

MAY/JUNE 2006

# IEEE Intelligent Systems

## In this issue:

- AI's 10 to Watch 5
- Intelligent surveillance 102
- Affective computing 104

P U T T I N G   A I   I N T O   P R A C T I C E



[www.computer.org/intelligent](http://www.computer.org/intelligent)

IEEE  
computer  
society  
60<sup>th</sup> anniversary

# Learning and Education: A Continuing Frontier for AI

Oliver G. Selfridge, *Massachusetts Institute of Technology Media Lab and BBN Technologies*

**A**rtificial intelligence is wonderful! We have found out so much, explored old ideas in new ways, given extraordinary new tools to computer programmers, and on and on. At only a half-century old, we can assess AI as successful and productive indeed. But we might wonder where it is going and how far we have come in the big

picture. In the long run, where will it go? Where can it go?

Right now, we gauge AI's success chiefly by comparison with human intelligence, but that comparison can be tricky indeed, and treacherous too, for we do not necessarily have the right standards. Is the chess-playing wonder Big Blue more intelligent than Garry Kasparov because it won? And did it win because it could look ahead to thousands of times more positions than Kasparov could? Is that intelligence? Of course not! It is obvious that intelligence is not a one-dimensional function; there's not even a widely accepted standard of assessing human intelligence, including the IQ. We are all good at being intelligent in different things in different ways. But there are some overall distinctions in how we use intelligence, in cognition, in behavior, in planning, in creativity, and so on.

Learning, however, is a profound key to intelligence. This is not to say that all intelligent acts or thoughts require learning, although many do. I mean learning in a broad sense, ranging from elementary adaptation to the intellectual creativity that builds new models of human descriptive capabilities and other intellectual activities.

Machine learning has been going on for a long time—consider Arthur Samuel, who in 1959 published a paper on a program that learned how to play checkers better.<sup>1</sup> We have made some major advances since then, but how far do we still have to go? Education is not pouring knowledge into an accepting organism, but rather opening doors to

invite the acquisition and improvement of cognitive and other intellectual skills and plans. How can we develop software that can be educated?

## Learning and education

A chief essence of human intelligence is continuing learning or adapting, which we rely on all the time. Machine learning is the most important aspect of AI. It should be about not just learning particular skills or subroutines but the whole shebang, including behavior, cognition, symbolic manipulation, models of the world, and, above all, goals and what to want.

In our culture, we usually couple the concept of learning with education—going to school or college. But I am discussing more generally the ideas behind an organism's or system's improving itself through its experiences.<sup>2</sup> All the improvements that AI might generate or even conceive of must result from learning through different experiences.

In people, learning occurs in rich and different contexts, nearly all the time, and at many levels. It is by no means a single simple topic. Learning to eat with chopsticks is very different from learning to integrate by parts.

When people start to learn something, they come with a large repertoire of capabilities and skills already learned and intact; these may be selected and perhaps multiply applied. This is almost never the case with machine learning. Instead, researchers usually consider single problems in isolation. Furthermore, the temptation to consider environments to be unchanging is hard to overcome.

*Learning is the most important part of artificial intelligence. A basic research effort can explore how to build software that can learn and be educated very broadly. The essence of learning is trying.*

The examples of learning below range from cognition to planning. Furthermore, they're mostly simple in human terms, and I try to avoid those that need mathematics to discuss or assess their performance.

### Human learning in a human environment

Many aspects of human learning are illuminating about its powers, especially concerning the breadth of human resources for learning (in the broad sense). The first obvious point is that children start to learn at birth (or even prenatally). Something once learned rarely stays unaltered, or undeveloped further. Here's a parable on the development of a simple but powerful concept about a certain form of learning.

An infant boy Bobby, three months old, is lying in his crib, and from the ceiling his mother hangs a sparkly mobile a foot or so above him. He looks at it and wants to hold it. He puts out his hand and moves it to the mobile. What he's trying to do is to move his hand to set to zero the visual angle of separation of his hand and the mobile. Of course, as many of you will remember with your own children, what happens is that Bobby knocks the mobile away.

