CONCENTRIC

www.concentric.ai
contact_us@concentric.ai

DEEP LEARNING

FOR DATA SECURITY PROFESSIONALS

Dr. Madhu Shashanka
Chief Scientist and Founder

CONTENTS

INTRODUCTION

Not a day goes by without seeing a mention of deep learning in the media. In a few short years, the impact deep learning has had on industry and popular culture is nothing short of remarkable. Back during my graduate student days, I made an annual pilgrimage to meet with a tight community of academics and researchers interested in neural networks (called NIPS then, now called NeurIPS). With the rise of interest in deep learning, it became so popular that organizers held a lottery for attendees for the 2019 conference.

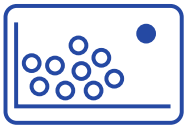
The excitement's understandable. Deep learning is responsible for several groundbreaking advancements that defied expectations of even leading researchers in the field. Over a single summer in 2010, for example, researchers at Microsoft used the technology to cut speech recognition errors - which had been stuck at 20 - 25% for over two decades - down to 15%. The solution outperformed professional human transcribers two years ago.

”

Versatility is one of deep learning’s most intriguing characteristics.

Allow me to explain.

APPLICATIONS



Concentric applies deep learning technology to the problem of data security. If you look at data security through a deep-learning lens, the problem has something in common with self-driving. Autonomous cars process an avalanche of information (cameras, radar, LiDAR) to make driving decisions. Concentric also processes an avalanche of information (words, sentences,and paragraphs, document location and usage) to make risk assessments. (Data scientists realize these are very different problems - but I offer the comparison to bring deep learning's versatility into sharper focus.)

RADIOLOGY
Highly specialized human expertise

DRIVING
Finely honed physical skills and close attention

MAIL SORTING
High visual acuity

LEARN MORE

DOCTORS
WWW

A radiologist's guide to deep learning

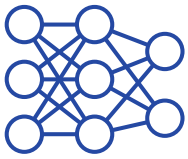
AUTOS
WWW

Deep learning techniques for autonomous driving

PROCESSORS
WWW

Deep learning for mail processing

FUNDAMENTAL NEURAL NETWORK CONCEPTS



To understand deep learning, we need to go all the way back to 1943. Warren McCulloch and Walter Pitts, for the first time, proposed a mathematical model of an artificial neuron as a simple computing machine. Individual neurons have many inputs and a single binary output. A neuron's weighted and summed inputs determine its output.

IMPLICATIONS

McCulloch and Pitts observed that a network of such computing units, in principle, could compute any possible boolean function.

WARREN McCULLOCH

Neurophysiologist

BACKGROUND
MD from Columbia University of Physicians and Surgeons

Department of Psychiatry at the University of Illinois, Chicago

WALTER PITTS

Logician

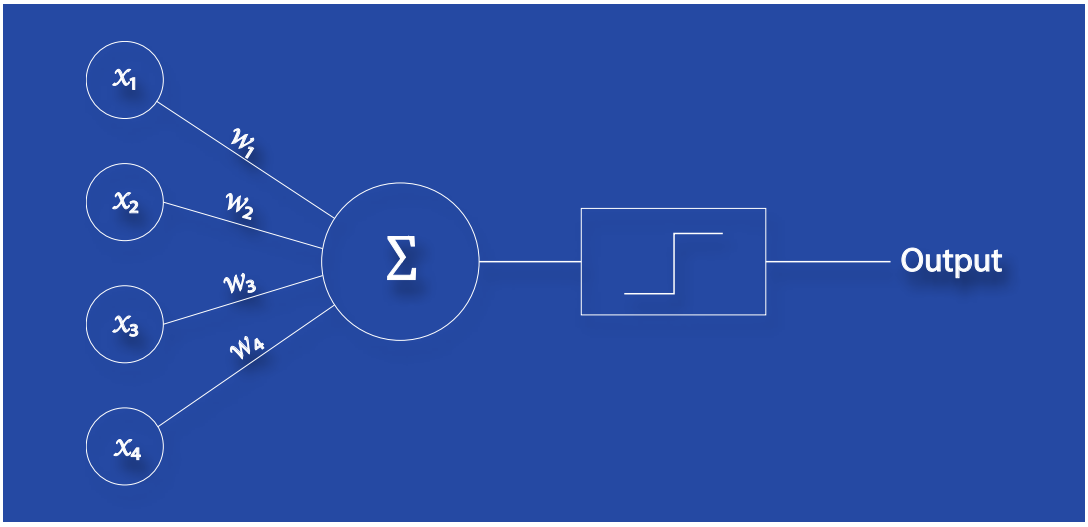
BACKGROUND
An autodidact with extensive informal training at the University of Chicago

