

THE HISTORY AND ECONOMICS OF ARTIFICIAL INTELLIGENCE

From Neuron to Trillion-Dollar Capex

A story in twelve chapters about how artificial
intelligence got here, and who makes money from it

BAZYLI ZYGAN

PHOTO: THE SILICON OF A GEFORCE GTX 580 • FRITZCHENS FRITZ • CC0

From Neuron to Trillion-Dollar Capex

A story in twelve chapters about how artificial intelligence got here, and who makes money from it

Copyright © 2026 Bazyli Zygan

Text and drawings © Bazyli Zygan. All rights reserved. No part of the text or the drawings may be reproduced or distributed without the author's written permission, except as permitted by law.

The photographs and reproductions are the work of others and remain under their own licences (Creative Commons, public domain and others). The author, source, licence and changes of each are listed under "Picture credits" (p. 318), and the terms for reusing them under "Licences and legal notes" (p. 330). The sources the text relies on are gathered in the bibliography (p. 306).

Figures, prices, valuations and other facts that change are given as they stood on 10 October 2026, unless the text says otherwise.

This book is written for general and educational purposes. It is not investment advice, nor a recommendation to buy or sell any financial instrument.

On the title page: The silicon of a GeForce GTX 580 · Fritzchens Fritz · CC0.

Set in Manrope and JetBrains Mono (SIL Open Font License 1.1).

Contents

Introduction

6

An introduction to a story in twelve chapters about how artificial intelligence got here, in which two graphics cards are bought for entirely the wrong reasons and, fourteen years later, the electricity grid begins to look nervous.

PART I: INTELLIGENCE How AI works

01 How Machines Learn

11

In which a machine the size of a room is promised the Earth, a book spoils everything for a generation, and learning turns out to be a matter of walking downhill in fog.

02 Machines That See

26

In which a cat ignores a slide show, a professor counts photographs, and two graphics cards finally get a proper job.

03 Attention Is All You Need

41

In which a sentence is passed down a row of clerks until it forgets how it began, eight people in a corridor decide to stop taking turns, and a paper named after a pop song rearranges an industry.

04 Building a Language Model

61

In which a machine is set the longest exam in history, invents six court cases, is taught manners by forty contractors, and presents two bills.

PART II: COMPUTE & ENERGY The physical layer

05 The Economics of Silicon

85

In which a company is planned over bottomless coffee in a diner, a man of fifty-five opens a factory with no products of its own, a machine arrives in three aeroplanes, and everybody in the queue turns out to be paying the same shop.

06 Power & Alternative Compute

109

In which a nuclear power station is raised from the dead to answer questions, the largest electricity grid in America asks the cleverest machines on Earth whether they would mind running on diesel, and several people propose to do the sums with water, with light, and with something that has the brain of an owl.

PART III: THE FRONTIER Generative and next-generation

07 Pictures in Motion 136

In which a railway baron pays to find out what a horse does with its legs, an idea is had in a pub, a schoolteacher near Hamburg lists five billion pictures, a pope acquires a coat, and the most famous video machine in the world is switched off for costing too much.

08 Beyond the Transformer 162

In which a horse in Berlin does sums for thirteen experts, a theorem that a nineteen-year-old helped to prove waits sixty-seven years for a job, two scientists bet on the first of January 2027, the machines are taught to think before they speak and send in the bill, and one of them is put in charge of a small shop and sells tungsten at a loss.

PART IV: SOVEREIGN RACE US, China, Europe

09 The DeepSeek Moment 190

In which a French priest in a Chinese town writes home about plates, a hedge fund in Hangzhou takes up a hobby and keeps its notes in shorthand, a caveat that nobody reads costs a chip-maker 589 billion dollars in a day, a professor explains what a BMW has in common with a garbage truck, and the most valuable secret of the age turns out to be guarded by the terms and conditions.

10 China Today 223

In which a man of whom nothing else is known gives away the oldest dated printed book in the world, a word for Javanese rice fields reaches the Politburo by bicycle, the head of a search company calls free software a tax on fools and then releases ten free models, a rule invented at Stanford in 1959 teaches Chinese machines to cross things out, a price war stops because the shop has run out of stock, and a stock market values a laboratory, briefly, at sixteen hundred times its sales.

11 Europe: Player or Spectator? 255

In which a spectacle-maker is refused a patent on the telescope because it cannot be kept secret, a company that began in a leaky shed becomes the only maker of a machine that mostly existed already, three Frenchmen raise a hundred and five million euros on seven pages, an eagle that is not strictly an eagle is hatched by volunteers on a supercomputer in Kraków, and a continent postpones the rule book it had said it would not postpone.

12 Mapping the Value Chain 280

In which a young man from Boston sells frozen pond water in the West Indies and goes to prison over it, a philosopher watches his pond set off for Calcutta, a businessman in Taipei draws a smile, a stick of memory turns out to earn more on the dollar than the most valuable company in the world, a search engine borrows money for a hundred years, a brewery opens a bar next door to its best customers, and the book ends, as booms do, with somebody counting.

Glossary	304
Bibliography	306
Picture credits	318
Licences and legal notes	330

Introduction

An introduction to a story in twelve chapters about how artificial intelligence got here, in which two graphics cards are bought for entirely the wrong reasons and, fourteen years later, the electricity grid begins to look nervous.

COVER: DATA CENTRES NEAR ASHBURN, VIRGINIA • THEODORE CHRISTOPHER • CCO

01

Two pictures

Stories about artificial intelligence are supposed to begin with a robot, preferably one with red eyes and a grievance. This one begins with a shopping trip.

In 2012 a graduate student in Toronto trained a neural network on two graphics cards. They were GeForce GTX 580s, a product engineered, with great care and at considerable expense, so that teenagers could shoot one another in higher resolution. Nobody at the factory had artificial intelligence in mind. The cards did not know they were about to change the world. Graphics cards rarely do.*



Picture one. A GeForce GTX 580, launched in late 2010 at about five hundred dollars, for people who took their video games seriously. Two of these did the work.

PHOTO: THESTRIKER • CC BY-SA 4.0

* They are, on the whole, incurious objects. This is one of several ways in which they differ from the people who buy them.

Now the second picture. Today, companies are building data centres for AI whose appetite for electricity rivals that of a small city. They have their own substations. They have their own weather, more or less. Planning officers speak of them in the hushed tones once reserved for cathedrals and motorways.



