3

Problems and Pitfalls in Developing AI

Artificial intelligence research has progressed significantly since the last years of the twentieth century. Computers today are more powerful than they were in the past, and their software is more complex. Still, although technology is more advanced, artificial intelligence knowledge seems the same as it was a few decades ago. As technology grows, appreciation of the field's challenges grows, too.

One problem is that both the brain and the way it handles information is more complex than earlier imagined. Scientists have known for decades that each neuron has connections to tens or even hundreds of other neurons. However, it was not until recently that researchers learned this network of connections is constantly changing. The changes occur as humans gain new information or experiences. Scientists are also still trying to understand what it is about the brain that lets people feel emotions. What is it about the brain that helps us recognize a face, even when it's half-hidden by a beard? What helps us to find a new way to solve an old problem? What makes it possible to write and appreciate beautiful music? In spite of more than a century of research, much of the way that the brain works is still a mystery. Until researchers really understand all of this, it will be hard to recreate a human brain in hardware and software.

SPEECH AND SPEECH RECOGNITION

Many animals communicate with each other. There are many examples of this in the animal kingdom. Dogs bark to each other. Birds and rodents make alarm sounds when they see predators. There is, however, a difference between communication and language. A mouse squeak that means, "There's a predator—hide!" is not the same as having a conversation about football with your friends. Although animals communicate, as far as is known, only humans have language. Unless a computer or other machine can learn to carry on a conversation, it can't really be considered intelligent. (Remember: This is part of the Turing test.) It is possible to type commands into a keyboard, but it would be easier if we could talk to an intelligent computer or robot the same way that we'd talk to another person. Today's research focuses on getting computers to recognize and understand human speech.

Consider the example of a computerized phone system: We go through all sorts of menus to finally get the information we need, or to speak with a person. Many of these systems use voice recognition. We speak into the phone and the computer on the other end seems to understand what we say. For example, we can speak a number into the phone or say "yes" or "no" and the computer makes the correct response. Speech recognition programs are also available that let us speak into a microphone to make a computer follow our commands or create the text of what we say. The sounds a person makes are changed into electrical impulses, and these in turn prompt commands in the computer. With all of this, it seems as though computers have mastered speech recognition and that we have nothing left to develop. This is far from the truth. Speech recognition is not a done deal. It works well under very controlled conditions (reading numbers or giving simple answers), but in the much more complicated world of following a conversation, speech recognition still has a long way to go.

One thing that makes speech recognition a challenge is that people say words differently. Every pronunciation can sound like a different word to the computer. A computer will do *exactly* what we tell it to do, no more and no less. Even if a computer is programmed to recognize the word *either*, it might not recognize the way a particular person pronounces it. Some people say "ee-ther" and others say "eye-ther." For a computer to recognize a single word, it has to be programmed to understand all of the different ways that people might pronounce it. A computer must be programmed to recognize a New York accent, a Georgia accent, a stuffy nose, a pronunciation mistake, or a little hiccup in the middle of a word. Just to recognize the sound of speaking the numbers zero to nine, a computer might have to be programmed to understand a hundred or more variations. The same can be said of any other batch of individual words. This is why many computer telephone systems stick with very simple lists of single words or numbers and yes and no questions.

There are even more problems that come up when trying to understand the spoken word, at least for computers. Think of all the ways that people speak when they're in a hurry or when they're talking to friends. "Yes" might turn into "yeah" or "yep," or maybe "sure thing" or "you bet." The computer would need to insist that everybody say the one word it is programmed to recognize. Or, the poor programmer has to think of all the different ways that people might say "yes" or "no" or whatever one-word answer is given.

When one thinks of all the ways people can say and pronounce the simplest words, it's a wonder that computers can understand anything at all. Besides that, this level of speech recognition still isn't really artificial intelligence. It's just programming a computer to recognize a specific pattern of sounds as a word, instead of recognizing the letters that a person types. To a computer, understanding the sound "yes" is the same as understanding the typed letters Y-E-S. It is also the same as when a computer user clicks a box marked "yes."

