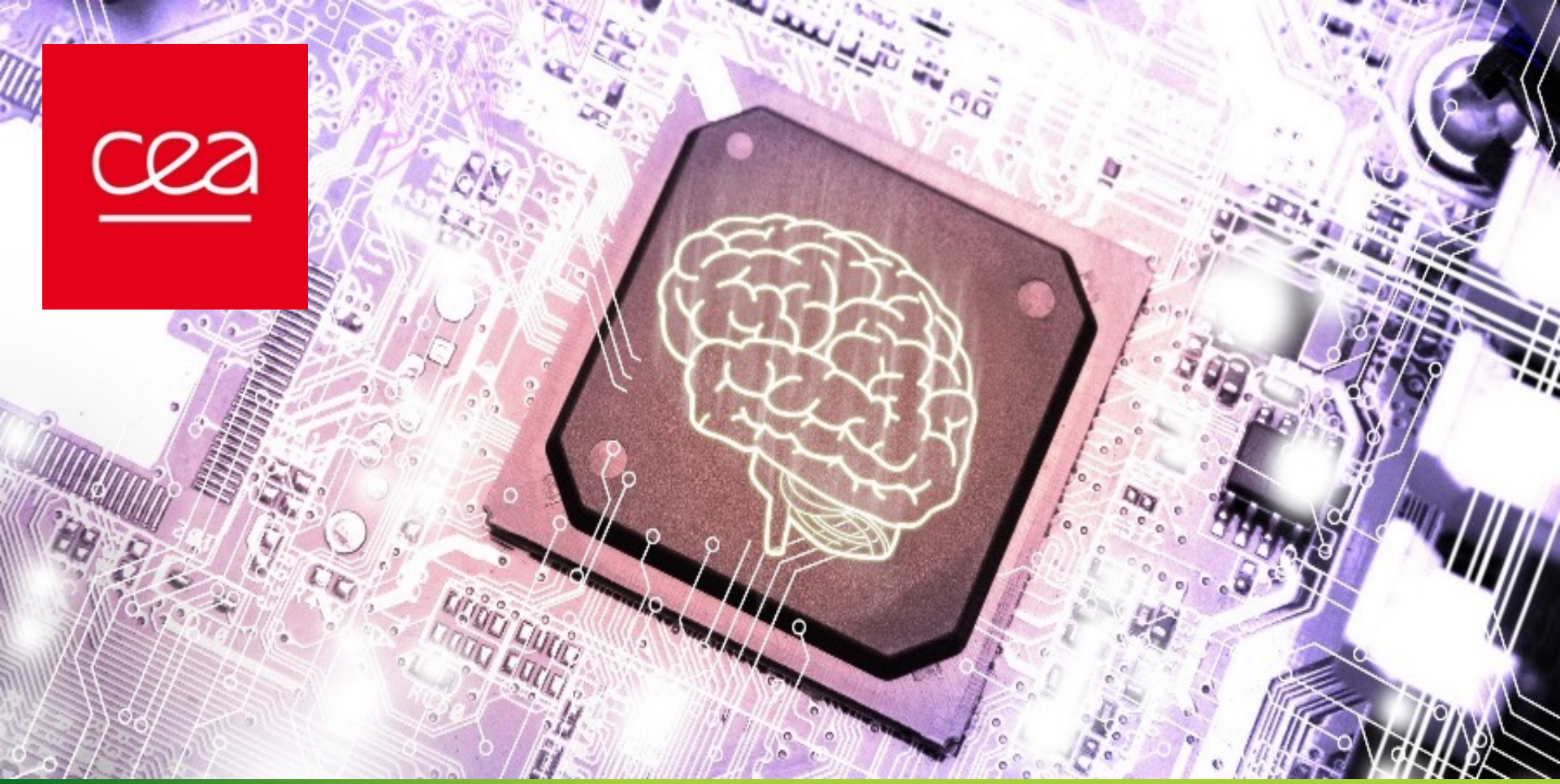


A SHORT STORY OR ARTIFICIAL INTELLIGENCE AND DEEP LEARNING

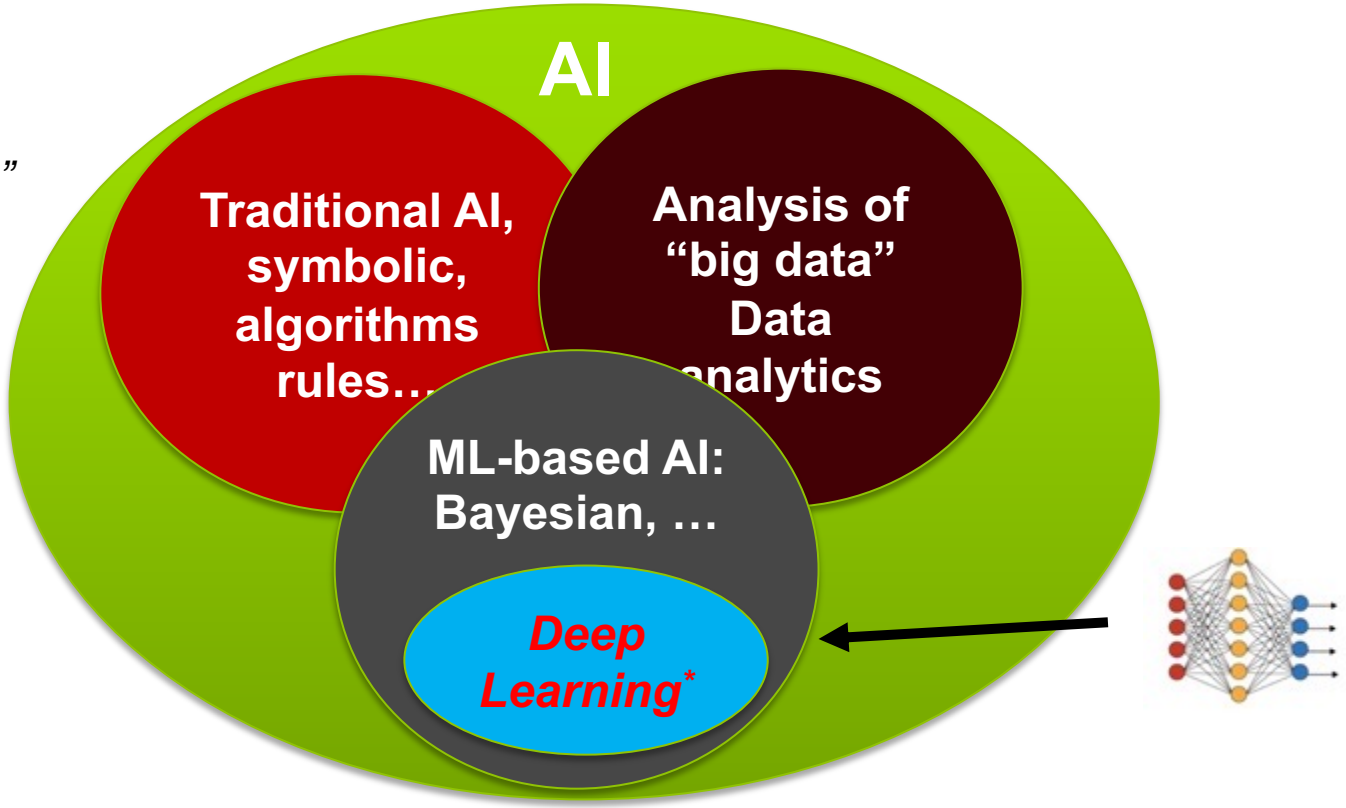
Marc Duranton
Commissariat à l'énergie atomique et aux énergies alternatives

October 9th, 2023

KEY ELEMENTS OF *ARTIFICIAL INTELLIGENCE*

"…as soon as it works, no one calls it AI anymore."

John McCarthy,
who coined the term "Artificial Intelligence" in 1956



*** Reinforcement Learning, One-shot Learning, Generative Adversarial Networks, Transformers, etc…**

From Greg. S. Corrado, Google brain team co-founder:

- *"Traditional AI systems are **programmed** to be clever*
- *Modern ML-based AI systems **learn** to be clever.*

335 BC: ARISTOTLE

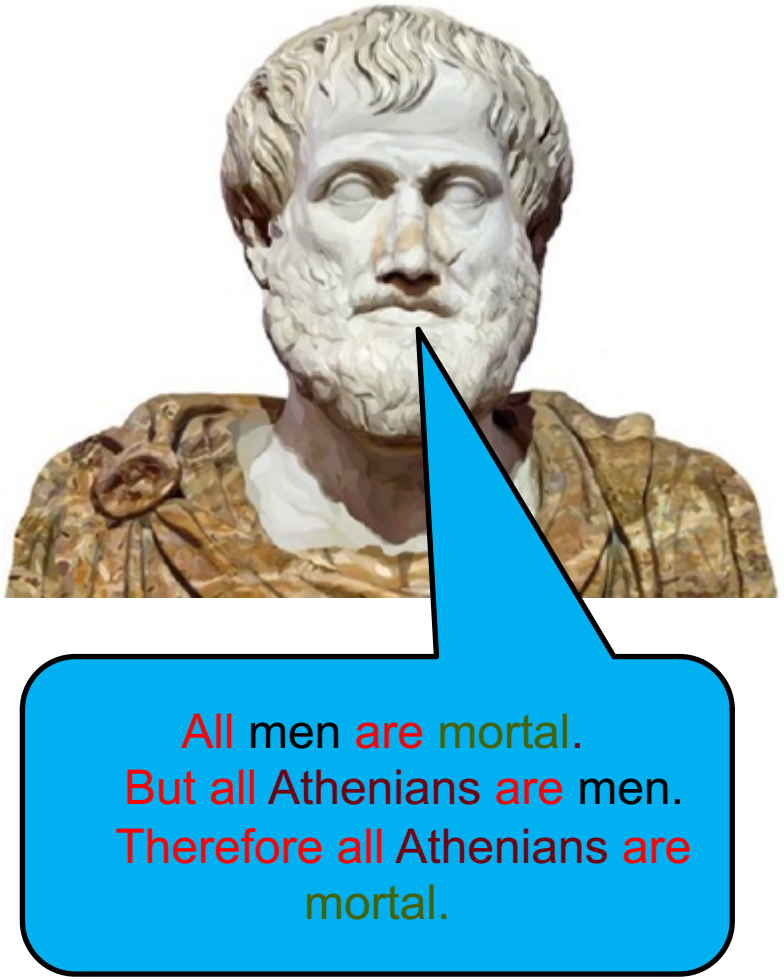
Aristotle invented the inductive reasoning system:

A formal method to represent how human reasons

But…

- He believed that the heart was the seat of behavior,
- He noted the importance of the brain…

…but for cooling the blood !

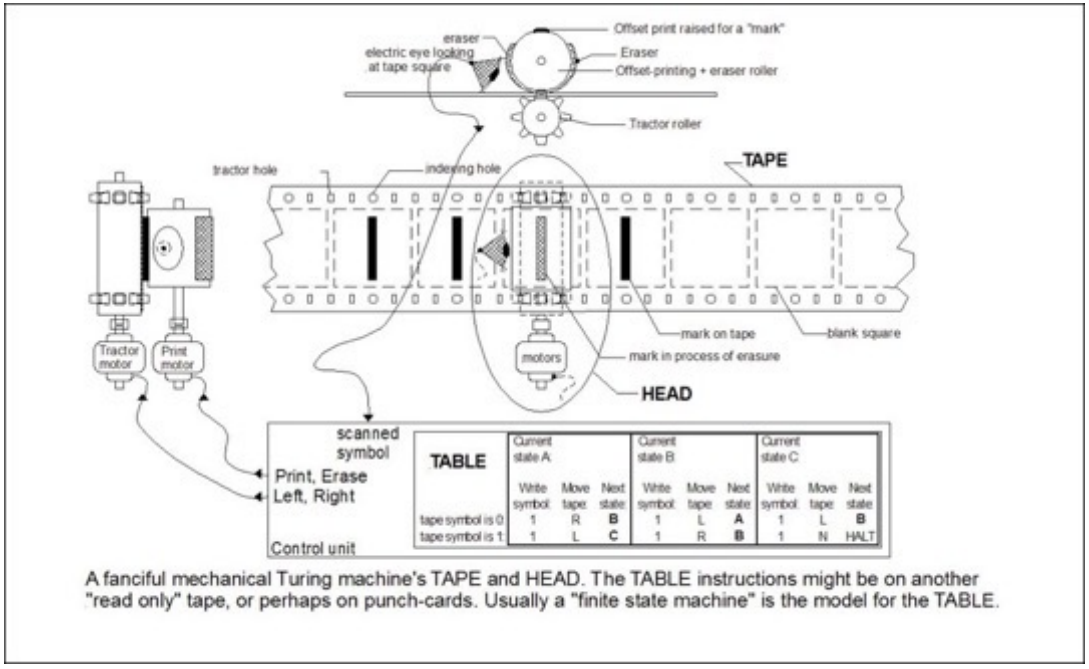


1936: TURING MACHINE

The Turing machine (a-machine - automatic machine) prove properties of computation in general—and in particular, the uncomputability of the Entscheidungsproblem ("decision problem") and prove fundamental limitations on the power of mechanical computation.

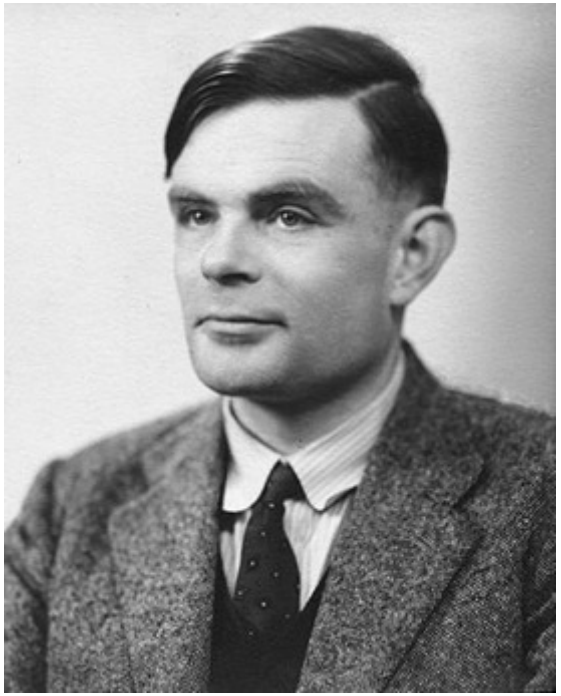
A programming language that is Turing complete is theoretically capable of expressing all tasks accomplishable by computers if the limitations of finite memory are ignored.

The "***halting problem***" is the problem of determining, from a description of an arbitrary computer program and an input, whether the program will finish running, or continue to run forever. Alan Turing proved in 1936 that a general algorithm to solve the halting problem for all possible program-input pairs cannot exist.

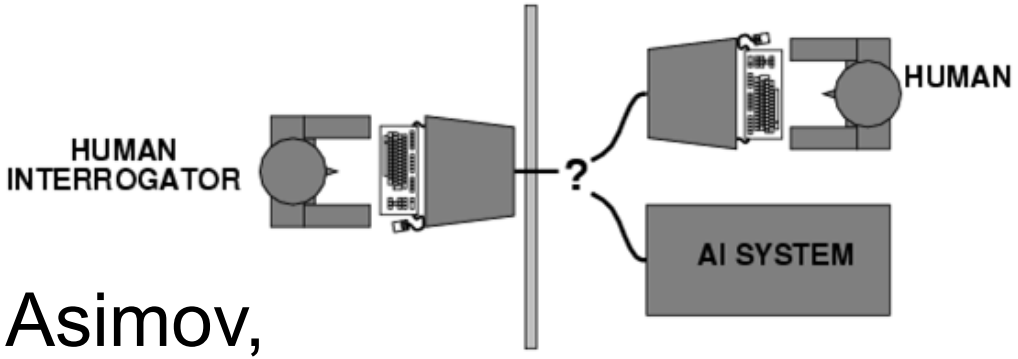


A fanciful mechanical Turing machine's TAPE and HEAD. The TABLE instructions might be on another "read only" tape, or perhaps on punch-cards. Usually a "finite state machine" is the model for the TABLE.

1942: ALAN TURING

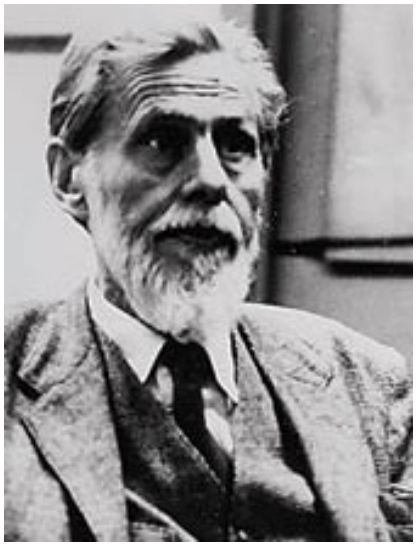


1942: Any form of mathematical reasoning can be made by a machine.
1950: He invented the “Turing test” to check if a system is “intelligent”, i.e. undisguisable from a human

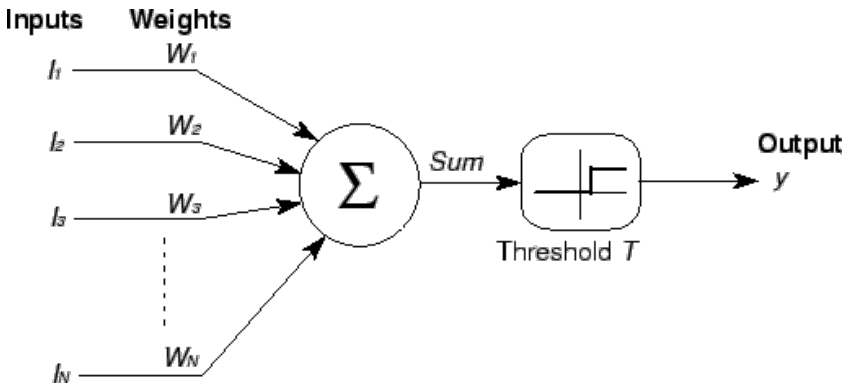


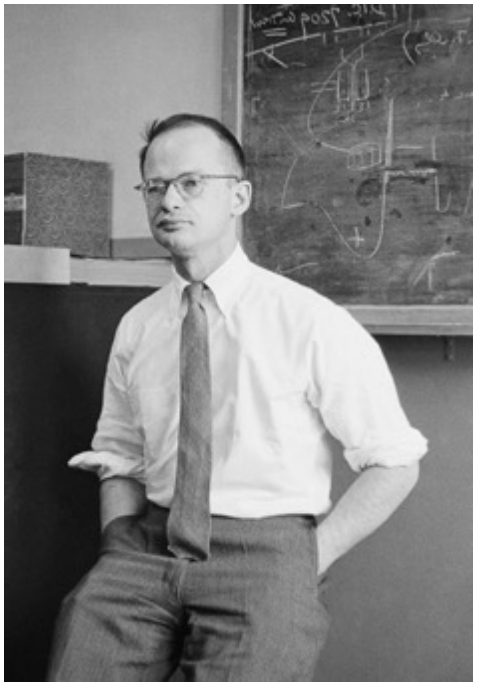
The same year, Isaac Asimov, invented the 3 (4) laws of robotics

1943: MCCULLOCH AND PITTS



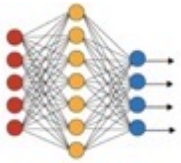
Neurophysiologist and cybernetician





Logician workingin the field of computational neuroscience

They laid the foundations of formal Neural Networks



1943: MCCULLOCH AND PITTS

BULLETIN OF
MATHEMATICAL BIOPHYSICS
VOLUME 5, 1943

A LOGICAL CALCULUS OF THE IDEAS IMMANENT IN NERVOUS ACTIVITY

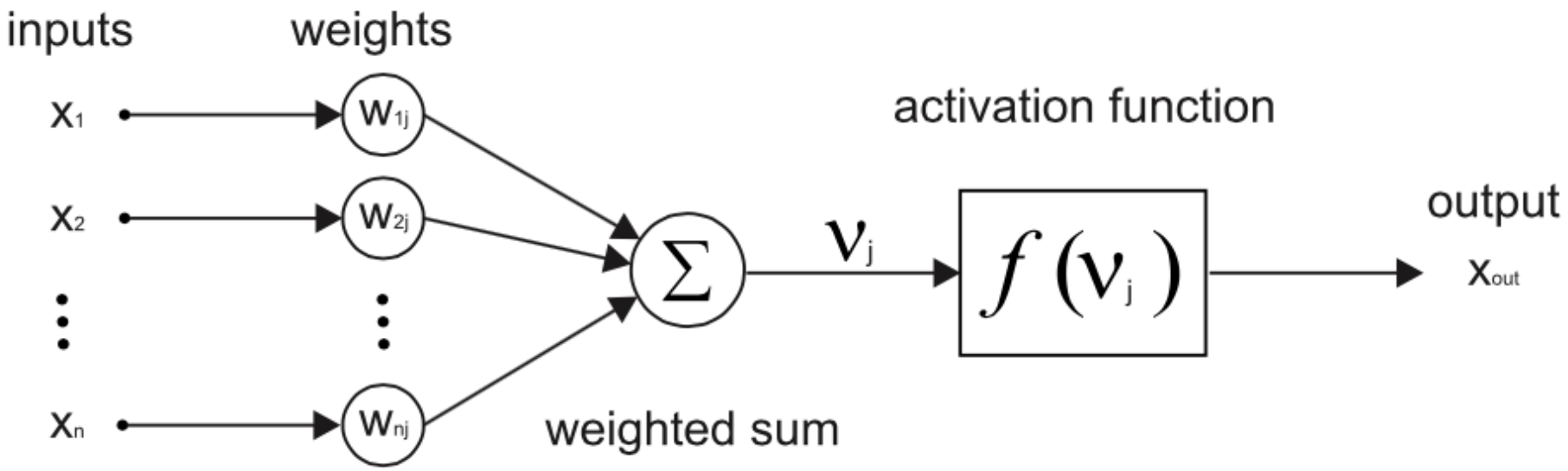
WARREN S. MCCULLOCH AND WALTER PITTS

FROM THE UNIVERSITY OF ILLINOIS, COLLEGE OF MEDICINE,
DEPARTMENT OF PSYCHIATRY AT THE ILLINOIS NEUROPSYCHIATRIC INSTITUTE,
AND THE UNIVERSITY OF CHICAGO

Because of the "all-or-none" character of nervous activity, neural events and the relations among them can be treated by means of propositional logic. It is found that the behavior of every net can be described in these terms, with the addition of more complicated logical means for nets containing circles; and that for any logical expression satisfying certain conditions, one can find a net behaving in the fashion it describes. It is shown that many particular choices among possible neurophysiological assumptions are equivalent, in the sense that for every net behaving under one assumption, there exists another net which behaves under the other and gives the same results, although perhaps not in the same time. Various applications of the calculus are discussed.

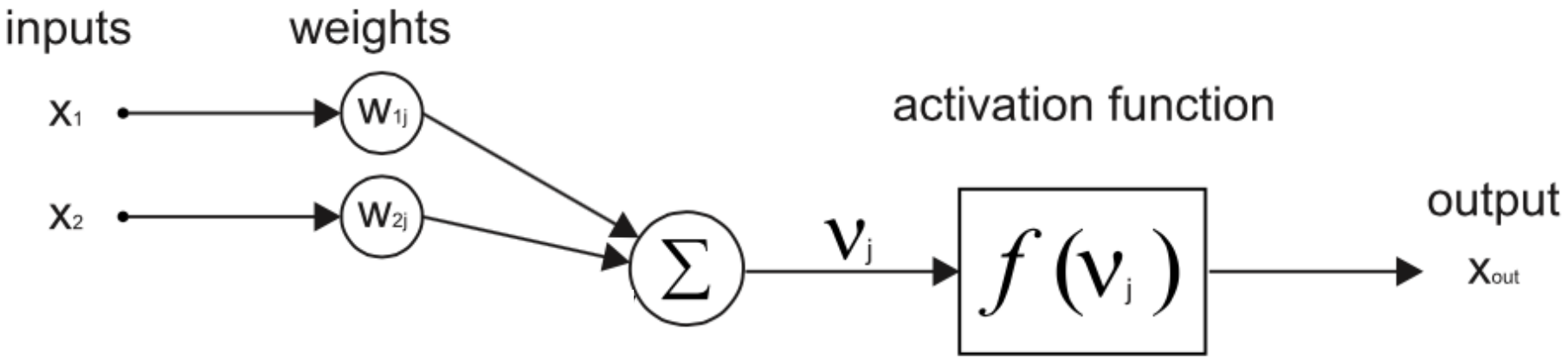
WHAT IS A NEURAL NETWORK?

A « formal » neuron:



WHAT IS A NEURAL NETWORK?

The « formal » neuron:



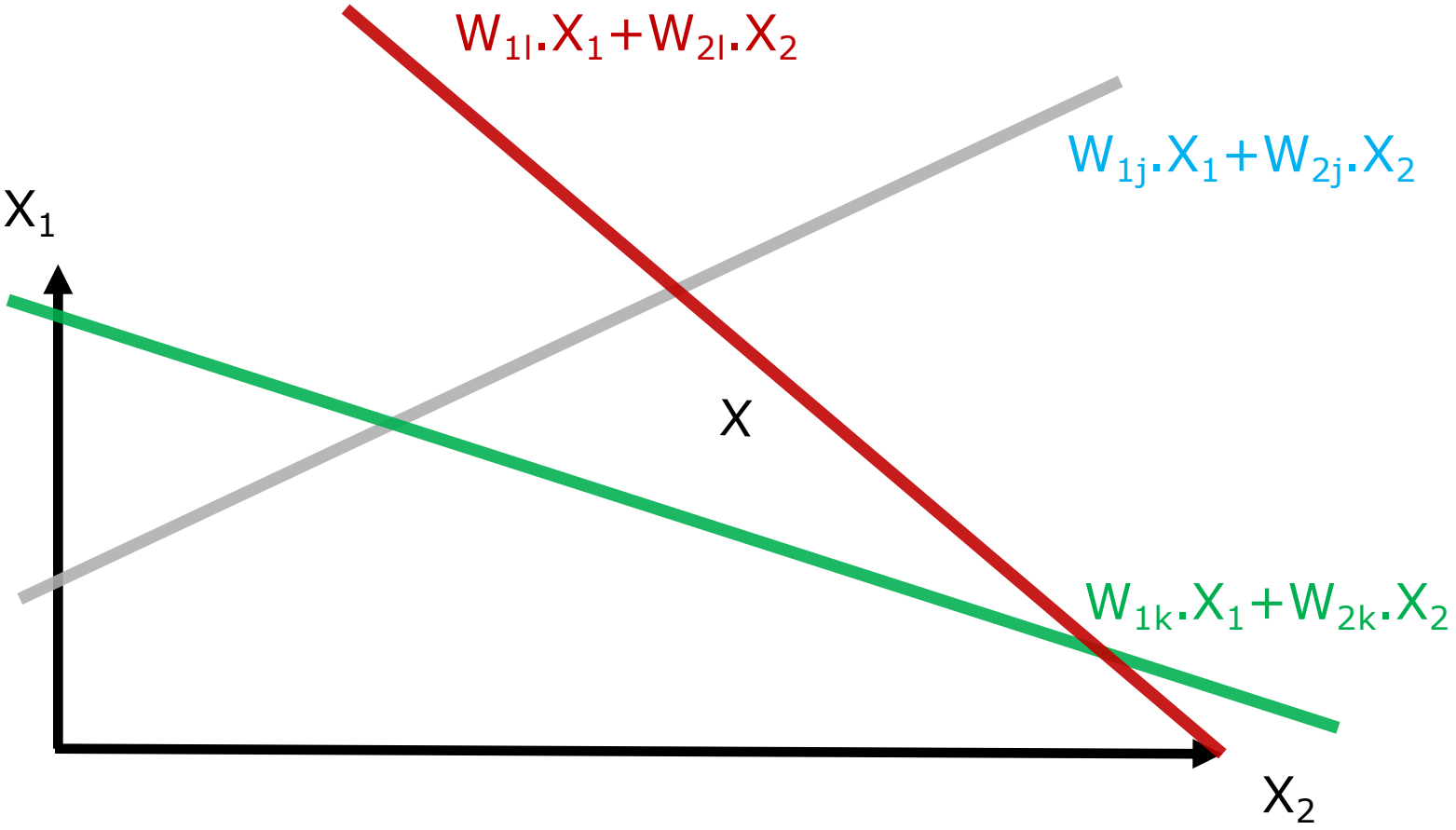
$V_j = W_{1j}.X_1 + W_{2j}.X_2$

It is the definition of an hyperplane

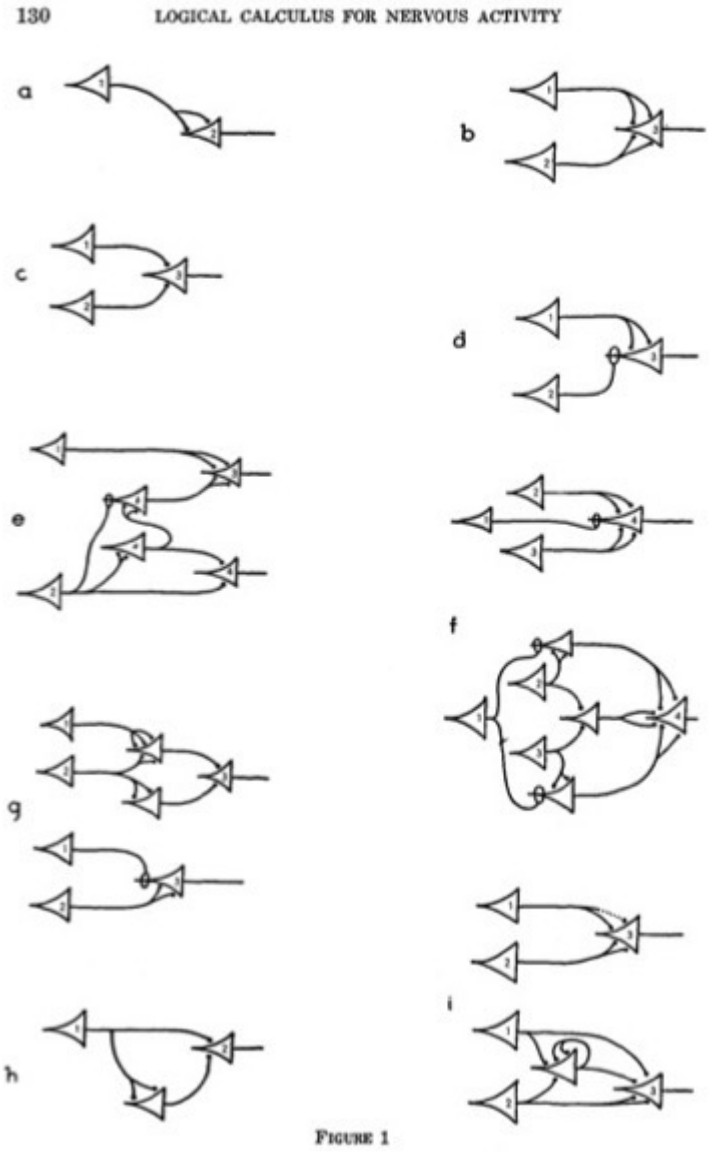
$F(V_j)$ non linear $\in\{-1,1\}$ e.g. sign() function

$X(X_1,X_2)$ is “above” or “below” the hyperplane

WHAT IS A NEURAL NETWORK?