A week later, his procedure is a little different: as his hand approaches the mobile, he slows it down, so that the mobile doesn't get knocked away.

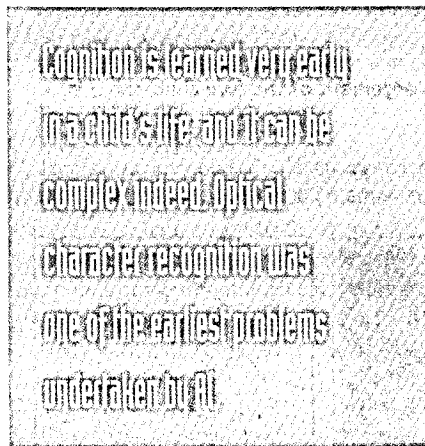
A year later, Bobby has learned to walk. He sees a red ball on the floor in the next room. He wants it and runs up to it to minimize the visual angle separating his heading from the ball's. Of course, he kicks it away. This time, however, it doesn't take a week to improve his performance. In a few minutes, he learns to slow down on getting close to reaching the ball.

A couple of years later, and he's over three years old. His father and all the children are in the large garden, and his father throws a ball along the ground to the side. Bobby runs directly towards it but changes his course as it goes past him ahead. His elder sister, however, starting at the same distance from the ball, leads it, and gets to it first. That makes Bobby reset his chasing algorithm. His solution is to modify his fundamental rule in chasing something: move so as to try to keep constant the horizontal visual angle between the target and his heading. Such a procedure will be familiar to sailors on the coastal ocean, as well as to those familiar with vector arithmetic.

Three years later, Bobby is in his backyard learning to catch a ball thrown by his father. Bobby's solution is to modify his rule by rotating it to vertical; the idea, then, is to move so that the vertical angle the target ball projects stays approximately constant and ... you'll catch it! It works!

That's why a suggested study is going to be called "Outfielder," after the term in baseball.

Other types of learning are even harder to understand at first. Consider a child's learning to ride a bicycle, typically at age six or seven. She can do it surprisingly quickly, even if she has never even ridden a tricycle. After she has succeeded, ask her "How do you steer where you want to go?" The answer will be something like "Oh, that's



easy, that's what the handlebars are for." Her parents, and most people, will tend to agree, but it's not true. You use the handlebars for balancing, and you steer by leaning. Think about it!

### Human and animal learning

We all know that different people can learn in different ways, and we hope that teachers take this into consideration as well. For example, how did Helen Keller learn to understand and respond to speech?<sup>3</sup>

Even animals have different cognitive skills that they can learn to use in different situations. 30 years ago, Dick Feynman wrote;

In 1937 a man named Young did a very interesting experiment. He had a long corridor with doors all along one side where the rats came in, and doors along the other side where the food was. He wanted to see if he could train the rats to go in immediately to the door where the food had been the time before. The question was, how did the rats know, because the corridor was so beautifully built and so uniform, that this was the same door as before? Obvi-

ously there was something about the door that was different from the other doors. So he painted the doors very carefully, arranging the textures on the faces of the doors exactly the same. Still the rats could tell. Then he thought maybe the rats were smelling the food, so he used chemicals to change the smell after each run. Still the rats could tell. Then he realized the rats might be able to tell by seeing the lights and the arrangement in the laboratory like any commonsense person. So he covered the corridor, and still the rats could tell.

He finally found that they could tell by the way the floor sounded when they ran over it. And he could only fix that by putting his corridor in sand. So he covered one after another of all possible clues and finally was able to fool the rats so that they had to learn to go in the third door. If he relaxed any of his conditions, the rats could tell. Now, from a scientific standpoint, that is an A-number-one experiment.<sup>4</sup>

Many other surprising animals do something very like human learning. Irene Pepperberg's African parrot Alexander has learned to count to six or so, making categories of surprisingly complex composition. (See [www.edge.org/3rd\\_culture/bios/pepperberg.html](http://www.edge.org/3rd_culture/bios/pepperberg.html) and Irene's book on the subject.<sup>5</sup>)

Humans too not only use different cognitive skills, but we make different cognitive models in almost everything we do. Nearly 60 years ago, my then-roommate Walter Pitts (yes, the one who cowrote "A Logical Calculus of the Ideas Immanent in Nervous Activity"<sup>6</sup>) and I walked into a cathedral, and I remarked that looking up into the twin spires was a visually remarkable thing. He said, "I don't care about the sight of the spire; what I remember is the stretch on the front of my neck."