For most people, understanding speech is more than recognizing individual words. It means understanding sentences and paragraphs as they are spoken. This is even more difficult than recognizing individual words. In addition to all the different ways that words can be pronounced, stringing them together into sentences makes things even harder for the computer. When people talk in sentences, they tend to run words together, shorten words, and use contractions. For a computer to recognize speech, it has to be programmed to understand that "can't" means the same as "cannot" and that "didja" means the same as "did you." In addition, there are all the different ways people can put sentences together that all mean pretty much the same thing: for instance, "What time is it?" versus "Hey buddy, you got the time?" and the other stray sounds that find their way into speech ("um" or "uh," for example). Therefore researchers must program a computer to understand that the sound *um* is meaningless when it's a separate word, but that it can also be part of the word *gum* or *bum*.

The next issue might be the hardest to program into a computer. People tend to run their words together when they talk. We hear the individual words because we understand the language and the way it sounds. If you listen to most people when they're talking, you'll notice that the words normally come out without much in the way of gaps between them. We don't say, "Hi. How. Are. You?" when we talk to other people. It comes out more like, "Hi, howareya?"

For a computer to understand what people are saying, it has to figure out where one word ends and the next begins. This isn't impossible. There are software programs that can take what a person says and turn the speech into text. These programs work best when each person "trains" the software to recognize his or her individual speech. This is a process that can take many minutes of speaking to the computer until it can recognize the peculiarities of each person's speech. This works when a person has the time and desire to work for a while with the computer. It won't work with, for example, an artificial intelligence computer that's designed to talk with hundreds or thousands of people every day. For a system that will really understand natural speech—the way people normally talk to each other—a computer has to be able to quickly sort out all of these matters.

As of today, this is not possible. It's a problem that has to be solved if we're going to be able to speak directly to intelligent computers someday. Speech recognition is one of the obstacles of artificial intelligence research. There are others, as well. The following sections will cover other issues that cause problems for artificial intelligence researchers.

CREATIVITY

Creativity is another of the characteristics that seems to make humans unique. Creativity is often thought of as writing novels, composing music, and taking part in other artistic activities, but there is much more to creativity than this. Using a hammer as a paperweight or to prop open a door is a creative act. This is because the person who does this realizes that he can use this tool for something other than the way it is usually used. The ability to be innovative and come up with new ideas is, as far as we can tell, something that not many animals can do, and non-human animals exhibit much less creativity than humans do (for example, a gorilla might realize it can use a twig to get termites out of a termite mound but it can't develop a new theory in physics or compose music). It is also something that we will have to figure out how to program our computers to do if they are going to be genuinely intelligent.

Before discussing creativity in machines, it is necessary to discuss more completely its meaning in humans. One of the challenges to understanding creativity is that it is shown in many different ways. Writers, painters, composers, and scientists will sometimes struggle for months or years, trying to get past a sticking point in their projects. Sometimes they solve their problems one step at a time, like building a house one brick at a time. At other times, inspiration strikes and a person can leapfrog to a solution in a moment. One example of the latter is the story of English physicist and mathematician Isaac Newton (1643–1727) and an apple falling from a tree. Soon after, Newton made a creative leap that helped him understand an important aspect of how gravity works. No matter how it happens, the creative process involves coming up with a new idea. This new idea can be using a hammer to keep papers from blowing away, or realizing that an apple pulls on Earth the same way that Earth pulls on an apple.

The key word here is *new* idea, because coming up with something new is the main problem computers have. We can program computers to follow a set of rules. Yet creativity often involves developing new rules or breaking some of the rules we've learned—for example, using a hammer as a paperweight breaks the "rule" that hammers are used to hit nails.

