TECHNICAL REPORT
R-476
May 2018

# A Personal Journey into Bayesian Networks

**Judea Pearl**
Cognitive Systems Laboratory
Computer Science Department
University of California, Los Angeles, CA 90024 USA
*judea@cs.ucla.edu*

**Abstract**

This report contains personal memories from the years that Bayesian networks were developed, 1978-1988. It was written as a section in *The Book of Why: The new science of cause and effect* (Pearl and Mackenzie, 2018) but was taken out to meet space limitations. I am archiving these memories with the hope that they will prove useful for future archivists.

The dream of artificial intelligence is, of course, an old one, dating back to Alan Turing and his 1950 paper, "Computing Machinery and Intelligence." For many years the leading approach to AI had been through logic, as formulated by George Boole's The Laws of Thought (1854) and his followers. The idea was to organize human knowledge as a collection of specific and general facts, along with inference rules to connect them. For example: Socrates is a man (specific fact). All men are mortals (general fact). From this knowledge base we can derive the fact that Socrates is a mortal, using the universal rule of inference: If all $A$'s are $B$'s and $x$ is an $A$ then $x$ is a $B$.

The approach was fine in theory, but real-life knowledge can rarely be captured by hard and fast rules, and as soon as exceptions to the rules had to be accommodated, the problem quickly got out of control. Consider, for example, the two plausible premises:

1. My neighbor's roof gets wet whenever mine does.

2. If I hose my roof, it will get wet.

Taken literally, these two premises imply the implausible conclusion that whenever I hose my roof, the neighbor's roof will also get wet. Such paradoxical conclusions are normally attributed to the coarseness of our language, reflected in the many exceptions to premise 1. The paradox disappears once we start listing the exceptions:

1*. My neighbor's roof gets wet whenever mine does, except when it is covered with plastic, or when my roof is hosed, etc.

As you can readily imagine, the number of exceptions needed for any reasonable rule quickly grows beyond the ability of even a computer to keep track of them all. Special provisions have to be made to deal with abnormalities without enumerating every single exception. One way of doing that was to develop new types of logical systems, called "default logics," that could accommodate and combine abnormalities in the knowledge base, whenever alerted to their presence.

Another type of system that managed to withstand the proliferation of exceptions was called an "expert system." Pioneered by Turing Award winner Edward Feigenbaum of Stanford University, these computer programs attempted to capture the step-by-step procedure by which an expert goes from evidence to conclusion. Some early expert systems included MYCIN, a medical diagnostic program, and PROSPECTOR, an early oil-exploration application that was supported by Schlumberger. But by 1980 it was clear that these programs struggled with making correct inferences from uncertain knowledge. The computer could not replicate the inferential process of the expert because, evidently, the experts were not able to articulate that process within the language provided by the system.

The late 1970s, then, were a time of ferment in the artificial intelligence community over the question of how to deal with uncertainty. There was no shortage of ideas. Lotfi Zadeh of Berkeley offered "fuzzy logic," in which statements are not either true or false but instead take a range of possible truth values. Glen Shafer of the University of Kansas proposed "belief functions," which assigned two probabilities to each fact, one indicating how likely it is to be "possible," the other, how likely it is to be "provable." The developers of MYCIN tried "certainty factors," which inserted numerical measures of uncertainty into their deterministic rules for inference. Thus, for instance, one rule may say, "If High Fever, then conclude Malaria with certainty c1." Another may say, "If Malaria, then conclude Trip to Africa with certainty $c_2$." The system would then combine the measures $c_1$ and $c_2$ to compute the certainty of concluding Trip to Africa, given the evidence High Fever. These rules of combination, however, were not based on probability theory or any other principled methodology, and therefore tended to produce unintended results. In particular, they could not exhibit the "explain away" effect that we discussed earlier in this chapter, in the section on colliders. Also, the rule "If Fire, then Smoke (with certainty $c_1$)" could not combine coherently with "If Smoke, then Fire (with certainty $c_2$)" without triggering a runaway buildup of certainty, like feedback in an amplifier.