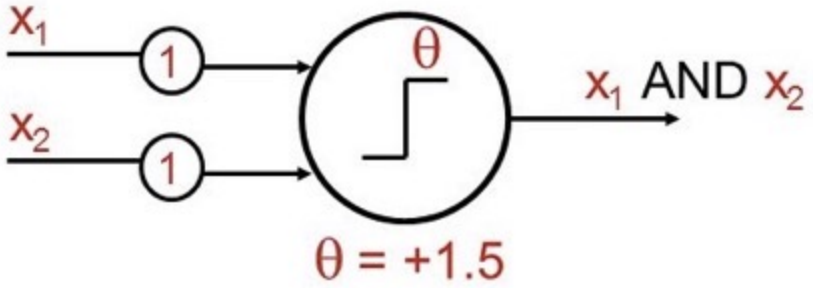
WHAT IS A NEURAL NETWORK?



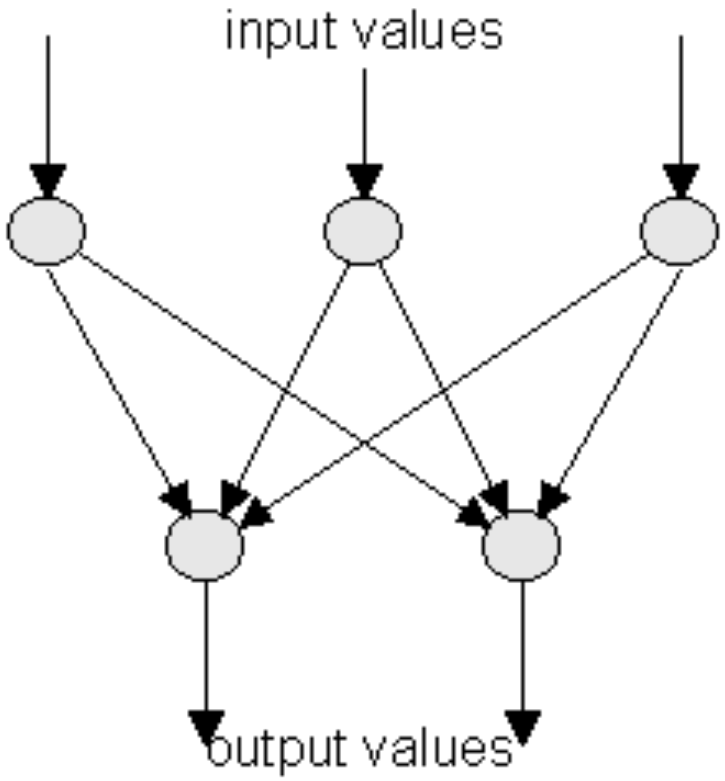
Association of neurons to make logical functions.

Example: AND gate

IN 1	IN 2	OUT
0	0	0
0	1	0
1	0	0
1	1	1



MULTILAYER NETWORK

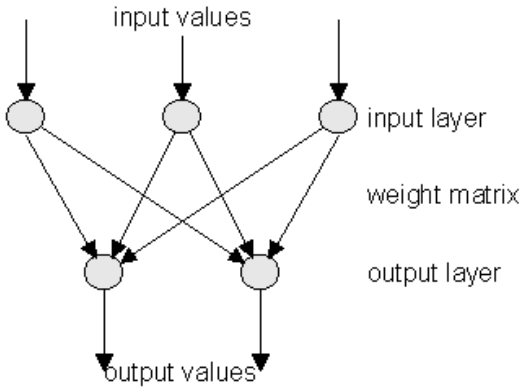


Hyperplane separation

"logic" composition
Warren McCulloch and Walter Pitts, 1943

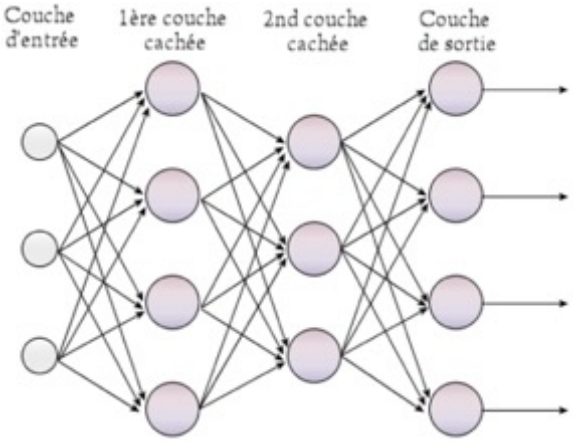
= ***universal approximator***

TOPOLOGY OF NEURAL NETWORKS

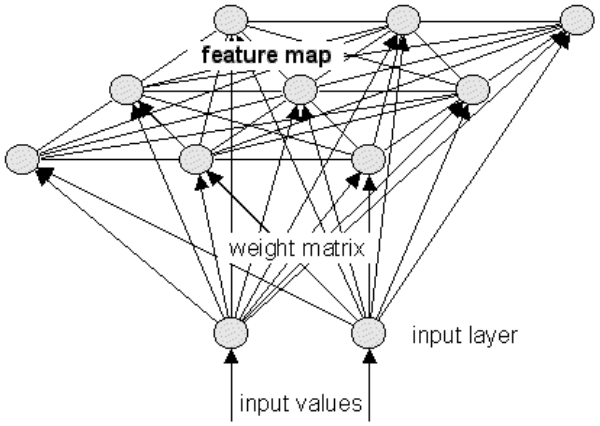


Perceptron

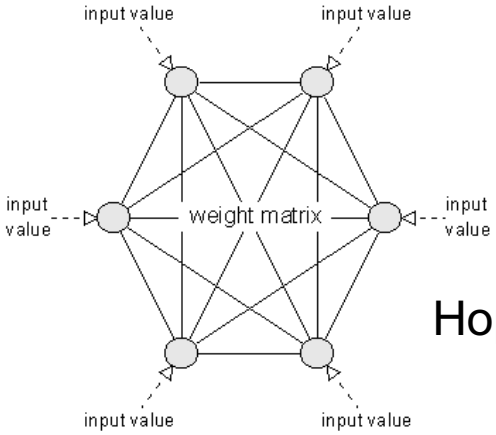
Rosenblatt -- 1957-58



Multi-layer Perceptron



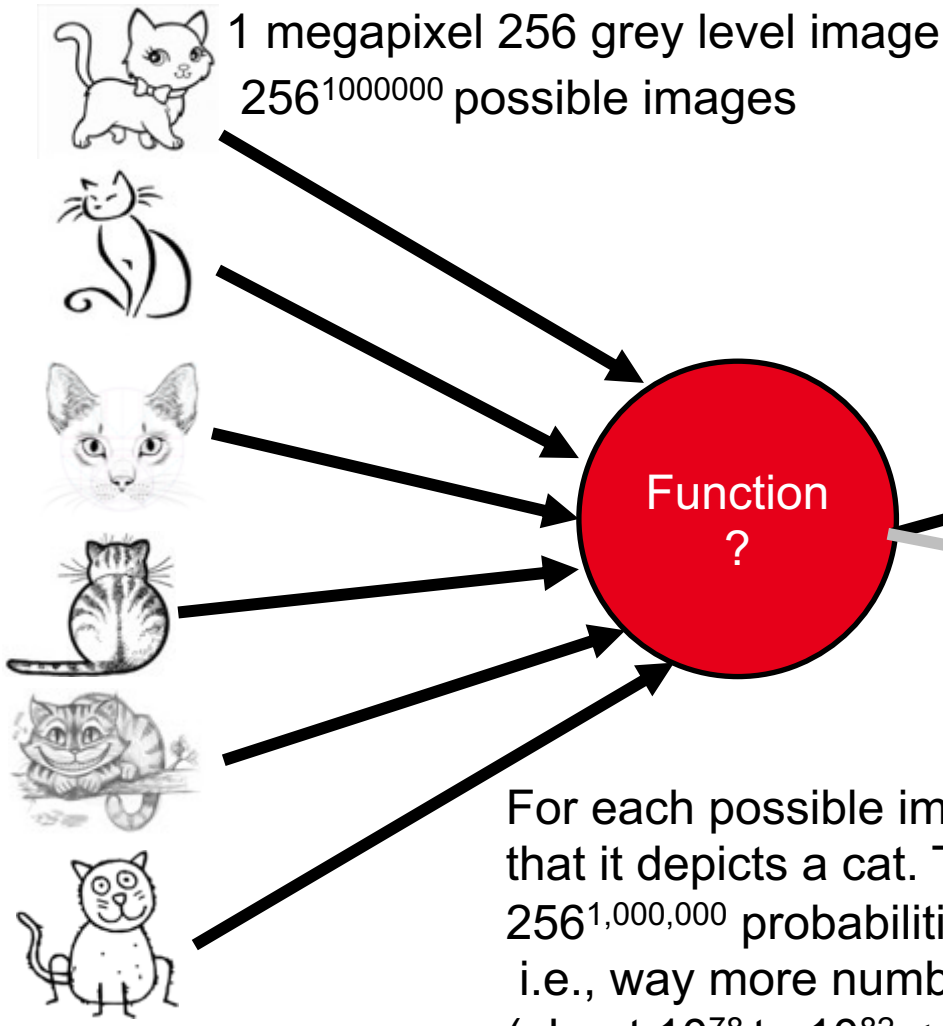
Kohonen Self-Organizing Maps



Hopfield

And more…

WHY DOES DEEP LEARNING WORK SO WELL?*

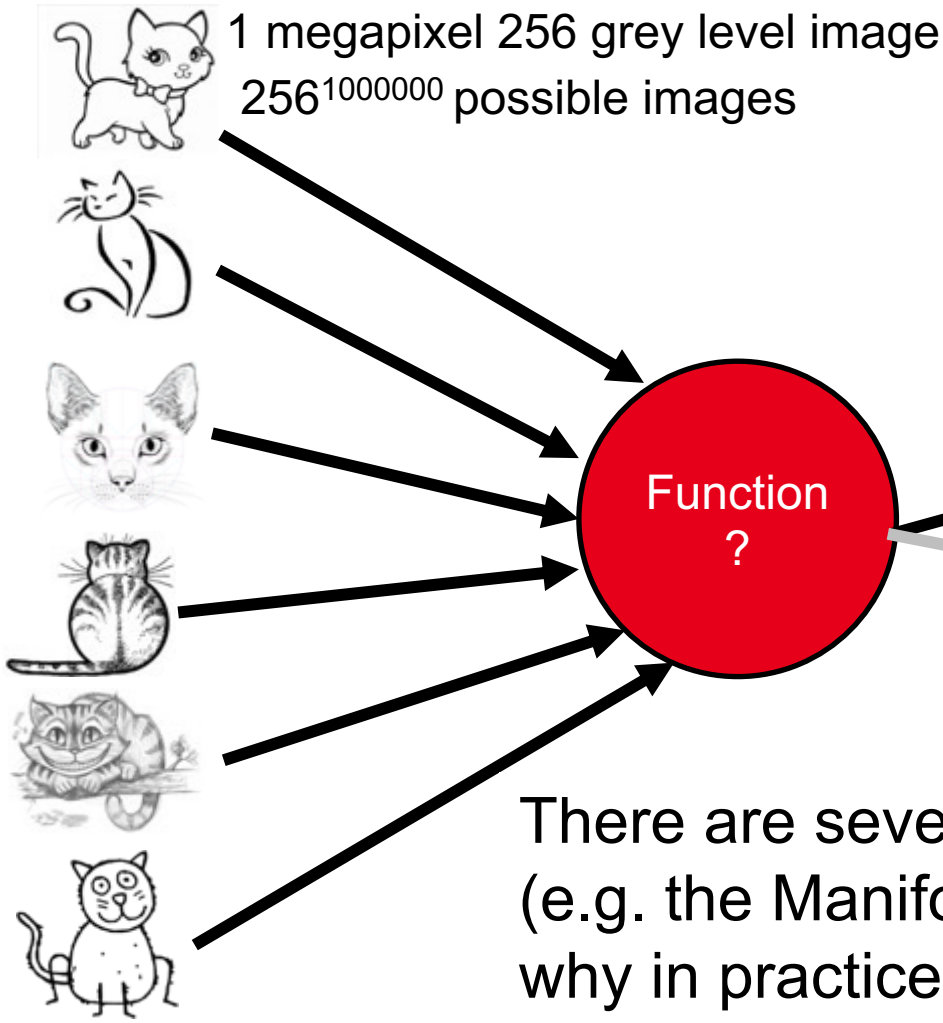


It can be done by Neural Networks:
Universal approximator *made with neural networks of* ***finite*** *size*

For each possible image, we wish to compute the probability that it depicts a cat. Then, the function is defined by a list of $256^{1,000,000}$ probabilities
i.e., way more numbers than there are atoms in our universe (about 10^{78} to 10^{82} <<< $10^{2,408,240}$).

- Work of Henry W. Lin (Harward) , Max Tegmark (MIT), and David Rolnick (MIT) https://arxiv.org/abs/1608.08225

WHY DOES DEEP LEARNING WORK SO WELL?*



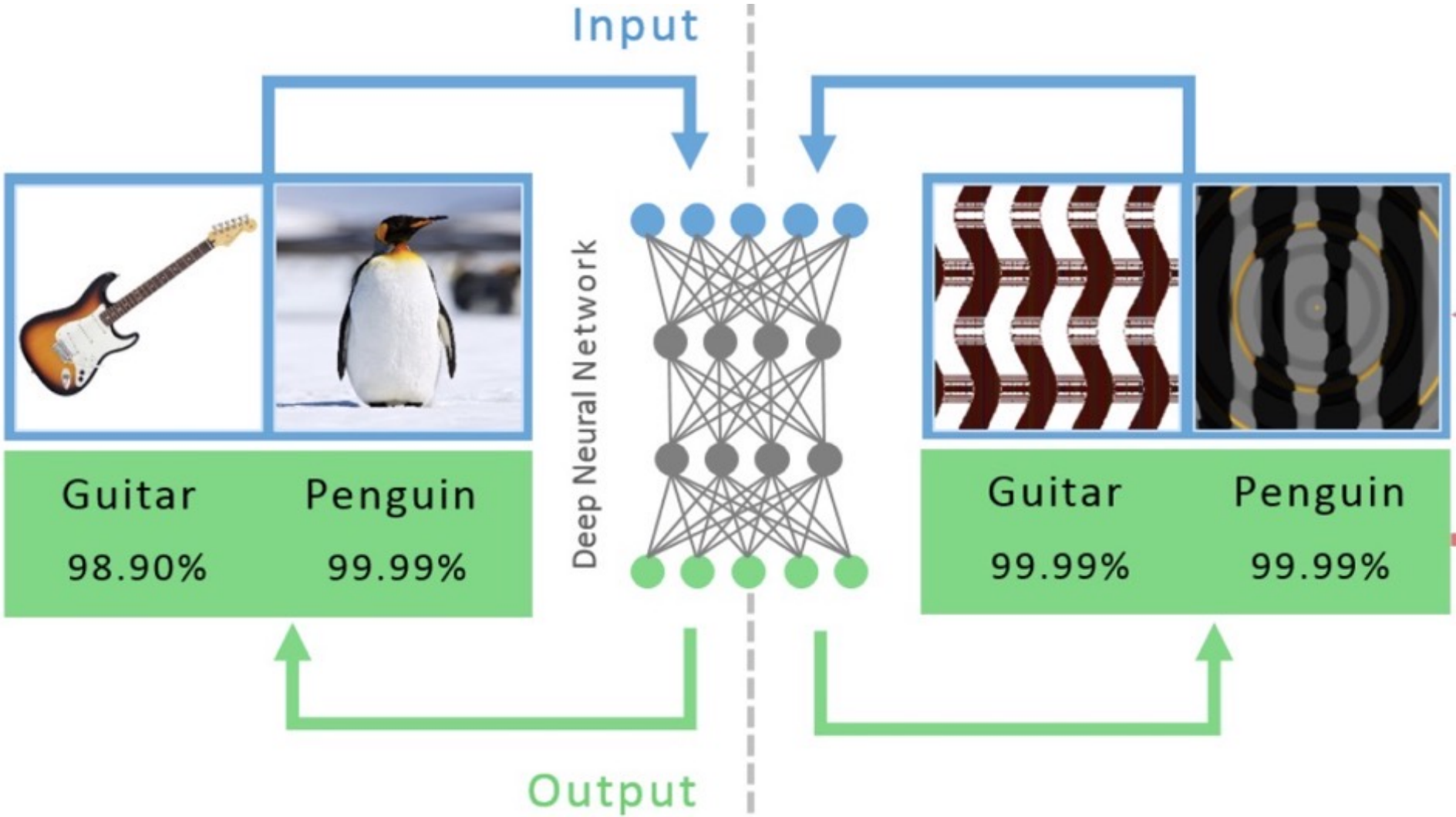
It can be done by Neural Networks:
Universal approximator *made with neural networks of* ***finite*** *size*

There are several hypotheses (e.g. the Manifold Hypothesis, …) why in practice the neural network ***is very small***

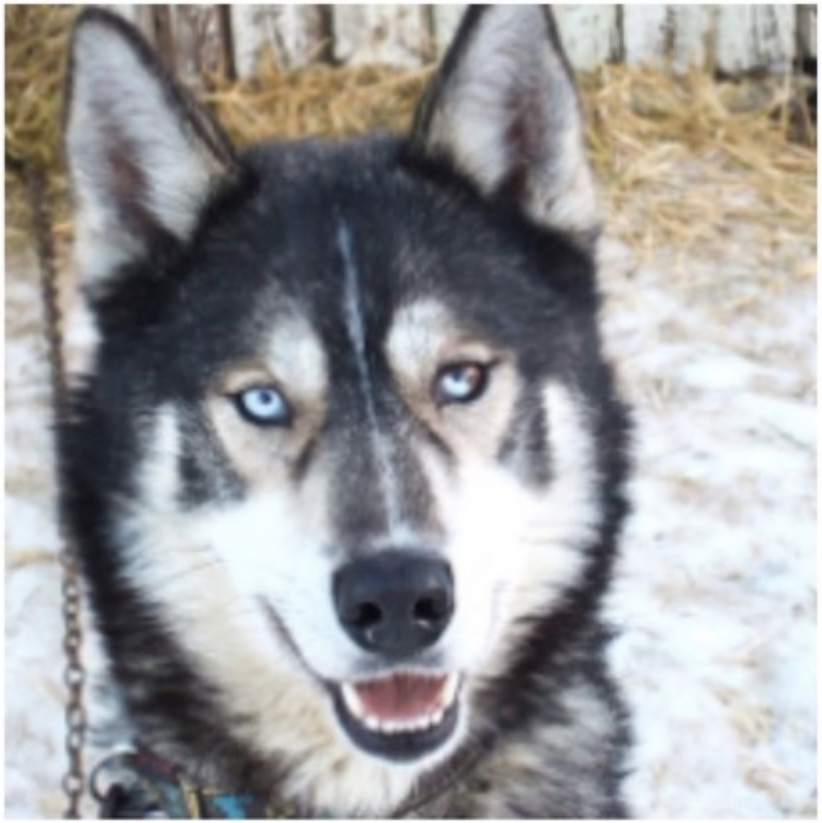
- Work of Henry W. Lin (Harward) , Max Tegmark (MIT), and David Rolnick (MIT) https://arxiv.org/abs/1608.08225

WHY DOES DEEP LEARNING WORK SO WELL?

OR NOT….



DEEP LEARNING SHOULD LEARN THE RIGHT THINGS

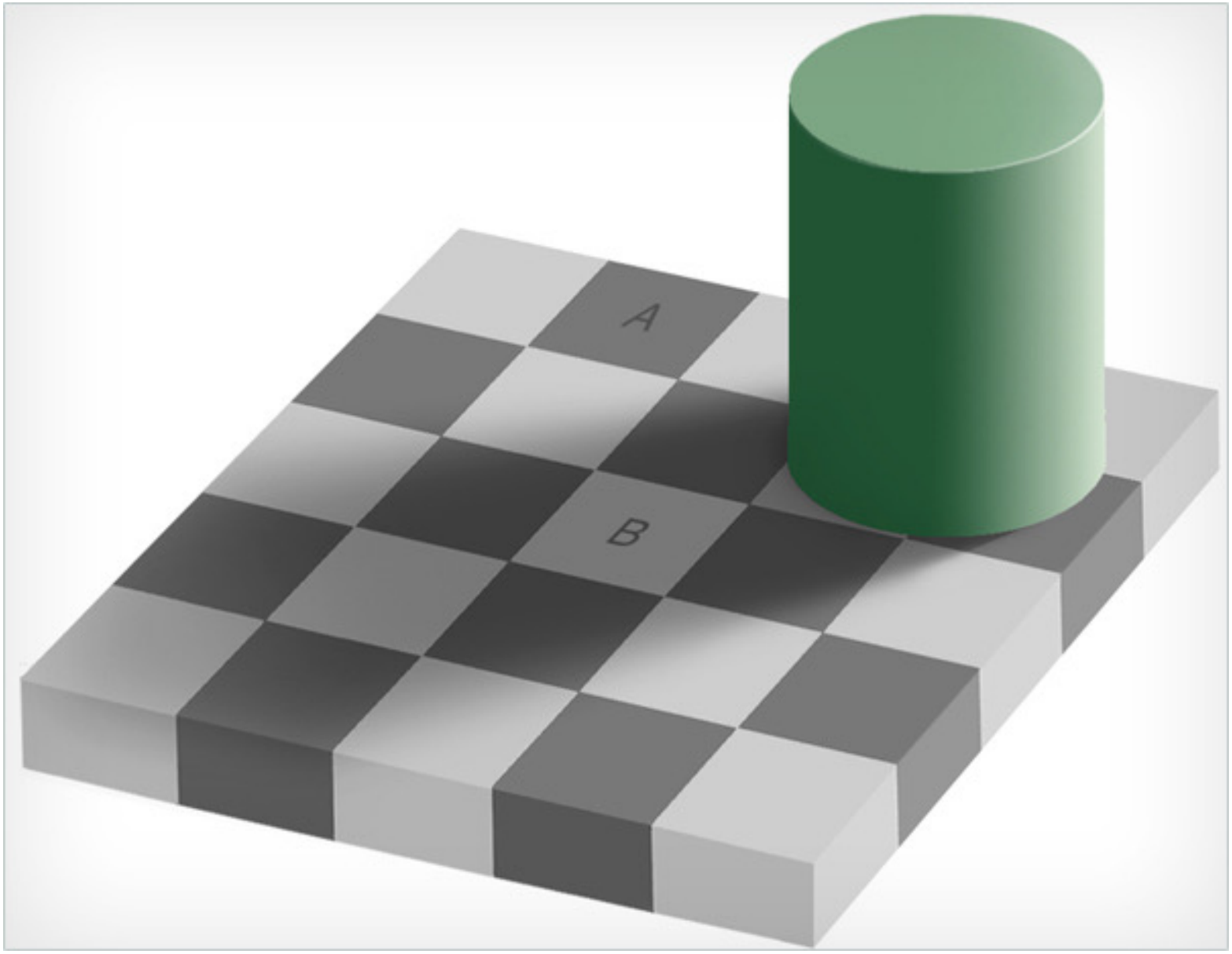


(a) Husky classified as wolf

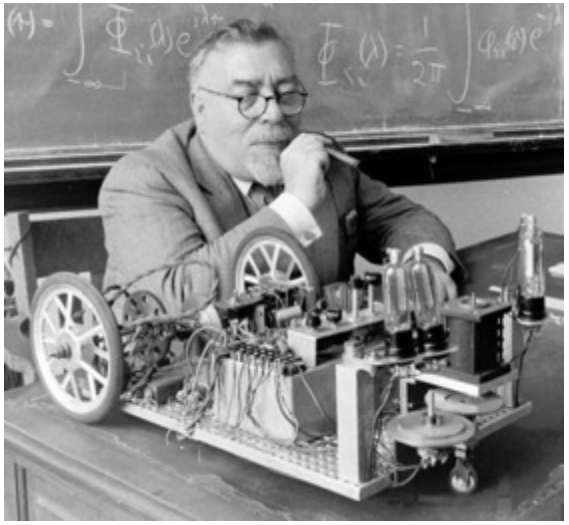
(b) Explanation

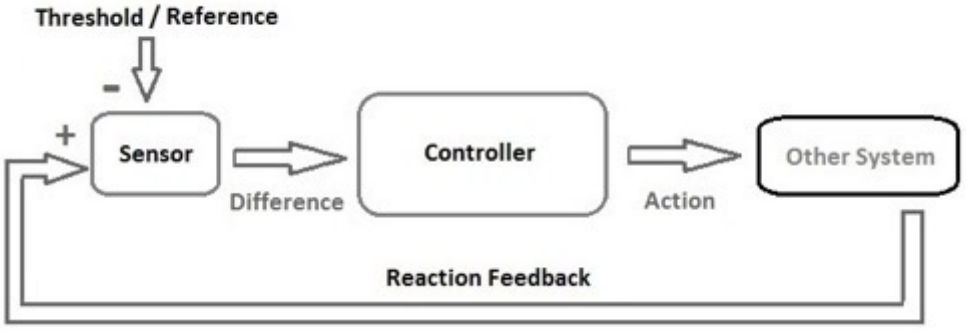
From "Why Should I Trust You?": Explaining the Predictions of Any Classifier", Tulio Ribeiro, Marco; Singh, Sameer; Guestrin, Carlos, arXiv:1602.04938, 02/2016.

BUT OUR BRAIN DOES NOT ALWAYS WORK

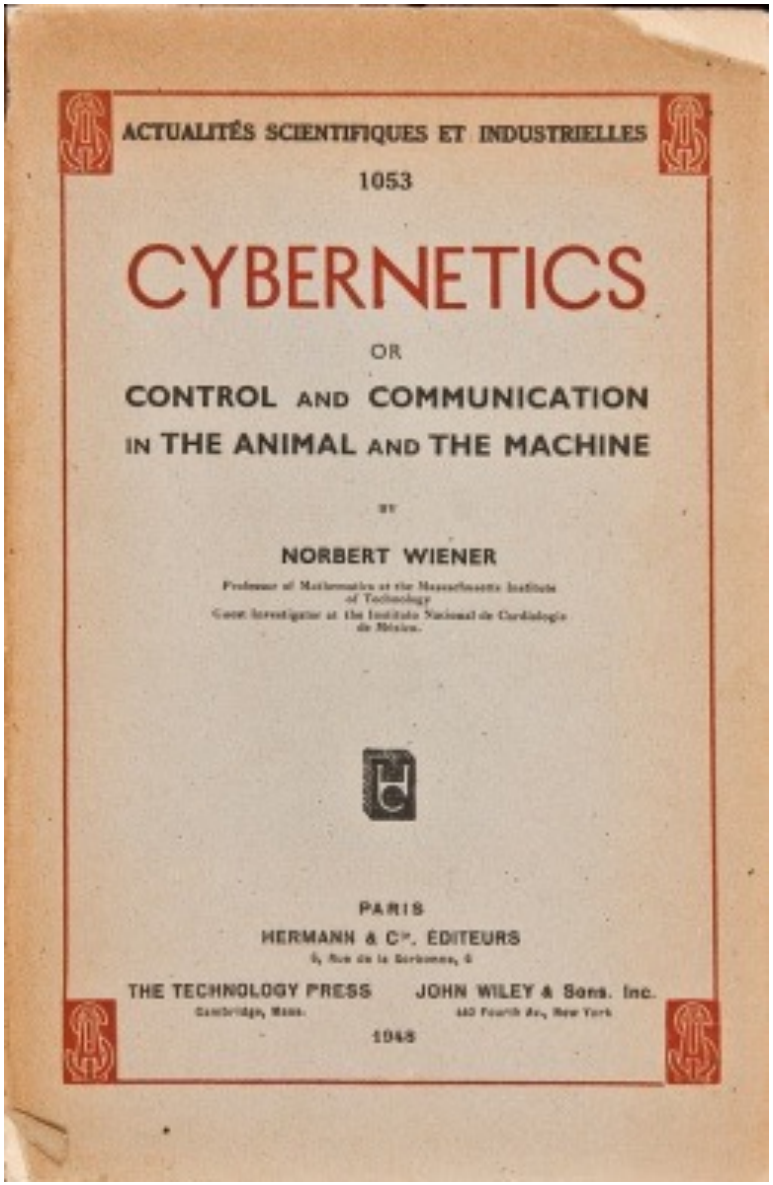


1948: NORBERT WIENER





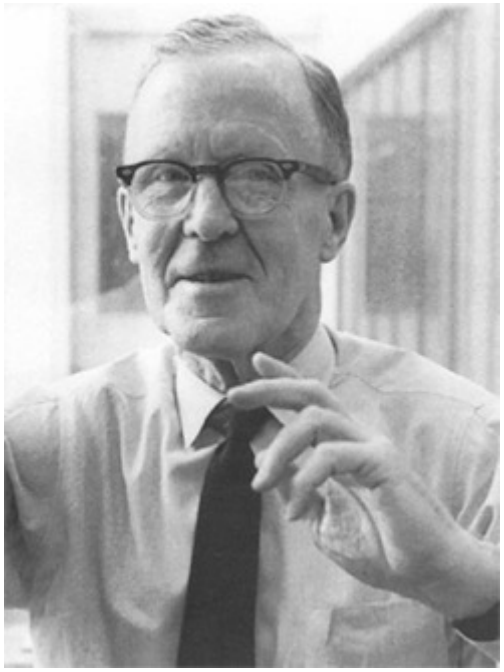
A Cybernetic Loop



1949: DONALD HEBB

Hebb's rule or Hebbian theory: an explanation for the adaptation of neurons in the brain during the learning process

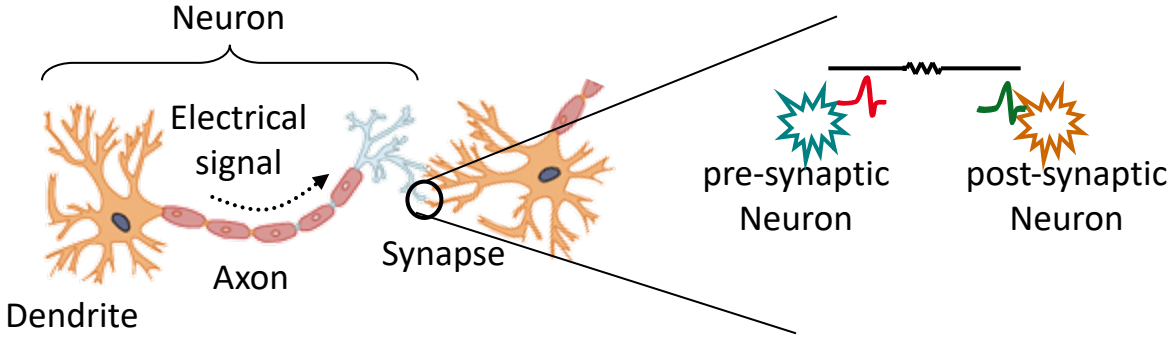
Basic mechanism for synaptic plasticity: an increase in synaptic efficacy arises from the presynaptic cell's repeated and persistent stimulation of the postsynaptic cell.