This is a problem. If computers only know how to follow the rules that we give them, is it possible for these rules to include instructions on when it's okay to break the rules? Although confusing, it is a question that is at the heart of this aspect of artificial intelligence research. We can't tell a computer that it's okay to break every rule. We also can't tell a computer that it's never okay to break a rule. Creativity is being able to understand *which* rules need to be broken to solve a problem. Breaking rules might help a person to develop a new scientific theory or to compose an original piece of music. Creativity also involves knowing *how* to break the rules in a way that helps to solve the problem.

There are different ways programmers work on this problem. One way is to program the computer to make minor, random changes to its program and to see if these small changes make the program better or worse when the computer tries to solve a problem. This is how evolution works. Mutations are minor changes in the information encoded in our DNA. Most of the time these changes are harmful or they have no effect at all. A small fraction of the time, they produce an improvement. The case is the same for the rules programmed into a computer: If the computer makes very small changes at random, most changes will have no effect, or they might even keep the program from running at all. Every once in a while, however, slight changes pave the way for answers.

Let's use the example of a hammer once again. Suppose the computer is loaded with the rule "hammers are used to hit things." Now suppose that, at random, the computer changes *hit* to *hold,* so this rule changes to "hammers are used to hold things." With this small change to one of possibly hundreds of thousands of rules programmed into it, the computer suddenly has a rule that will make it possible to use a hammer as a paperweight. (The purpose of a paperweight is to hold papers in place or to be used as a doorstop, which holds open a door.) Adding one rule—that a computer should make small, random changes to its set of other rules—makes it possible for a computer to be creative . . . at least up to a point.

This sort of experimenting on the part of the computer can be very useful. However, there are two problems present. One is that the computer needs to have a way to know which changes are actually useful and when they can be used. Also, if a computer is making changes at random, there's no guarantee that the changes will go in the direction of a solution to a particular problem. It might, of course, get to an answer by sheer luck.

It is possible to use this process in a more purposeful way. After each random change, the computer could evaluate whether it's closer to a solution than it was before. If the change brings the computer closer to a solution, then the computer can concentrate its next changes in the same direction. By doing this, a computer could work towards a solution to a problem in much the same way as a person would work.

Figure 3.1 This computer-generated art was created based on a mathematical function called the Mandelbrot set, which was created by a human.

These two approaches are similar to how people solve problems. The random changes in computer programs are like flashes of insight. The more directed approach is similar to how a person works steadily towards an answer. The important thing with both of these approaches is to make sure the computer can recognize when it has reached an answer.

There is one sort of creativity that helps people solve problems or recognize other ways of using tools and objects. There is another, more common definition for creativity: writing stories, composing music, creating art, and other artistic activities. Computers can be programmed to do all of these things. So far, though, computer-generated art and music are more successful than computer-generated writing. There are some general rules for both music and art: which colors and shapes go together, which musical notes sound okay together, and so forth. There is still, however, a lot of room for flexibility in these rules.

Think of all of the different types of music—jazz, rock, classical, pop, rap, and more—and think of all the different styles within each of these genres. With art, look at the differences within a group of paintings. There is an infinite number of ways to put together a painting or song that will please an audience. We can program a computer with a set of rules for painting. We can also program it to choose colors and shapes at random, while following rules of composition and color matching. So far, however, computer-generated art and music at their best are okay. Computer-generated masterpieces have not yet been created.

Writing, on the other hand, seems to be a little harder to program. There are all sorts of rules for writing that can be programmed into a computer. There are rules for grammar, vocabulary, sentence structure, and more. What's difficult is to come up with a set of rules for how to make believable characters, place them in believable situations, and tell a story that readers can enjoy. Computers can follow grammar rules, but creating realistic characters who interact with each other and with situations in a believable way calls for an understanding of how people think and behave. This is something that computers can't yet do on their own, although there are some computer programs that can help to write books by providing the outlines of basic sentences, paragraphs, and plot points. These programs can help a person put together a book, but the creativity and the character development—making the characters come across as human—still have to come from the human author. In other words, to date, computer-generated writing is not very good.

