9 Knowledge-Based Systems

Throughout this book, we have insisted that intelligent behavior in people is often conditioned by knowledge. A person will say a certain something about the movie *2001* because he or she has a belief about when the movie was released. But we have not attempted to explain the *mechanism* behind this conditioning. For all we have said, it might be the case that all this talk of knowledge and belief is just a stance, the intentional stance of chapter 3, a placeholder for some better explanation yet to come, perhaps in terms of the electrochemical workings of the brain.

Let's now consider a possible mechanism.

Gottfried Leibniz

First some history.

Gottfried Wilhelm Leibniz (1646–1716) was a German philosopher and polymath and an amazing thinker. Among many other ideas and discoveries, he invented the calculus (the derivatives and integrals we study in high-school mathematics) at the same time as Newton. Newton was more interested in calculus as a tool for physics and chemistry. But Leibniz was

somewhat less interested in science; he was not even much of a mathematician until much later in life. But he was a deep thinker intrigued by, among many other things, symbols and symbol manipulation.

Here is what he observed about arithmetic. We wanted to be able to do numerical calculations, for example, to figure out the area of a piece of land to be able to price it in an appropriate way. But numbers really are just abstract ideas. They have no physical presence, no mass or volume. How can we interact with the numbers to do the necessary calculation? The answer is *symbols*.

Leibniz realized that when we write down a number as a sequence of digits, we have a certain system in mind, decimal (base 10) numbers, where we use digits and the powers of ten in a very specific way. Every number can be written in decimal notation, but it is possible to write numbers in other ways too. (Leibniz is credited with having invented *binary* (base 2) arithmetic, the system now used by digital computers.) But most important, he insisted on keeping straight the difference between the purely abstract number on the one hand (like the number fourteen, say), and the much more concrete symbolic expression we actually write down (like 14 in decimal, or 1110 in binary, or XIV in Roman numerals).

He observed that in doing arithmetic, for example to figure out the area of a rectangle, we interact not with numbers but with their symbolic expressions. We take the expressions apart, cross parts out, add new parts, reassemble them, and ultimately produce new symbolic expressions from old. This is precisely the symbol processing seen in chapter 8. And as we saw, if we do our job right, the end result will be a new symbolic expression that

makes plain the answer we are looking for, whether it is the area of a piece of land or the age of Tommy and Suzy.

Of course what is essential about these symbolic expressions is not that we write them down. We can sometimes do all the arithmetic in our heads and, with certain limitations resulting from our rather poor memories, this can work too.

Leibniz wondered whether there were symbolic solutions of this sort to problems involving tangents and areas more generally. And the invention of the infinitesimal calculus (derivatives and integrals) is what came out of this.

An idea about ideas

Next comes the conceptual leap that only a genius like Leibniz could have come up with. Here is the story (or my version of it).

Thinking, as Leibniz realized, is going over certain ideas we believe in. But ideas are abstract too, just like numbers. What does it mean to say that Sue is jealous *because* she thinks John loves Mary? How can the idea of John loving Mary cause Sue to behave in a certain way? There is a physical John and a physical Mary, of course, but the idea of John loving Mary has no physical presence, no mass or volume. It might even be false, if the person who told Sue about John and Mary was lying.

As we have been saying all along in this book, Sue's behavior is conditioned by what she believes, and in this case, by a belief about John and Mary. Take that belief away, and sure enough, her behavior will change accordingly. But how can this possibly work? How can a purely abstract thing like a belief cause a person like Sue to do anything?

Leibniz has a proposal.

His proposal, based on his observation about arithmetic, is that we do not interact with ideas directly. We interact with *symbolic expressions of those ideas*. Leibniz suggests that we can treat these ideas as if they were written down in some (as yet unspecified) symbolic form, and that we can perform some (as yet unspecified) kind of arithmetic on them, that is, some sort of symbol processing, to go from one idea to the next. Of course we never actually write the ideas down on paper, we do it all in our heads, but the effect is the same.