### Examples of cognition: What a child learns easily

Cognition is learned very early in a child's life, and it can be complex indeed. Optical character recognition was one of the earliest problems undertaken by AI.<sup>7</sup> But the flexibility of a child's visual recognition of characters far surpasses that of any current OCR system.

Here are some examples that are mostly easy for a child of, say, 10 years old. Figure 1 is an enlargement of an advertisement; the lower half is eminently readable for most of us, although the individual characters are not obviously so at all. The upper half is an enlargement of the two characters immediately below, showing their enormous fuzziness. Children have little difficulty with the lower half.

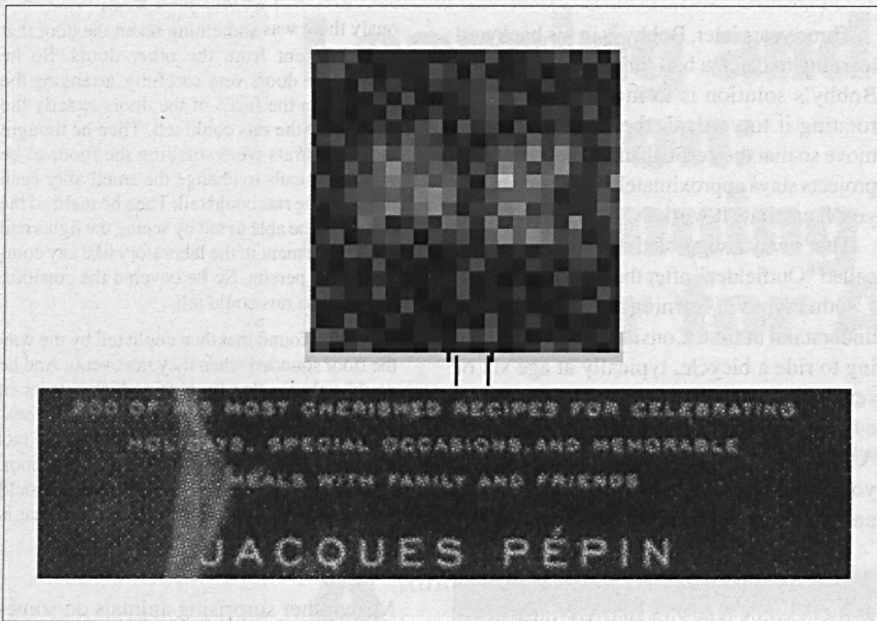


Figure 1. The upper part of the figure is an enlargement of the two characters immediately below.

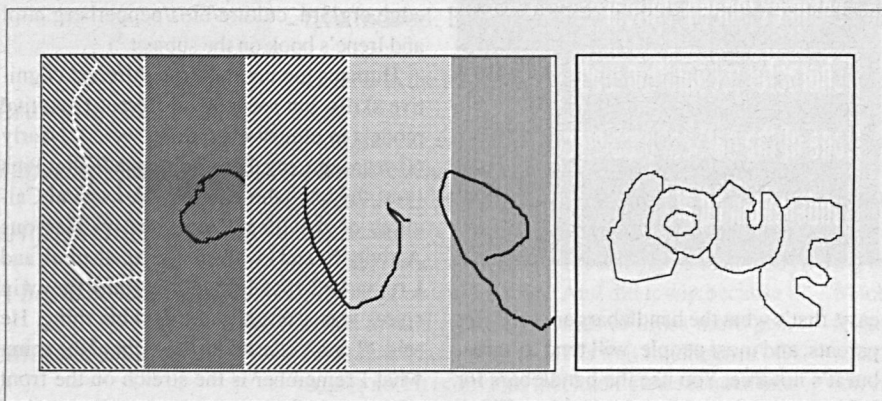


Figure 2. Different fonts, styles, and contexts are often hard to handle.