CONTRIBUTION

"A Logical Calculus of Ideas Immanent in Nervous Activity"

A seminal contribution to neural network theory that was inspired by nature: McCulloch and Pitts developed their model after considering the brain as an information processing device

THE PERCEPTRON



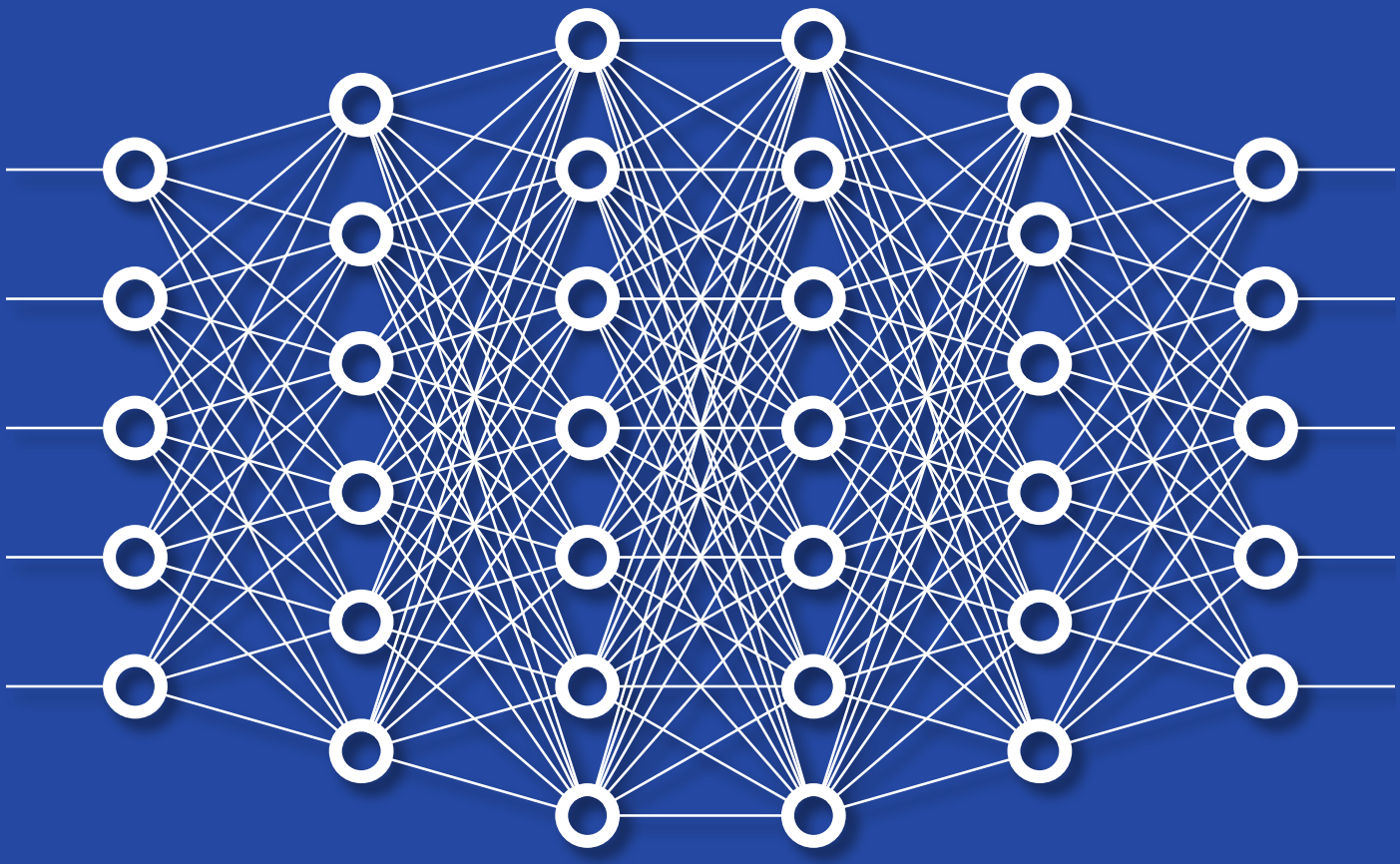
The first trainable neural network - the perceptron - was introduced by Frank Rosenblatt in 1957. The proposed neural network was capable of simple linear classifications. Researchers theorized networks could be built with multiple hidden layers but limitations of computing power and fundamental algorithmic difficulties prevented any meaningful demonstration of their usefulness.[1]

