

A Brief History of Neural Networks

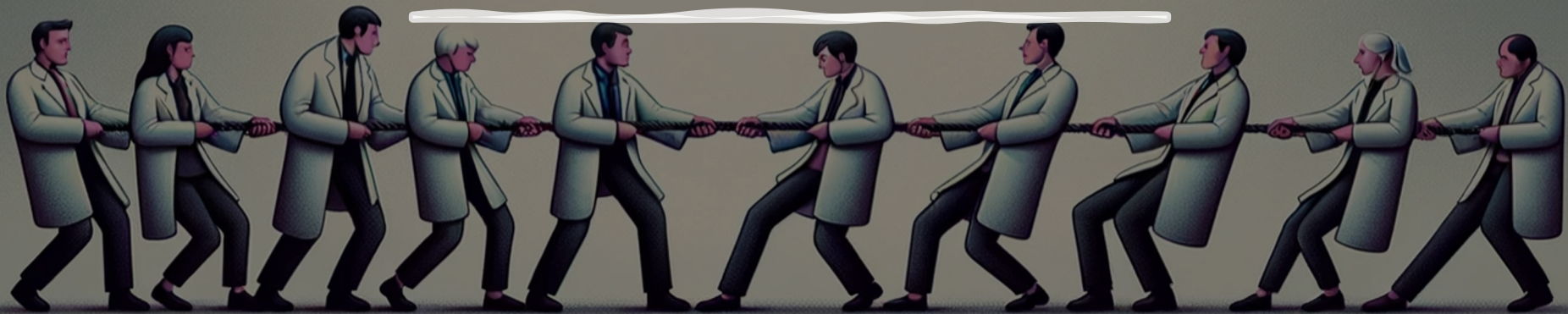
Zachary Lipton & Henry Chai

October 23rd, 10701

email: zlipton@cmu.edu



Strikingly interdisciplinary
stakeholders, diverse aims



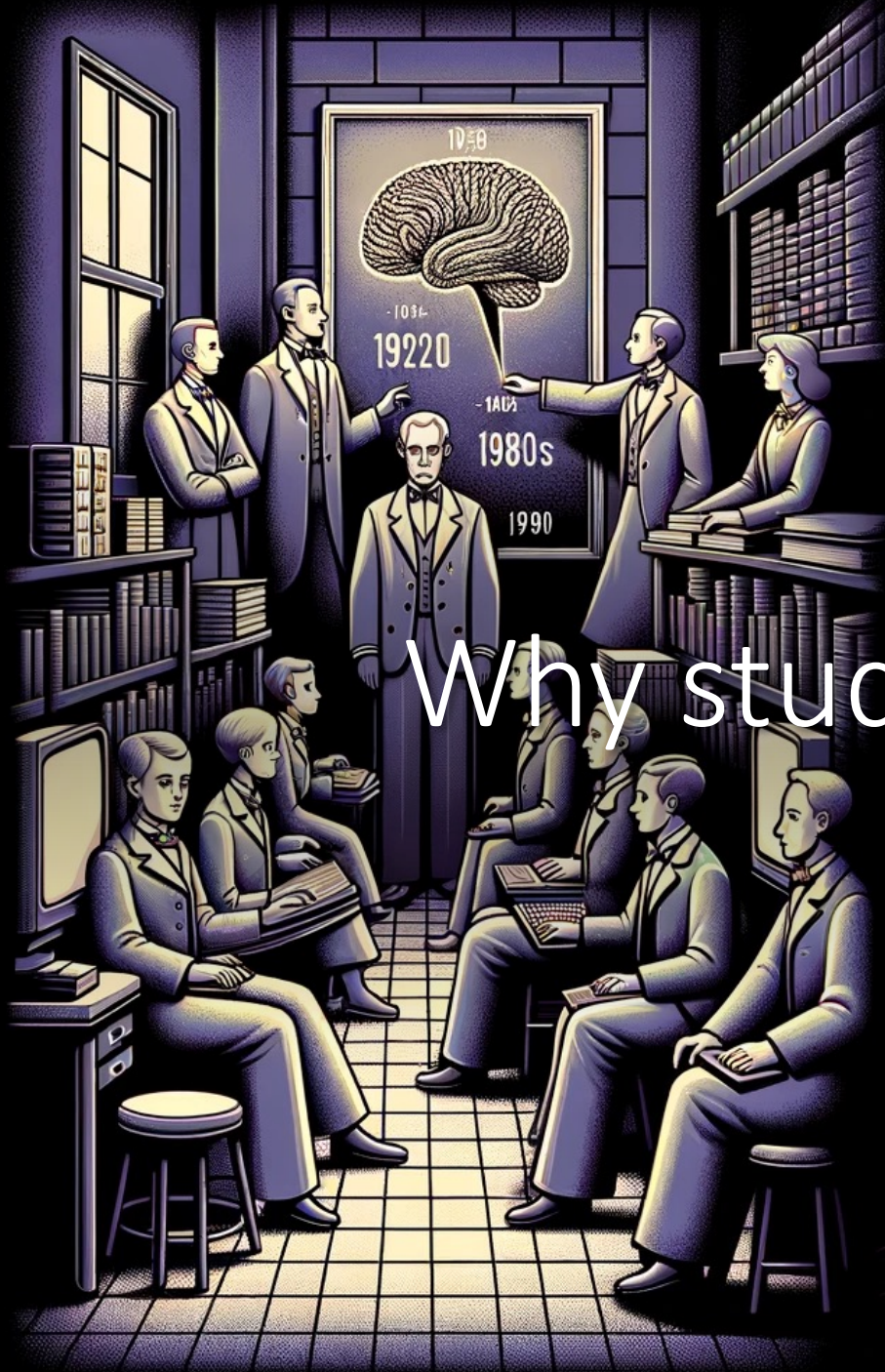
Participating Disciplines

- Psychology
- Cognitive science
- Neuroscience
- Machine Learning
- Statistics
- Physics
- Mathematics
- Philosophy

Aims:

- Understand biological brains
- Simulate biological brains
- Applications in AI
- Non-parametric hammer for statistical inference





Why study the history?



Why study the history?

• Technology as it happens

- A complicated process
- Competing ideas
- Many different goals
- Frequent rediscovery
- Promising directions abandoned
- Many important ideas premature
- Massive uncertainty
- Guided by intuition

• Technology in the textbooks

- Story told by the victors
- Can forget there even was a story
- Declarative “here is a thing, here is another thing”
- Hard to figure out “why these things”?
- Straight lines from past to present
- Intuition often lost / buried

Why especially for neural networks

- Everything is {new/old} is {old/new}
- The field moves fast! (*but not in a straight line*)
- Mathematical facts are universal, but engineering facts are ephemeral
- NN knowledge consists of a lot of **what**, not a lot of **why**
- The literature has never been settled



Everything {old/new} is {new/old}

- ReLU activations (1967 → 2010)
- Backprop (1763 → 1960 → 1962 → 1970 → 1982 → 1986)
- Data Augmentation (1958 → 2012)
- The deep-ification of everything:
 - Q-learning (1989) → Deep Q-learning (2013)
 - Double Q-learning (2010) → Deep Double Q-learning (2015)

Engineering facts change quickly!

What's the best way to train an image classifier?

- 1958 → Perceptron
- 1980 → Optics-based features
- 1989 → Convolutional Neural Networks
- 1990s → Optical features + SVMs
- 2010 → Pre-trained unsupervised nets, fine-tuned to labeled data
- 2012–2020 → Supervised models trained from scratch, or fine-tuned from larger supervised models
- 2020s → Pre-trained unsupervised and/or (differently supervised) multimodal nets, fine-tuned to labeled data

What's the best way to build a text classifier?

- Rules!
- 1990-2010: linear models with TF-IDF / ngram features!
- 2014: LSTM neural networks
- 2015: Convolutional neural networks
- 2016: LSTM neural networks
- 2017: pre-trained LSTM neural networks, finetuned on labeled data
- 2018: pre-trained Transformer models, fine-tuned on labeled data
- 2023: Just ask ChatGPT!




Every field has dirty laundry,
deep learning wears it proudly

Goals for the second half of semester

- Cover prevailing ideas and methods in deep learning / generative AI.
- Dig into the key results that have driven practice.
- Cover competing ideas, including recently outmoded techniques.
- Present material through the lens of a living, breathing literature.
- Prepare you to ingest new knowledge as it arrives



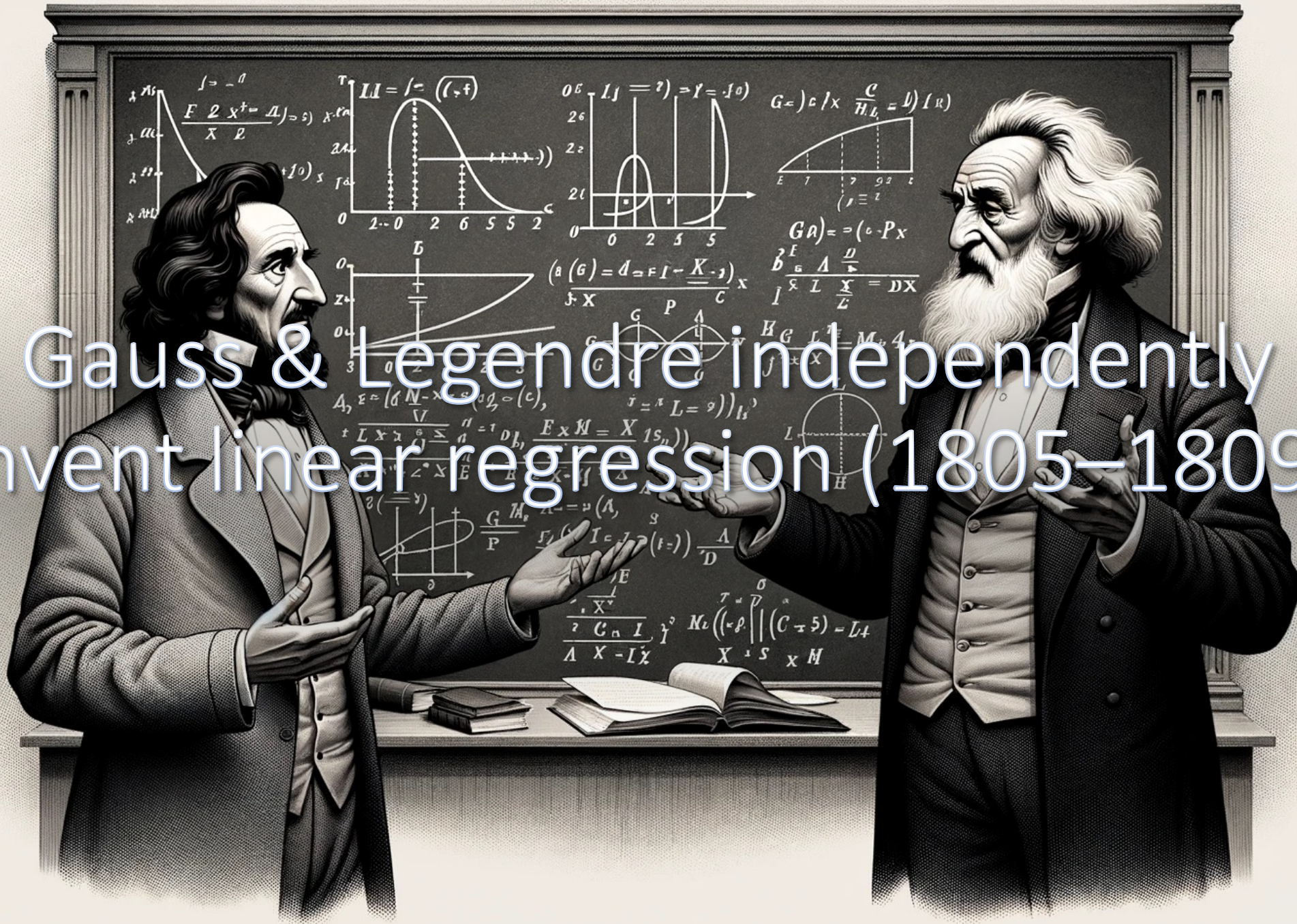
The background is a complex, multi-layered digital landscape. It features a central perspective view of a city-like structure made of glowing circuitry and data lines, receding into the distance. The sky is filled with abstract, swirling patterns of light and color, resembling a nebula or a complex data visualization. The overall color palette is dominated by blues, greens, and oranges, with a warm, golden glow emanating from the center.