In 1982, I entered the arena with a paper showing an efficient belief-updating algorithm using Bayesian probabilities, an approach that I would eventually (1985) call "Bayesian networks." The idea was, first, to model nature, not the expert. Second, to model nature in the form of a hierarchy of cause-effect relationships, compatible with the organization of scientific knowledge. Third, to insist on coherence with orthodox probability theory and Bayesian reasoning. Fourth, to execute the computation in a message-passing style resembling the firing of neurons. To convey these desiderata, I titled the paper "Reverend Bayes on Inference Engines: A Distributed Hierarchical Approach." I believe that the combination of a principled Bayesian analysis conducted in a message-passing style was the feature that attracted attention to this paper, and eventually to Bayesian networks in general.

What was the genesis of the Bayesian networks idea? Here one needs to distinguish communal genesis as viewed from bits and pieces that appear and disappear in the literature, and personal genesis as viewed from my own perspective at the time of its development. I will try to cover both in the same narrative.

From a communal perspective, Bayesian networks can be traced to multiple sources. Sewall Wright's path diagrams are certainly an antecedent, but I dismissed them immediately. At the time, I thought they were too wedded to linear relations (see Chapter 7) and improperly embedded in deterministic equations. Even earlier, in 1913, an American legal scholar named John Henry Wigmore had introduced "Wigmore charts," which represented courtroom evidence and conclusions in graphical form, with arrows pointing from evidence to hypotheses. This is the opposite of the causal direction: the evidence about the bullet points to the shotgun that fired the bullet. Wigmore never developed his diagrams into a mathematical system, and they were never used to compute numerical probabilities of guilt or innocence. But they could be. In 2005, statisticians Joseph Kadane and David Schaum published an entire book on the Sacco-Vanzetti case, the famous trial in 1920 in which two anarchists, Nicola Sacco and Bartolomeo Vanzetti, were charged (and convicted) with murdering two security guards in the course of a robbery. Ever since their execution in 1927, historians have debated whether Sacco and Vanzetti really committed the crime or whether they were framed. Kadane and Schaum converted the 395 pieces of evidence introduced at that trial into a Bayesian network. (Their verdict: Not guilty.)

Before engaging in the Sacco-Vanzetti project, Schaum had done a lot of work on evidential reasoning using probability theory that, ironically, nearly convinced me that probability is not the way to go! As soon as the evidence affected more than two or three different hypotheses, the number of equations that were needed to keep track of its impact became insurmountable. Clearly, I told myself, if probability theory were to be used one day for common-sense reasoning, it would need new methods of representation and computation.

A much closer antecedent to Bayesian networks was decision theory, which I had been teaching about and thinking about at UCLA since the 1970s. It is strange to think that if I had not been asked to teach a decision theory class, the course of my life might have been different. Several of the writers in this field influenced me strongly, among them Jimmy Savage, Howard Raiffa, Dennis Lindley, Amos Tversky and Daniel Kahneman. To sum up a rich intellectual experience in far too few words, I would say that teaching this discipline shaped my passion for probabilistic logic, not merely as a principled way of managing data, but as a way of organizing scientific knowledge and as a protector of coherent reasoning.

Perhaps the biggest influence on me, though, was an article by David Rumelhart, written in 1976, on children's reading. The article made it clear (see Figure 1) that reading is a complex process in which neurons on many different levels are active at the same time. Some of the neurons are simply recognizing individual features—circles or lines. Above them, another layer of neurons is combining these shapes and forming conjectures about what the letter might be. In Figure 1, the network is struggling with a great deal of ambiguity about the second word. At the letter level, it could be "FHP," but that doesn't make much sense at the word level. At the word level it could be "FAR" or "CAR" or "FAT." The neurons pass this information up to the syntactic level, which decides that after the word "THE" it's expecting a noun. Finally this information gets passed all the way up to the semantic level, which realizes that the previous sentence mentioned a Volkswagen, so the phrase is likely to be "THE CAR," referring to that same Volkswagen. The key point is that all of the neurons are passing information back and forth, from the top down and from