FRANK ROSENBLATT
Psychologist

BACKGROUND
Cornell Aeronautical Laboratory

Built the Mark I Perceptron in 1960

[1]Marvin Minsky and Seymour Papert's 1969 book "Perceptrons" outlined limitations of Rosenblatt's technique and demonstrated a few simple functions (such as boolean XOR) the perceptron was unable to model. The book had a chilling effect on neural network research for almost two decades.



NEURAL NETWORKS

The perceptron was the starting point for what we now know as neural networks, composed of computing "neurons" arranged in layers with data flowing from one layer onto the next. As data flows through each layer, it gets transformed as the outputs of each neuron feed into the next layer as data inputs. The passengers in a car see only the visible layers: a light turns red, the self-driving car comes to a stop. All the while, unseen hidden layers process large volumes of visual and sensor data to arrive at the decision to apply the brakes.

VERSATILITY, EXPLAINED

UNIVERSAL APPROXIMATION THEOREM

A groundbreaking result from the 80s by Cybenko proved that a neural network with a single hidden layer and a finite number of neurons could approximate any continuous mathematical function to arbitrary precision. Often referred to as the universal approximation theorem, this is the foundation of why deep learning is so versatile.

But how do we go from a technology that can "approximate any continuous mathematical function" to a place where it finds brain lesions as accurately as a skilled radiologist? Wrapping your head around how this can possibly happen, I admit, isn't easy. Tasks like self-driving, radiology, and data security are monumentally complex. It seems impossible they could be modeled like some data regression from a high school physics experiment.

But they can.

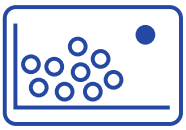
These tasks are, in fact, composed of a finite number of inputs (e.g. the patterns and shades on an x-ray or the context and use of a document) with unequivocally correct and repeatable outputs (the patient either does or doesn't have a brain tumor; the document does or doesn't contain sensitive information). Those inputs are complex. The insights needed to arrive at an answer are incredibly sophisticated. Deep learning is the technology that can do it.

”

Enough with the pleasantries.

Let’s dig in.

MODEL ANYTHING



> According to Cybenko,
>
> Arbitrary decision regions can be arbitrarily well approximated by continuous feedforward neural networks with only a single internal, hidden layer and any continuous sigmoidal nonlinearity.