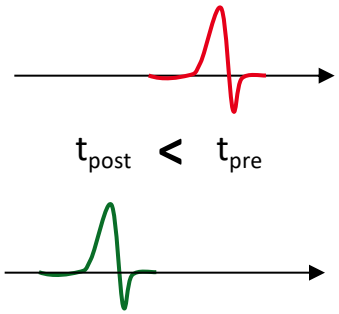


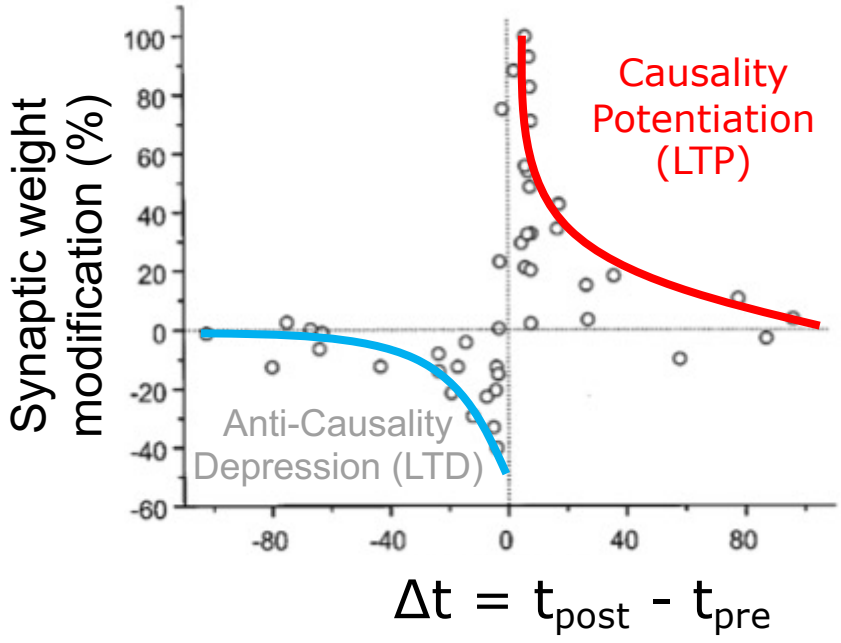
Psychologist, working in the area of neuropsychology

Introduced by Donald Hebb in his 1949 book « *The Organization of Behavior* »

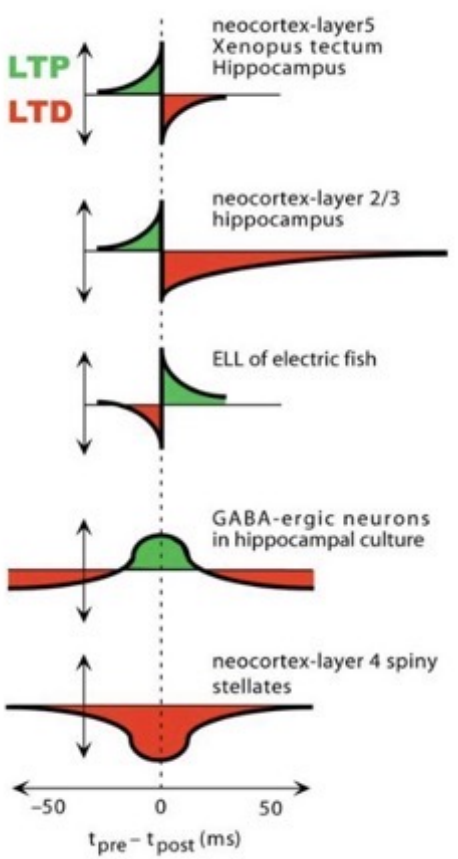
DERIVED FROM HEBB'S RULE: STDP (SPIKE TIMING DEPENDENT PLASTICITY)







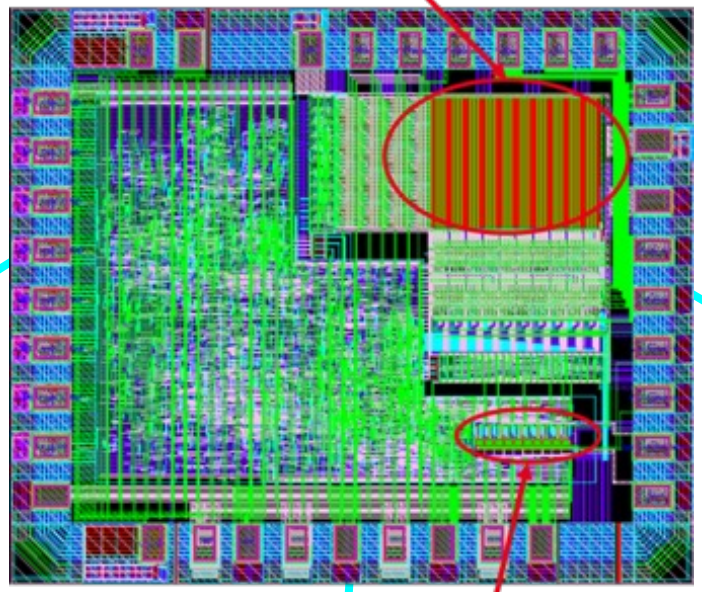
STDP = correlation detector



SIDE REMARK: INVESTIGATION OF RRAM AS SYNAPSES UNSUPERVISED LEARNING (INFORMATION CODED BY SPIKES)

Analog computing: using physical phenomenon to make computations

OxRAMs

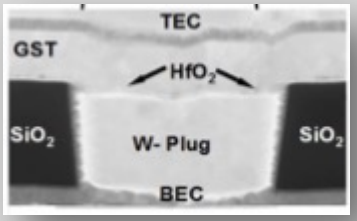


Thermal effect

Electrochemical effect

PCM

GST
GeTe
GST + HfO_2



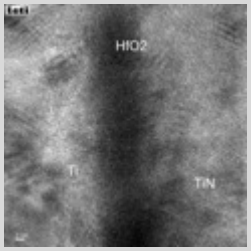
M.Suri, et. al, IEDM 2011
M.Suri, et. al, IMW 2012 , JAP 2012
O.Bichler et al. IEEE TED 2012
M.Suri et al., EPCOS 2013
D.Garbin et al., IEEE Nano 2013

Electronic effect
oxygen vacancies

Neurons

OXRAM

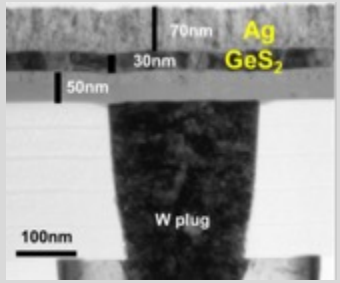
$TiN/HfO_2/Ti/TiN$



D.Garbin et al. IEDM 2014
D.Garbin et al., IEEE TED 2015

CBRAM

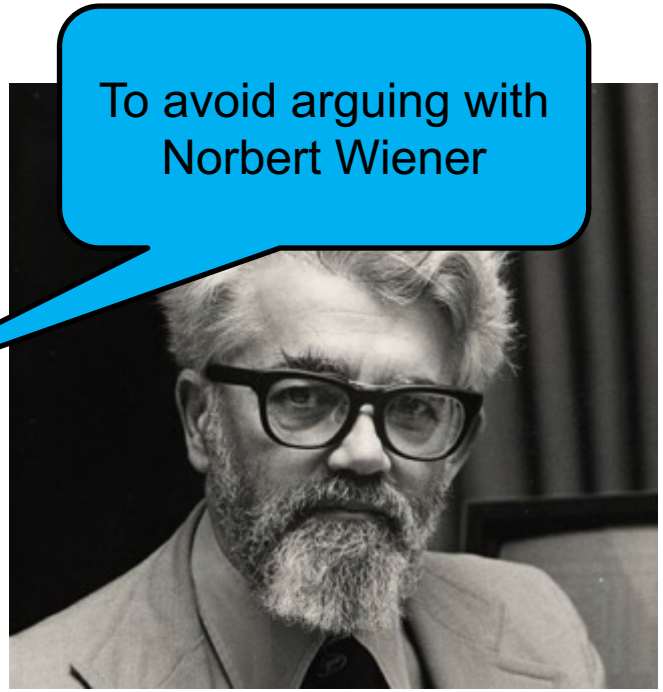
Ag / GeS_2



Leading to ***neuromorphic*** chips

1955: JOHN MCCARTHY

John McCarthy is one of the "founding fathers" of artificial intelligence, together with Marvin Minsky, Allen Newell and Herbert A. Simon.



McCarthy coined the term "artificial intelligence" in 1955, and organized the famous **Dartmouth Conference** in Summer 1956. This conference started AI as a science field.

While at MIT, McCarthy developed the programming language ***LISP*** in 1950, one of the two oldest programming language

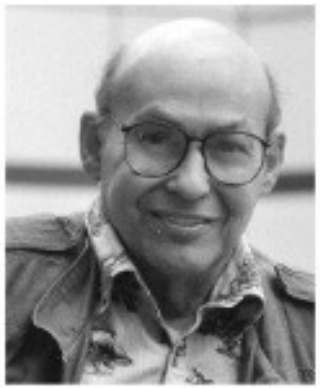
```
(defun factorial (n)
  (if (= n 0)
      1
      (* n (factorial (- n 1))) ) )
```

Recursive definition of a factorial

1956: DARTMOUTH CONFERENCE

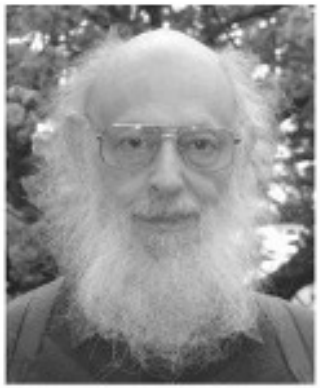
The Founding Fathers of AI

John MacCarthy



Marvin Minsky

Claude Shannon

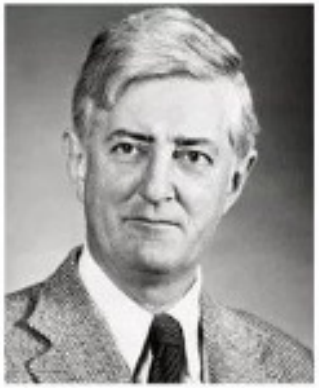


Ray Solomonoff

Alan Newell

Herbert Simon

Arthur Samuel



Oliver Selfridge

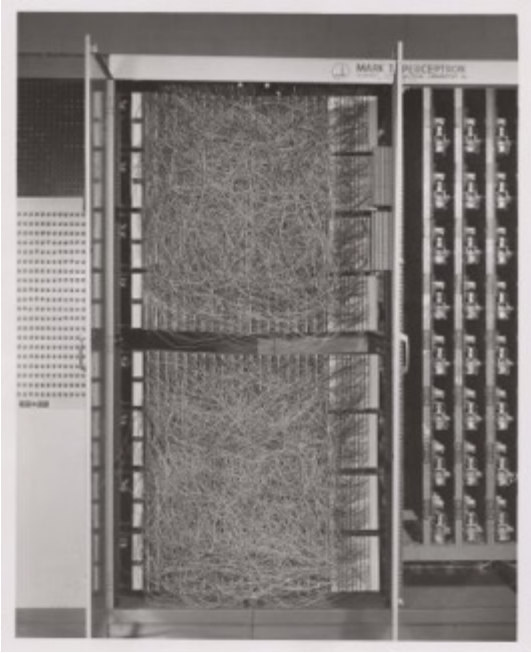
Nathaniel Rochester

Trenchard More

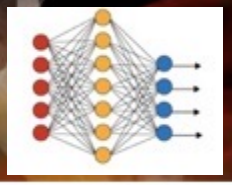
1957: THE PERCEPTRON AND F. ROSENBLATT

The perceptron algorithm was invented in 1957 at the Cornell Aeronautical Laboratory by Frank Rosenblatt.

The perceptron was intended to be a machine, rather than a program, and while its first implementation was in software for the IBM 704, it was subsequently implemented in custom-built hardware as the "Mark 1 perceptron". This machine was designed for image recognition: it had an array of 400 photocells, randomly connected to the "neurons". Weights were encoded in potentiometers, and weight updates during learning were performed by electric motors.



The Perceptron Learning Algorithm

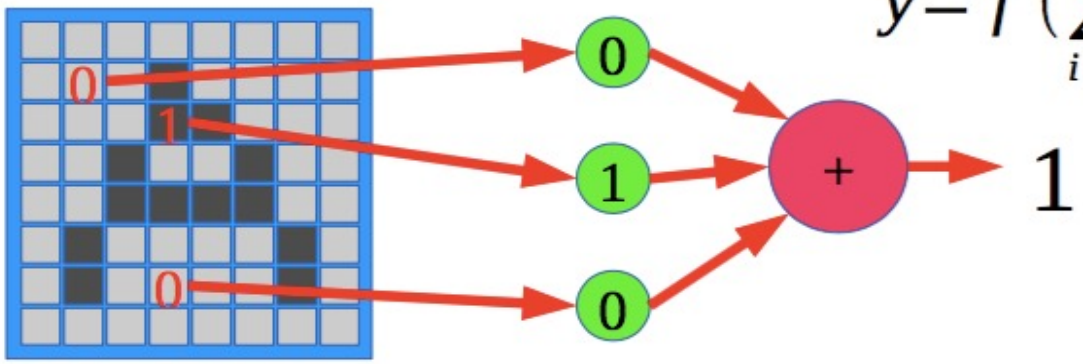


https://www.college-de-france.fr/site/yann-lecun/course-2015-2016.htm

- Training set: $(X^1,Y^1),(X^2,Y^2),.....,(X^p,Y^p)$
- Take one sample (X^k,Y^k), if the desired output is +1 but the actual output is -1
  - Increase the weights whose input is positive
  - Decrease the weights whose input is negative
- If the desired is -1 and actual is +1, do the converse.
- If desired and actual are equal, do nothing

$$w_i(t+1) = w_i(t) + (y_i^p - f(W'X^p))x_i^p$$

1986: David E. Rumelhart, Geoffrey E. Hinton and Ronald J. Williams

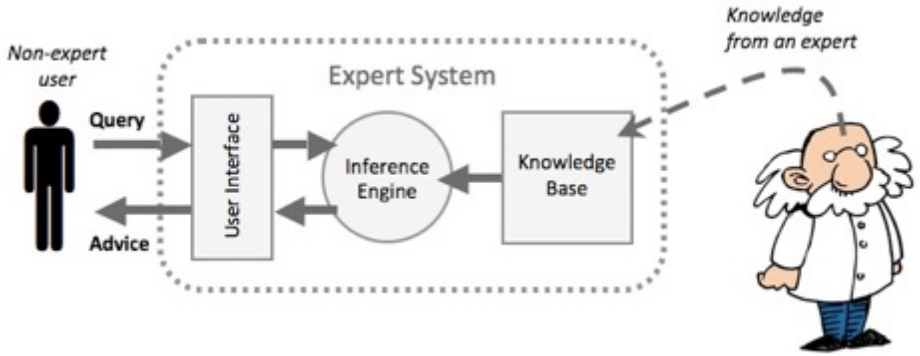


$$y = f\left(\sum_{i=1} w_i x_i + w_0\right) = f\left(W'X\right)$$

Supervised Learning

1965: EXPERT SYSTEMS

Expert systems were introduced by the Stanford Heuristic Programming Project led by Edward Feigenbaum,
Can also use predicate logic or even Fuzzy Logic

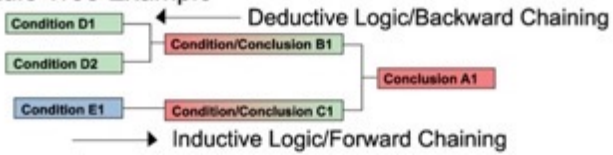


A) Rulebase Example

IF
- "yes" is equal to uniform_layer_flow
- THETA is greater than 45.0
- THETA is less than or equal to 90.0
- C4 is greater than (Lm/(0.8*(Hs-H0)))
- C6 is greater than (Lb/(0.8*(Hs-H0)))
- C9 is less than or equal to (Lt/(0.8^(Hs-H0)))

Then
- flow_type_ok is confirmed
- "V2" is asssigned to flow_type
- "No" is assigned to wake_attachment
- *Find* coanda_attachment_value

B) Rule Tree Example



A Lisp machine

Decoupling:
- The (inference) engine
- The knowledge base

1966: ELIZA THE CHAT BOT !

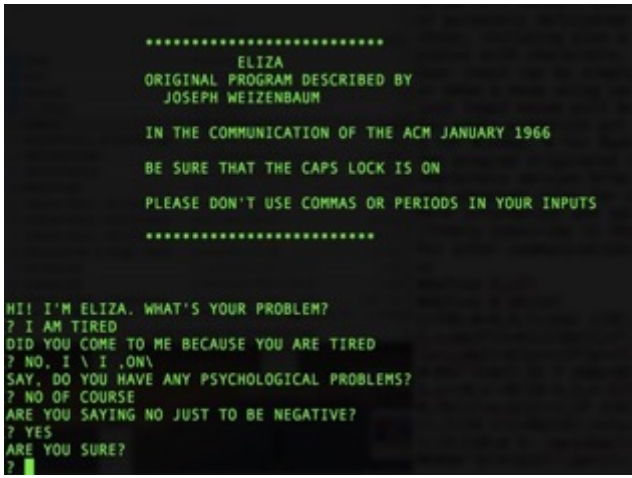
```
;;; /DOCFNS/   31 JULY 1969  1007:42

(PROGN (PRIN1 (QUOTE FILE" CREATED ")
              T)
       (PRIN1 (QUOTE 08/22/68" 1522:26")
              T)
       (TERPRI T))
(DEFINEQ

(DOCTOR
  (LAMBDA NIL
    (PROG (SENTENCE KEYSTACK MEMSTACK TIMON)
;;          (SETSEPR 109 106 0)
;;          (SETBRK 14 12 31 1 13 8 9 27 26 3)
;;          (CONTROL T)
;;          (GCGAG NIL)
          (SETQ FLIPFLOP 0)
          (SETQ TIMON (QUOTIENT (CLOCK)
             60))
          (RECONSTRUCT (QUOTE (TELL ME YOUR PROBLEMS "."
                 PLEASE TERMINATE INPUT WITH A PERIOD OR A
                 QUESTION MARK ".."))
            T)
          (SETNONE)
      A   (PRIN1 (QUOTE "
*"))
          (COND
            ((NULL (SETQ SENTENCE (MAKESENTENCE)))
              (GO A)))
          (SETQ KEYSTACK (CDR SENTENCE))
          (SETQ SENTENCE (CAR SENTENCE))
          (COND
            ((EQUAL SENTENCE (QUOTE (GOODBYE)))
              (RETURN (RECONSTRUCT (APPEND (QUOTE (IT'S BEEN
                           MY PLEASURE "," THAT'S))
                     (CONS (PACK (LIST (QUOTE $)
                           (REMAINDER (PLUS (QUOTIENT
```

ELIZA is an early natural language processing computer program created from 1964 to 1966 at the MIT Artificial Intelligence Laboratory by Joseph Weizenbaum.

Created to demonstrate the superficiality of communication between man and machine, Eliza simulated conversation by using a 'pattern matching' and substitution methodology that gave users an illusion of understanding on the part of the program.



1969: MARVIN MINSKY

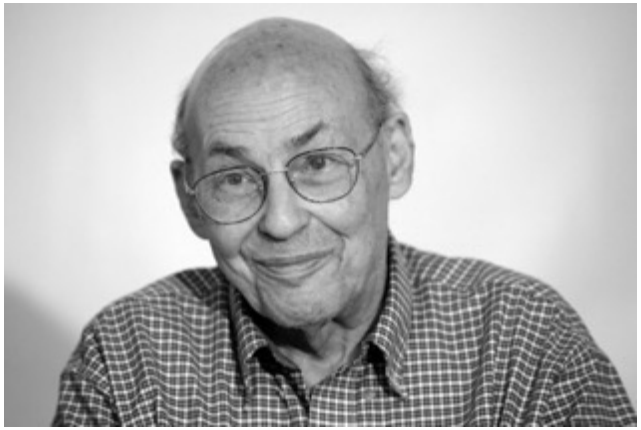
He developed, with Seymour Papert, the first Logo "turtle".
Minsky also built, in 1951, the first randomly wired neural network learning machine, SNARC.

Minsky wrote the book **Perceptrons** (with Seymour Papert), which became the foundational work in the analysis of artificial neural networks. This book is the center of a controversy in the history of AI, as some claim it to have had great importance in discouraging research of neural networks in the 1970s, and contributing to the so-called ”**First AI winter**".

On the surface, XOR appears to be a very simple problem, however, Minksy and Papert (1969) showed that this was a big problem for neural network architectures of the 1960s, known as Perceptrons which are only one layer.

Input 1	Input 2	Output
0	0	0
0	1	1
1	1	0
1	0	1

$$\begin{aligned} p \oplus q &= (p \wedge \neg q) \vee (\neg p \wedge q) \\ &= (p \vee q) \wedge (\neg p \vee \neg q) \\ &= (p \vee q) \wedge \neg(p \wedge q) \end{aligned}$$

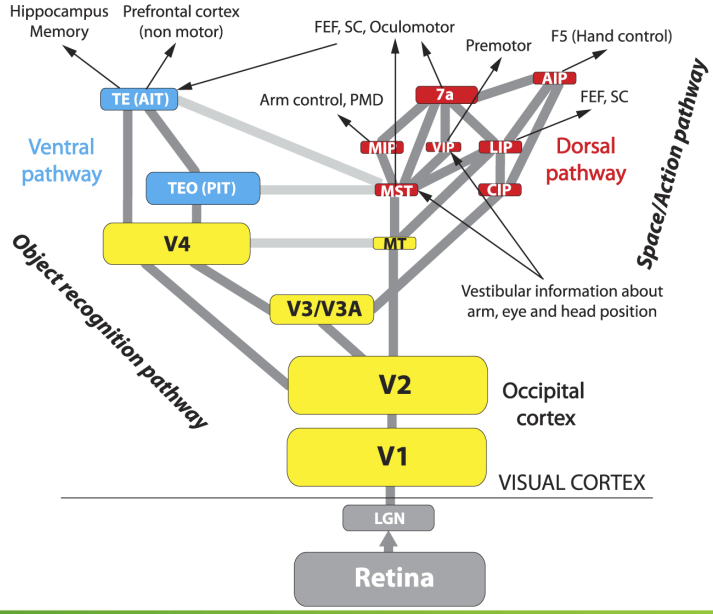


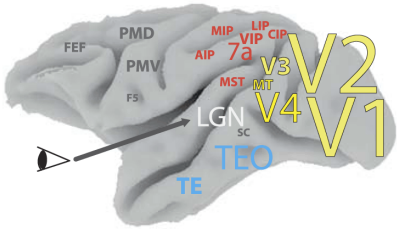
1981: DAVID MARR, DAVID HUBEL ET TORSTEN WIESEL

Better understanding how the biological visual system works:

- David Marr: Vision: A computational investigation into the human representation and processing of visual information, which was finished mainly on 1979 summer, was published in 1982 after his death

- Hubel and Wiesel were awarded the Nobel Prize in 1981 for their work on ocular dominance columns in the 1960s and 1970s.





1980: KUNIHIKO FUKUSHIMA

The first Deep Neural Network, inspired by the visual cortex.

But no real algorithms to set the values of the synaptic weights

Neocognitron: A Self-organizing Neural Network Model for a Mechanism of Pattern Recognition Unaffected by Shift in Position

Kunihiko Fukushima

NHK Broadcasting Science Research Laboratories, Kinuta, Setagaya, Tokyo, Japan

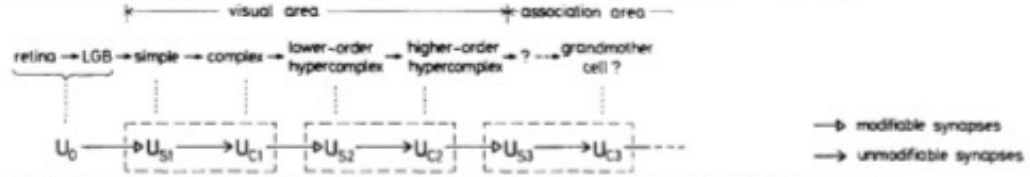


Fig. 1. Correspondence between the hierarchy model by Hubel and Wiesel, and the neural network of the neocognitron

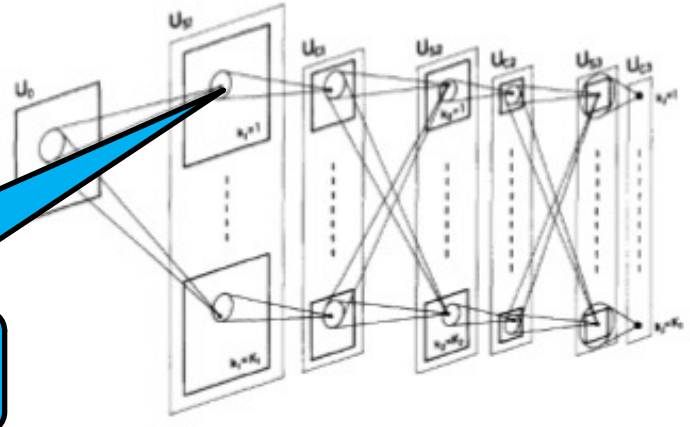


Fig. 2. Schematic diagram illustrating the interconnections between layers in the neocognitron

Biol. Cybernetics 36, 193–202 (1980)

AROUND 1986: GEOFFREY HINTON

He was one of the first researchers who demonstrated the use of **generalized back-propagation algorithm** for training multi-layer neural networks.

He co-invented **Boltzmann machines** with David Ackley and Terry Sejnowski.

His other contributions to neural network research include distributed representations, time delay neural network, mixtures of experts, Helmholtz machines and Product of Experts

He is now working for Google.

Cognitive psychologist and computer scientist

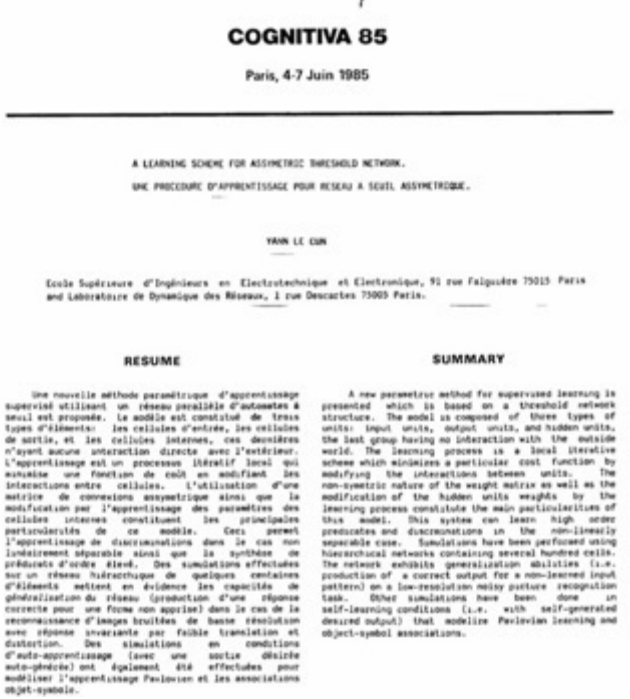
AROUND 1985: YANN LE CUN

In 1985, he proposed and published (in French), an early version of the learning algorithm known as **error backpropagation**

Near 1989, he developed a number of new machine learning methods, such as a biologically inspired model of image recognition called **Convolutional Neural Networks**, the "Optimal Brain Damage" regularization methods, and the Graph Transformer Networks method which he applied to handwriting recognition and OCR.

The **bank check recognition system** that he helped develop was widely deployed by NCR and other companies, reading over 10% of all the checks in the US in the late 1990s and early 2000s.

In 2013, LeCun became the first director of Facebook AI Research in New York City.



COGNITIVA 85

Paris, 4-7 Juin 1985

A LEARNING SCHEME FOR ASSYMETRIC THRESHOLD NETWORK.

UNE PROCEDURE D'APPRENTISSAGE POUR RESEAU A SEUIL ASSYMETRIQUE.

YANN LE CUN

Ecole Supérieure d'Ingénieurs en Electrotechnique et Electronique, 91 rue Falguière 75015 Paris and Laboratoire de Dynamique des Réseaux, 1 rue Descartes 75005 Paris.