According to rscheearch at an Elingsh uinervtisy, it deosn't mttair in waht oredr the ltteers in a wrod are, the olny iprmoatnt tihng is taht the frist and lsat ltteer is at the rghit pclae. The rset can be a toatl mses and you can sitll raed it wouthit porbelm. Tihs is bcuseae we do not raed ervey lteter by istlef but the wrod as a wlohe. □  
□  
sdrawkcab eman ym si egdirfles revilo

Figure 3. Other pieces of reading material.

Another interesting example is the surprising pair in figure 2.

The examples in figure 3 might be a little

hard for the 10 year old, but you will have no difficulty.

Actually, I disagree with the conclusions

in the first of those two, but that's another matter. Again, it would be (moderately) straightforward to concoct a special program to read either or both of those two examples, but the point is that we humans don't have to; we do it without thinking.

The second one of those has other interesting points. Once you get it, you have the secret, and it will never be a surprise again.

Sometimes we must remember special fonts (not needed above, obviously), but that seems to be no problem; for example, see figure 4.

Even if we don't recognize it at first, we will remember it soon!

Another simple example involves finding the relevant clue to an apparent classification. Figure 5 is a popular one at certain levels.

I should like AI software to be able to tackle problems like figure 5 by exercising general principles of exploration just as people do, rather than having to be programmed for each one.

### Where is machine learning now?

In computer science today, machine learning typically refers to a field that concentrates on induction algorithms and others that can be said to "learn."<sup>8</sup> There is a fair acceptance of the limitations of that point of view. I myself embrace a much wider one, hoping that it might include all aspects of what we can term learning in people, especially young people. I think it is reasonably clear that no machine learning system could possibly learn to drive an automobile without a large number of ad hoc assumptions and significant programming. Learning to drive involves far more than driver ed can possibly teach; this is not to find fault with driver ed, but to emphasize the lessons we must learn about teaching our software.

I consider adaptation (of a single variable parameter, for example) as one extreme of a broad range of system improvement. By and large, most work on machine learning has been the establishment and justification of algorithms that optimize a single parameter or set of parameters.

This has been the main thrust of studies on learning by software. Machine learning's first and oldest problem was to apply adaptation to a particular problem in the construction of an algorithm; there is a large library of examples, many of which you can find in the journal *Machine Learning* (Springer Netherlands). For many of them, mathematics is widely and profitably

applied. One key here is to evaluate a proposed solution by how well it works on a typical problem, using different values for some parameter of the solution and possibly hill-climbing to the optimum.

A second thrust is to propose general techniques to evolve solutions to tough problems (that is, to learn to satisfy an overall evaluation function). There are two main such techniques, both enjoying fair and widespread reputations. The first derives fundamentally from extending the work shown in a 63-year-old paper by Walter Pitts and Warren S. McCulloch, which (roughly) proved that anything that a computer could do could also be done by an appropriate network of idealized simulated neurons.<sup>6</sup> Later work extended that by constructing such networks in such a way that they were easily modifiable,<sup>9</sup> evaluating such modifications by assessing whether the net would work better on the task assigned and so forth. In this way, a kind of hill-climbing can improve the system to an optimum (this field is also termed *connectionism*).

The second technique uses the notion of biological evolution, primarily started by John Holland 30-odd years ago.<sup>10</sup> The basic idea is to express a piece of software as a much shorter string of binary digits—a kind of binary genome! We can modify the software by performing changes to that string analogous to those performed in real life in evolutionary processing. Then, as before, we can do some hill-climbing, and thereby improve the system to an optimum. This field, termed genetic programming, is enjoying widespread and enthusiastic support from researchers. It has achieved some notable successes.

