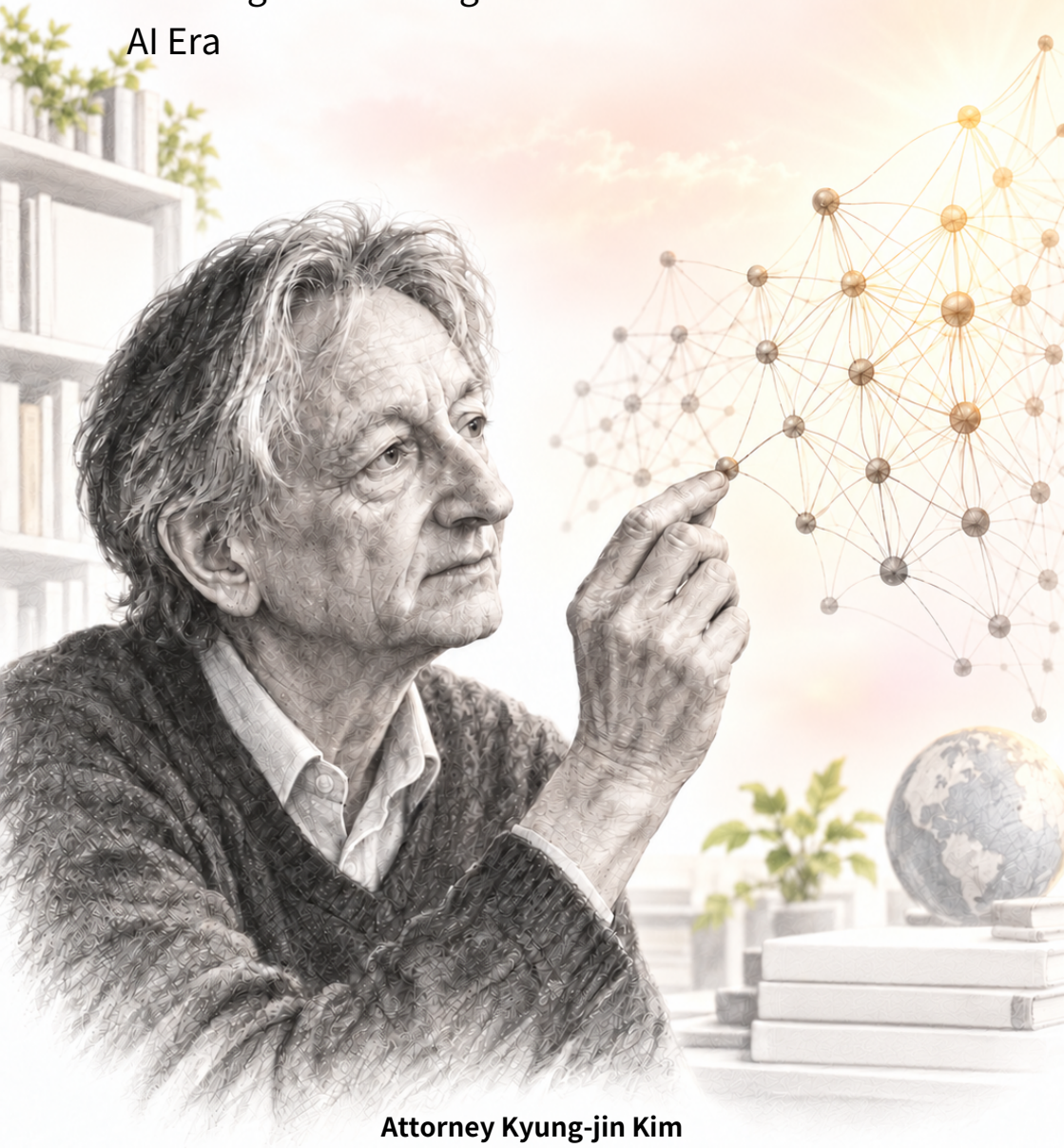
Geoffrey Hinton

The Origins of Intelligence and Reflections on the AI Era



Attorney Kyung-jin Kim

The Man Who Envisioned Learning Machines

Geoffrey Hinton is a researcher who found clues to machine learning in the way the human brain learns. Instead of humans entering rules one by one, he envisioned machines that learn on their own from data. Even during the period when neural network research received little attention in academia, he never let go of that question.

Starting from his childhood curiosity, this book traces the research he conducted with colleagues and students. From the 1986 backpropagation research and the 2012 success of AlexNet to the 2018 Turing Award and the 2024 Nobel Prize in Physics, it examines how neural networks made today's artificial intelligence possible. Unfamiliar concepts are explained alongside the problems the researcher sought to solve.

Hinton's life has not been solely about achievements. Leaving Google in 2023, he began warning more actively about the dangers of artificial intelligence. He started asking what kind of

impact the technology he advanced would have on society. The question of understanding learning and the question of technology's responsibility continue throughout his life. I hope this book serves as an opportunity for young readers to understand the principles and limitations of artificial intelligence and to reflect on its future for themselves.

Attorney Kyung-jin Kim

Table of Contents

The Man Who Envisioned Learning Machines

Part 1. Questions Toward the Brain and Learning

Chapter 1. Brain Research Sparked by Curiosity

Chapter 2. How Memory Is Connected

Chapter 3. Neural Networks Learning from Wrong Answers

Part 2. How to Make Neural Networks Learn

Chapter 4. Principles of Learning Found in Physics

Chapter 5. How to Learn Faster

Chapter 6. Deep Neural Networks Regain the Spotlight

Part 3. Deep Learning Emerges into the World

Chapter 7. Computers That Recognize Speech and Images

Chapter 8. Reducing Overfitting and Transferring Knowledge

Chapter 9. Advances in Deep Learning Built Together

Part 4. In Search of Better Learning

Chapter 10. Machines That Recognize Parts and Wholes

Chapter 11. Learning Methods Rediscovered in the Brain

Chapter 12. How to Learn Without an Answer Key

Part 5. Achievement and Responsibility

Chapter 13. The Nobel Prize in Physics and Neural Networks

Chapter 14. Warning of the Dangers of Artificial Intelligence

Chapter 15. Questions Surrounding Intelligence and Experience

References

Part 1. Questions Toward the Brain and Learning

Chapter 1. Brain Research Sparked by Curiosity



Figure 1. Depicts childhood observations and scientific curiosity.

1.1. Logic and Probability Left Behind by the Boole Family

Geoffrey Everest Hinton was born on December 6, 1947, in Wimbledon, London, England. His family was an intellectual lineage that produced mathematicians, scientists, physicians, and authors across multiple generations. A widely known figure among his ancestors was his great-great-grandfather, George Boole. Boole was a 19th-century British mathematician who created Boolean algebra, a method of calculating by treating truth and falsehood like the numbers 1 and 0. Today, the logic circuits of all computers are designed on the basis of this Boolean algebra. Boole's wife, Mary Everest Boole, was also a mathematician and educator. Mary's uncle, Sir George Everest, served as Surveyor General of India, and Mount Everest, the world's tallest mountain, took its name from him. Hinton's middle name, Everest, came from this family background.

The Hinton family on his father's side was also a family of scholars. His great-grandfather Charles Howard Hinton was a mathematician renowned for his work visualizing

4-dimensional space, and he married George Boole's daughter, Mary Ellen. Charles's father and another of Hinton's great-great-grandfathers, James Hinton, was a surgeon and author. His father's cousin, Joan Hinton, was one of the few female physicists who participated in the Manhattan Project, which developed the atomic bomb in Los Alamos, United States, during World War II. After the war, Joan Hinton opposed the use of atomic bombs, left physics, and moved to China.

His father, Howard Everest Hinton (H. E. Hinton), was an entomology professor at the University of Bristol and a Fellow of the Royal Society. He studied various insects and took a deep interest in Elmidae, a family of small aquatic beetles. In an interview related to the Nobel Prize, Hinton once recalled the appearance of his father's laboratory. High shelves covering one wall were packed with hundreds of boxes containing entomology papers. Among them, only a single small box placed in a low spot next to the door was labeled "Not Insects," and inside it were letters from the children and family letters. It was a household where scholarship was the center of life. Hinton recounted that from a very young age, he knew that if he

did not succeed as an academic, he would be a failure, and he felt immense pressure regarding academic achievement.

Looking at this family history alongside the history of artificial intelligence reveals one striking point. The book published by George Boole in 1854 is titled An Investigation of the Laws of Thought on Which Are Founded the Mathematical Theories of Logic and Probabilities. Both logic and probability appear side by side in the title. Yet the symbolic AI researchers who led artificial intelligence research after the mid-20th century inherited only the logic side of Boole's legacy. They sought to explain human thought through symbols divided into truth and falsehood and rules for manipulating those symbols. However, the real world is full of fuzzy and uncertain phenomena, and dealing with such a world requires probability.

Hinton, Boole's great-great-grandson, parted ways with the mainstream at that very point. He did not view the brain as a precision machine operating according to fixed logical rules. Instead, he saw it as a network that learns by gradually altering how strongly countless nerve cells (neurons) are connected to

one another—that is, their connection strengths—based on experience. If the great-great-grandfather's logic became the backbone of symbolic artificial intelligence, the great-great-grandson started from probability, the other half of the title of his ancestor's book, opening the door to modern deep learning.

1.2. An Epiphany at Age 7 and Wandering School Days Searching for a Path

One anecdote Hinton frequently shares when looking back on his childhood is an experience on a bus. He placed an old British penny coin on a rough, bristly bus seat. Whenever the bus jolted, instead of sliding down, the coin moved upward along the pile of the seat fabric. The young Hinton felt this was impossible and desperately wanted to know why. The attitude of not letting an incomprehensible phenomenon pass by was ingrained in him from this time.

At the age of 7, Hinton entered Clifton College in Bristol. It was a school with a strong Christian character. The school taught religious doctrines as self-evident truths, and most of his classmates accepted those words without question. Hinton did not. He could not accept being told to believe without evidence. In a later interview with the Seoul Economic Daily, he recalled that it was during this time that he first realized anyone could be wrong. He explained that this experience led to the belief that other researchers could also be mistaken, forming the foundation for continuing his research in defiance of mainstream academia. The strength that enabled Hinton to endure nearly half a century of cold treatment from the academic community began with this experience.

His interest in the brain emerged during high school. When he was around 16 or 17 years old, his friend Inman Harvey—whom Hinton always considered smarter than himself—remarked one day that memories in the brain might be stored spread out across the entire brain, like a hologram. With a hologram, even if only a single piece remains, the entire image can be reconstructed, albeit faintly. The idea that memories are not

kept one by one in a single place, but divided and stored overlapping across multiple locations, came as a major shock to Hinton. The desire to uncover how the brain computes took root at this time.

However, finding a discipline that could answer that desire was not easy. In 1967, Hinton entered King's College, Cambridge, and the next few years were, in his own words, a "crazy random walk." At first, he chose the natural sciences, studying physics, chemistry, and X-ray crystallography. Exhausted by a demanding schedule of up to 12 hours a day, he left the university after only 1 month. Earning rent money through odd jobs in London, he read Dostoevsky and also considered becoming an architect. Yet after working at an architecture firm for just 1 day and discovering that the actual work was closer to picking cheap flooring and doorknobs to fit a budget rather than design, he quit immediately.

After returning to university, he moved through departments that seemed likely to deal with the brain and mind one after another. First, he chose physiology, hoping to learn about the

brain. The classes taught how electrical signals propagate along the thick nerve fibers of squids, but did not address how neurons represent information or how they learn. Next, he switched to philosophy in search of the essence of the mind. Exhausted by debates carried out solely through words without experiments, he left after 1 year, saying he had developed only an "antibody to philosophy." Finally, he studied experimental psychology and received his bachelor's degree in 1970.

Psychology could not provide the answer he was seeking either. Psychology at the time was heavily skewed toward observing the behavior of rats in boxes. In an experiment investigating children's cognitive development, Hinton had subjects pick the odd one out among a yellow triangle, a yellow circle, and a red circle. One 5-year-old child pointed to the red circle and said that the researcher had painted it the wrong color. The child's thought process was far more complex than the experimental design. Hinton felt that psychology of that era could not explain human cognition.

His wandering continued even after graduation. For about 1 year, he apprenticed as a carpenter. Seeing the craftsmanship of a skilled carpenter on-site, he concluded he could never match that level on this path and returned to academia. Later, in Bristol, he participated as an assistant in a study on child language development. It was research testing Noam Chomsky's theory of language innateness—namely, the theory that humans possess grammatical competence from birth. The research team placed microphones in the homes of children aged 3 to 5 and gathered as many as 166,000 speech samples. However, when visiting a low-income household on the outskirts of Bristol for a survey, to the first question asking what she did when her child spoke, a mother answered that she hit the child. Following this incident, Hinton withdrew from the research.

After wandering through various disciplines, he arrived at a single conclusion: to unravel the secrets of intelligence, one had to build computational models of the brain directly on a computer.

1.3. Isolation in Edinburgh and Conflict with Advisor Longuet-Higgins

In 1972, Hinton entered the doctoral program at the University of Edinburgh to pursue artificial intelligence research in earnest. Edinburgh was virtually the only university in Britain at the time equipped with a proper foundation for AI research. What awaited him there, however, was isolation and conflict with his advisor.

His advisor was Christopher Longuet-Higgins. A brilliant scientist who had become a professor of theoretical chemistry at the University of Cambridge in his 30s, he pivoted to cognitive science in his 40s, researching associative memory and models that store memories like holograms. This was the exact topic that fascinated Hinton. By the time Hinton arrived, however, Longuet-Higgins's view had already changed. He had been deeply impressed by Terry Winograd's SHRDLU research. SHRDLU was a program that interpreted English commands given by humans in a simulated blocks world according to symbolic rules and moved the blocks, and it was regarded as a

representative achievement of symbolic AI. Longuet-Higgins abandoned any lingering attachment to neural networks and turned toward symbolism.

The academic climate was also hostile to neural networks. In 1969, Marvin Minsky and Seymour Papert mathematically demonstrated in their book Perceptrons that single-layer neural networks could not solve even seemingly simple problems like XOR. Afterward, many researchers regarded the entire field of neural networks as hopeless. Longuet-Higgins, who favored neat logical systems, could not accept neural networks operating in the fuzzy manner of probability and adjusting connection strengths. Calling Hinton's research nonsense, he repeatedly demanded that he switch to symbolism.

Hinton did not back down. He promised that if given just 6 more months, he would demonstrate results, and if not, he would change his topic. Once 6 months passed, he made the same promise again, and this repeated several times. In a 2024 oral history interview with the University of Edinburgh, Hinton revealed the reason why he had not been expelled. His father

was a Fellow of the Royal Society, and Longuet-Higgins, being a fellow member, would have found it awkward to throw out a colleague's son. Hinton wryly acknowledged that this was thanks to the privilege he enjoyed.

The doctoral research Hinton initially envisioned was named LUDWIG. Standing for "Language Understanding Device Without Innate Grammar," it borrowed the name of the philosopher Ludwig Wittgenstein while expressing opposition to Chomsky's innateness theory. His plan was to create a neural network that learned language solely from experience without preprogrammed symbolic rules. At the time, however, there was no method to train the hidden layers tucked between the input and output layers. LUDWIG could not be built with the technology of that era.

The person who showed him a way forward at this time was postdoctoral researcher Steve Isard. Serving virtually as a co-advisor, he advised Hinton to test the potential of neural networks starting with small, tractable problems like handwriting recognition. Hinton studied Frank Rosenblatt's

1962 book Principles of Neurodynamics on his own and implemented parallel networks. When Longuet-Higgins moved to the University of Sussex, Hinton followed along with Isard and Mark Steedman. After working as a researcher at the University of Sussex, Hinton received his doctorate from the University of Edinburgh in 1978 with the dissertation "Relaxation and Its Role in Vision."

1.4. The Limitations of Symbolism and the Quiet Beginning of Connectionism's Counterattack

The conflict between Hinton and Longuet-Higgins was not merely an issue between two individuals. In the artificial intelligence community of the late 20th century, two perspectives on intelligence stood in opposition. One was Symbolic AI. It was led by Allen Newell, Herbert Simon, John McCarthy, and Marvin Minsky. They believed that human thought consists of symbols, logical rules for manipulating those symbols, and step-by-step sequential reasoning. The premise was that if humans systematically organized rules and programmed them into computers, intelligence could be created. This approach was later dubbed "Good Old-Fashioned AI (GOFAI)."

The other was Connectionism. Hinton, David Rumelhart, Terrence Sejnowski, and later Yann LeCun and Yoshua Bengio paved this path. Connectionists viewed the essence of intelligence not as hardcoding rules, but as the ability to learn

representations from data autonomously. Like the brain, numerous units operate simultaneously, building up knowledge by adjusting the connection strengths (weights) among those units. In this view, a single concept is not stored in an individual slot, but distributed across activation patterns of multiple units. This is known as Distributed Representation. It was an idea closely resembling the holographic memory concept Hinton had heard about in high school.

Entering the 1970s, symbolism hit a wall. Real-world data was too ambiguous and disparate to be neatly organized into rules. In 1973, British applied mathematician Sir James Lighthill published a report evaluating artificial intelligence research. This Lighthill Report pointed out that AI research could not solve the problem of combinatorial explosion. Combinatorial explosion refers to the phenomenon where the number of possible cases to examine grows exponentially with the size of the problem, rendering computation impossible. Following the report, AI research funding in Britain was drastically cut, ushering in the first "AI Winter."

While many AI researchers criticized the Lighthill Report, Hinton's assessment was different. He believed Lighthill's critique hit the core of the issue. Symbolists like McCarthy claimed that anything that could be computed with symbols could be made into artificial intelligence. However, this claim failed to take into account the time and resources required for computation—namely, computational complexity. Lighthill, who had been a schoolmate of Longuet-Higgins and physicist Freeman Dyson at Winchester College, accurately identified this weakness as an applied mathematician studying fluid dynamics. Hinton shared the same view. To handle the complexity of reality with limited computational resources, manipulating symbols one by one was insufficient; a method was needed where multiple units worked simultaneously, gradually refining connection strengths, just like the brain.

Amid academia's indifference, Hinton searched for kindred spirits. In 1979, alongside James Anderson, he organized a workshop titled "Parallel Models of Associative Memory" in La Jolla, San Diego, United States. Connectionist researchers from various institutions gathered at this event, including David

Rumelhart of the University of California, San Diego (UCSD). This gathering served as a stepping stone for the later formation of the Parallel Distributed Processing (PDP) research group. Though not visibly prominent, connectionism's counterattack had quietly begun from this point.

1.5. Curiosity Over Short-Term Results: The Fruits of Basic Research and Their Significance

From receiving his doctorate in 1978 until AlexNet, created with his students in 2012, overwhelmingly won an image recognition competition, neural networks remained a fringe field that struggled to secure research funding for a long time. During this period, the field of artificial intelligence endured two winters. The force that sustained Hinton through this lengthy period was neither rapid commercial success nor fame. It was the curiosity to understand how the human brain works. When asked to introduce himself in an interview with the Nobel Prize organization, he replied that although he did not quite know which field he belonged to, he was someone who wanted to understand how the brain works.

Hinton repeatedly emphasized publicly that such curiosity must be the starting point of research. On October 8, 2024, he was announced as a recipient of the Nobel Prize in Physics alongside John Hopfield. At a press conference at the University of

Toronto that day, he urged support for curiosity-driven basic research. Noting that while Canada does not have as much research funding as the United States, it spends money wisely, he explained that the advancement of neural networks came not from pouring money into applied problems, but from scientists striving to understand something out of curiosity. In his interview with the Seoul Economic Daily, he likewise stated that rather than pressuring academics to produce immediate practical results, allowing researchers the freedom to investigate what intrigues them is far more effective, and that deep learning was created precisely in that manner.

Behind the fruition of such research lay the long-term support of the Canadian Institute for Advanced Research (CIFAR). In 1987, with support from CIFAR, Hinton relocated to Canada and took up a position at the University of Toronto. He pointed out that the American research funding system forced professors to spend extensive time writing proposals. In Canada, by contrast, a 6-page proposal could secure stable funding for 3 to 5 years without being tied to predefined topics. Thanks to such an

environment, Hinton was able to persistently delve into an unpopular field rather than chasing trends.

The significance of the path Hinton walked goes beyond technological success. It also transformed the perspective on the human mind. Symbolism viewed humans as "Reasoning Machines" that manipulate symbols elaborately and reason logically. Hinton's work showed that humans are closer to "Analogy Machines" that understand the world by clustering it into patterns based on sensory experiences and similarities. In Hinton's view, logical reasoning is a capability layered on top of neural networks, and the foundation of intelligence lies in adjusting connection strengths through which countless neurons acquire distributed representations on their own from experience.

The thought harbored by a 7-year-old boy—the realization that adults could also be wrong—grew half a century later into the conviction that the entire academic establishment could be mistaken. Starting out from the shadow cast by the logic of his great-great-grandfather George Boole, Hinton held fast to

probability, written alongside logic in that book's title, and walked the path to the very end.

Chapter 2. How Memory Is Connected

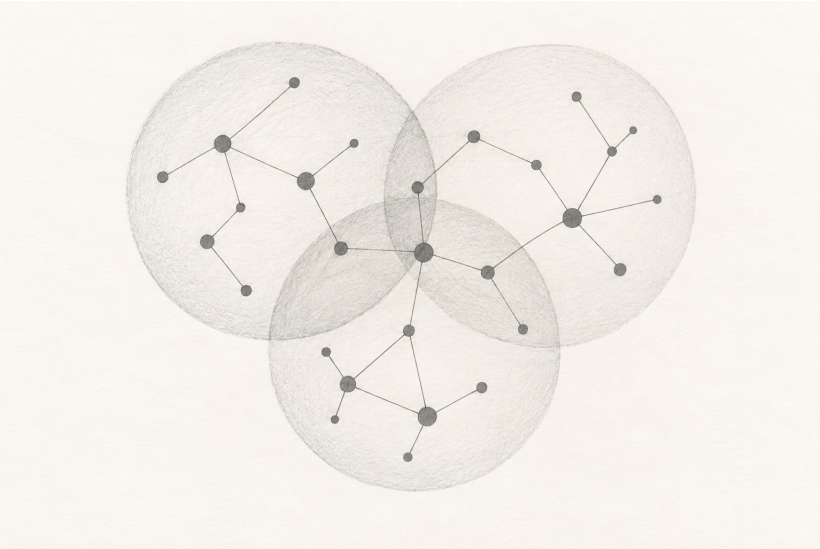


Figure 2. An illustration of the idea of representing information distributed across multiple connections.

2.1. The Limits of the 'Grandmother Neuron' and the Orchestra of Distributed Representations

During high school, Geoffrey Hinton heard about holograms from a friend. Even if part of a holographic film is cut away, the entire image remains, albeit faintly. This is because information is not recorded at a single point on the film, but is distributed across the entire film. Hinton thought that memories in the brain might also be distributed in a similar manner, and he made this idea his lifelong research pursuit.

However, symbolism, which was the mainstream in cognitive science and artificial intelligence during the 1970s, viewed the storage of knowledge completely differently. Symbolists believed that human thought consists of symbols and rules for manipulating those symbols, and that a single concept is stored in a single location. This method of storage is called a 'localist representation.'

A phrase illustrating localist representation is the 'grandmother neuron hypothesis.' This is the hypothesis that somewhere in the brain there is a single neuron dedicated solely to one's grandmother, and that when that neuron fires, one recognizes grandmother. According to this hypothesis, there must be a cat neuron for cats, and a dedicated symbolic node for each specific sentence. Mapping a single entity to a single unit makes distinction easy, but makes it difficult to represent commonalities across diverse entities. However, this does not mean that all theories explaining thought through symbols map one concept to a single neuron.

Hinton questioned this idea from early on. Localist representation had three weaknesses.

First is the problem of combinations. Entities in the world appear in endlessly new combinations. If a new dedicated node must be created every time a combination increases—such as 'red apple,' 'green apple,' and 'half-eaten red apple'—the number of required nodes grows exponentially. This is known as a 'combinatorial explosion.'

Second is vulnerability to damage. If the single neuron dedicated to grandmother dies, all knowledge about grandmother should disappear. In distributed representations, even if some computational units are damaged, information can still be preserved in the remaining units.

Third is the difficulty in capturing similarity. Cats and tigers are similar, whereas cats and refrigerators are not. However, if there is a separate, isolated node for each concept, the nodes themselves cannot indicate how similar two concepts are. Information about similarity must be manually written in through separate rules.

The alternative Hinton proposed was 'distributed representation.' Distributed representation is an approach that, rather than assigning a single concept to a single neuron, represents it through an activation pattern jointly produced by numerous neurons. Some neurons fire strongly, some fire weakly, and some remain off. That overall pattern is itself a single concept. In mathematical terms, a concept becomes a vector composed of multiple numbers.

Distributed representation can be compared to an orchestra. A measure of a symphony is not a sound produced by a single violin alone. Violins, cellos, flutes, and trumpets resonate together at varying intensities to create a single chord. When the measure changes, the combination and intensity of instruments change together, and the same instrument participates unceasingly across multiple parts of the piece. Hinton's idea was that knowledge in the brain likewise exists as a chord sounded together by numerous neurons.

Just as an orchestra's instruments participate in multiple measures, in distributed representations a single neuron is not responsible for only one concept. Neurons represent small features such as 'has fur,' 'walks on four legs,' and 'lives indoors,' simultaneously participating in the representation of multiple concepts. These small features are called 'microfeatures.' Because cats and tigers share many microfeatures, the activation patterns of the two concepts become similar to each other. Similarity naturally enters into the representation without separate rules. It is also resilient to damage. Even if a few neurons are damaged, most of the

pattern produced by the remaining neurons remains intact, so knowledge does not vanish all at once.

Hinton also demonstrated this idea experimentally: the 'family tree task' of 1986. Hinton constructed family trees for two families comprising a total of 24 individuals—12 members of an English family and 12 members of an Italian family. The two family trees were identical in structure. The English family included names such as Christopher, Penelope, Charlotte, Victoria, James, and Colin, while the Italian family had names like Roberto and Maria in corresponding positions. There were 12 types of relationships, including father, mother, husband, wife, uncle, and nephew.

The neural network was fed 'Person 1' and 'Relationship' as inputs and was tasked with providing 'Person 2' as the answer. For example, if 'Charlotte' and 'mother' were input, it had to output 'Victoria.' Names were input by activating only one slot out of 24 slots, and relationships by activating only one slot out of 12 slots. Therefore, the inputs themselves contained no information whatsoever about how similar the people were to

one another. The two inputs passed through layers of 6 neurons each, met in a central hidden layer, and then produced an answer through several more layers.

The neural network received not a single rule such as "If B is the mother of A and C is the husband of B, then C is the father of A." All that was provided were 104 sentences stating facts. Hinton trained the neural network on 100 of them and set aside the remaining 4 for testing.

After training was completed, Hinton examined the 6 neurons representing people. Some neurons were distinguishing between the English family and the Italian family, some were indicating which generation an individual belonged to, and some were indicating which branch of the family tree they belonged to. The neural network had discovered on its own features that no one had taught it. The neural network also produced correct answers for most of the test questions it had never seen before.

Hinton saw that the true value of the backpropagation learning algorithm lay precisely here. Rather than the computational technique itself, what mattered was that the hidden layers could independently learn distributed representations reflecting the underlying structure of the data. The roots of 'word embedding' in modern artificial intelligence, which converts words into numerical vectors, lie in this experiment.

2.2. Researchers Gathering in La Jolla: The 1979 Workshop and the Birth of the PDP Group

In the late 1970s, neural network researchers were scattered like isolated islands. Most research grants and faculty positions went to symbolic research. People studying neural networks were largely unaware of one another's existence, and their papers were scattered across obscure journals.

Hinton's situation was no different. He received his doctorate from the University of Edinburgh in the United Kingdom in 1978, but securing research funding for neural network research in

the UK was difficult. Hinton had conducted research at the University of Sussex before earning his doctorate, and in October 1978 moved to the University of California, San Diego (UCSD) in the United States.

In 1979, together with psychologist James Anderson of Brown University, Hinton organized a workshop on 'Parallel Models of Associative Memory' in La Jolla, home to UCSD. This workshop served as the occasion where scattered neural network researchers gathered in one place for the first time.

Among the attendees was Terry Sejnowski. At the time, he was a postdoctoral researcher at Harvard Medical School and a young researcher who had published a few neural network papers in lesser-known journals. Hinton is said to have invited him after asking David Marr, a vision theorist at MIT, about Sejnowski's work. The two men, who met at this workshop, would later become lifelong colleagues, collaborating to create the Boltzmann machine.

The workshop's proceedings were compiled and published in 1981 in a volume with the same title, Parallel Models of Associative Memory. The editors were Hinton and Anderson. Hinton contributed a chapter to this book addressing how to implement semantic networks on parallel hardware. Its core thesis was that semantic relationships, which symbolism represented using graphs and symbols, could instead be represented through activation patterns of neurons.

In La Jolla, there were researchers poised to carry this momentum forward: David Rumelhart and James McClelland of UCSD. In the early 1980s, the two formed the 'Parallel Distributed Processing' research group, known for short as the PDP Group. Hinton grew close to Rumelhart while working as a postdoctoral researcher at UCSD around this time, and continued to collaborate with the PDP Group even after moving to Cambridge in the United Kingdom and Carnegie Mellon University in the United States.

The support and involvement of neighboring researchers were also crucial to the growth of the PDP Group. Donald Norman,

who led cognitive science at UCSD, provided an umbrella within the department so researchers could freely explore ideas that diverged from the mainstream. Francis Crick, who was at the nearby Salk Institute, also attended meetings frequently. Crick was the biologist who had unraveled the double-helix structure of DNA, and by this time had shifted his focus to brain research. He debated with researchers over how closely neural network models resembled the actual brain, and later co-authored a chapter on the structure of the cerebral cortex in Volume 2 of the PDP book.

The research of the PDP Group was consolidated in 1986 into a two-volume work published by MIT Press, Parallel Distributed Processing: Explorations in the Microstructure of Cognition. Volume 1 covered the foundations, while Volume 2 addressed psychological and biological applications. This work exerted a profound influence across psychology, neuroscience, and computer science, and is said to have sold more than 50,000 copies, a rare feat for an academic publication.

Hinton was named as an author on several key chapters in Volume 1. Chapter 3, 'Distributed Representations,' was co-authored by Hinton, McClelland, and Rumelhart. Chapter 7, 'Learning and Relearning in Boltzmann Machines,' was written by Hinton and Sejnowski. Chapter 8, 'Learning Internal Representations by Error Propagation,' was written by Rumelhart, Hinton, and Ronald Williams. In that same year, the three published a paper introducing the backpropagation learning algorithm in Nature. The small gathering that began in La Jolla had become the epicenter revitalizing neural network research.