Taking all of this into consideration, computers do show some signs of creativity. However, this creativity isn't yet up to human standards. Right now, researchers are not sure if the solution to the problem is faster computing, better programming, or something else. Is creativity a matter of power and programming? Is there a creative spark that only intelligence can have? For the moment, we don't know, but artificial intelligence scientists are working to find answers.

There is one last thing to mention before moving on. There is a difference between breaking rules in the way we're discussing here, and breaking rules that parents, lawyers, or teachers have set. One part of creativity is the breaking of what are called *conceptual* rules. These are the rules about how we view the world. Such rules are different from the ones that help keep us safe and allow us to live as members of society. Researchers are trying to teach computers to be able to break conceptual rules. For example, the rule that "hammers are only used to hit things" may not be programmed into code so that a computer can think of a hammer as a paperweight or a doorstop. Researchers are *not* trying to teach computers to break the rules that say it's bad to steal from others, or that bedtime is at 10 P.M.

ARTIFICIAL SENSES

Yet another problem for the AI researcher is the problem of senses. It is possible to hook up a computer with cameras and microphones that help it to "see" and "hear." They can also be connected to sensors that give them a sense of touch. Robot hands can pick up an egg without breaking it. With **tactile** sensors, a computer can tell if an object is hard or soft, rough or smooth, fragile or sturdy. In fact, engineers have even come up with mechanisms that can tell the difference between different molecules that are floating in the air or are dissolved in water. Artificial taste and robot "noses" are being used to search for explosives, toxins, and other dangers.

People understand the world in which they live thanks to their senses. Try to imagine what life would be like if we couldn't see,

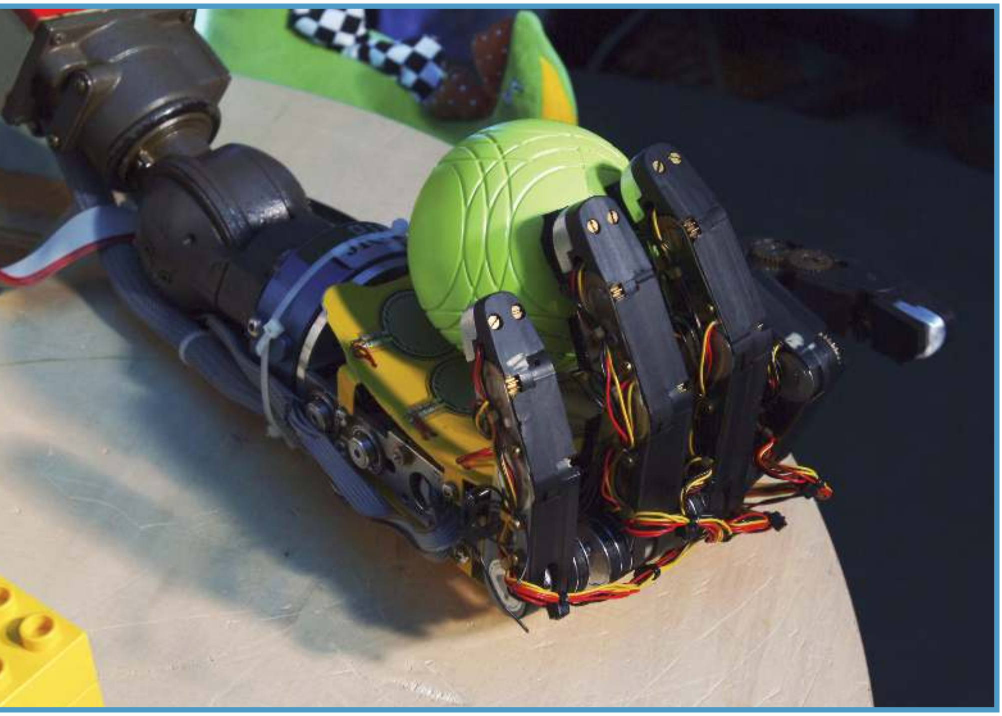

Figure 3.2 A BabyBot robot hand grasps a ball. The BabyBot is a humanoid robot that has been developed at the LIRA-Lab at the University of Genoa in Italy. The BabyBot project, started in 1996, is investigating sensory and control systems, how they interact, and what this can tell us about human brains and how they develop. The hand has five fingers, fifteen joints, and tactile sensors that react by producing a "grasp reflex." Touching one of these sensors (seen under ball) makes the hand close in a grasp, as seen here.

hear, smell, taste, or feel anything. It's almost impossible to think of any creature that can be intelligent without having some way to sense the world around it.