RESUME

Une nouvelle méthode paramétrique d'apprentissage supervisé utilisant un réseau parallèle d'automates à seuil est proposée. Le modèle est constitué de trois types d'éléments: les cellules d'entrée, les cellules de sortie, et les cellules internes, ces dernières n'ayant aucune interaction directe avec l'extérieur. L'apprentissage est un processus itératif local qui automise une fonction de coût en modifiant les interactions entre cellules. L'utilisation d'une matrice de connexions assymetrique ainsi que la modification par l'apprentissage des paramètres des cellules internes constituent les principales particularités de ce modèle. Ceci permet l'apprentissage de discriminations dans le cas non linéairement séparable ainsi que la synthèse de prédicats d'ordre élevé. Des simulations effectuées sur un réseau hiérarchique de quelques centaines d'éléments mettent en évidence les capacités de généralisation du réseau (production d'une réponse correcte pour une forme non apprise) dans le cas de la reconnaissance d'images bruitées de basse résolution avec réponse invariante par faible translation et distortion. Des simulations en conditions d'auto-apprentissage (avec une sortie désirée auto-générée) ont également été effectuées pour modéliser l'apprentissage Pavlovien et les associations objet-symbole.

SUMMARY

A new parametrix method for supervised learning is presented which is based on a threshold network structure. The model is composed of three types of units: input units, output units, and hidden units, the last group having no interaction with the outside world. The learning process is a local iterative scheme which minimizes a particular cost function by modifying the interactions between units. The non-symetric nature of the weight matrix as well as the modification of the hidden units weights by the learning process constitute the main particularities of this model. This system can learn high order predicates and discriminations in the non-linearly separable case. Simulations have been performed using hierarchical networks containing several hundred cells. The network exhibits generalization abilities (i.e. production of a correct output for a non-learned input pattern) on a low-resolution noisy picture recognition task. Other simulations have been done in self-learning conditions (i.e. with self-generated desired output) that modelize Pavlovian learning and object-symbol associations.

1987: NETTALK. A 3 LAYERS PERCEPTION LEARN TO READ

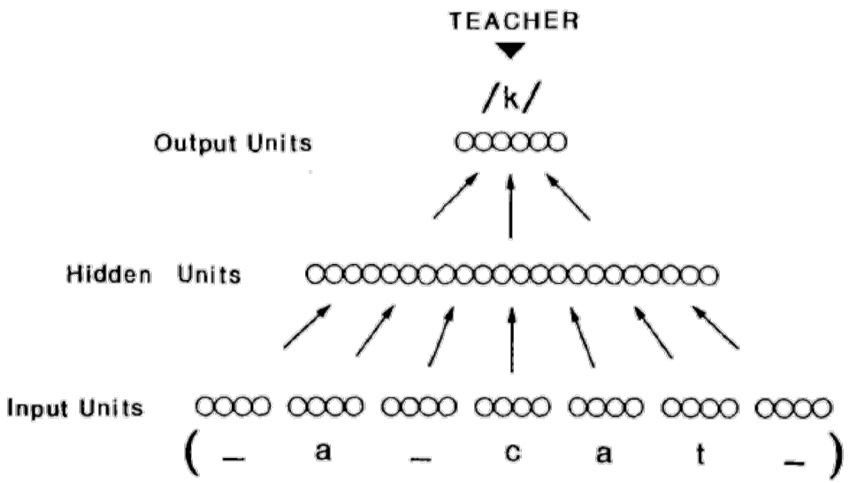


Figure 1: Schematic drawing of the NETtalk network architecture. A window of letters in an English text is fed to an array of 203 input units. Information from these units is transformed by an intermediate layer of 80 "hidden" units to produce patterns of activity in 26 output units. The connections in the network are specified by a total of 18629 weight parameters (including a variable threshold for each unit).

From T. J. Sejnowski and C. R. Rosenberg, "Parallel networks that learn to pronounce English text," *Complex Systems*, vol. 1, no. 1, pp. 145–168, 1987.

Slide from Christian Gamrat

1990'S NEUROCOMPUTERS...

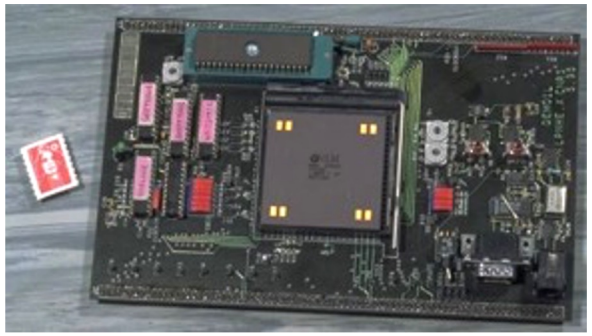
Siemens : MA-16 Chips (SYNAPSE-1 Machine 1994)

- Synapse-1, neurocomputer with 8xMA-16 chips
- Synapse3-PC, PCI board with 2xMA-16 (1.28 Gpcs)
- about 8,000 times as fast as a Sun Workstation (Sparc-2)

1990'S NEUROCOMPUTERS...

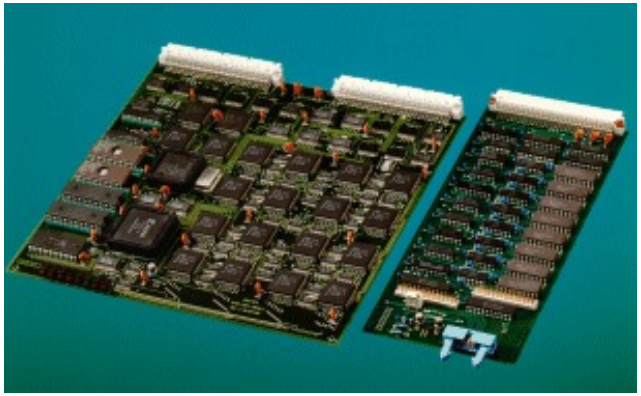
Philips : L-Neuro

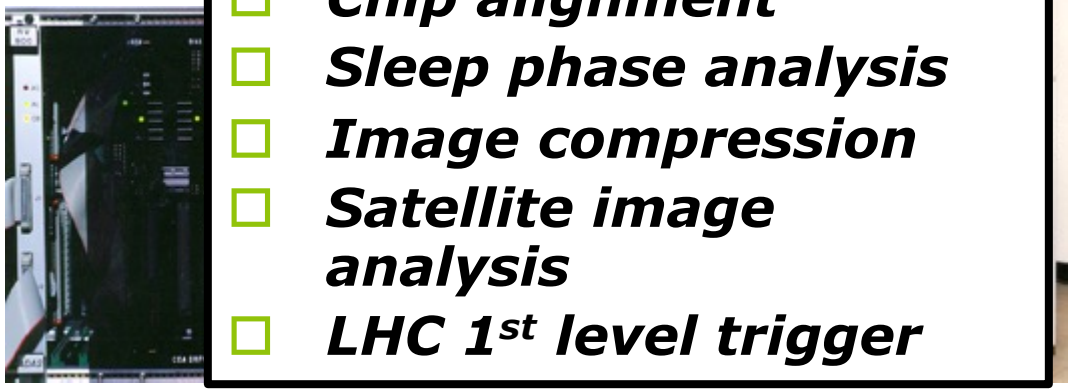
- 1st Gen 16 PEs 26 MCps (1990)
- 2nd Gen 12 PEs 720 MCps (1994)
- ➢ Used in satellite, fruit sorting, PCB inspection, sleep analysis, …



CEA's MIND machine

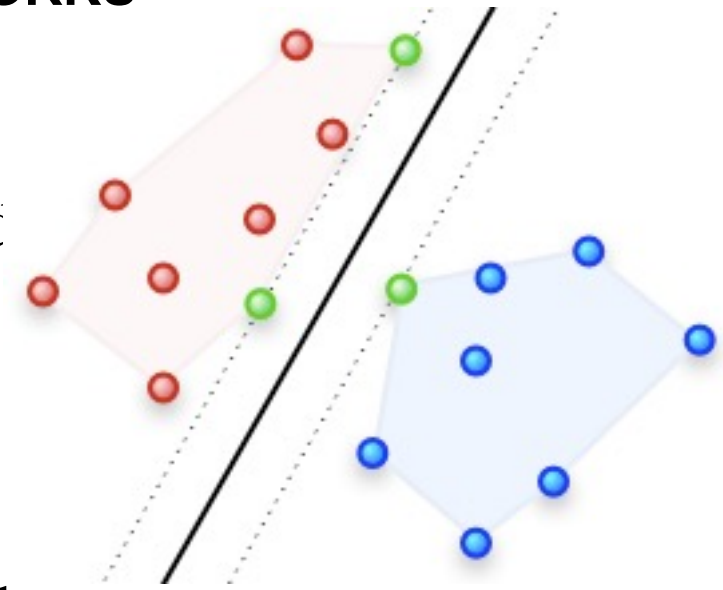
- Hybrid analog/digital: MIND-128 (
- Fully digital: MIND-1024 (1991)





- ***Orange video-grading***
- ***Chip alignment***
- ***Sleep phase analysis***
- ***Image compression***
- ***Satellite image analysis***
- ***LHC 1st level trigger***

1995: SVM OR THE 2ND WINTER OF NEURAL NETWORKS



Support Vector Machines (SVMs)

The original SVM algorithm was invented by Vladimir N. Vapnik and Alexey Ya. Chervonenkis in **1963**.

In 1992, Bernhard E. Boser, Isabelle M. Guyon and Vladimir N. Vapnik suggested a way to create nonlinear classifiers by applying the kernel trick to maximum-margin hyperplanes. The current standard incarnation (soft margin) was proposed by Corinna Cortes and Vapnik in 1993 and published in 1995.

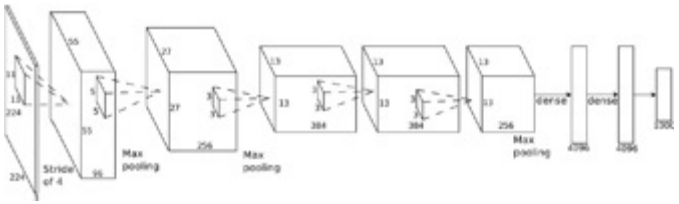
1997: CHESS AND DEEP BLUE

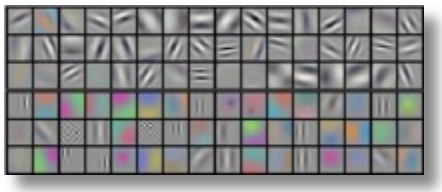
As far back as the mid-60s, chess was called the "Drosophila of artificial intelligence" – a reference to the fruit flies biologists used to uncover the secrets of genetics –
1997 – Deep Blue wins a six-game match against Garry Kasparov.

2012: DEEP NEURAL NETWORKS RISE AGAIN

They give the ***state-of-the-art performance*** e.g. in image classification

- **ImageNet classification (Hinton's team, hired by Google)**
  - 14,197,122 images, 1,000 different classes
  - Top-5 17% error rate (huge improvement) in 2012 (now ~ 3.5%)





"**Supervision**" network
Year: 2012
650,000 neurons
60,000,000 parameters
630,000,000 synapses

- **Facebook's 'DeepFace' Program (labs headed by Y. LeCun)**

The 2018 **Turing Award recipients** are Google VP Geoffrey Hinton, Facebook's Yann LeCun and Yoshua Bengio, Scientific Director of AI research center Mila.

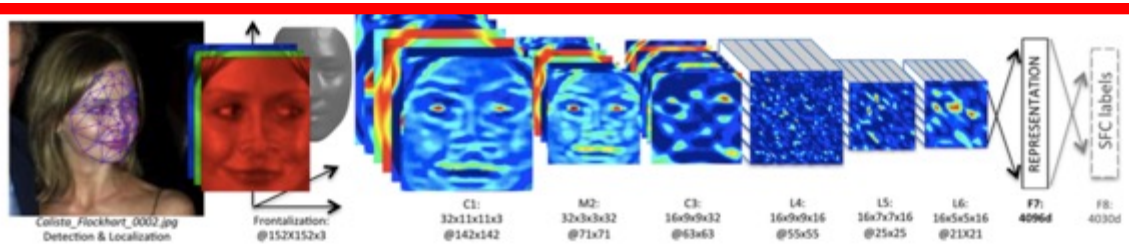
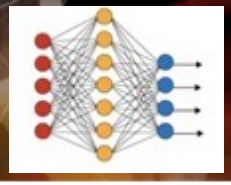


Figure 2. **Outline of the *DeepFace* architecture**. A front-end of a single convolution-pooling-convolution filtering on the rectified input, followed by three locally-connected layers and two fully-connected layers. Colors illustrate feature maps produced at each layer. The net includes more than 120 million parameters, where more than 95% come from the local and fully connected layers.

From:Y. Taigman, M. Yang, M.A. Ranzato, "DeepFace: Closing the Gap to Human-Level Performance in Face Verification"

ImageNet: Classification

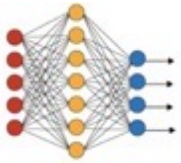


- Give the name of the dominant object in the image
- Top-5 error rates: if correct class is not in top 5, count as error
  - Black:ConvNet, Purple: no ConvNet

2012 Teams	%error
Supervision (Toronto)	15.3
ISI (Tokyo)	26.1
VGG (Oxford)	26.9
XRCE/INRIA	27.0
UvA (Amsterdam)	29.6
INRIA/LEAR	33.4

2013 Teams	%error
Clarifai (NYU spinoff)	11.7
NUS (singapore)	12.9
Zeiler-Fergus (NYU)	13.5
A. Howard	13.5
OverFeat (NYU)	14.1
UvA (Amsterdam)	14.2
Adobe	15.2
VGG (Oxford)	15.2
VGG (Oxford)	23.0

2014 Teams	%error
GoogLeNet	6.6
VGG (Oxford)	7.3
MSRA	8.0
A. Howard	8.1
DeeperVision	9.5
NUS-BST	9.7
TTIC-ECP	10.2
XYZ	11.2
UvA	12.1



COMPETITION ON IMAGENET !

Image classification:
Discriminative model

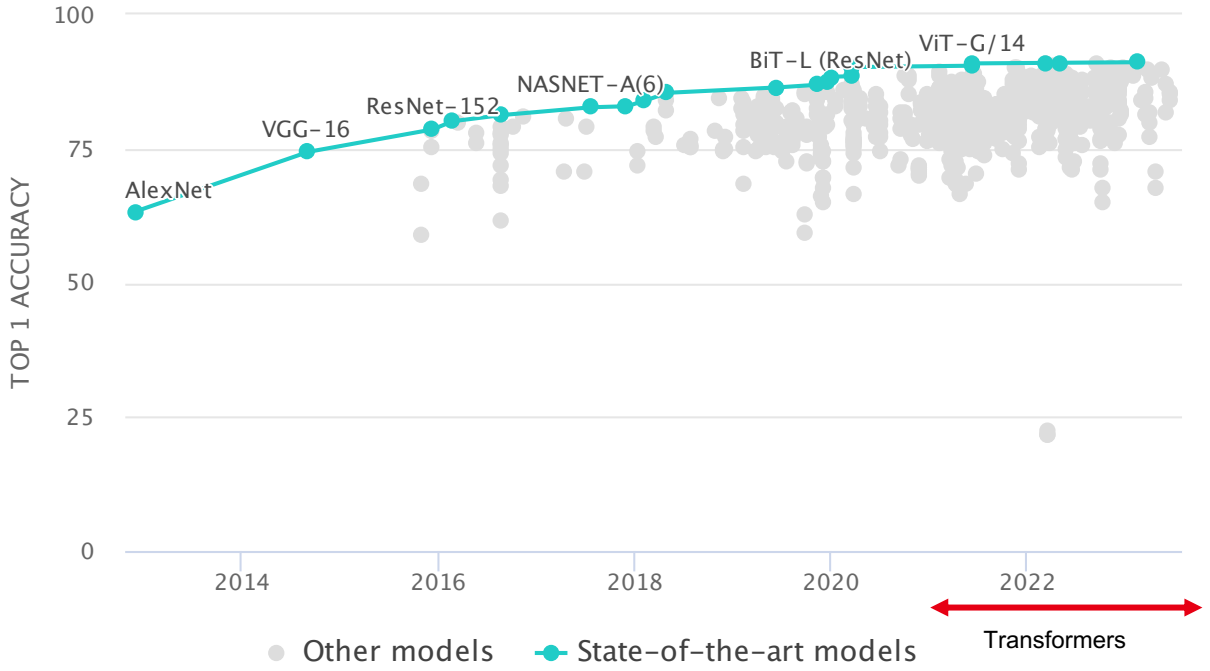
Name of the algorithm	Date	Error on test set
Supervision	2012	15.3%
Clarifai	2013	11.7%
GoogLeNet	2014	6.66%
Humain level (Adrej Karpathy)		**5%**
Microsoft	05/02/2015	4.94%
Google	02/03/2015	4.82%
Baidu/ Deep Image	10/05/2015	4.58%
Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences	10/12/2015 (le CNN a 152 couches!)	3.57%
Google Inception-v3 (Arxiv)	2015	3.5%
WMW (Momenta)	2017	2.2%
	Now	0.98 %

From https://paperswithcode.com/sota/image-classification-on-imagenet

COMPETITION ON IMAGENET !

Name of the algorithm	Date	Error on test set
Supervision	2012	15.3%
Clarifai	2013	11.7%
GoogLeNet	2014	6.66%

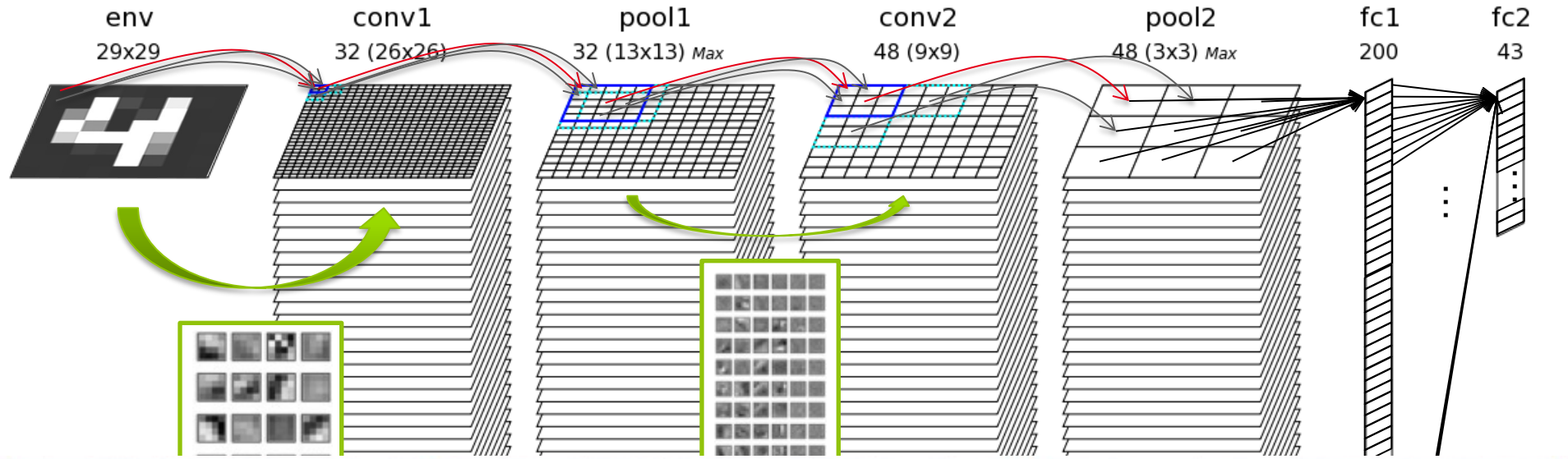
Image classification:
Discriminative model

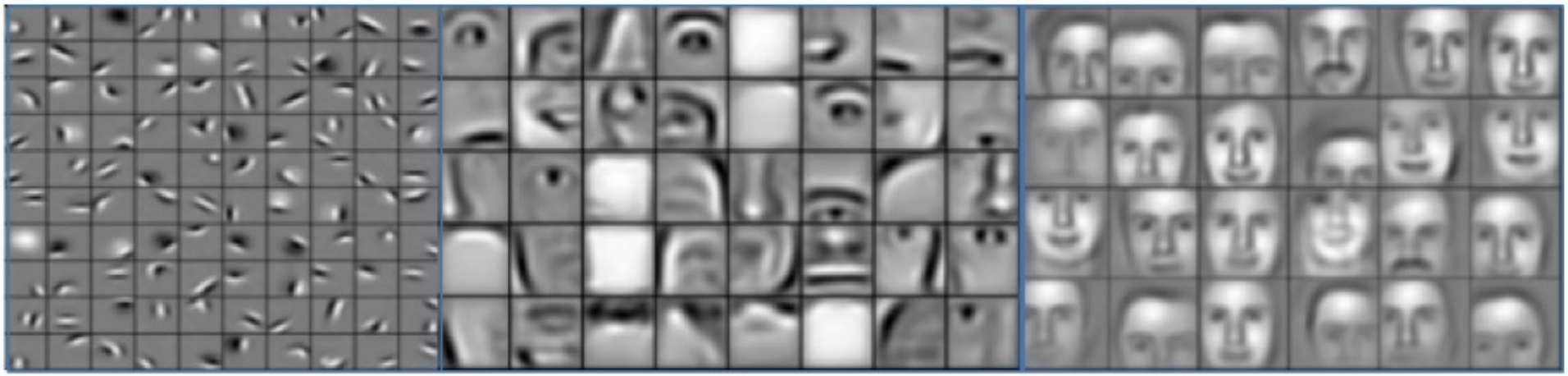


From https://paperswithcode.com/sota/image-classification-on-imagenet

EXAMPLES OF RESULTS (IMAGENET)

WHAT IS A CNN?


env
29x29
conv1
32 (26x26)
pool1
32 (13x13) Max
conv2
48 (9x9)
pool2
48 (3x3) Max
fc1
200
fc2
43


hidden layer 1
hidden layer 2
hidden layer 3
input layer

USE OF DEEP LEARNING

Supervised DL works amazingly well, when you have data

Y. LeCun

- **And services like Facebook, Instagram, Google, Youtube… are built around it.**
  - Content understanding, filtering, ranking, translation, accessibility…
  - Content understanding, filtering, ranking, translation, accessibility…

From Yann LeCun

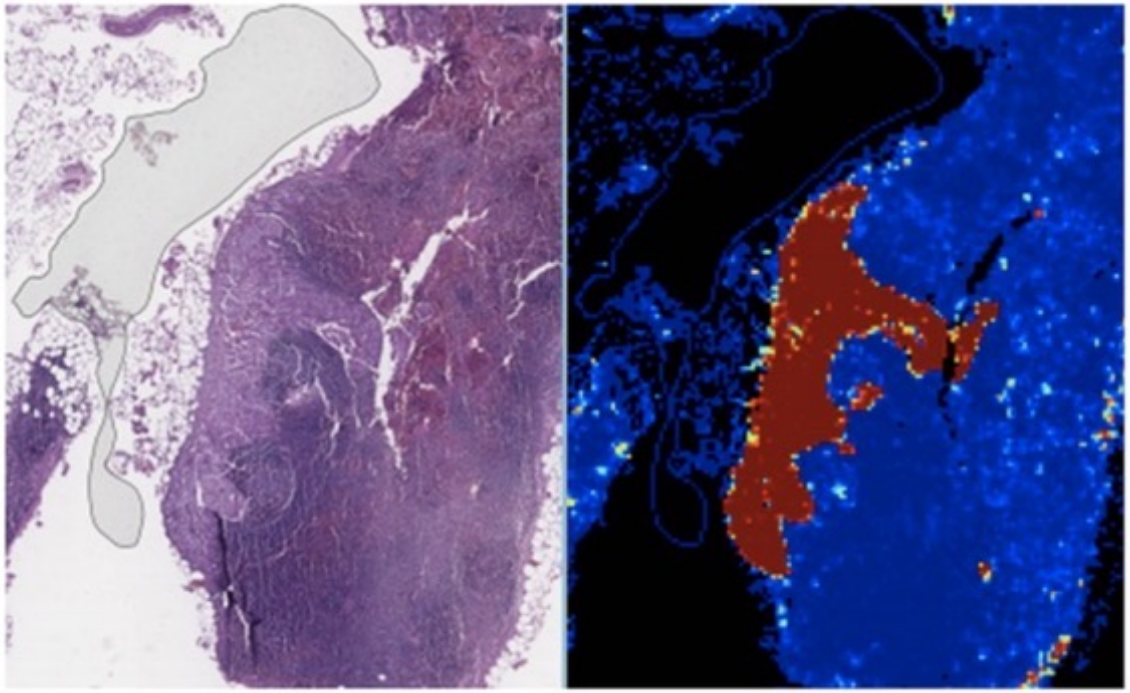
IMAGE ANALYSIS

Detecting Cancer Metastases

Tumor localization score (FROC):

Pathologist: 0.73
AI model: 0.89
(better)

Detecting Cancer Metastases on Gigapixel Pathology Images (2017)



From Olivier Temam

DEEP MANTA

MANY-TASK DEEP NEURAL NETWORK FOR VISUAL OBJECT RECOGNITION

Applications

Driving assistance, autonomous driving
Smart city
Video-protection
Advanced Manufacturing

Technology

1. Object detection
2. Fine-grained recognition
3. Accurate pose estimation
4. 2D/3D localisation
5. Part localisation
6. Part visibility characterization

Performance

KITTI Benchmark:

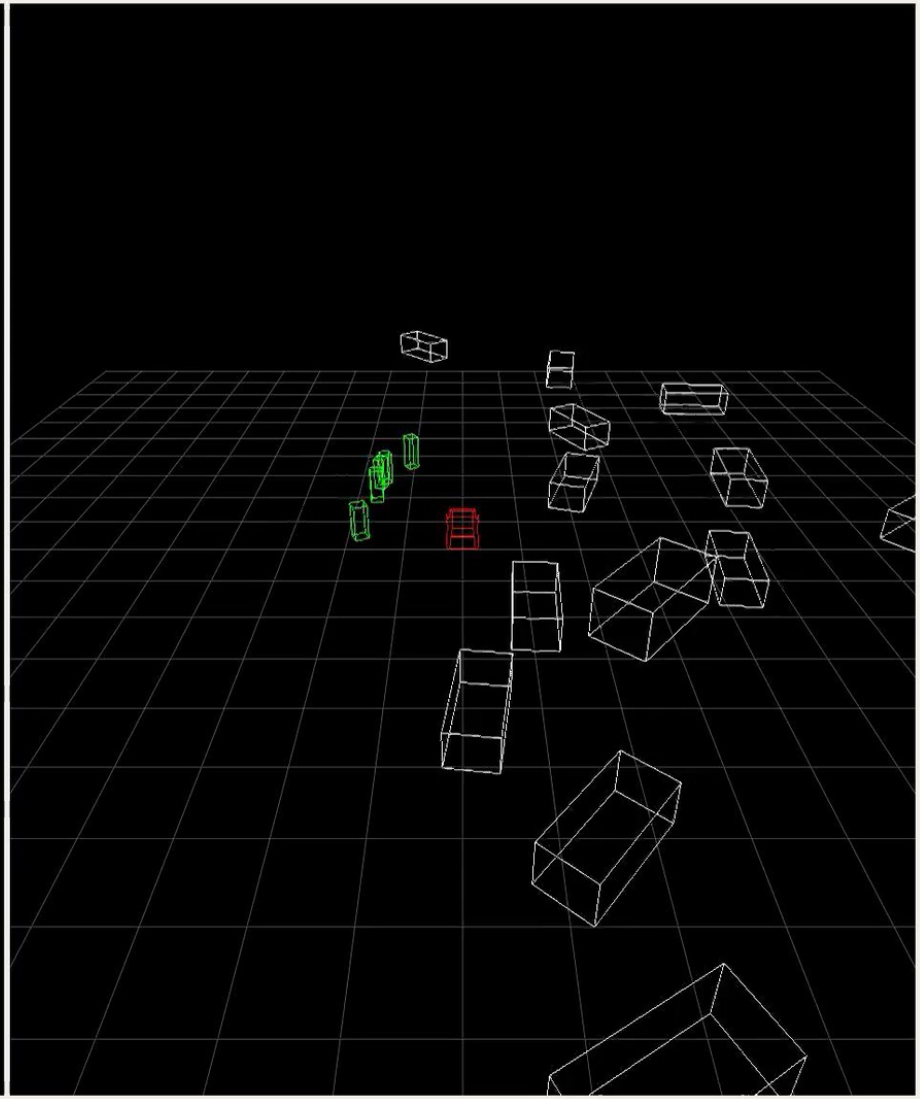
- 1st rank in vehicle orientation estimation
- Top-10 in object detection

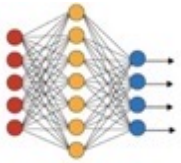
Runs at 10 Hz on Nvidia Gtx 1080

CVPR 2017 : F. Chabot, M. Chaouch, J. Rabarisoa, C. Teulière and T. Château
Deep MANTA: A Coarse-to-fine Many-Task Network for joint 2D and 3D vehicle analysis from monocular image.