Both of these techniques use just a single success (or improvement) factor—an evaluation function—and to me that's questionable. It's rather like believing that beauty is the only quality to consider in choosing a wife, or income in choosing a husband (and there's a sexist approach!). But deeper than that, should a janitor cleaning a toilet decide to go clockwise or counterclockwise on the basis of how that choice will affect the company's stock price? Nonsense! Subordinate tasks have their own purposes, which are assigned by superiors. A janitor's evaluation function depends on the tasks he has been assigned, and a toilet makes different demands of cleanliness from those in a machine shop or a biology lab.

Some feel that in machine learning an

approach of logical clarity and mathematical exactness will provide a way of expressing the truth—that is, the propositional calculus should be the language of choice.

A concept is represented by a set of Horn clauses. ... That is, an object X belongs to the concept P if the predicates Q and R and S are true. The clause will be called a definition of the concept P.<sup>11</sup>

Leslie G. Valiant goes much further than this:

The adequacy of the propositional calculus for representing knowledge in practical learning systems is clearly a crucial and controversial question. Few would argue that this much power is not necessary. The question is whether it is enough. ... The search for extensions to the propositional calculus that have substantially learnable classes may be a difficult one.

... This idea has considerable philosophical implications: A learnable concept is nothing more than a short program that distinguishes some natural inputs from others.<sup>12</sup>

I am at the other end of such an approach; many of us do not believe that a mathematical or binary logical approach is necessarily universally profitable. It is not so much that our points of view contradict each other, but rather, that they seem to me to be orthogonal. My notion of a concept is distinctly different from Valiant's.

Can you think of some chore or duty that a person does that she doesn't do better the second time? Or can you think of some chore or duty that a computer does that it does do better the second time?

... AI software should be more concerned with being changeable—and all that that implies—than with satisfying specifications; that is, with being "correct." ... In people ... learning and adapting take place at many levels simultaneously and continually, and we in Machine Learning must find out how to build our models and software to do so too.

Find a bug in a program, and fix it, and the program will work today. Show the program how to find and fix a bug, and the program will work forever.

### Particular points about learning

There are several points that need special attention in machine learning.

### Purpose structures

People learn because they *care*—that is, they have purposes or motivation. Machine learning will work well only if that is also true in software. Most of us would certainly acknowledge the truth of that with respect to

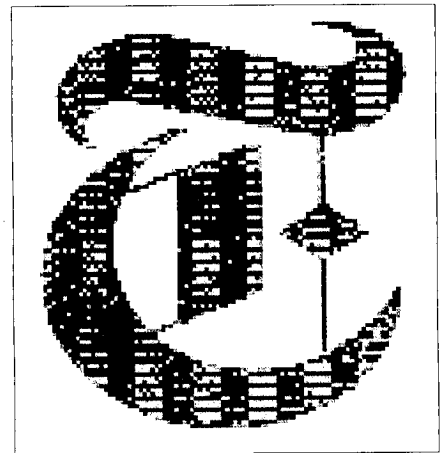


Figure 4. What letter is it?

CABBAGE	CHOCOLATE
CARROT	LIMESTONE
STRAWBERRY	GRANITE
HERRING	GRAPEFRUIT
BASS	WATERMELON
RABBIT	TITMOUSE
BABYSITTER	LOBSTER

Figure 5. Where do you put MUSSELS? How about CAPERS?

our own children. We can argue whether reward is more effective than punishment, and in what circumstances, but the main point is widely agreed upon.

The essence of a purpose is choice and evaluation through some form of feedback: the marksman has the purpose, not the trigger. A responsible agent ought not to be designed and programmed, for even if I think I know what I want now, I really want my agent to be what I am going to want in the future, even as those wants change and evolve.

Does your computer give a damn?