We can give senses to machines so that they can relate to the world around them. The next question is what they can be programmed to do with that information. There are robots that use their senses to hold parts together with just the right amount of pressure during the assembly of a machine. These robots can also use cameras to help them see what they are holding. Another kind of robot, called RABiT, uses cameras and manipulators to analyze laboratory samples. RABiT's analysis helps with research in medical and biological laboratories around the world. Then, of course, there are the speech recognition systems that were mentioned earlier, which use microphones as "ears." With proper programming, these can be used to recognize any sounds, not just speech.

We can give our machines ways to sense the world around them. We can even program computers to make some sense of what

Figure 3.3 A researcher uses a Rapid Automated Bacterial Impedance Technology (RABiT) machine, which is an automated device for measuring the number of microbes in a culture. He is moving a sample that contains microbes into one of the machine's tubes. The tube can be kept at a constant temperature while the microbes multiply by feeding on a nutrient gel. The growth of any microbe colonies can be monitored over time and the results displayed as a graph on a computer screen.

The Challenge of Artificial Vision

For more than a century, cameras have existed to capture images. Yet artificial vision is still an elusive goal. The reason for this is that there is a difference between capturing an image and using that image. Any camera, or eye, can capture an image. However, until that image is interpreted by a mind (or software), it is only an image. The difference between imaging and artificial vision is that vision includes understanding what we've seen. It also includes being able to make use of the image in some way. Facial recognition is one of the ways that artificial vision is being used now. It captures almost all of the challenges of computer vision.

Say, for example, we're trying to program a computer to recognize faces. A digital camera can capture images. Even a simple digital camera can figure out which part of the image is a face. Yet, at first glance, many faces are relatively alike. Most people have the same number of eyes, noses, mouths, and ears arranged in the same pattern. Most of us have hair on the top of our heads. Some of our features might really stand out, such as baldness or having a beard. Still, many of the characteristics that help

their senses are relaying to them. But can we say that computers are actually hearing, seeing, feeling, smelling, and tasting? Or are they simply receiving digital data—the same as when we click on a box for "yes"—without understanding anything of what their senses are receiving?

On the other hand, we can say the same about us: Our eyes, ears, fingers, noses, and tongues produce electrical signals that are sent to the brain in the same way as artificial sensors send electrical signals to a computer. Our brains take these electrical impulses and turn them into images, smells, and other sensory information. Then, our brains turn these impulses into information that means something to us. Is what our brains do with the impulses from our eyes any different from what a computer does with the impulses from a camera? Or maybe a different way to think of it is to ask if *sight* means us to tell the difference between faces are minor: the color of our eyes, the exact shape of a nose and mouth, how far our ears stick out from our heads. It's the sum of all of these minor differences that helps us to tell the difference between faces.

This raises a problem for computers. How can a computer recognize these differences, especially when some things about our faces can change? For instance, we can cut our hair, grow or shave a beard, use colored contact lenses, or wear lipstick. So with all that can change, how do we decide what makes a face recognizable as, say, your sister instead of her best friend?

One way to do this is to program computers to recognize characteristics that don't change. These characteristics might include the exact shape of a person's eye sockets and chin, the distance between their eyes, and so on. Facial recognition software has been used by law enforcement in London and by the U.S. State Department. It has even been used in a few nations to stop voting violations during elections. With good visual conditions, the most recent systems are even able to tell the difference between identical twins.

receiving the images or understanding what they mean. Does this happen in the eye and brain, or in the mind? A computer might receive an image from a camera, if seeing is something that happens in the mind. However, unless it's programmed to understand that image, it is not really seeing. The same can be said of any of the other senses. The body (our "hardware") collects and transmits the information and the mind (our "software") makes sense of it.