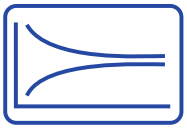
This fascinating result explains why neural networks work - essentially it shows that neural networks can, in principle, model anything. The basic idea is as follows: computations in each hidden neuron can be approximated as a step function. The contribution of each step to the output can be tuned based on the values chosen for each neuron's parameters. These simple building blocks combine into a network that's capable of modeling any arbitrary continuous function to an arbitrary level of precision. Curious readers are encouraged to explore this remarkable visual explanation of how it works. Go ahead, I'll be here when you get back.

THAT WAS FUN, WASN'T IT?

If you made it to the curve-fitting exercise you're a believer now. Once we understand that even highly complex processes - evaluating a medical image, driving a car, assessing risk in millions of documents - can be captured as mathematical models (ridiculously complex models, yes, but still models) the challenge changes from "if" to "how." How do we build solutions that can perform these tasks quickly and accurately? I had a physics teacher once claim that if we knew the momentum and position of every particle in the universe, we could predict the future. Obviously some systems are too complex for even the mighty neural network.

That, of course, begs the question: what are the limits? Just how deep can deep learning go?

NEURAL NETWORKS GO DEEP



DEEP VS. SHALLOW

By definition, deep learning encompasses a family of neural network models that have many layers of neurons as opposed to "shallow" networks which have fewer. The number of neurons may be similar but the arrangement is very different. And while flatter networks are, in theory, just as capable as deep learning models, deeper networks have some important advantages.

Deep learning networks are, for a given number of neurons, more accurate than shallower ones. (In the machine learning vernacular, we call this "performance," although I'll use "accuracy" here to avoid confusion with speed or other definitions of performance.) Comparisons between the relatively shallow VGGNet model (with 16 layers and ~140M parameters) show the deeper ResNet model (with 152 layers but only ~2M parameters) to be more accurate. In fact, it can be shown that for approximating certain functions, shallow networks will need exponentially more neurons compared to a deep network. And since the deeper model requires fewer parameters, it also alleviates the problem of overfitting (where a model is so tightly tuned to a sample data set that its predictive ability takes a nosedive when used on real-world data).

Deep learning techniques have another advantage: they're better at modeling data that's inherently hierarchical. In nature, for example, the cascade of neurons leading from the retina to the brain is hierarchical. At each step (or "layer" in the deep-learning vernacular) neurons become selective for increasingly complex stimuli. Retinal neurons connect to neurons specialized to sort out bars and edges. Those neurons connect to neurons specializing in orientations and positions. Orientations and positions feed into corners and features. And so on, until the brain recognizes a face.

VISUALIZING

Here's another analogy to help you visualize what the data itself looks like as layers work in a computerized deep learning network. Suppose you're given a ball of paper with 1,000 gridpoints on it. As you uncrumple the paper, you'll record the three dimensional position of each gridpoint in a journal. Our first entry would be chaotic: the 1,000 coordinates would look completely random. As we continue uncrumpling the sheet, we'll occasionally pause to record positions for each coordinate. (You can think of each recording as a "layer" in our deep learning model).

What will our journal look like over time? It will look less and less chaotic until you finally smooth out the paper on your desk. Now the data reveals the paper for what it is: a 2 dimensional object. (Thanks to Francois Chollet for this visual analogy). Like any analogy, don't take this one too far - it's intended simply as a way for you to imagine what the data looks like as it's processed.

At Concentric, deep learning reveals the meaning in the millions of files created and used by an organization's employees. You can think of these files as points on the balled-up sheet of paper. As we "uncrumple" it, our Semantic Intelligence© solution reveals clusters of files with similar meaning. For security practitioners, understanding what you have is an essential step before you can protect it.