Is your computer—or rather, your computer and its software—happy or rewarded when it does well? Is it rebuked or punished for doing badly? Of course, you know what doing well and doing badly mean, but your computer...? Has your computer ever been surprised? angry? hungry? afraid? Does your computer care about what has happened and is happening, or is it just the programmer? What I am doing here is to propose that we explore ways of a computer's starting to be able to have feelings like those; for in us, those feelings direct our attentions to what is important for us. Notice that what is important will change as time goes by. Indeed, it is vital to pay more attention to the changes and

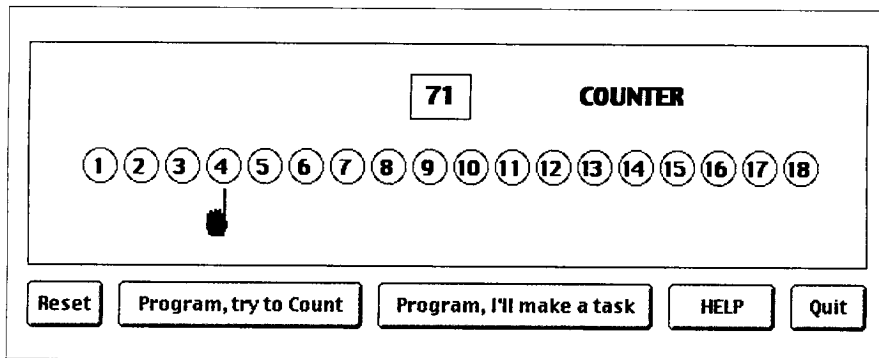


Figure 6. The first display in COUNT, a program illustrating that some learning must follow other things learned.

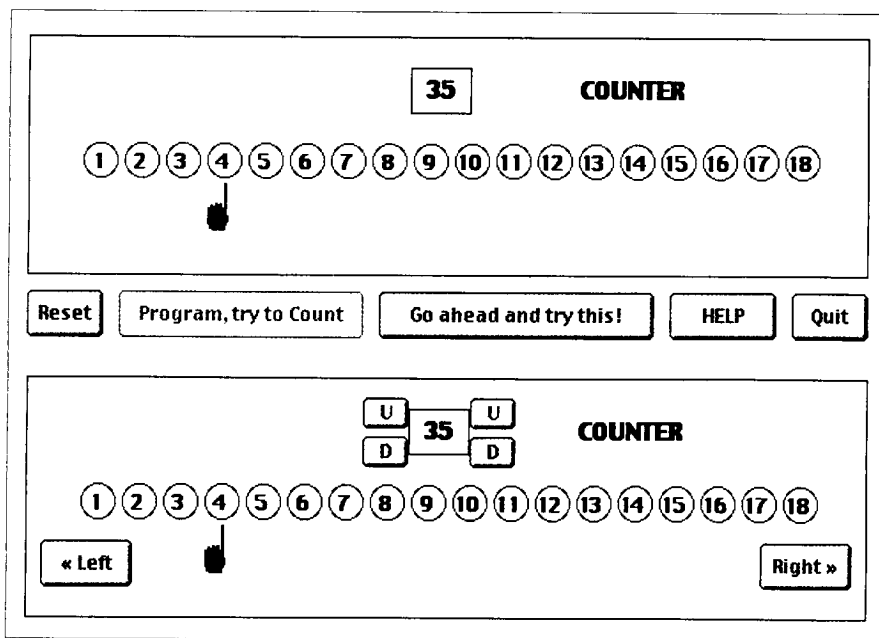


Figure 7. The user should enter a task to help the COUNT program learn to count.

how they occur at every level, for after all, adaptation and learning are all about change. As Marvin Minsky has said, 'The best is enemy of the good,' meaning that when we use the word optimum, we are considering a static world, though in fact what is optimum today will almost certainly not be so, soon in the future.

So what are the key elements of purpose in machine learning? The high- and higher-level goals form something like a hierarchy (though not exactly), together with the lower-level ones. The lower-level goals can be extremely simple—as the operation of a bacterium shows:

Certain bacteria, notably *Escherichia coli*, have these habits of movement: in a neutral environment, each bacterium moves for some length of time, typically a second or so, in more or less a

straight line, propelled by a flagellum or rotating tail; it then reverses the rotation of its flagellum, which flies apart into its eleven constituent fibers, and which turn it more or less at random directions (that is, it twiddles). Soon the flagellum reverses its direction of rotation and again propels the bacterium forward, but in a new direction.