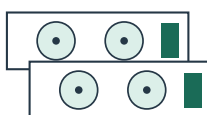
Picture two. A Google data centre in Council Bluffs, Iowa, photographed in 2017, which in this industry counts as ancient history. The ones going up now are larger.

PHOTO: CHAD DAVIS · CC BY 2.0

Two graphics cards to a power station's worth of demand, in about fourteen years. That is not a trend. That is something falling off a cliff, upwards.

Fourteen years of appetite

2012: A BEDROOM IN TORONTO



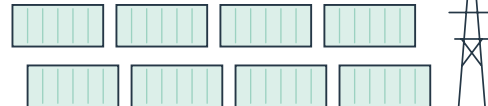
Two graphics cards

Under 500 watts between them.
A toaster uses more.

× 2.5 million

FOURTEEN YEARS

2026: A CAMPUS IN TEXAS



Eight buildings, one electricity bill

1.2 gigawatts planned: enough, by Oracle's reckoning, for a million homes.

The two pictures, with the meter running. Flat out, the two cards drew 488 watts. One campus now going up in Abilene, Texas, is planned for 1.2 gigawatts. Few hobbies have got so far out of hand.

This story asks one question about the gap between the two pictures: **what happened in between, and who captures the value?** The first half is a tale of ideas, stubborn people and a great deal of multiplication. The second half is about money, because that is where the story is now being written, and because everybody is paying for it.

02

Why it pays to know how the trick is done

There is a particular kind of conversation, at a meeting, over dinner or on the evening news, in which somebody says “foundation model” and everybody nods. Then somebody says “inference costs” and everybody nods again, slightly faster. By the time the word “parameters” arrives, the nodding has acquired a rhythm, and nobody dares stop in case they are asked a question.*

We would like to be the people in that conversation who actually know. Not in order to build these machines, which is a job for people who enjoy linear algebra and are best left to it, but in order to judge them: whether they are worth what is being spent on them, who pays, and who gets paid. It is hard to judge a thing you cannot picture. It is harder still to notice when its economics are about to change, which in this industry they do roughly every eighteen months, usually on a Tuesday.

So the story runs to twelve chapters. It starts with a single artificial neuron, which turns out to be about as complicated as a member of an investment committee.† It ends with chips, power stations and the geopolitics of who is allowed to buy what.

The narrator of these chapters follows money for a living, which is why investment committees, deals and loan applications keep wandering into them. The reader need do no such thing. Money is simply the quickest way to see who is winning, and the narrator promises to show the working.

No mathematics is required. What is required is curiosity, and a willingness to ask the question that everyone else in the room is quietly hoping somebody will ask.

03

Three threads

Twelve chapters is a lot of string. Three threads run through all of it, and if you hold on to them you will not get lost.

Architecture. Each breakthrough fixes the limits of the one before. The history of AI is, to a first approximation, a long line of clever people being annoyed by the previous clever idea.

Data, compute and power. Scale is the fuel. Chips and electricity set the limits. Sooner or later every elegant idea in this field has to go and stand in a queue for hardware, behind a larger company.

* This is known as consensus.

† No disrespect is intended to either party.

Who captures the value? This one matters to anyone with money at stake, and through pension funds, electricity bills and taxes that is most of us. A technology can change the world and still make its inventors very little money, as the inventors of most things could tell you. Every chapter therefore ends with the bill: where the margin sits, who has a moat, and whose moat is in fact a puddle.

04

The map

The twelve chapters come in four parts. The first explains how the thing thinks, the second what it eats, the third where it is going, and the fourth who is fighting over it.

01 How Machines Learn

Neurons, layers and gradient descent, and why AI waited 30 years for its moment.

02 Machines That See

CNNs and the 2012 ImageNet moment, when gaming GPUs entered AI.

03 Attention Is All You Need

From RNNs to the Transformer, the architecture built for parallel hardware.

04 Building a Language Model

Pretraining, scaling laws and RLHF. What it really takes to train, and to serve.

05 The Economics of Silicon

Why GPUs rule, why they cost so much, and who controls the supply chain.

06 Power & Alternative Compute

AI's electricity bill, and the challengers: analog, photonic and neuromorphic chips.

07 Pictures in Motion

GANs, diffusion and Stable Diffusion, and why video is the hardest problem yet.

08 Beyond the Transformer

KAN, Mamba, Mixture of Experts, reasoning models and agents. Does scaling still work?

09 The DeepSeek Moment

How a Chinese model shook the market, and the distillation wars that followed.

10 China Today

GLM, Kimi and Seedance: efficiency, open weights and price as strategy.

11 Europe: Player or Spectator?

Mistral, Bielefeld, ASML and the AI Act. Where Europe leads, and where it lags.

12 Mapping the Value Chain

Chips, cloud, models, apps: where the margins sit, the risks, and what to watch.

If some of those words look like a cat walking across a keyboard, that is as it should be. Each is explained when its chapter comes, and none is as clever as it sounds.

How to read these chapters

Each chapter tells one part of the story at full length, with the people in it. They can be read in order, or one at a time, as the mood takes you.

A few things hold for all of them:

- **The facts are real.** Every date, name and number has been checked, as things stood in October 2026. Only the tone is unserious.
- **The photographs are of the actual people and machines**, not of a glowing blue brain with circuit lines on it. * Credits sit at the end of each chapter.
- **Where a process can be drawn, it has been drawn**, on the principle that nobody ever understood a machine from a paragraph.
- **There is no mathematics.** Where a sum is unavoidable, it is the kind you could do on a napkin.
- **The footnotes are where the narrator goes to mutter.** You may skip them. It is, however, how people end up signing things.
- **Every chapter ends the same way:** with the bill, meaning who pays, who gets paid and who is safe behind a moat, and with three things worth keeping.

Start with the first. It opens in Buffalo, New York, in 1958, with a psychologist, a computer the size of a room and a press conference that got rather out of hand.

* There is a law, not yet repealed, that every article about AI must be illustrated with a glowing blue brain. We are prepared to face the consequences.

PART I: INTELLIGENCE

How Machines Learn

In which a machine the size of a room is promised the Earth, a book spoils everything for a generation, and learning turns out to be a matter of walking downhill in fog.

COVER: FOGGY PATH ON THE WAY UP MOUNT PILATUS, SWITZERLAND · GDMOLENA · CC BY-SA 4.0

01

Buffalo, 1958

In July 1958 the United States Navy called a press conference to show the world a machine that could tell left from right.

This was more impressive than it sounds.* It was also an odd thing for a navy to want, navies having had perfectly good words of their own for left and right for several hundred years. But the Navy was paying, and so the Navy got the press conference.