A Super-brief History of Machine Intelligence (& Mostly, Neural Nets)



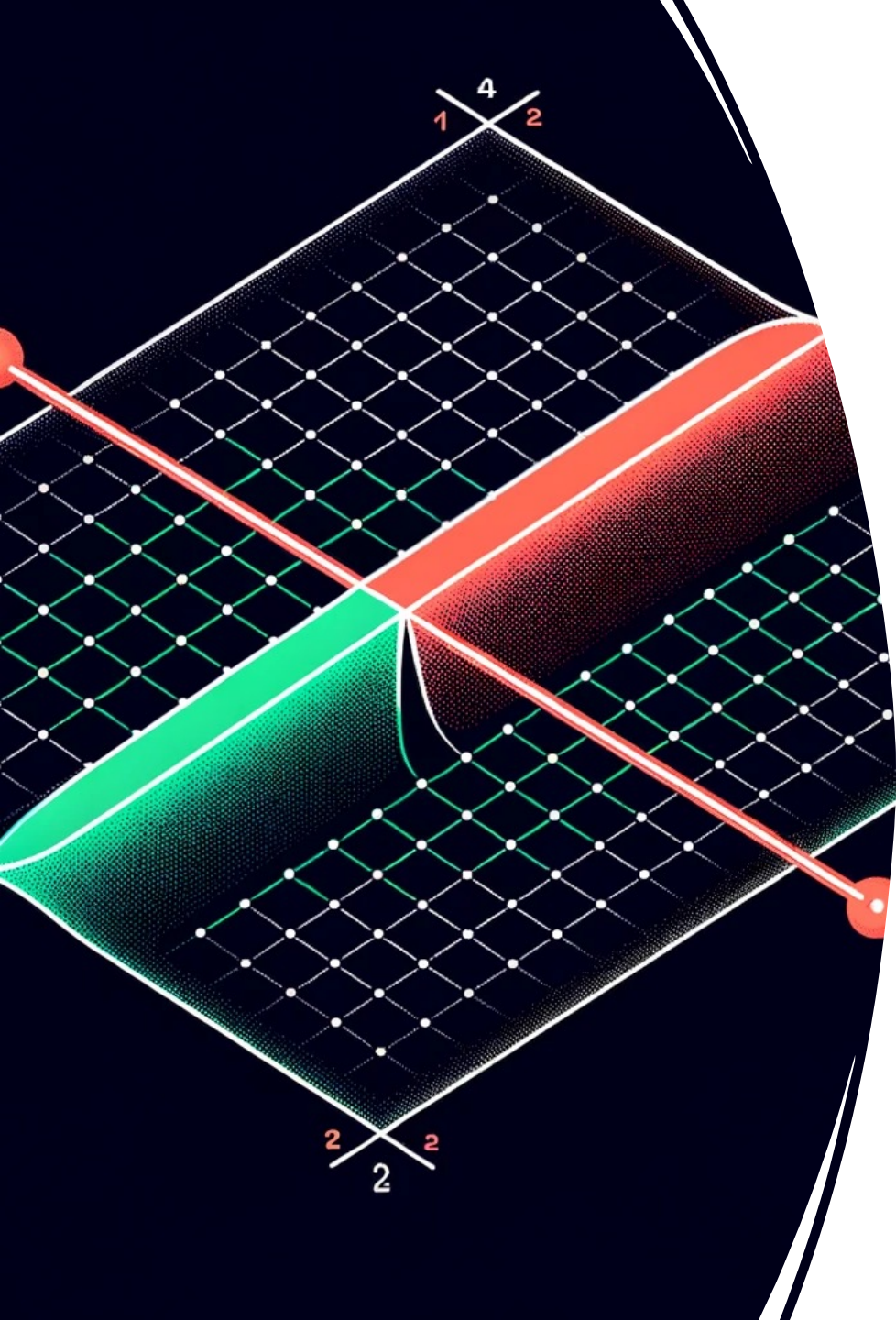
1676 — Leibniz {discovers/invents) Chain Rule

Gauss & Legendre independently invent linear regression (1805–1809)



Probabilistic Classification Emerges

- 1934 — probit Regression
 - Linear model using the probit activation function (CDF of a normal)
 - Parameters estimated by MLE by Fischer in 1935
- 1943 — logistic regression proposed as an alternative
 - Logistic function previously used in 1930s to model population growth
 - Later refined by statistician David Cox in 1958



WWII — Emergence of “Cybernetics”

- Led by Norbert Wiener, who publishes “Cybernetics” in 1948
- Interdisciplinary group of scientists
- Observes similarities between organisms and machines
- Develops technical language for describing both as “systems”
- Focused on feedback loops, statistical principles, information

1943 — McCulloch & Pitts Artificial Neuron

Bulletin of Mathematical Biology Vol. 52, No. 1/2, pp. 99–115, 1990.
Printed in Great Britain.

0092-8240/90\$3.00 + 0.00
Pergamon Press plc
Society for Mathematical Biology

A LOGICAL CALCULUS OF THE IDEAS IMMANENT IN NERVOUS ACTIVITY*

■ WARREN S. MCCULLOCH AND WALTER PITTS
University of Illinois, College of Medicine,
Department of Psychiatry at the Illinois Neuropsychiatric Institute,
University of Chicago, Chicago, U.S.A.

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.



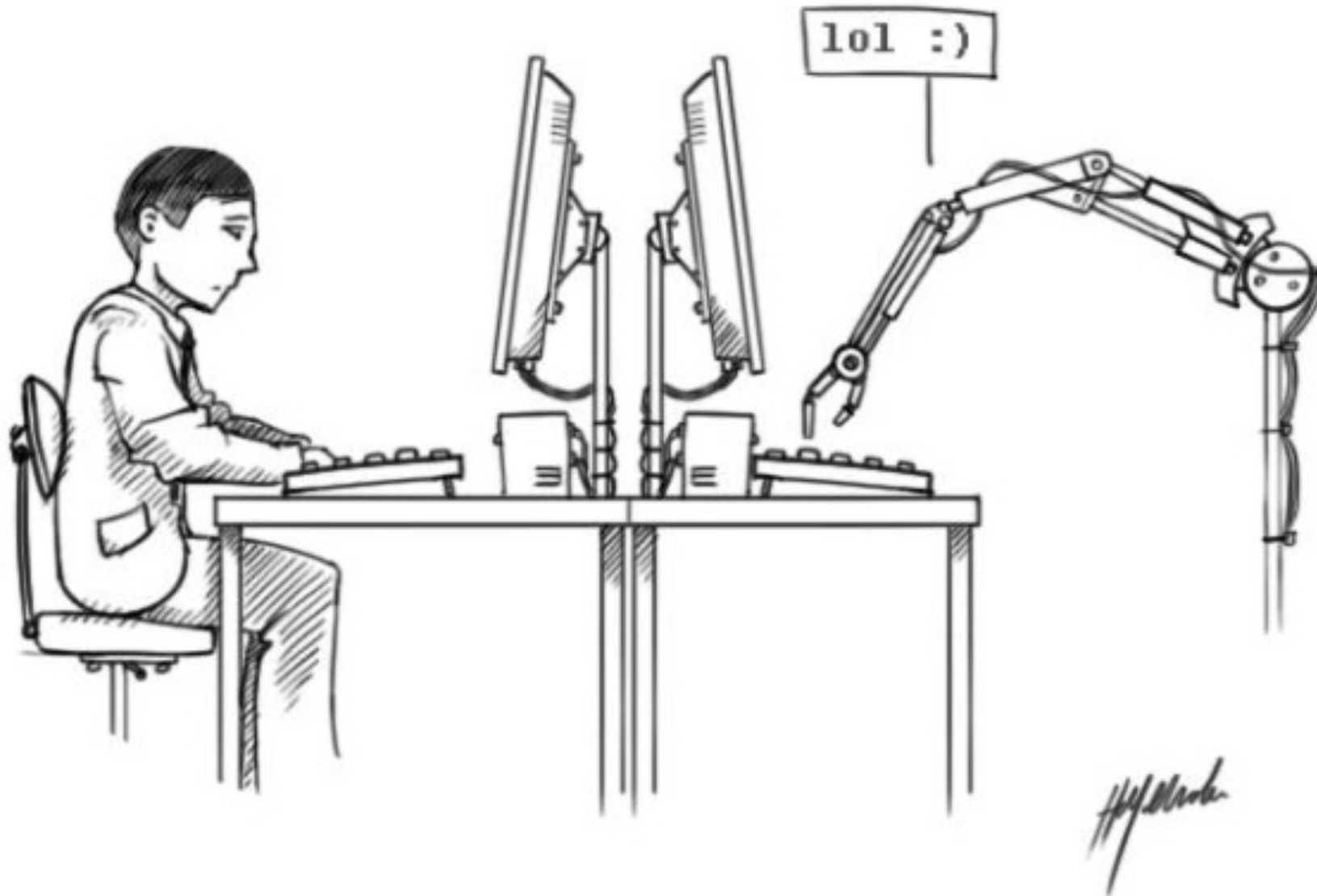
1945—Eniac, first general-purpose computer

The image features two stylized human figures, one on the left and one on the right, facing each other. Their heads are replaced by glowing, textured brains. From the necks and shoulders of each figure, a complex network of branching, tree-like structures extends outwards, resembling a neural network or dendritic tree. The background is dark with a subtle starry pattern. The text is overlaid in white, centered between the figures.

Donald Hebb's learning hypothesis
(1949)

“cells that wire together,
fire together”

1950—Turing asks, “Can machines think?”





1955—McCarthy & Gang Propose to Solve AI
in a Summer



1958—Rosenblatt's
Perceptron

First Neural Network Hype Cycle Kicks Off

(New York Times — July, 1958)

Electronic 'Brain' Teaches Itself

The Navy last week demonstrated the embryo of an electronic computer named the Perceptron which, when completed in about a year, is expected to be the first non-living mechanism able to "perceive, recognize and identify its surroundings without human training or control." Navy officers demonstrating a preliminary form of the device in Washington said they hesitated to call it a machine because it is so much like a "human being without life."

Dr. Frank Rosenblatt, research psychologist at the Cornell Aeronautical Laboratory, Inc., Buffalo, N. Y., designer of the Perceptron, conducted the demonstration. The machine, he said, would be the first electronic device to think as the human brain. Like humans, Perceptron will make mistakes at first, "but it will grow wiser as it gains experience," he said.

recognize the difference between right and left, almost the way a child learns.

When fully developed, the Perceptron will be designed to remember images and information it has perceived itself, whereas ordinary computers remember only what is fed into them on punch cards or magnetic tape.