If, however, the motion of the bacterium moves it into a region of greater concentration of food "attractants," or if for some other reason that concentration is increasing, then the tendency to stop forward motion and twiddle is suppressed. In this way the bacterium tends to keep going forward while the concentration of foodstuffs is increasing. When that ceases, it twiddles, and tries again.

The result is overall that, when things are getting better, the bacterium proceeds in the same direction; when they are not, it chooses new directions at random. Such a motion strategy is termed run and twiddle.<sup>13</sup>

In people, purpose structures form a vast source of control at every level, and usually we switch from level to level without being aware of it. For example, if I want to drink a cup of coffee, I must undertake a sequence of actions (subgoals). For each one, I might have to make choices: if I've hurt my right hand, then I will use my left, and so on. More about this approach can be found elsewhere.<sup>14</sup>

### Learning that is needed to learn other things

Some learning must follow other things learned; for example, learning to count must precede the learning of most other kinds of arithmetic. A couple of decades ago, I wrote a program called COUNT that illustrates this (and some other things too).

The idea is for you, the user, to teach a piece of software to count. COUNT starts out with certain innate or primitive capabilities, which are unknown to you. You have to try to teach the program to count by tasking it to try to do certain things and seeing which ones it can do. If it succeeds in a task, you can tell it both to remember its success and also to use the way it solved that task as another primitive capability. Once you have given it the right tasks that it can learn to do, then it will be able to count.

That program's purpose was to illustrate some of the difficulties of a very young child's learning new things. Instruction is almost impossible because there's no shared language to instruct in, and the child's basic or innate capabilities might not be known very well, or perhaps not at all. But as the child learns some easy things and vocabulary, the hard things to learn become easier.

Of course, most of the previous paragraph is intended to apply as well to learning by a piece of software. It is notable that the user does not initially know what the program's primitives are, although he might be able to make reasonable guesses about them. When you start COUNT, it displays the interface in figure 6.

COUNT has a limited field of competence; it cannot do much, and so it cannot learn much. It can, however, in some sense learn how to count, though not at first. You can decide whether that description is accurate. If you tell it to try to count, it will tell you that it has tried, but it cannot do it. So you make a task by tapping the corresponding block. Figure 7 shows how to do this.

You can change only two things: the num-

ber in the counter and which object the finger is pointing at. The task you give to the program is to try to find some operations that will change its board into the one you have specified below. When you have changed the board to represent the task you want it to try, then push the button "Go ahead and try this!" The program forms concepts, unknown to you, which it uses as primitives to form newer and more powerful capabilities.

So, how would you get this program to count?

For people, old and young, we must do something rather like *initialization*. That is, we set the counter as low as possible with the finger all the way to the left or right; then we can move the finger and at the same time increment the counter as long as possible—that is, until the finger hits the end.

And that is in fact how COUNT works. It has these as primitives:

- **Increment:** Increment the counter by 1 (unless at 100).
- **Decrement:** Decrement the counter by 1 (unless at 1).
- **Right:** Move finger one block to the right (unless ...).
- **Left:** Move finger one block to the left (unless ...).
- **Again:** Repeat the next primitive until you can't go on.
- **Prim:** Make the current success on a task a primitive operation.

The last two of those are rather metaprimitives in that they affect the board not at all. The rather obvious solution is to construct a primitive that you can call **Begin**, which may be: **AD AL**. This will set the counter at 1 and move the finger all the way to the left. Then make another primitive that **Increments** the counter by 1 and moves the finger to the right by 1; call it **Step**, perhaps. This new primitive would be **AR**.

Then the program can count with **B AS**; seems simple, doesn't it?