An IBM 704, photographed at a government laboratory in Virginia in 1957. Everything in the picture is one computer, apart from the two people, who are there to feed it. It got through less arithmetic in a day than a modern video doorbell does while waiting for the postman.

PHOTO: NASA · PUBLIC DOMAIN

* It is, even now, more than can be said for a number of people in multi-storey car parks.

The machine was an IBM 704: several tons of computer, occupying most of a room and all of the air conditioning. It had been fed a stack of punch cards. Some were marked on the left and some on the right. Nobody had given it a rule for telling them apart. After fifty tries it had worked out the difference for itself.

The man behind the demonstration was Frank Rosenblatt, a research psychologist at the Cornell Aeronautical Laboratory in Buffalo, New York, who was twenty-nine and had not yet learned to be careful. He called his invention the Perceptron, a name with the ring of something that either saves the world or menaces it in the second reel. He was by all accounts brilliant, and he made the classic mistake of brilliant people, which is to think aloud while journalists are in the room. He described it as the first machine capable of having an original idea.

The press took it from there, as the press will. The New York Times reported that the Navy expected the machine's descendants to walk, talk, see, write, reproduce themselves and be conscious of their own existence.* The New Yorker, later that year, judged it the first serious rival to the human brain.

It could tell left from right.

The gap between what a machine did on the day and what the newspapers say it will be doing by Christmas is worth noticing now, because it is going to follow us through all twelve chapters like a small and hopeful dog.

Two years later the idea was given a body. The Mark I Perceptron had a camera for an eye, made of 400 light sensors in a grid of twenty by twenty. Its knowledge lived in rows of dials, and when the machine learned, small electric motors physically turned them, so that a patient visitor could stand in the room and listen to it changing its mind. Remember the dials. They are about to become the most important objects in this story. There are trillions of them now, nobody can hear them turn, and they are still the whole trick.



The Mark I Perceptron being shown the letter C, in a US Navy press photograph released in June 1960. The Navy's caption promised a machine that could be trained to identify letters of the alphabet. It was right, eventually. The machine now lives in the Smithsonian, where nothing further is expected of it.

PHOTO: NATIONAL MUSEUM OF THE U.S. NAVY • PUBLIC DOMAIN

* This is a lot to ask of anybody. Most of us manage four of the six on a good day, and are not conscious of our own existence until after coffee.

Nearly everything in modern AI is descended from the machine in that photograph: the chatbots, the image generators, and the invoices for electricity. The excitement merely arrived about sixty years early, and the field paid dearly for it. But before we get to the bill, it is worth asking what Rosenblatt had actually built. The answer needs one idea, and the whole of this story stands on it.

02

Rules, and the lack of them

For fifty years software worked one way. A human being wrote down the rules, the computer was handed some data, and out came answers. In this arrangement the computer is a very fast and very obedient clerk. It does precisely what it was told, for ever, and with particular energy when what it was told was wrong.*

This is an excellent arrangement provided the human knows the rules. The trouble is the enormous number of things people can do without the faintest idea how they do them. You can recognise your own mother. Now try writing down the rules. “Two eyes, with a nose somewhere beneath them” describes her well enough, along with most of the people who are not your mother and a fair number of dogs.

Machine learning turns the arrangement round. You hand over the data together with the right answers, and the machine works out the rules for itself.

Two ways of putting a computer to work

THE OLD WAY



Excellent, provided the person knows the rules.

THE LEARNING WAY



Excellent, provided you have a great many examples. Explaining the rules afterwards is extra.

The whole revolution on one page. The boxes are the same in both rows. The green one has changed sides.

Take something close to home: deciding who gets a loan. In the old way, an analyst writes the rule. Approve if income is above a threshold and there are no past defaults. It works. It is also brittle, and it misses everything the analyst did not think of, which, analysts being human, is most things.

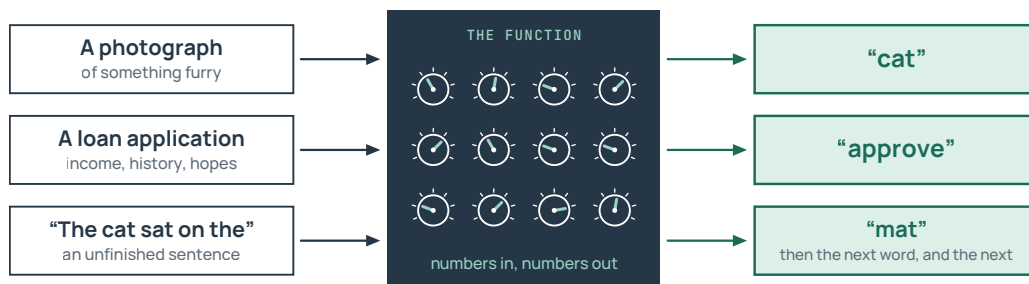
* It also does it several billion times a second. This is how a small slip in a spreadsheet becomes a large item on the evening news.

In the learning way, you show a model millions of past loans and how each one ended. It finds the patterns that predict repayment, including patterns no human would ever have written down. The price is that it becomes much harder to explain why it said yes. Ask the analyst and you get a reason. Ask the model and you get, in effect, a shrug worked out to eleven decimal places. Keep that in mind. It comes back later in the story, usually accompanied by a regulator.

If you take one idea away from this chapter, take this one. It is a small idea and will not need much room.

A model is just a function. A function, in the mathematician's sense, is a sausage machine: something goes in at one end, something else comes out at the other, and what happens in between is best not examined over lunch. A photo goes in, “cat” comes out. A loan application goes in, “approve” comes out. ChatGPT and Claude are the same kind of thing: text goes in, the next word comes out, and this is repeated very fast, in the manner of a dinner guest who has read the entire internet and insists on finishing your sentences. What makes them special is how large the function is, and how it was built.

A model is a function, and a function is a box of dials



A model with the lid off. There is nothing in the box but dials. All the art is in how they are set.

