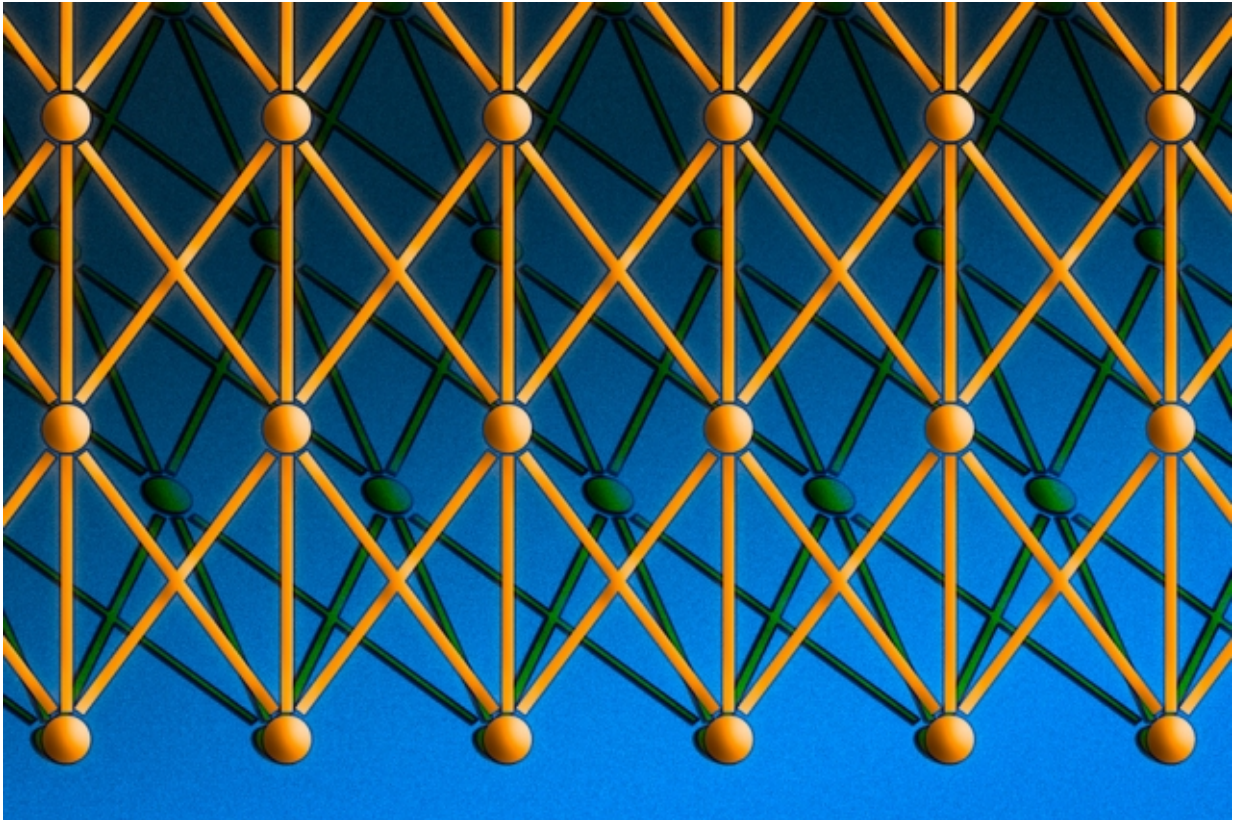# Neural networks explained

April 17 2017, by Larry Hardesty

Most applications of deep learning use "convolutional" neural networks, in which the nodes of each layer are clustered, the clusters overlap, and each cluster feeds data to multiple nodes (orange and green) of the next layer. Credit: Jose-Luis Olivares/MIT

In the past 10 years, the best-performing artificial-intelligence systems—such as the speech recognizers on smartphones or Google's

latest automatic translator—have resulted from a technique called "deep learning."

Deep learning is in fact a new name for an approach to artificial intelligence called neural networks, which have been going in and out of fashion for more than 70 years. Neural networks were first proposed in 1944 by Warren McCullough and Walter Pitts, two University of Chicago researchers who moved to MIT in 1952 as founding members of what's [sometimes called](#) the first cognitive science department.

Neural nets were a major area of research in both neuroscience and computer science until 1969, when, according to computer science lore, they were killed off by the MIT mathematicians Marvin Minsky and Seymour Papert, who a year later would become co-directors of the new MIT Artificial Intelligence Laboratory.

The technique then enjoyed a resurgence in the 1980s, fell into eclipse again in the first decade of the new century, and has returned like gangbusters in the second, fueled largely by the increased processing power of graphics chips.

"There's this idea that ideas in science are a bit like epidemics of viruses," says Tomaso Poggio, the Eugene McDermott Professor of Brain and Cognitive Sciences at MIT, an investigator at MIT's McGovern Institute for Brain Research, and director of MIT's Center for Brains, Minds, and Machines. "There are apparently five or six basic strains of flu viruses, and apparently each one comes back with a period of around 25 years. People get infected, and they develop an immune response, and so they don't get infected for the next 25 years. And then there is a new generation that is ready to be infected by the same strain of virus. In science, people fall in love with an idea, get excited about it, hammer it to death, and then get immunized—they get tired of it. So ideas should have the same kind of periodicity!"