DEEP MANTA

MANY-TASK DEEP NEURAL NETWORK FOR VISUAL OBJECT RECOGNITION



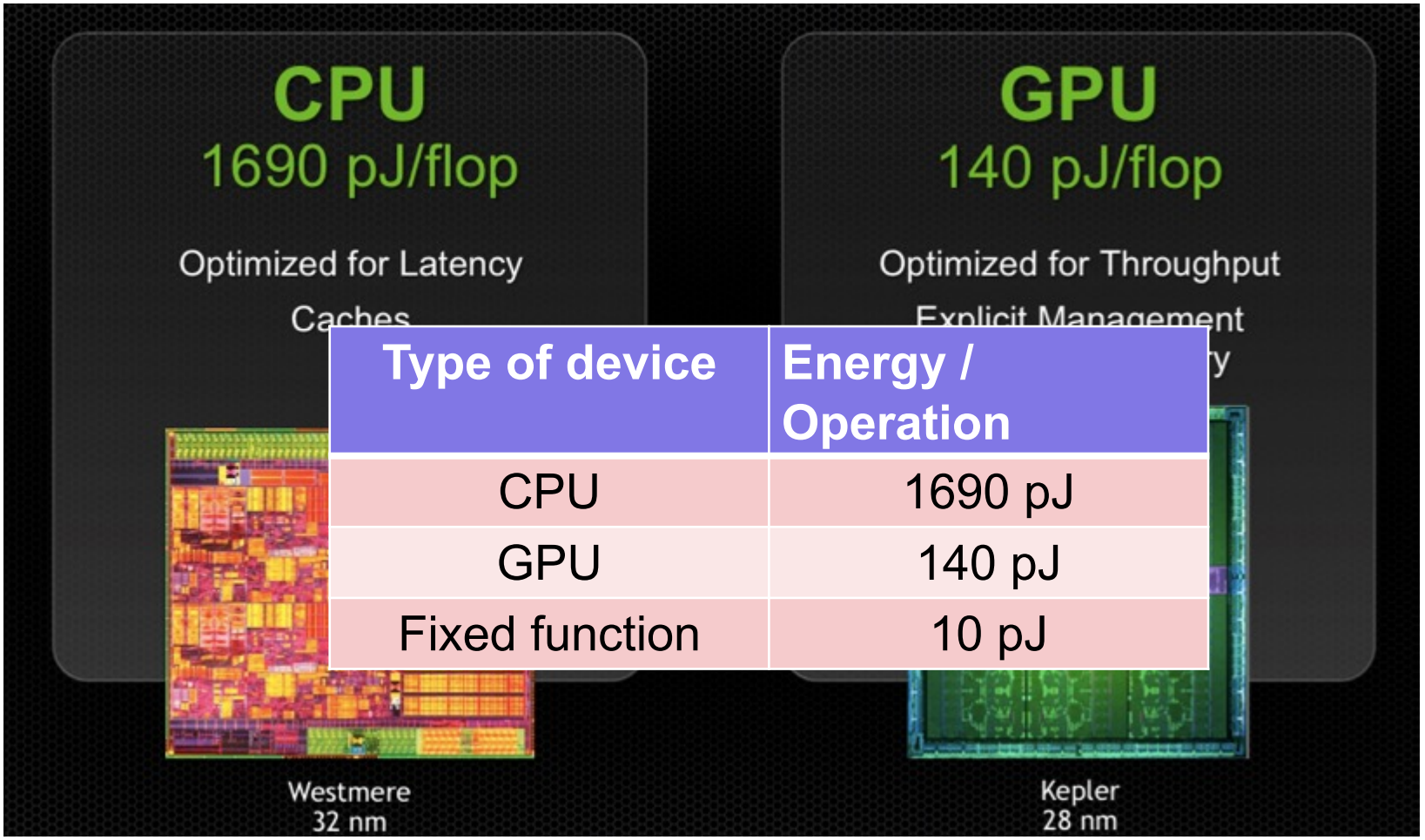


DEEP LEARNING AND VOICE RECOGNITION

" The need for TPUs really emerged about six (12) years ago, when we started using computationally expensive deep learning models in more and more places throughout our products. The computational expense of using these models had us worried. If we considered a scenario where people use Google voice search for just three minutes a day and we ran deep neural nets for our speech recognition system on the processing units we were using, we would have had to *double the number of Google data centers*!"

[https://cloudplatform.googleblog.com/2017/04/quantifying-the-performance-of-the-TPU-our-first-machine-learning-chip.html]

SPECIALIZATION LEADS TO MORE EFFICIENCY EFFICIENCY



Type of device	Energy / Operation
CPU	1690 pJ
GPU	140 pJ
Fixed function	10 pJ

From Bill Dally (nVidia) « Challenges for Future Computing Systems »
HiPEAC conference 2015

2017: GOOGLE'S CUSTOMIZED TPU HARDWARE…

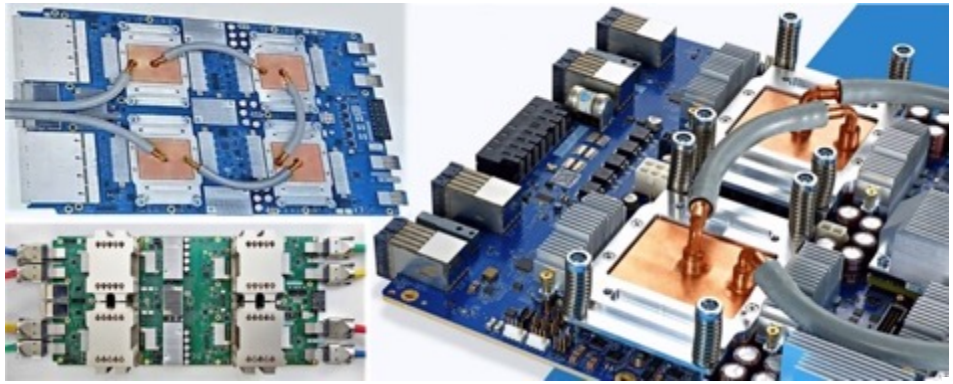
… required to increase energy efficiency
with accuracy adapted to the use (e.g. float 16)

Google's TPU2 : 11.5 $petaflops_{16}$ of machine learning number crunching
(and guessing about 400+ KW…, 100+ $GFlops_{16}$/W)

From Google

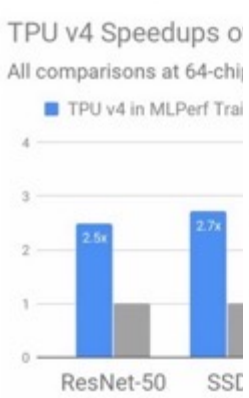
Peta = 10^{15} = million of milliard

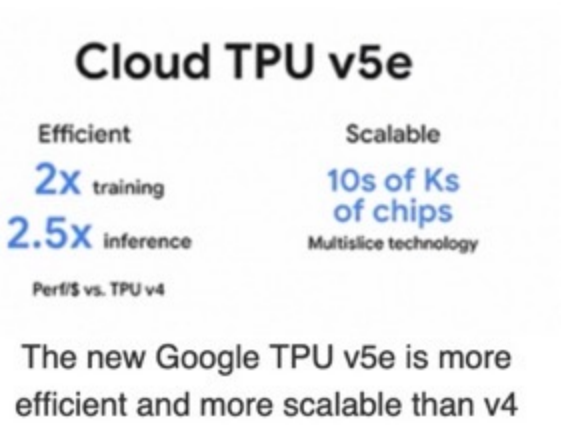
Google's Customized TPU (V3, V4, V5) hardware



From https://www.nextplatform.com/2018/05/10/tearing-apart-googles-tpu-3-0-ai-coprocessor/

Chip	TPUv1	TPUv2	TPUv3
Announced	2016	May-17	May-18
Access	Internal-Only	Service Beta	*Undisclosed*
Introduction	2015	Feb 2018	*Undisclosed*
Process	28nm	*20nm est.*	*16/12nm est.*
Die Size	~300mm2	*Undisclosed*	*Undisclosed*
TOPS	92 / 23	45	90
Matrix Input	INT8 / INT16	bfloat16	bfloat16
Memory	8GB DDR3	16GB HBM	32GB HBM
CPU Interface	PCIe 3.0 x16	PCIe 3.0 x8	*PCIe 3.0 x8 est.*
Power Consumption	40W	*200-250W est.*	*200W est.*

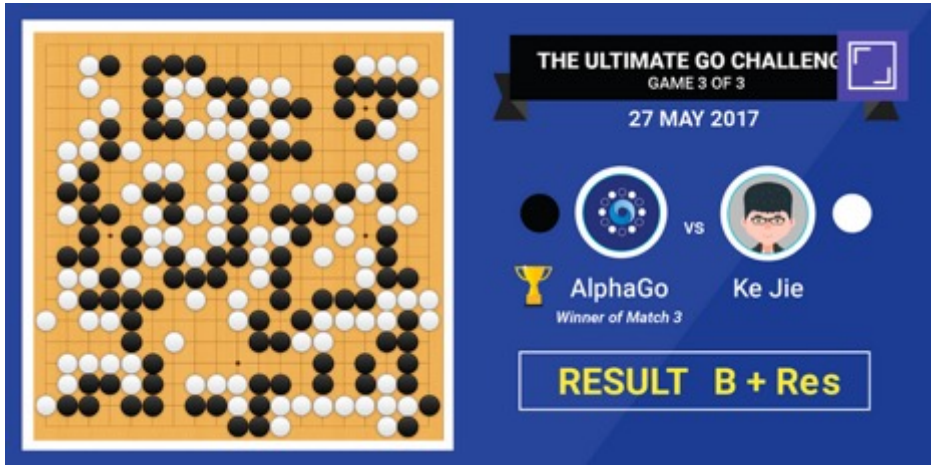




https://www.hpcwire.com/2023/08/30/google-tpu-v5e-ai-chip-debuts-after-controversial-origins/

2017: THE GAME OF GO

Ke Jie (human world champion in the “Go” game), after being defeated by AlphaGo on May 27th 2017, will work with Deepmind to make a tool from AlphaGo to further help Go players to enhance their game.

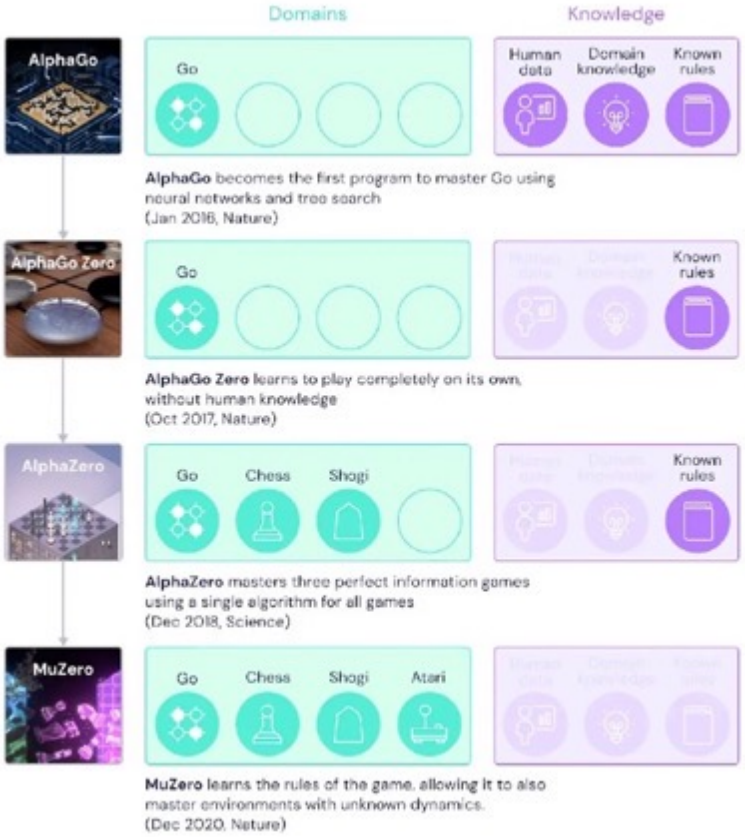


ALPHAGO ZERO: SELF-PLAYING TO LEARN

From doi:10.1038/nature24270 (Received 07 April 2017)

MUZERO: MASTERING GO, CHESS, SHOGI AND ATARI WITHOUT RULES

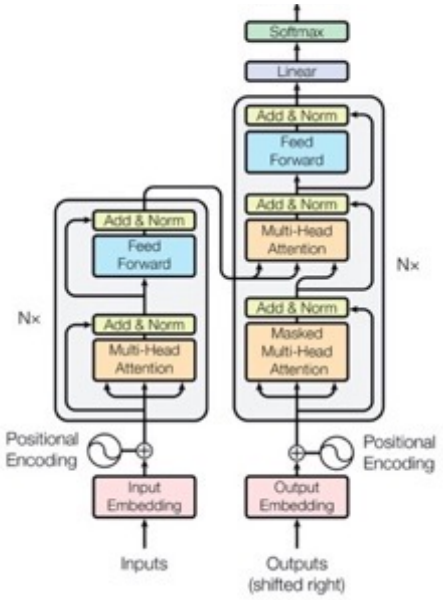
In 2016, we introduced AlphaGo, the first artificial intelligence (AI) program to defeat humans at the ancient game of Go. Two years later, its successor - AlphaZero - learned from scratch to master Go, chess and shogi. Now, in a paper in the journal Nature, we describe MuZero, a significant step forward in the pursuit of general-purpose algorithms. MuZero masters Go, chess, shogi and Atari without needing to be told the rules, thanks to its ability to plan winning strategies in unknown environments.



More details at: https://deepmind.com/blog/article/muzero-mastering-go-chess-shogi-and-atari-without-rules

2017: TRANSFORMERS

We propose a new simple network architecture, the **Transformer**, **based** solely on **attention mechanisms**, dispensing with recurrence and convolutions entirely. Experiments on two machine translation tasks show these models to be superior in quality while being more parallelizable and requiring significantly less time to train. On the WMT 2014 English-to-French translation task, our model establishes a new single-model state-of-the-art BLEU score of 41.8 after training for 3.5 days on eight GPUs, a small fraction of the training costs of the best models from the literature. We show that the Transformer generalizes well to other tasks by applying it successfully to English constituency parsing both with large and limited training data.



Attention Is All You Need

Ashish Vaswani* Google Brain avaswani@google.com

Noam Shazeer* Google Brain noam@google.com

Niki Parmar* Google Research nikip@google.com

Jakob Uszkoreit* Google Research usz@google.com

Llion Jones* Google Research llion@google.com

Aidan N. Gomez* † University of Toronto aidan@cs.toronto.edu

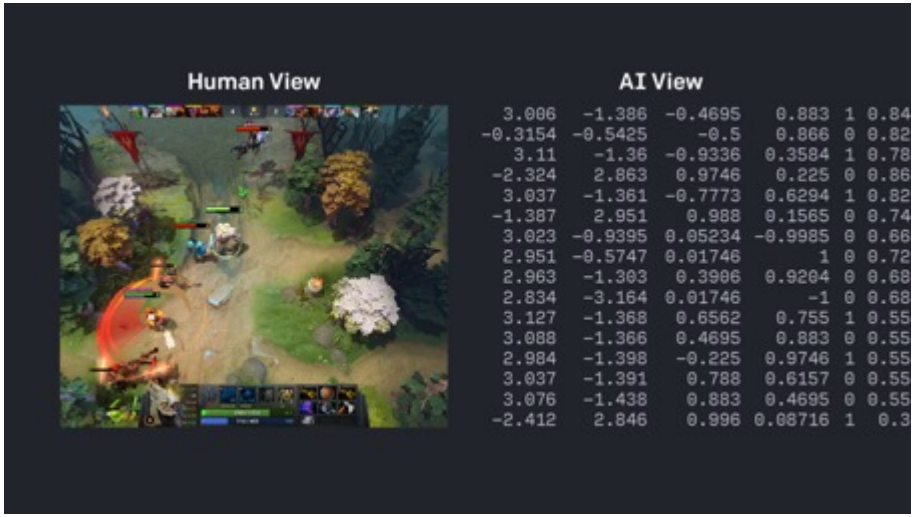
Łukasz Kaiser* Google Brain lukaszkaiser@google.com

Illia Polosukhin* ‡ illia.polosukhin@gmail.com

2019: OPENAI WINS DOTA 2 ESPORT GAME

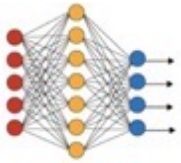
OpenAI Five Defeats Dota 2 World Champions

OpenAI Five is the first AI to beat the world champions in an esports game, having won two back-to-back games versus the world champion Dota 2 team.
It is also the first time an AI has beaten esports pros on livestream.



Cooperative mode

OpenAI Five's ability to play with humans presents a compelling vision for the future of human-AI interaction, one where AI systems collaborate and enhance the human experience. Our testers reported feeling supported by their bot teammates, that they learned from playing alongside these advanced systems, and that it was generally a fun experience overall.



2020: OpenAI's GPT-3, an autoregressive language

"GPT-3 shows that language model performance scales as a power-law of model size, dataset size, and the amount of computation.

GPT-3 demonstrates that a language model trained on enough data can solve NLP tasks that it has never encountered. That is, GPT-3 studies the model as a general solution for many downstream jobs without fine-tuning.

The size of state-of-the-art (SOTA) language models is growing by at least a factor of 10 every year.*"

GPT-3 175B is trained with 499 Billion tokens:

Dataset	# Tokens (Billions)
Total	499
Common Crawl (filtered by quality)	410
WebText2	19
Books1	12
Books2	55
Wikipedia	3

Rank	System	Cores	Rmax (TFlop/s)	Rpeak (TFlop/s)	Power (kW)
14	CEA-HF - BullSequana XH2000, AMD EPYC 7763 64C 2.45GHz, Atos BXI V2, Atos Commissariat a l'Energie Atomique (CEA) France	810,240	23,237.6	31,761.	4,959

"GPT-3 175B model required 3.14x10^{23} FLOPS (so about 87h of exaflop machine, 156 days on 5 MW computer) for computing for training.

Even at theoretical 28 TFLOPS for V100 and lowest 3 year reserved cloud pricing we could find, this will take 355 GPU-years and cost $4.6M for a single training run.

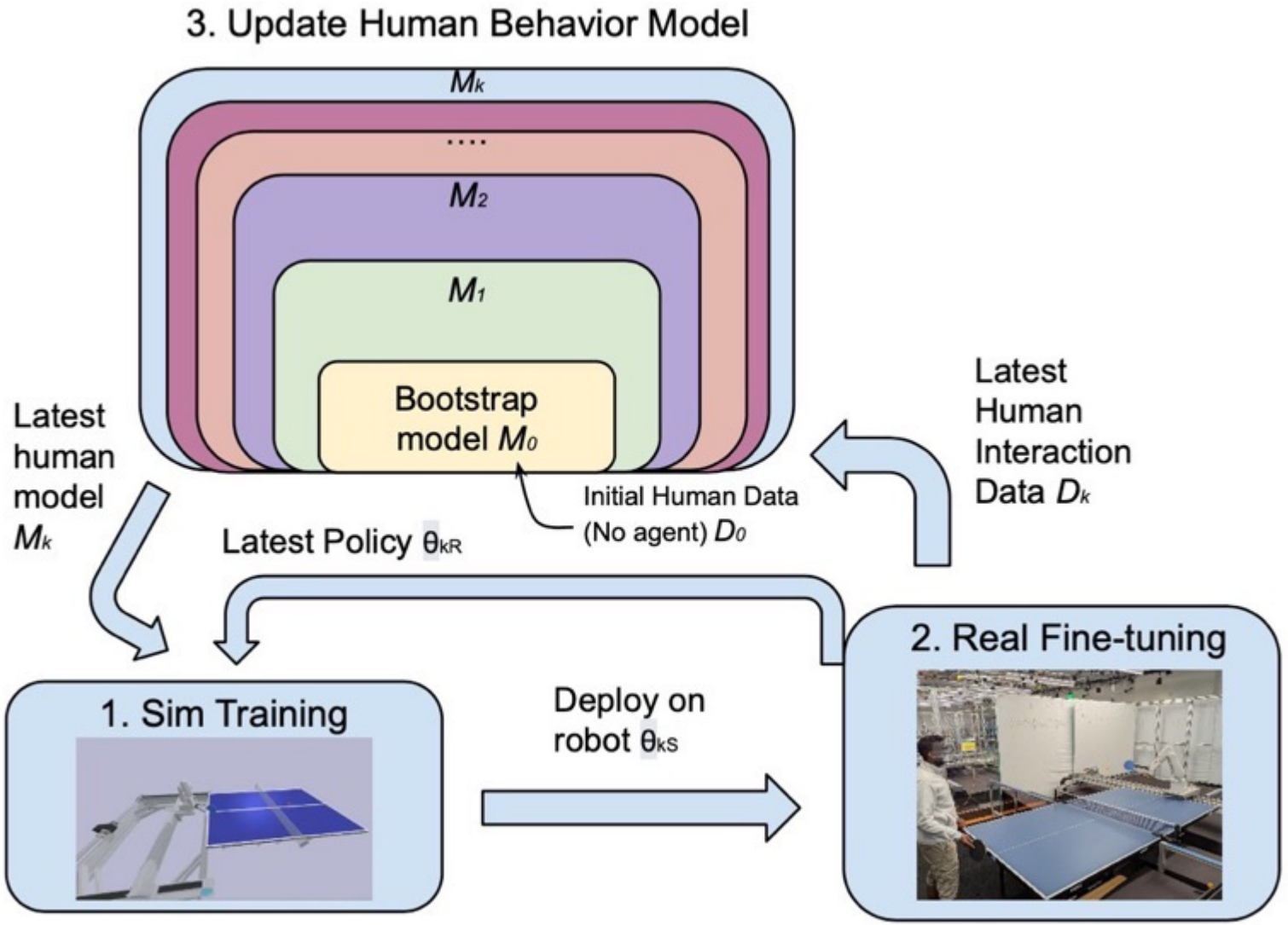
Similarly, a single RTX 8000, assuming 15 TFLOPS, would take 665 years to run*".

* https://lambdalabs.com/blog/demystifying-gpt-3/

2022: Increase of image quality by sequences of images

Low luminosity image improvement from B. Mildenhall et al. , “NeRF in the Dark: High Dynamic Range View Synthesis from Noisy Raw Images,” in 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, 2022, https://ieeexplore.ieee.org/document/9878457

2022: Reinforcement with simulation in the loop



The reinforcement technique with simulation in the loop allow to learn and adapt with minimum numbers of real data (from S. Abeyruwan et al., “i-Sim2Real: Reinforcement Learning of Robotic Policies in Tight Human-Robot Interaction Loops (pre-print), Arxiv, 22 November 2022. Available: https://arxiv.org/abs/2207.06572.

2022: “Art” generated by an AI (Stable Diffusion)

The large Language Models (LLM) (generally based on Transformers) have very interesting results…

Parameters
Seed : 58862 | Scale : 7.5 | Steps : 25 | Img Width : 512 | Img Height : 512 | model_version : 1.5fp16
Trigger sentence: ***cute otter, disney pixar, detailled fur, lot of details***

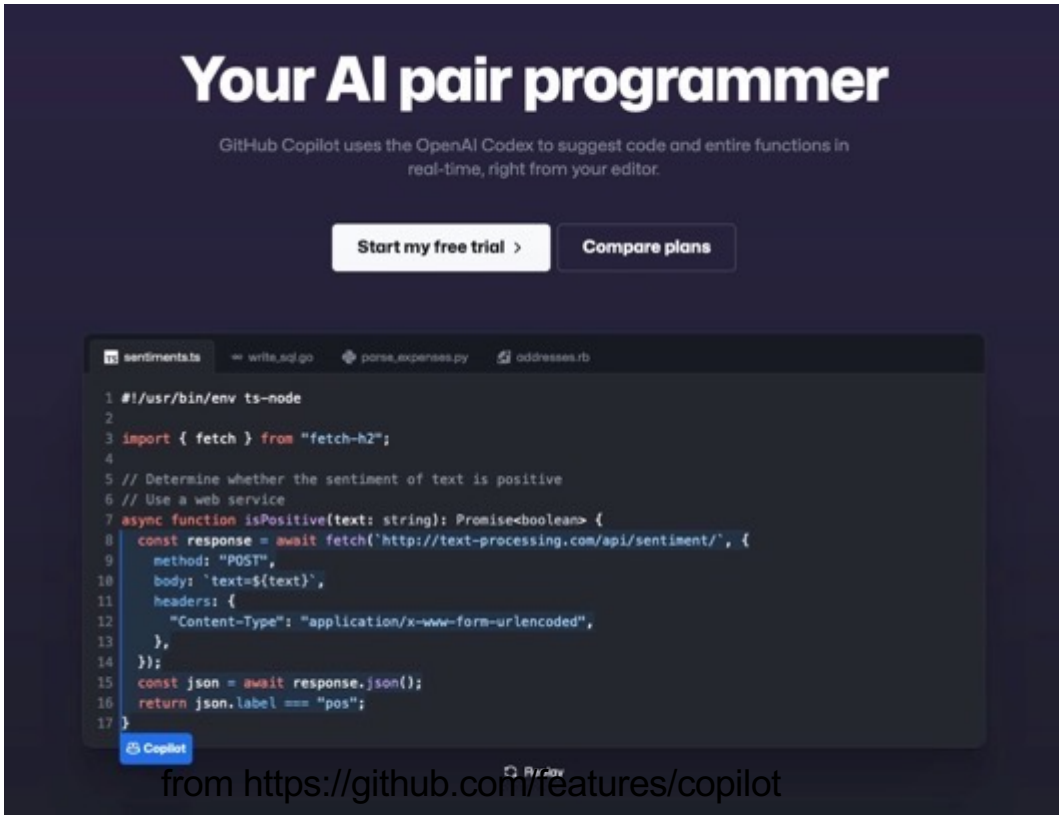
IMAGE GENERATION FROM SKETCHES

https://stability.ai/blog/clipdrop-launches-stable-doodle

LLMs for software programming

- Generating (small) pieces of code
- Get code from comments
- Chat to help programming
- Explain code
- Show examples
- Refactoring
- Create README
- Etc…

From https://www.hipeac.net/vision/#/latest/



from https://github.com/features/copilot

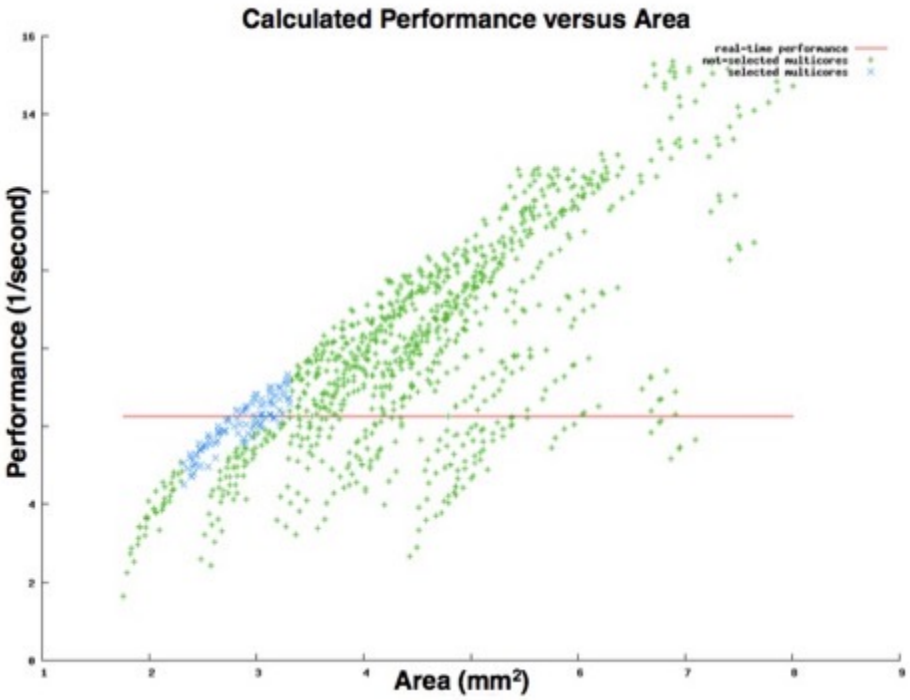
Selecting the right architecture: Design Space Exploration for the design of multi-core processors[1] (2010)

- Ne-XVP project – Follow-up of the TriMedia VLIW (https://en.wikipedia.org/wiki/Ne-XVP)
- 1,105,747,200 heterogeneous multicores in the design space
- 2 millions years to evaluate all design points
- AI inspired techniques allowed to reduce the induction time to only few days

=> *x16 performance increase*

AI techniques can be used to optimize/design efficient AI accelerators!

