different bacteria is believed to be common, and to be a factor in the spread of resistance to drugs. Some consider horizontal gene transfer to have been important in the evolution of single-celled organisms. All these forms of gene transfer can be regarded as computations on the sum total of the genomes of a population, as can also the mutations of individual genes. The one-genome model can embrace all of these by having different internal mechanisms for producing variation.

There are many aspects of evolution that we do not address at all. Diversity in the gene pool may be a good defense mechanism against the unexpected and hence be critically important for survival. Survival is no doubt indispensable, but by itself it does not explain increasing complexity—not all mechanisms that are needed for survival are necessarily useful for enabling increasing complexity. The analysis here can be viewed as one that isolates the imperatives of complexity in evolution from the many other facets of biology.

Another aspect of our theory is that it presupposes a static world. It considers how a phase of target pursuit can be accomplished while the world is kept fixed. This is consistent with the notion that once a target becomes accessible and beneficial, evolution toward it will proceed quite predictably and rapidly. However, the theory can be adapted also to slowly changing worlds.[24]

I have sought a solution from the study of learning. This in retrospect is an obvious place to look. After all, machine learning is the general field that studies how complex mechanisms can be created without a designer. Darwin and Wallace were investigating a very important but special case of this.

Darwinian theory now pervades biology as well as many other disciplines. In biology evolution is identified not just with the Darwinian mechanism, but also with its apparent by-product, the history of life on Earth. This history has been filled with much drama, from the Cambrian Explosion and the Permian Extinction, to the appearance of creatures that can launch themselves into orbit around the planet. My concern here has not been with chronicling the history of these events. It has been only to understand one question: How can any mechanism account for this remarkable unfolding drama?

Chapter Seven

The Deducible

How can one reason with imprecise concepts?

> *True genius resides in the capacity for evaluation of uncertain, hazardous, and conflicting information.*
> WINSTON CHURCHILL

7.1 Reasoning

The tension between reasoning and learning has a long history, reaching back at least as far as Aristotle, who, as already mentioned, contrasted the "syllogistic and inductive" in his *Posterior Analytics*. In his treatise, however, Aristotle dealt almost entirely with the syllogistic, which may have triggered the high regard Western civilization has had for reason ever since. More recently, logical reasoning has fallen from the pedestal of highest repute. As has been pointed out often, humans are bad at logic. But that is not the only problem. Computers are very good at logic, yet we do not typically trust them for evaluating uncertain, hazardous, and conflicting information—and even when we do, the computer systems that succeed in this are usually based not on logical reasoning but on learning from large amounts of data.[1]

This chapter adopts Aristotle's dictum that beliefs come from two fundamental sources: syllogism and induction, or reasoning and learning. Despite the beating that logic has taken in recent years, my goal is to describe how these two sources can be unified into a consistent whole. In doing this, primacy will be given to learning, but reasoning will still remain essential.

In previous chapters, I have distinguished subject matter that is theoryful (in the sense that an explanatory theory of it is known) from that which is theoryless, and I have argued that beliefs about the theoryless have the semantics of PAC learning because they are acquired, in the first instance, inductively by learning. (Once one individual has learned such a belief, it may be transferred to another, but even then the semantics remains that of learning.) I shall now address the question of how reasoning with such theoryless information can be justified at all. This is of some relevance since, more often than not, humans reason about theoryless subject matter.

That reasoning makes sense and is theoryful for theoryful content has been long established by mathematical logicians. Following the work of George Boole and Gottlob Frege in the nineteenth century, much progress was made in understanding the forms that a mathematically rigorous view of reasoning can take. We can now ask questions about the nature and power of reasoning mechanisms within mathematics itself. Notions of truth and provability have been defined and distinguished, and questions have been asked as to whether, in a given system of logic, everything true is provable, and everything provable is true.

The mathematical logic that developed from this work has proven to be applicable to formal subject matter, particularly mathematics itself, and to have something significant to say about such subjects. By the beginning of the twentieth century the concerns of this field had moved center stage in the intellectual arena. Can all mathematical questions be translated into one unified formal language? Can any true mathematical statement so expressed, but no false ones, be deduced from a common set of axioms whose truth is self-evident? Some notable figures, including the philosopher Bertrand Russell and the mathematician David Hilbert, had believed that such a program could be carried through. To widespread astonishment, in 1930, twenty-four-year-old Kurt Gödel showed otherwise. In particular, he proved that in any rich enough logical system there were true statements that were not provable. This development, negative as it may have seemed, had profound intellectual impact. It was perhaps the first result that gave a glimpse of some ultimate limitations of what could be achieved by reasoning, even in a completely theoryful arena. Perhaps most importantly it led within a few years to the investigations of Turing and others into computation and *its* limitations.