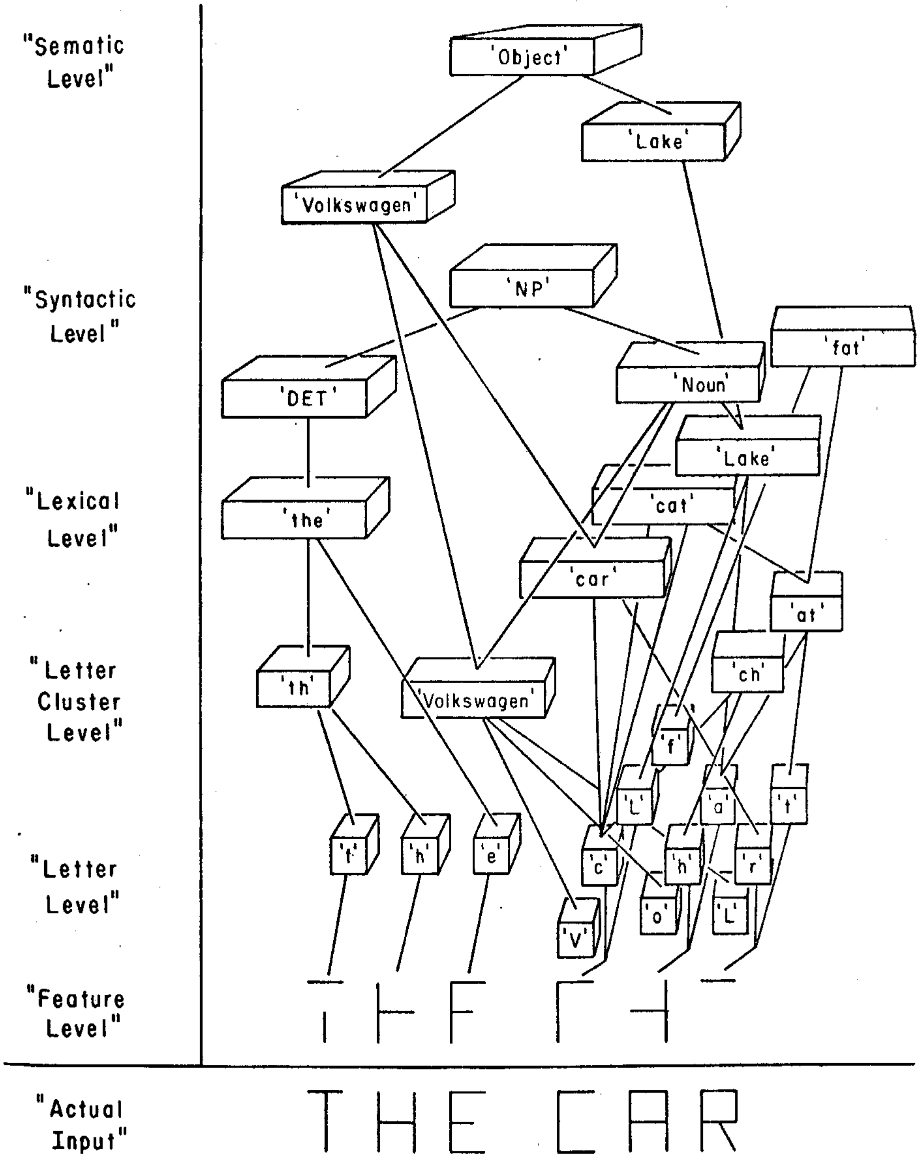

Figure 10 The message center well into the processing sequence.

Figure 1:

the bottom up and from side to side. It's a highly parallel system, and one that is quite different from our self-perception of the brain as a monolithic, centrally controlled entity.

Reading Rumelhart's paper, I felt convinced that any artificial intelligence would have to model itself on what we know about human intelligence, and that machine reasoning under uncertainty would have to be constructed with a similar message-passing architecture. But what are the messages? This took me quite a few months to figure out. They are subjective conditional probabilities, like the ones we met in the teahouse example earlier in the chapter. This neuron believes that the letter has a line on the top, that neuron believes the letter is $R$, and another neuron believes the word is CAR. Each neuron adjusts its belief based on what the neurons near to it are saying. And Bayesian reasoning is essential if the messages are to be passed both up and down the network, and combined properly.

At first I was only able to show that the Bayesian belief-propagation algorithm works in a tree—that is, a network in which each node has at most one parent, and hence the network contains no colliders. No matter what evidence you enter into such a network, it will ripple through the nodes and then they will settle down again into a new probability distribution, and this distribution is identical to what you would get if you took the time to do it by computing proportions in a gigantic probability table. This was the substance of my 1982 paper.