Nb: the system discovered **Amdahl's law** and **near memory computing**…



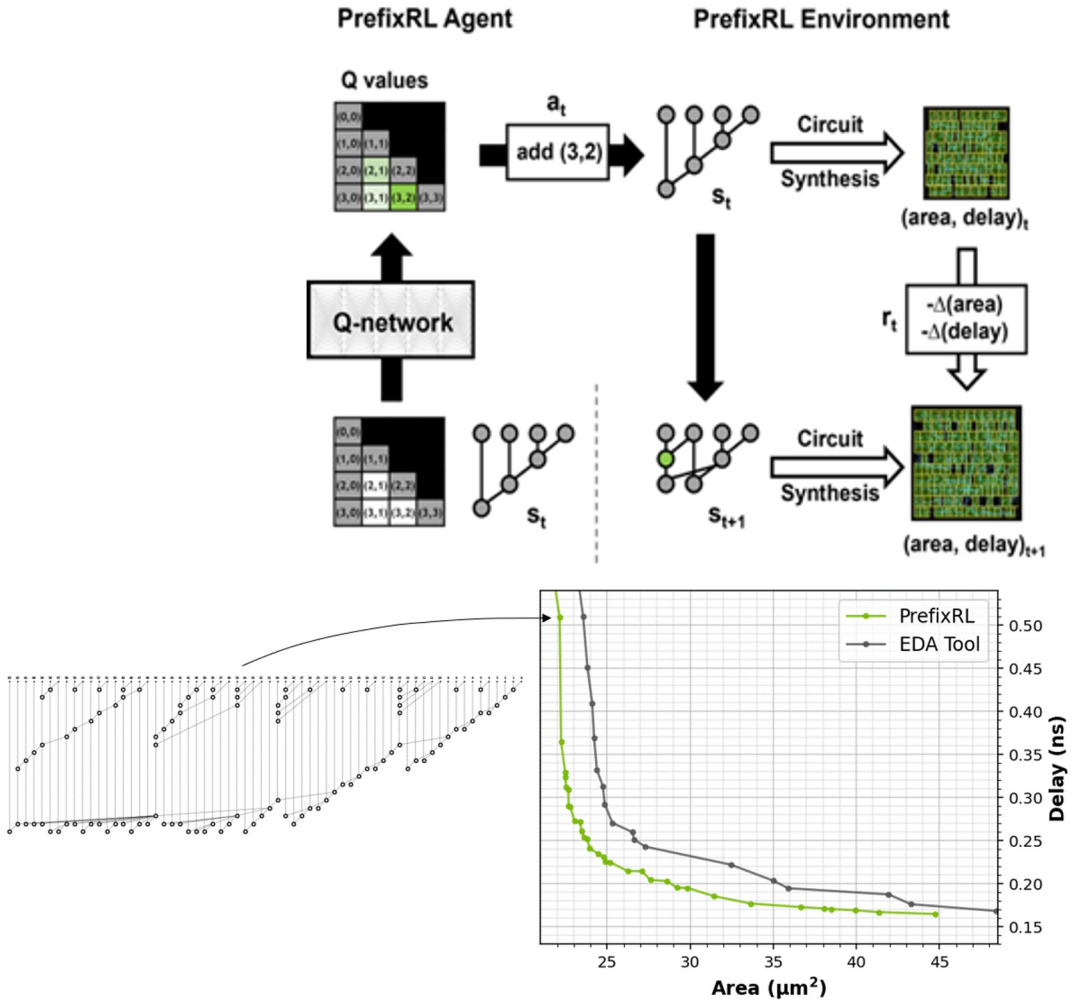
Gain ~ 16

[1] M. Duranton et all., "Rapid Technology-Aware Design Space Exploration for Embedded HeterogeneousMultiprocessors" in Processor and System-on-Chip Simulation, Ed. R. Leupers, 2010

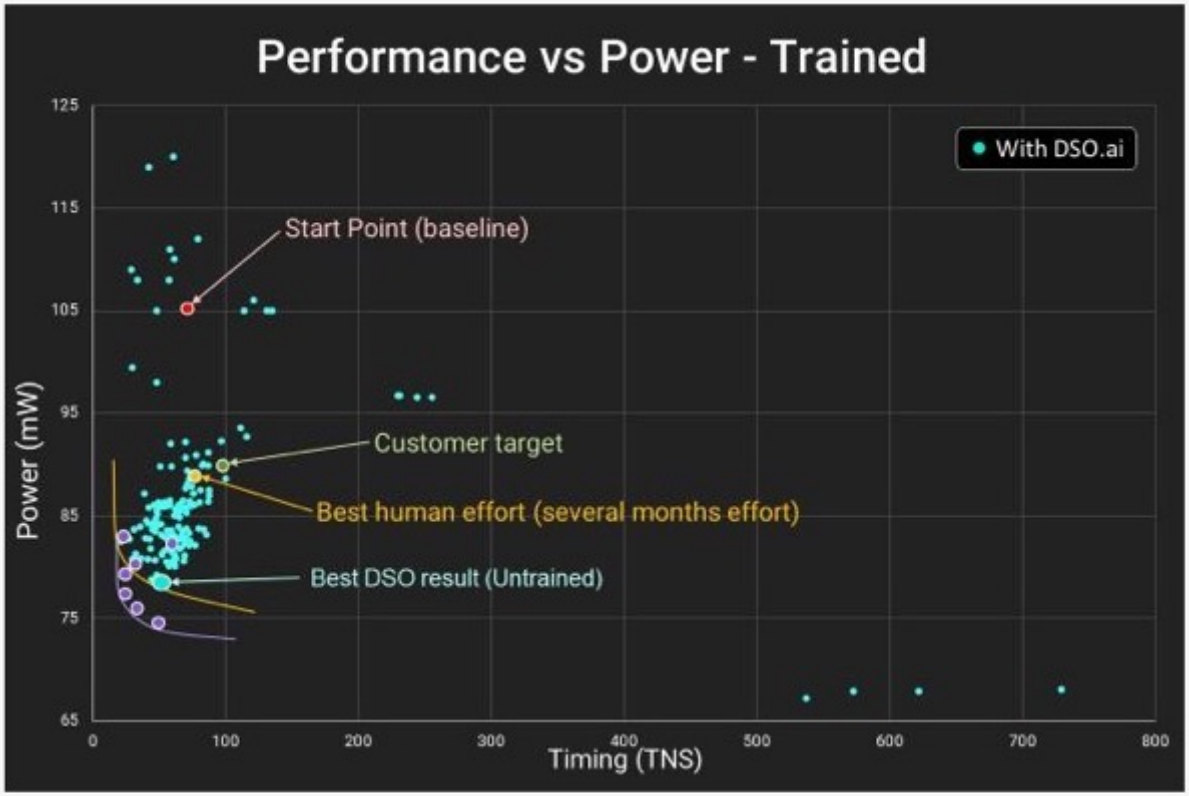
2022: AI creating AI chips (Nvidia)

Vast arrays of arithmetic circuits have powered NVIDIA GPUs to achieve unprecedented acceleration for AI, high-performance computing, and computer graphics. Thus, improving the design of these arithmetic circuits would be critical in improving the performance and efficiency of GPUs. What if AI could learn to design these circuits? In PrefixRL: Optimization of Parallel Prefix Circuits using Deep Reinforcement Learning, we demonstrate that not only can AI learn to design these circuits from scratch, but AI-designed circuits are also smaller and faster than those designed by state-of-the-art electronic design automation (EDA) tools. ***The latest NVIDIA Hopper GPU architecture has nearly 13,000 instances of AI-designed circuits.***

From https://developer.nvidia.com/blog/designing-arithmetic-circuits-with-deep-reinforcement-learning/



Optimization of hardware design

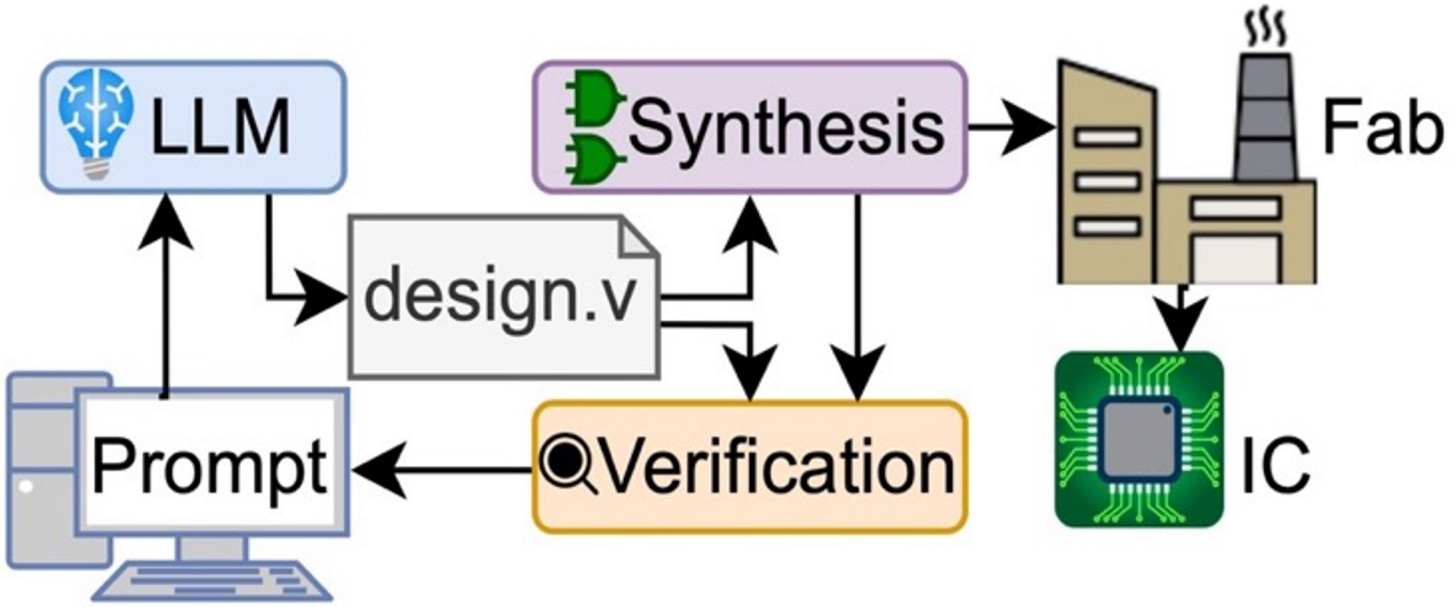


In this graph, we are plotting power against wire delay. The best way to look at this graph is to start at the labeled point at the top, which says Start Point.

1. *Start Point, where a basic quick layout is achieved*
2. *Customer Target, what the customer would be happy with*
3. *Best Human Effort, where humans get to after several months*
4. *Best DSO result (untrained), where AI can get to in just 24 hours*

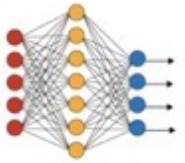
Result of Synopsys DSO.ai for optimizing design (from https://www.anandtech.com/show/16784/using-ai-to-build-processors-google-was-just-the-start-says-synopsys)

2023: Can Large Language Models create chips: Chip-Chat experiment



Can conversational LLMs be used to iteratively design hardware?

From https://arxiv.org/abs/2305.13243



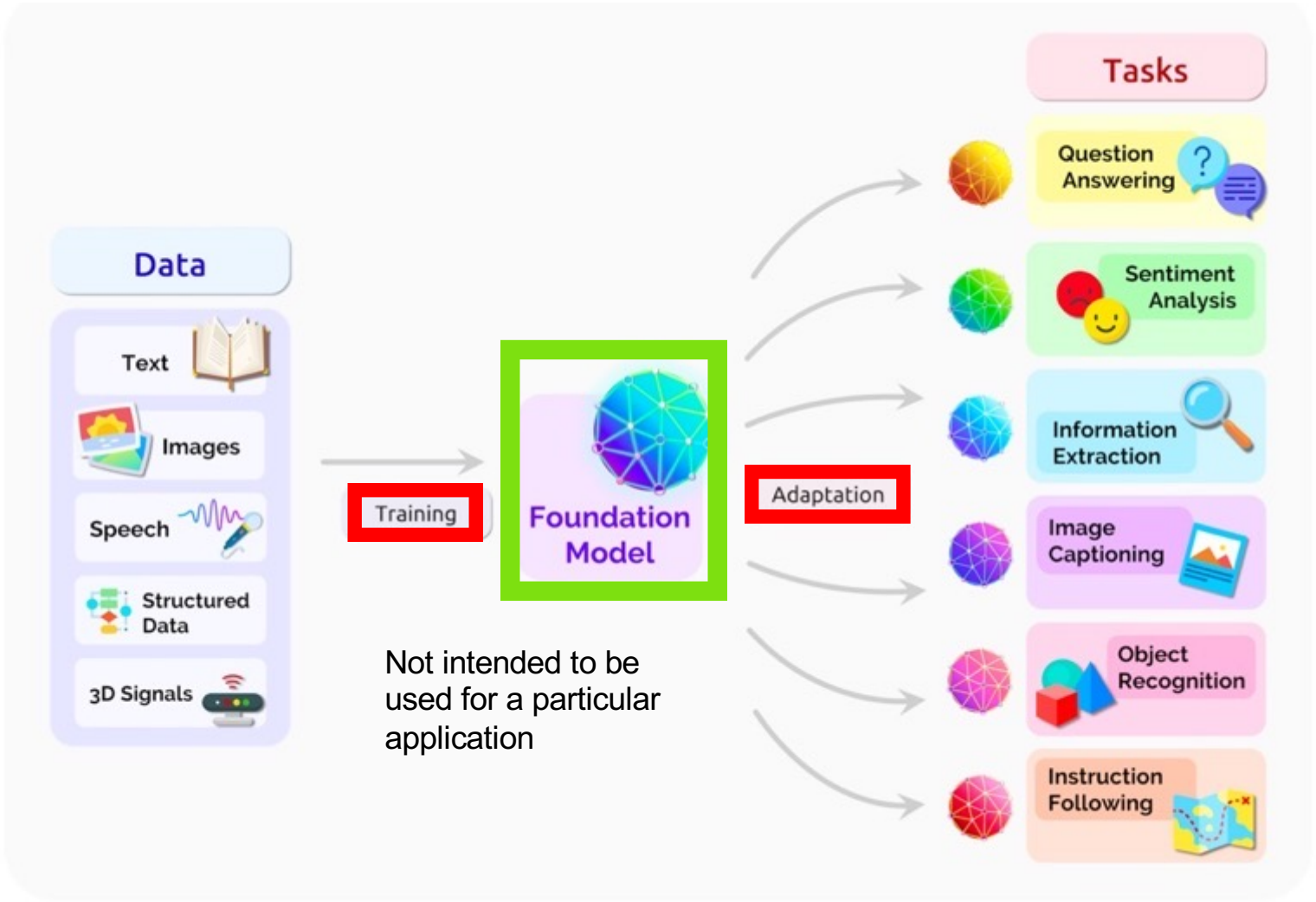
2023: Can Large Language Models create chips: Chip-Chat conclusion

*"**Challenges:** While it is clear that using a conversational LLM to assist in designing and implementing a hardware device can be beneficial overall, **the technology is not yet able to consistently design hardware with only feedback from verification tools**. The current state-of-the-art models do not perform well enough at understanding and fixing the errors presented by these tools to create complete designs and testbenches with only an initial human interaction.*

***Opportunities:** Still, **when the human feedback is provided** to the more capable ChatGPT-4 model, or it is used to co-design, **the language model seems to be a 'force multiplier', allowing for rapid design space exploration and iteration.** In general, ChatGPT-4 could produce functionally correct code, which could free up designer time when implementing common modules. Potential future work could involve a larger user study to investigate this potential, as well as the development of conversational LLMs specific to hardware design to improve upon the results."*

From https://arxiv.org/abs/2305.13243

FOUNDATION MODELS



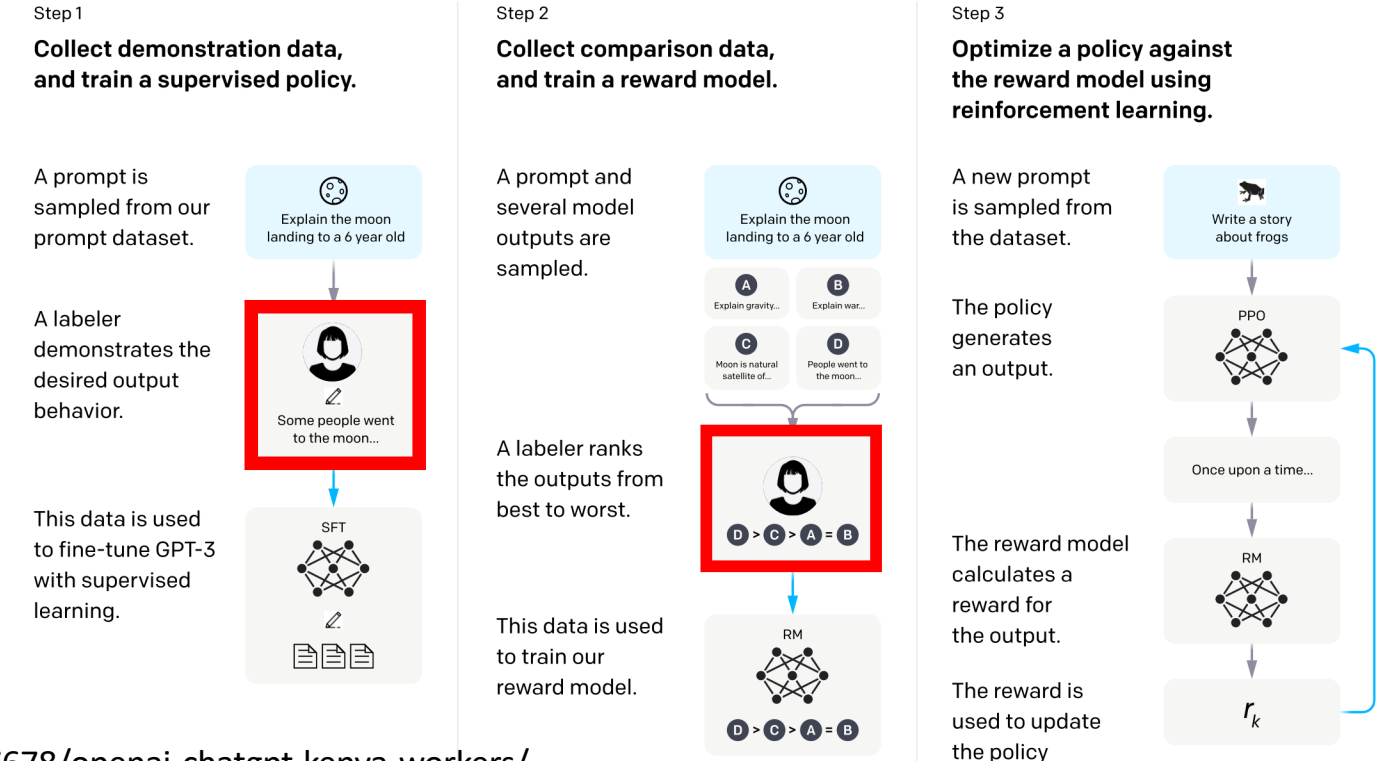
« *A **foundation model** can centralize the information from all the data from various modalities. This one model can then be adapted to a wide range of downstream tasks.* »
From « On the Opportunities and Risks of Foundation Models » https://arxiv.org/abs/2108.07258

2022: CHATGPT: REINFORCEMENT LEARNING WITH HUMAN FEEDBACK

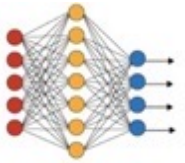
ChatGPT:

*« OpenAI's outsourcing partner in Kenya was **Sama**, a San Francisco-based firm that employs workers in Kenya, Uganda and India to label data for Silicon Valley clients like Google, Meta and Microsoft.»*

From https://time.com/6247678/openai-chatgpt-kenya-workers/



From https://openai.com/research/instruction-following



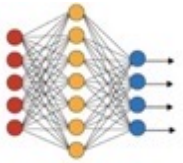
2022: Some results of ChatGPT: a fine-tuned version of GPT-3.5 that's essentially a general-purpose chatbot.

….performs better in understanding the 'context' behind a request and then using that to produce better content" while "hallucinating" less than GPT-3-based models. (Where it concerns text-generating AI, hallucination refers to an AI writing inconsistent, factually incorrect statements.)*

Explain what will be the next breakthroughs in computing

"It is difficult to predict exactly what the next breakthroughs in computing will be, as the field of computer science is constantly evolving and advancing. However, some areas that are likely to see significant developments in the near future include artificial intelligence, quantum computing, and the integration of advanced technologies such as machine learning and natural language processing into everyday devices. Additionally, there is likely to be continued progress in the development of more efficient and powerful computing hardware, as well as the expansion of high-speed networks and internet connectivity."

* https://techcrunch.com/2022/12/01/while-anticipation-builds-for-gpt-4-openai-quietly-releases-gpt-3-5/



Evolution of Generative Pre-trained Transformers. (GPT) in OpenAI

Model	Architecture	Parameter count	Training data	Release date	Training cost
GPT-1	12-level, 12-headed Transformer decoder (no encoder), followed by linear-softmax.	117 million	BookCorpus: 4.5 GB of text, from 7000 unpublished books of various genres.	June 11, **2018**	"1 month on 8 GPUs", or 1.7e19 FLOP.
GPT-2	GPT-1, but with modified normalization	1.5 billion	WebText: 40 GB of text, 8 million documents, from 45 million webpages upvoted on Reddit.	February 14, **2019** (initial/limited version) and November 5, 2019 (full version)	"tens of petaflop/s-day", or 1.5e21 FLOP.
GPT-3	GPT-2, but with modification to allow larger scaling	175 billion	499 Billion tokens consisting of CommonCrawl (570 GB), WebText, English Wikipedia, and two books corpora (Books1 and Books2).	May 28, **2020**	3640 petaflop/s-day, or 3.2e23 FLOP.
GPT-3.5	Undisclosed	175 billion	Undisclosed	March 15, 2022	Undisclosed
GPT-4	Also trained with both text prediction and RLHF; accepts both text and images as input. Further details are not public.	Undisclosed (1.8 trillon aka 1.8e12)	Undisclosed (13 trillon tokens, aka 1.3e13)	March 14, **2023**	Undisclosed. Estimated 2.1e25 FLOP.

Compute requirement

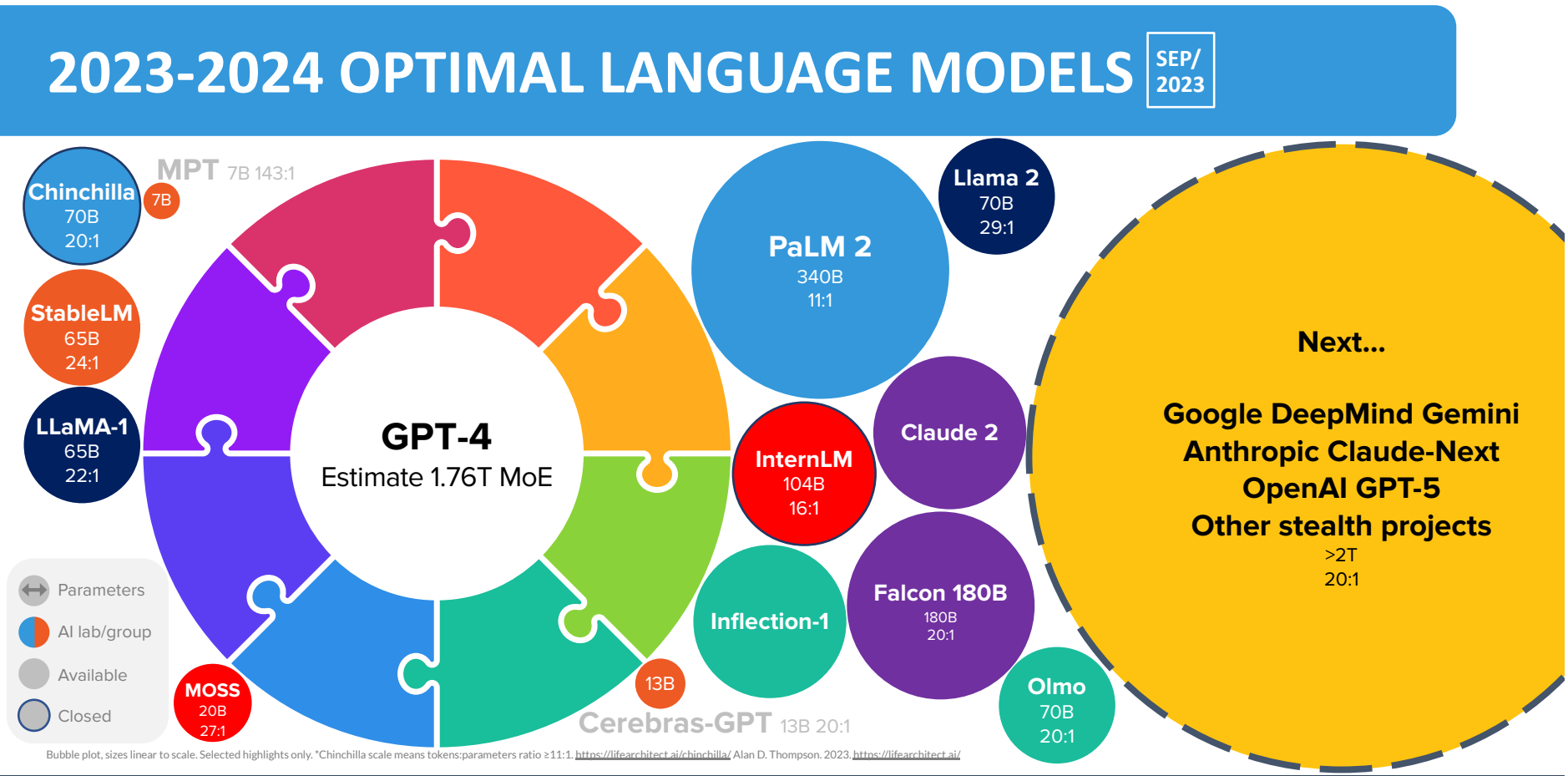
~ x 10

~ x 20

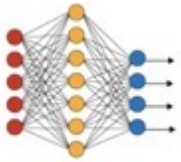
~ x 10

From https://en.wikipedia.org/wiki/Generative_pre-trained_transformer

Evolution of large Language models (LLMs)



From Dr Alan D. Thompson, LifeArchitect.ai, https://lifearchitect.ai/models/#laptop-models



2022: Bigger, not always better, but often "significantly undertrained"

Table 1 | **Current LLMs**. We show five of the current largest dense transformer models, their size, and the number of training tokens. Other than LaMDA (Thoppilan et al., 2022), most models are trained for approximately 300 billion tokens. We introduce *Chinchilla*, a substantially smaller model, trained for much longer than 300B tokens.

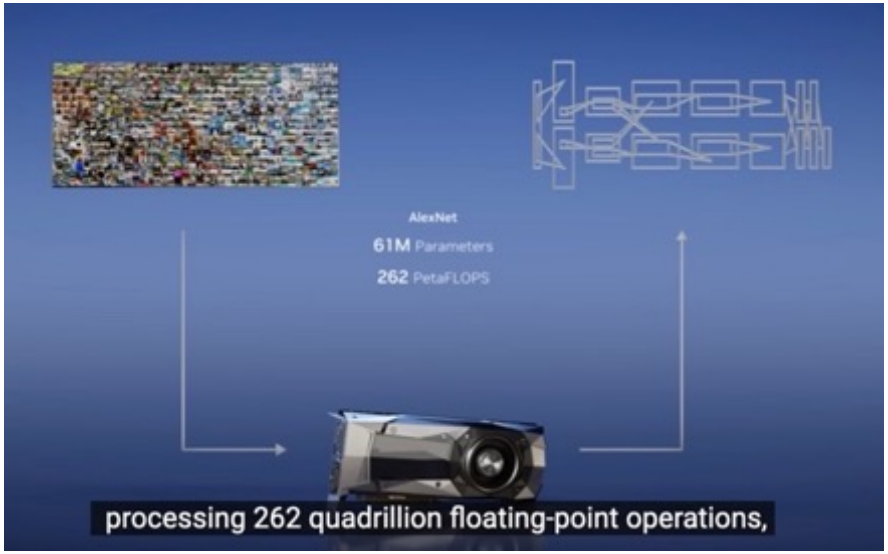
Model	Size (# Parameters)	Training Tokens
LaMDA (Thoppilan et al., 2022)	137 Billion	168 Billion
GPT-3 (Brown et al., 2020)	175 Billion	300 Billion
Jurassic (Lieber et al., 2021)	178 Billion	300 Billion
Gopher (Rae et al., 2021)	280 Billion	300 Billion
MT-NLG 530B (Smith et al., 2022)	530 Billion	270 Billion
Chinchilla	70 Billion	1.4 Trillion

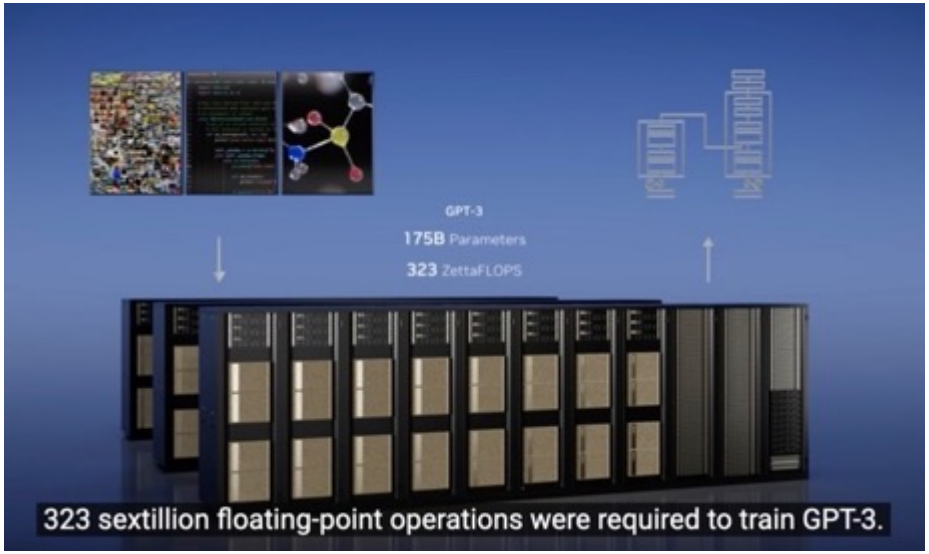
Hoffmann, J. et al. , "Training Compute-Optimal Large Language Models," DeepMind, 29 March 2022. Available: https://arxiv.org/abs/2203.15556.