These discoveries in mathematical logic, however significant they may have been in their own right, did not address directly the problem of reasoning about the theoryless. It was left to researchers in artificial intelligence, from the 1950s onward, to attack that problem.

The logical approach to artificial intelligence, pioneered by John McCarthy, treated the theoryless essentially as if it were theoryful. Axioms were to be constructed for any concept for which a word could be found in the dictionary. This included everyday concepts that were well outside the domain of any known science, and for which such axiomatization had never been attempted before. Rules of inference were then applied that were sound in the sense that they never yielded false conclusions when used within appropriate formal systems. This approach and its equivalents became the conceptual basis for much of the early work in the newly established field of artificial intelligence. More recently, analogous approaches have been pursued in probabilistic formalisms, where the rules of inference are those that apply in probability theory. These approaches treat the subject matter also as being theoryful, making them broadly equivalent to the logical approach.

While both logical and probabilistic modeling are mathematically principled when applied to the theoryful, they offer no principled guarantees when it is not clear how the models relate to the underlying reality, which is the case when the subject matter is theoryless. From a learning viewpoint, however, as we shall show, one can salvage some guarantees on the results of reasoning, even in this unpromising setting. The guarantees that can be achieved through learning are in the qualified PAC sense that, while errors are inevitable, their level can be controlled by putting in an effort commensurate with the quality of the guarantees that one is seeking.

It is important, I believe, to make a clear distinction between the two approaches—mathematical modeling using some kind of logic or probabilistic model, as opposed to learning. In practice it is easy to blur the difference by mixing the two. For example, consider a model, intended for use in a speech recognition system, of how people pronounce the words "yes" and "no." Such a model will be typically probabilistic. The question is whether the model is to be entirely programmed. If it is, then this would be treating the phenomenon as theoryful since one is attempting to have a model or theory of the outside reality. However, more often than not, after such a

model has been designed its numerical parameters are tuned by learning. If the resulting model turns out to be useful, the question arises as to whether that success was due to the learning or to the initial programmed model. As another example, we may start with a model for medical diagnosis that first incorporates some beliefs derived from interviews with physicians about the relationships among various physical conditions and symptoms. When tested against real data, the parameters that represent these relationships may have to be revised.

Regardless of the starting point, once we embark on learning from data we have to acknowledge that we are seeking the benefits of the learning phenomenon. It is then most appropriate to evaluate success by the criteria of accurate and efficient learning as described at length in Chapter 5. What is less clear, if success is to be measured by the criteria of learning, is why we need reasoning at all. There may be a portfolio of tasks, such as recognizing dangerous animals, walking up steps, or uttering appropriate greetings, that is sufficient for life. And perhaps these tasks can all be learned.

7.2 The Need for Reasoning Even with the Theoryless

For simple creatures it may well be that a repertoire of learned responses is sufficient. The application of a single learned circuit, which I shall call a reflex response, approximates the best behavior for a specific situation in life.[2] (Here, the word *circuit* is used in the same general sense in which I introduced it in Chapter 4.) Certainly, even for humans, such a repertoire of reflexes is often sufficient. In driving a car, experienced drivers are believed to cope by invoking reflex responses learned from similar previously seen situations. They do not need to go through an explicit reasoning process that considers the possible alternative sequences of events that would follow from alternative actions. Nevertheless, these reflex responses are not sufficient to explain all of intelligent behavior. There is a place for something more, and that something is reasoning. If, unlike mathematical logic or probabilistic reasoning, this reasoning is to be compatible with theoryless knowledge, it will need to be able to manipulate uncertain and unreliable knowledge in a principled way, so that some guarantees are provided on the accuracy of predictions.

The simplest form of reasoning that meets these demands is the application of two learned circuits, each having the semantics of PAC learning, in succession. Chaining two or more circuits may be effective in a range of situations in which it is unreasonable to believe that a single circuit might have been learned.

Consider the following example. If you are asked whether Aristotle ever climbed a tree, or whether he had a cell phone, you probably can answer both these questions with some confidence, despite not having previously had an opportunity to gain much statistical evidence for either question directly. It is implausible that in answering either of these questions you are invoking a single circuit that has been learned to recognize instances of some single concept such as cell phone ownership or tree climbing. Having a reflexive response to such a question would require a specially trained circuit in your brain that takes a name as input and outputs whether that person owns a cell phone or ever climbed trees. Implausible, indeed. It seems more plausible that you rather apply a sequence of circuits, each encapsulating a common sense rule. In this way you successively add more and more information to a picture in your mind. You would start by identifying Aristotle as a particular human who lived in a certain era. You would then apply some common sense rules in succession to make some deductions in order to build up a more complete picture. Ideas about children liking to climb trees and the era in which cell phones were invented would be expressed in these rules. Some of these rules may express facts, while many have theoryless content and will have been learned inductively.