In other words, Leibniz is proposing the following analogy:

- The rules of arithmetic allow us to deal with abstract numbers in terms of concrete symbols. The manipulation of those symbols mirrors the relations among the numbers being represented.
- The rules of some sort of logic allow us to deal with abstract ideas in terms of concrete symbols. The manipulation of those symbols mirrors the relations among the ideas being represented.

What a breathtaking idea! It says that although the objects of human thought are formless and abstract, we can still deal with them concretely as a kind of mental arithmetic, by representing them symbolically and operating on the symbols. When it comes time to think, when we have issues to resolve, conclusions to draw, or arguments with others to settle, we can calculate. As Leibniz famously put it in the Latin he often used in his letters, "*Calculemus!*" that is, "Let us calculate!"

What the Leibniz proposal does is offer a solution to what is arguably the single most perplexing feature of the human animal: how physical behavior can be affected by abstract belief. For the very first time, Leibniz has given us a plausible story to

tell about how ideas, including ideas that are not even true, can actually cause us to do something.

The knowledge representation hypothesis

Following Leibniz then, let us consider a system that is constructed to work with beliefs explicitly in the following way:

- Much of what the system needs to know will be stored in its memory as symbolic expressions of some sort, making up what we will call its *knowledge base*;
- The system will process the knowledge base using the rules of some sort of logic to derive new symbolic representations that go beyond what was explicitly represented;
- Some of the conclusions derived will concern what the system should do next, and the system will then decide how to act based on those conclusions.

Systems that have this basic design are what we are calling *knowledge-based*.

So what makes a system knowledge-based, at least according to this rough definition, is not the fact that its behavior is complex and versatile enough to merit an intentional stance. Rather it is the presence of a knowledge base, a collection of symbolic structures in its memory representing what it believes, that it uses in the way first envisaged by Leibniz to make decisions about how to behave.

The fundamental hypothesis underlying the McCarthy vision of AI is this: to achieve human-level intelligent behavior, a system needs to be knowledge-based. This is what the philosopher Brian Smith calls the *knowledge representation hypothesis*, which he presents (much more abstractly) as follows:

Any mechanically embodied intelligent process will be comprised of structural ingredients that a) we as external observers naturally take to represent a propositional account of the knowledge that the overall process exhibits, and b) independent of such external semantic attribution, play a formal but causal and essential role in engendering the behavior that manifests that knowledge.

Breaking this down into pieces, his version goes something like this. Imagine that there is a system whose behavior is intelligent enough to merit an intentional stance (a "mechanically embodied intelligent process"). The hypothesis is that this system will have symbolic structures stored in its memory ("structural ingredients") with two properties. The first property is that we—from the outside—can interpret these symbolic structures as propositions of some sort (a "propositional account") and in particular, as propositions that are believed by the system according to the intentional stance we are taking. The second property is that these symbolic structures do not just sit there in memory. We are imagining a computational system that operates on these symbolic structures (they "play a causal role in engendering the behavior"), just like the symbolic algebra and logic seen in chapter 8. In other words, the system behaves the way it does, making us want to ascribe beliefs to it, precisely because those symbolic structures are there in its memory. Remove them from memory, or change them in some way, and the system behaves differently.

So overall, the knowledge representation hypothesis is that truly intelligent systems will be knowledge-based, that is, systems for which the intentional stance is grounded by design in the processing of symbolic representations.

Is the hypothesis true?

The knowledge representation hypothesis is just that, a hypothesis. It may or may not be true. There are actually two interesting questions to ask:

1. Is there any reason to believe (or disbelieve) that humans are designed (by evolution) to be knowledge-based?
2. Is there any reason to believe (or disbelieve) that the best way for us to build artificial systems that have human-level intelligence, AI systems in other words, is to design them to be knowledge-based?

Unfortunately, neither question can yet be given a very definite answer.

When it comes to people being knowledge-based, it might seem ridiculously far-fetched to imagine that evolution would produce a species that depends in this precise and convoluted way on symbols and symbol processing. But many things seem to be unlikely products of evolution, at least at first. (The eye and the visual system is one such discussed by Charles Darwin.) It is obvious that written language is symbolic, and so evolution clearly *has* produced a species that can do all the processing necessary to make sense of those written symbols. We are the *symbolic species*, as the anthropologist Terrence Deacon puts it. It is perhaps not so far-fetched to imagine that an ability to use and process *internal* symbols is connected in some way with our ability to use and process the *external* ones.