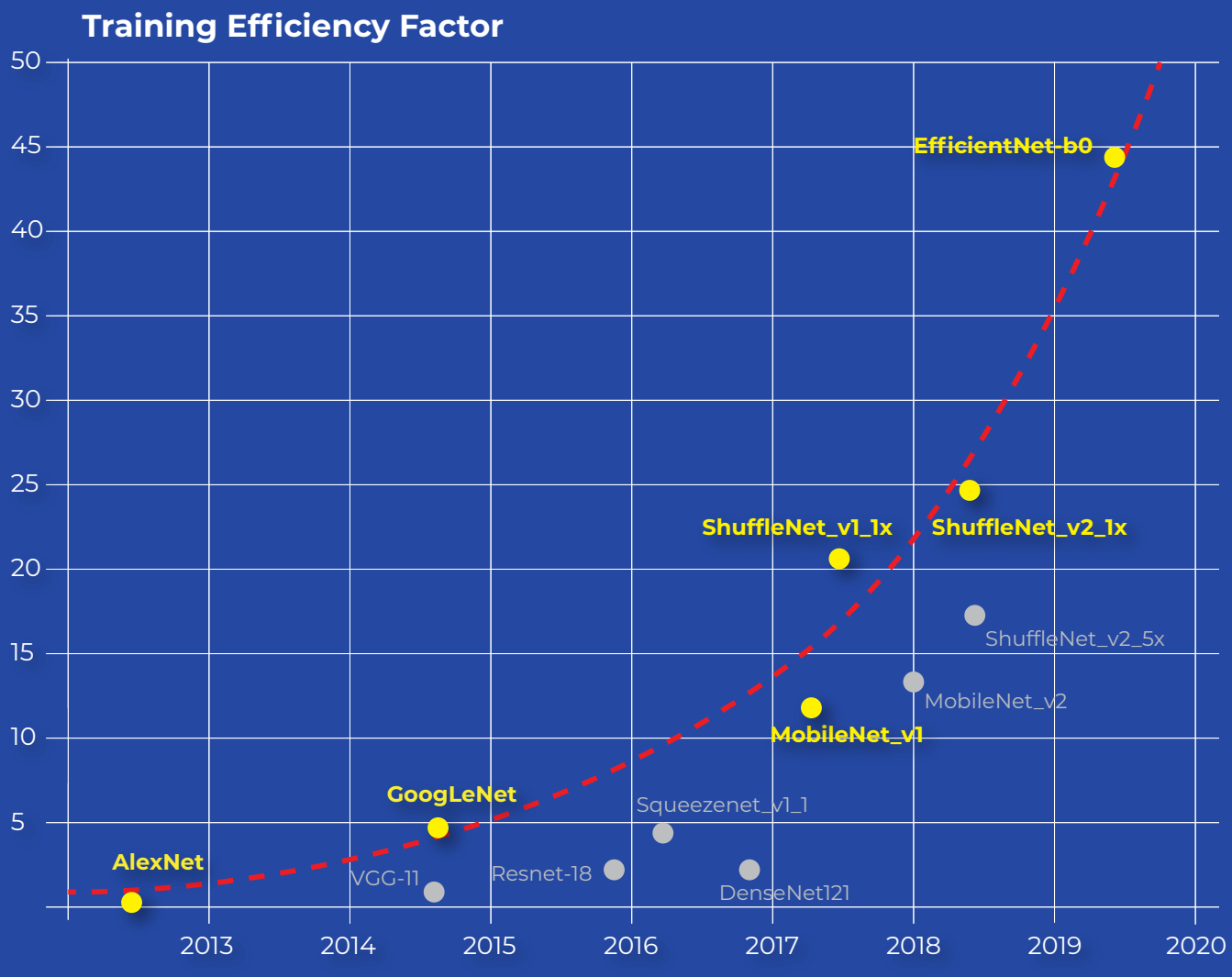
OKAY, BUT HOW?

IT STARTS WITH TRAINING

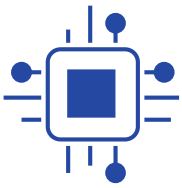
Training is the key to establishing a model that solves real-world problems. At the most basic level, "training" means adjusting the "settings" for each artificial neuron in our network of neurons. (Remember that visual explanation I mentioned earlier? If you haven't already, go give it a try. It'll clarify this notion of neuron settings.)