A possible criticism of this example is that humans do not learn any such rule about tree climbing and cell-phone ownership simply because the issue is not of vital value to us. If making such decisions instantly were important, then perhaps we would learn reflex responses for them. Hence the possibility remains that reasoning is only useful for contrived questions of little importance. If animals survive in the main by performing tasks that are all reflex acts, then perhaps human reason is useful only for arcane puzzle solving and is not so fundamental after all.

However, more basic arguments can be put in favor of the extra power of reasoning. The chaining of learned circuits may enable capabilities unattainable by the application of a single learned circuit. For example, the needed single circuit may be of a form that falls outside the class that is learnable, but it may be equivalent to a pair of circuits that each can be learned separately. So, to continue with Aristotle, once we have recognized him as being

human, there may be a simple condition to decide whether he ever climbed trees. However, without first identifying the species of the individual in question, a general criterion for what can climb trees may be too complex to be within the learnable class.

As with PAC learning itself, the conclusions derived through reasoning will be permitted to be wrong sometimes, simply because the learned rules are permitted to be wrong sometimes. We can never be absolutely certain of our conclusions; Aristotle may just have aroused enough curiosity in the cosmos to have attracted extraterrestrial visitors bearing cell phones. Nevertheless, even in the presence of all these uncertainties, principled reasoning with some guarantees of accuracy is still possible. Furthermore, both the strengths and weaknesses of the guarantees are important.

7.3 The Challenge of Complexity

Several challenges to the endeavor of understanding reasoning have been encountered in the course of a half-century of research in artificial intelligence. I regard four—computational complexity, brittleness, semantics, and grounding—as the most pertinent to the approach presented here. My approach to addressing all these challenges is that of ensuring that there is some unambiguous relationship between the information represented in the reasoning system and what this representation refers to outside of itself. The relationship for this will be the same as for PAC learning, and will be called PAC semantics.

The first challenge is the impediment of noncomputability and computational complexity. In Section 5.1 I quoted Turing's reference to this, in the context when reasoning is equated with mathematical logic. Similar issues of complexity have been found to arise in other formulations as well. Indeed, Turing's proof that the Halting Problem is not computable can be viewed as an early warning of what has been called the doom of formalism. Expressing what we wish for in a formal framework is often futile if that framework is too broad to permit efficient computation. I do not accept that formalism itself is doomed. The challenge is to identify a formalism that works—one that is extensive enough for the task at hand, without being so extensive as to be computationally intractable. Just as for learning and evolution, we have to sail again between Scylla and Charybdis. Given the extent of our discussion of the issue of computational complexity already, I will leave this without further discussion here.

7.4 The Challenge of Brittleness

Over the decades extensive efforts have been made to imbue computers with knowledge so as to enable them to answer common sense questions. In the main the knowledge has been programmed by humans, and a formal logical system has been used to reason about this knowledge.[3] Overall, these efforts have had only limited success to date, and in situations not foreseen by the programmer computers regularly fail, giving responses that have often been unreasonable or even absurd. The impediment of computational complexity by itself does not account for this failure. Even when the intended, mathematically principled deductive processes have reached a conclusion in the time allowed, the computer's conclusions often fall short.

The reason for such failures must be that the programmed statements, as interpreted by the reasoning system, do not capture the targeted reality. Though each programmed statement may seem reasonable to the programmer, the result of combining these statements in ways not planned for by the programmer may be unreasonable.

This failure is often called *brittleness*. Regardless of whether a logical or probabilistic reasoning system is implemented, brittleness is inevitable in any system for the theoryless that is programmed. If the represented information is not consistent within itself and in exact correspondence with the domain being modeled, then no claims can be made about the accuracy of the deductions. While these systems are mathematically principled for theoryful content, they offer no useful guarantees for the theoryless.

The only way of avoiding this brittleness and achieving robustness is to have the systems learn. The predicate calculus and Bayesian probability are both well-founded mathematical systems. However, within these systems the issue of robustness in controlling errors in the face of limited data and limited computation is not addressed. Learning theory does address these very issues and is therefore a more appropriate basis for this enterprise.[4] Indeed, the advantages are two-fold, at least. First, PAC learning, by definition, is concerned with robustness to computation and data. It quantifies how the accuracy guarantees of the learned rules get stronger and stronger with more and more data and computation. Without some such guarantees little can be said about any system that is less than perfect. Second, a learning system has the fundamental advantage that it can check its predictions against the world. If it finds that it is making false predictions, it can adapt itself so that it will be more likely to be accurate in the future. With such

feedback the system can recover from almost any gaps or inconsistencies in its knowledge.

A closely related issue is that of resilience to noise. Humans can cope even in situations when some of the information provided is false, perhaps corrupted by noise during transmission. Happily, the basic PAC model, which does not discuss noise, is easily extended to accommodate it. Further, it has been shown that learning algorithms can be made resilient to certain kinds of noise in some generality.[5] For example, if one introduces noise by randomly changing the label of the examples with some probability, then classes that are PAC learnable in the absence of noise often (and SQ learnable classes always) remain PAC learnable in its presence.