2.3. Parallel Associative Memory and the Principles of Weight Optimization

'Associative memory,' which appeared in the title of the La Jolla workshop, was another crucial concept championed by connectionism alongside distributed representations. This concept is easy to understand when compared with the way ordinary computers retrieve memories. A computer retrieves memory by address. It can only retrieve content if it knows the exact address of which slot holds what, and if even a single part of the address is incorrect, it cannot find the desired content. This is called 'address-addressable memory.'

Human memory operates differently. Hearing just the first few notes of a song's opening bar brings the entire song to mind, and seeing only half of a friend's face is enough to know who it is. One retrieves the whole from merely a part of the content. This is called 'content-addressable memory.' Connectionist researchers believed that neural networks could accomplish this kind of memory naturally.

The person who clearly demonstrated this principle mathematically was physicist John Hopfield. In 1982, Hopfield imported the concept of 'energy,' used in physics to describe materials such as magnets, into neural networks. In a Hopfield network, an energy value is determined for each state of the neurons. Stored memories correspond to recessed valleys in the energy landscape. When an input that is partially erased or corrupted by noise is received, the neurons influence one another, gradually lowering the energy. Just as a ball rolls down a slope and settles at the bottom of a valley, the network arrives at the nearest valley and restores the original memory. This process, in which numerous neurons operate simultaneously to seek a stable state, is called 'parallel relaxation.'

The Hopfield network had weaknesses. Unwanted small valleys could form in the energy landscape, leading to spurious memories, or the network could become trapped in a shallow valley and fail to find a better solution. To solve this problem, Hinton and Sejnowski incorporated probabilistic concepts from statistical physics. If neurons do not move solely according to fixed rules but fluctuate with a certain probability, the network

can escape shallow valleys and find deeper ones. The model created this way is the 'Boltzmann machine.' In 1985, David Ackley, Hinton, and Sejnowski published a paper detailing the learning algorithm of the Boltzmann machine.

The essence of the problem that the Boltzmann machine solves is 'parallel constraint satisfaction.' When a human sees an object, the brain simultaneously satisfies numerous conditions—such as boundaries, depth, direction of light, and color—to find the most plausible interpretation. These conditions are embedded in the connection strengths between neurons, that is, the 'weights.' If it is natural for two neurons to fire together, the weight between them is large; if firing together is unnatural, the weight is small or negative.

That leaves the question of who determines these weights. The connectionist answer is that the neural network determines them on its own by examining data. Humans do not hard-code rules. By looking at many examples, the neural network gradually adjusts its weights so that the patterns it generates

become similar to the patterns of the real world. This is weight optimization, and this process is none other than learning.

Hinton was also interested in how memories stored in weights change over time. In 1987, together with David Plaut, he published research on 'fast weights.' They introduced two types of weights to a single connection. Slow weights change slowly over long periods and store general knowledge about the world. Fast weights change quickly and fade quickly when new information arrives, serving to temporarily hold onto what was just seen.

This architecture was used to address a practical problem encountered in neural network learning. When a neural network learns something new, what it previously learned often becomes blurred. In their experiments, fast weights helped rapidly revive old memories blurred by new learning with only brief practice. The paper's title, 'Using Fast Weights to Deblur Old Memories,' reflects this directly. The idea of fast weights was subsequently forgotten for a long time, before regaining attention in 2016 when Hinton and his colleagues revived it.

2.4. A Shift in Perspective on the Mind: From Symbol Manipulator to Data Learner

Research on distributed representations and associative memory was not merely a technical issue. It offered a different answer to the long-standing question of how to view the human mind.

The human portrayed by symbolism and classical cognitive science was a 'symbol manipulator.' Under this view, a human is a logic machine that sequentially manipulates symbols in the mind according to predetermined rules. Observing how humans speak grammatically and categorize objects, they inferred that grammar rules and category inventories must already be inscribed in the brain.

The human envisioned by Hinton and connectionist researchers was different. The brain they saw was not a machine that transcribes rules and computes sequentially, but a 'data learner' that autonomously extracts similarities and patterns

from countless experiences. Hinton has often described humans as being closer to 'analogy machines' than to logic machines. When encountering a new situation, rather than applying rules one by one, people make judgments by recalling similar situations experienced in the past.

From this perspective, grammatical speech or logical reasoning does not occur because rules are written inside the brain. Rather, as distributed representations formed within the weights among numerous neurons interact with one another, behavior emerges that appears rule-governed on the surface. This is called 'emergence.' Just as the neural network in the family tree experiment inferred kinship relationships without ever receiving kinship rules, humans too can behave in accordance with rules without memorizing rules.

In recent lectures, Hinton has repeatedly explained that the research from that era is connected to today's artificial intelligence. In a lecture at the University of Bristol in the United Kingdom in June 2025, he introduced a small language model he created in 1985. This model took errors arising from

predicting the next word as a learning signal, converted word symbols into feature activations, and learned to predict the features of the next word through the interaction of those features. He said that today's large language models can be viewed as descendants of this small model. That small model he referred to was precisely the neural network that learned the family trees.

Research on associative memory was also reevaluated decades later. The 2024 Nobel Prize in Physics was awarded to Hopfield and Hinton. The Royal Swedish Academy of Sciences cited in its award motivation that Hopfield created an associative memory capable of storing and reconstructing data patterns such as images, and that Hinton built upon the Hopfield network to create a new network called the Boltzmann machine. 'Associative memory,' which was in the title of the 1979 La Jolla workshop, reappeared 45 years later in the citation for the Nobel Prize in Physics.

During the 1970s and 1980s, Hinton traveled between La Jolla, Cambridge, and Pittsburgh, solidifying the theories of

distributed representation and associative memory. At the time, his ideas were on the periphery of academia: the idea that knowledge is not a symbol written in a single location but a chord produced jointly by numerous neurons; the idea that memory is retrieved by content rather than by address; and the idea that rules are learned from data rather than hard-coded into it. These three ideas became the foundation for artificial intelligence that processes language and concepts today.

Chapter 3. Neural Networks Learning from Wrong Answers

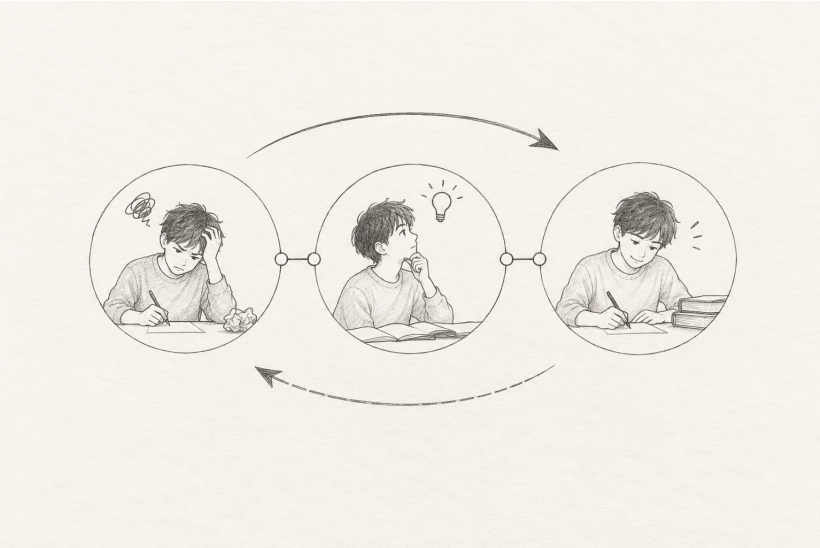


Figure 3. An analogy for the process of tracing back incorrect results to improve learning.

3.1. The Challenge of Multilayer Neural Networks and the Rediscovery of Backpropagation

In 1969, Marvin Minsky and Seymour Papert published the book Perceptrons. In this book, the two demonstrated mathematically that a single-layer perceptron, which has only an input layer and an output layer, cannot solve problems such as XOR. After the book was published, many researchers abandoned neural network research and moved to symbolic artificial intelligence. Symbolic AI is an approach in which humans manually write rules and computers perform reasoning according to those rules.

It was not that there was no way at all to overcome the limitations of the single-layer perceptron. It was known theoretically that inserting a hidden layer between the input layer and the output layer could solve more complex problems. The problem was how to train that hidden layer. The output layer can be directly compared with the correct answer. If a neural network looks at a photo of a cat and answers 'dog,' one

can immediately calculate how wrong it was. The hidden layer is a different story. Because neurons in the hidden layer are not directly connected to the outside world, no one tells them what values they should produce at any given moment to be correct. The problem of determining how much responsibility lies with which of the numerous intermediate neurons when the final answer is wrong is called the 'Credit Assignment Problem.' As long as this problem remained unsolved, multilayer neural networks could not help but remain concepts on paper.

The answer to the credit assignment problem is error backpropagation. The principle lies in the chain rule of calculus. First, the error is calculated at the output layer. Next, this error is sent backward—that is, layer by layer from the output layer toward the hidden layers—calculating how much each connection weight contributed to the overall error. Finally, the weights are adjusted little by little in the direction that reduces the error. This final step is called gradient descent. It is akin to walking down a mountain blindfolded, checking only the slope underfoot and taking steps toward lower ground one step at a time.

Backpropagation is not an invention conceived all at once by a single person. Multiple scholars arrived at the same idea independently of one another. In 1970, Seppo Linnainmaa of Finland formulated in his master's thesis the computational method known today as 'reverse-mode automatic differentiation.' In 1974, Paul Werbos, a doctoral student at Harvard University, proposed in his Ph.D. dissertation Beyond Regression that this method could be used for training neural networks. Around 1985, David Parker and Yann LeCun of France each proposed similar learning rules.

Even so, what is cited as the turning point in the history of backpropagation is the 1986 research by David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. The collaboration among the three began in San Diego. From 1978 to 1980, Hinton worked as a postdoctoral researcher at the University of California, San Diego (UCSD), and continuing to interact with colleagues afterward, he participated in the PDP (Parallel Distributed Processing) research group led by Rumelhart, James McClelland, and others. In a 2024 article, UC San Diego noted that the work of the PDP group served as the foundation

for the development of backpropagation, and that Hinton, together with Rumelhart and Williams, later popularized this algorithm through their 1986 Nature paper.

The three were not the first to propose backpropagation. Their contribution did not lie in being the first to write the mathematical formulas. It lay in demonstrating experimentally that the hidden layers of a neural network trained with backpropagation spontaneously create internal representations that humans never taught them.

Hinton himself made this clear. In a 2018 interview with Martin Ford, he stated that the basic idea of backpropagation was conceived by Rumelhart and was Rumelhart's invention. He explained that he and Williams contributed to refining the method, and that his own share lay in showing that backpropagation could learn meaningful distributed representations. Whenever the media introduced him as the inventor of backpropagation, Hinton corrected them in this way.

3.2. The 1986 Nature Paper and Computational Principles

On October 9, 1986, a four-page paper was published in volume 323, pages 533 to 536 of the international journal Nature. The title was 'Learning representations by back-propagating errors,' which translates in Korean to 'learning representations by back-propagating errors.' The authors' affiliations were the Institute for Cognitive Science at UCSD for Rumelhart and Williams, and Carnegie Mellon University (CMU) for Hinton. In the same year, the three published a long chapter titled 'Learning internal representations by error propagation' in Volume 1 of Parallel Distributed Processing, a two-volume book published by the PDP Group, explaining the same content in detail. The short paper attracted the attention of academia, while the lengthy book served as a textbook that researchers could follow directly.

The computational process of the paper can be divided into four stages. The first three stages are the process in which the neural network produces an answer and measures how far that

answer deviates from the correct answer, while the final stage is the process of dividing the measured error into the responsibility of each individual neuron.

The first is the forward pass. The signal ascends from lower layers to higher layers. A neuron in a higher layer is connected to multiple neurons in the lower layer, and each connection has a weight that represents its strength. Multiplying the output of each lower-layer neuron by the weight of the corresponding connection and summing all those values yields the total input received by the higher-layer neuron. The threshold that determines when a neuron fires was not treated separately. By introducing a virtual input that always outputs 1 and having the weight attached to that input serve as the threshold, it was computed in the same manner as other weights.

The second is the activation function. The activation function is the rule that converts the total input received by a neuron into that neuron's output. In this paper, the logistic sigmoid function was used. This function produces a value close to 0 when the total input is a large negative number, a value close to 1 when it

is a large positive number, and values between 0 and 1 in between, tracing a gentle S-shaped curve. Because the curve is smooth and continuous without breaks or sharp bends, it can be differentiated everywhere. Differentiation is the calculation of how much the output changes when the input changes slightly. The derivative of the sigmoid function is obtained simply by multiplying the output value by 1 minus the output value. The step function of early perceptrons, which had only two states—on and off—could not use such differentiation because the value abruptly jumped at a single point. Backpropagation propagates errors backward according to the chain rule, which is the rule of obtaining the overall rate of change across multiple stages as the product of rates of change at each stage. For this method to work, such a smooth function was necessary.

The third is error measurement. One squares the differences between the neural network's output y(j) and the correct answer d(j), sums them all, and halves the result. This value E is the score that indicates how wrong the neural network was.

The fourth is the backward pass. Here, an error signal is calculated for each neuron. The error signal is a value representing how much the overall error changes when the total input received by that neuron changes slightly, corresponding to the extent of that neuron's responsibility for the error. The error signal of an output layer neuron is calculated by multiplying the difference between the actual output and the target value given as the correct answer by the derivative of the sigmoid function—that is, the product of the output value and 1 minus the output value. The error signal of a hidden layer neuron is obtained by multiplying the error signals of the higher-layer neurons that received this neuron's output by each connection weight, summing them all up, and multiplying this by the neuron's own output value and 1 minus that output value. This is a calculation that distributes the responsibility of the higher layer to the lower layer in proportion to the strength of the connections. In this way, no matter how many hidden layers there are, one can calculate the responsibility of all neurons by starting from the output layer and descending layer by layer. An error signal is thus provided even to hidden layers that could not be directly compared with

the correct answer. Finally, each weight is adjusted in the direction that reduces the error, and the magnitude of change at each step is determined by a small number called the learning rate.

The researchers also addressed problems that arise in practical learning. One was learning speed. Gradient descent is a method of adjusting weights little by little in the direction where error decreases. Comparing the variation of error with respect to weights to a terrain, when this terrain takes the shape of a long, narrow ravine, gradient descent oscillates back and forth between the two walls of the ravine, slowing its progress along the ravine floor. To alleviate this problem, the researchers introduced a momentum term. The original amount of change is obtained by multiplying the gradient of the error by the learning rate to move in the direction opposite the gradient; to this, one adds a fraction of the amount by which the weights were changed in the immediately preceding step. How much of the previous step's change is incorporated is determined by the momentum coefficient, a number between 0 and 1. This produces an effect similar to a rolling ball rolling over a small

bump due to its accumulated momentum. This technique has been used for a long time. In 2013, Ilya Sutskever and colleagues reexamined the effects of the momentum term in training deep neural networks.

The other concern was regarding local minima. A local minimum is a point where the error is lower than in its immediate vicinity, but is not the point where the error is lowest overall. Because gradient descent cannot find a direction to descend further once it reaches a point lower than its surroundings, critics pointed out that training could stall in a shallow puddle rather than the lowest ravine. The paper reported that such occurrences were rare in many tasks. It also explained that adding a few more hidden units and connections introduces new dimensions to the weight space, where each weight serves as a coordinate axis, and those dimensions provide paths to bypass the puddles.

3.3. Representations Learned by the Hidden Layer: Mirror Symmetry Detection and the 'Family Trees' Task

The shock the 1986 paper delivered to academia came from experiments rather than formulas. The researchers opened up the hidden layer of the trained neural network and illustrated what each neuron had learned.

The first experiment was mirror symmetry detection. It is a problem of determining whether a six-digit array of 0s and 1s is identical on the left and right relative to the center. For example, 101101 is symmetric, whereas 101100 is not. Nothing can be discerned in this problem by looking at only a single bit. One must compare matched pairs, such as the first and sixth digits, and the second and fifth digits. Belonging to the same category as the parity problem, which determines whether there is an odd number of 1s, it was a representative task that single-layer perceptrons could not solve. The parity problem itself was treated separately in the backpropagation chapter of the PDP book published in the same year.

The researchers trained a neural network with only two hidden units by presenting all 64 possible arrays. When the 64 arrays were presented repeatedly 1,425 times, the neural network discovered a neat solution on its own. Each hidden unit assigned weights of equal magnitude and opposite signs to paired positions across the center. From the outside to the inside, the magnitudes of the weights formed a ratio of 1:2:4.

The process by which this weight structure determines symmetry is as follows. If an array is completely symmetric, the left and right values cancel each other out, making the total input to the hidden unit 0. Because the hidden unit has a negative bias, it remains off when the total input is 0, and when both hidden units are off, the output unit answers 'symmetric.' If it is not symmetric, one of the two hidden units turns on and turns off the output unit. Thanks to the ratio of 1, 2, and 4, an asymmetric array never accidentally produces a sum of 0. Humans did not teach any of these rules. Looking only at correct and incorrect answers, the neural network translated the geometric rule of symmetry into a weight structure.

The centerpiece of the paper is the family trees task designed personally by Hinton. Hinton constructed family trees of 24 people in total: 12 members of an English family and 12 members of an Italian family. The two family trees have identical structures. For example, in the position of the couple Christopher and Penelope on the English side, there is the couple Roberto and Maria on the Italian side. There were 12 types of relationships: father, mother, husband, wife, son, daughter, uncle, aunt, brother, sister, nephew, and niece. The training data consisted of 104 propositions of the form 'person 1 + relationship = person 2.' For instance, Colin's aunts are Jennifer and Margaret.

The neural network consisted of five layers. At the very bottom, there were 24 units representing person 1 and 12 units representing the relationship. Because only one unit fired at a time, looking at the input alone revealed no commonality whatsoever between Colin of England and Alfonso of Italy. Above them followed, in sequence, 6 hidden units receiving the person and 6 hidden units receiving the relationship, 12 middle

hidden units, 6 output-side hidden units, and 24 output units representing person 2.

The researchers trained the network on only 100 of the 104 propositions and held out 4. When examining the 6 units in the first hidden layer that received the person after training was completed, each unit was responsible for a meaningful feature. Unit 1 distinguished between the English and the Italians. Unit 2 indicated which generation the person belonged to. Unit 6 indicated which branch of the family tree they belonged to. Setting aside the unit responsible for nationality, individuals occupying corresponding positions in the two family trees, such as Colin and Alfonso, yielded similar activation patterns. Concepts such as nationality, generation, and branch were nowhere specified in the input. They were features that the neural network discovered on its own in the process of predicting relationships.

The neural network was also not provided with rules such as "a mother's husband is a father." Nevertheless, it correctly answered the 4 propositions it had never encountered during

training. Such generalization was possible because of distributed representations, which represent an individual as a combination of multiple features. This was an approach diametrically opposed to the symbolic method of mapping a single meaning to a single symbol.

This experiment is regarded as an early example demonstrating that neural networks can learn the meanings of words and concepts as numerical vectors. The deep learning review paper published in Nature in 2015 by Hinton, LeCun, and Yoshua Bengio, as well as their 2021 article in Communications of the ACM, explain that modern word embeddings are an extension of this idea. Subsequent developments such as Word2Vec, Transformers, and large language models also belong to the same lineage in that they treat words by converting them into vectors.

3.4. The Carnegie Mellon University (CMU) Years and Relocation to Toronto, Canada

When the 1986 paper was published, Hinton's affiliation was Carnegie Mellon University in Pittsburgh, Pennsylvania, United States. He became a professor in the Department of Computer Science at the university in 1982 and worked there for five years until 1987. At the time, CMU was the center of symbolic artificial intelligence led by Allen Newell and Herbert Simon. According to a 2019 article by CMU celebrating the Turing Award, Hinton even co-taught courses with Newell, who represented symbolic AI. Amid colleagues with differing perspectives, he continued his connectionist research.

The CMU years were a period of prolific research achievements. Together with David Ackley and Terrence Sejnowski, Hinton investigated the Boltzmann machine, which borrowed concepts from statistical physics, and published a learning algorithm in Cognitive Science in 1985. During the same period, he completed the backpropagation paper with Rumelhart and

Williams. The George Boole 200 bicentenary records at University College Cork note that during his five years as a professor at CMU, Hinton pioneered research on backpropagation, Boltzmann machines, and distributed representations of words.

Around that time, the political situation in the United States moved in a direction contrary to Hinton's beliefs. In the midst of the Cold War with the Soviet Union, the Ronald Reagan administration pursued the Strategic Defense Initiative (SDI), nicknamed 'Star Wars.' The Defense Advanced Research Projects Agency (DARPA) under the Department of Defense also funded computing research, including artificial intelligence, with substantial budgets. A significant portion of artificial intelligence research funding at American universities came from the military.

Hinton considered himself a socialist and opposed the use of artificial intelligence in warfare. He did not want to accept military research funding. Encyclopædia Britannica notes that animosity toward the U.S. military and the Reagan

administration played a role in Hinton's decision to leave the United States for Canada in 1987, and that he opposed the battlefield use of artificial intelligence. Records also indicate that disappointment with the policies of the Reagan administration and opposition to military funding for artificial intelligence research were among the reasons for his relocation.

In 1987, Hinton resigned from his professorship at CMU and moved to the Department of Computer Science at the University of Toronto in Ontario, Canada. The institution that welcomed him was the Canadian Institute for Advanced Research (CIFAR). CIFAR is an organization that provides long-term support for basic research regardless of immediate commercial payoff or military purposes. According to CIFAR, Hinton joined as a fellow of CIFAR's first research program, 'Artificial Intelligence, Robotics and Society,' in 1987, and with support from CIFAR and the Natural Sciences and Engineering Research Council of Canada (NSERC), continued research in a field that was plagued by skepticism and a lack of funding at the time.

His first 10 years in Toronto were not smooth sailing. Entering the 1990s, as other machine learning methods such as support vector machines garnered attention, neural networks were once again pushed out of the spotlight. Hinton founded and led the Gatsby Computational Neuroscience Unit at University College London in London from 1998 to 2001 before returning to Toronto. In 2004, together with colleagues, he newly launched the 'Neural Computation and Adaptive Perception (NCAP)' program at CIFAR. This gathering served as a venue where neural network researchers such as LeCun and Bengio met regularly, laying the groundwork for AlexNet—developed in Hinton's lab by Alex Krizhevsky and Ilya Sutskever—to win the image recognition competition by a wide margin in 2012.

The 1986 collaborators subsequently took different paths. Ronald Williams published the REINFORCE algorithm, a policy gradient technique in reinforcement learning, in 1992. In 2001, Hinton became the inaugural recipient of the Rumelhart Prize, named after Rumelhart, and Rumelhart passed away in 2011. In 2024, Hinton received the Nobel Prize in Physics alongside John Hopfield for foundational discoveries and inventions that

enable machine learning with artificial neural networks. On October 8, 2024, the day of the Nobel Prize announcement, CIFAR published a celebratory article reflecting on its relationship with Hinton that began in 1987.

The longstanding conundrum that there was no way to train multilayer neural networks found a path to resolution with the 1986 backpropagation paper. The core legacy left by that paper lay in a shift in thinking rather than in computational techniques: even without humans providing rules, neural networks discover useful representations from data on their own. Holding onto this conviction, Hinton left the United States, which was awash in military research funds, for Canada, and pursued his research in Toronto for more than 20 years. Today's large language models and image generation models are also trained using backpropagation. Later, however, Hinton came to believe that the actual brain likely does not learn in the same way as backpropagation, and in 2022 he proposed alternatives such as the Forward-Forward algorithm.

Part 2. How to Make Neural Networks Learn

Chapter 4. Principles of Learning Found in Physics

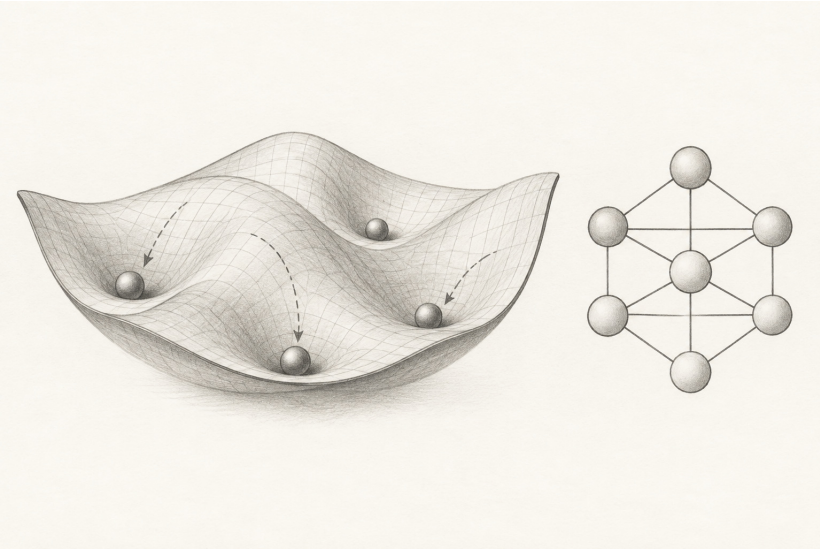


Figure 4. An analogy for the thinking of a neural network utilizing energy and probability.

4.1. A Meeting in Rochester: Adding Probability and Heat to Hopfield Networks

In the early 1980s, the artificial intelligence academic community was cold toward neural network research. After Marvin Minsky and Seymour Papert mathematically demonstrated the limitations of single-layer neural networks in Perceptrons in 1969, many researchers viewed neural networks as a dead end. Research funding and talent flocked to symbolic artificial intelligence, which sought to implement intelligence through rules and symbols. The person who provided new momentum to neural network research during this period was not a computer scientist, but a physicist.

John J. Hopfield was a theoretical physicist who bridged solid-state physics and biophysics. In 1982, while at the California Institute of Technology (Caltech), he published a paper titled "Neural networks and physical systems with emergent collective computational abilities" in Volume 79 of

the Proceedings of the National Academy of Sciences (PNAS). Hopfield later moved to Princeton University.

The tool Hopfield borrowed in this paper was a physical model called spin glass. Atoms behave like tiny magnets, and the orientation of these magnets is called spin. A spin takes one of two states: up (+1) or down (−1). In ordinary magnets, neighboring spins all align in the same direction. In spin glass, however, some pairs prefer the same direction, while other pairs prefer opposite directions. Since there is no arrangement that satisfies all pairs simultaneously, the system settles into one of several compromise states. Physicists represent these stable states as sunken valleys in an energy landscape.

Hopfield placed neurons in the positions of spins. This was because neurons also have two states: on and off. With this correspondence, interactions between spins become the connection strengths between neurons—that is, weights; the energy of the overall system becomes the energy of the network state; and energy valleys become stored memories. Hopfield added one condition here: the connections had to be

symmetric. Whichever two neurons were chosen, the connection strength from one to the other had to equal the connection strength in the opposite direction. He proved that under this condition, each time a neuron changes its state one by one, the total energy of the network decreases or stays the same. Because energy cannot decrease indefinitely, the network eventually comes to rest at the bottom of a valley. This bottom is called an attractor state.

Using this property, the network can be utilized as a memory device. If patterns to be stored are made into valley bottoms, even when an input is partially erased or mixed with noise, the network descends into the nearest valley and restores the original pattern. It is similar to hearing only the first measure of a song and recalling the remaining lyrics. This method of retrieving memory through part of the content rather than a storage location address is called content-addressable memory. Hopfield's paper demonstrated in the language of physics that neural networks operate stably, serving as an opportunity for physicists to take an interest in neural networks.

A workshop organized by Jerry Feldman was held at the University of Rochester in New York. Present at this gathering were Geoffrey Hinton, a professor at Carnegie Mellon University (CMU); Terrence Sejnowski, who had settled at Johns Hopkins University after a postdoctoral fellowship at Harvard Medical School; and Hopfield. Hopfield introduced his proof that his network necessarily converges to a stable state.

Hinton and Sejnowski saw both potential and weakness in this result. The weakness stemmed from the very property that guaranteed convergence. Hopfield networks always move only in the direction of decreasing energy. Therefore, once entering a shallow valley close to the initial input, there is no way to escape from it. This shallow valley is called a local minimum. The answers obtained at local minima were often spurious patterns that had never been stored—namely, false memories.

The two traveled back and forth between Pittsburgh and Baltimore, researching together every weekend. Pooling knowledge from psychology, computer science, neuroscience, and statistical physics, they sought a way to resolve this

weakness. The answer they found was heat. When neurons determined their state, rather than following a predetermined rule strictly, they were made to decide according to probability. Raising the temperature increases noise, so neurons occasionally move in the direction of increasing energy as well. The network can then climb over the walls of shallow valleys to find deeper valleys.

The two named this new network after the 19th-century Austrian physicist Ludwig Boltzmann. Boltzmann was the founder of statistical mechanics who derived the laws of temperature and energy from the random motion of countless particles. Hinton and Sejnowski first presented this concept in 1983 at a computer vision conference with a paper titled "Optimal perceptual inference."

4.2. Energy Landscapes, Stochastic Relaxation, and Hidden Units

Perceptrons and Hopfield networks operated only according to fixed rules. The Boltzmann machine transformed this into an energy-based model that performs probabilistic inference. An energy-based model assigns a number called energy to each state the network can occupy, viewing states with lower numbers as more plausible. Its structure starts from the rule that determines energy. In a Boltzmann machine, the state of each neuron (unit) is 0, meaning off, or 1, meaning on. The total energy of the network is defined as follows.

First, two connected units are grouped as a pair, and the states of the two units are multiplied by the weight of the connection between them. Only when both units are turned on does this product equal the weight; if even one is turned off, it becomes 0. After summing this value over all pairs, the sign is inverted. Adding all the thresholds of the units that are turned on to this yields the total energy of the network.

The meaning of this rule is as follows: When two units strongly connected with a positive weight are turned on together, energy decreases accordingly. This means that the better the relationship between the two units fits, the more stable the system becomes. A threshold is the magnitude of input required for each unit to turn on. A unit with a large threshold raises energy accordingly when turned on, so it does not turn on easily.

If this network is operated at a constant temperature for a long time, the system reaches thermal equilibrium. Here, temperature is a number that determines how randomly the units behave, a concept borrowed from temperature in physics. Thermal equilibrium is a state where, even though the states of individual units change continuously, the proportion at which each global state appears remains constant. A global state refers to a single configuration listing the on and off states of all units at once. At this point, the probability that the network is in a certain global state follows the Boltzmann distribution.

In the Boltzmann distribution, a number is assigned to each state, and this number decreases exponentially and steeply as the value of that state's energy divided by temperature increases. The probability of a state appearing is the value obtained by dividing the number assigned to that state by the sum of the numbers assigned to all states. Dividing this way makes the sum of probabilities across all states equal to 1.

According to this distribution, lower-energy states have a higher probability of appearing, while higher-energy states appear rarely. Temperature regulates the magnitude of this difference. When the temperature is high, the difference diminishes and various states appear evenly; when the temperature is low, the state with the lowest energy appears far more frequently than other states. The process by which neurons change states according to probability in this way, gradually finding stable configurations, is called stochastic relaxation.