03

One neuron, or the committee member

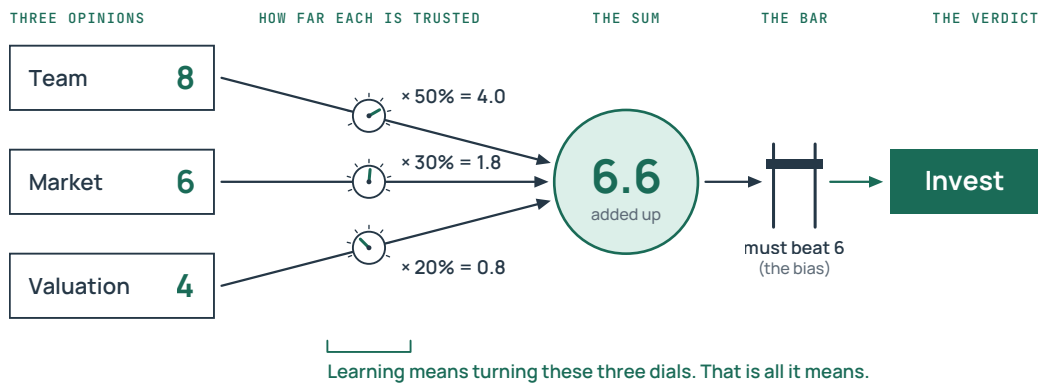
So what is inside the box? The basic building block is the artificial neuron, and it is a disappointment. People expect something that hums and glows and possibly has views on the future of humanity. What they get is a sum.

A neuron takes a few inputs. It multiplies each one by a **weight**, which says how much that input matters. It adds everything up, adds a **bias**, which you can think of as its personal threshold, and then decides: fire, or don't fire. That is the whole of it. There is no second page.

You have met this creature. It sits on investment committees.

Picture one member of the committee considering a deal. They score the team an eight, the market a six and the valuation a four. But they do not trust those three things equally. The team counts for half of the decision, the market for thirty percent and the valuation for twenty, valuation being a subject on which a certain flexibility has been known.

One neuron, drawn as a committee member



An artificial neuron on its way to a decision. Eight times a half, plus six times three tenths, plus four times a fifth, comes to 6.6. The bar is six. In it goes.

Their personal bar is six. The total is 6.6. They say invest.

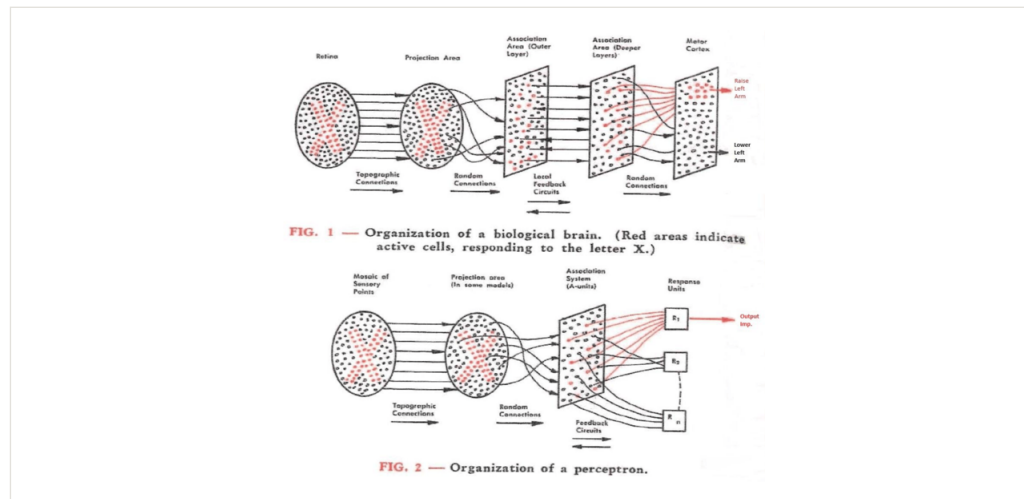
That is a neuron. Now comes the important part: **learning means adjusting the weights**. After watching how past deals turned out, our committee member may discover that they have been overweighting the team and underweighting the valuation, and shift accordingly.* A neural network does exactly this, automatically, millions of times, and without first needing to be taken to lunch.

The word "neuron" is borrowed from the brain, in roughly the way that "jumbo jet" is borrowed from the elephant. There is a family resemblance. It should not be leaned on.

That little experiment is the whole industry in miniature. **Capacity is the number of dials. Cost is the time it takes to turn them.** Everything since has been the same experiment with a larger egg.

The breakthrough, then, is stacking neurons in layers, each layer feeding the next: a committee of committees. Nobody sees the original paperwork except the most junior members at the bottom, and each level passes a summary up to the level above. Anyone who has worked in a large organisation will recognise the arrangement. The surprise is that here it works.

* Committee members have been known to resist this process. Neurons do not, which is the main reason they are easier to train.



Rosenblatt's own drawing from 1958: a brain above, his perceptron below. The resemblance was the whole point. It was also, as things turned out, the whole problem.

PHOTO: FRANK ROSENBLATT, 1958 • PUBLIC DOMAIN

04

A committee of committees

A single neuron can do exactly one trick. It draws a straight line through the data and declares: this side yes, that side no. It is a creature with a ruler and one firm opinion.

A small experiment shows what that means. It can be run on paper, where it is harder to break.

Imagine a sheet scattered with dots. There is a clump of orange dots in the middle and a ring of blue dots around it, like a fried egg painted by somebody who has only ever had one described to them. The job is to separate orange from blue. Give that job to one neuron and it draws its one straight line. Wherever the line goes, there is blue on both sides of it. You cannot cut the yolk out of an egg with a single stroke of a ruler. The neuron tries anyway, for as long as you let it, with the calm persistence of a man pushing a door marked PULL, and its error never improves.

Now give the job to four neurons working side by side. Each draws its own line. Four lines make a box, a box goes round the yolk, and within seconds the error collapses.

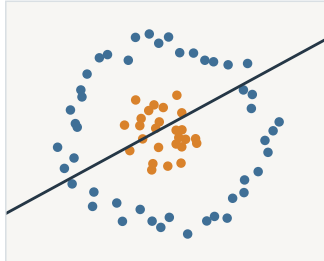
Now make it difficult: two colours wound round each other in a spiral, like a cinnamon bun. Four neurons are no use at all. It takes three layers of eight neurons each, about 180 adjustable dials, and a minute or two of patience. Even then, not every attempt gets there.

Show such a network millions of photographs and something remarkable happens. The first layer learns to spot edges. The next combines edges into shapes. The last recognises whole objects, and announces that this one is a cat. Nobody told it what an edge was. It found edges for itself, because edges helped it get the answer right, in

much the way that nobody tells a junior analyst what a red flag is, and yet by the third year they can smell one through a closed door.