Empirical efforts toward endowing machines with common sense knowledge have shown that the amount of knowledge needed is much higher than was ever expected. This is a yet further source of difficulty, but its sheer scale points even more forcefully to the need for principled automation, and hence learning, in the knowledge-acquisition process.

I do not claim that the learning-based approach I advocate here is without its challenges. Studies of learning and reasoning have shown that unless these problems are formulated very carefully, the computational complexity of each may become too large to be tractable. I shall come to a suggested resolution to this question, but not before discussing the remaining two challenges.

7.5 The Challenge of Semantics

I believe that no system that reasons with large-scale general knowledge can effectively work without there being a clear correspondence between the information represented in the system and the outside reality to which it refers. To understand or construct such systems one needs a principled view of this correspondence. It seems unreasonably optimistic to adopt an unprincipled view, and expect to meet this most basic requirement by pure chance.

If one programs a machine in terms of everyday concepts expressed in English, then one needs to be sure that each word is used in a consistent way throughout. Almost any word in a natural language has some range of meanings, and some words have several distinct meanings. If in a programmed rule words such as port, green, or conservative are used, then one has to be sure that every rule uses these concepts in exactly the same sense.

If several meanings of "port" are to be distinguished—a drink versus a mooring place, for example—then the variants have to be named (e.g., port1, port2) and used consistently. The difficulty of doing this consistently enough accounts for a significant aspect of brittleness.

PAC learning offers an approach to addressing this problem. At each instant for each concept, such as port2, the system will have a hypothesis or program for it that recognizes it, in the sense of saying, "Yes, this is an example of port2," or "No, it is not." This program will have been learned in terms of features that were already recognized. Some of these features may have been themselves learned previously. Others will have been preprogrammed, or, as is the case for living organisms, learned through evolution. Examples of this latter category are light detectors in the retina or pixels in a camera input device for a computer. But, whatever these features are, the end product in the overall system will be a recognizer for the concept of "port2." Inputs presented to the system can be labeled by this recognizer to indicate which are examples of "port2" and which are not.

If all the recognizers in a system are largely consistent in the sense that in most natural situations the labels attached to the inputs do not contradict what the recognizers say, then we can consider the system to be consistent in the PAC sense. If, however, contradictions are often encountered in natural situations, then the system can detect this for itself and seek to reach a more consistent state by modifying its recognizers. An important goal of learning is to reach PAC consistency in this sense.

For such PAC consistent systems the meaning of a concept is simply whatever the circuit labeled by that concept recognizes. Thus, after training to PAC consistency, the meaning of the concept port2 in such a system is nothing other than the function computed by the circuit that has port2 as its target. This circuit may involve other learned concepts but ultimately will depend on preprogrammed features that take external sensory inputs. The relationship between the function this circuit computes and the outside reality is one of PAC semantics.

7.6 The Challenge of Grounding

Finally we arrive at the fourth challenge, which I call grounding. It is intimately related to both semantics and brittleness, and it deals, to put it simply, with two primary issues: the scope of the knowledge that is claimed to

be represented, and the constraints of time, space, or other limitations within which the PAC semantics are to be accurate.

These concerns arise even for the simplest of logical statements. Consider the assertion, "All humans are mortal." A logician might phrase it, "For all t, if t is human then t is mortal," and abbreviate it as

$$\forall t \, \text{human}(t) \rightarrow \text{mortal}(t),$$

where the inverted A, the $\forall$ operator, denotes "for all." Even with this simple example there are some obvious difficulties. What exactly is the range of t that this statement applies to? If it is to apply everywhere in the universe, how can we be so sure that the assertion is true? Does it apply to humans described in fiction? This last question is not irrelevant if we wish, for example, to learn about the world from written text, since much text refers to fictitious individuals. The fact that Superman is fictitious is good to know if we want to learn only about real people.

We can always add some preconditions to statements that specify in more detail what scope is being asserted. For example,

$$\forall t \, \text{nonfictitious}(t) \rightarrow (\text{human}(t) \rightarrow \text{mortal}(t)).$$

This would ensure that human mortality is asserted only for nonfictitious humans. However, we have an infinite regress here. Are we sure this is a complete definition? More to the point, how do we define the terms nonfictitious, human, and mortal? Will not these have the same problem? Note that the complementary existential quantifier $\exists$, the mirror image of "E," which denotes "there exists," raises all the same issues.

The severity of this challenge can be appreciated even more if we recognize that human intelligence is applied effectively every day to issues with much less universality and permanence than this example. We interact with different people with different personalities and desires. We need to predict their behavior without having axioms that describe what they will do and under what circumstances.