Weighty matters

Neural nets are a means of doing machine learning, in which a computer learns to perform some task by analyzing training examples. Usually, the examples have been hand-labeled in advance. An object recognition system, for instance, might be fed thousands of labeled images of cars, houses, coffee cups, and so on, and it would find visual patterns in the images that consistently correlate with particular labels.

Modeled loosely on the human brain, a neural net consists of thousands or even millions of simple processing nodes that are densely interconnected. Most of today's neural nets are organized into layers of nodes, and they're "feed-forward," meaning that data moves through them in only one direction. An individual node might be connected to several nodes in the layer beneath it, from which it receives data, and several nodes in the layer above it, to which it sends data.

To each of its incoming connections, a node will assign a number known as a "weight." When the network is active, the node receives a different data item—a different number—over each of its connections and multiplies it by the associated weight. It then adds the resulting products together, yielding a single number. If that number is below a threshold value, the node passes no data to the next layer. If the number exceeds the threshold value, the node "fires," which in today's neural nets generally means sending the number—the sum of the weighted inputs—along all its outgoing connections.

When a neural net is being trained, all of its weights and thresholds are initially set to random values. Training data is fed to the bottom layer—the input layer—and it passes through the succeeding layers, getting multiplied and added together in complex ways, until it finally arrives, radically transformed, at the output layer. During training, the weights and thresholds are continually adjusted until training data with

the same labels consistently yield similar outputs.

Minds and machines

The neural nets described by McCullough and Pitts in 1944 had thresholds and weights, but they weren't arranged into layers, and the researchers didn't specify any training mechanism. What McCullough and Pitts showed was that a neural net could, in principle, compute any function that a digital computer could. The result was more neuroscience than computer science: The point was to suggest that the human brain could be thought of as a computing device.

Neural nets continue to be a valuable tool for neuroscientific research. For instance, particular network layouts or rules for adjusting weights and thresholds have reproduced observed features of human neuroanatomy and cognition, an indication that they capture something about how the brain processes information.

The first trainable neural network, the Perceptron, was demonstrated by the Cornell University psychologist Frank Rosenblatt in 1957. The Perceptron's design was much like that of the modern neural net, except that it had only one layer with adjustable weights and thresholds, sandwiched between input and output layers.

Perceptrons were an active area of research in both psychology and the fledgling discipline of computer science until 1959, when Minsky and Papert published a book titled "Perceptrons," which demonstrated that executing certain fairly common computations on Perceptrons would be impractically time consuming.

"Of course, all of these limitations kind of disappear if you take machinery that is a little more complicated—like, two layers," Poggio says. But at the time, the book had a chilling effect on neural-net

research.

"You have to put these things in historical context," Poggio says. "They were arguing for programming—for languages like Lisp. Not many years before, people were still using analog computers. It was not clear at all at the time that programming was the way to go. I think they went a little bit overboard, but as usual, it's not black and white. If you think of this as this competition between analog computing and digital computing, they fought for what at the time was the right thing."

Periodicity

By the 1980s, however, researchers had developed algorithms for modifying neural nets' weights and thresholds that were efficient enough for networks with more than one layer, removing many of the limitations identified by Minsky and Papert. The field enjoyed a renaissance.

But intellectually, there's something unsatisfying about neural nets. Enough training may revise a network's settings to the point that it can usefully classify data, but what do those settings mean? What image features is an object recognizer looking at, and how does it piece them together into the distinctive visual signatures of cars, houses, and coffee cups? Looking at the weights of individual connections won't answer that question.

In recent years, computer scientists have begun to come up with ingenious methods for deducing the analytic strategies adopted by neural nets. But in the 1980s, the networks' strategies were indecipherable. So around the turn of the century, neural networks were supplanted by support vector machines, an alternative approach to machine learning that's based on some very clean and elegant mathematics.

The recent resurgence in neural networks—the deep-learning

revolution—comes courtesy of the computer-game industry. The complex imagery and rapid pace of today's video games require hardware that can keep up, and the result has been the graphics processing unit (GPU), which packs thousands of relatively simple processing cores on a single chip. It didn't take long for researchers to realize that the architecture of a GPU is remarkably like that of a neural net.

Modern GPUs enabled the one-layer networks of the 1960s and the two- to three-layer networks of the 1980s to blossom into the 10-, 15-, even 50-layer networks of today. That's what the "deep" in "deep learning" refers to—the depth of the network's layers. And currently, deep learning is responsible for the best-performing systems in almost every area of artificial-intelligence research.

Under the hood

The networks' opacity is still unsettling to theorists, but there's headway on that front, too. In addition to directing the Center for Brains, Minds, and Machines (CBMM), Poggio leads the center's research program in Theoretical Frameworks for Intelligence. Recently, Poggio and his CBMM colleagues have released a three-part theoretical study of neural networks.

The first part, which was published last month in the International Journal of Automation and Computing, addresses the range of computations that deep-learning networks can execute and when deep networks offer advantages over shallower ones. Parts two and three, which have been released as CBMM technical reports, address the problems of global optimization, or guaranteeing that a network has found the settings that best accord with its training data, and overfitting, or cases in which the network becomes so attuned to the specifics of its training data that it fails to generalize to other instances of the same

categories.

There are still plenty of theoretical questions to be answered, but CBMM researchers' work could help ensure that neural networks finally break the generational cycle that has brought them in and out of favor for seven decades.

  More information: Tomaso Poggio et al. Why and when can deep-but not shallow-networks avoid the curse of dimensionality: A review, *International Journal of Automation and Computing* (2017). DOI: 10.1007/s11633-017-1054-2