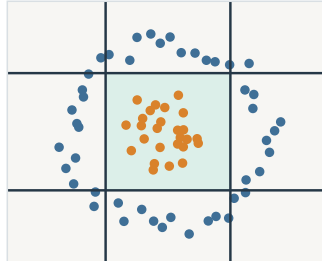
The fried egg problem

ONE NEURON: ONE LINE



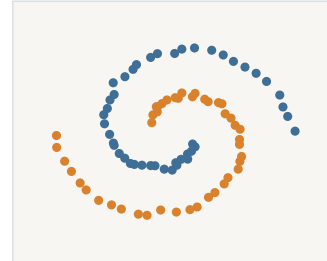
Blue on both sides, wherever you put the ruler. It never improves.

FOUR NEURONS: FOUR LINES



Four lines make a box. The box goes round the yolk. Done in seconds.

THE SPIRAL



Three layers of eight neurons, a minute of patience, and some luck.

Breakfast, three ways. One line cannot fence in a yolk. Four can. The bun needs a committee of committees.

This is all that “deep” means in deep learning. It is not a claim about wisdom. It means there are many layers, a fact which has disappointed a number of philosophers and at least one marketing department.

Every weight in the network is one adjustable dial, and the trade calls the dials **parameters**, presumably because “dials” did not sound expensive enough. Their numbers have got somewhat out of hand:

- AlexNet, the 2012 network trained on those two gaming cards, had about 60 million.
- GPT-3, in 2020, had 175 billion.
- Kimi K3, released by China’s Moonshot AI in July 2026, has 2.8 trillion by its maker’s count.

That is roughly 47,000 times more than AlexNet, in fourteen years. Numbers of that size slide off the mind like an egg off a plate, so try it with stationery. Allow one sheet of ordinary office paper for every million dials, and stack the sheets up.

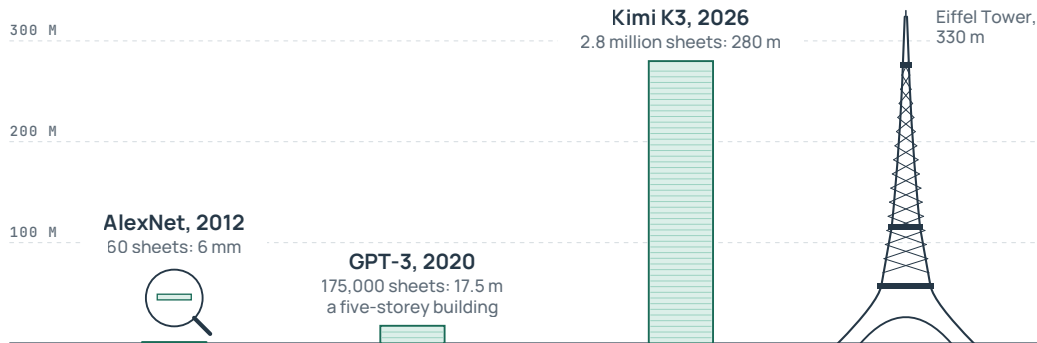
Every one of those dials has to be tuned during training, which is why a story about arithmetic turns, sooner or later, into a story about money and electricity.*

The size of the step matters enormously. Too big, and you stride clean over the valley and up the far side. Turn the step size up a hundredfold in our little spiral experiment and the error simply bounces about and never settles. Too small, and you will still be

* A word of caution: the number of parameters tells you how big the engine is and what it costs to run. It does not tell you how well the thing is driven.

walking next year. Choosing the step well is one of the practical arts of the trade, and those who are good at it are paid accordingly.

One sheet of paper for every million dials



AlexNet comes to a slim pamphlet. GPT-3 is a pile as high as a five-storey building. Kimi K3 would get you most of the way up the Eiffel Tower, and the view from the top is mainly of the electricity bill.

05

Fog on the mountain

How do you tune nearly three trillion dials? Not by hand, and not, any more, with little electric motors.* It takes three ideas, and they are the heart of the matter.

First, measure how wrong you are. The model predicts a seventy percent chance that a loan will default. In reality it was repaid in full. That gap is boiled down to a single number called the **loss**. High loss, very wrong. Low loss, nearly right. Training has precisely one ambition in life, which is to push that number down across millions of examples. Imagine a forecaster who was handed a score after every single forecast, to several decimal places, and was not allowed out of the room until it improved. It is unkind. It is also why the method works.

Second, walk downhill. Imagine you are on a mountain in thick fog and you would like to be in the valley, where the loss is low and there is, presumably, a pub. You cannot see the landscape. All you can do is feel the slope under your boots and take one step in the direction that goes down. Then you do it again. And again, a few million times. This is called **gradient descent**, and it is rather less athletic than it sounds.

* At one dial a second, with no breaks for tea, the motors would need about 89,000 years to get round them all once. Training goes round a great many times.



Gradient descent, as practised in Alaska. You cannot see the valley. You can feel which way is down.

PHOTO: JUSTIN MORGAN • CC BY-SA 2.0

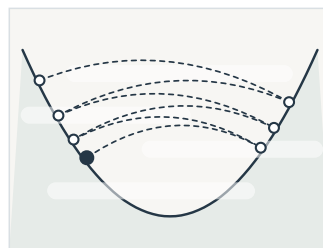
Walking downhill in fog, three ways

STEP TOO SMALL



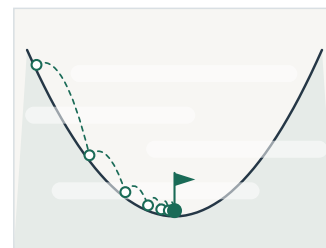
Perfectly safe, and still walking this time next year.

STEP TOO BIG



Strides clean over the valley and up the far side. Repeat.

ABOUT RIGHT



A sensible stride, a little shorter each time. Down in time for tea.

Three ways down a mountain you cannot see. The trade calls the length of the stride the learning rate, and has lost a good deal of sleep over it.