By the next year one of my students, Jin Kim, had generalized the result to polytrees, which are Bayesian networks in which a node may have multiple parents, although no loops are allowed. This result was good enough for many applications but still not fully satisfactory. Certainly the human brain is not a tree or a polytree, and many of the Bayesian or causal networks in practical applications are not either. (Look again at the smallpox example in Chapter 1 for a simple case of a loop.)

In the mid-1980s, loopy Bayesian networks were a challenging case for many AI researchers, because one could imagine the information introduced by a new observation chasing itself around the loop forever. As it turns out, we can keep this from happening in two ways, by "conditioning" or by "clustering." Conditioning involves finding a set of nodes that break all the loops in the network and fixing every node in the set to some value. The remainder of the network then acts like a tree, for which we can propagate evidence using the polytree propagation method. Clustering involves forming local groups of variables in such a way that the topology of the resulting network (treating each group as a single compound variable) is loop-free.

I did not dare to publish my 1985 paper on Bayesian networks until I discovered the conditioning scheme, which, as it seemed to me, proved that any Bayesian network lends itself to probabilistic evaluation. In the meantime, Ross Shachter had found another method of evaluating Bayesian networks (called node elimination), and David Spiegelhalter and Steffen Lauritzen found an efficient method of structuring cluster trees (which they called the junction tree algorithm). After that, it seemed as if the race for an algorithm capable of evaluating arbitrary networks was over. But it wasn't! The interesting part was still ahead of us.

It so happened that in one of my 1988 papers, I had written the following remark: "One should not overlook a simple but important approximation method called 'ignore the loops,' namely propagate the messages according to the equations developed for a polytree." In my book on Bayesian networks, which came out that year, I even assigned this "approximation method" as a homework exercise, and asked the reader to explore its convergence properties. It did not even occur to me that this offhand remark and "homework exercise" would determine the fate of Bayesian networks.

Surprising even its makers and developers, this approximation scheme turned out to be not only efficient but correct in most practical applications. The reason for this phenomenon is still not completely understood, but it appears that messages get "tired" going around never-ending loops, while those messages that propagate up and down the network without getting caught in a loop survive and prevail, and eventually give us the correct belief for every variable of the network.

It was this "homework exercise" that gave Bayesian networks their final push toward acceptance as the main scheme for processing uncertain information on computers. The advantages are clear: the user need not supervise how the messages propagate, there is no need to break up loops, and no need to organize clusters. It is all done autonomously by the nodes themselves, in much the same asynchronous way that neurons are activated in our brain. Each receives messages from its neighbors, gets activated, and sends messages to its neighbors. There is no master neuron that supervises the sequence of activations over the whole network.

In parallel with these technical developments, another struggle took place: winning the hearts and minds of my colleagues. That process may have begun on August 8, 1984, when I was invited to appear on a panel at the American Association for Artificial Intelligence conference in Austin, Texas. The topic was "The Management of Uncertainty in Intelligent Systems." I was not aware of it at the time, but this panel discussion was probably a pivotal moment for Bayesian networks.

All of the protagonists were there—Zadeh, Shafer, and others—and each of us was surely filled with the inspiring belief that we had the right answer to uncertainty. By now, all of the arguments pro and con are familiar and well-worn, but back then they were new. Peter Cheeseman, a machine-learning theorist who would later work at NASA, was in the audience and wrote this report a year later:

> Anyone who attended this panel discussion must have come away very confused because of the variety of formalisms proposed for representing and manipulating uncertainty, and the contradictory views expressed. Some speakers implied that more than one number was necessary to represent uncertainty, while others stated that numbers should not be used at all! Except for a valiant rearguard defense by Judea Pearl, everyone on the panel agreed that probability as a representation of uncertainty either was misguided or inadequate for the task. Several of us who have been using probability within AI, as well as engineers and physicists, know this conclusion to be false, and our outrage at this denigrating of probability was the spur that triggered this workshop.

The last sentence refers to the first Uncertainty in Artificial Intelligence meeting, which he helped organize in 1985. I can still remember him driving the signs into the ground on the UCLA campus, directing attendees to the meeting.