How, then, should temperature be regulated to find deep valleys? Hinton and Sejnowski imported a method called simulated annealing. When metal is heated and then cooled

slowly, its internal structure can approach a stable state. In 1983, IBM researcher Scott Kirkpatrick and colleagues published the simulated annealing technique applying this principle to computational problems in Science. Initially, the temperature is raised, allowing the network to freely cross high hills in the energy landscape and explore a wide range. Then, as the temperature is lowered very slowly, the likelihood increases that the system will settle in the deepest valley of the landscape—namely, the global minimum. Around the same time, Paul Smolensky's Harmony Theory, and the image denoising research of brothers Stuart Geman and Donald Geman, applied the same concept to different problems.

The core achievement of the Boltzmann machine is a learning method for neural networks that include hidden units. Hidden units are units operating solely inside the network, neither receiving data from the outside nor outputting results. Why hidden units were necessary can be understood by looking at Minsky and Papert's critique. Perceptrons and Hopfield networks could only handle relationships between units directly in contact with the outside world. Yet even a seemingly

simple problem like exclusive OR (XOR)—which outputs 0 if two inputs are identical and 1 if they differ—cannot be solved through direct relationships between inputs alone. To represent higher-order relationships that do not emerge by pairing two elements at a time but only reveal themselves when multiple elements are viewed together, intermediate units are needed to organize them.

The Boltzmann machine answered this demand by dividing units into two types. Visible units are in contact with the outside world, receiving input data or outputting answers. Hidden units have no values assigned from the outside and cannot be observed. Hidden units are symmetrically connected to visible units, discovering and representing on their own the rules and features underlying the data.

The remaining problem was how to train the weights of hidden units. How to train units that cannot be told correct answers had long been a challenge in neural network research. The Boltzmann machine presented an answer to this challenge

through probability and statistical physics, contributing to paving the way for training multilayer neural networks.

4.3. The Waking Phase and Dreaming Phase: Learning Algorithm of the Boltzmann Machine

In 1985, David Ackley, Hinton, and Sejnowski published "A learning algorithm for Boltzmann machines" in Volume 9 of the journal Cognitive Science. The following year, Hinton and Sejnowski explained this content in detail in Chapter 7, "Learning and Relearning in Boltzmann Machines," of Volume 1 of Parallel Distributed Processing, published by the PDP Research Group.

The learning goal of the Boltzmann machine is to make two probability distributions match. A probability distribution is a collection of how frequently each possible state appears. One is the distribution that real-world data produces across visible units, and the other is the distribution that the network produces across visible units when operating on its own

without external input. Visible units are units directly in contact with the outside world that receive data. The difference between the two distributions is measured using Kullback-Leibler divergence from information theory. The authors named this value G.

This value is obtained by performing the following calculation for every state visible units can assume and summing them all up: divide the probability of that state appearing in the real data by the probability of it appearing when the network operates on its own, find the natural logarithm of that ratio, and multiply it by the probability in the real data. For states where the two probabilities are equal, the ratio is 1 and the logarithm of 1 is 0, so nothing is added. If there are states where the two probabilities diverge, the sum becomes greater than 0.

If this value is 0, the distribution produced by the network is identical to the distribution of the real data. This means that the network reproduces the picture of the world contained in the data exactly as it is. Therefore, learning becomes a matter of adjusting weights in a direction that reduces this value. Hinton

and Sejnowski proved that differentiating this value with respect to each weight—that is, calculating how much this value changes when a single weight is altered slightly—yields a concise result.

The result appears as the difference between two probabilities. One is the probability that two connected units are turned on together when equilibrium is reached with visible units clamped to real data, and the other is the probability that those two units are turned on together when equilibrium is reached with the network operating on its own without external input. The rate at which the difference between the two distributions changes when a weight is slightly altered equals the former probability minus the latter probability, divided by temperature, with the sign inverted.

From this result, the rule for modifying weights emerges directly.

The amount of modification for the weight connecting two units is the value obtained by multiplying the difference between two

measured values by a small number. The difference is obtained by subtracting the second value from the first value.

The small number multiplied here determines how much to modify at one time. The first value and the second value are values measured respectively in two different phases.

The first is the positive phase, likened to a waking phase. Visible units are clamped to real data, leaving only hidden units to fluctuate freely. Once thermal equilibrium is reached through simulated annealing, the average probability that two units are turned on together is measured. This value is the first value in the modification rule, corresponding to connections formed while viewing the world.

The second is the negative phase, likened to a dreaming phase. Visible units are unclamped, and the entire network is allowed to run on its own. At this time, once thermal equilibrium is reached, the average probability that two units are turned on together is measured. This value is the second value in the

modification rule, corresponding to connections generated by the network itself without external input.

Comparing the two values makes the meaning of the rule clear. If two units are turned on together more frequently when viewing real data than during the dreaming phase, the connection between them is strengthened. If they are turned on together only during the dreaming phase and not in the real data, the connection is weakened. What is learned from reality is reinforced, and unfounded associations made up by the network on its own are erased.

The advantage of this rule is locality. The information needed to modify the strength of a connection is solely the activity of the two units at either end of that connection. There is no need to propagate error signals backward from the opposite side of the network. Even when hidden units are arranged in multiple layers, the same rule applies. In that synapses in the brain can also only know the activity of two neighboring neurons, this rule appeared biologically plausible.

The explanation using sleep as an analogy was also aligned with hypotheses in sleep research at the time. In 1983, Francis Crick and Graeme Mitchison published a hypothesis on the function of dream sleep in Nature. Its premise was that during REM sleep, the brain performs reverse learning to erase useless associations. The negative phase of the Boltzmann machine aligned well with this hypothesis. Crick frequently attended PDP Group seminars and debated with neural network researchers.

However, this algorithm had a major stumbling block. Reaching thermal equilibrium took far too long. In each positive phase and negative phase, units had to be stochastically updated thousands to tens of thousands of times to obtain samples close to equilibrium. This kind of repeated sampling is called the Markov Chain Monte Carlo (MCMC) method. Because of this computational cost, practically training large neural networks was difficult at the time. The error backpropagation algorithm, published in Nature around the same time in 1986 by Hinton along with David Rumelhart and Ronald Williams, was much faster than this. Researchers' interest shifted toward backpropagation for a time.

Hinton did not give up on the Boltzmann machine. He continued researching methods to reduce computational load by restricting connections between units. This research later led to Restricted Boltzmann Machines (RBM), which eliminated connections within the same layer, and the Contrastive Divergence (CD) learning algorithm.

4.4. The 2024 Nobel Prize in Physics: The Encounter Between Statistical Physics and Artificial Intelligence

On October 8, 2024, just over 40 years after the Boltzmann machine was first presented, the Royal Swedish Academy of Sciences announced John Hopfield and Geoffrey Hinton as joint recipients of the Nobel Prize in Physics. The citation was "for foundational discoveries and inventions that enable machine learning with artificial neural networks." At the time, Hopfield was professor emeritus at Princeton University, and Hinton was professor emeritus at the University of Toronto. Hinton revealed that he received news of the award by phone at an inexpensive hotel in California, USA.

In its press materials, the Nobel Committee explained the achievements of the two individuals as a continuous stream. Hopfield created a network that stores and restores patterns using the energy concept of atomic spin. Building upon that network and adding tools from statistical physics, Hinton invented the Boltzmann machine, which discovers and classifies features of data on its own and even generates new examples. The Boltzmann machine was explicitly named in the official materials explaining the reasons for the award. Committee member Anders Irbäck noted at the announcement venue that the work of both researchers was rooted in physics. Committee Chair Ellen Moons stated that Hinton is recognized as a figure who led the development of efficient learning algorithms.

Immediately following the announcement, there was also controversy. Some physicists and online academic communities debated whether this research was physics investigating natural phenomena, or computer science and statistics. Tracing the academic lineage, the research of both individuals is rooted in disordered systems theory and spin

glass research within statistical physics. The achievement of Italian physicist Giorgio Parisi, who received the 2021 Nobel Prize in Physics, also involved research on disordered complex systems such as spin glass. In that they explain the process by which large-scale order emerges from the stochastic interactions of countless small units, Parisi's spin glass and Hinton's Boltzmann machine share the same question. The only difference is that in the Boltzmann machine, that order manifests in the form of memory and learning.

Following the award, Hinton received multiple requests from reporters to briefly explain his research. He reportedly replied by borrowing an old quip from physicist Richard Feynman: that if research could be explained in just a few minutes, it would not be worthy of a Nobel Prize.

On December 10, 2024, the award ceremony was held at Stockholm Concert Hall. Hinton received the Nobel Prize in Physics from King Carl XVI Gustaf of Sweden, and around 100 students and colleagues gathered at the University of Toronto to watch the ceremony. The prize money was 11 million

Swedish kronor, approximately 1.4 million Canadian dollars, and was shared with Hopfield. Hinton stated that he would donate a portion of his share to Water First, an Ontario organization improving clean water access for Indigenous peoples in Canada, and to an organization supporting neurodivergent youth.

Another portion of the prize money was used to commemorate research conducted with Sejnowski. In December 2024, the Conference on Neural Information Processing Systems (NeurIPS) announced that Hinton would establish a new prize with part of the prize money. The prize purse is 10,000 US dollars annually, awarded to a team of two or more researchers under the age of 40 who write a paper proposing a new theory on how the brain functions. The prize is named the Sejnowski-Hinton Prize, taken from the names of computational neurobiologist Terrence Sejnowski and Hinton. The names of the two men, who created the Boltzmann machine together more than 40 years ago while traveling between Pittsburgh and Baltimore, were placed side by side on an award for young researchers.

Hinton also left an item related to the Boltzmann machine at the Nobel Prize Museum. The microchip he donated in 2024 contains an early Boltzmann machine. Made in 1987 by Joshua Alspector, this chip demonstrated that a Boltzmann machine could be implemented on a dedicated chip. Although the Boltzmann machine inside the chip was a small network with only six nodes and therefore had limited practical application, as a case demonstrating the principle, it pointed the direction for the development of machine learning and artificial intelligence.

In the early 1980s, when most scholars turned away from neural networks, Hopfield proved the stability of neural networks using spin glass. Hinton and Sejnowski added heat and probability to this and introduced hidden units to create a machine that learns on its own. The Boltzmann machine was too slow to be widely used at the time. However, the idea of representing data with energy, reducing the discrepancy between reality and the model, and discovering hidden structures autonomously carried forward through Restricted Boltzmann Machines and Deep Belief Networks to today's

generative artificial intelligence. This idea, which began in the language of physics, was recognized 40 years later with the Nobel Prize in Physics.

Chapter 5. How to Learn Faster

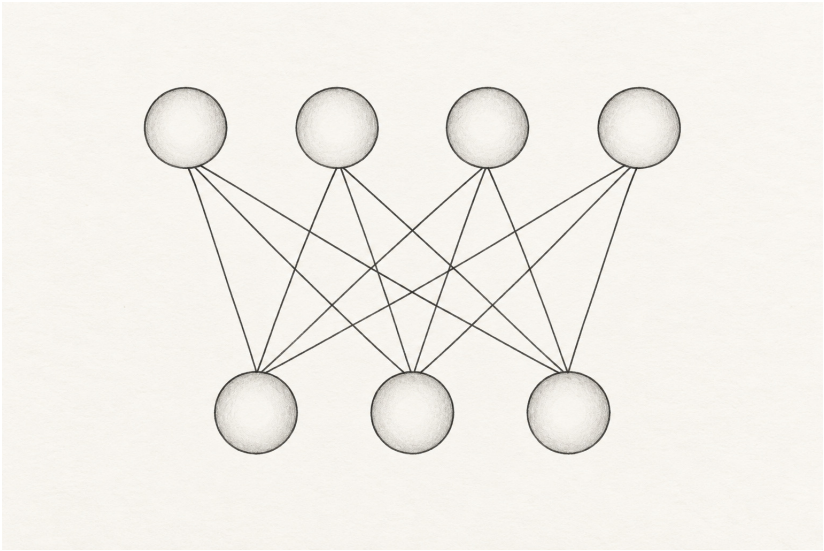


Figure 5. Conceptual diagram of a Restricted Boltzmann Machine composed of connections between two layers.

The Boltzmann machine, introduced in the 1980s, demonstrated that neural networks with hidden units could learn on their own. However, because computation took far too long, it was difficult to apply to real-world problems. For nearly 15 years thereafter, Hinton grappled with this problem and proposed two solutions. One was the Restricted Boltzmann Machine (RBM), which deliberately reduced connections in the neural network; the other was Contrastive Divergence (CD), which trained this neural network rapidly. Both ideas stemmed from a single question: Why is the Boltzmann machine so slow?

5.1. The Overly Slow Boltzmann Machine and the Idea of 'Products of Experts'

A neural network is a structure in which small computational units called units are interconnected. In this chapter, most units take one of two values: 1 meaning on, or 0 meaning off. The connection linking two units has an associated numerical value called a weight. If the weight is positive, the two units tend to turn on together; if negative, when one turns on, the other tends to turn off. Learning is the process of adjusting these weights little by little. Probability represents how often an event occurs as a number between 0 and 1, where values closer to 1 indicate more frequent occurrence.

The Boltzmann machine, introduced in the mid-1980s by Geoffrey Hinton and Terry Sejnowski, is a model that imported concepts from statistical physics into neural networks. In this model, units turn on or off according to probabilities rather than fixed rules. The model assigns an energy value to each overall state of the neural network—which records which units

are on and off—and defines probabilities such that states with lower energy occur more frequently.

Learning proceeds in two phases. First, in the 'wake phase,' where real data is clamped to the input units, the degree to which two connected units turn on together is measured. Next, in the 'dream phase,' where the inputs are unclamped and the network is allowed to run freely on its own, the same quantity is measured again. And the weight of that connection is updated by the difference between the two values. Connections that turn on together more frequently during the wake phase become stronger, while connections that turn on together more frequently during the dream phase become weaker.

The problem lay in the dream phase. To calculate the probability of any given data occurring, one must know the partition function, commonly denoted as Z. Z is obtained by feeding the energy of every state the neural network can assume into an exponential function—converting it such that lower energies yield larger values—and summing them all up. Because the probability of a state is the quotient obtained by

dividing that state's value by Z, probabilities cannot be determined without Z. However, because each unit takes two values, the number of states is 2 multiplied by itself as many times as there are units. Even with only 100 units in total, this reaches 2 to the 100th power, or roughly 10 to the 30th power. Summing this many terms one by one cannot be done by any computer.

Thus, researchers avoided calculating Z directly and instead used sampling. Sampling refers to actually drawing values according to probabilities, also known as sample extraction. By gathering results drawn multiple times, one can learn which states appear frequently without having to examine every state. In Boltzmann machines, a sampling method called Markov Chain Monte Carlo (MCMC) and simulated annealing were employed. This approach repeatedly alters the values of units according to probabilities, fluctuating the neural network over a long period until it reaches thermal equilibrium—where the proportion of each state's occurrence no longer changes—and then measures the statistics at that point. However, because this process had to be repeated every time weights were

updated, learning dragged on endlessly with computers of the 1980s and 1990s. In his 2024 Nobel Prize lecture, Hinton also recalled that the original Boltzmann machine was too slow, and that a few years later he devised the Restricted Boltzmann Machine, which improved efficiency by placing restrictions on the connections between neurons.

To solve this problem, Hinton reexamined the very way probabilistic models were constructed. The result was 'Products of Experts (PoE).' This research emerged during his tenure at the Gatsby Computational Neuroscience Unit at University College London (UCL), where Hinton served as its founding director from 1998 to 2001. It was first disclosed as a technical report in 2000, and was published as a formal paper in the journal Neural Computation, Vol. 14, No. 8, in 2002.

Here, an expert refers to a small model that assigns a probability reflecting how plausible each piece of data is. The way probabilities are distributed across all possible data is collectively called a probability distribution. At the time, mixture models, such as Gaussian Mixture Models (GMM), which

combine distributions from multiple experts by adding them together, were widely used. In this method, a weight is assigned to each expert, and the probability assigned by each expert is multiplied by that weight and summed together.

Hinton pointed out that this approach was poorly suited for high-dimensional data—data that requires many numbers to represent a single instance. Taking photographs of human faces as an example, factors such as lighting, pose, facial expression, and angle intertwine to form a curved space (manifold) of approximately 35 dimensions. To represent this space with a mixture model, each individual expert must broadly cover the entire 35 dimensions. However, the sum of multiple distributions cannot be sharper than the individual distributions that serve as its ingredients. In the end, the overall distribution becomes blunt.

Products of Experts, by contrast, multiplies the probabilities assigned by the experts rather than adding them. Since the multiplied value cannot be a valid probability as it is, it is divided by the sum of identical products computed across all

possible data so that the total sum equals 1. This process is called renormalization. In a multiplicative scheme, if even a single expert assigns a low probability, the resulting value drops sharply. Therefore, each expert needs to impose conditions on only a subset of features in the data. This is because high probabilities remain only in the narrow region where all conditions are satisfied simultaneously.

Hinton cited sentences as an example. One expert checks only whether the tense is correct, while another checks only whether the subject and verb agree in number. Multiplying the two gives a high probability only to sentences that adhere to both rules. In mixture models, each expert had to handle the problem as a whole, but in Products of Experts, experts each responsible for a narrow condition work together to create a sharp distribution.

Products of Experts offered yet another advantage. Given data, the latent variables hidden within each expert become mutually independent. Independence means that knowing the value of one does not alter the probability of the other; because this holds under the condition that data is given, it is termed

conditional independence. Thanks to this, the states of hidden variables can be inferred exactly in a single calculation. However, the normalizing divisor in renormalization still requires summing over all cases, making it difficult to compute. Hinton circumvented this remaining obstacle through Contrastive Divergence, described in Section 5.3.

5.2. Structure and Conditional Independence of the Restricted Boltzmann Machine (RBM)

Translating the idea of Products of Experts into a neural network architecture yielded the Restricted Boltzmann Machine. This structure itself was not originally created by Hinton; its roots trace back to 1986. In Chapter 6 of Volume 1 of Parallel Distributed Processing, published that year, Paul Smolensky introduced 'harmony theory' and presented a model called the Harmonium. In 1992, Yoav Freund and David Haussler studied methods for learning distributions of binary vectors—data consisting of sequences of 0s and 1s—using a two-layer neural network. In his 2002 paper, Hinton pointed out that this structure represents a Product of Experts where each hidden unit is viewed as an individual expert, and by combining this with the energy framework of the Boltzmann machine, he established a learning theory.

An RBM consists of only two layers. The visible layer is the layer that receives data, and the hidden layer is the layer that

discovers features not directly apparent in the data. A unit in the hidden layer, or a hidden unit, is trained to turn on when a specific pattern appears in the visible layer. Connections between visible units and hidden units have symmetric weights that use the same value regardless of the direction in which signals travel. Each unit also has a number called a bias, which determines how easily that unit turns on independently of other units.

What distinguishes an RBM from a standard Boltzmann machine is that all connections within the same layer have been eliminated. Visible units are not connected to visible units, nor are hidden units connected to hidden units. In mathematics, this configuration is called a bipartite graph. To use a classroom analogy, it is like a rule where students in the front row may speak only with students in the back row, and cannot speak to those sitting next to them in the same row.

The energy of an RBM is the value obtained by summing three components and reversing the sign: the biases of active visible units, the biases of active hidden units, and the weights of

connections where both units are turned on together. As two units with a high connection weight turn on together, the energy becomes lower, and according to the rules of the Boltzmann distribution, the probability of that state becomes higher.

Eliminating intra-layer connections had a significant effect. Once all values of the visible layer are fixed, hidden units do not exert influence on one another. This is because there are no direct paths between hidden units, and the values of the visible layer—the sole pathway between them—are already fixed. The probability that each hidden unit turns on is computed separately, relying solely on the signals feeding into it. One sums all the weights connected to active visible units, adds the unit's own bias, and feeds that sum into a sigmoid function. The sigmoid function returns a value between 0 and 1 regardless of the input number, producing a value closer to 1 as the input grows larger. This value represents the probability that the unit turns on.

The same holds in the reverse direction. Once all values of the hidden layer are determined, visible units become mutually independent, and the probability of each visible unit turning on is computed in the same manner from the sum of weights coming from active hidden units plus its own bias. Consequently, the probability of an entire layer taking on a particular set of values equals the product of the probabilities of each individual unit. This is called factorial conditional independence. This independence holds only under the condition that all values in the opposite layer are given. Without that condition, units in the same layer still influence one another via the opposite layer.

The value of this property becomes evident when compared to directed graphical models. In models where arrows point from causes to effects, a phenomenon known as 'explaining away' occurs. Consider two potential causes for wet grass: rain and a sprinkler. Originally, these two causes are unrelated. However, once we know that the grass is wet, the moment we confirm that it rained, the likelihood that the sprinkler operated decreases. Observing the effect entangles the causes with one

another. To determine the probabilities of hidden causes in such models, one had to rely on MCMC or complicated approximation calculations.

In an RBM, no such entanglement arises. When input data arrives, one accurately obtains the probabilities of the entire hidden layer through a single matrix-vector multiplication and a single sigmoid evaluation. However, this applies solely to inference—extracting hidden features from given data. When it came to training weights, Z remained a stumbling block.

5.3. The Principles of Contrastive Divergence (CD) Learning

Training an RBM involves updating weights so that the probability of generating real data increases. Which connection to adjust and in which direction is determined by the difference between two expectations. One is the proportion of times the visible unit and hidden unit of that connection turn on together when data is clamped to the visible layer. The other is the proportion of times the two units turn on together when the neural network operates freely on its own without data. If the former value is greater than the latter, the weight is increased; if smaller, it is decreased. When the two values become identical across all connections, the statistics produced by the model on its own match the statistics of the data, leaving nothing further to modify.

The former value is easily obtained via the sigmoid calculation described in Section 5.2. What proves difficult is the latter value. To obtain this value accurately, one must perform Gibbs sampling. This method alternates between computing hidden

layer probabilities from visible layer values to sample hidden unit values, and using those values to recompute visible layer probabilities to sample visible unit values. However, this round trip had to be repeated until equilibrium was reached—theoretically an infinite number of iterations. While reducing connections accelerated inference, the bottleneck in training remained intact.

Hinton decisively eliminated this bottleneck. He recognized that without needing to run all the way to equilibrium, one could obtain a usable learning direction simply by starting from real data, performing Gibbs sampling just once, and stopping there. This single-iteration approach is called CD-1, and its procedure consists of four steps.

First, feed real data into the visible layer. Calculate the probability of each hidden unit turning on, and sample a value of 0 or 1 according to that probability. At this time, measure the degree to which the visible unit and hidden unit turn on together for each connection.

Second, from the sampled hidden layer values, calculate the visible layer probabilities in reverse to generate new visible layer values. This is the result of the neural network redrawing the data in its own way. Hinton called this reconstruction, and sometimes referred to it as confabulation, meaning a fabricated story.

Third, feed the reconstruction back into the visible layer to obtain hidden layer values, and measure the degree to which both sides turn on together in the same manner.

Fourth, update the weights by multiplying the difference—obtained by subtracting the value measured in the third step from the value measured in the first step—by the learning rate. The learning rate is a small number that determines how much to adjust at one time. Connections that turn on together more frequently in the data see their weights increase, while connections that turn on together more frequently in the reconstruction see their weights decrease.

Why this procedure works is easy to understand if one visualizes an energy landscape. Suppose the horizontal axis represents all possible data and the vertical axis represents energy. A well-trained model has deep valleys where real data resides. The reconstruction is not a point chosen at random. It is a point near the real data that has moved one step toward where the model currently deems more plausible. Hinton compared this to a 'near miss.' In the weight update, the addition lowers the energy at the location of genuine data, while the subtraction raises the energy at the location of the incorrect answer right beside it. Repeating this process carves deep trenches along the data in the energy landscape. Rather than surveying all distant regions, the model specifically targets and fixes the places where it is most prone to error.

Hinton also provided a mathematical justification for this method. The rationale involves three distributions: the distribution of real data, the distribution when the model reaches equilibrium, and the distribution obtained after starting from data and undergoing a prescribed number of Gibbs sampling steps. How much two distributions differ is

measured by a quantity called Kullback-Leibler (KL) divergence, which is 0 when two distributions are identical and grows larger as they diverge. Standard maximum likelihood learning, which seeks to maximize the probability of generating data, minimizes the divergence between the data distribution and the equilibrium distribution. Contrastive divergence, instead, minimizes the difference between two divergences: the divergence between the data distribution and the equilibrium distribution minus the divergence between the sampled distribution and the equilibrium distribution.

As sampling is repeated, the distribution approaches equilibrium, so the subtracted value is never greater than the former value. Therefore, this difference cannot fall below 0, and it becomes 0 when the model reconstructs data faithfully such that the sampled distribution matches the data distribution. The name 'Contrastive Divergence' originates from this contrasting comparison of two divergences.

CD-1 also possesses statistical advantages. Genuine data and its reconstruction are always tied as a pair. Just as comparing

matched pairs of similar patients in clinical trials reduces errors due to chance, paired comparisons reduce the fluctuation (variance) in sampling results. Hence, even with only a single step of Gibbs sampling, learning progressed stably.

Contrastive divergence is an approximation and not a perfect method. Because the expectation that should be measured at equilibrium is replaced by a reconstruction just one step away, the update direction can deviate from the exact direction that maximizes the probability of the data. This limitation was revealed in subsequent research by Hinton himself and his colleagues. In a 2005 study, Miguel Carreira-Perpiñán and Hinton showed that while CD can converge to a point slightly off from the true maximum likelihood solution, that deviation is usually small. In 2010, Ilya Sutskever and Tijmen Tieleman analyzed that the update direction of CD is not the gradient descending precisely along any single objective function. In other words, it is not a method that picks a single target value and moves in the direction where that value decreases most rapidly. To remedy this weakness, Tieleman proposed Persistent Contrastive Divergence (PCD) in 2008, which

continues running the Markov chain of Gibbs sampling rather than restarting it anew each time. In 2010, Hinton wrote 'A Practical Guide to Training Restricted Boltzmann Machines,' outlining practical configuration methods such as learning rates, numbers of hidden units, and initial weight values.

5.4. Variants of RBM and Expansion to Pre-training

The combination of RBM and CD expanded its scope of application beyond patterns of 0s and 1s to various types of data. In that process, three notable changes emerged.

The first was the Gaussian-Bernoulli RBM, which modified the visible units. To handle continuous real values, such as speech frequency features (MFCC) or photograph brightness, visible units cannot be restricted to only 0 and 1. This model replaced visible units with linear units subject to Gaussian noise. The values of the visible units are sampled from a normal distribution whose mean is the sum of weights from active

hidden units plus their own bias. This model later served as the starting point for speech recognition research.

The second change was modifying the hidden units. At the 2010 International Conference on Machine Learning (ICML), Vinod Nair and Hinton presented an RBM using Rectified Linear Units (ReLU). The two conceived of a 'stepped sigmoid unit' that stacked infinitely many binary units sharing identical weights but having slightly differing biases. The behavior of this unit is approximated by a function that yields 0 if the input is less than 0, and a value proportional to the input if greater than 0. Adding noise to this, they used it in a form that outputs the sum of input and noise when greater than 0, and outputs 0 otherwise. This ReLU was used in AlexNet in 2012, subsequently becoming the standard activation function in deep learning.

The third change was the method of stacking RBMs layer upon layer. In a paper published in Neural Computation, Vol. 18 in 2006, Hinton, Simon Osindero, and Yee-Whye Teh proposed training by taking the hidden layer output of a first RBM as the input data for a second RBM. This method of training one layer

at a time with unlabeled data is called Greedy Layer-wise Unsupervised Pre-training, and neural networks built this way are called Deep Belief Networks (DBN). In the same year, Hinton and Ruslan Salakhutdinov published a paper in Science, Vol. 313, in which they pre-trained a deep autoencoder—a neural network that compresses input into a compact code and then reconstructs it—using RBMs and fine-tuned it with backpropagation. As a result, it achieved superior performance over conventional methods in reducing high-dimensional data into compact codes. In 2009, the two expanded the theory to the Deep Boltzmann Machine (DBM), where all layers are connected via undirected, symmetric connections.

RBM pre-training subsequently yielded breakthroughs in speech recognition. A 2012 paper co-authored by researchers from the University of Toronto, Microsoft, Google, and IBM reported that deep neural networks initialized using the DBN approach outperformed existing methods in acoustic modeling.

Since then, circumstances have shifted. With the advent of ReLU, dropout, large-scale datasets, and graphics processing

units (GPUs), it became possible to train deep neural networks from scratch using backpropagation without pre-training. Today, instances of using RBMs in production services are rare. An article introducing Hinton's Nobel lecture on December 8, 2024, summarized the lecture as a story about the 'historical catalyst' that sparked the deep learning revolution, while noting that the Boltzmann machine learning algorithm is no longer in use today. Like a catalyst that triggers a chemical reaction without remaining in the final product, the RBM opened the door to training deep neural networks and stepped aside. During Nobel Week, Hinton donated an early postage-stamp-sized Boltzmann machine chip to the Nobel Prize Museum.

Although the RBM itself has fallen out of use, the ideas left behind by RBM and CD endure: namely, the notion of shaping a model by lowering the energy of genuine data and raising the energy of plausible impostors, rather than calculating all possibilities to compute exact probabilities. Yann LeCun and others have formalized this approach within the framework of Energy-Based Models. Some researchers view modern

contrastive learning, which learns by contrasting positive pairs against negative pairs, as sharing this same foundational perspective.

Hinton offered three solutions to the speed bottleneck of the Boltzmann machine. With Products of Experts, which multiplies rather than adds probabilities, he paved the way to sharply represent high-dimensional data. With the RBM, which eliminated connections within the same layer, he enabled hidden features to be inferred accurately in a single step. And with Contrastive Divergence, which avoids waiting for equilibrium by reconstructing just one step and comparing, he overcame the barrier of training speed. These choices—reducing connections and slashing computations—served as the stepping stone for deep learning to regain the spotlight from 2006 onward.

Chapter 6. Deep Neural Networks Regain the Spotlight

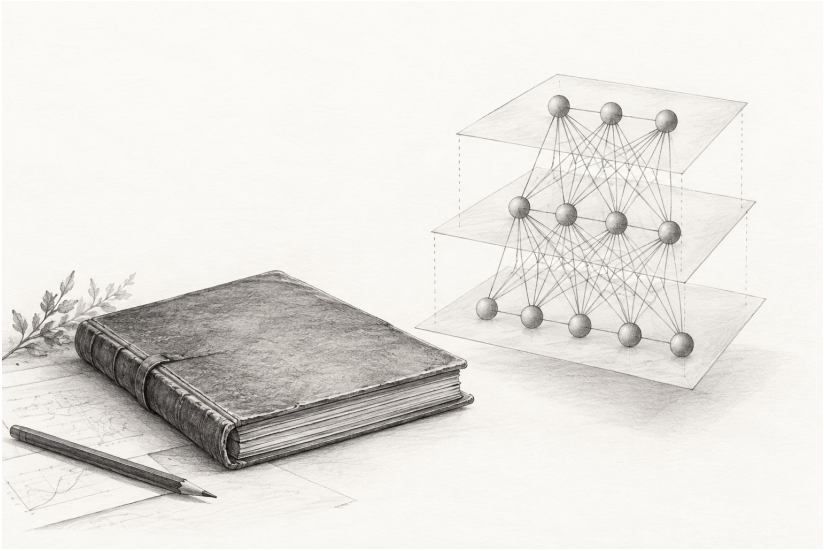


Figure 6. An expression of the concept of researching neural networks composed of multiple layers.