Parameters	FLOPs	FLOPs (in *Gopher* unit)	Tokens
400 Million	1.92e+19	1/29,968	8.0 Billion
1 Billion	1.21e+20	1/4,761	20.2 Billion
10 Billion	1.23e+22	1/46	205.1 Billion
67 Billion	5.76e+23	1	1.5 Trillion
175 Billion	3.85e+24	6.7	3.7 Trillion
280 Billion	9.90e+24	17.2	5.9 Trillion
520 Billion	3.43e+25	59.5	11.0 Trillion
1 Trillion	1.27e+26	221.3	21.2 Trillion
10 Trillion	1.30e+28	22515.9	216.2 Trillion

Optimum number of tokens from training megamodels, and the corresponding compute power, according to Hoffmann, J. et al. , "Training Compute-Optimal Large Language Models," DeepMind, 29 March 2022. Available: https://arxiv.org/abs/2203.15556

Computing power is driving the advance of AI



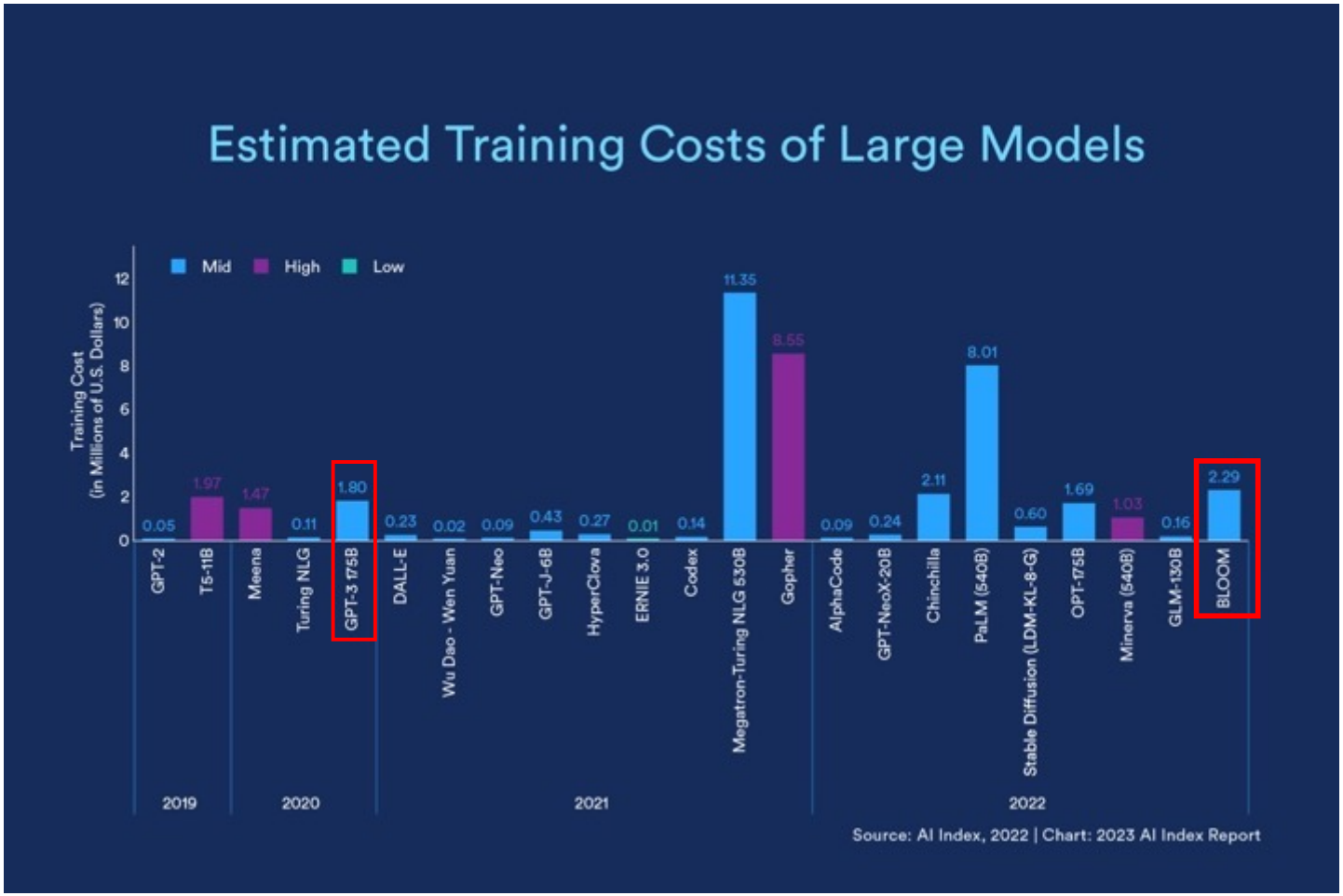


2012: AlexNet
GeForce GTX 580
Won ImageNet Challenge
262 x 10^{15} FLOPS

2020: GPT-3
323 x 10^{21} FLOPS
X 1 000 000 more floating point operations

From GTC 2023 Keynote with NVIDIA CEO Jensen Huang

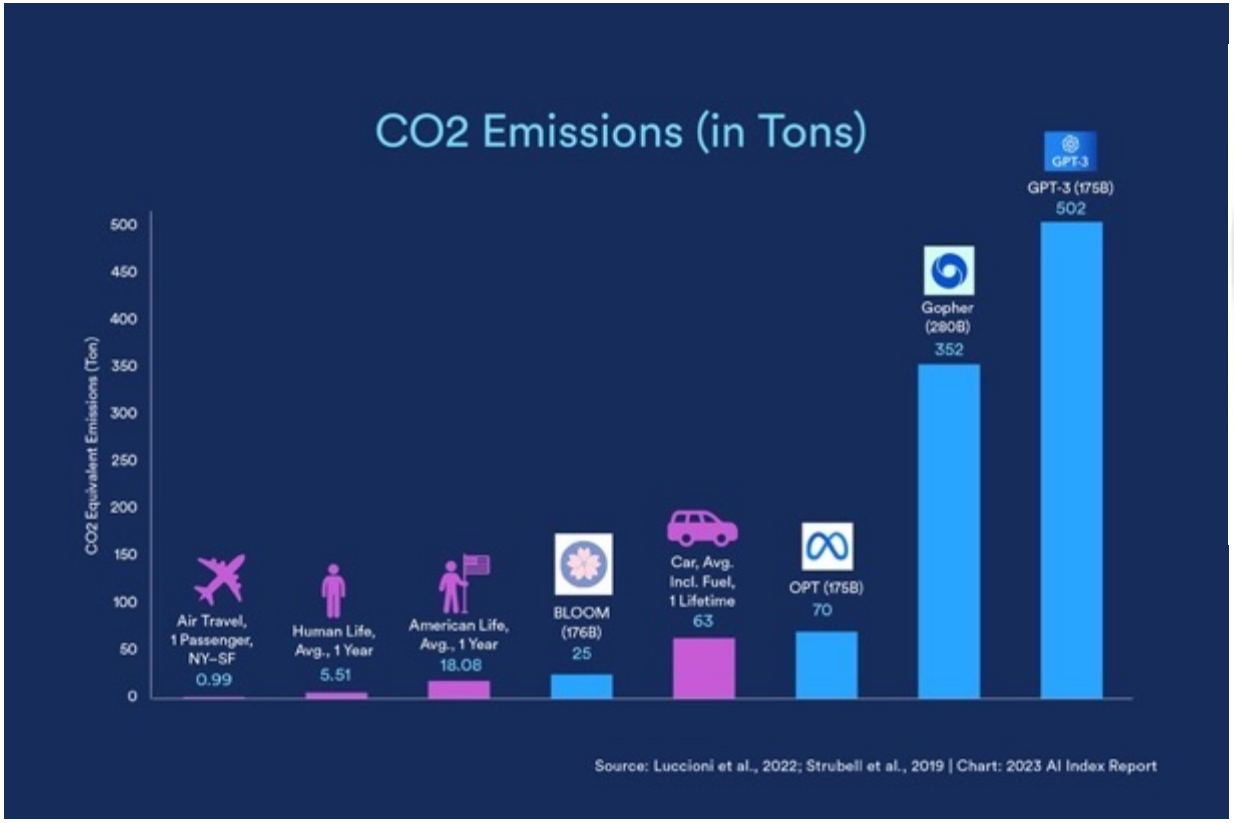
TRAINING LARGE LANGUAGE MODELS IS NOT CHEAP!



The estimated **training cost for GPT-4 is around $63 million**.

From “2023 State of AI in 14 Charts” available at https://hai.stanford.edu/news/2023-state-ai-14-charts

TRAINING LARGE LANGUAGE MODELS HAS AN ECOLOGICAL IMPACT

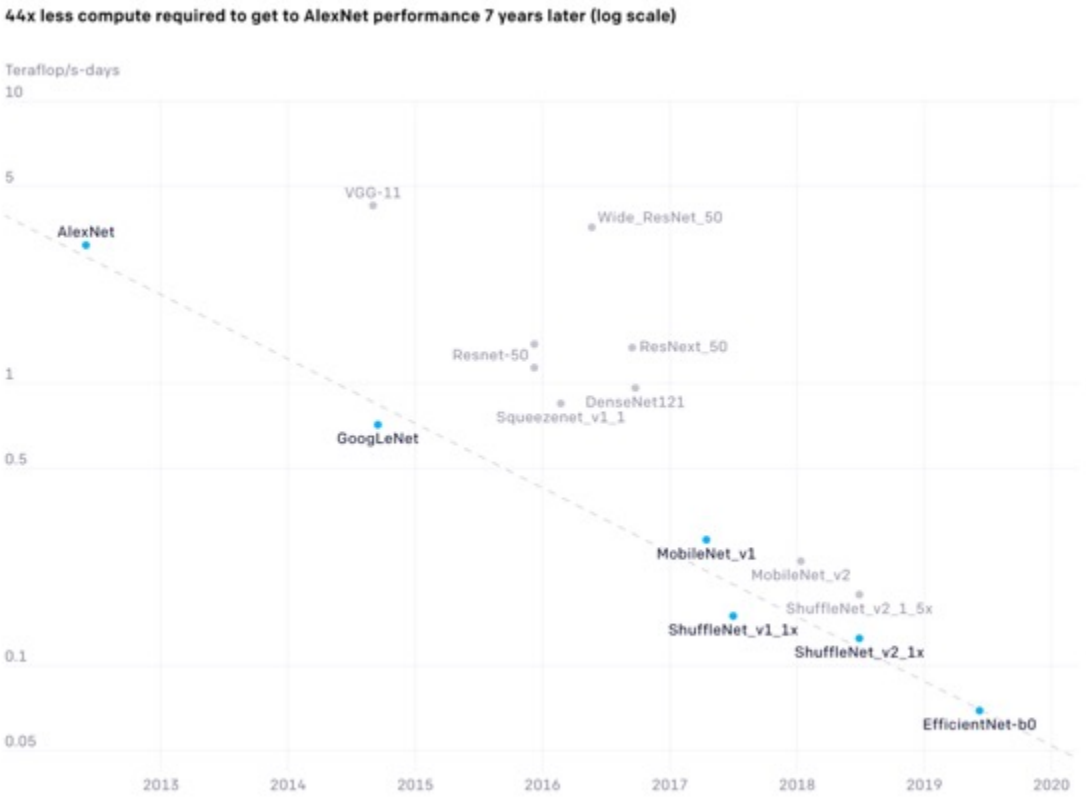


From https://www.hipeac.net/vision/#/latest/

From “2023 State of AI in 14 Charts” available at https://hai.stanford.edu/news/2023-state-ai-14-charts

Algorithmic improvement is a key factor driving the advance of AI

"Since 2012 the amount of compute needed to train a neural net to the same performance on ImageNet1 classification has been decreasing by a factor of 2 every 16 months. *"



GPT-3: May 2020
Meta OPT: May 2022
1/7 of the CO_2 footprint of GPT-3
Similar performances

Google: … new models..
bringing 100GB of models in the cloud down to less than half a gigabyte.
With these new models, the AI that powers the Assistant can now run locally on your phone.

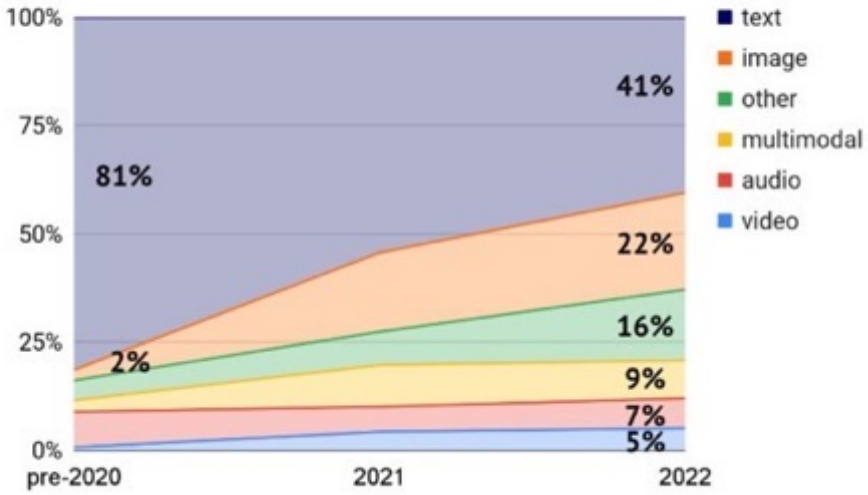
And it is often only inference at the edge,
So orders of magnitude less compute power is needed.

- [1] https://openai.com/blog/ai-and-efficiency/
- [2] Google IO , May 7th 2019

From text to multi-modal in 2 years

Transformers are becoming truly cross-modality

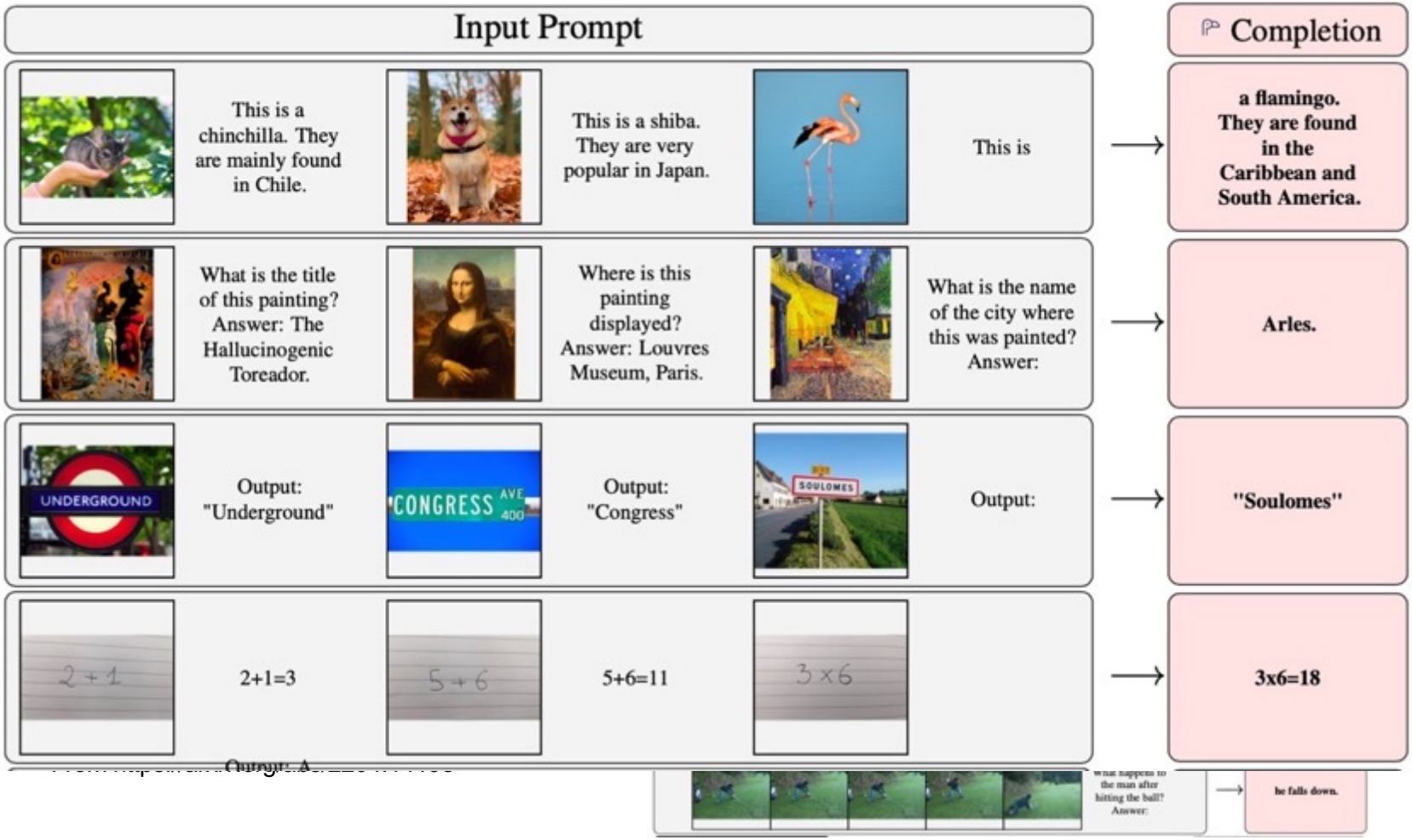
In the 2020 State of AI Report we predicted that transformers would expand beyond NLP to achieve state of the art in computer vision. It is now clear that transformers are a candidate general purpose architecture. Analysing transformer-related papers in 2022 shows just how ubiquitous this model architecture has become.



Zeta Alpha

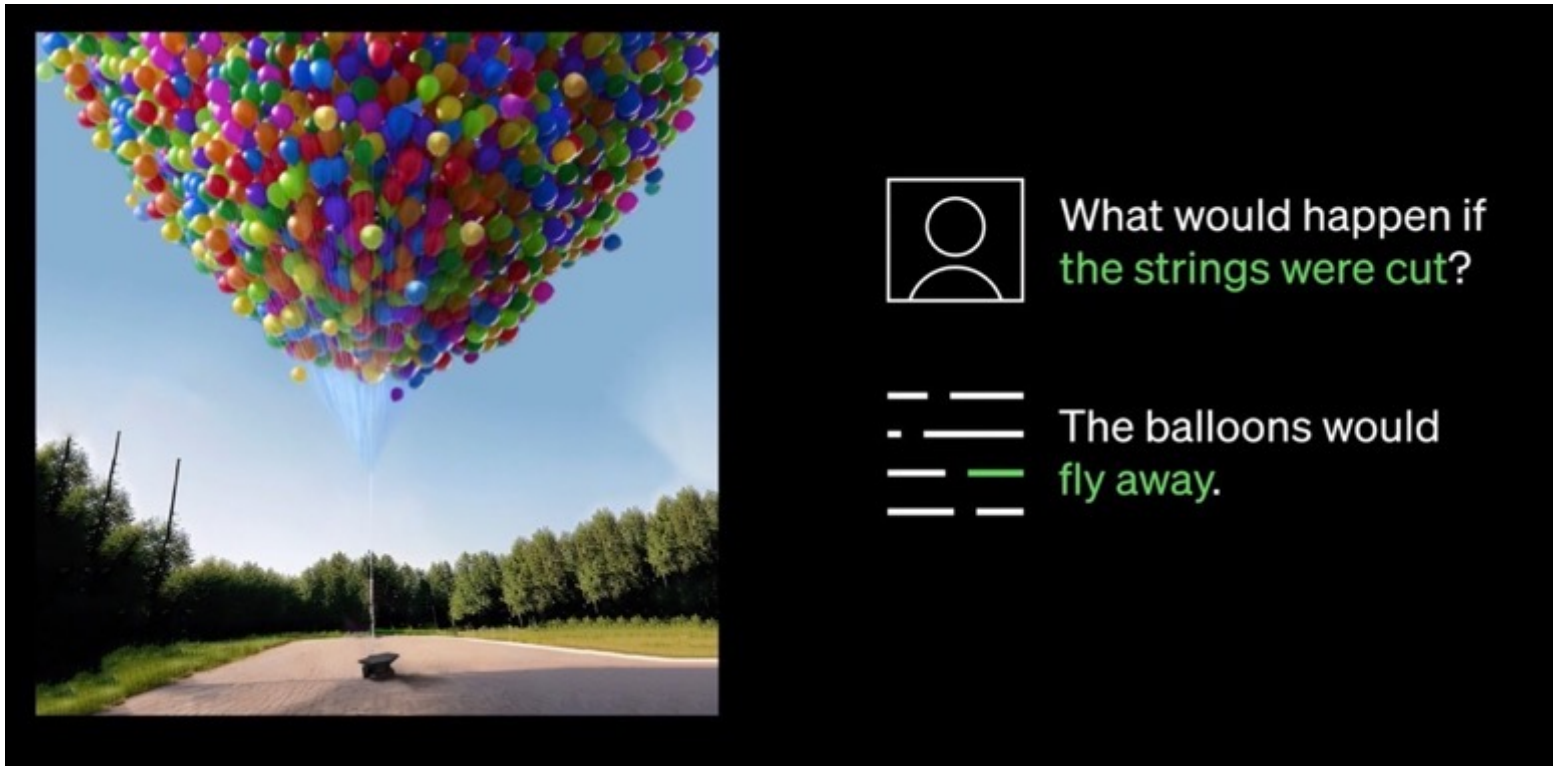
stateof.ai 2022

2022: Flamingo (Deepmind) : image + text + few shot learning



GPT-4 (OPENAI)

But also GPT-4 can Process images natively…



2023: BARD (GOOGLE)

C What would happen if the strings were cut?

View other drafts

If the strings were cut, the balloons would fall. The balloons are filled with helium, which is a lighter-than-air gas. This means that the balloons will rise until they reach a height where the air pressure is equal to the pressure inside the balloons. If the strings were cut, the balloons would no longer be tethered to the ground and would simply float away.

The image you sent shows a bunch of colorful balloons flying over a road. If the strings were cut, the balloons would likely drift away and eventually land somewhere else. They could land in a field, a park, or even a body of water. It's also possible that the balloons could get caught in power lines or other objects, which could cause them to pop.

It's important to be careful when releasing balloons, as they can pose a hazard to people and animals. If you're planning to release balloons, make sure to do it in a safe area where they won't be a nuisance or a danger.

From https://bard.google.com/

2023: The rise of Open-source models

ONE OF THE EARLY OPEN SOURCE LLM (MARCH-JULY 2022)

BigScience

BLOOM: open-source alternative to GPT-3

https://bigscience.huggingface.co

https://huggingface.co/bigscience/bloom

1.5TB of text, 350B tokens

43 languages, 16 programming languages

118 days of training on 384 A100 GPUs

Estimated cost of training: Equivalent of $2-5M in cloud

Server training location: Île-de-France, France

Environmental Impact: The training supercomputer, Jean Zay, uses mostly nuclear energy. The heat generated by it is reused for heating campus housing.

More details at https://huggingface.co/blog/bloom-megatron-deepspeed

Smaller versions are available : 560M, 1.1B, 1.7B, 3B, 7.1B

BLOOMZ models (same sizes) are fine-tuned for **instruction following**
https://huggingface.co/bigscience/bloomz

2023: LLAMA FROM META LEAKED ON LINE, TRIGGERING MULTIPLICITY OF DERIVED MODELS…

Pinned Tweet

Guillaume Lample @GuillaumeLample · Feb 24

Today we release LLaMA, 4 foundation models ranging from 7B to 65B parameters.
LLaMA-13B outperforms OPT and GPT-3 175B on most benchmarks.
LLaMA-65B is competitive with Chinchilla 70B and PaLM 540B.
The weights for all models are open and available at
research.facebook.com/publications/l...
1/n

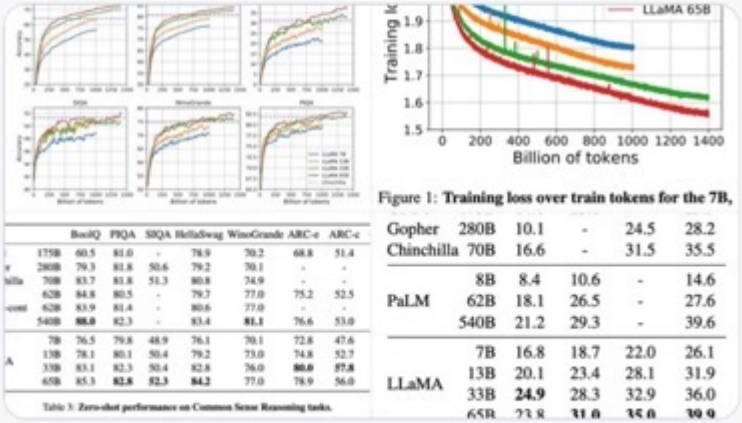


Figure 1: **Training loss over train tokens for the 7B,**

	BoolQ	PIQA	SIQA	HellaSwag	WinoGrande	ARC-e	ARC-c
175B	60.5	81.0	-	78.9	70.2	68.8	51.4
280B	79.3	81.8	50.6	79.2	70.1	-	-
70B	83.7	81.8	51.3	80.8	74.9	-	-
62B	84.8	80.5	-	79.7	77.0	75.2	52.5
62B	83.9	81.4	-	80.6	77.0	-	-
540B	**88.0**	82.3	-	83.4	**81.1**	76.6	53.0
7B	76.5	79.8	48.9	76.1	70.1	72.8	47.6
13B	78.1	80.1	50.4	79.2	73.0	74.8	52.7
33B	83.1	82.3	50.4	82.8	76.0	**80.0**	**57.8**
65B	85.3	**82.8**	**52.3**	**84.2**	77.0	78.9	56.0

Table 3: **Zero-shot performance on Common Sense Reasoning tasks.**

Gopher	280B	10.1	-	24.5	28.2
Chinchilla	70B	16.6	-	31.5	35.5
PaLM	8B	8.4	10.6	-	14.6
	62B	18.1	26.5	-	27.6
	540B	21.2	29.3	-	39.6
LLaMA	7B	16.8	18.7	22.0	26.1
	13B	20.1	23.4	28.1	31.9
	33B	**24.9**	28.3	32.9	36.0
	65B	23.8	**31.0**	**35.0**	**39.9**

173 1,837 6,993 3.1M

ARTIFICIAL INTELLIGENCE / TECH / REPORT

Meta's powerful AI language model has leaked online – what happens now?

Illustration: Alex Castro / The Verge

/ Meta's LLaMA model was created to help researchers but leaked on 4chan a week after it was announced. Some worry the technology will be used for harm; others say greater access will improve AI safety.

By James Vincent, a senior reporter who has covered AI, robotics, and more for eight years at The Verge.
Mar 8, 2023, 2:15 PM GMT+1 | 4 Comments / 4 New

If you buy something from a Verge link, Vox Media may earn a commission. See our ethics statement.

From https://www.theverge.com/2023/3/8/23629362/meta-ai-language-model-llama-leak-online-misuse

Most info can be found on https://ai.meta.com/blog/large-language-model-llama-meta-ai/

2023: THE TRIGGER: ALPACA FROM STANFORD

Alpaca: A Strong, Replicable Instruction-Following Model

Authors: Rohan Taori* and Ishaan Gulrajani* and Tianyi Zhang* and Yann Dubois* and Xuechen Li* and Carlos Guestrin and Percy Liang and Tatsunori B. Hashimoto

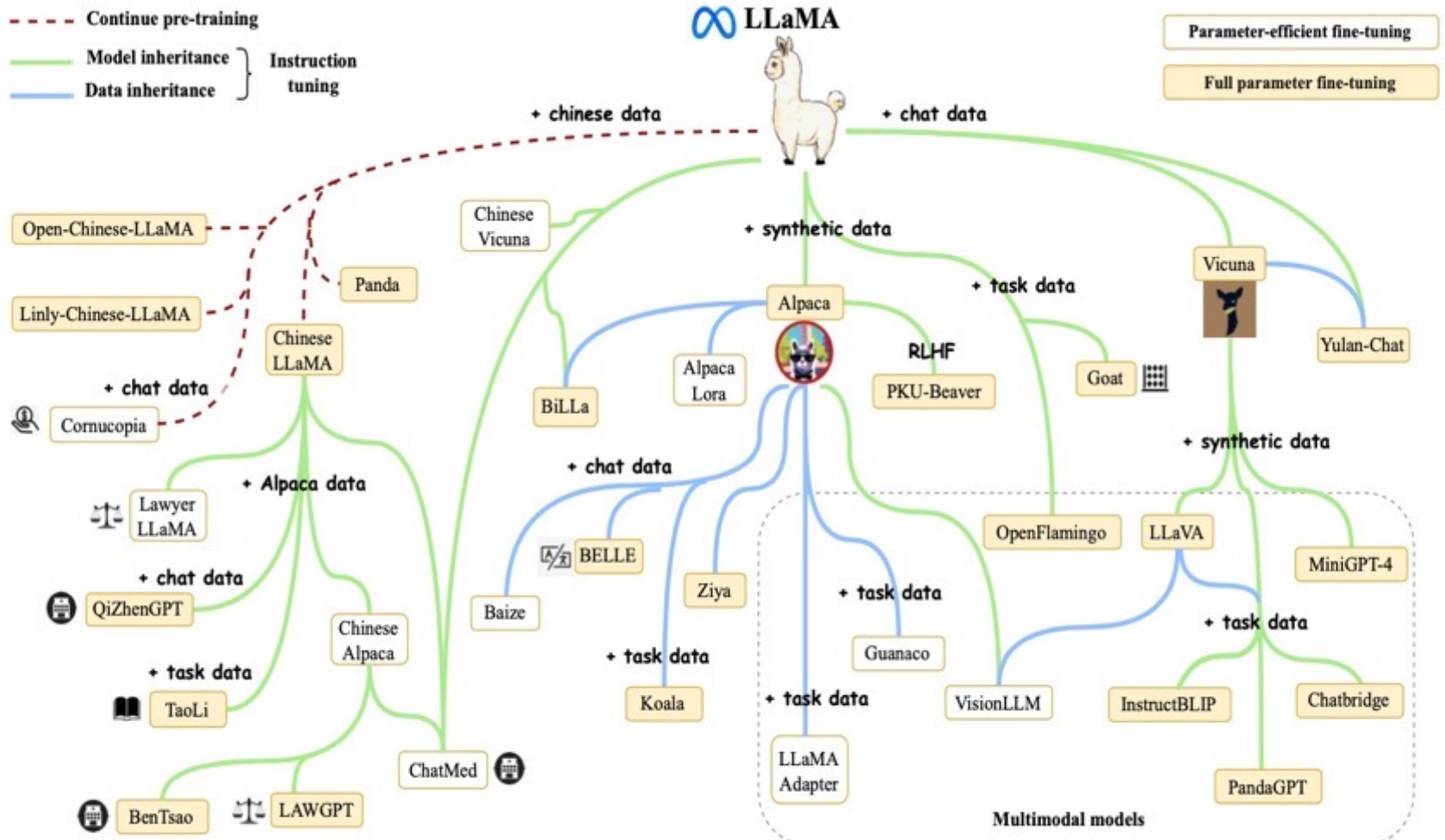
*We introduce **Alpaca 7B**, a model fine-tuned from the LLaMA 7B model on 52K instruction-following demonstrations. On our preliminary evaluation of single-turn instruction following, Alpaca behaves qualitatively similarly to OpenAI's text-davinci-003, while being surprisingly small and easy/cheap to reproduce (<600$). Checkout our code release on GitHub.*

Update: The public demo is now disabled. The original goal of releasing a demo was to disseminate our research in an accessible way. We feel that we have mostly achieved this goal, and given the hosting costs and the inadequacies of our content filters, we decided to bring down the demo.