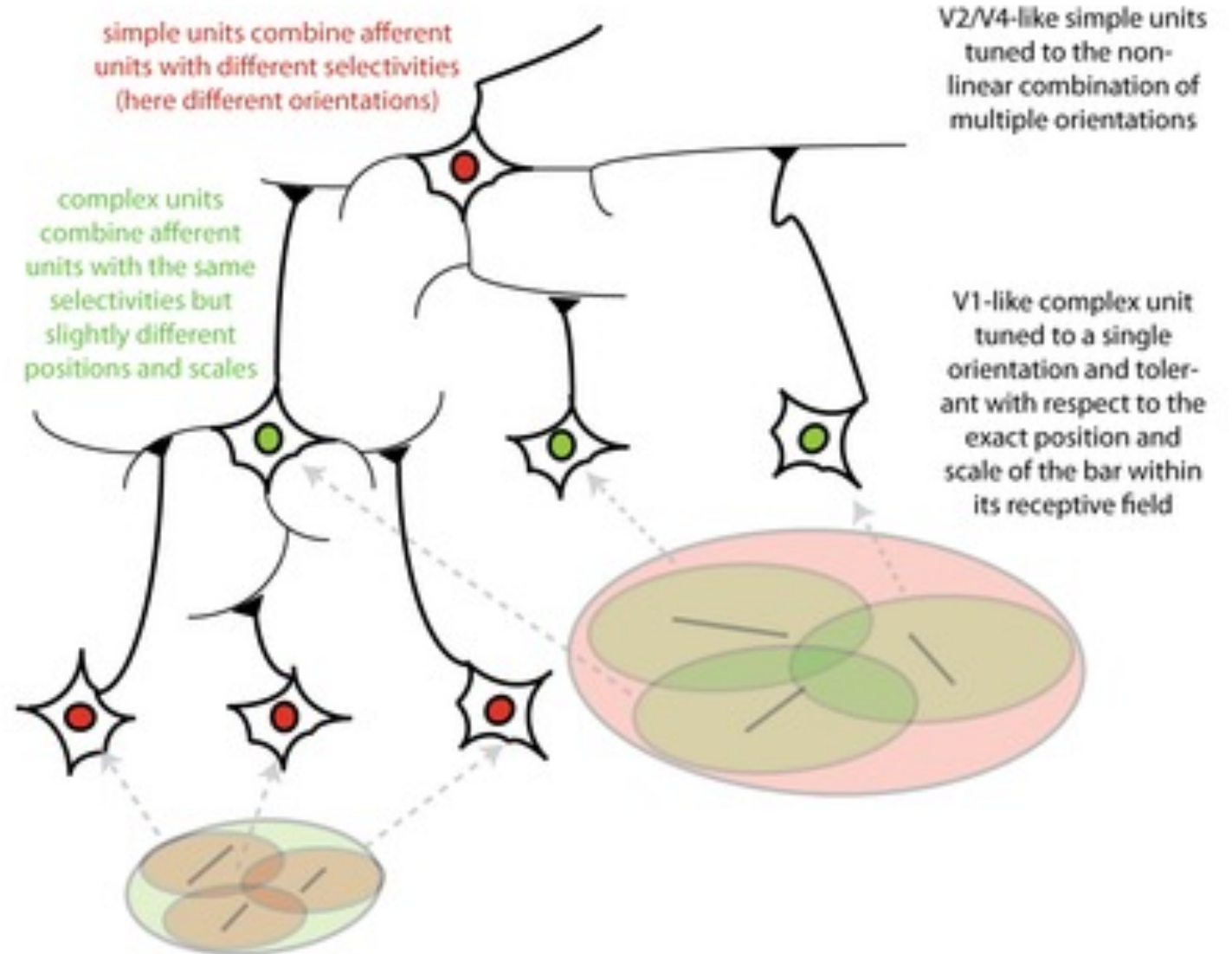
Later Perceptrons, Dr. Rosenblatt said, will be able to recognize people and call out their names. Printed pages, longhand letters and even speech commands are within its reach. Only one more step of development, a difficult step, he said, is needed for the device to hear speech in one language and instantly translate it to speech or writing in another language.

Self-Reproduction

In principle, Dr. Rosenblatt said, it would be possible to build Per-

1959—Hubel & Wiesel Propose Model of Simple & Complex Neurons

- Simple cells have smaller receptive fields, act like edge detectors
- Complex cells have larger receptive fields, less sensitive to deviations in exact position of objects



1965 — Deep Learning Begins Quietly

- First non-linear networks trained by stochastic gradient descent (Amari et al)
- Layer-by-layer training of deep models by Grigorivich & Ivakhenko

A Theory of Adaptive Pattern Classifiers

SHUNICHI AMARI

Abstract—This paper describes error-correction adjustment procedures for determining the weight vector of linear pattern classifiers under general pattern distribution. It is mainly aimed at clarifying theoretically the performance of adaptive pattern classifiers. In the case where the loss depends on the distance between a pattern vector and a decision boundary and where the average risk function is unimodal, it is proved that, by the procedures proposed here, the weight vector converges to the optimal one even under nonseparable pattern distributions. The speed and the accuracy of convergence are analyzed, and it is shown that there is an important tradeoff between speed and accuracy of convergence. Dynamical behaviors, when the probability distributions of patterns are changing, are also shown. The theory is generalized and made applicable to the case with general discriminant functions, including piecewise-linear discriminant functions.

Index Terms—Accuracy of learning, adaptive pattern classifier, convergence of learning, learning under nonseparable pattern distribution, linear decision function, piecewise-linear decision function, rapidity of learning.

needs a parametric treatment, that is, the distributions must be limited to those of a certain known kind whose distributions can be specified by a finite number of parameters. Moreover, the discriminant functions thus obtained depend directly on all of the past patterns so that they are not able to quickly follow the sudden change of the distributions. In order to avoid these shortcomings, we shall propose nonparametric learning procedures, by which the present discriminant function is modified according only to the present misclassified pattern.

The steepest-descent method is often used in order to minimize a known function. However, in our learning situation, we cannot obtain the descending directions of the average risk which we intend to minimize, because the probability distributions of the patterns are unknown. What we can utilize is the present pattern only,

CYBERNETICS AND FORECASTING

Cybernetics and Forecasting Techniques

By A. G. Ivakhnenko and V. G. Lapa. Translated by Scripta Technica, Inc. Translation edited by Robert N. McDonough. (Modern Analytic and Computational Methods in Science and Mathematics.) Pp. xxvii + 168. (New York: American Elsevier Publishing Co.; Amsterdam and London: Elsevier Publishing Co., 1967.) 130s.

THIS is an intriguing and exasperating book. The field of knowledge encompassed by the term cybernetics is embarrassingly wide and there is some danger that in England this book's title will appeal primarily to the statistician and the operations research worker whereas in fact its background is strictly that of the communications and control systems engineer.

Backpropagation Developed in Control Theory

- Ahead of its time. Didn't make quite the splash.

Gradient Theory of Optimal Flight Paths

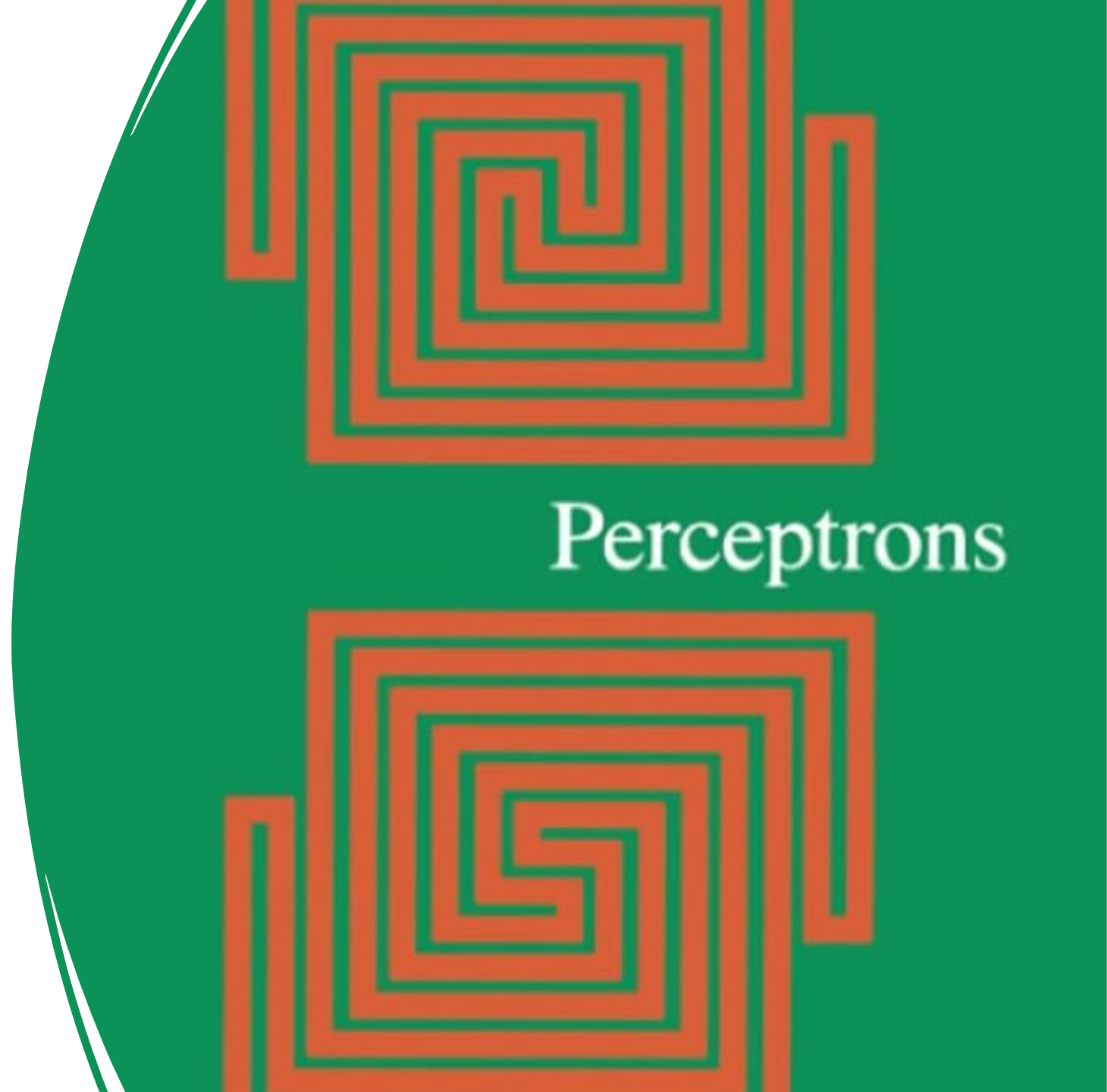
HENRY J. KELLEY¹

Grumman Aircraft Engineering Corp.
Bethpage, N. Y.

An analytical development of flight performance optimization according to the method of gradients or "method of steepest descent" is presented. Construction of a minimizing sequence of flight paths by a stepwise process of descent along the local gradient direction is described as a computational scheme. Numerical application of the technique is illustrated in a simple example of orbital transfer via solar sail propulsion. Successive approximations to minimum time planar flight paths from Earth's orbit to the orbit of Mars are presented for cases corresponding to free and fixed boundary conditions on terminal velocity components.

Minsky & Papert Publish *Perceptrons*

- Credited (rightly or wrongly) with quelling enthusiasm for line of research
- Demonstrated limitations of simpler (single-layer) perceptrons
- Subject of confusion in lore.