In 1974, Paul Werbos introduced the backpropagation algorithm as a way to train the parameters in a neural network. It's an iterative method that compares the model's output to a known set of training data. After each iteration, errors are fed back into the model to guide the adjustment of model parameters. Adjusting parameters in the hidden inner layers is tricky. Werbos' algorithm gave us an elegant way to propagate output errors back into the inner layers where they are used to tune model parameters.

Backpropagation works fine for small, shallow networks but it completely breaks down with deep architectures. This is because each neuron needs to "see" its contribution to the model's error. In a deep model, error signals decrease exponentially the deeper you go. It doesn't take long before we hit the limits of computer precision and the error signals become essentially invisible. This is called the "vanishing gradient problem," and it has been the focus of much work in the field.


Training Efficiency Factor
50
45
40
35
30
25
20
15
10
5
2013
2014
2015
2016
2017
2018
2019
2020
EfficientNet-b0
ShuffleNet_v1_1x
ShuffleNet_v2_1x
ShuffleNet_v2_5x
MobileNet_v2
MobileNet_v1
GoogLeNet
Squeezenet_v1_1
AlexNet
VGG-11
Resnet-18
DenseNet121

MOORE'S LAW FOR NEURAL NETWORKS



BREAKTHROUGH

Many researchers (notably Schmidhuber) proposed methods to overcome the vanishing gradient problem. But it was Geoffrey Hinton's breakthrough work in 2006 that cleared the way for complex deep neural networks. He proposed training two layers at a time (instead of trying to train all the layers at once), using the output from each pair of layers as the input for the next. These layer pairs are known as Restricted Boltzmann Machines and, when stacked into a single deep network, they make training practical without sacrificing accuracy.

Hinton's influential work made complex deep neural networks feasible for real-world applications and it took academic researchers in new directions. Training algorithms are improving at a breathtaking rate. Recent analysis from OpenAI suggests we're seeing gains akin to Moore's law: the amount of compute needed to train a neural net to the same performance on ImageNet classification (a dataset used to benchmark image recognition algorithms - more on ImageNet to come) has been decreasing by a factor of 2 every 16 months. Compared to AlexNet in 2012 (the best performing image recognition algorithm at the time) it now takes 44 times less compute to train a neural network model to the same level of accuracy.

THE REST OF THE PICTURE

Training improvements are only one reason deep learning has come into its own. Two other developments - the availability of large datasets and the dramatic increase in available computing resources - contribute to deep learning's practicality.

AVAILABILITY OF COMPUTATIONAL RESOURCES

Compared to traditional modeling techniques, deep learning requires orders of magnitude more compute power. Nvidia's CUDA programming platform, released in 2007, allowed researchers to leverage their GPU's general purpose parallel processing capabilities for deep learning computations. A breakthrough paper in 2009 showed that massively parallel graphics processors were over 70 times faster than multi-core CPUs. That's had a dramatic impact on training (which used to be a lengthy, iterative process.) Now training experiments that once took weeks are done in a few hours. Today, academia as well as industry is working on hardware designed specifically for deep learning and it has developed into an exciting area of research.

AVAILABILITY OF LARGE DATASETS

Models with more parameters can reflect increasingly complex underlying systems. (As a professor of mine in grad school used to say - “give me 5 or more parameters and I can fit an elephant.”) But as the parameter count grows, so does the problem of overfitting. Overfitting becomes a real concern when the number of parameters gets close to the number of samples in the dataset used to develop the model. And as our models grow to hundreds of thousands or even millions of neurons, we need larger and larger datasets to ensure model accuracy.