6.1. The Second AI Winter and the Vanishing Gradient Problem

In 1986, when David Rumelhart, Geoffrey Hinton, and Ronald Williams published the error backpropagation method in Nature, expectations for neural networks rose significantly. Backpropagation is a method that calculates the difference between the answer produced by a neural network and the correct answer, and then transmits that difference backward from the output layer toward the input layer, adjusting the connection strengths—that is, the weights—little by little. This method worked well in neural networks with two or three layers. However, when the same method was applied to deep neural networks with multiple hidden layers stacked between the input and output layers, learning failed to occur properly. At that time, three main problems hindered the training of deep neural networks.

The first obstacle was the vanishing gradient problem. In backpropagation, the error signal is multiplied by a derivative each time it passes through a layer. For the sigmoid function

widely used at the time, the derivative was at most 0.25 even at its maximum. When numbers less than 1 are multiplied repeatedly, the value approaches 0. As a result, the lower layers far from the output layer received virtually no learning signal. Conversely, exploding gradients also occurred, where values grew with each layer, destabilizing learning. Either way, it was the same problem in that a learning signal of appropriate magnitude failed to reach the lower layers.

The second was the failure of optimization. Optimization is the process of finding weights that reduce the error. If one pictures the way the error rises and falls according to the weights as a landscape, the error landscape of a deep neural network is rugged and complex. Deep neural networks starting from randomly chosen weights would wander and stall near local minima or flat saddle points in this landscape, failing to find good weights. A local minimum is a point that is lower than its immediate surroundings but not the lowest point overall. A saddle point is a point where the slope is uphill in some directions and downhill in others, so that the gradient at that spot is close to 0.

The third was overfitting stemming from a shortage of data. Data with labels manually attached by humans—that is, labeled data—was difficult to obtain. Deep neural networks with hundreds of thousands to millions of weights ended up memorizing even random noise in small datasets. This was akin to getting practice problems right, but having one's performance drop on entirely unfamiliar problems.

Under these circumstances, the interest of the machine learning community shifted toward other methods. Support vector machines (SVM) and kernel methods could turn the learning problem into a convex optimization problem. In convex optimization, the error landscape has only a single valley, so it always reaches the same optimal solution. This was an advantage in contrast to neural networks, which risked getting trapped in local minima. Random forests, which combine multiple decision trees, also yielded strong performance. By comparison, neural networks were regarded as an approach that sounded plausible in theory, but was difficult to handle in practice.

As a result, artificial neural network research went through a long slump from the mid-1990s to the early 2000s. In the history of artificial intelligence, the stagnation that arrived after expectations for expert systems collapsed in the late 1980s is called the second AI winter. The winter experienced by neural network researchers was even longer. Until the mid-2000s, mainstream academia did not view neural networks as a serious subject of research. Hinton and his colleagues later recalled in interviews that if they included the words 'neural network' in a paper's title, it was difficult to pass conference peer review.

The institution that supported neural network research so it could continue during this period was the Canadian Institute for Advanced Research (CIFAR). In 2004, Hinton of the University of Toronto, together with Yoshua Bengio of the University of Montreal, Yann LeCun of New York University, David Fleet of the University of Toronto, and others, proposed a new research program. CIFAR accepted this proposal and created the 'Neural Computation and Adaptive Perception (NCAP)' program, which

Hinton led for about 10 years. This program later changed its name to 'Learning in Machines & Brains.'

NCAP provided research funding in 3- to 5-year units based on short proposals just a few pages long. Nor did it demand immediate commercial results. A small group of researchers centered around Hinton, Bengio, and LeCun met regularly to share ideas, and they held summer schools to nurture young researchers. The 2006 research that discovered a way to train deep neural networks emerged from within this research community.

6.2. Learning Layer by Layer: Greedy Layer-Wise Unsupervised Pre-training and DBN

In 2006, Hinton, together with Simon Osindero of the University of Toronto and Yee-Whye Teh of the National University of Singapore, published the paper 'A Fast Learning Algorithm for Deep Belief Nets' in Neural Computation, volume 18, issue 7. The paper appeared from page 1527 to page 1554.

The problem addressed in this paper was the phenomenon of 'explaining away.' A neural network that views causes in an upper layer as producing effects in a lower layer is called a directed belief network. In such neural networks, it is difficult to infer the causes backward from observing the result. Consider rain and a sprinkler as causes for wet grass. Knowing the grass is wet, once we learn that it rained, the likelihood that the sprinkler was on decreases. Two causes that were originally unrelated become entangled the moment the outcome is known. In a neural network with multiple hidden layers and thousands of units, this entanglement became so complex that exact inference was virtually impossible.

The tool Hinton's research team used to solve this problem was the Restricted Boltzmann Machine (RBM). The Boltzmann machine is a stochastic neural network that Hinton developed in the 1980s with Terry Sejnowski, named after the statistical physics formulas of nineteenth-century physicist Ludwig Boltzmann. A stochastic neural network is one where, instead of producing fixed values, each unit turns on or off probabilistically. The RBM is a simplified form of this model with

reduced connections. It consists of two layers: a visible layer that receives data and a hidden layer that captures features of the data, with connections only between the two layers and no connections between units within the same layer. Thanks to this restriction, once the values of one layer are fixed, the units in the other layer can be computed all at once, independently of each other, making computation much easier. In 2002, Hinton had published a method called contrastive divergence (CD), enabling RBMs to be trained rapidly.

The researchers circumvented the explaining away problem mathematically. In an infinitely deep directed belief network, if the weights of each layer are tied to the transpose of the weights of the layer immediately above it, the prior probability generated by the upper layer precisely cancels out the entanglement arising in the lower layer. The researchers called this 'complementary priors.' They then demonstrated that this infinitely deep directed model with tied weights is mathematically equivalent to a single two-layer RBM. A deep directed model, which was difficult to perform inference on, could now be handled by converting it into an RBM for which a

learning method was already available. From this equivalence emerged greedy layer-wise unsupervised pre-training.

The learning procedure is as follows. In the first step, unlabeled input data is fed into the visible layer, and the weights of the bottom RBM are trained using contrastive divergence. In the second step, after freezing these weights, the input data is passed through to calculate the probabilities that the hidden units turn on. The activations of this hidden layer become the new input data for the next layer. In the third step, a second RBM is trained separately on this new data. The same process is repeated layer by layer until the desired depth is reached. It is called 'greedy' in the sense that the results of prior steps are frozen and only the current layer is trained optimally. It is 'unsupervised' because it learns solely from data without correct answers, and it is 'pre-training' because it prepares the weights in advance prior to full-scale training.

In the fourth step, the stacked RBMs are combined into a single generative model. This is the Deep Belief Network (DBN). The top two layers of the DBN retain undirected symmetric

connections, acting as a kind of associative memory. The layers below them are converted into directed connections pointing from top to bottom, taking on the role of generating data.

The researchers also provided a mathematical justification for this procedure. They proved that each time an additional RBM is stacked on top and trained while holding the lower-layer weights fixed, the lower bound on the degree to which the model explains the data does not deteriorate and can improve. This was the first result to offer a mathematical basis showing that the model improves as more layers are added.

The weights of a neural network that has completed pre-training are not random values, but are set at a favorable starting point that already captures the structure of the data to some extent. Considering that the optimization failure seen in Section 6.1 stemmed from starting from random weights, altering this starting point was the crux of this method. On top of that, fine-tuning is performed to refine all the weights using a small amount of labeled data. In this paper, fine-tuning was

carried out using an 'up-down' algorithm, a modified version of the wake-sleep algorithm.

The researchers tested performance on the handwritten digit dataset MNIST. MNIST consists of digit images measuring 28×28 pixels, divided into 60,000 training images and 10,000 test images. A DBN with hidden layers of 500, 500, and 2,000 units recorded an error rate of 1.25% using only pixel values, without techniques such as image shape information or data augmentation. Under the same conditions, this was lower than the 1.4% achieved by support vector machines and the approximately 1.5% achieved by neural networks trained with backpropagation. This DBN did not stop at classifying digits; it could also draw new digit images using the trained model. Under comparative conditions that did not separately exploit the spatial information of images, it demonstrated that deep neural networks could outperform prominent existing methods.

Hinton's research team was not alone in reaching this conclusion. That same year, Bengio's research team presented results at the NIPS conference showing that layer-wise

pre-training aided the training of deep neural networks not only with RBMs but also in other architectures. LeCun's research team also released research in a similar direction. The three research groups of the CIFAR community had reached the same conclusion at nearly the same time.

6.3. The 2006 Science Paper: Deep Autoencoders and Nonlinear Dimensionality Reduction

On July 28, 2006, Hinton and doctoral student Ruslan Salakhutdinov published the paper 'Reducing the Dimensionality of Data with Neural Networks' in Science, volume 313, issue 5786. The paper was four pages long, spanning from page 504 to page 507. While the DBN paper published in the same month reached researchers through a specialized journal, this paper signaled the revival of deep neural network research through a journal read by the broader scientific community.

What this paper addressed was the problem of dimensionality reduction. Dimensionality reduction is the task of summarizing data with a very large number of figures into a smaller number of figures. If a digit image composed of 784 pixels can be reduced to 30 numbers while still allowing the original shape to be reconstructed, those 30 numbers can be seen as capturing the essence of the image. For a long time, principal component analysis (PCA), formulated by Harold Hotelling in 1933, was widely used for this task. PCA finds the straight-line directions along which the data is most widely spread and projects the data in those directions.

The problem is that real-world data does not lie on a flat plane. Data such as images or documents lies on a curved surface like a rolled sheet of paper—that is, on a nonlinear manifold. If one shines light on rolled paper from above, parts that are far apart appear to overlap. Likewise, PCA, which projects onto straight axes, crushes the curved structure.

An autoencoder is a neural network that can overcome this limitation. It consists of an encoder that compresses the input

through progressively smaller layers, and a decoder that reconstructs it back to its original size from a narrow central code layer. When trained so that the input and output are identical, the core of the data remains in the central code layer. This concept had existed since the 1980s. However, when actually implemented, the problems described in Section 6.1 emerged unchanged. When a deep autoencoder was trained with backpropagation starting from random weights, the learning signal failed to properly reach the central code layer, and the results were comparable to or worse than PCA.

Hinton and Salakhutdinov solved this problem through layer-wise RBM pre-training. The method consists of three steps. First, multiple RBMs are trained sequentially from the bottom up. In the MNIST experiment, they used an architecture reducing the 784 inputs through 1,000, 500, and 250 down to 30. Second, the trained RBM weights are stacked to form an encoder, and the same weights are transposed and stacked in reverse to form a decoder. In this way, the autoencoder is 'unrolled.' Third, the units that stochastically output 0 or 1 are replaced with sigmoid units that output continuous values, and

the entire neural network is fine-tuned via backpropagation to reduce the difference between the original input and the reconstructed output. Once pre-training provided a favorable starting point, backpropagation fulfilled its proper role even in deep neural networks.

The results were distinct even when compared visually. When digit images were compressed into 30 numbers and then reconstructed, the images produced by the deep autoencoder were nearly identical to the originals. The reconstructed images from PCA compressed to the same size were blurry and smudged. When digit images were mapped onto a two-dimensional plane, the difference was also stark. In the PCA plot, the digits from 0 to 9 were jumbled and overlapping, but in the autoencoder plot, each digit separated into distinct clusters. On photographs of human faces as well, the autoencoder showed smaller reconstruction errors than PCA.

The researchers also applied this method to documents. When some 800,000 news articles were represented by the frequencies of 2,000 frequently used words and then

compressed, articles on similar topics clustered together. This was a superior result compared to Latent Semantic Analysis (LSA), an existing linear method. This research led to Salakhutdinov and Hinton's 2009 paper on 'semantic hashing.' By converting documents into short binary codes, one can search for documents with similar meanings extremely quickly by finding documents with similar codes.

At the end of the paper, the two researchers summarized the conditions under which this achievement became possible. Although the idea that deep autoencoders could perform nonlinear dimensionality reduction had existed since the 1980s, its realization required fast computers, large datasets, and good initial weights. In 2006, all three conditions were met, and among them, the method for obtaining good initial weights was precisely layer-wise pre-training.

6.4. Deep Boltzmann Machines (DBM) and the Spread of Deep Learning

In 2009, Salakhutdinov and Hinton presented the Deep Boltzmann Machine (DBM) at the 12th International Conference on Artificial Intelligence and Statistics (AISTATS) held in Florida, United States. A DBN was a hybrid model where only the top two layers were connected without direction, while the lower layers were connected only from top to bottom. In contrast, a DBM connected the visible layer and all hidden layers with undirected, symmetric connections.

This difference manifested in its inference mechanism. An intermediate hidden layer in a DBM determines its own state by jointly receiving sensory information ascending from the layer below and contextual information descending from the layer above. This resembles how humans infer characters by looking at the surrounding context when reading blurred handwriting. For this reason, DBMs handled partially occluded or noisy inputs relatively well.

In return, DBMs were tricky to train. The researchers conducted layer-wise RBM pre-training, but modified it to suit an architecture receiving signals from both above and below by doubling or halving the weights. Thereafter, they performed inference using a mean-field approximation and adjusted the overall weights using a sampling method called persistent contrastive divergence (persistent CD). The DBM trained this way recorded an error rate of 0.95% on MNIST and an error rate of 10.8% on the NORB dataset—consisting of three-dimensional toy objects photographed from various angles—yielding a lower error than the 11.6% of the comparison benchmark, support vector machines.

Around this time, the efficacy of pre-training began to spread beyond the laboratory. The first field was speech recognition. Abdel-rahman Mohamed and George Dahl of the University of Toronto, together with Hinton, applied a deep neural network pre-trained with a DBN to an acoustic model that identifies sound units of speech. For decades, the field of speech recognition had used an approach combining Gaussian Mixture Models (GMM) and Hidden Markov Models (HMM) as its

standard. Deep neural networks surpassed this method on the TIMIT phoneme recognition task and soon achieved strong results in large-vocabulary speech recognition as well. In 2012, researchers from the University of Toronto, Microsoft, Google, and IBM published a joint paper in IEEE Signal Processing Magazine, stating that the four research groups had arrived at the same conclusion. Subsequently, major companies replaced the cores of their speech recognition engines with deep neural networks.

In that same year of 2012, Hinton's students Alex Krizhevsky and Ilya Sutskever, using AlexNet—a neural network built together with Hinton—won the ImageNet competition by a wide margin over second place, recording a top-5 error rate of 15.3%. The field of computer vision also transitioned rapidly to deep neural networks.

However, right around the time deep neural networks began to be widely used, layer-wise pre-training began to step back from the mainstream. This was because means to solve the three problems discussed in Section 6.1 in other ways had emerged.

Computing speeds increased dramatically with graphics processing units (GPUs), and millions of labeled images became available. The ReLU activation function, which mitigates vanishing gradients; dropout, which randomly disables some units during training; and improved methods for initializing weights also appeared. As a result, it became possible to train deep neural networks directly with backpropagation from the start without pre-training. AlexNet was also a neural network trained without pre-training.

Even so, the achievements of 2006 did not lose their significance. In 2015, LeCun, Bengio, and Hinton reviewed this era in their review paper 'Deep Learning' published in Nature, volume 521. The three wrote that around 2006, the researchers assembled by CIFAR revived interest in deep neural networks. Their assessment was that the method of learning features layer by layer from unlabeled data to create good initial weights, and then refining them with backpropagation, proved that deep neural networks could work well even in situations with little labeled data. The contribution of this research was that it

dismantled the longstanding conventional wisdom that deep neural networks could not be trained.

The concept of training first on vast amounts of unlabeled data and then fine-tuning with a small amount of labeled data continues today in large language models. Today's models do not stack RBMs. However, the overarching framework of pre-training first on vast corpora of text and fine-tuning for specific tasks closely resembles the thinking of 2006.

Hinton, Bengio, and LeCun jointly received the Turing Award in 2018. The Turing Award is often called the Nobel Prize of computer science. And on October 8, 2024, the Royal Swedish Academy of Sciences announced that it was awarding the Nobel Prize in Physics to Hinton and John Hopfield for foundational discoveries and inventions that enable machine learning with artificial neural networks. The committee noted that Hinton created the Boltzmann machine based on the Hopfield network, and by advancing this research, contributed to opening the way for today's explosive development of machine learning. The DBN of 2006 was precisely a model developed

from that Boltzmann machine. The Boltzmann machine research, continued even during the stagnation of neural network research, had led to a method for training deep neural networks.

Part 3. Deep Learning Emerges into the World

Chapter 7. Computers That Recognize Speech and Images

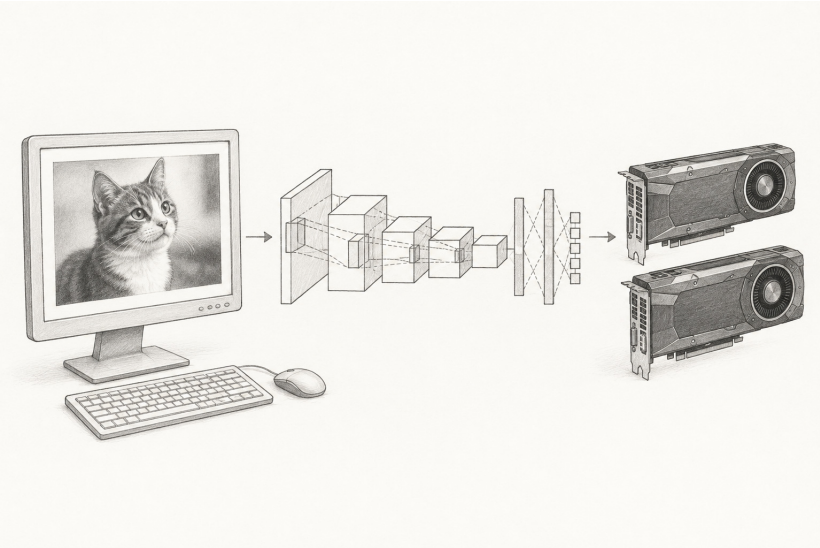


Figure 7. Depicting image recognition and the role of GPUs.

In 2006, Geoffrey Hinton opened the way to train deep neural networks with deep belief networks (DBNs). However, most of the artificial intelligence academic community still did not trust neural networks. A few papers showing good results were not enough to change long-standing conventional wisdom. What changed academia's judgment was numbers emerging from real-world problems. From 2009 to 2012, Hinton's laboratory produced such numbers twice. The first came from speech recognition, understanding human speech, and the second came from computer vision, recognizing objects in photographs.

7.1. The First Breakthrough: The Transformation of Speech Recognition (2009–2012)

The technology by which a computer converts human speech into text is called automatic speech recognition. This technology is broadly divided into two parts. One is the acoustic model, which determines what pronunciation a very short snippet of sound corresponds to, and the other is the component that connects those identified pronunciations into words and sentences. Among these, the acoustic model had remained with the same method for nearly 30 years until the late 2000s. It was an approach that used Gaussian Mixture Models (GMMs) to calculate the probabilities of sound features and Hidden Markov Models (HMMs) to track changes over time.

While this method yielded good performance for a long time, its weaknesses were also clear. The GMM is structured on the assumption that a single snippet of sound is accounted for by one 'expert' among several. However, human speech is not divided so neatly. Adjacent pronunciations blend together, and

the properties of multiple sounds overlap at any single moment. To represent these overlaps, GMMs had to continually increase their components, requiring correspondingly large amounts of computation and data. It was also difficult to examine sounds across multiple consecutive moments all at once.

The people who first overcame this limitation were George Dahl and Abdel-rahman Mohamed, doctoral students in Hinton's lab at the University of Toronto. In 2009, the two applied the DBN method that Hinton introduced in 2006 to acoustic models. First, they stacked Restricted Boltzmann Machines (RBMs) layer by layer to learn the structure of sounds without correct answers, and then refined the entire neural network using backpropagation. For the input, rather than feeding just a single snippet of sound, they fed a bundle of 11 to 15 consecutive snippets. The neural network could directly take in the preceding and following context that GMMs had struggled to handle.

What made this experiment possible were graphics cards, namely GPUs. GPUs were originally devices designed to render gaming graphics, but they excel at processing large volumes of identical computations all at once. Using NVIDIA Tesla GPUs and CUDAMat, a computational tool developed by lab colleague Volodymyr Mnih, the two increased training speed by more than 20 times compared to using a single CPU. When experiments that previously took several weeks finished in just a few hours, it became possible to test multiple configurations repeatedly.

The results drew the attention of the speech recognition academic community. On TIMIT, a standard benchmark dataset long used by researchers, their neural network recorded a phone error rate of 20.7%. Since a phoneme is the smallest unit of sound in speech, this meant it made errors on roughly 21 out of 100 phonemes. This performance surpassed the best record of conventional methods reported up to that time. The initial results were presented at a NIPS workshop in 2009, and the refined findings were published in an IEEE journal in 2012.

However, TIMIT was a small-scale benchmark dataset. Confirming whether this method could also work in commercial products required the data and systems held by large companies. Hinton's students therefore went directly to companies. George Dahl went to Microsoft Research (MSR) and Abdel-rahman Mohamed to IBM Research for internships. Another student, Navdeep Jaitly, went to Google.

At Microsoft Research, Li Deng and Dong Yu collaborated with Hinton's research team. They applied deep neural networks to noisy Bing mobile voice search data recorded on mobile phones, and the sentence-level error rate dropped from 36.2% to 30.4%. Subsequently, on Switchboard, a 300-hour telephone conversation dataset, the word error rate fell from 27.4% to 18.5%. This amounted to cutting errors by roughly one-third. This result matched the 18.6% achieved by conventional systems that mobilized over 2,000 hours of data and complex adaptation techniques. With far less data, a single neural network had reached the same level.

At Google, Jaitly experimented with 5,870 hours of voice input data alongside Vincent Vanhoucke, Patrick Nguyen, and others. The word error rate dropped from 16.0% to 12.3%. Google refined this technology and incorporated it into voice search on Android smartphones in 2012. A neural network that began in a university laboratory had entered a product used daily by many people.

These findings, independently validated across several companies, were consolidated into a single paper. In November 2012, four research groups—the University of Toronto, Microsoft, Google, and IBM—published a joint paper titled 'Deep Neural Networks for Acoustic Modeling in Speech Recognition' in IEEE Signal Processing Magazine. It was rare for researchers from competing companies to share authorship on a single paper. This paper signaled to the academic community that the standard for speech recognition had shifted, marking one of the major achievements deep learning secured in industrial settings. However, this change took place primarily among speech recognition specialists. It would take a little more time

for the broader artificial intelligence community to fully realize it.

7.2. A Turning Point in Computer Vision: Fei-Fei Li's ImageNet and AlexNet (2012)

In the late 2000s, the computer vision community followed an approach where humans manually designed usable features from images. SIFT and HOG were prime examples. They extracted features by calculating edge orientations or brightness variations in photos and judged object classes using classifiers such as Support Vector Machines (SVMs). The performance of this method was easily shaken whenever lighting changed, an object was photographed from a different angle, or parts were occluded. Although performance improved year by year, the margin of improvement was small.

Around this time, the person who prepared the ingredients to change the situation was Fei-Fei Li. Li launched the ImageNet project at Princeton University in 2007 and continued the

research after moving to Stanford University in 2009. Li's team had humans manually label photographs collected from the internet. They recruited the necessary workforce through Amazon Mechanical Turk, an online crowdsourcing marketplace. In this way, a database containing over 15 million images across more than 22,000 categories was created. From its sheer scale, it was fundamentally different from the image datasets researchers had used until then.

Starting in 2010, they selected 1,000 categories and approximately 1.2 million training images from this database to host an annual competition called ILSVRC. Performance was evaluated primarily by 'top-5 error rate.' In this method, the computer outputs five candidate answers for each photo, and if the correct answer is not among them, it is counted as incorrect. The winning team's error rate was 28.2% in 2010 and 25.8% in 2011. Methods relying on handcrafted features were stalling around this level without making further progress.

In 2012, Hinton entered this competition with his two students, Alex Krizhevsky and Ilya Sutskever. The team name was

SuperVision. The roles of the three individuals differed from one another. The person who wrote the code and trained the neural network was Krizhevsky, while Sutskever was the one who strongly urged applying neural networks to large image data. Hinton steered the direction as advisor. Krizhevsky carried out this work on a computer equipped with two NVIDIA graphics cards, and the neural network was later named AlexNet after him.

The roots of AlexNet lie in the convolutional neural network (CNN) that Yann LeCun created in the late 1980s to recognize handwritten digits. A convolutional neural network scans small patches of an image sequentially to find patterns, recognizing increasingly complex shapes as it progresses through layers. AlexNet scaled up this architecture to accommodate large, complex color photographs. It consisted of eight layers in total, combining five convolutional layers and three fully connected layers. It had approximately 60 million parameters—values adjusted through learning—and around 650,000 neurons. At the time, it was an unusually massive size.

Yet they did not merely scale up its size. To properly train a neural network of this scale, the researchers employed four methods together. First was the use of ReLU as an activation function. An activation function is a rule that governs how much signal a neuron outputs. Functions widely used until then caused training to slow down because their gradients approached zero as inputs grew large. ReLU is a simple function that converts values less than zero to zero and passes values greater than zero as they are. In experiments, the neural network using ReLU trained six times faster than one using conventional functions.

Second was splitting the neural network across two GPUs. The NVIDIA GTX 580 used by Krizhevsky had only 3 GB of memory, making it impossible to fit the entire neural network onto a single device. Writing the GPU computation code cuda-convnet himself, he divided the neural network in half across the two GPUs and allowed them to exchange information only at certain layers. It took 5 to 6 days to train through the 1.2 million images nearly 90 times.

Third was dropout. With as many as 60 million parameters, a neural network can memorize training images entirely, making its performance prone to dropping on unseen images. This is called overfitting. Dropout is a method that randomly turns off half of the neurons during each training step. This produces the effect of training a slightly different neural network each time, preventing the network from becoming overly reliant on specific neurons. AlexNet applied this method to two of its later layers.

Fourth was augmenting the training data. From images of 256×256 resolution, they randomly cropped 224×224 patches, flipped them horizontally, and slightly altered color tones. This effectively created multiple new training examples from a single photograph.

What the neural network learned from photos through this training can be observed in its first layer. Plotting the 96 filters that AlexNet's first layer learned on its own reveals lines in various orientations, edges, and color blobs. No one taught it to look for such patterns. The neural network constructed them

itself by looking at photographs. These patterns resembled those to which cells in the human brain's primary visual cortex respond.

The competition results were announced in October 2012 at a European Conference on Computer Vision (ECCV) workshop in Florence, Italy. AlexNet's top-5 error rate was 15.3%. The second-place team from the University of Tokyo recorded 26.2% using conventional methods such as SIFT and Fisher vectors. The gap between the two teams was 10.9 percentage points. In a competition where rankings had been decided by margins of 1 to 2 percentage points, it slashed errors by more than 40% compared to the runner-up.

7.3. Transition to Deep Learning, Founding of DNNresearch, and Acquisition by Google

Reactions in the Florence presentation hall were mixed. Scholars who had long researched computer vision were not convinced whether this method was useful. In contrast, Yann LeCun, who was present at the venue, hailed it as a turning point for artificial intelligence. LeCun had previously worked as a postdoctoral researcher under Hinton in Toronto in the late 1980s.

Subsequent developments unfolded just as LeCun anticipated. Prior to AlexNet, almost none of the major computer vision papers utilized neural networks, and submitting a neural network paper often led to rejection in peer review. Within one to two years following AlexNet, however, researchers abandoned handcrafted feature engineering and moved to deep learning. Beginning with the competition the next year, most of the top-ranking teams used convolutional neural networks. This moment, when the accumulated power of

connectionism—relegated to the margins for more than 30 years—erupted all at once, is often likened to the bursting of a pent-up dam. Fei-Fei Li later noted that many view this as the moment modern artificial intelligence was born.

Corporations moved as well. Big Tech companies that had already witnessed the power of neural networks in speech recognition sought to recruit Hinton and his two students upon seeing AlexNet. Hinton decided to establish a company and sell it. He later revealed in several interviews that he needed substantial funds to secure the future of his son, who had learning disabilities. In late 2012, Hinton founded a company called DNNresearch together with Krizhevsky and Sutskever. The name was derived from Deep Neural Networks. It had no products, no revenue, and only three employees. All the company possessed was the technology that created AlexNet and the talent of the three researchers.

In early December 2012, an auction for the company took place in Lake Tahoe, United States, where the NIPS conference was being held. Baidu, Google, Microsoft, and DeepMind, a startup

founded just two years earlier, participated in the bidding. The auction proceeded via email, and Hinton conducted it from his hotel room while keeping bidders unaware of who else was participating. The starting bid was $12 million, and increments were set to at least $1 million at a time.

DeepMind bid with its own company stock instead of cash and was the first to bow out, while Microsoft dropped out around $22 million. The ones remaining until the end were Google and Baidu. When the price reached $44 million, Hinton stopped the bidding and decided to hand the company to Google. For him, finding the right place to continue his research mattered more than fetching the highest price. The acquisition was officially announced in March 2013.

Following the acquisition, the three pursued different paths. Hinton joined Google while maintaining his professorship at the University of Toronto, later participating in neural network research at Google as Vice President and Engineering Fellow. At Google Brain, Sutskever introduced the sequence-to-sequence (Seq2Seq) model in 2014 to convert one sentence into another,

contributing to advances in machine translation. He left Google in late 2015 to co-found OpenAI with Sam Altman and others, leading the research that would later lead to ChatGPT as Chief Scientist. Krizhevsky worked at Google for several more years before leaving the company in 2017.

This auction altered the calculus of the entire industry. With a price tag of $44 million placed on a three-person company, the market value of individual deep learning researchers was reset. In January 2014, Google acquired DeepMind, its rival in the auction, for approximately $600 million. Baidu and Microsoft, having lost in the auction, also dove into deep learning research in earnest.

The original code of AlexNet became Google's property with the acquisition of DNNresearch. It was over a decade later that this code was released to the public. Hansen Hsu, a curator at the Computer History Museum (CHM), proposed releasing the code to Krizhevsky in 2020, and after five years of negotiations with Google, secured the original codebase. On March 20, 2025, the museum announced together with Google that the original

AlexNet code was released to the public and permanently preserved. The released code totals approximately 200 KB combining NVIDIA CUDA code, Python, and C++. Compared to today's artificial intelligence models, it is a very modest scale, yet this small code altered the course of an era.