1970s—Expert Systems all the Rage

- DARPA funds massive projects around “Knowledge Engineering”
- Herb Simon and Alan Newell win Turing Award in 1975, focused on modeling psychological systems as collections of “if-then” statements
- Focused on applying logical deduction to curated collections of facts (knowledge-bases)

Fukushima introduces Cognitron architecture, Rectified Linear Unit (ReLU) activation (1975)

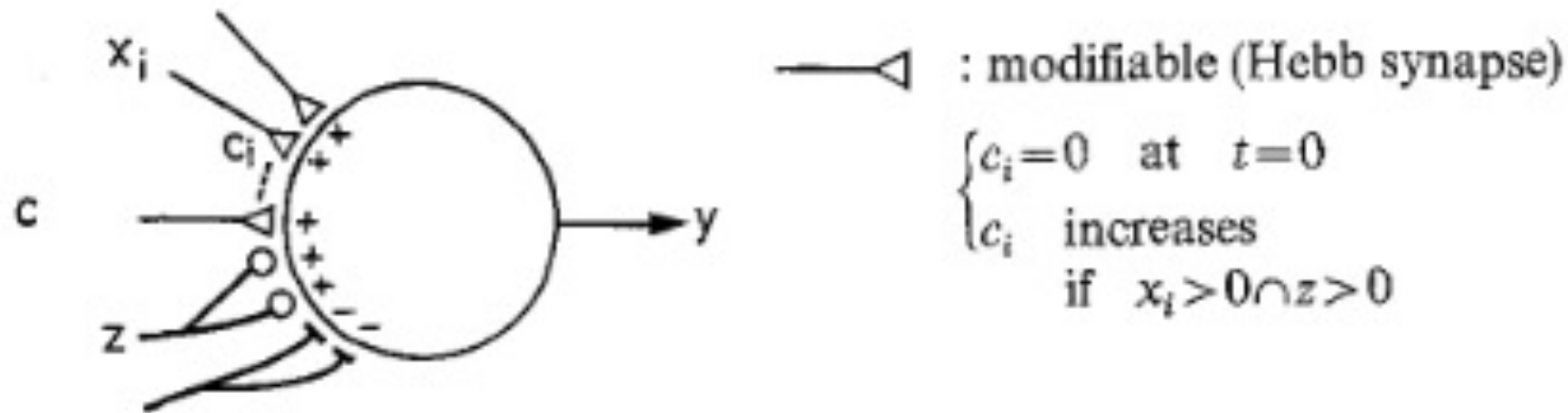


Fig. 1a-c. Hitherto-proposed three hypotheses on the modification of synapses

Fukushima Proposes Neocognitron: inventing Convolutional Architecture (1980)

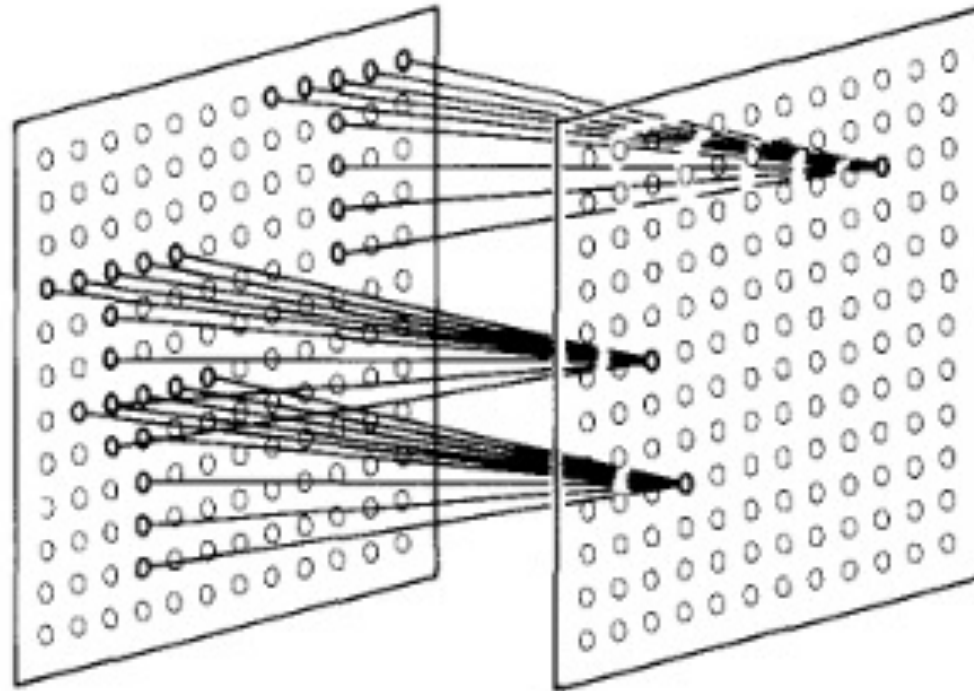


Fig. 3. Illustration showing the input interconnections to the cells within a single cell-plane

Werbos reinvents Backpropagation (1974), Applies to MLPs (1982)

The name *back propagation* actually comes from the term employed by Rosenblatt (1962) for his attempt to generalize the perceptron learning algorithm to the multilayer case. There were many attempts to generalize the perceptron learning procedure to multiple layers during the 1960s and 1970s, but none of them were especially successful. There appear to have been at least three independent inventions of the modern version of the back-propagation algorithm: Paul Werbos developed the basic idea in 1974 in a Ph.D. dissertation entitled

From Backpropagation: The Basic Theory — Rumelhart, Durbin, Golden, Chauvin

1986—Rumelhart & Hinton Popularize Backpropagation, train larger nets

Learning representations by back-propagating errors

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

* Institute for Cognitive Science, C-015, University of California, San Diego, La Jolla, California 92093, USA

† Department of Computer Science, Carnegie-Mellon University, Pittsburgh, Philadelphia 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 procedure¹.

There have been many attempts to design self-organizing

more difficult when we introduce hidden units whose actual or desired states are not specified by the task. (In perceptrons, there are 'feature analysers' between the input and output that are not true hidden units because their input connections are fixed by hand, so their states are completely determined by the input vector: they do not learn representations.) The learning procedure must decide under what circumstances the hidden units should be active in order to help achieve the desired input-output behaviour. This amounts to deciding what these units should represent. We demonstrate that a general purpose and relatively simple procedure is powerful enough to construct appropriate internal representations.

The simplest form of the learning procedure is for layered networks which have a layer of input units at the bottom; any number of intermediate layers; and a layer of output units at the top. Connections within a layer or from higher to lower layers are forbidden, but connections can skip intermediate layers. An input vector is presented to the network by setting the states of the input units. Then the states of the units in each layer are determined by applying equations (1) and (2) to the connections coming from lower layers. All units within a layer have their states set in parallel, but different layers have their states set sequentially, starting at the bottom and working upwards until the states of the output units are determined.

The total input, x_j , to unit j is a linear function of the outputs,

Jordan Nets with Recurrent Nets 1986

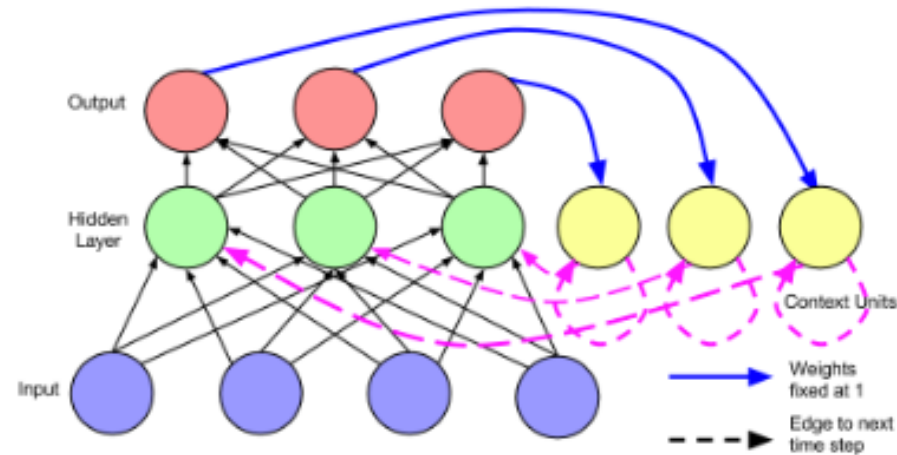


Figure 5: A recurrent neural network as proposed by [Jordan \[1986\]](#). Output units are connected to special units that at the next time step feed into themselves and into hidden units.

Finding Structure in Time: Seeds of Language Modeling and Modern RNNs

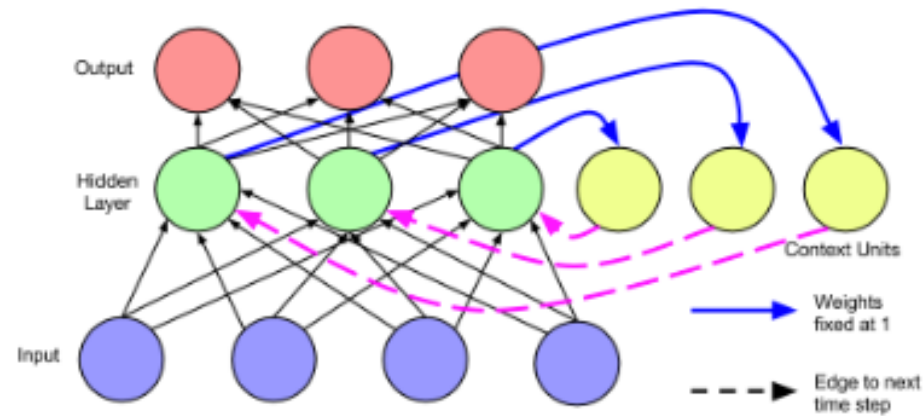
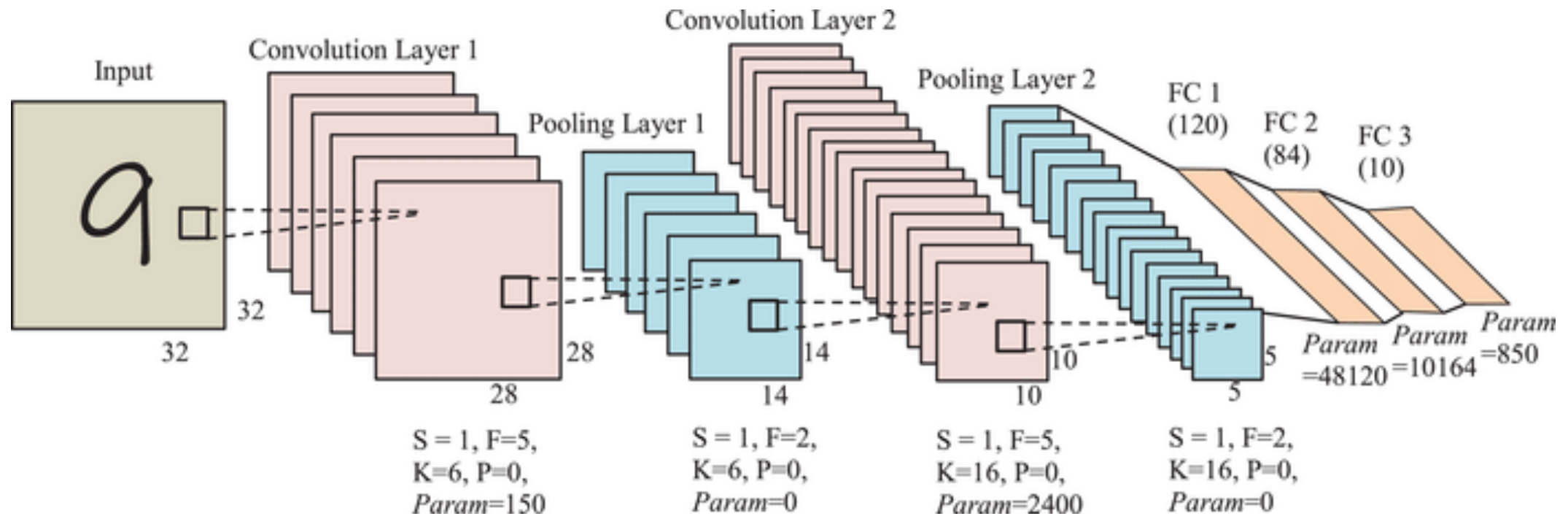