Stanford Alpaca

From https://crfm.stanford.edu/2023/03/13/alpaca.html

2023: THE TRIGGER: ALPACA FROM STANFORD



From https://arxiv.org/abs/2303.18223

2023: THE TRIGGER: ALPACA FROM STANFORD

Introducing Llama 2

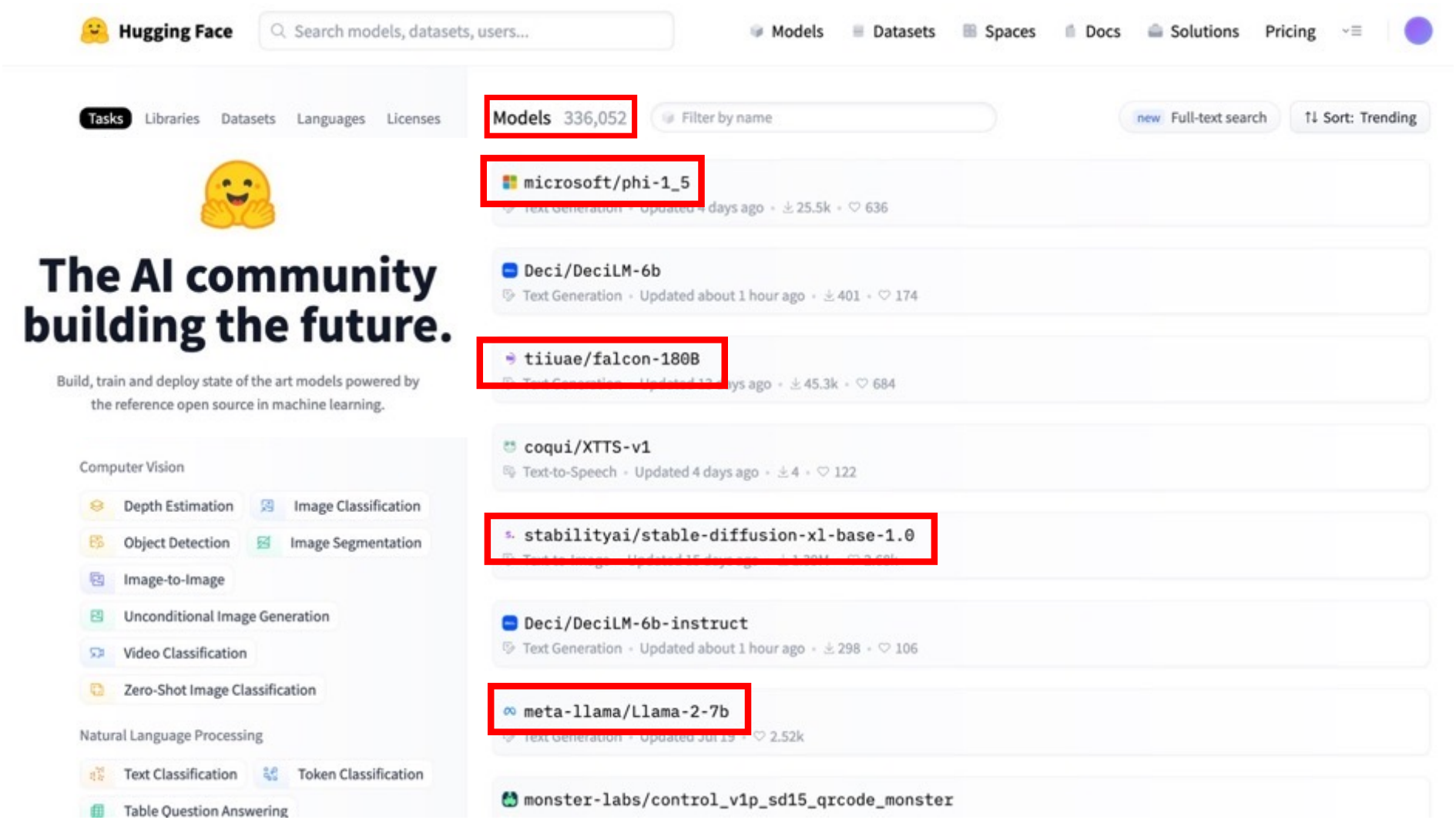
The next generation of our open source large language model

Llama 2 is available fo free for research and commercial use.

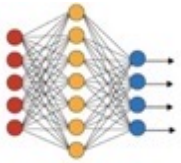
News from July 18th, 2023,
you can play with it on https://www.llama2.ai/, ***you can download and run it locally***
You keep your data locally and no fees to use it (unlike GPT-4, $20 a month)

From https://arxiv.org/abs/2303.18223

THE ENABLER: HUGGINGFACE



From https://huggingface.co/



STARCODER: A MODEL FOR HELPING CODING

Model Summary

The StarCoder models are 15.5B parameter models trained on 80+ programming languages from The Stack (v1.2), with opt-out requests excluded. The model uses Multi Query Attention, a context window of 8192 tokens, and was trained using the Fill-in-the-Middle objective on 1 trillion tokens.

- **Repository:** bigcode/Megatron-LM
- **Project Website:** bigcode-project.org
- **Paper:** 💫StarCoder: May the source be with you!
- **Point of Contact:** contact@bigcode-project.org
- **Languages:** 80+ Programming languages

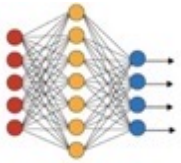
From https://huggingface.co/bigcode/starcoder

STARCODER: A MODEL FOR HELPING CODING

Write a Python function to tell me what the date is today	Sure thing. Here goes...
	`from datetime import date`
	`today = str(date.today()).replace('-', '/')`
	`print("Today is " + today)`

List the prime numbers between 20 and 30.	Here is a simple algorithm which uses trial division to find the prime numbers within a given interval.
	`primes = []`
	`for num in range(20, 31):`
	`  found = False`
	`  for j in range(2,num//2):`
	`  if ((num % j)==0):`
	`  found = True`
	`  break`
	`  if((found==False) & (num!= 1)):`
	`  primes.append(num)`

From https://arxiv.org/abs/2305.06161



LLM RUNNING LOCALLY ON A MAC MINI

```
* 29
> Write a program that lists the prime numbers between 20 and 30.
Here is a program in Python that lists the prime numbers between 20 and 30:
```
def is_prime(n):
    if n <= 1 or n % 2 == 0:
        return False
    for i in range(3, int(n ** 0.5) + 1):
        if n % i == 0:
            return False
    return True

def list_primes(start, end):
    prime_list = []
    for i in range(start, end + 1):
        if is_prime(i):
            prime_list.append(i)
    return prime_list

print(list_primes(20, 30))
```
This program uses the `is_prime` function to check whether a given number is pri
me or not. The `is_prime` function checks if the number is less than or equal to
 1, or
```

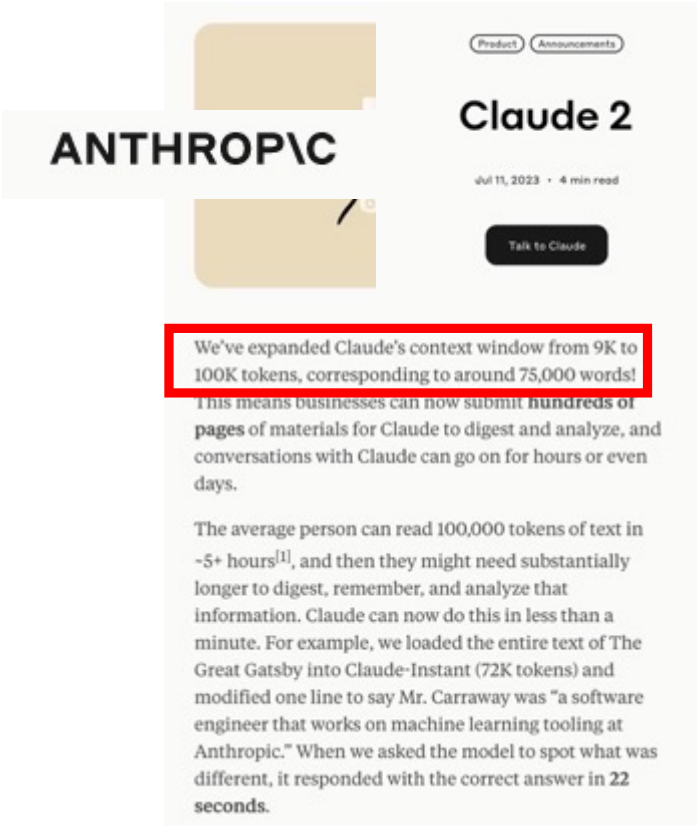
2024?: LLM RUNNING LOCALLY ON YOUR DEVICE

Qualcomm Technologies, Inc. and Meta are working to optimize the execution of Meta's Llama 2 large language models directly on-device – without relying on the sole use of cloud services.

From https://www.qualcomm.com/news/releases/2023/07/qualcomm-works-with-meta-to-enable-on-device-ai-applications-usi

2023: INCREASING THE NUMBER OF INPUT TOKENS

"The average person can read 100,000 tokens of text in ~5+ hours[1], and then they might need substantially longer to digest, remember, and analyze that information. Claude can now do this in less than a minute. For example, we loaded the ***entire text of The Great Gatsby*** *into Claude-Instant (72K tokens) and modified one line to say Mr. Carraway was "a software engineer that works on machine learning tooling at Anthropic." When we asked the model to spot what was different, it responded with the correct answer in 22 seconds.*



From https://www.anthropic.com/index/100k-context-windows

2023: VOYAGER: AN OPEN-ENDED EMBODIED AGENT WITH LARGE LANGUAGE MODELS

Voyager is evolving in the Minecraft game, continuously explores the world, acquires diverse skills, and makes novel discoveries without human intervention.

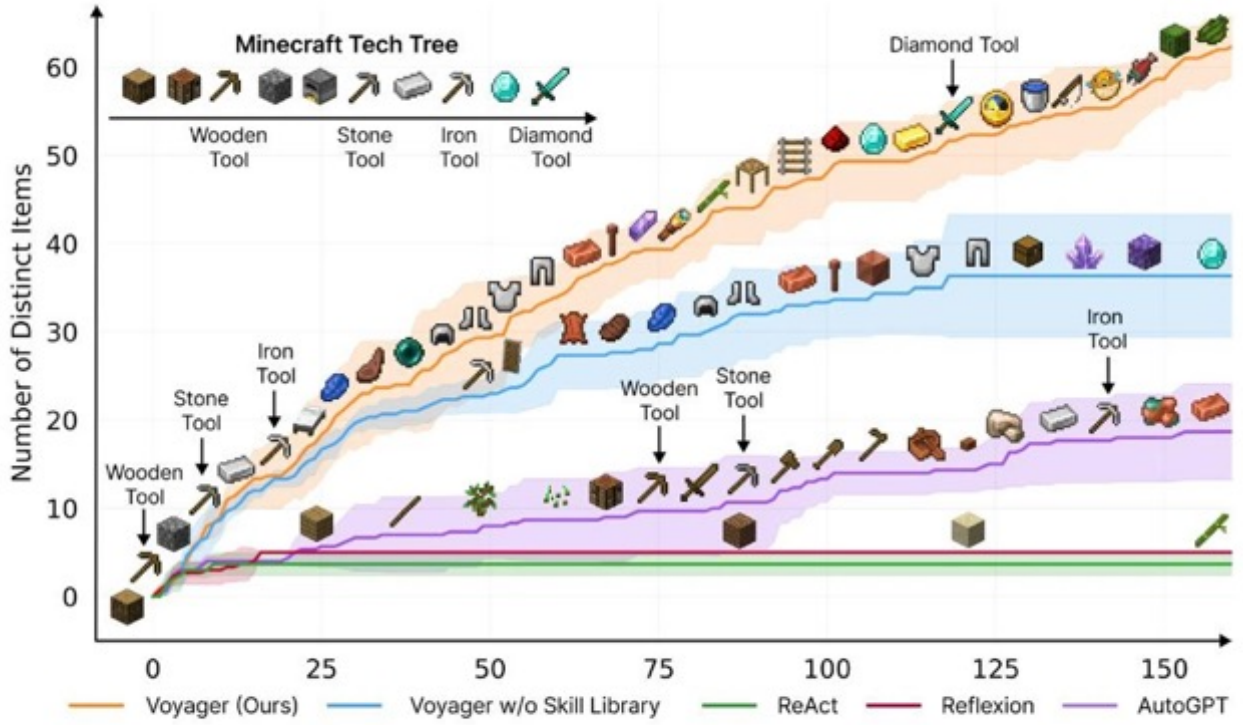
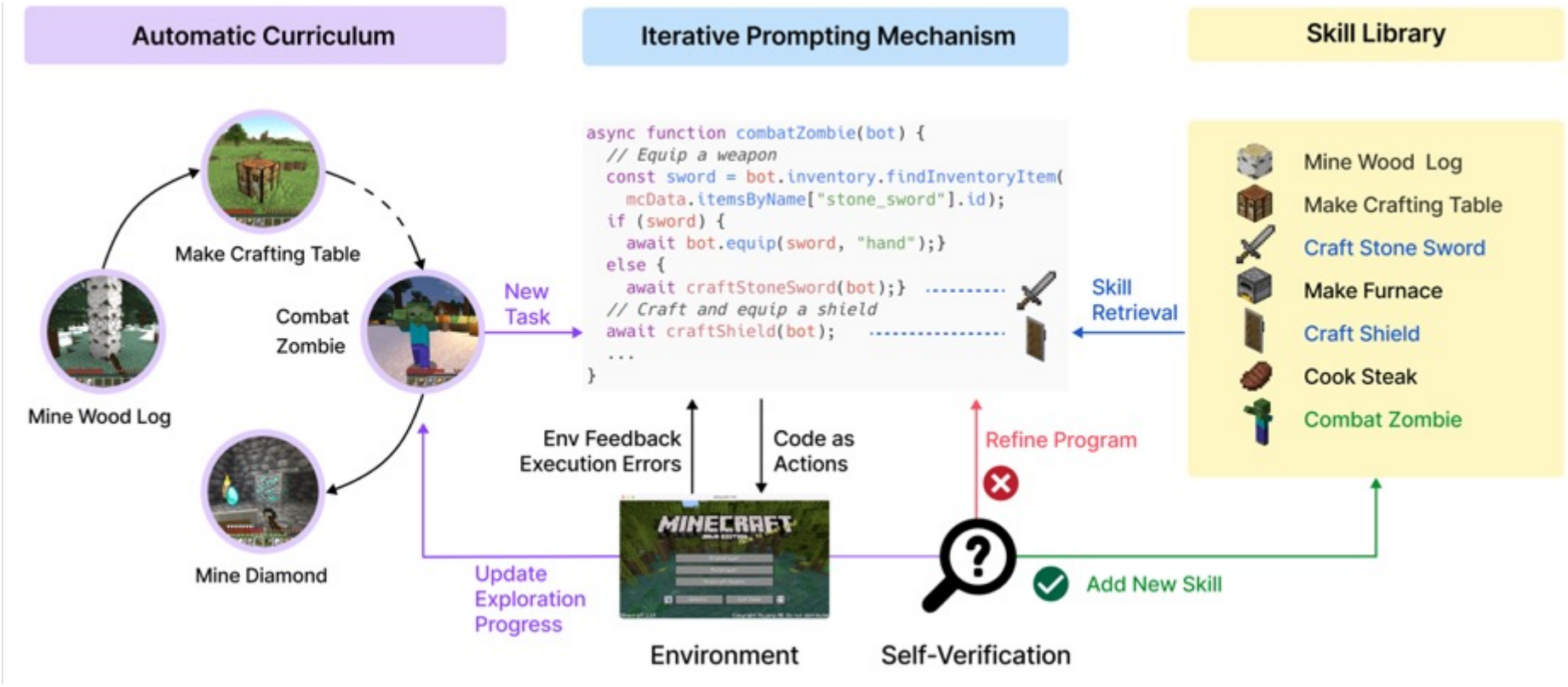


Figure 1: VOYAGER discovers new Minecraft items and skills continually by self-driven exploration, significantly outperforming the baselines. X-axis denotes the number of prompting iterations.

From: https://github.com/MineDojo/Voyager or https://voyager.minedojo.org/

2023: VOYAGER: AN OPEN-ENDED EMBODIED AGENT WITH LARGE LANGUAGE MODELS



From: https://github.com/MineDojo/Voyager

2023: PALM-E (GOOGLE)

PaLM-E: An Embodied Multimodal Language Model

Danny Driess [1 2] Fei Xia [1] Mehdi S. M. Sajjadi [3] Corey Lynch [1] Aakanksha Chowdhery [3]
Brian Ichter [1] Ayzaan Wahid [1] Jonathan Tompson [1] Quan Vuong [1] Tianhe Yu [1] Wenlong Huang [1]
Yevgen Chebotar [1] Pierre Sermanet [1] Daniel Duckworth [3] Sergey Levine [1] Vincent Vanhoucke [1]
Karol Hausman [1] Marc Toussaint [2] Klaus Greff [3] Andy Zeng [1] Igor Mordatch [3] Pete Florence [1]

[1]Robotics at Google [2]TU Berlin [3]Google Research

https://palm-e.github.io

Figure 1: PaLM-E is a single general-purpose multimodal language model for embodied reasoning tasks, visual-language tasks, and language tasks. PaLM-E *transfers* knowledge from visual-language domains into embodied reasoning – from robot planning in environments with complex dynamics and physical constraints, to answering questions about the observable world. PaLM-E operates on *multimodal sentences*, i.e. sequences of tokens where inputs from arbitrary modalities (e.g. images, neural 3D representations, or states, in green and blue) are inserted alongside text tokens (in orange) as input to an LLM, trained end-to-end.

From https://palm-e.github.io/

PALM-E: AN EMBODIED MULTIMODAL LANGUAGE MODEL

Robot: I am a robot operating in a kitchen. Given **img**, When a human asks me to do a task, I will respond with the sequence of actions I would do to accomplish the task with only the items I see. Human: Use all of the ingredients you see to make a cake batter.

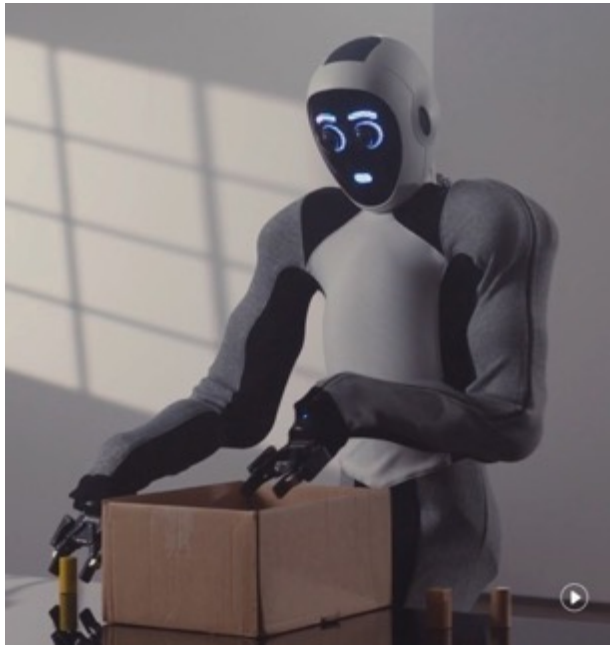
1. crack egg. 2. put egg in bowl. 3. put flour in bowl. 4. put sugar in bowl. 5. mix. 6. put in pan. 7. bake. 8. eat. 9. clean up.

From https://palm-e.github.io/

NEXT POSSIBLE STEPS FOR LLMS?

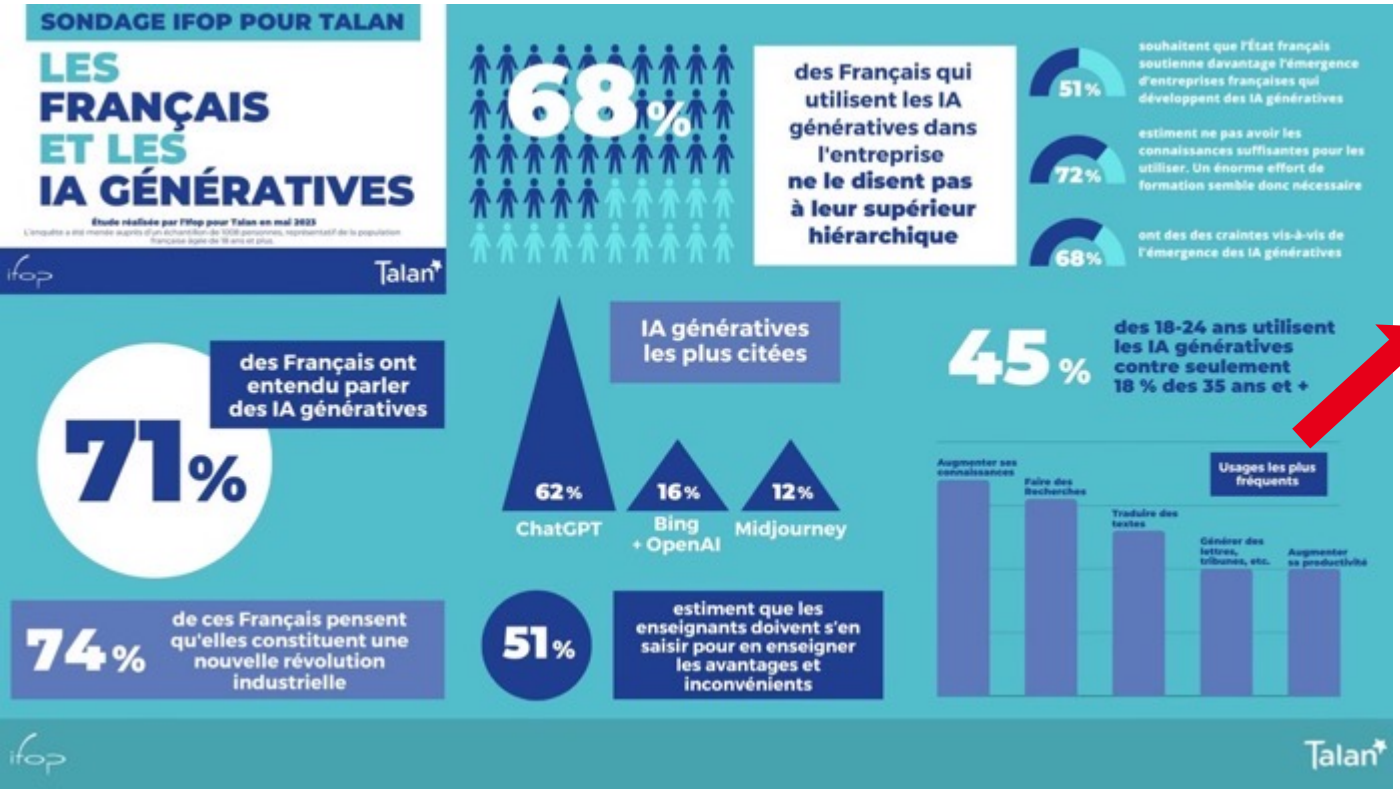
- LLMs ***learned the world indirectly*** by texts
- With multimodality, they will learn it by images and other modalities
- They are already linked to simulators (“digital twins”) to ***experience*** the laws of physics (e.g. Nvidia Isaac sim)
- They will be proactive when they don’t know (e.g. https://arxiv.org/abs/2307.01928)
- Next step will be that they could ***experience the world directly by embodiment.***

From https://www.youtube.com/watch?v=VW-dOMBFj7o



From https://www.1x.tech/

FRENCH PEOPLE AND GENERATIVE AI

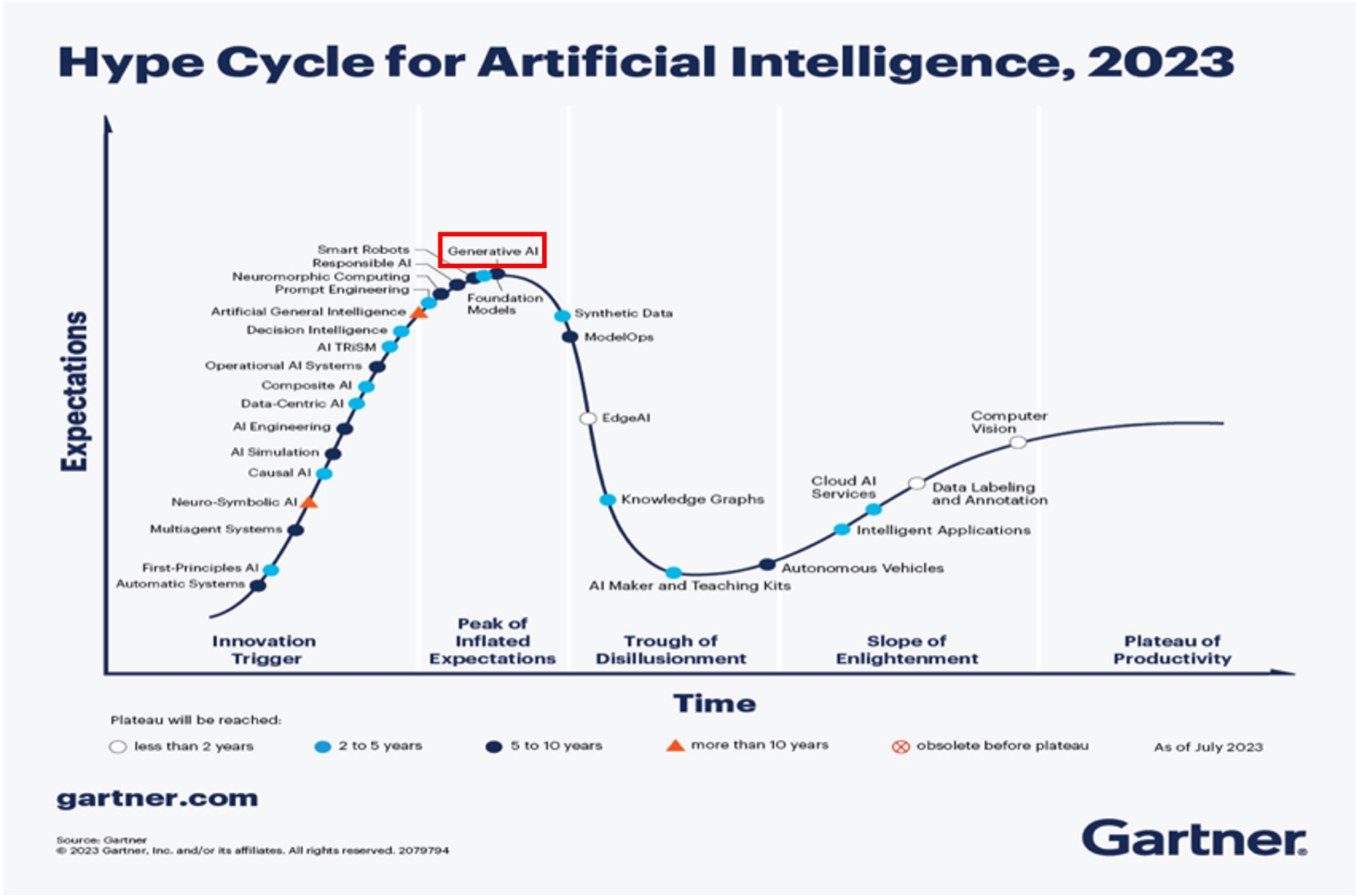


Most frequent utilizations:

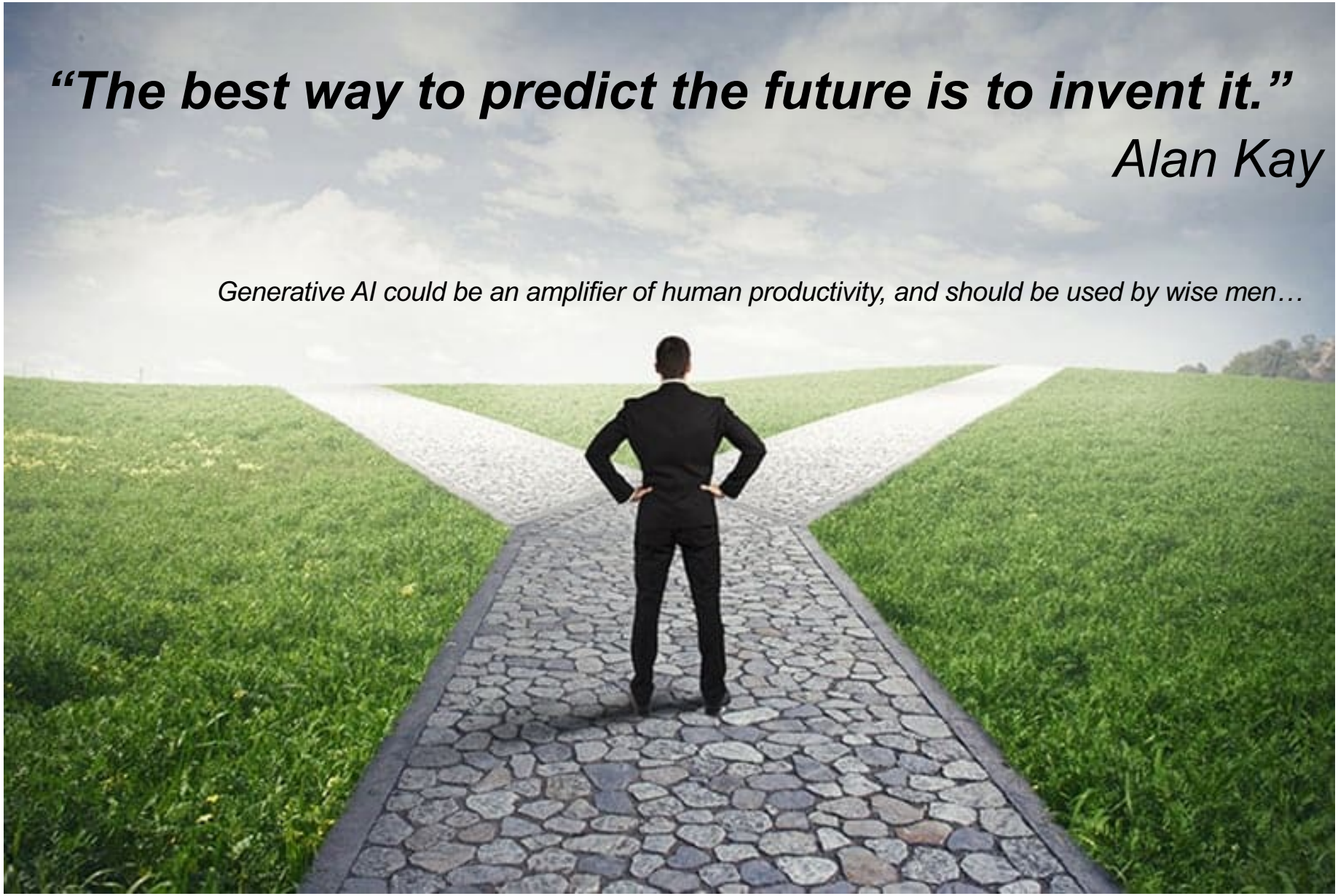
- **Increase knowledge**
- **Information search**
- **Translate texts**
- **Create letters, blogs, …**
- **Increase its own productivity**
- **…**

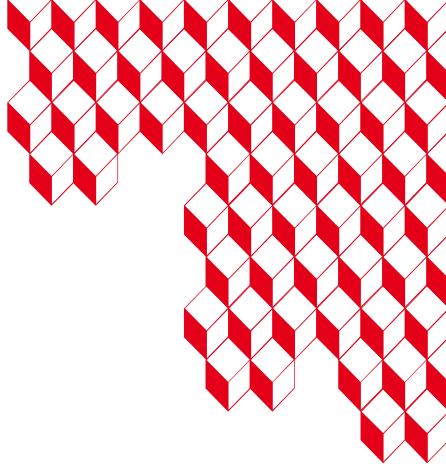
From https://talan.com/actualites/detail-actualites/news/sondage-ifop-talan-les-francais-et-les-ia-generatives/
Full report available from https://www.ifop.com/publication/le-regard-des-francais-et-des-actifs-sur-les-ia-generatives/

AI in the hype cycle



CONCLUSION: WE LIVE AN EXCITING TIME!





THANK YOU FOR YOUR ATTENTION!

CEA

marc.duranton@cea.fr