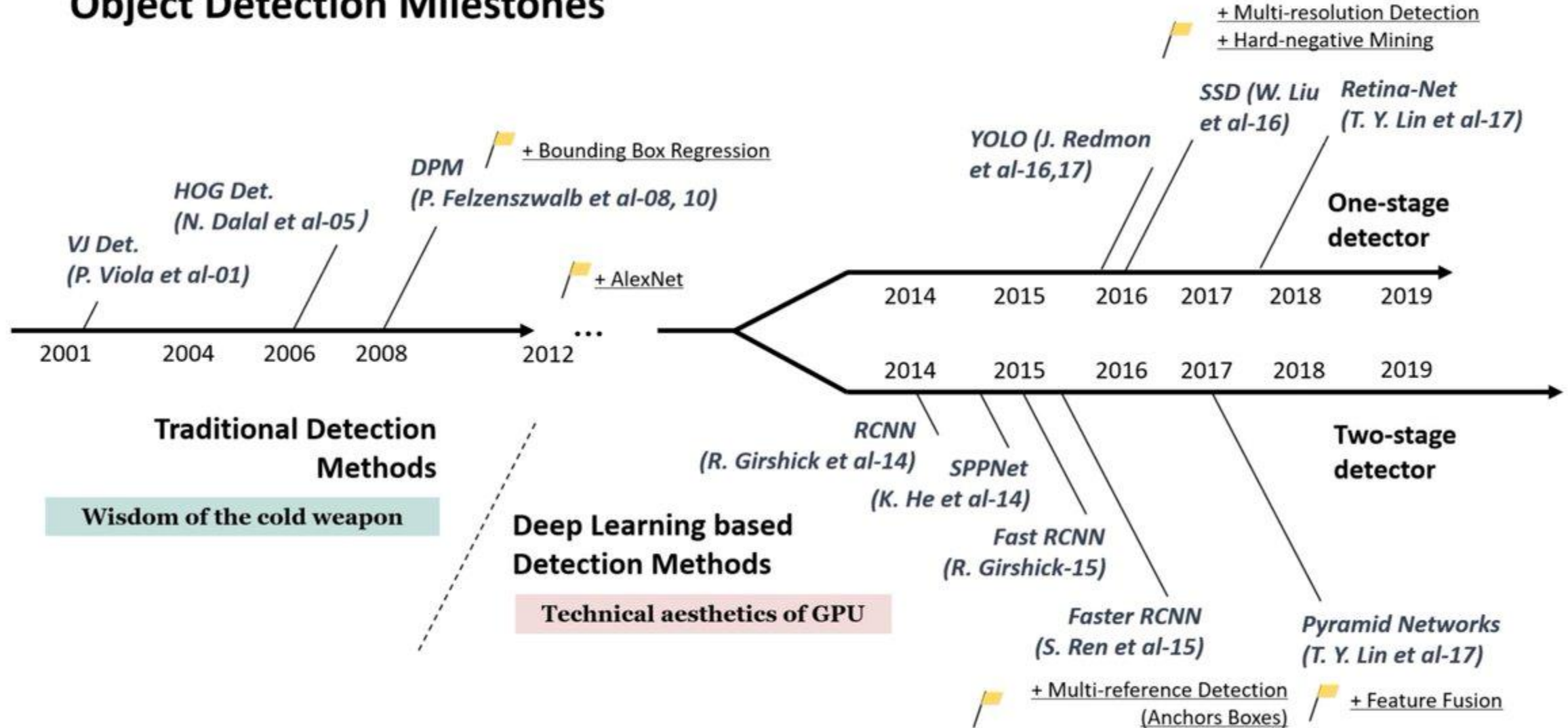
Yoshua Bengio, Geoffrey Hinton, and Yann LeCun wins Turing Award 2018 for their immense contribution in advancements in area of deep learning and artificial intelligence. This is a defining moment for those who had worked relentlessly on neural networks when entire machine learning community had moved away from it in 1970s.