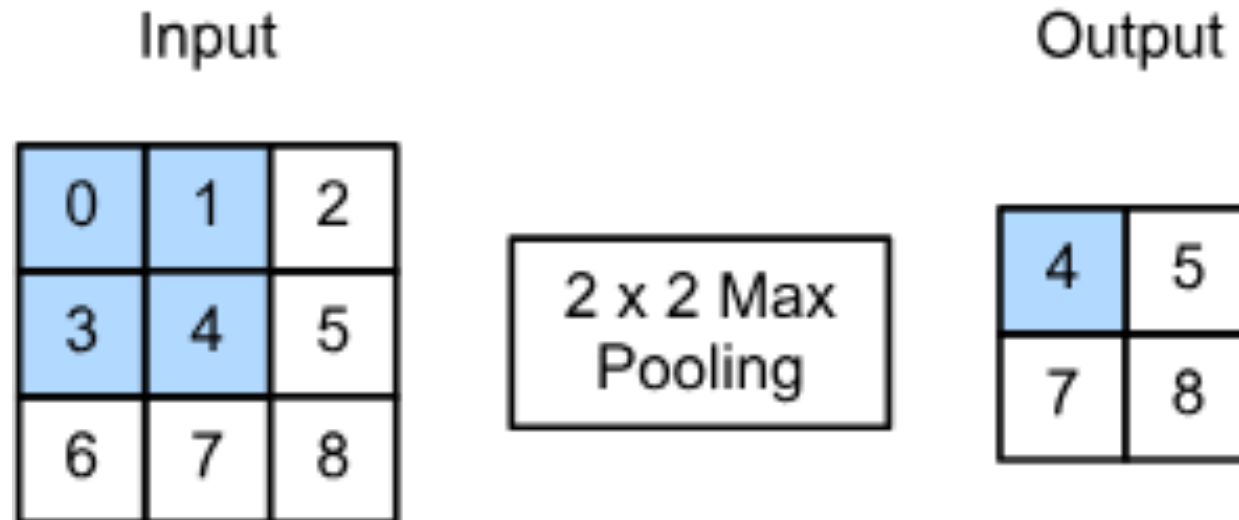
Figure 6: A recurrent neural network as described by [Elman \[1990\]](#). Hidden units are connected to context units, which feed back into the hidden units at the next time step.

Yann LeCun trains ConvNets for OCR (1989)



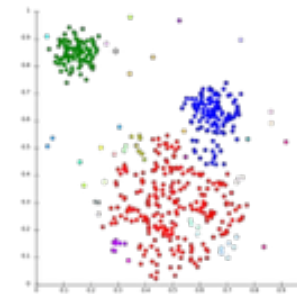
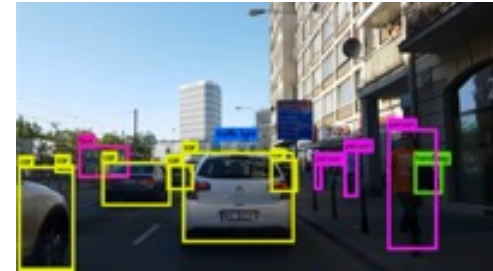
Yamaguchi introduce Max-Pooling (1990)

- Applied in neural network for speech recognition (“speaker-independent isolated word recognition”)



1990s—“Textbook ML” comes into focus

- **Supervised learning**
Predict y given x
- **Unsupervised learning**
Uncover the *structure* of x ,
without pre-specifying any
prediction task
- **Reinforcement learning**
Learn a policy to optimize a
delayed reward signal

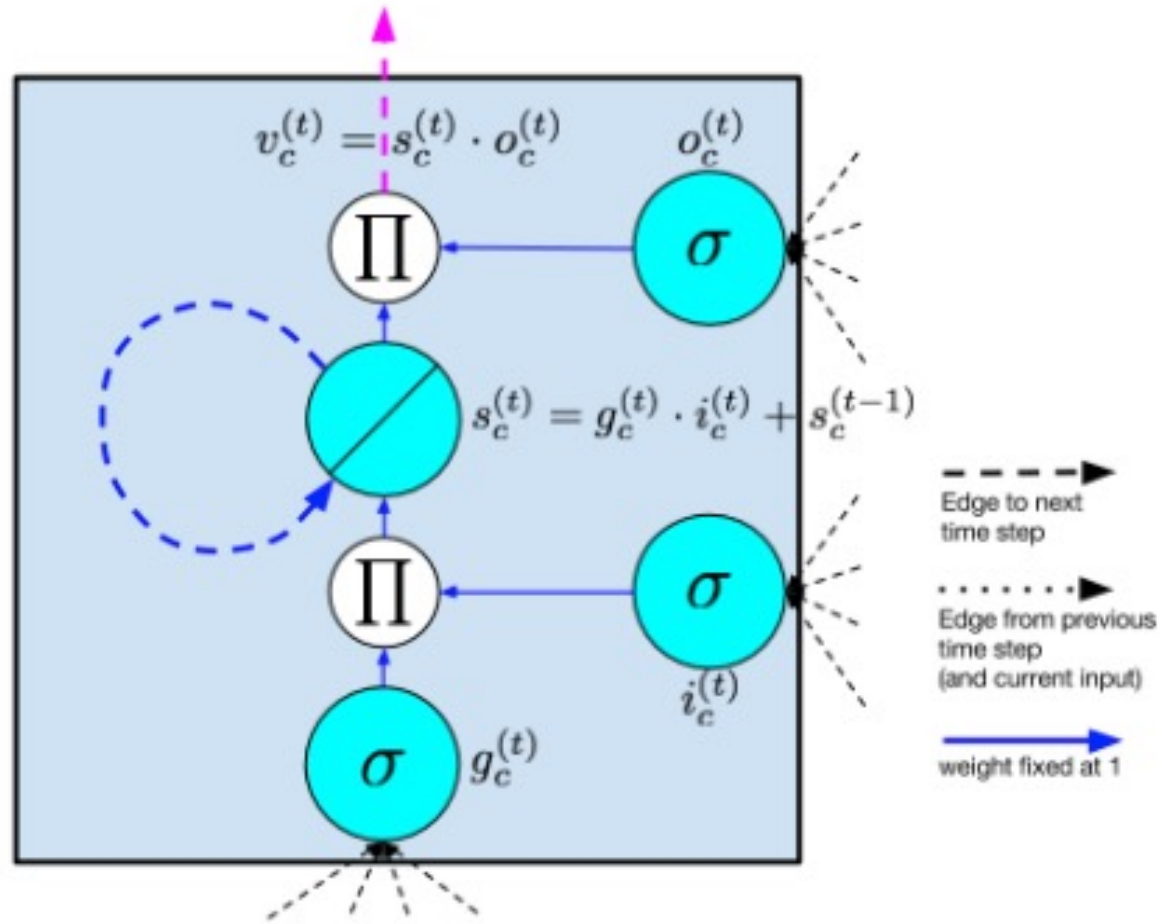


1991 LeNet Applied for OCR

1995 Adopted by Banks (for check-reading)

| | | |
|---|-----------|--|
| 1 | 5,058,179 | 2 |
| HIERARCHICAL CONSTRAINED AUTOMATIC LEARNING NETWORK FOR CHARACTER RECOGNITION | | SUMMARY OF THE INVENTION |
| CROSS-REFERENCE TO RELATED APPLICATION | | Highly accurate, reliable optical character recognition is afforded by a hierarchically layered network having several layers of parallel constrained feature detection for localized feature extraction followed by several fully connected layers for dimensionality reduction. Character classification is also performed in the ultimate fully connected layer. Each layer of parallel constrained feature detection comprises a plurality of constrained feature maps and a corresponding plurality of kernels wherein a predetermined kernel is directly related to a single constrained feature map. Undersampling occurs from layer to layer. |
| This application is related to U.S. patent application Ser. No. 444,455 filed Nov. 30, 1989 and commonly assigned herewith. | | 10 |
| TECHNICAL FIELD | | 15 |
| This invention relates to the field of pattern recognition and, more particularly, to massively parallel, constrained networks for optical character recognition. | | In an embodiment according to the principles of the invention, the hierarchical network comprises two layers of constrained feature detection followed by two fully connected layers of dimensionality reduction. Each constrained feature map comprises a plurality of units. Units in each constrained feature map of the first constrained feature detection layer respond as a function of both the corresponding kernel for the constrained feature map and different portions of the pixel image of the character captured in a receptive field associated with the unit. Units in each feature map of the second constrained feature detection layer respond as a function of both the corresponding kernel for the constrained feature map and different portions of an individual constrained feature map or a combination of several constrained feature maps in the first constrained |
| BACKGROUND OF THE INVENTION | | 20 |
| Computation systems based upon adaptive learning with fine-grained parallel architectures have moved out of obscurity in recent years because of the growth of computer-based information gathering, handling, manipulation, storage, and transmission. Many concepts applied in these systems represent potentially efficient approaches to solving problems such as providing automatic recognition, analysis and classification of character patterns in a particular image. Ultimately, the value of these techniques in such systems depends on their effectiveness or accuracy relative to conventional approaches. | | 25 |
| | | 30 |

1997 — Invention of LSTM RNNs




[Hochreiter and Schmidhuber \[1997\]](#)

2010—The Rise of Modern Deep Learning

- 2008 Graves/Schmidhuber make strides in handwriting recognition/generation
- 2010 Dahl/Hinton Win Kaggle Competition for predicting drug binding sites

Scientists See Promise in Deep-Learning Programs

 Give this article



2012 Khrizhevsky/Sutskever/Hinton win ImageNet Challenge

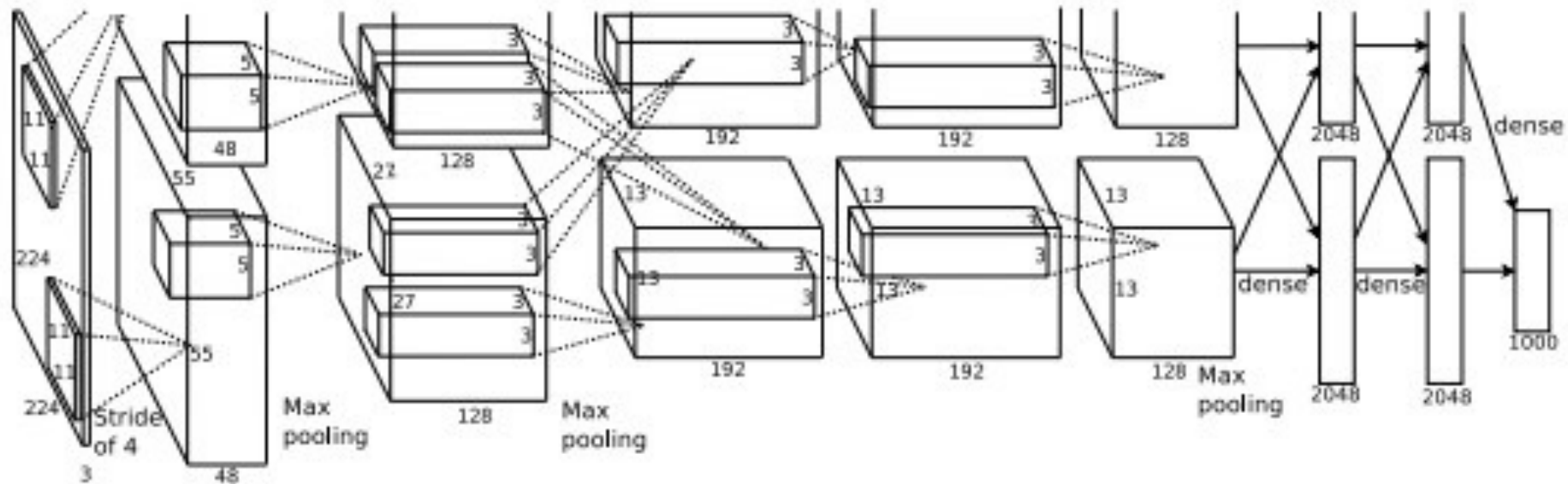


Figure 2: An illustration of the architecture of our CNN, explicitly showing the delineation of responsibilities between the two GPUs. One GPU runs the layer-parts at the top of the figure while the other runs the layer-parts at the bottom. The GPUs communicate only at certain layers. The network's input is 150,528-dimensional, and the number of neurons in the network's remaining layers is given by 253,440–186,624–64,896–64,896–43,264–4096–4096–1000.