When we think about it, we can ask this same question about animals as well. Most animals—even the simplest—have sight, among other senses. The simplest eyes can only tell the difference between light and dark, enough to tell if a possible predator is looming overhead. These are the eyes of a scallop or some worms. But these creatures have virtually no brain and they certainly can't think about what they are seeing. So, does a scallop really "see" the same way that we do? Most of us would say that a scallop is seeing, even without a mind.

What about a computer that's as complex as a scallop's brain and is hooked up to a camera? How complicated does a computer have to be in order for us to say that it can see, hear, taste, smell, and touch? This is a hard question to answer. We don't have a good answer yet. Our ability to create senses for machines has grown faster than our understanding of what happens inside those machines.

THE ABILITY TO MAKE DECISIONS

Another part of being intelligent is being able to make decisions. Some decisions are easy—for example, whether it's safe to cross the street. Easy decisions are ones in which the stakes aren't very high, the questions are not complicated, and there is plenty of information available. This is why crossing the street is an easy decision. We can look in both directions to see if there are any cars coming, we can see how wide the street is, we know how quickly we can walk or run, and so forth. Of course, the stakes can be high. If we don't make it across the street safely, we can be hurt or killed. But, for the most part, deciding whether to cross the street is a relatively easy decision and one that a computer can be programmed to make.

Teaching a machine to make simple decisions is a matter of coming up with rules to follow that will lead to a correct decision. For street crossing, the decision-making program might be something like this:

1. Rule: I can cross the entire street in 10 seconds.
2. Rule: The speed limit on this street is 30 miles per hour (48 kilometers per hour).
3. Rule: Cars moving at 30 miles per hour travel 440 feet in 10 seconds.
4. Rule: Look to the left and then to the right.
5. Question: Do you see any moving cars? If no, make a decision to cross the street. If yes, go to number 6.
6. Question: Does the car look like it's more than 440 feet away? If yes, make a decision to cross the street. If no, make a decision to wait until the car goes by and the street is clear before crossing.

Making a decision about whether or not it's safe to cross a street is therefore simple. There aren't many factors to consider and it's often possible to have all of the information that we need. Yet, even crossing the street is not necessarily as easy as it seems. How does this affect a computer's ability to make this decision? As an example, what if you can't see 440 feet down a street because of a curve or something else that blocks your view? Or what if you don't know whether the cars are moving at the speed limit? Will you have good traction on the road, or might you slip and fall in front of the car?

These are all issues that can make your decision more difficult, and it might be that not all of this information is available the moment you're trying to decide if you can cross safely. If a computer can't see 440 feet (134 meters) away, it might be hit trying to cross when a car is just out of sight. Or it might wait forever to cross since it can't see if there's a car 440 feet down the road or not. Or, for that matter, what about a car that's only 300 feet (91 m) away but moving only 20 miles (32 km) per hour? Either way, the rules listed earlier can't help a computer cross the street.

By the same logic, unless we tell it to do so, a computer won't look at the road to see if it will have good traction. In fact, a computerized system won't look at *anything* that we don't first instruct it to observe. Computers will follow the directions we give them, but they can't go beyond these directions. We can program a computer to cross safely at any specific location. Yet, unless we can think of every single thing that might matter in the decision, we can't program a computer to make a good choice.

It's clear that computers can't always make simple decisions. This suggests that computers might not be able to make some more complicated decisions, either. We will need to come up with complex rules that cover all possible circumstances. We will also need to come up with a way to help computers compensate for missing information and confusing circumstances. This is the sort of thing we do all the time, but computers have problems with it.