It is worth remembering, however, that the knowledge-based question is a design issue. Even if people really are knowledge-based, we do not necessarily expect neuroscientists to be able to find symbolic structures in the brain for the reasons discussed in

chapter 2: we may not be able to reverse-engineer neurons. So even if we do come to believe that people are knowledge-based, it may not be because we have figured out how knowledge is stored in the brain. Rather, I suspect that it will be more like this: we will come to believe that only a knowledge-based design has the power to explain how people can do what they do. We will look at the design of artificial systems of a wide variety of sorts, and we will see that it is the knowledge-based ones that are able to produce certain kinds of intelligent behavior, behavior that will otherwise seem like magic. In other words, we will end up answering the first question by appeal to the second.

So what about that second question? Here the experts are quite divided. The knowledge-based approach advocated by McCarthy completely dominated AI research until the 1990s or so. But progress in GOFAI has been held back by what appears to be two fairly basic questions that remain unresolved:

- Just what kinds of symbolic structures are needed to represent the beliefs of an intelligent system?

and

- What kinds of symbol processing are needed to allow these represented beliefs to be extended so that they can affect behavior in the right way?

These might be called the *representation* and *reasoning* questions respectively.

Knowledge representation and reasoning

In 1958, when McCarthy first described his vision of AI and the research agenda it should follow in his "Programs with Common Sense" paper, he had in mind something very specific regarding

the representation and reasoning questions. He imagined a system that would store what it needed to know as symbolic formulas of the *first-order predicate calculus*, an artificial logical language developed at the turn of the twentieth century for the formalization of mathematics. And he imagined a system whose reasoning would involve *computational deduction*. The proposed system, in other words, would calculate the logical consequences of its knowledge base. Here is what he says:

> One will be able to assume that [the proposed system] will have available to it a fairly wide class of immediate logical consequences of anything it is told and its previous knowledge. This property is expected to have much in common with what makes us describe certain humans as having common sense.

In the time since then, many researchers including McCarthy himself have come to believe that these answers to the representation and reasoning questions are too strict. First-order predicate calculus is not ideal as a symbolic representation language, and logical consequence is not ideal as a specification for the sort of reasoning that needs to take place.

Indeed, the role to be played by classical *logic* in the answer to the reasoning question is a subtle and complex one. For very many cases, "using what you know" does indeed involve drawing logical conclusions from the beliefs you have on hand (as it did for Henry, for example, in the discussion of "Intelligent behavior" in chapter 3). But there is much more to it than that.

First, there will be logical conclusions that are not relevant to your goals and not worth spending time on. In fact, if you have any contradictory beliefs, *every* sentence will be a logical consequence of what you believe. Second, there will be logical conclusions that might be relevant but are too hard to figure out

without solving some sort of puzzle, perhaps using pencil and paper. (A relatively simple example of a logical puzzle was determining the guilt status of Bob, in the section "Symbolic logic" in chapter 8.) Third, there will be conclusions that are not logical conclusions at all, but only assumptions that might be reasonable to make barring information to the contrary. For example, you might conclude that a lemon you have never seen before is yellow, but this is not a logical conclusion since you do not believe that every lemon is yellow. (The unseen one might have been painted red, for all you know.) Finally, there are ways of using what you know that do not involve drawing conclusions at all, such as asking yourself what are the different things that could cause a lemon to not be yellow.

In sum, the gap between what we actually need to think about on the one hand, and the logical consequences of what we know on the other, is large enough that many researchers believe that we need to step back from classical logic, and consider other accounts of reasoning that would touch on logic somewhat more peripherally. As Marvin Minsky puts it: "Logical reasoning is not flexible enough to serve as a basis for thinking."