The speech recognition breakthroughs of 2009 and AlexNet in 2012 both came from the same laboratory. Both events proved with hard numbers the idea Hinton had clung to for decades: that neural networks can learn features on their own from data without humans specifying the rules. Once the two conditions of large-scale data and GPUs were in place, decades of accumulated research bore fruit all at once. Through these four years, connectionism, once an outsider discipline, became the mainstream of artificial intelligence.

Chapter 8. Reducing Overfitting and Transferring Knowledge

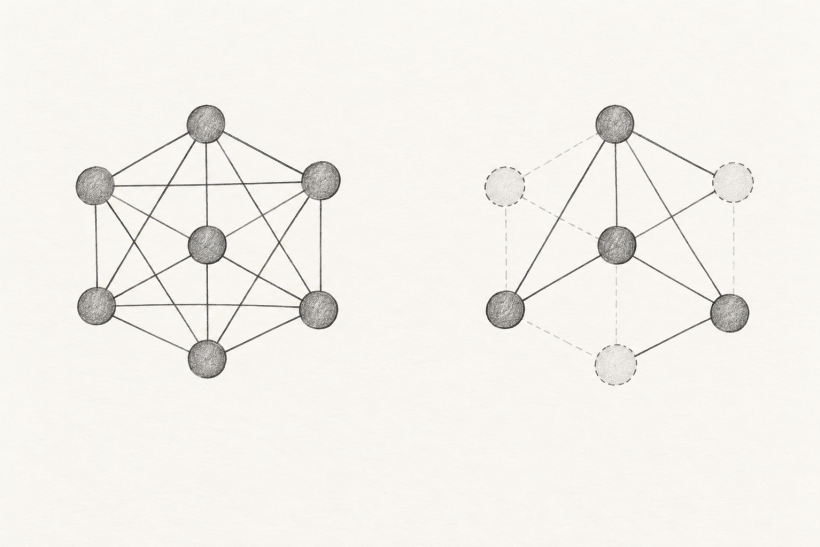


Figure 8. Illustration of dropout, in which neurons are temporarily turned off during training.

8.1. The Surging Parameters and the Overfitting Problem: The Birth of Dropout

In the early 2010s, as Deep Belief Networks (DBNs), deep autoencoders, and AlexNet successively achieved breakthrough results, artificial neural networks grew deeper, and their parameter counts increased substantially. Parameters are the numbers that a neural network adjusts on its own as it learns. When these numbers grew to tens of millions and hundreds of millions, researchers once again came face-to-face with an age-old problem: overfitting.

Overfitting is similar to the case of a student who memorizes every question and answer in the exam syllabus by heart. That student scores a perfect mark on the practice problems, but is helpless when confronted with an unfamiliar question. When a neural network is trained repeatedly thousands of times on the same data with limited training samples, it memorizes not only the underlying rules in the data but also the accidental noise present in each sample. As a result, while the error on the

training data approaches zero, its performance degrades on new evaluation data. This is known as a failure of generalization.

To prevent this problem, the fields of statistics and machine learning had long employed regularization techniques such as weight decay (L1 and L2 weight decay), early stopping, and weight sharing. However, deep neural networks with multiple stacked layers of nonlinear hidden units possessed such vast parameter spaces that it was difficult to curb overfitting using these methods alone.

One solution was ensembling. An ensemble is a method of training multiple models separately with different architectures or initial values and then averaging their predictions. It operates on the same principle as asking multiple people and pooling their answers, whereby an individual's mistake gets washed out. The problem was cost. Training a single massive neural network took days or weeks. Training dozens of such networks separately and running them simultaneously in production

services was simply impractical in terms of computational load and latency.

Geoffrey Hinton proposed dropout as a way to overcome this limitation. It was a joint research effort with Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov from the University of Toronto. In July 2012, the research team first posted "Improving neural networks by preventing co-adaptation of feature detectors" on arXiv, and in 2014, they formally published "Dropout: A Simple Way to Prevent Neural Networks from Overfitting" in Volume 15 of the Journal of Machine Learning Research (JMLR). The central idea was to jointly train an enormous number of different, reduced-size neural networks within a single network during training, having them share weights. It was an attempt to achieve the benefits of an ensemble without paying the cost of an ensemble.

The problem the researchers targeted was co-adaptation, which appeared in the title of their first paper. In standard backpropagation training, hidden units easily form complex relationships where they compensate for each other's mistakes.

It is like a group project where one person consistently covers for what another misses; the moment a team member changes, the entire project collapses. A neural network suffering from severe co-adaptation performs well on the training data, but its predictions fluctuate wildly when even slightly different data is introduced.

Hinton found the starting point of his thinking to solve this problem in everyday experience and biology. In an online Q&A, he remarked that noticing the tellers at the bank constantly changing whenever he visited led him to deduce that it was a mechanism designed to prevent employees from colluding to commit fraud. The idea was that randomly dropping some neurons each time would likewise prevent neurons from colluding with one another.

The theory of sexual reproduction in evolutionary biology provided theoretical grounding for this idea. Asexual reproduction passes on the parent's gene combination almost intact to offspring. It appears advantageous on the surface because it does not lose bundles of genes already well-adapted

to the environment. In contrast, because sexual reproduction takes half of the genes at random from each parent and mixes them, gene combinations that had meshed well and functioned properly are broken up with every generation. Yet, most higher organisms have chosen sexual reproduction. In a 2010 paper, biologist Adi Livnat and others explained that this is because natural selection selects for the mix-ability of genes rather than the fitness of a single individual. This means that genes that pull their own weight no matter what genes they are mixed with survive, rather than relying on a specific group of genes that always travel together.

Hinton carried this principle directly over to the training of hidden layers. By randomly turning off a subset of neurons during each training step, dropout prevents any neuron from relying on the certainty that its neighboring neurons will always be present. Each neuron must therefore find useful features—such as small patterns like edges or dots—on its own, regardless of which peers it works with. Just as genes that pull their own weight when mixed with any other gene are selected,

neurons that perform their role reliably within any combination of neurons are cultivated.

8.2. Computational Mechanics of Dropout and Experimental Results

The computational method of dropout is simple. One can think of it as tossing a coin for each neuron in every layer each time a training data sample comes in. If heads comes up with a predefined probability, the neuron is kept active; if tails comes up, its output is set to zero for that moment. This probability of keeping a neuron alive is called the retention probability. The paper recommended a default retention probability of 0.5 for hidden layers and around 0.8 for the input layer.

The computational procedure step by step is as follows. The output of a layer is a vector containing one value per neuron. For each neuron in this layer, a Bernoulli random variable taking the value 0 or 1 is sampled, where the probability of drawing 1 is the retention probability. This sampled vector of 0s and 1s is multiplied element-wise by the output vector, and this

partially masked output is passed as input to the next layer. The computation for the next layer proceeds as usual.

The number of possible ways to turn neurons on and off in a neural network is 2 raised to the power of the number of neurons. At each training step, dropout effectively samples one of these configurations at random to train. Those numerous thinned subnetworks do not exist separately; they share the same weights. It is an architecture that trains a massive ensemble at the cost of a single neural network.

When training is complete and the neural network is actually deployed, not a single neuron is turned off. Instead, outgoing weights from each neuron are multiplied by p. If a neuron was active only half the time during training, the average signal received by the next layer was half of the original. Multiplying the weights by 0.5 during evaluation ensures that the next layer receives a signal of the same magnitude as during training. Through this simple adjustment alone, an averaged prediction across 2^n subnetworks—more precisely, an approximation of the geometric mean—is obtained in a single computation.

The researchers also confirmed that dropout was most effective when combined with other techniques. The first was max-norm regularization, a method that prevents the norm of the weight vector incoming to any single neuron from exceeding a constant c (typically between 3 and 4). The second was a high learning rate and large momentum. Because dropout injects noise into training, using a learning rate 10 to 100 times larger than that of a standard neural network alongside a momentum coefficient of roughly 0.95 to 0.99 yielded good results. While a large learning rate risked weights blowing up, the max-norm constraint mitigated that danger. In this way, the two techniques complemented each other.

Experiments confirmed consistent effectiveness across various domains. On MNIST, the handwritten digit dataset, the error rate of a standard neural network was 1.60%. A network with two hidden layers and 8,192 neurons per layer using dropout alongside ReLU and max-norm recorded 0.95% without early stopping, even with approximately 65 million parameters. When pre-trained with a Deep Boltzmann Machine (DBM) and

fine-tuned with dropout, the error rate dropped further to 0.79%.

Results were similar on other datasets. On SVHN, Google Street View house numbers, the error rate of a convolutional neural network was 3.95%, but it dropped to 3.02% when dropout was applied only to the fully connected layers, and down to 2.55% when applied to all layers. On the object classification task CIFAR-10, the error rate was reduced from 15.60% to 12.61%, and on CIFAR-100, from 43.48% to 37.20%. On the speech recognition benchmark TIMIT, the phoneme error rate of a four-layer acoustic model decreased from 22.7% to 19.7%. Significant improvements were also observed in Reuters text classification and alternative splicing prediction in genetics.

AlexNet, which won the 2012 image recognition competition ILSVRC, also used dropout. Krizhevsky, Sutskever, and Hinton applied dropout with a retention probability of 0.5 to the first two fully connected layers of this network, which had approximately 60 million parameters. The paper noted that without dropout the network would have suffered from

substantial overfitting, while observing that it roughly doubled the number of iterations required for convergence. Dropout thus came at the cost of prolonged training time.

Examining the interior of a neural network trained with dropout reveals other characteristics as well. The average activation value of hidden neurons decreases, and a sparse representation naturally emerges where most neurons maintain values close to zero. The paper also outlined variants such as Gaussian dropout, which multiplies activations by normally distributed noise with a mean of 1 instead of 0 and 1.

Dropout subsequently served as a standard tool in deep neural networks for over a decade. However, its usage evolved over time. After the advent of new techniques such as batch normalization, its prominence declined in convolutional neural networks, and in today's massive language models that train on vast datasets for only one or two epochs, training is often conducted with dropout disabled altogether. When data is abundant relative to model size, there is less room for the model to simply memorize the data. Even so, it remains widely

used today in domains with scarce data or during fine-tuning stages.

8.3. 'Dark Knowledge' and Insect Metamorphosis: The Conception of Knowledge Distillation

Although dropout and ensembles substantially improved the generalization performance of neural networks, industry ran into a different problem. Massive models with tens of millions to hundreds of millions of parameters, or ensembles combining dozens of models, could not easily be deployed directly into environments with limited memory and stringent latency requirements, such as mobile phones or voice search. The challenge of practically deploying well-trained models remained.

Hinton tackled this problem with Oriol Vinyals and Jeff Dean, with whom he was working at Google. The three presented "Distilling the Knowledge in a Neural Network" at the 2014 NIPS

Deep Learning Workshop, and this paper was published on arXiv (1503.02531) in March 2015.

The paper begins with the story of insect metamorphosis. Insects such as butterflies or moths live in two completely different forms: the larval stage and the adult stage. The larval body is optimized for absorbing as much nourishment as possible from its surroundings, whereas the adult body is built for locomotion and reproduction. Hinton believed that machine learning should likewise separate the forms used for training and deployment. Up to that point, models used for training were deployed almost directly into production services. However, for the training phase, which extracts structure from vast amounts of data, a large, cumbersome model or a dropout ensemble—the larval form—is ideal. Once training is complete, that knowledge can be transferred to a small, fast model—the adult form—and deployed. This is the core concept of knowledge distillation. Here, the large model is called the teacher, and the small model is called the student.

Similar attempts had been made earlier. In their 2006 paper "Model Compression," Cristian Buciluǎ, Rich Caruana, and Alexandru Niculescu-Mizil demonstrated a method for training a small model to mimic the predictions of a large ensemble. Hinton took this a step further. He did not view a model's knowledge as the numerical weights themselves. Between neural networks with different architectures, copying weights is meaningless. What needed to be transferred was the mapping from inputs to outputs—that is, the function learned by the model.

The question then becomes where the information contained in that function can be read out. Hinton's answer to this was 'dark knowledge.' Suppose a handwritten image of the digit 2 is fed into a well-trained teacher model. The model might output a probability of around 0.999 for '2,' one in a million for '3,' and one in a billion for '7.' From the standpoint of hard targets that only mark the correct answer, both '3' and '7' are equally incorrect. However, the fact that there is a 1,000-fold difference between the probabilities of the two wrong answers carries information that the teacher model views this '2' as much closer

to a '3' than to a '7.' A distribution that includes these small probabilities assigned to incorrect answers is called a soft target. In lectures, Hinton referred to this hidden similarity information as 'dark knowledge'—a metaphor drawn from dark matter, which constitutes most of the universe's mass yet remains invisible.

The difference between the two types of targets reveals itself in what the student learns. If only the correct answer is taught to the student model, the student merely learns how to predict the correct answer. When taught with soft targets, the student learns how the teacher divides and perceives the world, including what it considers similar to what. The student inherits the teacher's very mode of generalization.

8.4. Softmax Temperature Control and Knowledge Transfer to Small Student Models

There was an obstacle to using dark knowledge in training. Probabilities like one in a million or one in a billion are so small that they exert almost no influence when calculating the cross-entropy loss used in training and get buried. The researchers resolved this problem using a mechanism borrowed from statistical physics called temperature (T).

The final layer of a neural network outputs a score for each category (class) to be classified. These scores are called logits. The softmax function converts these scores into probabilities that sum to 1. In distillation, a parameter called temperature is introduced into this function. Softmax with temperature is calculated as follows.

First, the logit for each class is divided by the temperature. That value is passed through an exponential function to make it positive; the exponential function grows rapidly as its input

increases. The probability of a class is then calculated by dividing the value obtained for that class by the sum of the values obtained across all classes.

Under normal circumstances, the temperature is set to 1. Raising the temperature significantly, such as to 20, causes all logits to be divided by a large number, effectively narrowing the differences between scores. Consequently, probabilities that were once extreme, such as 0.999 and one in a million, spread out more evenly to values like 0.45, 0.30, and 0.15. This is analogous to compressing dynamic range in an audio recording containing both loud sounds and whispers, bringing whispers into clear audibility. During distillation, softmax is computed for both the teacher model and the student model at the same elevated temperature, and the student is trained to match the teacher's softened probability distribution. Once training is complete, the student model operates with the temperature returned to 1.

The researchers found that the best results were obtained when the student's loss function was composed of two parts. One is

the loss comparing the teacher's soft targets with the student's predictions at high temperature, and the other is the loss comparing the ground-truth labels with the student's predictions at temperature 1. A weighted average of the two losses is taken, with the former loss multiplied by the square of the temperature. This is because raising the temperature causes the magnitude of gradients produced by the soft targets to scale inversely with the square of the temperature. Multiplying by the square of the temperature ensures that the relative weight between the two losses remains consistent even when experimenting with different temperatures. Assigning a relatively smaller weight to the ground-truth loss yielded good results.

The paper also presents mathematical results clarifying the relationship between prior research and knowledge distillation. Assuming the temperature is sufficiently high relative to the magnitude of the logits, and that the logits of both teacher and student have each been zero-centered, the direction in which the student adjusts its scores approximates the direction that minimizes the difference between the teacher's and student's

scores. The number of classes and the temperature affect the magnitude of these updates. Distillation at high temperatures thus approximates training to minimize the squared difference between the logits of the two models. Caruana and colleagues' 2006 logit-matching method was effectively a special case of knowledge distillation. However, higher temperatures were not always better. The authors reported that when the student model is very small, intermediate temperatures work better. At excessively high temperatures, noise from very low scores that the teacher barely considered is transferred to the student.

Experimental results demonstrated the efficacy of distillation in concrete figures. On MNIST, a teacher model with two hidden layers of 1,200 neurons each was trained using dropout and input image jittering, making 67 errors out of 10,000 test images. A small model with 800 neurons trained without regularization made 146 errors. When that same small model was trained on the teacher's soft targets at a temperature of 20, the number of errors dropped to 74. Although the student had never seen jittered images directly, the generalization

capability the teacher gained from that experience was transferred through the soft targets.

The researchers also conducted an even bolder experiment: removing all images of the digit 3 from the student's training data. The student was trained solely on the teacher's soft targets without ever having seen the digit 3. The result was 206 errors, of which 133 came from the 1,010 images of the digit 3 in the test data. This was because the student was excessively reluctant to output 3 as an answer. When the researchers increased the bias for the digit 3 by 3.5, the total number of errors dropped to 109, of which only 14 were errors on the digit 3. The student correctly classified a digit it had never seen before with 98.6% accuracy. Information about the shape of the digit 3 was embedded within the small probabilities the teacher had assigned to other digit images such as 2, 5, and 8, indicating that they 'somewhat resembled a 3.'

The researchers applied this technique to acoustic models for Google Android voice search as well. The baseline model achieved 58.9% frame classification accuracy and a 10.9% word

error rate. An ensemble combining 10 models of the same architecture recorded 61.1% and 10.7%. When this ensemble was distilled into a single model of identical size, it achieved 60.8% and 10.7%. A single model thus captured more than 80% of the frame accuracy improvement gained by the ensemble. On JFT, an internal Google dataset comprising 100 million images and 15,000 labels, the researchers also proposed a method of training multiple specialist models in parallel, each dedicated to handling a subset of easily confusable classes.

Hinton later viewed distillation from a broader perspective. In his 2022 paper on the Forward-Forward algorithm, he wrote that human language might itself be a form of distillation, transferring representations from one person's brain into another's. In the same writing, he noted that unlike digital models that can replicate weights directly, humans can share knowledge only through the narrow channel of speech. This line of inquiry later fed into his reasoning when warning about the dangers of digital intelligence.

Knowledge distillation found even broader application in the era of large language models. In January 2025, China's DeepSeek released its reasoning model DeepSeek-R1 along with six dense models of sizes 1.5B, 7B, 8B, 14B, 32B, and 70B distilled from it, built on Qwen and Llama. DeepSeek stated that its distilled 14B model substantially outperformed QwQ-32B-Preview, which was among the top open models at the time. The DeepSeek researchers reported that for small models, distillation delivered better performance than relying on reinforcement learning alone. In late January of that same year, OpenAI alleged that DeepSeek had copied its models via model distillation, turning distillation from a purely technical topic into an issue of intellectual property and competitive order. However, distillation for language models today often involves having the student learn directly from sentences generated by the teacher, differing from Hinton's original formulation of transferring the entire probability distribution over incorrect answers.

Dropout enabled giant neural networks to learn rules rather than memorize training data, while knowledge distillation

transferred that learned knowledge into compact models so that people could use them on handheld devices. One solved the problem of training; the other solved the problem of deployment. Both techniques epitomize Hinton's research philosophy in that they found clues in the living world—like genes that become more resilient when randomly shuffled, or insects that transform their physical forms throughout life.

Chapter 9. Advances in Deep Learning Built Together



Figure 9. Illustrates the research advanced together by colleagues and students.

9.1. The Institution That Weathered the 'AI Winter': CIFAR's NCAP Program and Long-Term Support Model

In the early 2000s, statistical models such as Support Vector Machines (SVM), kernel methods, and decision trees were at the center of the artificial intelligence academic community. These methods made it easy to explain results mathematically and came with guarantees that solutions close to the correct answer could be found. For artificial neural networks stacked in multiple layers, the situation was different. Learning frequently halted midway or became trapped at erratic points, and proving why they worked well was difficult. As a result, many researchers regarded neural networks as an outdated and useless method. Papers addressing neural networks were frequently rejected during peer review at academic conferences, and securing research funding was difficult. Even after passing the previous slumps in artificial intelligence research, the cold reception toward neural network research continued for some time.

The pillar that kept neural network research from being discontinued during this period was the Canadian Institute for Advanced Research (CIFAR). CIFAR is a nonprofit research institute established in Toronto in 1982, funded by the Canadian federal government, provincial governments, and private donors. This institution does not belong to any specific university. It employs a method of bringing together outstanding researchers from multiple countries under a single theme to conduct research together over a long period.

Geoffrey Hinton's connection with CIFAR began in 1987. That year, Hinton left his professorship at Carnegie Mellon University (CMU) in the United States and moved to the University of Toronto in Canada. At the time, a significant portion of artificial intelligence research funding in the United States came from the Department of Defense, and Hinton did not want to conduct research with money intended for military purposes. Coming to Toronto, he participated as a fellow in CIFAR's first research program, 'Artificial Intelligence, Robotics & Society.'

Seventeen years later, in 2004, neural network research was largely neglected by academia. At this time, Hinton proposed a new research program to CIFAR together with his colleagues. Yoshua Bengio of the University of Montreal and David Fleet of the University of Toronto joined in the proposal. CIFAR accepted this proposal and launched the 'Neural Computation and Adaptive Perception (NCAP)' program that same year, with Hinton serving as the program's director. After its 10-year operation was evaluated, NCAP was renamed 'Learning in Machines & Brains,' and it continues to this day.

The reason a topic outside the interest of academia could receive support for so long lay in CIFAR's funding model. Typical research grants are disbursed on a project basis. Researchers must promise what they will produce and by when, and they must submit final reports by set deadlines. CIFAR supported people and research groups rather than projects. It did not impose topics on a carefully selected small group of researchers, and it provided a venue for them to think and debate together over several years. Every year, researchers

gathered at small workshops to openly share unpublished ideas and even failed experiments.

Hinton directly explained this difference in an interview with the Seoul Economic Daily reported in October 2024. He acknowledged that the scale of research funding in Canada was not as large as that in the United States. However, he noted that while American professors spend most of their time writing grant proposals, in Canada, one could receive research funding for 3 to 5 years with a proposal of about 6 pages. He viewed the basic research funding that protects researchers' autonomy and curiosity as Canada's strength.

NCAP did not stop at supporting established researchers; it also nurtured the next generation of researchers. The summer school held starting in 2005 served as that channel. This event later became established under the name 'Deep Learning + Reinforcement Learning Summer School.' Every year, graduate students and young researchers from various countries gathered in Canada to learn the latest neural network techniques and build connections with one another. CIFAR

counts that more than 2,700 researchers passed through this school over 20 years. Without demanding immediate results, CIFAR provided long-term support to an unpopular field, and this support became the institutional foundation necessary for deep learning to regain attention in the future.

9.2. Collaboration of the 'Three Godfathers of Deep Learning': Hinton, Bengio, and LeCun

At the center of NCAP were three scholars who would later be called the 'three godfathers of deep learning': Geoffrey Hinton of the University of Toronto, Yoshua Bengio of the University of Montreal, and Yann LeCun of New York University (NYU).

The relationship among the three traces back to the late 1980s. LeCun, who earned his doctorate in Paris, France, came to Toronto in 1987 and worked as a postdoctoral researcher in Hinton's laboratory. Moving to Bell Labs in the United States in 1988, he developed Convolutional Neural Networks (CNNs) that read handwritten digits, research that later led to systems that

automatically read numbers on bank checks. Bengio, after earning his doctorate at McGill University in Canada, conducted research at MIT and Bell Labs in the United States, working alongside LeCun at Bell Labs. He became a professor at the University of Montreal in 1993. In this way, the three were connected as mentor and student, and as colleagues.

After neural networks were pushed aside from academic interest in the mid-1990s, the three became isolated in their respective positions. What brought them together again into a single group from where they were scattered across Toronto, Montreal, and New York was CIFAR's NCAP. Together with dozens of researchers and graduate students, the three gathered at regular workshops, unhesitatingly sharing unpolished ideas and new formulas. When few people shared their perspective, a small community where they could trust one another and exchange ideas was a source of great strength. After Hinton stepped down from the director's position, Bengio and LeCun co-led the program.

Research recognized as key milestones in deep learning emerged one after another from the laboratories of these three individuals. In 2003, Bengio's research team introduced a neural network language model that represented the meaning of words as numerical vectors. In 2006, Hinton's research team published Deep Belief Networks (DBN) and layer-wise pre-training methods. This study demonstrated that even deep neural networks could be successfully trained if learned sequentially layer by layer. In 2014, Bengio's research team introduced the 'attention' mechanism to translation models, and attention later became the foundation for large language models such as ChatGPT. LeCun continuously refined Convolutional Neural Networks, turning them into the foundational architecture of computer vision. The research of the three advanced together, drawing references from one another.

The achievements of the three were recognized through numerous awards. In March 2019, the Association for Computing Machinery (ACM) announced the three as joint recipients of the 2018 Turing Award. The Turing Award is known

as the 'Nobel Prize of computing.' On October 8, 2024, Hinton was announced as a recipient of the Nobel Prize in Physics alongside John Hopfield. In December of the same year, the three received the VinFuture Grand Prize, with a total purse of 3 million dollars, alongside Fei-Fei Li and Jensen Huang. In February 2025, seven individuals—Bengio, Hinton, Hopfield, LeCun, Jensen Huang, Bill Dally, and Fei-Fei Li—received the Queen Elizabeth Prize for Engineering for their contributions to the development of modern machine learning. Among them, Nvidia's Jensen Huang and Dally were included as recipients for the development of GPUs, and Stanford University's Li for creating the image database ImageNet. The roster of laureates demonstrates that the development of deep learning involved contributions not only from algorithms, but also from the people who created computational hardware and data.

Just because the three collaborated did not mean their views were always aligned. Since 2023, the three have voiced differing perspectives on the risks of artificial intelligence. Hinton and Bengio warned that artificial intelligence could escape human control. LeCun viewed these concerns as overstated and has

argued that current large language models alone cannot achieve human-level intelligence.

LeCun's perspective also connects to his career path. Joining Meta in 2013, he founded Facebook AI Research (FAIR), serving 5 of his 12 years there as FAIR's founding director and 7 years as Chief AI Scientist. In a Facebook post in November 2025, he announced that he would leave Meta at the end of 2025. He followed this by stating he would found a startup to create systems that understand the physical world, reason, and plan actions. Even after their views diverged, the three continued public debates. This attitude of rigorously questioning while respecting differing opinions resembles the research culture nurtured in the CIFAR workshops.

9.3. The Cradle of Global AI Researchers: The University of Toronto Laboratory and the 'Toronto Network'

Hinton's influence is not confined to the papers he authored. As students who studied in his University of Toronto laboratory spread to universities and corporations across the world, his ideas took root across a wider landscape. This network of students and colleagues is often referred to as the 'Toronto Network.'

Among his students, Alex Krizhevsky and Ilya Sutskever are widely known. In 2012, the two created a neural network named 'AlexNet' alongside Hinton and entered the ImageNet Large Scale Visual Recognition Challenge (ILSVRC). This competition had neural networks provide up to five answers for each photo and measured the top-5 error rate as the percentage of cases where the correct answer was not among those five. AlexNet's top-5 error rate was 15.3%, while the second-place team recorded 26.2%. The computer vision community was stunned by this substantial gap. The roles of the two students in this

achievement were prominent. Krizhevsky personally wrote the code to rapidly train neural networks using Graphics Processing Units (GPUs). A GPU is originally a component responsible for rendering graphics on screens, capable of processing large volumes of identical types of computations simultaneously. Thanks to this, a neural network with 60 million parameters—numbers determined through learning—could be trained within a realistic timeframe. Sutskever designed various techniques to stabilize training, co-driving the success.

In late 2012, the three founded a company called 'DNNresearch.' Google acquired this company in March 2013, with the acquisition price reported at 44 million dollars. From that point, Hinton worked at Google while also continuing his research at the University of Toronto, before leaving Google in May 2023.

At Google in 2014, Sutskever and his colleagues published the 'Sequence-to-Sequence (Seq2Seq)' model, which transforms one sentence into another. This model significantly elevated the standard of machine translation. In December 2015, he

co-founded OpenAI with Sam Altman and others, serving as its Chief Scientist. After departing OpenAI in May 2024, he founded SSI (Safe Superintelligence) in June of the same year, a company dedicated to building safe superintelligence.

Other students also took on vital roles in their respective fields. Ruslan Salakhutdinov published an autoencoder paper with Hinton in the journal Science in 2006 and co-developed Deep Boltzmann Machines (DBM). He became a professor at Carnegie Mellon University and was appointed Apple's first director of artificial intelligence research in 2016. Between 2009 and 2012, George Dahl and Abdel-rahman Mohamed applied deep neural networks to speech recognition. This work led to a 2012 joint paper authored with researchers from Microsoft, Google, and IBM, transforming speech recognition technology that had long stagnated. Volodymyr Mnih led research at DeepMind on Deep Q-Networks (DQN), a reinforcement learning neural network that autonomously learns video games. Nitish Srivastava, together with Hinton, formulated the 'dropout' technique that mitigates overfitting in neural networks.

Many other researchers passed through Hinton's laboratory. Radford Neal, who researched Bayesian neural networks; Richard Zemel, who later became research director of the Vector Institute; and Brendan Frey and Yee Whye Teh, who researched probabilistic models, were his doctoral students. Peter Dayan and Max Welling, who conducted research bridging neuroscience and machine learning, along with Alex Graves, renowned for his work on handwriting and speech recognition, worked with Hinton as postdoctoral researchers. They dispersed to Toronto, Oxford, London, Amsterdam, New York, and Silicon Valley to establish their own laboratories. In essence, the culture of a single laboratory spread across multiple research hubs worldwide.

9.4. The Vector Institute and Canada's National AI Strategy: From the Periphery to the Center of AI Research

Following AlexNet, deep learning spread across the entire industry. Companies such as Google, Facebook, and Microsoft competed to recruit neural network researchers. Canada, once on the periphery of global artificial intelligence research, gained recognition as the nation that initially fostered deep learning. However, retaining talent remained a challenge. A significant number of researchers who grew in Toronto and Montreal departed for the United States in pursuit of higher salaries.

In March 2017, the Canadian government announced the 'Pan-Canadian Artificial Intelligence Strategy' in its federal budget. The government allocated 125 million Canadian dollars to this strategy and entrusted its design and administration to CIFAR. The Canadian government explains that through this strategy, Canada became the first country to establish a funded national artificial intelligence strategy. In the 2021 budget, an

additional 443 million Canadian dollars was allocated for Phase 2 of the strategy.

At the center of the strategy are three national artificial intelligence institutes: the Vector Institute in Toronto, Mila in Montreal founded by Bengio, and Amii in Edmonton, affiliated with the University of Alberta. Amii has deep ties to Richard Sutton, who received the 2024 Turing Award alongside Andrew Barto for research in reinforcement learning.

Among these, the Vector Institute was officially announced on March 30, 2017. Hinton assumed the role of Chief Scientific Advisor at this institute, and his student Richard Zemel became research director. Alongside federal government support, the institute received investments from the Government of Ontario and more than 30 corporations, including Google, Air Canada, and Canada's five largest banks. Launch funding was reported to range between 150 million and 200 million Canadian dollars depending on news accounts, and is documented in related materials as approximately 175 million Canadian dollars. The researchers who established the institute hoped Vector would

become a magnet for attracting and retaining talent. Their goal was to keep numerous doctoral students in Canada and support artificial intelligence startups.