We need a principled basis from which to approach this problem of grounding. PAC learning addresses this by identifying a specific distribution D with respect to which it learns and performs. Without some such notion of

grounding there cannot be a theory of learning. But how are we to specify this distribution D of typical situations from which a human individual learns? How is it ensured that in frequently occurring situations we will do the right thing, and how are we protected from making rash decisions that are not supported by our experience? Whatever the mechanism is, we need the same kind of protection in artificial systems. If we create a system to perform a task, but have no target distribution in mind on which we expect it to behave well, then we really have no idea what we are trying to accomplish.

To press the point in a different way, we conclude here with a paradox that illustrates the fallacy of discussing probabilistic events without defining what distributions they refer to. Consider the following proposal, and decide whether it would be profitable for you. You are to go up to a random person in the street and offer to compare the amount of cash you each have on you and agree that whoever has more gives it to the other. You convince yourself by the following (false) argument that this will be profitable. You argue as follows: "I have x amount of money. The other either has more, say y, or less, say z. (Ignore the case that they are equal, when there is no gain or loss.) The two possibilities occur with the same probability 0.5. If the other has more then I win y, otherwise I lose my x. Hence my expected gain is $0.5y - 0.5x$, which is greater than zero since y is greater than x." Would you get rich by repeating this? Since the other person could argue the same as you, and it is not possible for both players to have an expected win, the argument just given must be fallacious. But which one is the fallacious step in the above argument? Common sense gives a clue. In practice you would not play the game if you had just taken money out of the bank, but might if you were on your way to do that.

7.7 The Mind's Eye: A Pinhole to the World

In this section and the next I shall describe robust logic.[6] It is an approach to the reasoning problem that addresses all four of the obstacles. It formulates learning and reasoning with a common semantics, maintains computational feasibility, and provides a principled approach to the problems of brittleness and grounding. The device that enables all these issues to be addressed simultaneously is a quantitative formulation of working memory, a notion originally proposed in less computational terms by cognitive scientists. I shall call this computational version the mind's eye.

The notion that in the process of thinking we employ some special mechanism, other than our general long-term memory, for bringing together the different threads of the subject we are thinking about is a central idea in cognitive science. The mechanism is called working memory, and it is closely related to other notions such as short-term memory, imagery, attention, and consciousness. It has been researched from numerous perspectives, and one of the most striking findings is how limited it is. Its restrictedness was memorably demonstrated by the cognitive psychologist George Miller, who in his celebrated paper "The Magical Number Seven Plus or Minus Two" showed that we could hold only about seven objects simultaneously in this working memory.[7] The mind's eye, as colloquially used, is this same notion. When we are thinking, we are usually aware of very few things at a time. For our discussions it will suffice to recall the following earlier piece of introspection offered by the nineteenth-century polymath Francis Galton:

> When I am engaged in thinking anything out, the process of doing so appears to me to be this: The ideas that lie at any moment within my full consciousness seem to attract of their own accord the most appropriate out of a number of other ideas that are lying close at hand, but imperfectly within the range of my consciousness. There seems to be a presence-chamber in my mind where full consciousness holds court, and where two or three ideas are at the same time in audience, and an ante-chamber full of more or less allied ideas, which is situated just beyond the full ken of consciousness. Out of this ante-chamber the ideas most allied to those in the presence-chamber appear to be summoned in a mechanically logical way, and to have their turn of audience.[8]

Such a restricted "presence chamber" or mind's eye might seem limiting, but instead, I argue, it has a critical role in keeping within feasible bounds the complexity of the learning tasks that our cognitive system needs to solve.

In a conventional computer we have a very small fraction of the overall hardware investment devoted to the registers, where information is placed that is to be operated on and changed. The remaining much larger fraction of the hardware either stores information or moves it around. In a parallel computer there will be replication of processors and registers, but the fraction of investment in registers, as compared with communication and memory capabilities, is still small.

In biological brains the working memory is probably not as localized physically as are the registers in present-day computers. However, it is believed that working memory at any one time contains much less information than the total contents of the long-term memory. The number of visual concepts that a human may be capable of recognizing has been estimated, by counting the relevant words in a dictionary, to be around 30,000.[9] Total human long-term memory capacity is presumably much larger than this since not all concepts are visual, and since we can recall specific facts and events as well as concepts. Expert knowledge has been estimated to be much larger, perhaps by a factor of ten or more. In contrast, recall that George Miller estimated the maximum number of distinct entities that can be represented in short-term memory as being seven, plus or minus two.

The computational reasons for the amount of information that can be stored in computer registers being small are of two kinds. First, the circuitry needed to perform operations on the registers may be complicated, more complicated than needed for storage and communication. Second, if operations are performed on many registers at the same time, then some systematic way is needed for organizing the cacophony of results that emerge—which is the problem of parallel computation.