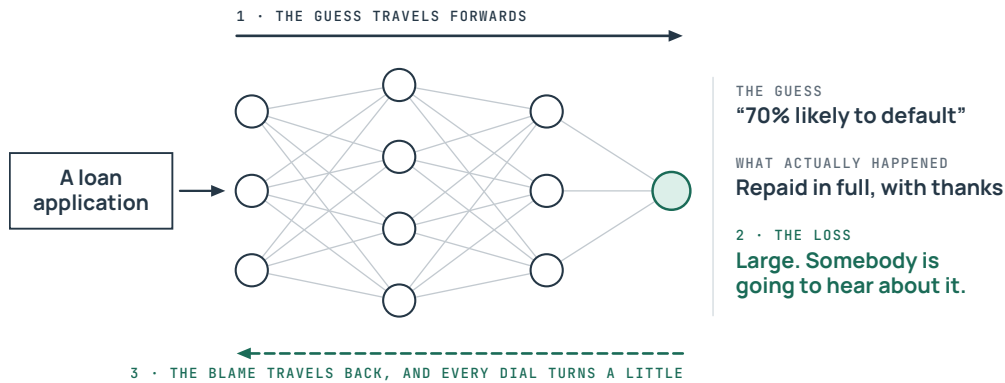
Third, find out whose fault it was. With trillions of dials, how do you know which ones to turn? The answer was made famous in 1986 by David Rumelhart, Geoffrey Hinton and Ronald Williams, and it is called **backpropagation**. The error at the output flows backwards through the network, layer by layer, and every dial is told how much it contributed to the mistake.

Think of the post-mortem after a bad deal. You trace back through the process, work out who contributed what, and everybody adjusts a little. The differences are that in a neural network nobody's feelings are hurt, nobody blames the market, nobody suggests that the real lesson is about communication, and the meeting takes a fraction of a second.

It is worth seeing how the blame is actually shared out, because the trick is simpler than its reputation. Take any one dial. What it needs to know is this: if it were turned a hair to the right, would the loss go up or down, and by how much? That is the slope under the boots on the foggy mountain, for one dial. It could be found by experiment: nudge the dial, run the whole network again and see what happens. For three trillion

dials that would mean three trillion reruns for every single step down the mountain, which makes the electric motors look brisk.

How the blame gets back to the dials



Backpropagation in one picture. The guess goes one way and the blame comes back the other. Every dial is told its share and turns a little. Then the whole thing happens again, a few billion times.



David Rumelhart at a neural network conference in Seattle, July 1991. Five years earlier he, Hinton and Williams had shown the world an efficient way to apportion blame.

PHOTO: ROLF KICKUTH • CC BY-SA 4.0

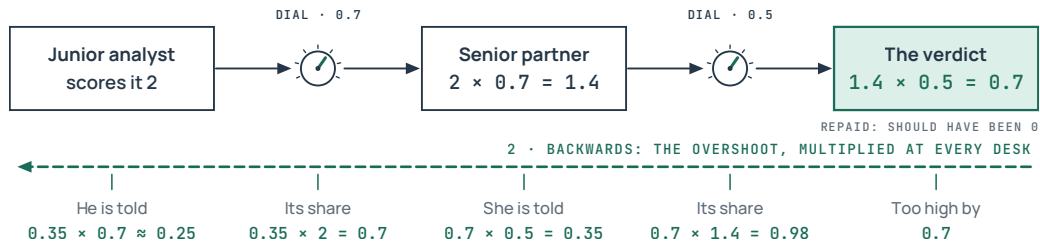
Backpropagation gets every answer from one trip backwards, and it manages this by never asking anybody to know more than their own job. Take the smallest committee there is: a junior analyst who reads the file and scores it, a senior partner who weighs the junior's score with a dial of her own, and a second dial between her and the final verdict. The verdict was 70 percent likely to default. The loan was repaid. The verdict was too high by 0.7, and that is where the blame starts.

The dial nearest the verdict goes first. Its share of the mistake depends on two things: how far the verdict overshot, and how much came in through that dial. A dial that carried a loud opinion did a great deal to push the verdict too high. A dial that carried a whisper did hardly anything. So its share is the one multiplied by the other. Then the partner is told how far off she was, which is the overshoot multiplied by the dial that connects her to the verdict: if her opinion counted for half, half of the overshoot is hers. Now she does exactly the same for the dial beneath her and passes a smaller note down to the junior, who does the same for his own dials. Nobody looks further

than one desk up and one desk down, which in most firms would be called working in silos and here is called efficiency. The note simply travels back through the building, multiplied at every desk, until every dial in it has its number.

Who pays for the overshoot

1 • FORWARDS: EACH OPINION, TIMES A DIAL



3 • EVERY DIAL TURNS AGAINST ITS SHARE, BY A TENTH OF IT

The dials go from 0.7 and 0.5 to 0.63 and 0.40. Next time the verdict is 51 percent, not 70. Nobody had to run the network again to find out.

The smallest committee in the world, after a bad loan. Opinions go forwards, each multiplied by a dial. The overshoot comes back, multiplied again at every desk, and every dial learns its share in a single trip.

Then every dial is turned a little against its share, and the next guess is a little better. In the drawing, turning each dial by a tenth of its share brings the verdict down from 70 percent to about 51 in one step. Real networks take much smaller steps, and a great many more of them.

Mathematicians call this the chain rule and teach it at school, usually without mentioning that it is worth a trillion dollars. Three things follow from it. First, a neuron that snapped between yes and no, like the committee member earlier in this chapter, would be no use here: a hair's turn of a dial would change nothing at all, and the blame would have nowhere to go. So the neurons in a trained network do not snap. They slide from “probably not” to “probably yes”, and their opinions can be nudged. Second, because the note is multiplied at every desk, it can shrink to almost nothing on its way down a tall building. Chapter 3 will meet this problem in a playground. Third, the trip backwards takes roughly twice as much arithmetic as the trip forwards, so one step of training costs about three times as much as simply giving an answer, before anybody has started counting the steps.*

One last observation from the same experiment. Go back to the fried egg, and the single neuron that could not solve it. Now tell that neuron one extra thing about each dot: how far it lies from the middle of the sheet. It solves the egg at once, alone, with

* Like most good ideas, this one was invented more than once before it became famous. A Finnish student, Seppo Linnainmaa, described the general method in his master's thesis in 1970, and an American, Paul Werbos, suggested using it on neural networks in his doctoral thesis in 1974. What the paper of 1986 added was proof that it worked, and that the middle layers learned useful things that nobody had told them, which is the part that sells.

no committee at all. Good data beats brute force. Put that somewhere safe; it will be worth money by the next chapter.