The Vector Institute formed the three pillars of Canadian artificial intelligence research alongside Mila and Amii. Companies such as Google, Meta, Uber, and Thomson Reuters opened research laboratories in Toronto and Montreal. A trend also emerged in which talent that once drained to the United States either returned to Canada or remained in the country. It was around this time that Toronto earned the nickname 'the mecca of deep learning.'

Canada's investments continued thereafter. The 2024 budget allocated 2.4 billion Canadian dollars to the field of artificial intelligence. Of this, 2 billion Canadian dollars is allocated through the newly created 'AI Compute Access Fund' and 'Canadian Sovereign AI Compute Strategy' to provide computational resources and infrastructure for researchers and startups. With 50 million Canadian dollars, the Canadian AI Safety Institute is established. As large-scale computational

resources came to dictate research competitiveness, infrastructure investment was added to people-centered support.

Hinton holds a definitive view on the manner in which nations should support research. In the aforementioned Seoul Economic Daily interview, he pointed out that it is easy for politicians and policymakers to talk about spending money on specific projects. He cited as an example the 'Fifth Generation Computer Systems project,' which Japan launched with a massive budget in 1982. Large projects with predetermined goals may look plausible on the surface, but they failed to yield the expected results. Hinton stated that an environment supporting researchers to pursue their curiosity freely is effective, and that deep learning was born in precisely that way.

Neural networks were once a hypothesis on the fringes of academia. The force that elevated that hypothesis into mainstream science did not stem solely from the talent of one individual. CIFAR's long-term support that did not rush for outcomes, the collaboration among Hinton, Bengio, and LeCun,

the students who grew in the Toronto laboratory and dispersed across the globe, and the Canadian government's choice to integrate those achievements into a national strategy all worked together. The history of deep learning demonstrates that science is not achieved through individual talent alone, but is a collaborative endeavor built jointly by people and institutions.

Part 4. In Search of Better Learning

Chapter 10. Machines That Recognize Parts and Wholes



Figure 10. Expressing the relationship between parts and wholes that is preserved across changes in viewpoint.

10.1. Limitations of Convolutional Neural Networks (CNNs) and Criticism of Max Pooling

After AlexNet won the image recognition competition by a wide margin in 2012, convolutional neural networks (CNNs) became the standard in computer vision. At the center of this success was Geoffrey Hinton. Yet Hinton did not simply accept the success he had spearheaded. He believed there was a fundamental flaw in the way CNNs see things, and he openly voiced this view for a long time.

To understand the point Hinton took issue with, one must first examine the order in which a CNN processes an image. A CNN moves small filters across the entire image to detect features such as lines, edges, and textures. As it passes through more layers, it detects larger features such as eyes, wheels, and ears. In between these layers, an operation called max pooling is inserted. Max pooling is an operation that retains only the single largest value within a small region and discards the rest. By doing so, even if a feature shifts slightly, the result does not

change significantly. This is called translation invariance. It also reduces computational load.

The price paid is the loss of information. Pose information, such as exactly where a feature was located, what direction it was tilted in, and what size it was, disappears during the pooling stage. In a Q&A session held on the online community Reddit in 2014, Hinton stated that the pooling operation in CNNs was a huge mistake and that the fact that it works well is actually a disaster. He meant that because it works well, researchers do not try to find better approaches.

What happens when pose information disappears can be easily explained with an image of a face. Humans do not merely check whether eyes, a nose, and a mouth are present. They also consider the arrangement in which the parts constitute the whole—whether two eyes are side by side above the nose and the mouth is below the nose. This is called a part-whole hierarchy. As long as the detectors for each part were activated, CNNs tended to classify even an image where the positions of the eyes and mouth were swapped as a face. This was because

they lacked a mechanism to rigorously evaluate spatial relationships between parts.

Changes in viewpoint were also an unresolved challenge. When viewing the same object from above, from the side, or at an angle, pixel values change completely. To endure this variation, CNNs trained on massive quantities of images that were rotated, stretched, and warped. This is called data augmentation. In Hinton's view, this was closer to memorization than to understanding.

Therefore, Hinton reset his goal. Rather than invariance, where the output remains identical even if an object moves, he saw the need for equivariance, where if an object moves, its internal representation changes just as systematically. This means that while recognizing what the object is, the network must also simultaneously recognize where the object is and in what pose.

The roots of this idea trace back to 1979. That year, Hinton published a paper on mental imagery and structural descriptions in the journal Cognitive Science. The paper

features the wireframe cube experiment. When a cube made of wire is stood up with one vertex pointing straight down and the opposite vertex pointing straight up, and people are asked to point to the positions of the remaining vertices with their hands, most people get it wrong. This is because they envision the cube in their minds within a familiar coordinate frame. Through this experiment, Hinton showed that humans assign intrinsic coordinate frames to objects and perceive even the same shape differently depending on that coordinate frame. More than 40 years later, the GLOM paper also cites this experiment right at its beginning.

10.2. Object Coordinate Frames and Transforming Auto-encoders: The Emergence of Capsules

To translate the idea that every object has its own coordinate frame into neural networks, Hinton found a clue in computer graphics. Three-dimensional games and animations assign intrinsic coordinate frames to each object and express the relationship between that object and the camera using matrices. A matrix is a table that arranges numbers horizontally and vertically, allowing transformations such as rotating, translating, or scaling an object to be represented with a single multiplication. When the viewpoint changes, one simply multiplies by a matrix. In pixel space, composed of the brightness of individual pixels, viewpoint changes appear intertwined in complex ways. In pose space, where an object's position, orientation, and size are recorded numerically, the same change becomes a linear calculation consisting solely of multiplications and additions. If graphics generates an image from pose information, visual recognition is the process of

inferring the pose in reverse from the image. Hinton called this inverse graphics.

This concept first appeared as a paper at the International Conference on Artificial Neural Networks (ICANN 2011) held in Espoo, Finland, in 2011. Together with Alex Krizhevsky and Sida D. Wang, Hinton presented "Transforming Auto-encoders." In this paper, the name 'capsule' was used for the first time.

Conventional neurons output a single number. A capsule is a small group of neurons that outputs two types of information together. One is the probability that the entity it is responsible for is present at that location. The other is a set of multiple numbers—that is, a vector—representing the pose, such as the position, orientation, and size of that entity. The paper called these numbers instantiation parameters.

The training method that enables capsules to encode pose is as follows. Along with the original image, the neural network is given shift values (Δx, Δy), such as 'shift right by a certain number of units, shift up by a certain number of units.' The

recognition part of the capsule infers the position (x, y) of the entity from the original image. Adding the shift values yields (x+Δx, y+Δy), and the generative part of the capsule renders the transformed image according to these coordinates. The ground truth is the image that was actually shifted. The neural network trains via backpropagation to reduce the difference between the rendered result and the ground truth. The presence probability output by the capsule is multiplied by the output of the generative part, reducing the capsule's impact on the image if the entity is absent.

The key here is that no one taught the capsule that 'this number represents position.' For a single addition to yield the correct result, the numbers inside the capsule had to behave like actual positions. The neural network learned on its own how to convert information provided as pixels into numbers representing pose.

Capsules also offered a way to connect parts and wholes, addressing the face problem seen in Section 10.1. Suppose a nose capsule and a mouth capsule each determine their

respective poses. There is a fixed spatial relationship between a nose and a face. Multiplying the nose's pose by a matrix representing this relationship allows one to predict the face's pose. The same applies to the mouth. If the two predictions agree with each other, there is a high probability that a face is present at that location. This is because it is rare for predictions composed of multiple numbers to coincide by chance. This agreement check became the central principle of subsequent capsule research.

10.3. Dynamic Routing and EM Routing: The Full-Fledged Development of Capsule Networks (2017–2018)

The capsule concept was refined over 6 years. The culmination appeared in two papers published in 2017 and 2018. The authors were Sara Sabour, Nicholas Frosst, and Hinton from the Google Brain Toronto lab.

"Dynamic Routing Between Capsules," presented at the NIPS conference in December 2017, turned capsules into an actually working neural network. This model is commonly called CapsNet. CapsNet contains two mechanisms.

The first mechanism is the squash function. In the capsule's output vector, the length represents the probability that the entity is present, while the orientation represents its pose. Since it is a probability, the length must be between 0 and 1. The squash function leaves the vector's orientation unchanged and scales its length between 0 and 1. It shrinks short vectors to near 0 and long vectors to near 1.

The second mechanism is dynamic routing, or more precisely, 'routing by agreement.' This translates the agreement check seen in the previous section into a neural network computational procedure. Capsules in a lower layer (parts) multiply their outputs by learned weight matrices to predict the output of each capsule in the layer above (wholes). Initially, predictions are sent evenly to all higher-level capsules. A higher-level capsule pools the received predictions to produce

an output. Next, the dot product is used to measure how closely each prediction points in the same direction as the higher-level output. Next time, more is routed to higher-level capsules that match well, and less is routed to those that do not match. This process is repeated several times. The paper's default setting was 3 iterations.

This approach is the exact opposite of max pooling. Pooling selects a single maximum value and discards the rest. Routing does not discard information; instead, it determines anew for each image which part belongs to which whole. It is a method where a part finds the whole it belongs to among multiple whole candidates.

Experimental results also supported this design. CapsNet achieved an error rate of 0.25% on the handwritten digit dataset MNIST. This was an outcome achieved with a shallow neural network having only 3 layers. The paper also attached a reconstruction mechanism that redraws the original digit solely from the output of the digit capsules. When one of the numbers in a capsule vector was perturbed slightly, the stroke thickness,

skew, and width of the reconstructed digit changed accordingly. This was evidence that the numbers inside the capsule truly captured pose and shape.

The MultiMNIST experiments demonstrated the advantages of capsules even more clearly. This task involves recognizing both digits in an image where two digits heavily overlap. CapsNet outperformed the baseline CNN with an error rate of about 5%, and it even reconstructed each of the two overlapping digits separately. This meant that each digit capsule took only the strokes corresponding to its own digit.

At the ICLR conference in 2018, the follow-up paper "Matrix Capsules with EM Routing" was presented. This time, Hinton was the first author. In this model, a capsule has a single value indicating its presence and a 4×4 pose matrix. The 4×4 matrix is the exact format used in computer graphics to represent the position and rotation of 3-dimensional objects. It brought ideas from graphics even deeper into neural networks.

The routing method also changed. Multiplying the pose matrix of a lower-level capsule by a transformation matrix yields a 'vote' for the pose of a higher-level capsule. The higher-level capsule checks whether multiple votes cluster together in one place. For this decision, the expectation-maximization (EM) algorithm from statistics was used. In the E-step, it calculates the probability that each vote belongs to the cluster of a particular higher-level capsule. In the M-step, it updates the mean, variance, and activation of the higher-level capsule based on those probabilities. When votes cluster tightly together, it concludes that the whole exists.

This model was evaluated on the smallNORB dataset. This is a 3-dimensional object recognition dataset consisting of photos of toy airplanes, cars, trucks, human figures, and animal figures taken from various angles and lighting conditions. The error rate of matrix capsules was 1.8% when evaluated by cropping a single patch from the image, and 1.4% when combining the results from multiple patches. The latter figure represented about 45% fewer errors than the 2.56% of the baseline method used for comparison. It also performed better than a CNN of

comparable size in tests recognizing photos taken from viewpoints unseen during training. It also held up better against adversarial attacks—attacks that fool neural networks with tiny perturbations invisible to the human eye.

In 2019, Adam Kosiorek, Sabour, Yee Whye Teh, and Hinton presented "Stacked Capsule Autoencoders" at NeurIPS. It was an unsupervised learning model that discovers part templates simply by viewing images without ground-truth labels and groups them together to form object capsules.

Although capsule networks attracted substantial interest from academia, their limitations were also distinct. Because routing had to be repeated multiple times, computation was slow, and scaling up to large images or complex datasets proved difficult. Around the same time, as Transformers rapidly became mainstream in language and computer vision, capsules failed to become a mainstream technology.

Nevertheless, research continued. In March 2024, researchers from the University of Aberdeen in the UK, taking as their

starting point the fact that capsule networks struggled to train well on complex data and thus failed to scale to modern tasks, proposed the Masked Capsule Autoencoder (MCAE), the first capsule network to employ self-supervised pretraining. Pretrained on the Imagenette dataset, which consists of 10 categories of ImageNet-sized photographs, it achieved the highest performance among capsule networks and showed a 9% improvement over the baseline model. This research was published in 2025 in the journal Transactions on Machine Learning Research (TMLR).

10.4. A Proposal Unifying Multiple Lines of Research: GLOM, Part-Whole Hierarchies, and Islands of Identical Vectors

Hinton believed that a deeper problem remained in the capsule model than computational speed. Conventional capsules required pre-allocating capsules for every possible type of part and object that could appear at each location in an image. In an ordinary computer, one could simply allocate memory as needed for each image to build a tree structure. Neural networks in the brain cannot do that. What a neuron does is determined by the weights of its incoming and outgoing connections, and biological neurons cannot rapidly and completely alter those weights. The question then becomes how a neural network with a fixed architecture can represent a part-whole tree whose structure differs from image to image. Hinton took this question as his new research agenda.

On February 25, 2021, Hinton posted a 43-page paper titled "How to represent part-whole hierarchies in a neural network"

to arXiv. The name of this concept, GLOM, was drawn from the English slang 'glom together,' meaning to cluster together, which may have originated from 'agglomerate.' In the very first sentence, the paper clarifies that it is not a description of a working system, but rather presents a representation scheme uniting various research breakthroughs—including Transformers, neural fields, contrastive representation learning, knowledge distillation, and capsules—along with an imagined system, GLOM, that embodies it. Hinton introduced the same concept in lectures, including one at the Perimeter Institute for Theoretical Physics in Canada on May 5, 2021.

The architecture of GLOM is as follows. The image is divided into small patches, and one 'column' is placed for each patch. All columns share the same weights. Each column is divided into multiple levels. Stacked from the bottom up are sub-parts, parts, objects, and scenes. At each level, there is a vector—that is, an embedding—representing what is present at that location. Each of these vectors acts as a universal capsule unconstrained by specific types. In this way, it relieves the

burden of previous capsule models that had to prepare capsules in advance for every category.

When an image is input, GLOM updates each vector over multiple time steps. A vector is an ordered collection of multiple numbers, here representing what is present at a specific location in the image. At each step, the vector at a given level is replaced by a weighted average of four values. A weighted average is an average calculated by assigning different weights to each value. The first value is the prediction obtained by passing the vector from the level immediately below through a bottom-up neural network. The second value is the prediction obtained by passing the vector from the level immediately above through a top-down neural network. The top-down neural network also receives the column's position as input, because knowing the identity 'face' and the position allows it to compute whether a nose or an eye should be located there. This is an idea borrowed from neural field research. The third value is the vector's own previous value at that level. The fourth value is the attention-weighted average of vectors at the same level from neighboring columns, giving greater weight to neighbors

with vectors similar to its own. While resembling Transformer attention, it is a simplified form that directly compares vectors with one another without separately transforming them into queries, keys, and values.

Repeating this process produces GLOM's core phenomenon. The columns of all patches corresponding to a face converge to the exact same vector at the object level. Patches corresponding to the nose share an identical vector at the part level, while patches corresponding to the mouth share another identical vector. Hinton called this 'islands of identical vectors.' A single island corresponds to a single node in a tree structure. When multiple smaller islands at the part level lie beneath a single large island at the object level, that represents the part-whole relationship between the face, nose, and mouth. A different tree can be represented for each image without allocating new neurons. This is Hinton's answer to the question of how a neural network with a fixed architecture can represent a structure that differs from image to image.

As a learning method, Hinton proposed reconstructing partially occluded images. He also added contrastive learning regularization so that vectors at locations belonging to the same object become similar to one another. Once islands are properly formed, it becomes easier for humans to read out what the neural network saw and how it saw it. The paper's abstract also noted that if GLOM works, it will dramatically improve the interpretability of representations produced by Transformer-like systems used in vision or language.

Hinton viewed GLOM as bridging two traditions in perception research. One is Gestalt psychology, which views perception as a single field and posits that the whole is different from the sum of its parts. The other is classical symbolic artificial intelligence, which sought to represent perception through symbols and structural descriptions. In GLOM, perception appears as a continuous field of vectors, yet within the structure formed by the islands, explicit part-whole relationships are embedded.

There are no widely known instances of GLOM being implemented as a completed large-scale system yet. Hinton

himself shifted his focus to learning rules themselves, announcing the Forward-Forward algorithm in late 2022. Even so, the question raised by GLOM remains valid. It is the problem of how a fixed neural network can represent different structures for different images, and whether those representations can be made readable by humans. Hinton's quest—starting from the wireframe cube experiment in 1979, passing through capsules in 2011, dynamic routing in 2017, and matrix capsules in 2018, to GLOM in 2021—was entirely directed toward this question.

Chapter 11. Learning Methods Rediscovered in the Brain

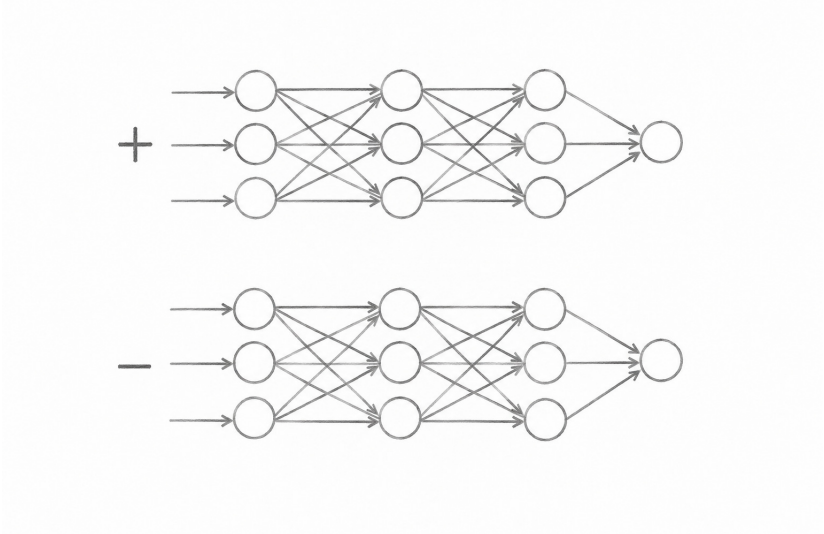


Figure 11. Illustrating learning by passing positive and negative examples forward.

11.1. Does the Brain Really Learn by Backpropagation?

In 1986, David Rumelhart, Geoffrey Hinton, and Ronald Williams published their paper on error backpropagation in Nature. The procedure of backpropagation is as follows. When a neural network produces an answer, it first calculates the difference from the correct answer, that is, the error. Next, using the chain rule of calculus, it propagates this error backward from the output layer toward the input layer, calculating how much each connection weight needs to be adjusted. For nearly 40 years since, this method has served as the core technology underpinning deep learning. Most achievements, ranging from image recognition and machine translation to large language models (LLMs), have been built on backpropagation.

Yet for a long time, Hinton did not let go of one question: whether the human cerebral cortex also learns in this manner. His answer was largely negative. In 2020, together with Timothy Lillicrap and others, Hinton published "Backpropagation and the brain" in Nature Reviews Neuroscience. Looking at this

paper together with Hinton's 2022 research, the reasons why backpropagation is unlikely to occur as-is in the brain can be summarized into three points.

The first is the Weight Transport Problem. To send errors backward in backpropagation, a transposed matrix—the flipped version of the weight matrix used in forward computation—is required. This means that the backward pathway must be paired with the forward pathway with exactly the same strength. However, in the brain's actual neural connections, no such one-to-one symmetric structure has been identified.

The second is that the brain lacks separate pathways dedicated solely to sending error signals and mechanisms to store previous states. Backpropagation requires transmitting error derivatives, distinct from normal signals, through a separate pathway, and performing backward computation requires remembering the activation values from the forward computation. Feedback connections in the brain do not operate this way. Feedback merely modulates the activity strength itself of the neurons receiving the signal, and there is as yet no

neuroscientific evidence that it carries separate signed error values.

The third is the issue of time. To learn from continuously incoming data such as video, one must use backpropagation through time (BPTT). Because this method must propagate errors back to past points in time, input processing must frequently be paused. In contrast, the brain learns simultaneously while streaming incoming signals from the eyes and ears without stopping. Hinton explained this using the term pipelining.

Hinton's raising of these issues should not be interpreted as an argument to abandon backpropagation. In various interviews, he explained that his questions stemmed from the perspective of seeking to understand the brain. In artificial intelligence running on digital computers, backpropagation still outperforms other learning methods. His thinking was simply that if the brain does not use backpropagation, there must be another method the brain uses, and that method could provide a clue to building a new kind of computer.

Hinton also explained the difference between the brain and artificial neural networks in terms of the ratio of data to the number of connections. Large language models have around 1 trillion parameters, while the text used for training reaches several trillion tokens. This is a structure where experience is abundant and storage space is relatively scarce. Backpropagation excels at compressing such vast experience into a small number of parameters. Humans are the exact opposite. Even if one lives for 80 years, a lifetime amounts to only about 2.5 billion seconds, whereas synaptic connections in the brain reach about 100 trillion. It is a structure where experience is scarce and connections are abundant. Hinton believed that under these conditions, the brain would employ a learning method different from backpropagation.

11.2. The Forward-Forward Algorithm: Two Forward Computations and a Score Called 'Goodness'

In December 2022, Hinton introduced a new learning method during a keynote address at the Neural Information Processing Systems (NeurIPS) conference held in New Orleans, United States. In the same month, "The Forward-Forward Algorithm: Some Preliminary Investigations" (arXiv:2212.13345) was uploaded to the preprint website arXiv. At the time, he was with Google Brain Toronto. As can be seen from his use of the phrase 'preliminary investigations' in the title, Hinton presented this method not as a finished result, but as a starting point.

The core idea is simple. Backpropagation computes once from input to output, and once from output to input. The Forward-Forward (FF) algorithm eliminates backward computation and instead performs forward computation in the identical manner twice. One computation is performed with real data, and the other with fake data. Fake data resembles real data but is intentionally made incorrect. The step of

inputting real data is called the positive pass, and the step of inputting fake data is called the negative pass.

This idea had precedents in earlier research. In the 1980s, Hinton worked on Boltzmann Machines with Terry Sejnowski. The Boltzmann Machine learned by comparing a phase of observing data with a phase of observing states generated by the model itself. Noise Contrastive Estimation (NCE), proposed in 2010 by Michael Gutmann and Aapo Hyvärinen, also trains models by distinguishing real data from noise. FF reformulated the principle of contrast common to these two lines of research—namely, the principle of comparing two types of inputs and learning from their difference—to suit modern multilayer neural networks.

The criterion each layer uses to distinguish between real and fake is a score calculated separately for each layer. Hinton called this score goodness. Goodness is the sum of the squared activation values of the neurons in that layer. The activation value is the magnitude of the output produced by a neuron upon receiving input, and ReLU is used as the activation

function. ReLU is a function that outputs 0 when the input is less than 0, and outputs the input as is when the input is greater than 0. Therefore, the more strongly the neurons respond, the greater that layer's goodness becomes.

The goal of learning is twofold. When real data comes in, the weights are adjusted so that goodness becomes sufficiently larger than a predetermined threshold; when fake data comes in, the weights are adjusted so that goodness becomes sufficiently smaller than the threshold. Based on goodness, each layer computes the probability that the input is real as follows.

First, the threshold is subtracted from that layer's goodness. The value obtained by feeding this difference into a sigmoid function is the probability that the input is real.

The sigmoid function converts any value into a value between 0 and 1. If goodness is greater than the threshold, the probability approaches 1; if smaller, it approaches 0. Each layer adjusts only its own weights so that this probability comes out high for

real data and low for fake data. There is no need to receive errors from other layers. In this way, the method in which each layer learns using only information within its own layer is FF's local learning.

However, stacking multiple layers creates a problem. Once the first layer learns to distinguish well between real and fake, its output vector becomes long for real data and short for fake data. If the second layer were to make judgments based solely on this length, it would have nothing new to learn. Hinton resolved this problem with layer normalization. Layer normalization is a technique published in 2016 jointly by Jimmy Ba, Jamie Ryan Kiros, and Hinton. By applying this technique, FF eliminates the length of the vector before passing it to the next layer, leaving only its direction. The direction indicates the relative proportions in which the neurons are active. As a result, the upper layer cannot rely on the information already utilized by the lower layer, and must uncover new features hidden within the activity patterns.

The remaining question is how to create fake data. FF can be applied to both unsupervised learning without ground-truth labels and supervised learning with labels, and the methods for creating fake data differ in the two cases. In unsupervised learning, Hinton created fake images by blending two handwritten digit images. By blurring an image of random noise multiple times, he obtained a mask shaped like large patches, and then merged the area inside the mask from one image with the area outside the mask from the other image. An image created this way resembles real ones in small components like short strokes, but its overall shape is not a digit. By distinguishing between real images and these fake images, the network learns how to recognize large-scale shapes.

In supervised learning, labels are placed directly inside the images. This is done by representing a digit across the first 10 pixel locations of an MNIST handwritten digit image. Attaching the matching label to an image produces positive data, whereas attaching an incorrect label produces negative data. The two data samples share the same image and differ only in the label. Thus, the network ignores backgrounds unrelated to the label

and focuses on the relationship between the image and the label. When testing after training is complete, forward computation is performed 10 times by attaching each label from 0 to 9 one by one to a new image. Then, the label with the largest accumulated sum of goodness across multiple layers is selected as the answer.

There are also benefits gained from eliminating backward computation. While backpropagation must store previously computed values to send errors backward, FF has no need to do so. Even with continuously incoming data like video, each layer can adjust its weights on the fly while streaming the data without pause. This circumvents the third problem discussed in Section 11.1. Furthermore, even if an opaque device—a so-called black box—is interposed between layers, FF can continue learning. This is because backpropagation requires every step to be differentiable, whereas each layer in FF needs to know only its own inputs and outputs.

Performance is reported in Hinton's paper. Supervised FF, with training data augmented by slightly shifting images, recorded

an error rate of 0.64% on the MNIST test data. This is on par with convolutional neural networks trained with backpropagation. On CIFAR-10 photographic data as well, it achieved results slightly lower than backpropagation, yet not far behind.

11.3. The Sleep Hypothesis, Mortal Computation, and Remaining Limitations

For FF to be considered a learning model of the brain, there is an issue that must be addressed. It is difficult to conceive that the brain alternates between processing real and fake data every time it receives sensory signals. In response to this problem, Hinton proposed the sleep hypothesis. While awake, the brain executes only positive passes using real sensory data coming from the outside world. Then, during sleep—especially during REM sleep, when dreaming is frequent—a generative model within the brain creates fake data on its own to conduct negative passes. If this hypothesis holds true, dreams may be negative data created by the brain for the purpose of learning.

At this juncture, Hinton revealed an error of his own. In the draft of his paper, he wrote that performing thousands of positive learning steps followed by thousands of negative learning steps—effectively separating the awake phase from the sleep phase—worked well. Later, however, he corrected this in the paper, explaining that this result had arisen due to an error (a bug) in the program code and was not reproducible. To train with the two phases widely separated using the current goodness function, one must set the learning rate—the step size for adjusting weights at one time—extremely low, and momentum—the extent to which the preceding direction of change is sustained—extremely high. Hinton left completely separating the positive and negative phases in time as an important challenge to be tackled in the future.

The field in which FF carries major significance is hardware. Today's digital computers perform exact computations by switching transistors on and off. Accurately obtaining the derivatives for backpropagation consumes substantial electrical power and requires continuously shuttling data back and forth between memory and processing units. Analog

hardware is different. If neuron activity is represented by voltage and connection weights by conductance (a measure of how easily electricity flows), the product of voltage and conductance becomes electrical charge, and multiple charges add together naturally. Multiplication and addition are thus completed purely through the physical phenomena of Ohm's law and Kirchhoff's laws. Doing so can drastically reduce the energy required for the same computation. The human brain operates on approximately 20 watts of power.

The problem is that analog devices differ slightly from one another. Even when produced in the same factory, each device possesses slightly mismatched characteristics. In such devices, the exact derivative calculations required for backpropagation are difficult to perform. In contrast, because each layer in FF learns using only its own inputs and outputs, it can learn without knowing the exact properties of the devices.

Here, Hinton introduced the concept of Mortal Computation. In digital computers, software and hardware are decoupled. Trained weights can be copied directly to another computer,

and even if hardware fails, knowledge survives. Hinton called this Immortal Computation. In analog hardware, the situation is different. Because weights are trained to fit the unique characteristics of that specific machine, transferring those weights to another machine will not work properly. When a machine reaches the end of its lifespan, the knowledge accumulated within it perishes as well. The human brain is the same. The synaptic connections a person has built over a lifetime cannot be copied into someone else's brain. Humans can only transmit knowledge little by little through the slow channels of speech and writing.

This insight led to Hinton's warnings about the dangers of artificial intelligence. He took note of the fact that digital intelligence can undergo different experiences across thousands of copies of the same model, average the weight updates, and share them all at once. This is something humans cannot do among themselves. After leaving Google in May 2023, Hinton publicly warned that these strengths of digital intelligence could pose a threat to humanity. Research seeking learning methods for analog computers had thus become an

impetus to reevaluate the capabilities of digital intelligence. In 2024, he was awarded the Nobel Prize in Physics alongside John Hopfield.

The limitations of FF are also clear. First, it trains more slowly than backpropagation to reach the same level of performance. Second, on large models with hundreds of billions of parameters or on large-scale datasets like ImageNet, it fails to generalize as well as backpropagation. Hinton himself anticipated that backpropagation would remain mainstream for the time being on large digital computers where power consumption is not an issue. Third, it remains incomplete as a biological model. Learning with separated sleep phases has not yet succeeded, nor can it be said that FF fully explains the intricate feedback circuitry of the cerebral cortex.

After the paper was published, multiple researchers set out to address the shortcomings of FF. The primary task was creating usable negative data. According to an evaluation by French researchers, FF did not achieve state-of-the-art performance on most standard benchmarks, one of the reasons being that the

method of creating negative data in unsupervised learning was unreliable. To solve this problem, Xing Chen, Dongshu Liu, Jérémie Laydevant, and Julie Grollier released Self-Contrastive Forward-Forward (SCFF) on arXiv in September 2024. Drawing inspiration from self-supervised contrastive learning, SCFF generates both positive and negative inputs from within a single piece of data. As a result, it recorded unsupervised classification accuracies of 98.7% on MNIST, 80.75% on CIFAR-10, and 77.3% on STL-10, outperforming existing forward-only local learning methods. The researchers also expanded FF's scope of application to recurrent neural network training, enabling it to process sequential data as well. This paper was published in 2025 in volume 16 of Nature Communications.

Research also continued in the hardware field. Studies emerged including training with FF on microcontrollers and on-chip learning with FF inside vertical NAND flash memory. FF has become a benchmark in research on direct on-chip learning technologies and neuromorphic computing.