“Human-level control through deep reinforcement learning” 2013

LETTER

doi:10.1038/nature14236

Human-level control through deep reinforcement learning

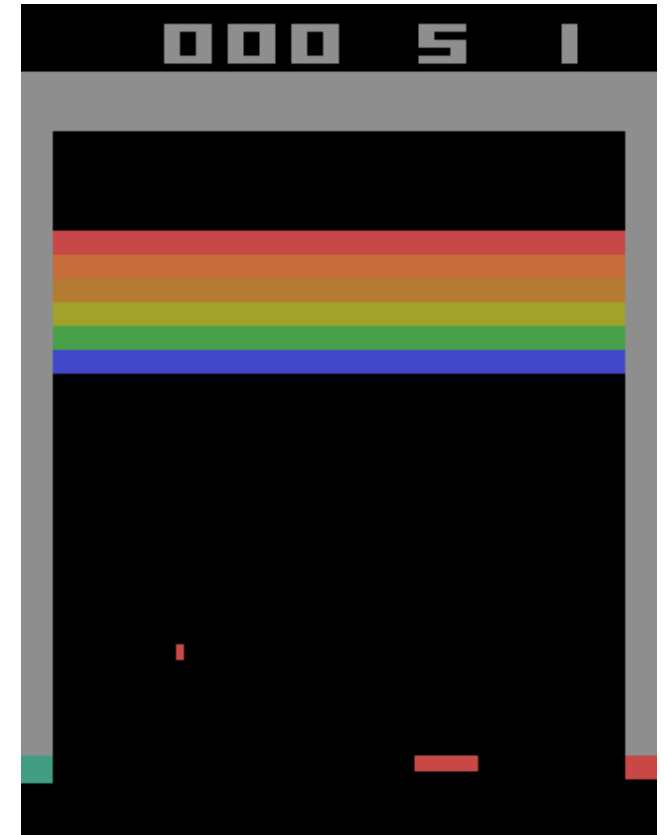
Volodymyr Mnih^{1*}, Koray Kavukcuoglu^{1*}, David Silver^{1*}, Andrei A. Rusu¹, Joel Veness¹, Marc G. Bellemare¹, Alex Graves¹, Martin Riedmiller¹, Andreas K. Fiedjeland¹, Georg Ostrovski¹, Stig Petersen¹, Charles Beattie¹, Amir Sadik¹, Ioannis Antonoglou¹, Helen King¹, Dharshan Kumaran¹, Daan Wierstra¹, Shane Legg¹ & Demis Hassabis¹

The theory of reinforcement learning provides a normative account¹, deeply rooted in psychological² and neuroscientific³ perspectives on animal behaviour, of how agents may optimize their control of an environment. To use reinforcement learning successfully in situations approaching real-world complexity, however, agents are confronted with a difficult task: they must derive efficient representations of the environment from high-dimensional sensory inputs, and use these to generalize past experience to new situations. Remarkably, humans

agent is to select actions in a fashion that maximizes cumulative future reward. More formally, we use a deep convolutional neural network to approximate the optimal action-value function

$$Q^*(s,a) = \max_{\pi} \mathbb{E}[r_t + \gamma r_{t+1} + \gamma^2 r_{t+2} + \dots | s_t = s, a_t = a, \pi],$$

which is the maximum sum of rewards r_t discounted by γ at each time-step t , achievable by a behaviour policy $\pi = P(a|s)$, after making an observation (s) and taking an action (a) (see Methods)¹⁹.



DeepMind's AlphaGo Masters Go

ARTICLE

doi:10.1038/nature16961

Mastering the game of Go with deep neural networks and tree search

David Silver^{1*}, Aja Huang^{1*}, Chris J. Maddison¹, Arthur Guez¹, Laurent Sifre¹, George van den Driessche¹, Julian Schrittwieser¹, Ioannis Antonoglou¹, Veda Panneershelvam¹, Marc Lanctot¹, Sander Dieleman¹, Dominik Grewe¹, John Nham², Nal Kalchbrenner¹, Ilya Sutskever², Timothy Lillicrap¹, Madeleine Leach¹, Koray Kavukcuoglu¹, Thore Graepel¹ & Demis Hassabis¹

The game of Go has long been viewed as the most challenging of classic games for artificial intelligence owing to its enormous search space and the difficulty of evaluating board positions and moves. Here we introduce a new approach to computer Go that uses 'value networks' to evaluate board positions and 'policy networks' to select moves. These deep neural networks are trained by a novel combination of supervised learning from human expert games, and reinforcement learning from games of self-play. Without any lookahead search, the neural networks play Go at the level of state-of-the-art Monte Carlo tree search programs that simulate thousands of random games of self-play. We also introduce a new search algorithm that combines Monte Carlo simulation with value and policy networks. Using this search algorithm, our program AlphaGo achieved a 99.8% winning rate against other Go programs, and defeated the human European Go champion by 5 games to 0. This is the first time that a computer program has defeated a human professional player in the full-sized game of Go, a feat previously thought to be at least a decade away.



Industrial Applications in Healthcare

A call for deep-learning healthcare

Beau Norgeot, Benjamin S. Glicksberg & Atul J. Butte

Nature Medicine 25, 14–15 (2019) | Cite this article

6188 Accesses | 9 Citations | 188 Altmetric | Metrics

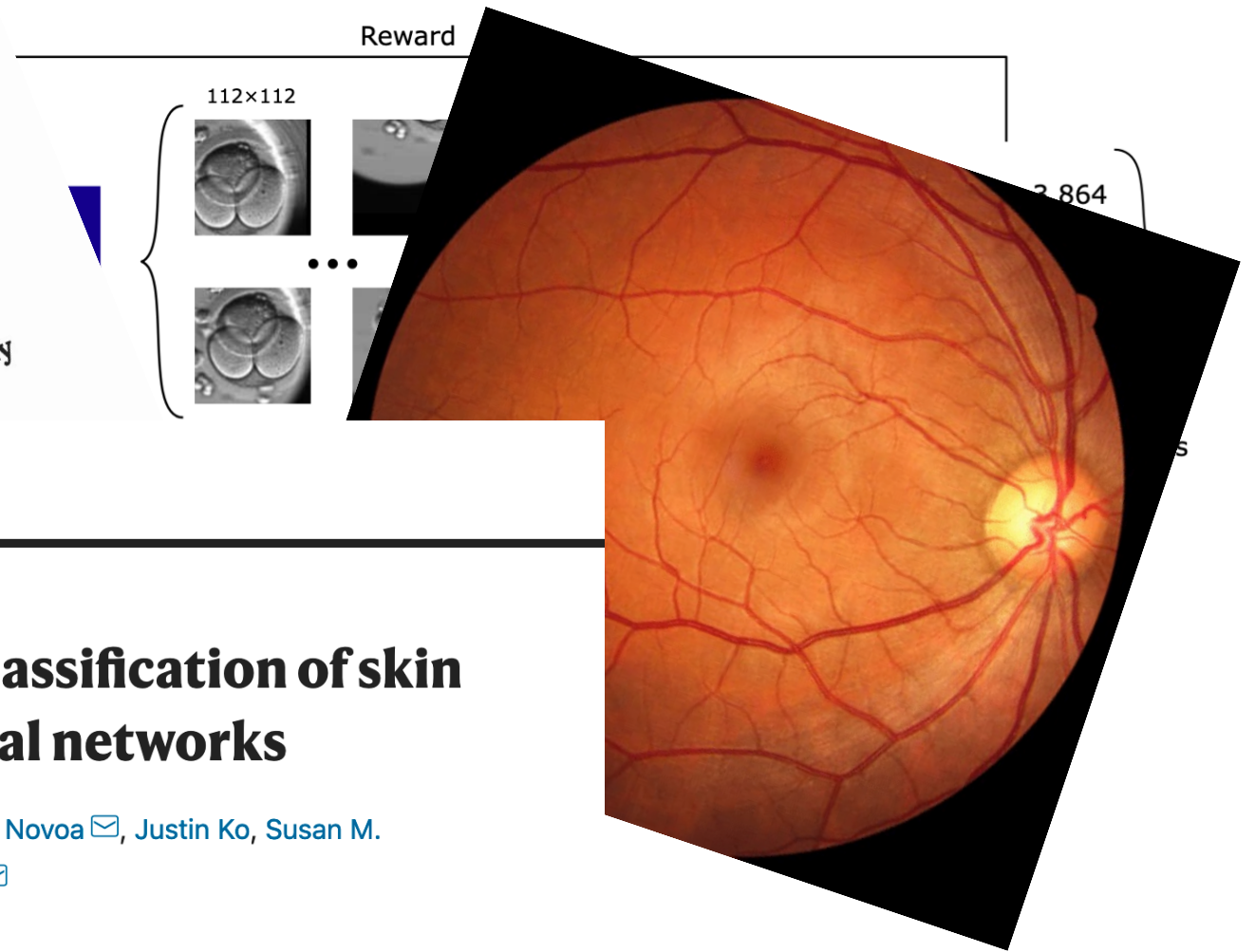
Here we argue that novel systems in which the learned from electrophysiological methodologies.

nature

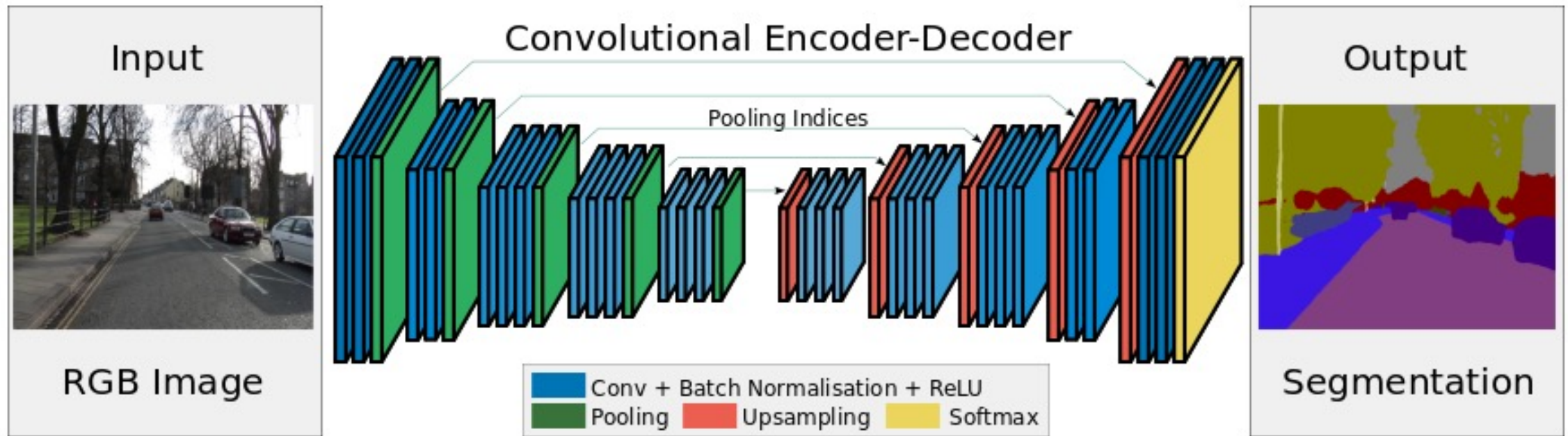
Letter | Published: 25 January 2017

Dermatologist-level classification of skin cancer with deep neural networks

Andre Esteva, Brett Kuprel, Roberto A. Novoa, Justin Ko, Susan M. Swetter, Helen M. Blau & Sebastian Thrun