Both registers in computers and working memory in brains bring together pieces of information in new combinations, to get results that may never have been computed before. In a computer we may wish to multiply two numbers retrieved from the computer's memory. In the working memory of a brain we may wish to predict the consequences of a novel combination of actions, to see, for example, whether that combination has a promising enough outcome for us to justify doing those actions the next day. In order to predict these consequences a variety of related pieces of knowledge may have to be retrieved from long-term memory.

The computational challenges that arise for registers in computers arise equally in biology.[10] The circuits needed to maintain the working memory may be complicated, as may be also the task of coordinating all that is happening. All this puts a ceiling on how much information the mind's eye can reasonably handle at any time.

I believe that these computational imperatives for having a small working memory, however constraining they may be, are by no means the ultimately

limiting ones. Even more severe constraints are imposed by the fact that the brain does not merely have to compute, but also needs to learn. The smallness of the field of view of the mind's eye is essential to make the world learnable. The more information we attend to at a time, the more complex is the task of abstracting regularities from it. Apparently seven (plus or minus two) strikes a useful balance between scope and efficiency. Having our consciousness streamed through such a small aperture serves the function of permitting learning.

Having a small mind's eye forces us to look at the world effectively through no more than a pinhole. Between the world and its enormous complexities and our memories with their highly complex contents is placed this very limited field of attention. As a result, we are forced to manipulate this limited field with some care. We make choices about where to cast our gaze next, what to think of next, and what knowledge from our long-term memories to bring to bear on our thoughts. Making these choices is challenging since, as we have to presume, they will also be based on the restricted amount of information available in our mind's eye.

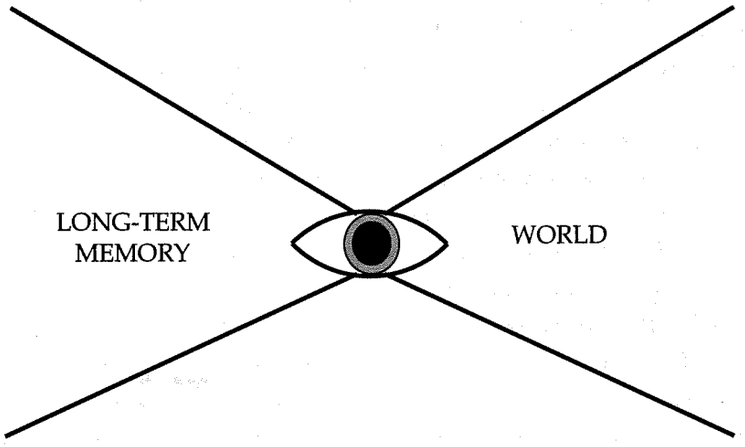

Figure 7.1 The mind's eye is shown occupying a metaphorical pinhole between two funnels, one facing the world, the other the individual's long-term memory. We regard the mind's eye as a computational device that contains the information of which an individual is conscious at any time. The basis of learning is the data that streams through the mind's eye.

The mind's eye therefore can be viewed as the focus of an information funnel between the world and the thinker. It is a two-way funnel that restricts information flow from the outside world as well as from the long-term memory. It corresponds roughly to information we are conscious of, but may include more. It summarizes each experience succinctly enough so as to make both computation on it and learning from it tractable. This succinct summary is informed by both the external input and the internal long-term memory. The succinctness of the description is important for addressing complexity. By permitting learning, it also addresses brittleness.

But most crucially, the mind's eye addresses both semantics and grounding by defining the arena to which all learned knowledge refers: our learned knowledge is derived from real-world experience, but only after filtering, and only to the extent that it is ever represented in the mind's eye. Our learned generalizations have validity (in the PAC sense) for the distribution of the contents of our mind's eye that is generated as we go through our experiences. This then is the semantics and grounding we ascribe to cognition.

7.8 Robust Logic: Reasoning in an Unknowable World

We are approaching the goal of describing the system of robust logic that addresses the four challenges to reasoning. To review, any such system that is to model cognition needs to satisfy two requirements:

(i) All the learning and reasoning processes need to be computationally feasible, in the sense of being polynomial time in the appropriate parameters. The learning process needs to be robust in the PAC sense, as opposed to being brittle, so that any errors in the knowledge can be reduced after sufficient further exposure to the environment to which the knowledge refers. The learned knowledge needs to have clear semantics and grounding.

(ii) Reasoning needs to have a principled basis, in the sense that if two pieces of knowledge each having some PAC accuracy guarantees are applied in succession, then any conclusion so derived should inherit some accuracy guarantees also.

To address these requirements, we have the mind's eye, which we shall now discuss a little more formally than we have so far. Let us call the contents

of the mind's eye at an instant a scene. A scene contains a fixed number, say twenty, of undifferentiated tokens, denoted by $t_1, \ldots, t_{20}$, and there is a fixed set of relations that may hold for various subsets of the tokens in a scene.