2019

Evolution of Deep Learning Architectures

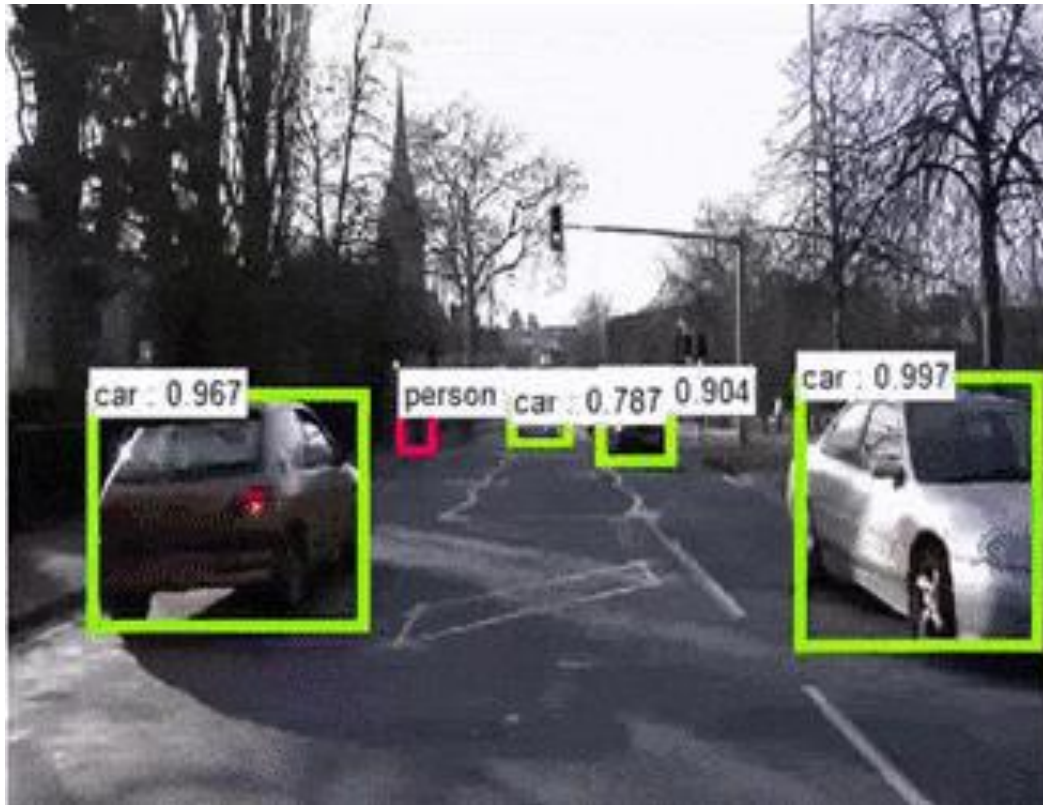


Object Detection Milestones



Object Detection

Deep Learning applications



Deep Learning Applications

Language
Modeling

Speech
Recognition

Machine
Translation

Machine
Transliteration

Conversational
modelling

Time Series
Forecasting.

Question
Answering

Object
Detection/
Recognition

Visual Tracking

Image
Captioning

Video
Captioning

Visual
Question
answering

Video Qs
Answering

Video
Summarisation

Generate data
sets

Deep Learning Models (Big Picture)

- **Deep Neural Networks**

- Neural Networks
- Deep Neural Networks/Feed forward Networks

- **Convolutional Neural Networks (CNN)**

- LeNet
- AlexNet
- VGG
- ResNet
- GoogleNet

- **Object Detection models**

- RCNN
- Yolo

- **Generative models**

- Generative Adversarial Networks,(GAN)

- **Sequential Models**

- RNN
- LSTM
- Encoder – Decoder models
- Attention mechanism
- **Transformers**
- **Transformers, Transformer-XL, Masked Language Modelling**
- **Generative Pre-trained Transformer 3 (GPT-3)**

- **Autoregressive models**

- Time series forecasting
- **Generative Pre-trained Transformer 3 (GPT-3)**
- **NADE, MADE(Masked Autoencoder Density Estimator), PixelRNN**

Namah Shivaya