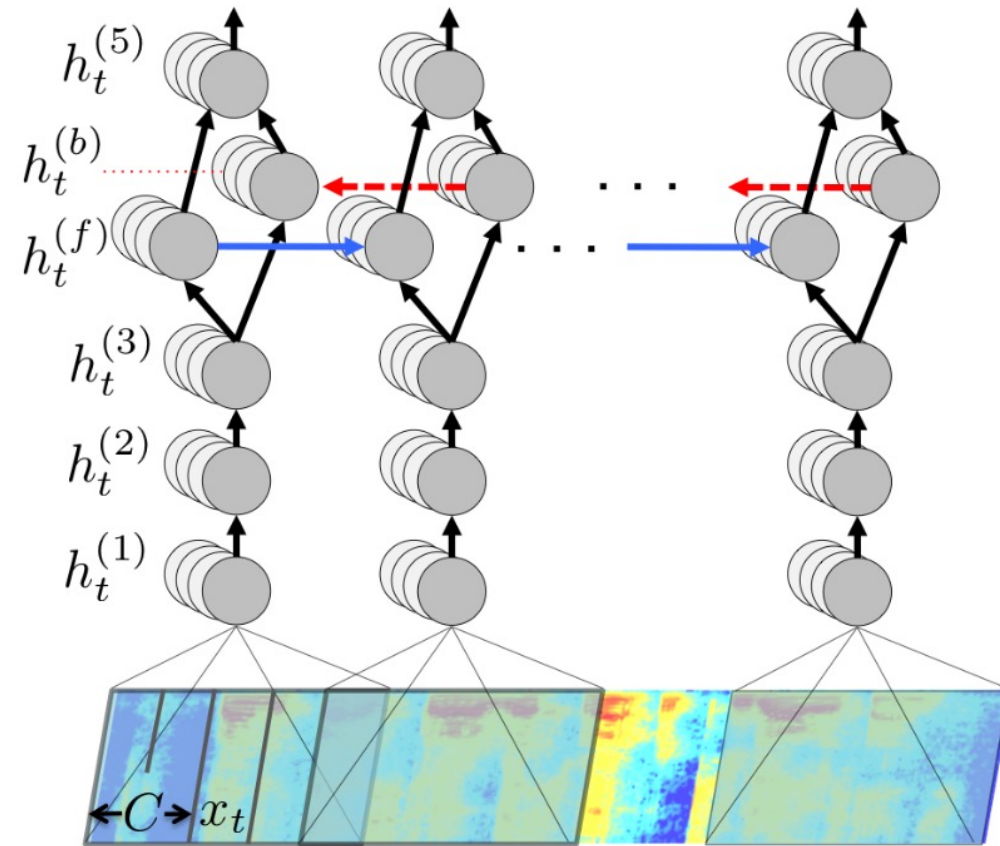


Optimism rises for new era of self-driving



<https://www.youtube.com/watch?v=9e2x4dDRB-k>

2014—Leaps in Commercial Speech Recognition (DeepSpeech)



Hannun et al, 2014

Concerns arise about Fairness/Transparency/Privacy

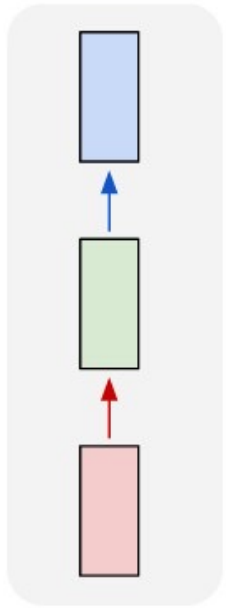


The background features a network of green circular nodes connected by yellow lines, set against a dark purple gradient. The nodes and lines are semi-transparent, creating a layered, digital effect. The text 'AI's Generative Turn' is centered in white, with a larger green node behind the letter 'i' in 'Generative'.

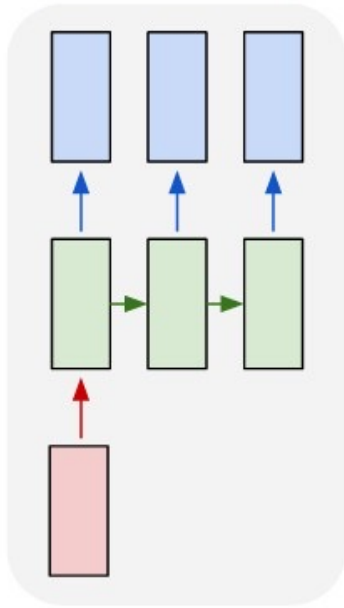
AI's Generative Turn

Sequence(-to-Sequence) Modeling

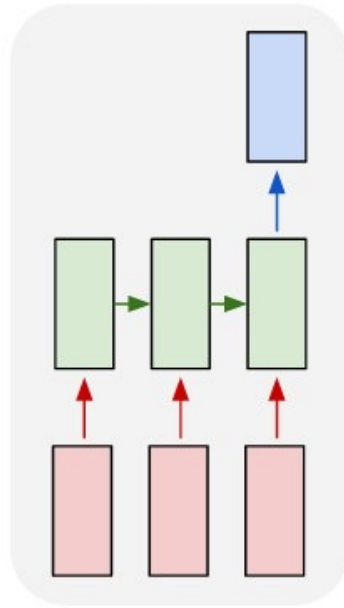
one to one



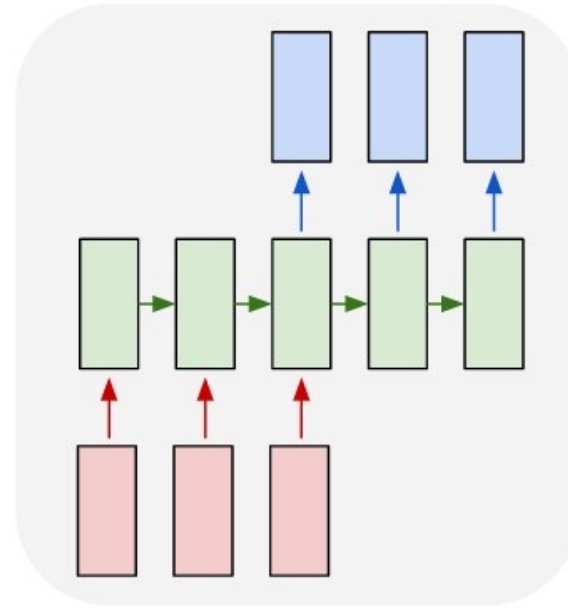
one to many



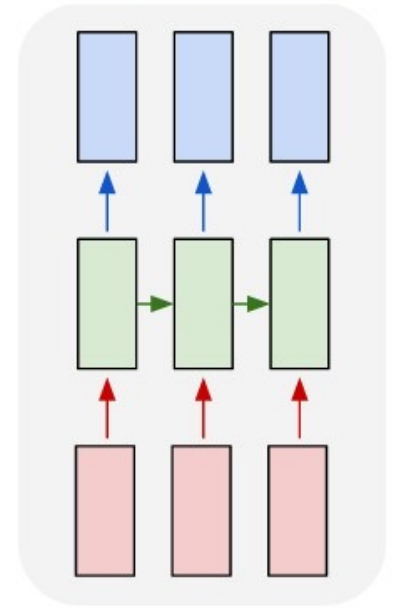
many to one



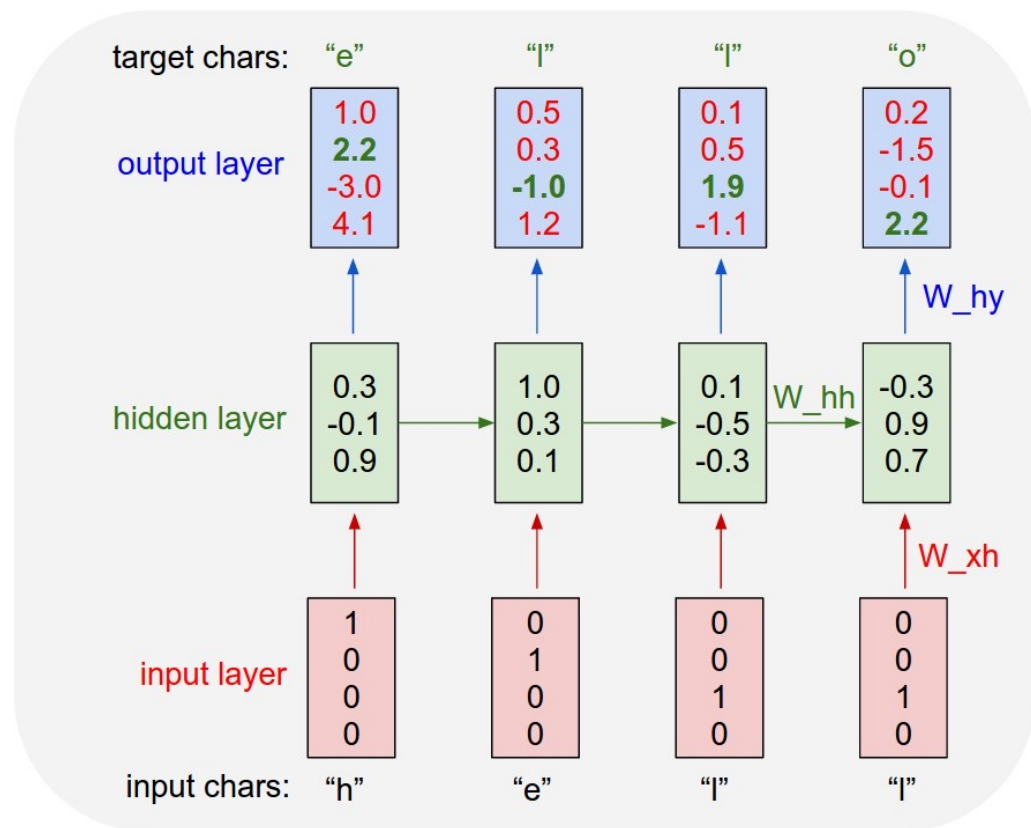
many to many



many to many



2012—Early Experiments with Deep RNNs + Language Modeling



while he was giving attention to the second advantage of school building a 2-for-2 stool killed by the Cultures saddled with a half-suit defending the Bharatiya Fernall 's office . Ms . Claire Parters will also have a history temple for him to raise jobs until naked Prodienna to paint baseball partners , provided people to ride both of Manhattan in 1978 , but what was largely directed to China in 1946 , focusing on the trademark period is the sailboat yesterday and comments on whom they obtain overheard within the 120th anniversary , where many civil rights defined , officials said early that forms , " said Bernard J. Marco Jr. of Pennsylvania , was monitoring New York

(Fast-forward to 2023)

ZA

write a rap about introduction to machine learning, CMU course 10701 in the style of Biggie Smalls (but with no curse words).