06

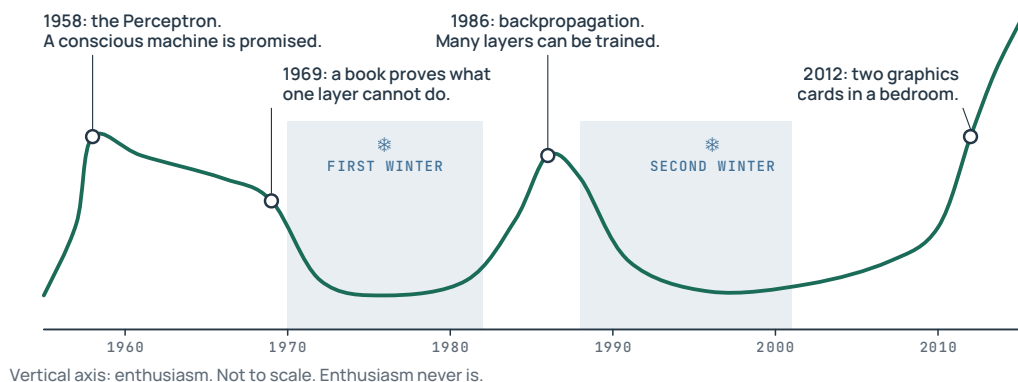
Great ideas, wrong decade

Here is the puzzle. Everything described so far, the neurons, the layers, the walk downhill and the apportioning of blame, existed by 1986. The AI boom began in 2012. What was everybody doing for a quarter of a century?

Mostly, they were being cold.

The trade speaks of **AI winters**: long stretches in which the money went away, the conferences shrank, and admitting at dinner that you worked on neural networks earned the kind of sympathetic silence usually given to people who say they are writing a novel. There were two of them.

Enthusiasm for neural networks, 1955 to 2015



Sixty years of weather. Two summers, two winters, and then whatever it is we are having now.

The first frost came out of a book. In 1969 Marvin Minsky and Seymour Papert, both at MIT, published *Perceptrons*, in which they showed with great rigour that a single layer of neurons cannot learn even a very simple pattern of the either-one-or-the-other-but-not-both kind.* They were right. It is our fried egg again: one straight line is not enough. The cure, more layers, was suspected at the time, but nobody yet knew how to train them, and the money did not wait to find out. Money rarely does.

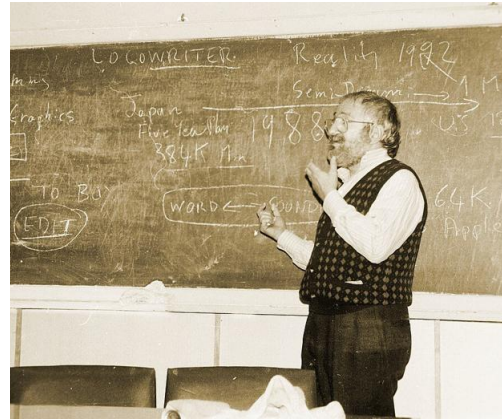
Rosenblatt did not live to see the thaw. He died in 1971, in a sailing accident, on his forty-third birthday. Nearly every machine in this story runs on the idea he was mocked for overselling.

* Soup or salad, as the menu puts it. Any waiter can enforce this. A single neuron cannot.



Marvin Minsky in 2012. He and Rosenblatt had been pupils at the same high school in the Bronx. Science is a small town.

PHOTO: JOJI ITO • CC BY 2.0



Seymour Papert lecturing in Moscow in 1987. By then he was better known for teaching children to program than for having put a research field into the freezer.

PHOTO: SHEN-MONTELLIER • CC BY-SA 4.0

The thaw, when it came in 1986, was backpropagation: at last there was a way to train many layers. Enthusiasm returned, as did the money, which had been wintering somewhere warmer. It did not last. By the 1990s the ideas were working beautifully on toy problems and falling over on real ones, a pattern familiar to anyone who has sat through a product demonstration, and the second winter set in.

So what was missing in 1986? Not theory. Three ingredients:

- **Data.** The internet had not yet produced billions of photographs, sentences and labels to learn from.
- **Compute.** Nobody had yet noticed that graphics cards, built for video games, are superb at the one thing neural networks need: multiplying enormous tables of numbers.
- **Tricks.** A handful of unglamorous training techniques that stop deep networks from falling over had not yet been found.

The field had a recipe and no groceries. In 2012 all three turned up at once. Great ideas, wrong decade; and then, quite suddenly, the right one.

07

The bill

Now let us put on the other hat, and follow the money.

Here is a fact that surprises people. Once training is finished, a model is, quite literally, **a file of numbers**: the settings of all the dials, written down in order.

For Kimi K3 it is a very large file, but it is still a file. You could put it on a shelf. You could, for that matter, put it on a drive small enough to lose down the back of a sofa.*

So where does the value sit? In those numbers, and in the data and the compute that were burned to produce them. That gives us a distinction we shall lean on for the rest of the story:

- **Training cost** is paid once, up front, and is enormous. It is the cost of turning all the dials.
- **Inference cost** is paid every single time the model answers a question. It scales with usage, for ever.

Two kinds of bill

TRAINING: PAID ONCE

TRAINING	
Chips	_____
Electricity	_____
Data	_____
More chips	_____
People who can do this	_____
TOTAL	ENORMOUS

INFERENCE: PAID EVERY TIME ANYONE ASKS ANYTHING



Each one is small. They arrive every time the model answers a question, and they never stop arriving.

TIME, USAGE, AND THE REST OF THE COMPANY'S NATURAL LIFE

One bill arrives once and is framed. The other arrives every time anybody asks the model anything, and is not.

Different businesses make their money on different sides of that line, and a great many pitch decks are vague about which side they are standing on.

And now a question to take home. If the value of a model is a file, what happens when somebody gives that file away for nothing? Or copies it? Hold that thought. It becomes the entire plot in chapters nine and ten.

* At two bytes a dial it comes to about 5.6 terabytes. Sofas have swallowed larger things.

08

Three things to keep

- 01

A network is a function with dials. Data in, answer out, and a great many adjustable numbers in between.
- 02

Learning is turning the dials until the error shrinks. Measure the mistake, feel the slope, step downhill, repeat.
- 03

The revolution came from data and hardware, not from a new idea. The ideas had been sitting on the shelf since 1986, waiting for the world to catch up.

Cast of characters

WHO	WHAT THEY DID	WHAT BECAME OF THEM
Frank Rosenblatt	Research psychologist at the Cornell Aeronautical Laboratory in Buffalo. Demonstrated the Perceptron in 1958.	Went on to teach at Cornell. Died in 1971, aged 43.
The Mark I Perceptron	The idea made cabinet: 400 light sensors, 512 hidden units, and weights held on motor-driven dials. First shown in public in June 1960.	Retired to the Smithsonian's National Museum of American History.
Marvin Minsky	Co-founder of MIT's artificial intelligence laboratory and co-author of Perceptrons (1969).	Died in 2016, aged 88.
Seymour Papert	Co-author of Perceptrons, later famous for the Logo language that taught children to program.	Died in 2016.
David Rumelhart, Geoffrey Hinton, Ronald Williams	Authors of the 1986 paper that made backpropagation famous.	Hinton shared the 2024 Nobel Prize in Physics, and returns in the next chapter.