For the longest time, neural network researchers used a small number of standard datasets to benchmark accuracy of their new methods and algorithms. In image recognition for example, one of the gold standards was the “MNIST Database of Handwritten Digits.” But it only had 70,000 examples and, as image recognition models improved, the database was no longer a challenge. Every algorithm worked equally well and it was impossible to measure whether an algorithm advanced the state of the art. It was a significant barrier to progress.

In 2006, Fei-Fei Li, then a CS professor at University of Illinois Urbana-Champaign recognized this problem and worked to create the ImageNet dataset, culminating with the ImageNet Challenge in 2010. It was a competition for teams to try out their new algorithms and benchmark against other approaches. ImageNet was an important catalyst for image recognition research and it inspired others in related areas to focus on building more extensive datasets as well.

WHAT'S THE DIFFERENCE?

Deep learning is a specialization within the discipline of machine learning. But deep learning has departed significantly from its machine learning roots to become a unique and fundamentally different approach. These differences boil down to two factors.

REPRESENTATION LEARNING

Deep learning learns features from the data instead of relying on human experts. To help clarify, let me explain what "representation" and "features" are. Every model (in machine learning and elsewhere) uses a set of inputs to capture the essence of the thing being modeled. If you were estimating home prices, for example, your inputs would need to include square footage and zip code (at least) or your model wouldn't have much predictive power. The collection of features needed to build an optimal model is the "representation."

Traditional machine learning models use humans to define features. That can work reasonably well in some cases (like homes and their prices). But for more complex phenomena, human reasoning and expertise often fail.

Document classification is a good example. Early machine learning models classified documents by counting the frequency of specific words in a document. As you can imagine, that approach misses many of the nuances found in human language. Does the word "bank," for example, mean a financial institution or the edge of a river? Nearby words like "river" or "ATM" might be clues. Finding the right features to represent a set of documents is a complex problem (and there are many more nuances beyond word proximity that complicate the issue).

NEW CATEGORIES

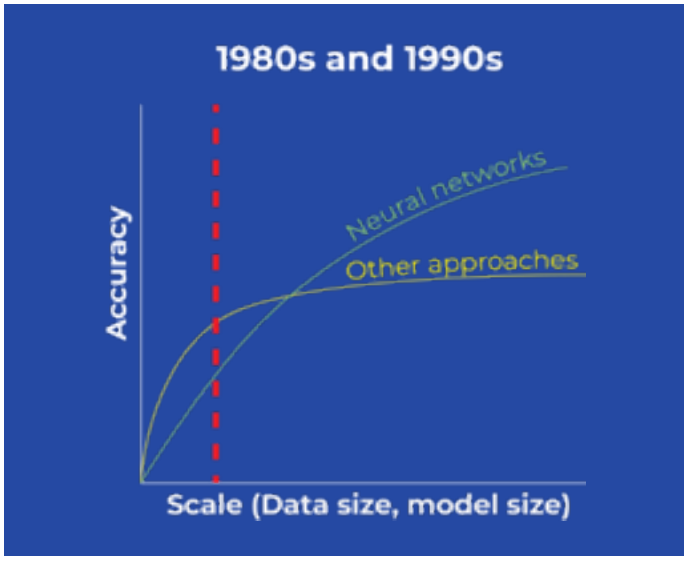
It might seem almost fantastical that a deep learning algorithm could, without any guidance whatsoever, accurately determine the right features to represent an unfamiliar set of documents. But it's not.