Although the UAI meeting was by no means an instant victory for Bayesian networks, it became an annual event. Over the years, researchers "voted with their feet" and the other approaches became less and less popular. David Heckerman, later of Microsoft Research, delivered an especially effective critique, explaining why belief functions and certainty factors could not adequately account for the "explain-away" effect that I mentioned earlier. As for fuzzy logic, it is still around. You can buy rice cookers with fuzzy logic-based controls. My only objection to it is that it pays too much attention to translating linguistic terms ("large," "very large," "extremely large") into technical quantities, and too little attention to how those quantities propagate over large and highly interconnected societies of variables. We didn't really need a whole new language to express uncertainty, because an adequate language already existed—probability. It was simply a matter of educating people to use it better. I even wrote a paper called "How to Do with Probabilities What People Say You Can't." It became a fighting slogan.

But you do not win hearts and minds with slogans alone. Indeed, what Bayesian networks asked probability theorists to believe was quite heretical. From day one, probability was defined by assigning weights to tables, like Tables 2 and 3, where each row in the table represents a combination of events and the weights represent frequencies or degrees of belief. Now, along come Bayesian networks and ask researchers to start with a diagram, not a table. All subsequent calculations are conducted on the diagram. Where is the theoretical or mathematical justification for replacing tables with diagrams, and how do we know that these two objects, which appear so different, will give us consistent answers? The answer came through the theory of "graphoids." I don't want to overtax the reader with jargon, but I do want to decorate the pillars that turn Bayesian networks into a temple. In the summer of 1985, Azaria Paz, a computer science professor at the Technion, came to visit me, and the two of us along with my students Danny Geiger and Thomas Verma worked out a precise correspondence between statements about graphical models (diagrams, networks, or maps) and statements about conditional independence in probability distributions. (See the section on "Bayesian Junctions" for examples.) The theory of graphoids conferred legitimacy on Bayesian networks. They could not be viewed any more as a recreational gimmick, merely a visual way to represent what we know already. Like Euclid's axioms in geometry, the graphoid axioms turn Bayesian networks into a tool for deriving new knowledge. No longer do you need to do formidable calculations to show that two variables are conditionally independent: you can simply read it off a figure, and be assured that what you read is as rigorous as doing it the hard way.

Later, of course, it would turn out that I had been a little bit too passionate in my educational crusade for probability. I had been under the impression that probabilities could express every aspect of human knowledge, even causality, but they can't. After I realized this, my research shifted completely to a new direction, as I tried to understand how to represent causal knowledge and draw causal conclusions. At this time I chose to leave behind the probabilistic side of Bayesian networks and entrust their beauty to the safekeeping of others.

To some extent I miss being involved in the Bayesian network community, but glad to see that they have been able to carry on perfectly well without me, as the examples in this chapter attest. For me personally, the change to working on causality was invigorating and transformative. In Bayesian networks, the crucible of discovery was perhaps starting to cool down, but in causality it was just heating up. And everything I had learned about Bayesian networks—junctions, screening off, *d*-separation (see Chapter 7), etc.—came in very handy in the new subject of causality.

The success of Bayesian networks had one unexpected effect: it turned me into a mini-celebrity in some circles. As media people began to interview me, they invariably wanted to know if I am endowed with some special genetic mutation, or whether my scientific affliction had come on in kindergarten. They were mighty disappointed to find out that I did not particularly excel in school, always being third or fourth in my class (never first) and probably twentieth when it came to handing out awards and prizes.

After I received the Turing Award from the Association for Computing Machinery in 2012, I was invited to attend a meeting of high-profile educators who were discussing strategies to improve science education all over the world. In one of the sessions I found myself sitting next to Bill Gates, who asked me squarely: "Can you recall a factor or an influence that can perhaps account for your scientific accomplishment?" This caught me off guard, because there was really nothing extraordinary about my school days that I could recall. Yes, I did fairly well in class. I understood the lectures, occasionally asked questions and occasionally came up with non-trivial answers. But I did not feel anything like the class geniuses who knew things way before the lecture and treated all questions as trivial. Now, here I was, standing before the educational leaders of the world, who were all eager to hear from humble me about the methods they should implement to make the next generation of students more science-minded!