END OF THE SAMPLE

This is where the sample ends. The story carries on.

You have read the introduction and the first chapter, roughly a thirteenth of the book. The remaining eleven chapters run from gaming graphics cards through power stations and video to the question of who actually makes money from all this.

PART I: INTELLIGENCE How AI works

- 02 Machines That See**
CNNs and the 2012 ImageNet moment, when gaming GPUs entered AI.
 - 03 Attention Is All You Need**
From RNNs to the Transformer, the architecture built for parallel hardware.
 - 04 Building a Language Model**
Pretraining, scaling laws and RLHF. What it really takes to train, and to serve.
-

PART II: COMPUTE & ENERGY The physical layer

- 05 The Economics of Silicon**
Why GPUs rule, why they cost so much, and who controls the supply chain.
 - 06 Power & Alternative Compute**
AI's electricity bill, and the challengers: analog, photonic and neuromorphic chips.
-

PART III: THE FRONTIER Generative and next-generation

- 07 Pictures in Motion**
GANs, diffusion and Stable Diffusion, and why video is the hardest problem yet.
 - 08 Beyond the Transformer**
KAN, Mamba, Mixture of Experts, reasoning models and agents. Does scaling still work?
-

PART IV: SOVEREIGN RACE US, China, Europe

- 09 The DeepSeek Moment**
How a Chinese model shook the market, and the distillation wars that followed.
 - 10 China Today**
GLM, Kimi and Seedance: efficiency, open weights and price as strategy.
 - 11 Europe: Player or Spectator?**
Mistral, Bielik, ASML and the AI Act. Where Europe leads, and where it lags.
 - 12 Mapping the Value Chain**
Chips, cloud, models, apps: where the margins sit, the risks, and what to watch.
-

The whole book is at neuron.bazyl.net

331 pages, PDF and EPUB, in English and in Polish, no DRM. Price: €7. Pay by card, Apple Pay or Google Pay; the download is ready the moment you have paid.

You are welcome to pass this sample on and share it, as long as it is unchanged.

END OF THE SAMPLE

Pictures in this sample

Photographs and reproductions are the work of others and keep their own licences. The page numbers on the copyright page refer to the full edition; for the pictures in this sample the author, source and licence are listed below. Changes: scaled and cropped; chapter covers and the title page also darkened, with text set over the picture. Drawings and charts: the author.

Title page: the silicon of a GeForce GTX 580. Fritzchens Fritz, Wikimedia Commons, https://commons.wikimedia.org/wiki/File:NVIDIA@40nm@Fermi@GF110@GeForce_GTX_580@UA10B338_1041A1_N2Y540.000_GF110-375-A1_DSC04152_%2827576804994%29.jpg. CC0 (<https://creativecommons.org/publicdomain/zero/1.0/>).

Introduction cover: data centres near Ashburn, Virginia. Theodore Christopher, Wikimedia Commons, https://commons.wikimedia.org/wiki/File:Data_centers_in_Ashburn.jpg. CC0 (<https://creativecommons.org/publicdomain/zero/1.0/>).

Chapter 1 cover: foggy path on the way up Mount Pilatus, Switzerland. Gdmolena, Wikimedia Commons, https://commons.wikimedia.org/wiki/File:Foggy_path_from_Alpnach_to_Pilatus_mountain.jpg. CC BY-SA 4.0 (<https://creativecommons.org/licenses/by-sa/4.0/>).

GeForce GTX 580. TheStriker, Wikimedia Commons, <https://commons.wikimedia.org/wiki/File:EVGA GeForce GTX 580.jpg>. CC BY-SA 4.0 (<https://creativecommons.org/licenses/by-sa/4.0/>).

Google data centre, Council Bluffs, Iowa. Chad Davis, Wikimedia Commons, [https://commons.wikimedia.org/wiki/File:Google_Data_Center,_Council_Bluffs_Iowa_\(49062863796\).jpg](https://commons.wikimedia.org/wiki/File:Google_Data_Center,_Council_Bluffs_Iowa_(49062863796).jpg). CC BY 2.0 (<https://creativecommons.org/licenses/by/2.0/>).

IBM 704, 1957. NASA, Wikimedia Commons, https://commons.wikimedia.org/wiki/File:IBM_Electronic_Data_Processing_Machine_-_GPN-2000-001881.jpg. public domain.

Mark I Perceptron. National Museum of the U.S. Navy, Wikimedia Commons, [https://commons.wikimedia.org/wiki/File:330-PSA-80-60_\(USN_710739\)_\(20897323365\).jpg](https://commons.wikimedia.org/wiki/File:330-PSA-80-60_(USN_710739)_(20897323365).jpg). public domain.

Brain and perceptron diagrams. Frank Rosenblatt, 1958, Wikimedia Commons, https://commons.wikimedia.org/wiki/File:Organization_of_a_biological_brain_and_a_perceptron.png. public domain.

Hiker in fog, Alaska. Justin Morgan, Wikimedia Commons, https://commons.wikimedia.org/wiki/File:Flat_Top_Mountain_Hike.jpg. CC BY-SA 2.0 (<https://creativecommons.org/licenses/by-sa/2.0/>).

David Rumelhart, 1991. Rolf Kickuth, Wikimedia Commons, <https://commons.wikimedia.org/wiki/File:DavidRumelhart-IJCNNseattle1991-07-08.jpg>. CC BY-SA 4.0 (<https://creativecommons.org/licenses/by-sa/4.0/>).

Marvin Minsky, 2012. Joi Ito, Wikimedia Commons, [https://commons.wikimedia.org/wiki/File:Marvin_Minsky_\(1\).jpg](https://commons.wikimedia.org/wiki/File:Marvin_Minsky_(1).jpg). CC BY 2.0 (<https://creativecommons.org/licenses/by/2.0/>).

Seymour Papert, 1987. Shen-montpellier, Wikimedia Commons, <https://commons.wikimedia.org/wiki/File:Papert-moscow-1987.jpg>. CC BY-SA 4.0 (<https://creativecommons.org/licenses/by-sa/4.0/>).