Each token can be associated temporarily with anything that our mind's eye is then contemplating. The relations come from a fixed set that the system knows about at the time. Suppose that in a particular scene the relation "elephant" is true for t_1, the relation "peanuts" is true for t_2, and the relation "likes" is true for the pair t_1, t_2, in that order, meaning that the elephant represented likes peanuts. These three relations applied to these tokens would represent the contents of the mind's eye at one instant, as illustrated on the left-hand side of Figure 7.2.

Robust logic has mechanisms for learning and reasoning. The novelty is that learning and reasoning will be based on the *same* semantics, in particular PAC semantics. That is the key.

In classical logic a rule would be written as

$$\forall t_1\ \forall t_2\ \text{elephant}(t_1)\ \textbf{and}\ \text{likes}(t_1, t_2) \rightarrow \text{peanuts}(t_2).$$

The $\forall$ symbol again means "for all." The statement would be interpreted to mean that for any two things t_1 and t_2, if the first is an elephant, and the first likes the second, then it follows that the second is peanuts. In other words, if an elephant likes something then that thing is peanuts.

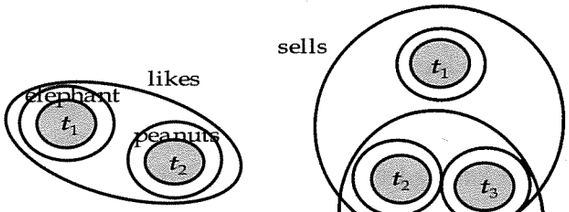

Figure 7.2 Two scenes. The left-hand panel is discussed in the text. The right-hand panel is a little more complicated, and shows a transaction where a person t_2 sells an object t_1 to another person t_3 and is given money t_4 by t_3 in return.

Unfortunately, this kind of semantics is somewhat alien to PAC learning. One reason is that this logical statement is only a one-way implication. It does not intend to imply that being a peanut necessitates that all elephants like it. However, when learning a concept, we want a two-way implication—the learned concept should be a good approximation of the target concepts both when the concept is true and also when it is false. For this reason we shall need to learn rules of the form

$$\text{"complicated condition"} \equiv \text{peanuts}(t_2),$$

where the $\equiv$ symbol denotes such an equivalence, or two-way implication. Note that these equivalences can be used for reasoning in exactly the same way as the logical implications. Whenever the left-hand side is satisfied, we can deduce that the right-hand-side assertion peanuts(t_2) also holds, at least probably.

Now one advantage that machine learning offers is that the "complicated condition" in rules like the one above can be arbitrarily intricate, as long as it can be acquired by learning. In that spirit, we shall allow rules of the general form

$$F([\exists t_1\ \text{elephant}(t_1)\ \textbf{and}\ \text{likes}(t_1, t_2)], \ldots.) \equiv \text{peanuts}(t_2),$$

where the function F is from a class C of functions that is PAC learnable and its arguments are certain restricted expressions to be described later. In this instance the first argument of F asserts that there exists ($\exists$) something (t_1) that is an elephant and likes (t_2). The intention of such a rule is to *predict* for any particular scene whether the set of tokens named on the right-hand side (just t_2 in this example) has certain properties, such as that of being peanuts.

For each word variant in the dictionary, for example peanuts, we can imagine having a rule with that word variant on the right-hand side. The left-hand side will amount to an approximate definition in terms of other words. More concretely, the left-hand side of each rule will express a criterion on scenes for the concept on the right-hand side to hold. The left-hand side may be very complicated and would typically enumerate all common conditions to each of which the answer "peanuts" is a reliable one. In this example one of the many such conditions may be "What to say if asked what elephants like to eat."

In rules of this general form, the left-hand side will be learned from examples in the PAC sense. The learned function F may be complicated—the conditions that provide evidence for something being a peanut may be multitudinous and complex. No human needs to be consciously able to describe it. All that matters is that the function F be from a learnable class C, in which case even large ugly expressions that capture all common forms of evidence of peanuts can be learned, both by computers and brains. It is not implausible that our brains are full of such circuits. After all, it is important for humans to take a position very fast, in a few hundred milliseconds, on whether what we are seeing is a peanut or a tiger.[11]

Robust logic is defined so that rules can be learned and reasoned with in polynomial time. A persuasive candidate for the learnable class C appears to be the following. The left-hand sides are defined to be the class of functions that can be expressed as linear separators, so as to be learnable, but where the variables now are independently quantified expressions (IQEs) such as "$\exists t_1$ elephant(t_1) **and** likes(t_1, t_2)." This last example is an instance of a *schema* that consists of one $\exists$ ("there exists") symbol and is the conjunction of two relations, one with one argument and the other with two, with one token shared between the two and the other quantified. Put a different way, we obtain other members of the same schema from this instance by substituting any other relations for elephant( ) and likes(,), provided they have the right number of arguments.