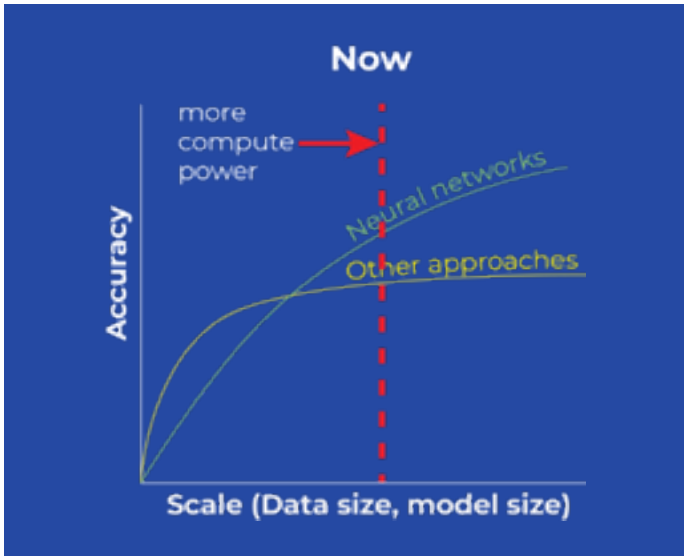
Think of it this way - were you to find yourself in a foreign country, you'd probably recognize a stop sign on the street as a stop sign (even if you didn't understand the language). You've never seen the

sign before, but you know what matters: it's at an intersection, it's red, it's oriented so drivers can see it and (in most places in the world) it's an octagon. Easy. You now have mental categories for Chinese and Moroccan stop signs. Nature is an expert at representation learning.

Representation learning today is an active area of research. Models such as BERT in the natural language processing field have demonstrated how task-independent representations can be effectively specialized for a variety of applications, like data security.

CAPACITY





MODEL CAPACITY

Deep learning can encode more information than traditional machine learning. The graphs here are from a talk by Jeff Dean, who notes that deep learning performance improves with more data.

In contrast, traditional machine learning performance plateaus beyond a certain dataset size. With deep learning (supported by the GPU-based compute power increases discussed earlier) modelers can build larger and larger models and train with larger and larger datasets.

In practice, this almost invariably leads to better accuracy. In the field of natural language processing, just last year we've seen publications of increasingly larger language models such as BERT (345M parameters), GPT-2 (2.5B parameters) and GPT-2 8B (8B parameters).

We don’t fully understand why deep learning is so effective.

WHAT'S AHEAD?

Deep learning is a powerful technology with the ability to effectively perform a wide variety of highly sophisticated tasks. The field has matured with hardware, tools and infrastructure ecosystems that allow engineers and practitioners to quickly prototype, test and deploy complex deep learning models in their domains. Novel applications of deep learning to real-world problems continue unabated.

We don't yet fully understand why deep learning is so effective. Our current mathematical frameworks lead us to predict far less accurate results. Terry Sejnowski calls this the "unreasonable effectiveness of deep learning," and it has inspired much academic research into the geometry of high-dimensional spaces and neural network models as high-dimensional dynamical systems. I encourage interested readers to read Sejnowski's entire article for a glimpse of what the future might hold - including his thoughts on the possibility of a generalized artificial intelligence.

ABOUT THE AUTHOR

Dr. Shashanka is Concentric's Chief Scientist and Co-Founder. Before Concentric, he was the Managing Director of Charles Schwab's Data Science and Machine Learning team. He also co-founded PetaSecure, where he served as Chief Scientist before the company was acquired by Niara.

He is a senior member of the IEEE and served as Associate Editor for IEEE Transactions on Neural Networks and Learning Systems. Dr. Shashanka received his doctorate in computational neuroscience from Boston University and a degree in computer science from the Birla Institute of Technology and Science, Pilani.

Madhu lives in Austin, Texas with his wife and two children.

CONCENTRIC

Concentric applies deep learning models for language to the task of data security. We use these models to understand the meaning of the millions of documents employees create and use every day. Deep learning allows us to automate data discovery and classification so security practitioners can effectively protect their most critical data.

www.concentric.ai
twitter: @IncConcentric
linkedin: linkedin.com/company/concentricinc

4340 Stevens Creek Blvd
Suite 112
San Jose, CA 95129

Vatika Business Centre,
Cessna Business Park
5th Floor, Embassy Signet
Kadubeesanahalli
Outer Ring Road
Bengaluru, India 560103