(Verse 1)

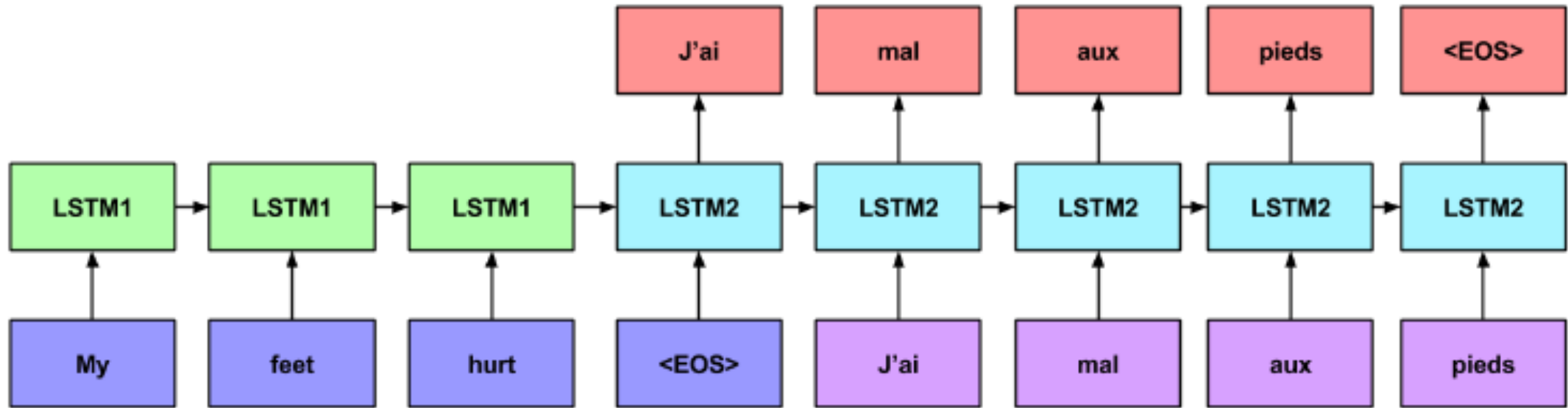
Yo, it's the intro to the ML scene,
CMU course 10701, where the knowledge is clean,
Binary classifications and decision trees,
Algorithms flowin' like a cool breeze.

From supervised learning to the unsupervised kind,
Neural networks in your mind, it's a gold mine.
Regression, clustering, and all in between,
This course got it all, if you know what I mean.

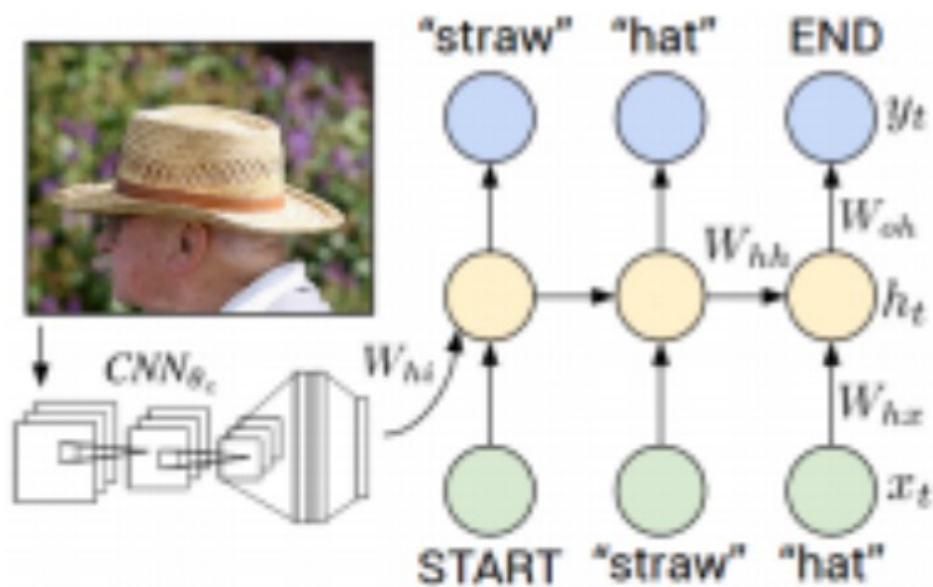
(Chorus)

Machine Learning, ML, the future's so bright,
CMU 10701, we're taking flight.
Biggie style, no curses, just pure insight,
Step into the class, and hold on tight.

Unaligned Seq-to-Seq Models for Natural Language Translation (2014)



2014/5 Image Captioning

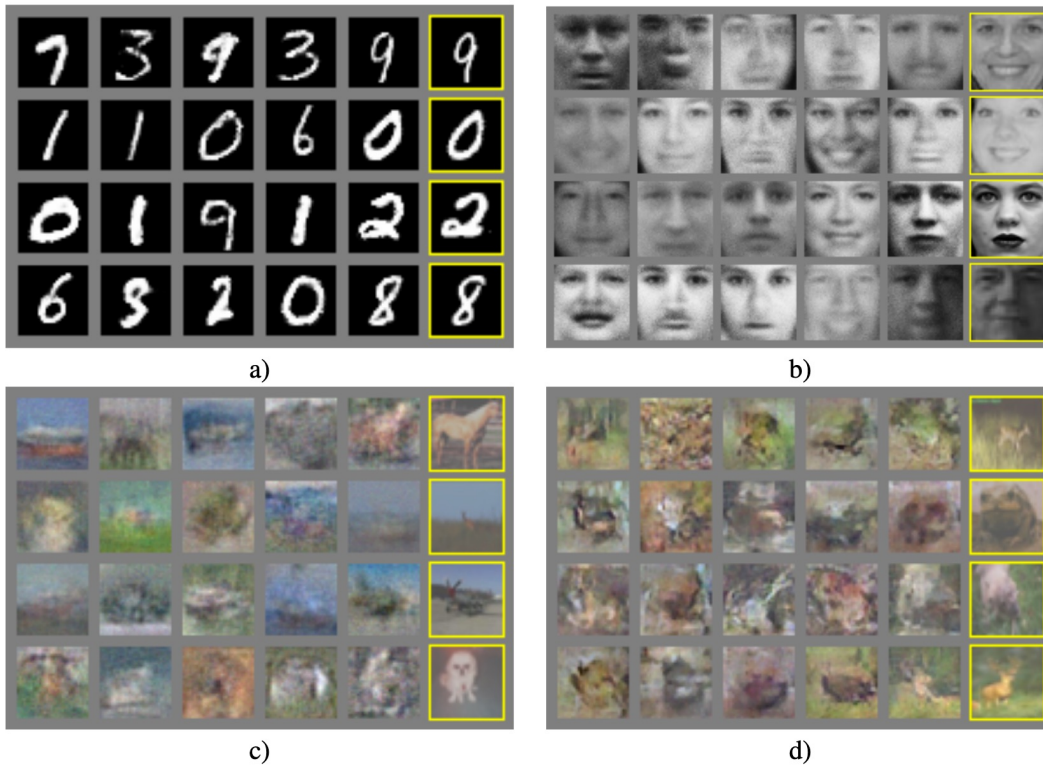


Human: "A green monster kite soaring in a sunny sky."

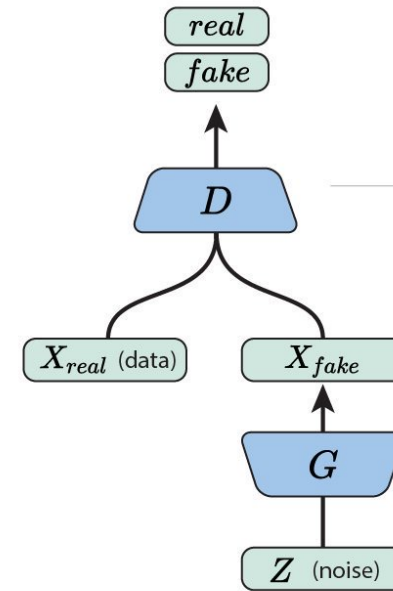
Computer model: "A man flying through the air while riding a snowboard."



2014 Generative Adversarial Networks



Generative Adversarial Networks (GANs) are a way to make a generative model by having two neural networks compete with each other.



The **discriminator** tries to distinguish genuine data from forgeries created by the generator.

The **generator** turns random noise into imitations of the data, in an attempt to fool the discriminator.

Earlier GAN results



Rapid progress in image quality



<https://www.youtube.com/watch?v=XOxxPcy5Gr4>

Conditional Diffusion Models

Prompt: *Anthropomorphic majestic blobfish knight, portrait, finely detailed armor, cinematic lighting, intricate filigree metal design, 4k, 8k, unreal engine, octane render*

Image via
<https://www.blueshadow.art/midjourney-prompt-commands/>





From Narrow Purpose-Built Models
to Webscale Capabilities

The Rise of Foundation Models

- 2017—ELMO pretrains forwards and backwards LSTMS for contextualized representations, fine-tunes on downstream tasks
- 2018—BERT trained on web crawl to learn representations useful for downstream classification with surprisingly little fine-tuning
- 2018—OpenAI releases GPT, a general web-scale language model
- 2019—OpenAI releases GPT2
- 2020—OpenAI releases GPT3
- 2021—OpenAI releases Dall-E, setting off rapid progress on text-to-image synthesis
- 2021—OpenAI releases CLIP, multimodal text + image embeddings
- 2022—OpenAI releases ChatGPT
- 2023—OpenAI releases GPT4

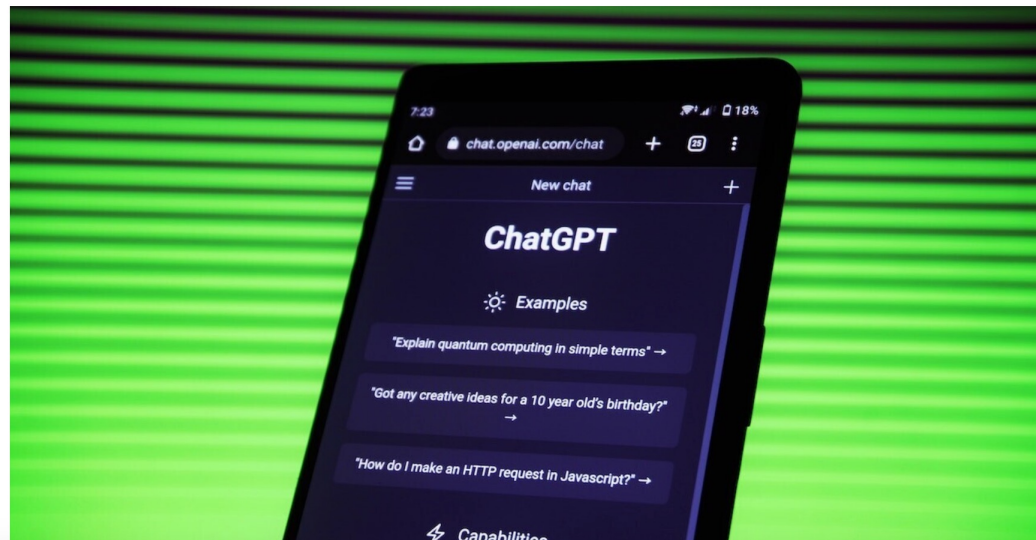
New Paradigms Emerge

FUTURE OF WORK

Prompt engineering could be the hottest job in tech, with a paycheck to match

Time to fire up your generative AI of choice

April 18, 2023 - 1:55 pm



BY JOBBIO



Zalando - Ansbach
Technical Support
Manager - 1st Level
(m/w/d)



HERO Software - Home
Office
Mitarbeiter
Telefonverkauf (w/m/d)

The emerging repertoire

- **Zero-shot prompting**

“The following is a movie review:

The sentiment of of the review was (positive/negative): ”

- **Few-shot prompting / In-Context learning**

X1: ..., Y1: ..., X2: ..., Y2: ..., X3: ..., Y3: _____

- **Chain-of-Thought reasoning**

Getting to final answer by means of a sequence of intermediate reasoning steps.

- **Task-specific fine-tuning**

What's new, What's the same?

- The role of data—it's possible in an unprecedented way, to get models to perform complex behaviors, without **any** additional training. This out-of-the-box capability is fascinating. Still, performance matters, and data/expertise are needed to guide the process.
- In many tasks, when clean data is available, fine-tuning models for narrow tasks still dominates.
- Many old skills remain relevant. Lots of capabilities require training models. There's also a new repertoire emerging where intuitions for prompts can be as important as intuitions for architectures (or feature engineering before that).