Fuzzy logic is a branch of research that studies how to make decisions when we don't have all of the information we need. Instead of using firm answers such as *yes* and *no*, fuzzy logic can find ways to deal with *maybe* and *sort of*. Going back to the street-crossing problem, maybe we can't see the full 440 feet to know whether it's safe to cross the street. However, we can program the computer to make an educated guess as to the chance that a car will be close enough to hit us.

Say, for example, we can see 200 feet (60 m) down the road, leaving 240 feet (73 m) that we can't see. If we see that cars come down the road an average of once each minute, we can calculate the chance a car will be on that important 240 feet of road that we can't see. This is the same stretch of 240 feet where a car might be driving. When it takes a car only 5 seconds to travel that length of road, there are 5 chances in 60 (or 1 chance in 12) that a car won't be seen, but will be close enough to hit us at any point in time. Using fuzzy logic, we can tell the computer, "If you don't see a car, there is a 1 in 12 chance that if you cross the road you'll be run over." We can also

How Computers Hear and Talk

Computer hearing is a lot like computer vision. Microphones can capture sounds and convert them into electrical impulses. However, these impulses must be interpreted before it can be said that the computer hears. Here's how that happens.

Microphones transform sound into electrical impulses. Every word is a group of sounds and each sound makes a very specific shape when examined electronically. A computer can use these patterns to recognize the word that was spoken.

A computer can reverse this process in order to speak. It can select the correct word or sentence based on its programming. It then looks up the correct electrical pattern that when fed into a speaker will create the word it wants. Computerized speech is not always pretty. We can nearly always tell when a computer is talking to us, but it's getting closer to sounding like a person all the time.

program the computer to determine an acceptable level of risk. If we tell the computer that we're willing to take a chance when the risk is less than 50%, the computer will decide to cross the road when it doesn't see a car. This will happen even though, in theory, it will be run over once every 12 times it crosses. If we tell the computer that it can only cross the road when there is less than a 5% chance of being hit, it will never cross the road, even though it can safely cross 11 times out of 12.

We can do the same thing to help a computer make more complex choices than it did in earlier circumstances. The secret is to be able to identify all of the information needed to make a good decision. Then, the way to compensate for missing information must be decided, as well as the acceptable amount of uncertainty.

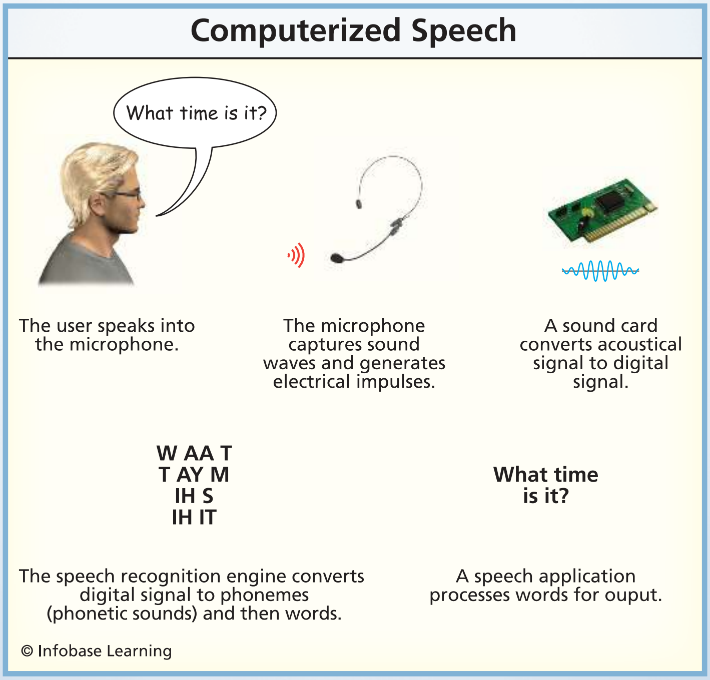

Figure 3.4 This schematic shows how a speech recognition system works. The process would be run in reverse to create computerized speech.