It took me a few seconds to get over my embarrassment, to reflect back on my education and compare it to the education I find in American schools. "Yes, there was something special about my education," I said. "I am a fortunate product of the greatest educational experiment in history. Just imagine if the state of California managed to convince every science professor in the state to serve five years as a high-school teacher. Further assume that they do it, not as an obligation but as a life mission. Just imagine the curiosity and creativity of a student coming out of these high schools."

I am one of those lucky students. My teachers were displaced academics from Germany, who had been at the top of their fields before the Nazis came to power. They knew the joy of scientific discovery and they understood that this joy would no longer be part of their professional life. Instead, they had to take high-school teaching jobs when they came to Israel, and they viewed us, their pupils, as substitutes for their lost dreams. They instilled in us the idea that each of us could change the world; we could find a totally novel proof of the Pythagorean theorem, or a simpler proof that doubling the cube is impossible.

They also taught us chronologically instead of logically. They took us to Archimedes' bathtub, before he jumped out screaming, "Eureka!" They took us to seventeenth-century Padua, where Galileo was smashing Aristotelian physics and telling the world how astronomy really worked. They put a human face behind every theorem and every discovery. In this way, I learned that science is not a book of facts and recipes, but a struggle of the human mind to unveil the mysteries of nature. In this way, they made each of us an active participant, not a passive recipient, of those theorems and discoveries.

In Yiddish there is a wonderful word that doesn't quite translate successfully into English: chutzpah. It has the sense of "arrogance," and is also sometimes translated as "gall" or "nerve" (as in, "He's got a lot of nerve!"). But these words in English almost always have a negative connotation. When Israelis use the word chutzpah, they do so with a mixture of humor and admiration. It's the arrogance of believing that you can do something good by yourself, even if you don't have a permission slip or a diploma that says you can do it. Larry King, the famous TV talk-show host, wrote, "Here's a good illustration of chutzpah: The Jewish women's organization Hadassah that raises money for Israel opens a fund-raising office in Libya. See what I mean by beyond gall?" To this one must add that the good ladies of Hadassah sincerely believed that Ghadafi would recognize the importance of their cause. Sincerity is a necessary part of chutzpah.

My generation probably has a higher than average share of chutzpah, but not vastly different from that of Galileo or of Hadassah. I think that it was strongly related to our experience of growing up in an era when a new nation was being built. Over the course of just a few years, 1918-1948, Israelis put together a state and a functioning society that hadn't existed before, with no traditions or authorities to tell us how to do it. My hometown was established just 12 years before I was born, and the founders knew that their future would depend entirely on the way they shaped it. I saw my grandfather transform from a merchant to a dairyman, because it was necessary. I rode the first bus that connected my town to Tel Aviv. My high school was established just the year before I enrolled. Education, transportation, health care, water supply— all had to be created by people who did not necessarily have any experience or training in those matters.

I was 12 years old when Israel was established in 1948, and with the naiveté of youth, I thought that everybody experiences something like this in their childhood years. It was only later in life that I realized how rare it is and how lucky I had been. This experience has given my generation the courage to think for itself and to challenge authority. It is this experience, perhaps, that prepared me for the challenge of speaking to a room full of scientific authorities from diverse fields, and trying to excite them about new ways of doing what they had been doing for decades before. Naturally, not every authority enjoys the experience of learning from an outsider, and some would even tell me (authoritatively) that I did not know what I was talking about. Here is where chutzpah comes in handy—I acknowledge their predicament and pray for their eventual awakening.

Perhaps it was chutzpah, then, that in the early 1980s led me to start writing and publishing articles in a field where I was at first a complete unknown. Unlike many of the people who work in artificial intelligence, I did not have a particular tradition or a school behind me. But I consider myself lucky that I didn't have to carry all the baggage that came with being an AI researcher, such as a prejudice against probability or an entrenched opinion on the quickest way to automating intelligence. It didn't matter that I was alone; for me that was part of the fun! I had an idea and a sense that history was on my side—my high-school teachers told me so.

In retrospect, fighting for the acceptance of Bayesian networks was a picnic—no, a luxury cruise!—compared with the fight I had to go through for causal Bayesian networks. That battle is still going on.

# References

PEARL, J. and MACKENZIE, D. (2018). *The Book of Why: The New Science of Cause and Effect.* Basic Books, New York.