The fact that so much of what we believe and use involves assumptions that are not guaranteed to be true has prompted many researchers to focus on *probability* and degrees of belief (noted in the section "Knowledge vs. belief" in chapter 3), rather than on logic. After all, we clearly distinguish sentences that are not known for sure but most likely are true, from sentences that are not known for sure but most likely are false. But probability quickly runs up against the same difficulties as logic: again there will be conclusions that are most likely true but irrelevant; there will be relevant conclusions that are most likely true but too difficult to figure out; there will be conclusions that should be

drawn only in the absence of information to the contrary; and there will be ways of using what you believe that have nothing to do with drawing conclusions at all.

Turning now to the representation question, complications arise here as well. If the first-order predicate calculus first suggested by McCarthy is not suitable, what works better? We might consider using sentences of English itself (or some other human language) as the symbolic representation language. We would use the string of symbols "*bears hibernate*" to represent the belief that bears hibernate. Maybe it is enough to store sequences of English words in the knowledge base. English is what we use in books, after all, and information expressed in English is readily available online. Indeed, perhaps the biggest difficulty with using English as the representation language is in the second question, the reasoning. (The representation and reasoning questions are clearly interdependent.) Just how would a system use a knowledge base of English sentences to draw conclusions? In particular, making sense of those sentences (such as resolving the pronouns that appear in Winograd schemas) is a task that appears to require knowledge. How can English be our way of providing knowledge if using it properly already requires knowledge? At the very least, we would have to somehow unwind this potentially infinite regress.

The subarea of AI research called *knowledge representation and reasoning* has been concerned with tackling precisely these representation and reasoning questions in a variety of ways. But progress has been slow and is being challenged by research in other subareas of AI (such as AML) where symbolic representations of beliefs play little or no role. On the one hand, progress in these other subareas has been truly remarkable; on the other, none of them has attempted to account for behavior that makes

extensive use of background knowledge (as discussed in the section "The return of GOFAI" in chapter 4).

9.6 The only game in town

In my opinion, the relationship between knowledge and symbol processing is somewhat like the relationship in science between evolution and natural selection. Evolution is a scientific fact, well supported by the fossil record and DNA analysis. But natural selection, the actual mechanism for evolution proposed by Charles Darwin, is not seen in the fossil record or in DNA. Putting doubts about evolution itself aside, we believe in natural selection largely because it is such a plausible story about how evolution could take place, and there is no reason to think we will ever come up with a better one.

To my way of thinking, the use of background knowledge in certain forms of commonsense behavior (for example, in answering Winograd schema questions) is likewise a fact. It is a fact that a person can read about the release date of the movie *2001* on one day, and that this can affect what he or she does on another. Dogs can't do this, and neither can chess-playing programs or thermostats. But people can.

If we now want a mechanism to account for this fact, then, as far as I can tell, the knowledge-based story outlined in this chapter is really the only game in town. There is as yet no other good story to tell about how what you acquired about *2001* ended up staying with you until you needed to use it. The story might end up being a dead end, of course; it is quite possible that we will never answer in a completely satisfactory way the representation and reasoning questions it raises. At this stage, however, I see no alternative but to ask them.

In addition, if this knowledge-based approach is to ever work, that is, if there is ever to be a computational system that has access to and can use as much knowledge as people have, then it will need a massive knowledge base and a computational implementation efficient enough to process such massive symbolic structures. These impose serious constraints of their own.

I believe that any attempt to construct a large knowledge-based system without a correspondingly large development effort is doomed to failure. The idea of putting some sort of *tabula rasa* computer on the web (say) and having it learn for itself—that is, getting *it* to do all the hard, painstaking work—is a pipe dream. Learning to recognize cats by yourself is one thing; learning to read by yourself is quite another; and learning to read Wittgenstein, yet another. Before a computational system can ever profit from all we know, it will first need to be spoon-fed a good deal of what we know *and* be able to use what it knows effectively. Even if we knew how to answer the representation and reasoning questions, putting these ideas into practice on a large scale remains a daunting challenge.

But this is all speculation, really. In the end, what we are left with is best seen as an *empirical* question: what sorts of computational designs will be sufficient to account for what forms of intelligent behavior?

This is where this discussion stops and the AI research begins.