IQEs have the following two contradictory aspects. On the one hand, they are quite powerful in being able to express complex relationships among objects in a scene. On the other hand, they are simple in that given a specific scene, such an IQE will be either true or false for that scene, and it is easy to determine which one is the case. For that reason we can treat each IQE as a Boolean variable that for any example scene takes value either 0 or 1. In this way we can treat IQEs as features in a standard PAC learning setting and use whatever learning algorithm we like. For example, we can interpret the Boolean values 0 and 1 as numbers and use the perceptron algorithm with these IQEs as features.

We can have IQEs based on more general schemas than the one illustrated, with, say, three or four relations rather than two. However, that could incur much higher computational costs in learning and reasoning. In particular, if we use a schema but have no information about which of the

IQEs defined by it are relevant, then we have to entertain them all during learning. This consideration limits the size of the schemas that are useful in practice.

Where we are heading here is the following. Learning will be done by a conventional learning algorithm, such as the perceptron algorithm, but the variables will now be these IQEs. Reasoning on any one scene will be done by invoking any rules whose left-hand sides are satisfied by the scene, updating the scene with the relations in the right-hand sides of those rules, and repeating this process as appropriate.

The reader might be asking by now: But what does this notation mean? What is the quantification over? Does the $\exists$ mean "there exists somewhere in the universe?" Does the $\forall$ mean "for every object on Earth?" No. Either would violate our desire for grounding. The symbol $\exists$ means simply existence in the one scene in question, and $\forall$ means universal for every object in that one scene. In other words, to make the IQE "$\exists t_1$ elephant(t_1) **and** likes(t_1, t_2)" true for token t_2 in a particular scene, there must exist some token t_1 in that scene such that the relations elephant(t_1) and likes(t_1, t_2) hold in the scene. Similarly, the $\forall$ symbol denotes all objects in the scene, not the universe.

In general, several variables can be quantified, some existentially ($\exists$), and some universally ($\forall$). However, the quantifiers in the different IQEs have to be interpreted independently of each other. This means that one cannot assert *directly* that there exists one token that satisfies two IQEs simultaneously. If one wants to assert that, then one has to extend the allowed schema to allow the needed combination of the original two IQEs as a single IQE of possibly double the size, and to accept the greater computational costs that would follow.

These definitions are construed so that, given the left-hand side of a rule and a particular scene, one can evaluate which IQEs are true and which are false for that scene, and hence determine whether the left-hand side holds for that scene. In other words, one can determine whether a rule applies to a scene from information in that scene. No other knowledge is required.

Once we are in this position, we can use any conventional learning algorithm for the chosen learnable class. If it is the class of linear separators, then we may use the perceptron algorithm. Our method of generating all possible IQEs for a fixed schema (i.e., by replacing elephant with any of the other words allowed) creates large but still polynomial numbers of IQEs.

has to have a role. Furthermore, any reasoning system for the theoryless, including the human system, will suffer from the same frailties as does our robust logic. Chaining together beliefs that we believe to be probably approximately correct can be justified, but the conclusions will also be only probably approximately correct, and the longer the chain of reasoning the larger the errors that we shall have to accept.

This concludes our formulation of the three phenomena of learning, evolution, and reasoning from learned data, in terms of ecorithms. Ecorithms comprise only a subset of the computations that Turing universal machines can execute. But up to Turing's time it was this subset that had domination on Earth.

Chapter Eight

Humans as Ecorithms

> *No, I'm not interested in developing a powerful brain. All I'm after is just a mediocre brain, something like the President of the American Telephone and Telegraph Company.*
> ATTRIBUTED TO ALAN TURING

8.1 Introduction

Science, whether of nuclear reactions or the impact on health of smoking, does not dictate how it should be applied. Its relevance needs a separate, if theoryless, discussion. Scientists are justified and perhaps obligated to speculate on the broader relevance of their work. This is the excuse that justifies what follows here.

The previous chapters have expounded the thesis that a decisive determinant of life is the ecorithmic relationship between living organisms and their environment—life coping with its environment by means of learning mechanisms. In the remaining chapters I shall try to provide a personal, and alas theoryless, answer to two questions: Can all the complexities of life, intelligence, and culture that we witness on Earth be explained by this hypothesis, and do any consequences of general interest follow from it? Necessarily the discussion will be much more speculative than before.

One difficulty with making this discussion more theoryful is that the fundamental ecorithms used in biology have not yet been identified. A major motivation of our study, of course, is exactly to encourage further work toward filling those gaps. Once these ecorithms are better understood, the topics we are about to discuss will become more amenable to scientific analysis.