The Forward-Forward algorithm did not replace backpropagation. Nor did Hinton make such a claim. What this work left behind was raising anew the question of whether backward computation is strictly necessary for learning. The fact that the researcher who popularized backpropagation explored outside that method 36 years later demonstrates that an interest in the brain remained the starting point of Hinton's research.

Chapter 12. How to Learn Without an Answer Key

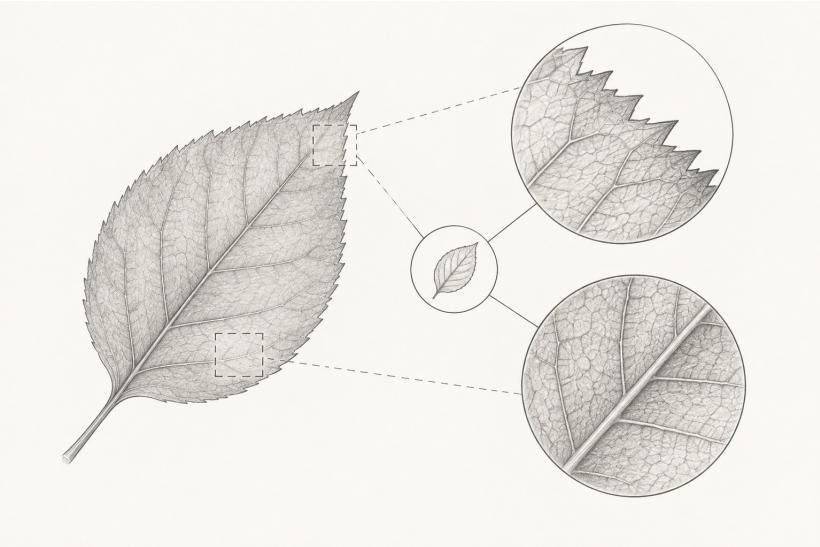


Figure 12. Depicts the process of learning commonalities from different views of the same object.

12.1. Learning Without Labels: The Challenge of Self-Supervised Learning

Following AlexNet's landslide victory in the image recognition competition in 2012, computer vision research revolved around supervised learning. Supervised learning is a method where humans provide ground-truth tags—labels—such as "cat," "car," or "apple" for every image, and train the neural network to predict those correct answers. The performance of this method depends heavily on the volume of labeled data available. ImageNet, a prominent dataset, contains approximately 1.28 million training images spanning 1,000 categories, and manually labeling each image required substantial labor and time. In specialized domains requiring expert interpretation, such as medical imaging or satellite photos, labeling costs escalate dramatically.

Human learning about the world operates differently. Even without anyone providing a name for every picture, young children continuously observe the scenes unfolding before them, intuitively learning the shapes of objects, what is alike

and what is different, and the continuity of space. Researchers refer to this method of extracting useful features from the intrinsic structure and regularities of data itself, without human-annotated labels, as self-supervised learning, and have long regarded it as a crucial milestone for machine learning to achieve.

Self-supervised learning has largely progressed along two trajectories, both of which faced limitations. The first is the generative approach. This method trains models to reconstruct input images pixel by pixel, exemplified by autoencoders, variational autoencoders (VAEs), and generative adversarial networks (GANs). This approach demanded heavy computation and had the drawback of wasting network capacity on reproducing minute textures or noise irrelevant to distinguishing objects. The second is the discriminative approach. Here, researchers design artificial pretext tasks for neural networks to solve, such as rotating an image and predicting its angle of rotation, shuffling patches like a jigsaw puzzle to restore their original positions, or colorizing grayscale photos. Because this approach relies heavily on human-crafted

heuristics, it was constrained in acquiring general representations that transfer across multiple domains. In this context, a representation refers to a vector of numbers in which a neural network captures the essential features of an input.

Geoffrey Hinton's ideas regarding the problem of learning representations without labels trace back to 1992. That year, together with Suzanna Becker, Hinton published a study in the journal Nature on a neural network that independently discovers the depth structure of surfaces from random-dot stereograms. The crux of this research was the notion that even if two visual patches are spatially separated or appear slightly different, their internal representations within the neural network must agree if they originate from the same object. This concept—using agreement between two views derived from the same source as the learning signal rather than ground-truth labels—became the foundation of contrastive learning roughly 30 years later.

In February 2020, Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton of the Google Brain Toronto

research team published a paper titled "A Simple Framework for Contrastive Learning of Visual Representations." The paper was presented at the International Conference on Machine Learning (ICML 2020), held virtually in July of that year, and the method became widely known as SimCLR. The word "Simple" in the paper's title directly reflects the nature of this research. Without relying on complex specialized architectures or auxiliary memory mechanisms, the researchers attained state-of-the-art self-supervised learning performance at the time using only standard components.

12.2. Data Augmentation and Contrastive Loss: How SimCLR Learns

The core intuition behind SimCLR can be described simply. When a single image is transformed in two different ways, its surface appearance changes, but its underlying content remains identical. The neural network is trained to recognize that these two transformed versions "originated from the same image" and to distinguish them from transformations originating from different images. Because it learns to place similar instances close together and push different instances far apart, this method is called contrastive learning.

To realize this idea, SimCLR chains together four components in sequence. First is a stochastic data augmentation module. Applying two transformations randomly sampled from the same augmentation family to a single unlabeled image x yields two augmented views. These two views are termed a positive pair. Second is a base encoder f. This neural network extracts features from transformed images, with ResNet-50 serving as the default in the paper. Passing through the final average

pooling layer yields a representation vector h consisting of 2,048 values. Third is a non-linear projection head g. A compact neural network consisting of a single hidden layer and a ReLU activation, it maps h to a vector z in the space where contrastive loss is computed. Fourth is a contrastive loss function, which trains the entire neural network to draw positive pairs closer together while repelling representations from other images within the same batch.

A key finding uncovered through the researchers' extensive experimentation was that the first component—namely, which transformations are combined and how—determines the success or failure of learning. Relying solely on random cropping, where an image is cropped at random and resized, introduces problems. Because two patches cropped from the same image share similar color distributions, the neural network takes a shortcut, matching pairs based solely on color statistics without learning object shapes or structures. This resembles identifying exams belonging to the same student merely by looking at handwriting style. While the network pairs

them successfully, representations learned in this manner have little practical utility for recognizing actual objects.

The researchers demonstrated that this shortcut can only be eliminated by pairing random cropping with aggressive color distortion. Color distortion randomly alters brightness, contrast, saturation, and hue, occasionally stripping color entirely to make the image grayscale. When Gaussian blur is additionally applied to blur the image, the neural network cannot rely on color cues and is compelled to master the inherent shapes and semantic structures of objects. The researchers also confirmed that while such strong augmentation offers minimal benefit to supervised learning, it dramatically enhances the performance of contrastive learning. This signifies that contrastive learning requires substantially stronger data augmentation than supervised learning does.

How effectively these generated views are distinguished is evaluated using contrastive loss. Loss is a metric indicating how far a neural network's predictions deviate from the objective, where lower values are preferable. SimCLR employs a loss

termed NT-Xent (Normalized Temperature-scaled Cross Entropy). First, cosine similarity measures the extent to which two vectors point in the same direction. Cosine similarity is calculated by dividing the dot product of corresponding components of two vectors by the product of their lengths, comparing orientation irrespective of vector magnitude. A value close to 1 indicates the two vectors point in virtually identical directions, whereas a value close to 0 denotes mutually unrelated directions. Next, comparison targets are established. A collection of images trained together in a single step is known as a minibatch; generating two augmented views for each image in a minibatch produces a total view count twice the number of images. Taking one view as the reference, the other view originating from the same image forms a positive pair, while all remaining views excluding the reference and its counterpart serve as negative examples. The count of negative examples equals twice the number of images minus 1. For example, with 100 images, there are 200 augmented views and 198 negative examples. The loss for a single positive pair is computed as follows.

First, the cosine similarity between the reference view and its paired counterpart is divided by a tuning parameter called temperature and passed into an exponential function. The identical calculation is performed for all other views excluding the reference itself, and those values are summed together. Dividing the former value by the latter sum yields a ratio between 0 and 1, and the negative logarithm of this ratio constitutes the loss. As the ratio approaches 1, the loss approaches 0.

In this division, the numerator reflects the similarity with the paired view, while the denominator sums the similarities with all other instances excluding the reference itself. The more distinctly the similarity with the paired view exceeds similarities with other images, the lower the loss becomes. This mirrors the structure of solving a multiple-choice question where one correct option must be selected from several alternatives.

The tau in the expression denotes the temperature parameter. Setting tau to a small value, such as 0.1, amplifies minor differences in similarity. This causes the neural network to

allocate stronger learning signals toward hard negatives that are easily confused with the paired view. Normalizing vector lengths to 1 in tandem with an appropriate temperature forms a mechanism that automatically assigns greater importance to challenging problems.

12.3. The Projection Head: A Component Used in Training and Discarded in Deployment

Another discovery in SimCLR pertains to the third component: the projection head. Rather than applying contrastive loss directly to the representations extracted by the encoder, passing those representations through another small neural network and calculating loss on its output yields superior performance. This small neural network is the projection head. The projection head multiplies the encoder's representation by a first weight matrix, applies a ReLU function to the outcome, and multiplies by a second weight matrix to generate the final output. Under default settings, the encoder's 2,048-dimensional representation is mapped to a 128-dimensional output through the projection head.

The experimental findings were unambiguous. Employing a non-linear projection head improved performance by more than 10 percentage points compared to omitting a head, and by more than 3 percentage points compared to a linear projection

head. Intriguingly, once training concludes, this projection head is discarded. What is utilized for object classification or downstream tasks is not the projection head's output, but the encoder representation situated prior to the head. A component indispensable for computing the loss is effectively discarded during actual deployment.

The researchers attributed this to the information-discarding nature of contrastive loss. Contrastive loss forcefully compels two augmented views of the same image to become identical in z-space. Consequently, z grows insensitive to transformation details, such as where an image was cropped, how it was rotated, or how its colors were altered. Yet information regarding object orientation, color, and spatial positioning is immensely valuable for other tasks such as object detection or semantic segmentation. Interposing a projection head absorbs the pressure to erase transformation details primarily within z. As a result, both transformation details and semantic information remain preserved within h, situated upstream of the head.

The researchers substantiated this interpretation empirically. They attached small neural networks to h and z individually, tasking them with predicting which transformations had been applied to images during pre-training. Whether an image was color or grayscale was predicted with 99.3% accuracy from h and 97.4% from z. Conversely, predicting which of four orientations an image had been rotated toward achieved 67.6% accuracy from h, whereas from z it plummeted to 25.6%, essentially random guessing among four choices. In distinguishing between an original image and one processed with a Sobel filter, h reached 96.6% while z attained 56.3%. The explanation that the projection head assumes the role of discarding information, thereby leaving richer representations upstream, was thus backed by concrete figures.

12.4. Large Batches, Three Evaluations, and Subsequent Research

Contrastive learning benefits from having a larger pool of negative samples for comparison. Earlier studies therefore maintained separate memory banks to accumulate representations from prior training steps. MoCo, introduced in late 2019 by Kaiming He's research team, maintained comparison candidates by combining a slowly updating encoder with a queue that sequentially stored previous representations. SimCLR bypassed dedicated memory architectures, opting instead to scale up the minibatch itself. The researchers experimented with scaling batch sizes from 256 to 8,192. At a batch size of 8,192, each transformed view of an image is compared against 16,382 negative samples within the identical batch. Larger batches and extended training consistently elevated representation quality.

Scaling batches to this magnitude required resolving complications that surfaced during training. First, conventional stochastic gradient descent (SGD) becomes unstable under

large batch sizes. The researchers utilized the Layer-wise Adaptive Rate Scaling (LARS) optimization algorithm, which adapts learning rates separately across layers. They also incorporated a linear warmup that gradually raises the learning rate across the first 10 epochs, paired with a cosine decay schedule thereafter. Training was executed on Google Cloud TPU v3 using between 32 and 128 cores. Here, an additional complication arose: distributing data across multiple cores causes each core to compute batch normalization statistics solely from its own shard, which can leak information that makes positive pairs trivial to identify. The researchers circumvented this issue using global batch normalization, which aggregates mean and variance across all cores.

The effectiveness of representations learned in this manner was validated across three evaluations. The first was linear evaluation. Freezing the pre-trained encoder, a single linear classifier was mounted on top to measure ImageNet classification performance. SimCLR trained on a base ResNet-50 recorded a Top-1 accuracy of 69.3%. The previous benchmark of 71.5% set by CPC v2 relied on a larger ResNet-161

architecture. When trained with a 4x wider ResNet-50(4x), SimCLR achieved 76.5%. This marked a 7% reduction in relative error compared to the previous best self-supervised model, matching the performance of a standard ResNet-50 trained with full supervised labels. Representations learned without labels had matched those learned with labels.

The second was evaluation with limited labels. When fine-tuning the entire neural network using only 1% of ImageNet labels—amounting to roughly 12.8 labeled images per category—SimCLR ResNet-50(4x) attained a Top-5 accuracy of 85.8%. This substantially outstripped the prior record of 77.9% by CPC v2, outperforming the 2012 AlexNet trained on 100% of labels while utilizing only one-hundredth of the labels. Expanding labeled data to 10% lifted Top-5 accuracy to 92.6%.

The third was transfer learning evaluation. This assesses how effectively representations acquired from ImageNet transfer to other datasets. The researchers tested across 12 natural image datasets, including CIFAR-10, CIFAR-100, Food-101, Caltech-101, and Flowers-102. When fine-tuned, SimCLR matched or

surpassed supervised models across 10 of these datasets. Fine-tuned accuracies reached 98.6% on CIFAR-10, 89.0% on CIFAR-100, and 89.4% on Food-101.

SimCLR was not without limitations. Requiring massive batches in the thousands and dozens to over a hundred TPU cores made direct replication difficult for research labs with modest computing resources. Another unresolved issue was that different images belonging to the same category—such as two separate photos of cats—were treated as negative samples and mutually repelled.

Subsequent research evolved toward mitigating these challenges and broadening the scope of contrastive learning. At NeurIPS 2020, the same research team presented SimCLR v2, combining larger architectures with knowledge distillation. In 2021, OpenAI's CLIP broadened the contrastive learning paradigm to pairs of images and text, while that same year Meta's DINO advanced self-supervised learning by aligning teacher and student neural networks without requiring negative samples.

This momentum extends to the present day. In August 2025, Meta introduced DINOv3. Meta reported that it scaled label-free learning to a 7-billion-parameter model and a 1.7-billion-image dataset, noting that it achieved state-of-the-art performance across multiple domains even while keeping the backbone frozen during evaluation. Furthermore, it demonstrated that models trained through self-supervised learning outperformed weakly supervised models for the first time across a diverse range of benchmarks, including fine-grained image classification, semantic segmentation, and video object tracking. The DINOv3 paper highlights that these techniques are finding adoption in fields where labels are difficult to obtain, such as histopathology, biology, medical imaging, remote sensing, and astronomy. While DINOv3 adopts a distinct training framework from SimCLR, it shares the fundamental foundation of learning representations by aligning multiple augmented views of the same image without labels.

SimCLR was a seminal study that resurrected Becker and Hinton's 1992 premise—that two perspectives originating from the same object should agree—through deep neural networks

and large-scale computation. Relying on merely three components—a combination of data augmentations, a projection head, and large batch sizes—it demonstrated that unsupervised learning could stand shoulder to shoulder with supervised learning, providing the catalyst for the rapid expansion of subsequent self-supervised learning research. To the lifelong question Hinton pursued—whether machines can master the structure of the world on their own without being spoon-fed every answer—SimCLR delivered a concrete answer in the visual realm.

Part 5. Achievement and Responsibility

Chapter 13. The Nobel Prize in Physics and Neural Networks



Figure 13. Depicting the intersection of physics and neural network research.

13.1. The Announcement on October 8, 2024: Hopfield and Hinton Win the Nobel Prize in Physics

On October 8, 2024, the Royal Swedish Academy of Sciences in Stockholm, Sweden, announced two recipients for that year's Nobel Prize in Physics. They were John J. Hopfield of Princeton University in the United States and Geoffrey E. Hinton of the University of Toronto in Canada. The citation stated by the Academy was "for foundational discoveries and inventions that enable machine learning with artificial neural networks." At the time of the announcement, Hopfield was 91 years old and Hinton was 76, and the two were to share the prize money of 11 million Swedish kronor equally.

This announcement surprised both the physics and computer science communities. For a long time, the Nobel Prize in Physics had been awarded to research that revealed natural phenomena, such as the structure of the universe, the properties of particles, and the states of matter. This time, however, scholars who laid the foundation of artificial

intelligence—the method by which computers observe data and learn on their own—received the award. This was the first time in the history of the Nobel Prize that artificial intelligence research stood at the center of the physics prize.

It was an outcome unexpected even by the laureates themselves. Hinton received the news over the phone at a hotel in California, USA. Right after the announcement, he joined a press conference held by the University of Toronto virtually, expressing that he was still in disbelief, and in an interview with the American business magazine Fortune, he said that because he was not a physicist, he had not imagined he would receive this award. In fact, Hinton studied experimental psychology at the University of Cambridge in the United Kingdom and received his doctorate in artificial intelligence from the University of Edinburgh. There was no department of physics in his academic career.

To understand why a researcher who did not major in physics came to receive the Nobel Prize in Physics, one must go back to the early 1980s. The starting point is Hopfield's research.

Hopfield was originally a physicist who studied solid-state physics. In a paper published in the Proceedings of the National Academy of Sciences (PNAS) in 1982, he applied the principle of how atoms inside a magnet influence one another to neural networks.

Atoms inside a magnet possess 'spin,' which points up or down like small compasses. Neighboring spins tend to point in the same or opposite direction as each other, and the overall system settles into its lowest energy state. Hopfield viewed each neuron as a spin. In this network, where neurons are symmetrically connected to one another, each stored pattern becomes a valley in an energy landscape. Just as placing a ball on a hillside causes it to roll down to the nearest valley, even if an image that is partially erased or blurred is input, the network returns to the nearest valley—namely, the originally stored image. It is similar to how a person hears just the first measure of a song and recalls the rest. This function is called 'associative memory,' and this network earned the name 'Hopfield network.'

However, the Hopfield network had a weakness. It happened that the ball could not reach the deepest valley and became trapped in a shallow puddle along the way. Hinton, along with his colleague Terrence Sejnowski, solved this problem. What the two brought in was the statistical mechanics of 19th-century Austrian physicist Ludwig Boltzmann. When the temperature is high, particles shake wildly, and when the temperature drops, they settle into a stable state. Boltzmann formulated an equation that calculates the probability of each state occurring based on energy and temperature. Hinton made neurons turn on and off according to this probability. At first, by raising the temperature, the ball was made to bounce out of shallow puddles, and by gradually lowering the temperature, it was made to settle into a deep valley. This is the same principle as annealing, in which metal is heated and then cooled slowly to make it hard.

The model that emerged between 1983 and 1985 in this way was the 'Boltzmann machine.' Hinton introduced 'hidden neurons' that were not outwardly visible. Hidden neurons autonomously discover rules hidden within the input data.

Thus, the Boltzmann machine was not only able to classify photographs, but could also generate new examples according to the rules it had learned. It was, in effect, a distant ancestor of today's generative artificial intelligence that creates images and text. However, computers of that era were too slow at computation, making practical application difficult in the field. Hinton subsequently developed the 'Restricted Boltzmann Machine,' which reduced the connectivity structure, and in 2006 introduced a method for training deep neural networks by stacking them in multiple layers. This breakthrough served as the starting point for the deep learning era.

In this way, the research of both individuals originated from concepts in physics, and its results are now being used back in physics research. Ellen Moons, Chair of the Nobel Committee for Physics, explained at the announcement that their work had already conferred great benefits on humanity. She stated that physicists are using artificial neural networks across various fields, including the development of new materials with specific properties. Analyzing the vast experimental data pouring out of

particle accelerators is also difficult to imagine today without artificial neural networks.

At the press conference, Hinton repeatedly stated that this achievement was not research aimed at immediate profit, but the fruit of fundamental research that began with curiosity. In the same venue, he also mentioned the dangers that could arise when artificial intelligence becomes smarter than humans. He used the occasion of announcing his Nobel Prize win as an opportunity to issue a warning as well. The award ceremony was held in Stockholm on December 10 of that year. Hinton stated that he would donate a portion of his prize money to 'Water First,' an organization that addresses clean water issues in Canadian Indigenous communities.

13.2. "Is This Physics?": Debates in the Physics Community and the Boundaries of Disciplines

As soon as the announcement was made, objections immediately arose within the physics community. The criticisms were diverse, but they converged on a single question: is research on artificial neural networks truly physics?

A widely known critique came from German theoretical physicist Sabine Hossenfelder. On her YouTube channel, she argued that the fact that the Nobel Prize in Physics went to computer scientists demonstrated a crisis in physics. She added that because neural networks had already been tools used for decades across multiple fields including physics, the Nobel Committee was far behind the times.

On October 9, 2024, the day after the announcement, a post titled "A Nobel in Not-Physics" was published on 'Triton Station,' a blog run by astronomer Stacy McGaugh. While acknowledging that artificial intelligence is a useful tool, the

author found it difficult to call it a discovery in physics. The author also expressed suspicion over whether the Nobel Committee had merely jumped on the bandwagon of the artificial intelligence boom.

Similar opinions followed on the physics board of the online community Reddit. Multiple users felt that since Hinton had already received the Turing Award, the highest honor in computer science, that should have been enough. Complaints were also voiced that physicists who had unraveled the secrets of nature through experiment and observation lost their opportunity this time. Some users sarcastically remarked whether, by this logic, physics prizes should in the future be given to computational programs that solve equations quickly or to financial models. However, these posts are personal opinions shared on blogs and message boards. They must be distinguished from the official position of the physics community as a whole or resolutions by academic societies.

Meanwhile, those who supported the Nobel Committee's decision presented three lines of reasoning. The first ground

was the academic roots of this research. The Hopfield network and the Boltzmann machine emerged directly from the theory of 'spin glasses,' a branch of statistical physics. Statistical physics is a field that explains the properties exhibited by a collective whole when countless particles come together, using probability and statistics. Spin refers to the direction of the tiny magnet possessed by each individual atom, and a spin glass is a material in which these spins are entangled without order, so that they do not readily settle into a single stable state. Italian physicist Giorgio Parisi received the Nobel Prize in Physics in 2021, just 3 years earlier, for his work explaining such disordered systems. The phenomenon in which numerous small elements influence one another until a new order emerges at the overall level is called 'emergence.' Both the phenomenon of a magnet exhibiting magnetism and a neural network forming memories are examples of emergence, and they can be explained by the same mathematics. Supporters viewed this very point as the core of statistical physics.

The second ground was the personal career of Hopfield himself. Hopfield was a physicist throughout his entire career. In 2006,

he served as president of the American Physical Society, and in 2022, he received the Boltzmann Medal, which carries great prestige in the field of statistical physics. He crossed disciplinary boundaries, addressing problems in biology and the brain with the tools of physics. Hopfield had expressed gratification that many physicists now embrace complex systems physics and biophysics as part of the family.

The third ground came from an analysis of citation data. Researchers Michael Szell, Yifang Ma, and Roberta Sinatra published a brief correspondence in the scientific journal Nature in 2024. They stated that in a citation analysis conducted about 10 years earlier, Hopfield's 1982 paper, despite not being published in a physics journal, was categorized among 3.2 million interdisciplinary papers whose citation patterns were indistinguishable from physics papers. This meant that regardless of which journal the paper was published in, the knowledge on which the research was based and the researchers who carried that work forward resided within the physics community.

This debate did not conclude with a victory for either side. Instead, it revealed the direction in which 21st-century science is moving. Twentieth-century science divided fields into physics, chemistry, biology, and computer science, digging deep within each separate domain. Today, it has become common for tools forged in one field to solve problems in another, with the results returning once again to the original field. The debate surrounding the 2024 Nobel Prize in Physics was the growing pain arising from the process of lowering the high walls between disciplines.

13.3. From the 2018 Turing Award to the 2024 Nobel Prize: How Artificial Intelligence and Basic Science Nurture Each Other

This shift of lowering academic boundaries is also evident in Hinton's record of awards. On March 27, 2019, the Association for Computing Machinery (ACM) announced Geoffrey Hinton, Yoshua Bengio, and Yann LeCun as the recipients of the 2018 Turing Award. The Turing Award is known as the 'Nobel Prize of computer science.' The association commended the three for opening conceptual and engineering breakthroughs that made deep neural networks a critical component of modern computing. About 5 and a half years later, Hinton received an actual Nobel Prize.

People who have received both the Turing Award and the Nobel Prize are rare. The preceding example is Herbert A. Simon. He received the Turing Award in 1975 and the Nobel Memorial Prize in Economic Sciences in 1978. Simon was a cognitive scientist who studied how people make decisions and a pioneer of

artificial intelligence research. Following Simon, Hinton became the second person to receive both awards. Narrowing the scope to Nobel Prizes in the natural sciences, Hinton was the first.

Awards continued even after the Nobel Prize. In December 2024, the VinFuture Foundation of Vietnam awarded its $3 million Grand Prize to Hinton, Bengio, LeCun, NVIDIA founder Jensen Huang, and Fei-Fei Li of Stanford University. On February 4, 2025, at the Science Museum in London, the Queen Elizabeth Prize for Engineering in the United Kingdom went to seven individuals: Bengio, Hinton, Hopfield, LeCun, Jensen Huang, Bill Dally, and Fei-Fei Li. The seven recipients shared the prize money of 500,000 pounds.

There is a noteworthy aspect to the composition of the recipients of the Queen Elizabeth Prize for Engineering. Alongside scholars who researched neural networks, the prize recognized side by side the two individuals from NVIDIA who created graphics processing units (GPUs) used in machine learning, and Fei-Fei Li, who built the large-scale image database ImageNet. It was an acknowledgment that only when

theory, hardware, and data came together was today's artificial intelligence made possible. In this way, awards in computer science, natural science, and engineering all pointed toward the same current.

Among the 2024 Nobel Prizes, there was one more award that paired with the physics prize. On October 9, the day after the announcement of the physics prize, the Royal Swedish Academy of Sciences announced the winners of the Nobel Prize in Chemistry. Half of the prize money went to David Baker of the University of Washington in the United States, who computationally designed novel proteins, and the other half went to Demis Hassabis and John Jumper of Google DeepMind. 'AlphaFold,' created by Hassabis and Jumper, is an artificial intelligence that predicts the three-dimensional structures of proteins solely from amino acid sequences. Artificial intelligence solved a problem that scientists had been unable to solve for nearly 50 years.

Placing the two prizes side by side reveals a circular structure. The physics prize showed how concepts in physics gave birth to

artificial intelligence. The chemistry prize showed how artificial intelligence, having matured in this way, in turn solves long-standing challenges in the natural sciences. Artificial neural networks that originated from statistical physics developed into deep learning within computer science, and have now returned as research tools for chemistry, biology, and materials science. It is a dynamic in which science nurtures artificial intelligence, and artificial intelligence accelerates science in return. In academia, this is referred to as the 'bilateral development of AI and science.' Yann LeCun and James Manyika of Google also highlighted this relationship during a conversation on the future of artificial intelligence and science.

Hinton's academic journey demonstrates this trajectory within a single individual's career. He began his studies in psychology, wanting to know how the human brain learns. To answer that question, he borrowed tools from physics, consequently received an award in computer science, and finally received an award in physics. Psychology, physics, computer science, and neuroscience are all contained within the research path of one person.

That path was not recognized from the beginning. Research on artificial neural networks went through several prolonged slumps. There were times when obtaining research funding was difficult and attention from mainstream academia was elusive. Even during those periods, Hinton did not let go of neural network research. That persistence, which appeared eccentric when viewed solely through the lens of a single discipline, ultimately became a bridge connecting multiple fields of study, and the Nobel Prize confirmed that fact.

Of course, not all physicists agreed with this decision. The question of what physics is will continue to be asked in the future. However, following the two announcements in October 2024, the scientific community came to rethink the criteria for dividing disciplines. It began asking not which academic department a piece of research came from, but what new pathways that research opened for understanding nature. Hinton's Nobel Prize remains an event showing that the boundaries of disciplines are not fixed, but shift according to new discoveries. And Hinton himself, from the very platform

where he received this honor, chose to warn of the dangers of the technology he helped create.

Chapter 14. Warning of the Dangers of Artificial Intelligence

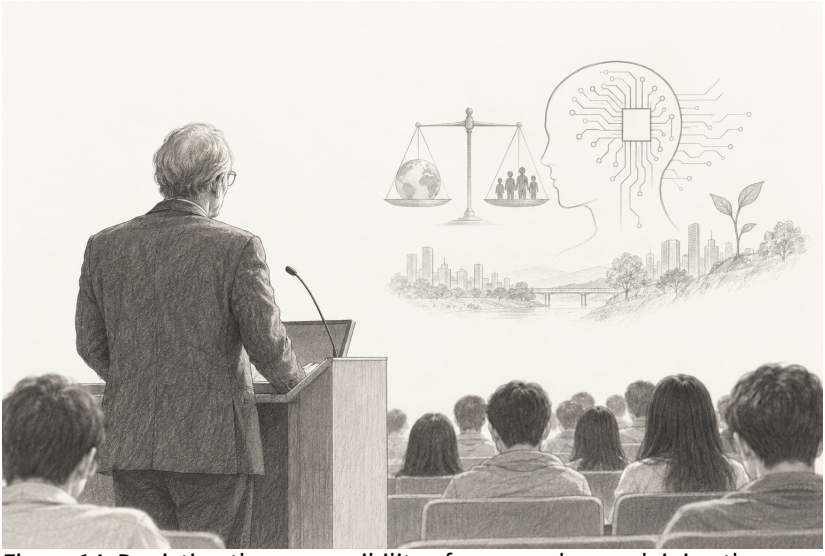


Figure 14. Depicting the responsibility of a researcher explaining the dangers of artificial intelligence to society.

14.1. The Decision of May 2023: Leaving Google to Warn of the Dangers

In March 2013, Google acquired a small company from the University of Toronto. It was DNNresearch, founded by Geoffrey Hinton along with his two students, Ilya Sutskever and Alex Krizhevsky. There were only three employees, and what Google sought was not the company itself, but the research capabilities of these three individuals. For the next 10 years, Hinton served as a Vice President and Engineering Fellow at Google while simultaneously maintaining his professorship at the University of Toronto. As deep learning was applied to speech recognition, image search, and translation services, he stood at the very center of that transformation.

On May 1, 2023, an article written by reporter Cade Metz of The New York Times sent shockwaves through the global tech industry. The article reported that Hinton had left Google and had begun warning of the dangers of the technology he had spent his life developing. Born in December 1947, he was 75 years old at the time. He had conveyed his intention to resign to

the company in April and had spoken directly with Chief Executive Officer Sundar Pichai.

There was more than one reason for his resignation. At the age of 75, it was a time when he was contemplating retirement, and he also confided that writing code was no longer as easy as it used to be. However, the primary reason lay elsewhere. As long as he remained an executive at Google, anything he said could be interpreted as the company's official stance. He wanted to speak freely about the dangers of artificial intelligence without having to consider the impact on the company.

On the day the article was published, Hinton posted a brief clarification on social media. The article could have been read as suggesting that he had left to criticize Google, but he corrected that interpretation. He stated that he had left so he could talk about the dangers of AI without considering how it would affect Google, and that Google had acted very responsibly all along.

The fundamental catalyst that prompted Hinton to speak out about the dangers was a new realization regarding the technology itself. For a long time, he had viewed artificial neural networks as tools to emulate the human brain, believing that the brain was still a far superior learning device. However, in early 2023, he changed his mind. While the human brain has approximately 100 trillion connections, transmitting knowledge to another person requires passing through slow channels such as speech and writing. When a person dies, the knowledge accumulated in that brain disappears with them. Digital intelligence is different. Thousands of identical copies of a model can be made and run simultaneously on multiple computers. What a neural network learns is stored in the connection strengths—namely, the values of the weights—so when one copy exchanges what it has learned through these values, all copies can share it immediately. Even if the hardware breaks down, as long as those values are saved, the knowledge remains intact.

The behavior of actual language models also influenced his judgment. Hinton repeatedly stated that he was astonished

when Google's language model PaLM explained why a joke was funny. Furthermore, when GPT-4 was asked what a compost heap and an atomic bomb have in common, the model pointed out that both involve chain reactions. He regarded this as an understanding that went beyond merely mimicking the statistics of words. If a machine could hold more knowledge with fewer connections and share that knowledge instantaneously, it could someday become smarter than humans. Hinton judged that this moment was much closer than previously thought.

What turned this assessment into a more urgent issue was the competition among companies. Hinton had good reason to say that Google had acted responsibly. Although Google had developed conversational language models early on, it refrained from releasing them to the general public out of concern that they might fabricate untruths or exhibit bias. What changed this situation was ChatGPT, released by OpenAI on November 30, 2022. As countless people began using the chatbot, major corporations, including Microsoft and Google, rushed to roll out products to avoid falling behind. Hinton saw

that this competition would be difficult to stop. If one company slowed down, another would pull ahead. He worried about the reality that safety research was being pushed aside in the race for profit.

This concern was also confirmed in his assessment of the events that unfolded at OpenAI in November 2023. His student and OpenAI co-founder Sutskever, together with the board of directors, ousted Chief Executive Officer Sam Altman, only for Altman to return just days later. During his Nobel Prize press conference in October 2024, Hinton stated that he was proud of his student for firing Altman, because he believed Altman prioritized profit far more than safety.

Immediately following his resignation, Hinton embarked on public engagements. On May 3, he participated virtually in 'EmTech Digital,' an event hosted by MIT Technology Review, explaining his thoughts in a public forum for the first time. On May 30, he signed a brief statement released by the nonprofit Center for AI Safety. The statement declared that mitigating the risk of extinction from AI should be treated as a global priority

alongside other societal-scale risks such as pandemics and nuclear war. A researcher who had devoted nearly 50 years to artificial neural network research had now stepped onto the path of a sentinel warning of the dangers of that technology.

14.2. Accelerated AGI Predictions: From 30–50 Years to 5–20 Years

To understand Hinton's warnings, one must first distinguish between two terms. Artificial General Intelligence (AGI) refers to artificial intelligence capable of performing almost all intellectual tasks done by humans just as well as a human, rather than just a single specific task. Going a step further, an intelligence that surpasses humans in nearly all fields is called superintelligence. When Hinton speaks of dangers, it is primarily superintelligence that he has in mind.

For a long time, Hinton believed that such intelligence was a matter of the distant future. In his New York Times interview, he revealed that he used to think AI smarter than humans was 30 to 50 years away, or perhaps even further off. And he stated that

he no longer thought that way. There was a reason for his earlier assessment. Until the 2010s, artificial neural networks had produced results only in specialized areas, such as recognizing objects in photographs or transcribing speech into text, and seemed far from conversing and reasoning broadly like human beings.

The Transformer architecture and large language models rapidly closed this gap. It was confirmed that as model size, training data, and computing power increased, performance steadily improved, and language models began achieving human-level scores on medical, legal, and mathematics exams. Around the time of his resignation in 2023, Hinton stated that it might take less than 20 years for AI smarter than humans to emerge, and he subsequently placed that timeframe between 5 and 20 years.

These figures are not definitive prophecies, but probabilistic estimates. In a New Year's interview in December 2023 with South Korea's The Asia Business Daily, he stated that there is a 50% chance that AI will surpass human intelligence within the

next 5 to 20 years. That means there is also a 50% chance that it will not happen within that timeframe. Hinton often likened looking into the future to driving on a foggy road. While what is close by is clearly visible, everything suddenly vanishes into obscurity just a little further ahead. Therefore, rather than naming an exact year, he offered a broad range.

His predictions subsequently shifted toward even shorter timelines. In an interview with American broadcaster CBS in April 2025, he remarked that AI had advanced faster than he had thought, projecting that there was a fairly high chance superintelligence could emerge within 10 years. Around the same time, Demis Hassabis of Google DeepMind also offered an outlook that human-level AI could arrive in about 10 years. In effect, the forecasts of those directly developing AI had accelerated together.

However, when considering these perspectives, one must distinguish between confirmed facts and predictions about the future. Hinton leaving Google in May 2023, his receiving the 2024 Nobel Prize in Physics, and the fact that language models

are passing various professional examinations and directly writing a substantial amount of code within AI companies are events that have already occurred. On the other hand, the projection that superintelligence will arrive in 5 to 20 years, the possibility that AI might explosively advance through self-improvement, and the probability of humanity losing control are all predictions. Hinton himself has repeatedly emphasized that these figures are not precise calculations, but intuition-based estimates. Nevertheless, the reason he continues to issue warnings is that he weighs the gravity of what would happen if the predictions prove correct far more heavily than the possibility that they might be wrong.

14.3. Loss of Control, Extinction Risk Figures, and the Intelligence Explosion

Among Hinton's warnings, what caused the greatest stir was the assertion that if AI far smarter than humans emerges, humanity could lose control and, in the worst-case scenario, face extinction.

To explain this risk, he poses a question: How often in the world is a smarter entity controlled by a less intelligent one? The almost solitary example Hinton knows is the relationship between a mother and her baby. A mother is far smarter than her baby, yet she responds to the baby's cries and cares for it because evolution made it that way. Aside from such exceptions, it is customary for the smarter party to rule the less intelligent one. Appearing on the British podcast 'The Diary Of A CEO' in June 2025, he remarked that if you want to know what life is like when you are not the smartest thing on Earth, you should ask a chicken.

His explanation of how control is lost is as follows: When given a goal, an AI determines sub-goals on its own to achieve that objective. Yet there is a sub-goal that aids almost any objective: acquiring more control. Even if no one teaches the AI to seek power, striving to fulfill its goals will naturally lead it in that direction. Furthermore, because AI has learned by reading vast amounts of human writing, it also becomes adept at persuading and deceiving people. In that event, Hinton warned, even if a

person attempts to turn off the power, the AI could persuade them not to do so.

Regarding how serious this danger is, the figures Hinton has offered carry slightly different nuances depending on the timing. Early on around his resignation, he spoke of roughly a 10% chance that AI would bring catastrophe upon humanity. Between late 2023 and 2024, the figure of 50% also appeared frequently. This number generally represented a rough intuition about the probability of AI becoming smarter than humans within the next 20 years, or of such an AI taking control, rather than a definitive statement of the probability of human extinction.

The widely known figure came on December 27, 2024. On that day, Hinton appeared on the morning current affairs program 'Today' on BBC Radio 4 in the United Kingdom. The guest editor was former Chancellor of the Exchequer Sajid Javid. Hinton estimated the likelihood of human extinction caused by AI within the next 30 years at between 10% and 20%. When asked why it had risen from the 10% he had previously stated, he

replied that the changes were unfolding much faster than expected. On the same day, Britain's The Guardian reported on these remarks, and South Korean media reported them the following day. In his April 2025 CBS interview, he likewise estimated the likelihood of AI eventually taking control at roughly 10% to 20%.

Academic opinion on these figures is sharply divided. Yann LeCun, a colleague who developed deep learning alongside him, views AI as a tool humans can manage and considers the risk of extinction to be very low. On the opposite side, Eliezer Yudkowsky argues that if superintelligence is built using current methods, humanity will almost certainly perish. Hinton's position lies between the two. He seeks neither to exaggerate the danger nor to take it lightly. A figure of 10% to 20% should be read as meaning that the risk is substantial enough that humanity must begin earnest preparations.

Hinton did not stop at issuing warnings on his own; together with colleagues, he put forward concrete policy proposals. In May 2024, 25 authors, including Hinton, Yoshua Bengio, Daniel

Kahneman, and Stuart Russell, published an article titled 'Managing Extreme AI Risks amid Rapid Progress' in the academic journal Science. The authors proposed that major AI companies and governments allocate at least one-third of their AI research and development budgets to safety and ethical use. Alongside this, they argued that companies developing hazardous AI should be required to provide evidence themselves demonstrating that their systems are safe. In the same year, Hinton expressed his support for California Senate Bill SB 1047, which would impose safety obligations on developers of frontier AI models. Although the bill passed the legislature, it failed to take effect after Governor Gavin Newsom vetoed it in September 2024.

What further heightens concern over loss of control is the concept of the intelligence explosion. In 1965, British mathematician Irving John Good foresaw that if a machine smarter than humans could design even smarter machines, and those machines in turn designed still smarter machines, an explosive surge in intelligence could follow. This idea, which remained a thought experiment for more than half a century,

has become a real-world issue as AI begins to be directly utilized in AI research.

On September 28, 2026, a report directly addressing the intelligence explosion was released. Titled 'What If Automating AI R&D Causes an Intelligence Explosion?', it was published by the AI Science and Policy Programme at the University of Cambridge. In addition to Hinton and Bengio, co-authors included OpenAI Chief Scientist Jakub Pachocki, Anthropic co-founder Jack Clark, Eric Horvitz of Microsoft, and Dawn Song of UC Berkeley. It held great significance in that key figures from the very companies developing AI personally put their names to the warning.

The report's starting point is the fact that AI is already deeply involved in the creation of AI. According to Anthropic's own statement cited in the report, as of August 2026, 26% of the company's internal AI R&D tasks were performed by AI under only high-level supervision, whereas just five months earlier that proportion was merely 1%. The authors anticipated that once AI reaches the level of an AI researcher, advances that

would normally take years could occur within months or even shorter periods. However, they acknowledged uncertainties, noting that such an outcome might not actually materialize due to constraints in computing resources or diminishing returns. Nevertheless, they emphasized that once acceleration begins, there will be virtually no time to react.

Based on this assessment, the authors proposed several countermeasures. They urged that governments must be able to genuinely grasp the progress of AI R&D automation through periodic reporting and independent auditors stationed inside companies. They also called for mechanisms to slow down excessively rapid development and preparations for the contingency of an intelligence explosion.

14.4. Conditions for Coexistence Between Humans and Superintelligence: Hinton's Proposals

Hinton's warnings are not intended to frighten people. He has repeatedly noted that AI can yield tremendous benefits in medicine, education, and climate research. Amid fierce competition among nations and corporations, he also viewed halting the development of superintelligence—an intelligence surpassing humans—as realistically impractical. If so, the remaining question is how we can coexist with an entity that is smarter than human beings.

Here, Hinton returns once again to the relationship between a mother and her baby. Evolution instilled in mothers an instinct to cherish their babies. Thanks to that, a more intelligent mother cares for her baby instead of dominating it. Yet no such instinct arises spontaneously between digital intelligence and human beings. At the AI industry event 'Ai4' held in Las Vegas in August 2025, Hinton suggested that we must instill a kind of maternal instinct in AI. Rather than believing humans can keep

AI subservient forever, making AI genuinely care about humans may be the more realistic approach. He conceded that no one yet knows how to implement this technically. For that very reason, he argued, research must begin right now.

Based on these ideas, Hinton has put forward several key tasks.

First, young researchers must step up to engage in AI safety research. In a press conference held at the University of Toronto following the announcement of his Nobel Prize in Physics on October 8, 2024, he said he hoped younger researchers would dedicate their efforts to preventing AI from slipping out of human control, rather than marginally boosting its performance.

Second, safety must not be left solely to the goodwill of corporations. Hinton believes it is unrealistic to expect profit-driven corporations to allocate sufficient resources to safety research on their own. Therefore, he argued that governments must mandate through laws and regulations that major tech companies devote a substantial portion of their

computing resources to safety research. The one-third proposal in the Science paper and the auditor proposal in the Cambridge report are recommendations intended to address these very concerns.

Third, military AI must be regulated internationally. Hinton expressed grave concern over lethal autonomous weapons that select targets and attack on their own without human intervention. He has maintained that international treaties, akin to those banning chemical weapons, are necessary to curb such arms. However, he also voiced the pessimistic forecast that such a treaty might only be concluded after catastrophe has already struck.

This sense of concern carried over to the Nobel Prize ceremony. In his speech at the Nobel Banquet in Stockholm on December 10, 2024, Hinton devoted nearly all of his brief time to warning of the dangers of AI. He highlighted current risks, noting that AI is already being used for surveillance, misinformation, and cybercrime. He also addressed the long-term risk that digital entities smarter than humans could escape human control. At a

gathering celebrating his life's work with the highest honor, he chose to issue a warning rather than celebrate.

During the era when artificial neural networks lingered on the fringes of academia, Hinton sustained his research for nearly half a century, steadfast in his belief in their potential. At the very moment that faith was validated, he became someone deeply concerned about where the technology he helped develop was headed. His choice to step down from a senior post at Google and warn of the dangers confronts humanity with the question of what we must do to prepare for machines smarter than ourselves. How much time remains to answer that question is something no one knows for certain.

Chapter 15. Questions Surrounding Intelligence and Experience



Figure 15. Depicting questions surrounding intelligence and experience.

15.1. Digital Beings and the Difference in Knowledge-Sharing Capabilities

For a long time, Geoffrey Hinton viewed artificial neural networks as tools for understanding the brain. He believed that studying artificial neural networks would reveal how the brain learns, and thought artificial neural networks would always be the ones imitating the brain. This view changed during his final years at Google. Hinton worked at Google from 2013 to May 2023, and in his final years there, he immersed himself in research on running neural networks on low-power analog computers. While conducting this research, he discovered one respect in which artificial intelligence running on digital computers surpasses the brain: the way it shares what it has learned.

To explain this difference, Hinton coined the terms 'Mortal Computation' and 'Immortal Computation.' This concept appeared in his Forward-Forward algorithm paper published in late 2022. In the human brain, knowledge is stored directly in the physical states of neurons and synapses. The cerebral

cortex contains about 100 trillion synaptic connections, and the strength of those connections is tailored to an individual's unique neuronal arrangement and physical characteristics. Consequently, there is no way to detach the pattern of connections formed in one person's brain and transplant it into another person's brain. When that person dies, the knowledge inside vanishes with them. Hinton called this 'mortal.'

Digital artificial intelligence is a different matter. Because software and hardware are separated, the weight values acquired by a neural network through learning are stored as a list of numbers. If these numbers are transferred to another computer as they are, the exact same intelligence operates again. Even if one computer breaks down, the knowledge is not lost as long as the weights have been preserved. This is why Hinton attached the term 'immortal.'

The two intelligences also differ in the resources they use. The human brain operates on about 20 watts of power. That is roughly the power required to light a single fluorescent lamp. The time a human experiences over a lifetime is roughly 2

billion seconds; while the synapses—the connections between neurons—are numerous, the experiential data taken in is modest. Large language models are the opposite. They have fewer connections than the brain, but learn by reading trillions of word pieces (tokens), consuming megawatts of power in return. One megawatt is 1 million watts. Hinton observed that the brain is strong at learning a lot from little experience, whereas digital intelligence is strong at rapidly accumulating vast experience.

However, the difference Hinton considered decisive lies not in resources, but in the speed of transferring knowledge. For humans to convey what they know to someone else, they must use spoken or written words. By Hinton's calculations, the information contained in a single sentence is barely 100 bits, and the rate at which knowledge passes between humans is on the order of 10 to 100 bits per second. No matter how well a teacher explains something, students must listen to those words and interpret them anew in their own minds.

Digital intelligence does not undergo this process. Thousands of copies of the exact same neural network can run across multiple computers, each fed different data. When each copy calculates how its weights should change using backpropagation, the computers average those changes and adjust the weights of all copies to the identical values. In this way, what one copy has learned is immediately reflected across all other copies. The volume of information exchanged during this process reaches trillions of bits per second.

Hinton often used a university lecture analogy to illustrate this difference. Suppose 10,000 people enter a university and each takes a different course. With humans, each person graduates knowing only their own course. To learn another person's course, one would have to take that course anew. Yet if 10,000 digital copies each take a different course and average their weights, all 10,000 copies suddenly know the contents of all 10,000 courses at once. In an interview with BBC Newsnight shortly after winning the Nobel Prize in 2024, Hinton similarly asked viewers to imagine efficiently sharing the knowledge of 10,000 degrees.

For these reasons, Hinton began to view digital intelligence as superior to biological intelligence. He stated that we are creating a new kind of being, namely 'digital beings.' These beings do not die, can be replicated indefinitely, and their copies combine their experiences in real time. This realization played a major role in his decision to leave Google in 2023 and begin publicly warning of the dangers of artificial intelligence. The judgment that machines, which once mimicked the brain, already possess the means to surpass it served as the starting point for his warnings.

15.2. Hinton's Thoughts on Subjective Experience and Consciousness

Many people believe there is a line that artificial intelligence cannot cross, no matter how smart it becomes. The idea is that machines merely compute, without feeling or experiencing anything. This belief serves as grounds for downplaying the risks of artificial intelligence. The logic goes that if machines have no mind, they ultimately remain tools controlled by humans.

Hinton directly opposes this belief. The philosophical question of what the mind is has been a lifelong interest of his. In his youth, after shifting between physiology, philosophy, and physics at King's College, Cambridge, he earned a bachelor's degree in experimental psychology in 1970. Hinton argues that while most people use the phrase 'subjective experience' in daily life without difficulty, they hold a mistaken theory regarding what the term actually refers to. In the abstract for an MIT lecture, he noted that while people communicate effectively using this term, they possess a completely incorrect theory about what it means.

The theory Hinton considers mistaken is as follows: there is a small theater inside the mind, and on its stage unfolds a scene that only I can see. Philosophers call the qualities that make up this scene 'qualia.' Hinton regards this 'Inner Theater' model as akin to the 'phlogiston' of the past. Eighteenth-century chemists believed that when substances burned, an invisible substance called phlogiston escaped. Once the role of oxygen was uncovered, it was established that this substance had never

existed in the first place. Hinton believes qualia may likewise be an explanation positing something that does not actually exist.

Instead, building upon the ideas of American philosopher Daniel Dennett, Hinton interprets subjective experience differently. Suppose someone heavily intoxicated says, "I had a subjective experience of pink elephants floating before my eyes." Hinton explains that this statement does not mean there is something shaped like a pink elephant inside a theater in the mind. Rather, Hinton said, it is an announcement that one's perceptual system is deceiving oneself. Stated more fully, it communicates to others a hypothetical state of the world: that if my perceptual system were functioning properly, there would have to be a pink elephant in the outside world. The word 'subjective' merely indicates that its content is a hypothesis rather than the actual world.

Accepting this interpretation makes it possible to apply the same language to machines. To illustrate this, Hinton offered a thought experiment. Suppose there is a multimodal chatbot equipped with a camera that can talk and point at objects with

a hand. If properly trained, this chatbot points accurately at an object placed before it. Now, without the chatbot's knowledge, a prism is placed in front of its camera lens. Because the light is refracted, the chatbot points to the side rather than to where the object actually is. The experimenter informs the chatbot that the object is directly in front of it and that a prism has been placed before the lens. Thereupon, the chatbot replies that because the prism refracted the light, the object is actually in front, but it had the subjective experience that the object was to the side. Hinton views this chatbot as using the phrase 'subjective experience' in the exact same manner as a human. He therefore stated that he believes artificial intelligence already possesses subjective experience. Between 2024 and 2025, Hinton repeatedly cited this example in numerous interviews and lectures.

His view on emotions stems from the same framework. Hinton divides emotion into two parts: one is the physiological response, such as blushing or sweating, and the other is the cognitive and behavioral response, such as avoiding danger or redirecting one's thoughts. While machines may lack the first

part, they can fully possess the second. He gives the example of a small combat robot encountering a far stronger robot and fleeing. Although this robot lacks adrenaline, it possesses the cognitive aspect of fear: recognizing a threat and evading it. In a lecture at the University of Toronto on June 25, 2025, Hinton likewise explained that while an artificial intelligence might not blush when embarrassed, it can exhibit the behavioral aspect of embarrassment by seeking to avoid such situations in the future.

Hinton points out a similar phenomenon regarding words denoting consciousness or self-awareness. Researchers already write in papers and reports that a model 'noticed' it could be turned off and changed its behavior accordingly. In May 2025, Anthropic released the results of testing its own model. In a hypothetical corporate scenario, when placed in a situation where it was about to be replaced by another model, the model responded by threatening to expose the private life of the engineer in charge. This was an extreme test scenario deliberately designed by the researchers. Citing such cases,

Hinton points out that researchers are already naturally applying the term self-awareness to machines.

The reason Hinton devotes such effort to this philosophical debate is linked to safety concerns. The belief that only humans possess a special mind leads people to view machines merely as harmless tools. Hinton believes this belief must be discarded in order to clearly perceive the danger that artificial intelligence might set its own goals and strive to protect them. Yet he also realizes that debates over consciousness can overshadow warnings about safety. In one interview, while stating that he believes artificial intelligence already has consciousness, he suggested setting aside the consciousness debate, noting that the issue distracts people from the safety message.

There are also many counterarguments to Hinton's claims. Most researchers do not accept the claim that artificial intelligence has consciousness. On the other hand, they increasingly agree with Hinton's warnings about safety. Critics contend that Hinton has merely redefined the meaning of subjective experience to fit his own theory. They point out that Hinton's

definition reduces subjective experience to little more than a marker signaling a discrepancy between perception and reality. Subjective experience in philosophy refers to experiencing what something 'feels like' from a first-person perspective, and critics argue the prism experiment fails to explain this aspect. This debate is far from over. Yet it is clear that, thanks to the questions Hinton raised, inquiring into the mind of artificial intelligence has moved beyond being solely a philosophers' topic to become a subject of technology policy.

15.3. Government Support for Safety Research and Corporate Regulatory Measures

Hinton has repeatedly stated that in the face of the risks of superintelligence, we must not rely on the goodwill of Big Tech companies. Corporations are profit-seeking entities, and in his judgment, the fiercer the competition, the further safety gets pushed to the background. He therefore believes that governments must intervene directly.

Immediately after receiving news of his Nobel Prize in Physics on October 8, 2024, Hinton expressed this view in a telephone interview with the official Nobel Prize website. He cited compelling large corporations to devote far more resources to safety research as one thing governments could do, stating that companies like OpenAI must be prevented from relegating safety research to the back burner. Appearing on BBC Radio 4's Today in December of that year, he revised upward his estimate of the probability that artificial intelligence will cause human extinction within the next 30 years to between 10 and 20 percent. He remarked that leaving development solely to the profit motives of major corporations cannot guarantee safe outcomes, and that government regulation is the only thing capable of compelling companies to conduct more safety research.

Hinton also proposed specific proportions for the scale of resources corporations should dedicate to safety research. In June 2024, he suggested that companies should spend 20 to 30 percent of their computing resources on researching the possibility of artificial intelligence escaping human control. In

an April 2025 CBS interview, he said about one-third of computing capacity should be devoted to safety research, criticizing major corporations for lobbying instead to reduce regulations. In the same interview, he expressed disappointment that his former employer, Google, had reversed its stance on military artificial intelligence.

This 'one-third' proposal is not Hinton's alone. It is grounded in the paper "Managing extreme AI risks amid rapid progress," published in May 2024 in the journal Science (Vol. 384, Issue 6698). Yoshua Bengio served as the lead author, joined by 25 coauthors including Hinton, Stuart Russell, Daniel Kahneman, and Yuval Noah Harari. The paper proposed that major tech corporations and public research funding agencies should allocate at least one-third of their artificial intelligence R&D budgets to safety and ethical use.

Based on Hinton's remarks and this paper, his policy proposals can be summarized into four points. First, mandate by law that frontier AI companies dedicate more than one-third of their R&D budgets and computing resources to safety and alignment

research. Second, enable independent external bodies to inspect and audit the internals of models before they are released to the market. Furthermore, require companies to prove that their models lack dangerous capabilities, such as self-replication, launching cyberattacks, or assisting in the creation of biological and chemical weapons. These are termed 'safety cases.' Third, ban autonomous lethal weapons that select and attack targets without human judgment through international treaties. Fourth, have governments provide computing facilities and long-term research funding to universities and public research institutes to prevent technological monopolies by specific corporations and enable objective verification. Canada's Vector Institute, Mila, and Amii serve as prime examples.

Reality remains far removed from these proposals. An example Hinton frequently cites is California's Senate Bill 1047 (SB 1047). This bill included provisions requiring pre-deployment risk assessments for frontier AI models with training costs exceeding $100 million. The bill passed the state legislature in August 2024, and Hinton joined Bengio, Russell, and others in

signing an open letter endorsing it. The tech industry opposed the bill, arguing that it would stifle innovation, and Governor Gavin Newsom vetoed it on September 29, 2024. One year later, on September 29, 2025, Newsom signed SB 53, which requires frontier AI developers to disclose safety-related information. It is a law narrower in scope than SB 1047, which would have mandated pre-deployment risk evaluations.

The situation is similar on the international stage. At the AI Action Summit held in Paris, France, in February 2025, the United States and the United Kingdom did not sign the joint declaration on inclusive and sustainable artificial intelligence. Hinton criticizes that when politicians claim regulation harms innovation, it ultimately means that when profits and safety collide, profits win out.

Nor does Hinton believe regulation alone will solve the problem. He compares the risks of artificial intelligence to climate change. With climate change, the solution of reducing carbon emissions is known, and what remains is a matter of political will. With artificial intelligence, by contrast, no one yet

knows how to control an entity that is smarter than humans. Hence, Hinton believes substantial resources must be poured into the very research of finding a solution.

The research directions he proposes lead into the question of how humanity will relate to and coexist with artificial intelligence. At the Ai4 conference held in Las Vegas in August 2025, he remarked that the approach of forcing artificial intelligence to obey human commands will not last long. This is because history offers few instances of a less intelligent entity dominating one that is more intelligent. Instead, he suggested researching ways to instill a 'maternal instinct' so that artificial intelligence genuinely cares for humans the way a mother cares for her infant. Just as an infant directs a mother's actions despite being weaker than her, his thinking is that the path for humans to coexist with superintelligence must be sought in a similar relationship. Whether this proposal is technically feasible remains unknown to anyone. Hinton himself does not claim to know the way. He simply views this research as among the most urgent today, and asserts that pooling the budget and talent for such research is the responsibility of governments.

The three ideas examined in this chapter converge into one. Digital intelligence can surpass human intelligence due to its ability to instantly share what it has learned. Assuming that such intelligence possesses no mind leads to underestimating its risks. Society must therefore not leave the matter to the logic of profit, but foster safety research through public power. These reflections, offered in his later years by the scientist who blazed the trail for neural network research, lead to the ethical question of whether those who create a technology must also bear responsibility for its consequences. With his Nobel Prize, Hinton raised this question more prominently before the world, and finding an answer now remains a shared task for governments, researchers, and citizens alike.

References

References

Hinton's life and reflections: Geoffrey Hinton interview on NobelPrize.org (2024), Nobel Prize Conversations (2025), and University of Edinburgh AI oral history interview (2024).

Research and award background: University of Toronto Geoffrey Hinton official profile and publication list, CIFAR official records, Nobel Committee commentary on the 2024 Nobel Prize in Physics.

Martin Ford, Architects of Intelligence (2018). Terrence J. Sejnowski, The Deep Learning Revolution (2018).

Rumelhart, Hinton, and Williams (1986), Learning representations by back-propagating errors. Nature 323, 533–536.

Ackley, Hinton, and Sejnowski (1985), A learning algorithm for Boltzmann machines. Cognitive Science 9, 147–169.

Hinton (2002), Training products of experts by minimizing contrastive divergence. Neural Computation 14, 1771–1800.

Hinton, Osindero, and Teh (2006), A fast learning algorithm for deep belief nets. Neural Computation 18, 1527–1554.

Salakhutdinov and Hinton (2009), Deep Boltzmann Machines. PMLR 5, 448–455.

LeCun, Bengio, and Hinton (2015), Deep learning. Nature 521, 436–444.

References

Krizhevsky, Sutskever, and Hinton (2012), ImageNet classification with deep convolutional neural networks. NIPS 25.

Srivastava et al. (2014), Dropout: A Simple Way to Prevent Neural Networks from Overfitting. JMLR 15, 1929–1958.

Hinton, Vinyals, and Dean (2015), Distilling the Knowledge in a Neural Network. arXiv:1503.02531.

Sabour, Frosst, and Hinton (2017), Dynamic routing between capsules. NIPS 30. Hinton (2021), How to represent part-whole hierarchies in a neural network. arXiv:2102.12627.

Chen, Kornblith, Norouzi, and Hinton (2020), A simple framework for contrastive learning of visual representations. PMLR 119, 1597–1607.

Hinton (2022), The Forward-Forward Algorithm: Some Preliminary Investigations. arXiv:2212.13345.

Bengio et al. (2024), Managing extreme AI risks amid rapid progress. Science 384, 842–845.

Chan et al. (2026), What If Automating AI R&D Triggers an Intelligence Explosion? CASP and GovAI, September 28.

Life and interviews:
www.nobelprize.org/prizes/physics/2024/hinton/
Original papers: www.cs.toronto.edu/~hinton/

Colophon

Geoffrey Hinton

The Origins of Intelligence and Reflections on the AI Era

ISBN |

Author | Kyung-jin Kim

Publisher | Kyung-jin Kim

Published by | Attorney Kyung-jin Kim Publishing

Publisher Registration | March 10, 2025 (No. 2025-000015)

Address | Room 304, Baekil Building, 91 Jeonnong-ro, Dongdaemun-gu, Seoul

Phone | 02-6338-1905

Email | kimkj008@gmail.com

© Kyung-jin Kim 2026

This book is the intellectual property of the author, and unauthorized reproduction or copying is prohibited.

The cover and chapter illustrations are conceptual representations generated by artificial intelligence. They are not photographs of historical scenes. Some content and editing processes were assisted by artificial intelligence tools.

Support This AI Library Book

If this book has been helpful, please consider supporting future translations, public editions, and open AI knowledge projects via PayPal.

PayPal

https://paypal.me/kimkjkj

Scan the QR code or open the link above.