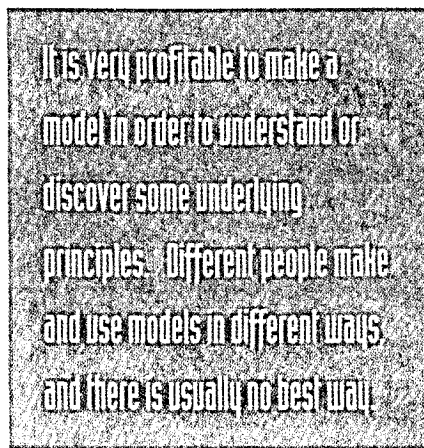
A key feature is that the program finds new useful procedures on its own and uses them for other ones, and the user might know nothing about what those procedures are or what's going on. The program isn't very complex; in fact, it is elementary in its structure, searching, and actions. Furthermore, the concept of counting that it achieves could hardly be simpler or more elementary. What it does avoid, however, is having the user tell it how to do something.

### Making a mental model

Learning often requires making a mental model. I use the word *model* in a very general sense indeed. Sometimes, of course, models can be complex, as in many mathematical studies—Fourier transforms, for example. A simpler illustrative one is this:

Many years ago I was in the dining room, replacing the usual light switch with a dimmer. A female friend was watching and seemed worried as I took the old switch apart: "Shouldn't I turn off the power downstairs?" "No, please don't do that, because I need to check it with the power on," I replied. And I took the two live wires and touched them, and the light went on (of course!).

She looked absolutely stunned. "You



mean," she said, "that that's all a light switch does?" One point here is that she had no need for a model of what happened. Yet many of us do build models that prove later to be valuable indeed. When I was eight years old, I used to play with electric trains. As many of you will remember, we learned as children that touching two wires together can sometimes turn on a light or start a train moving. That kind of mental model is obviously valuable for someone who hopes to become a scientist or any kind of mechanic.

For us in general, and for researchers especially, it is very profitable to make a model in order to understand or discover some underlying principles. Different people make and use models in different ways, and there is usually no best way in any absolute sense.

It's easy enough to underestimate the kinds of differences in the models that we make about the world we live in. One key factor in creativity is being able to alter models, bringing in new factors and so on.

### The ability to describe what happened

When are two experiences (one past, one future) similar? ... In the sense that lessons learned in the former can help in the future one. The key question to begin with is, how do we tell that two situations are similar enough that the lessons learned in the first are applicable to the second? The general rules are pretty obvious, and many of them have been incorporated in studies about common sense, which many researchers are studying now.<sup>15</sup>

**Lactose intolerance.** Thirty-odd years ago, I cooked a meal for nine or 10 friends and family: cream of spinach soup, roast chicken stuffed with mushrooms and onions, roast sweet potatoes, salad with tomatoes, and so on, including a nice Chardonnay. At the end of the meal I felt terrible in my gut and had to stay at the table for another 40 minutes. So what was it? Something I ate or drank? Something else?

Two days later on a similar occasion, the same thing happened. The only common factor in the two meals was the cream soup, this time with another vegetable. It was the cream or milk, for I had developed an intolerance to lactose rather suddenly. What saved me was remembering in detail the environments in which the discomfort occurred. We have all had experiences like that, but by and large, software trying to learn from experience does not remember and categorize all the features of its experiences.

**Studying the tides and the moon.** Ancient man, say 40,000 years ago, discovered that the moon caused the tides. That must have been one of the first mysteries of the world. He had known that a piece of wood, a stone, the wind, his hands, and so on, could move water. How could the moon? To discover that, he had to remember the environment, including the shape of the moon as the (very) relevant feature. Merritt Ruhlen has discovered fairly convincing evidence that at least 30,000 years ago, mankind had charted the phases of the moon in cave drawings.<sup>16</sup> How did mankind make the connection? Was that because of the tides and the gigantic discovery? Actually, we are still not sure exactly what concepts of *causality* worked in prehistoric mankind, but the connection of tides and moon must have been wonderful indeed.

**Sharing with other learning systems**

Much learning derives from sharing with other learning systems and thereby requires appropriate communication. I'll go further than that: in people, most of the advanced learning (as in classical ideas of education) cannot be acquired or maintained through direct experience.

Very early on in life, the learning systems are in full play: learning to reach and touch (as in the "Outfielder" example), to talk, to make friends, and so on. I shall not try to make a deep analysis of educational procedures; the need to translate them to machine learning must be stressed.

Social learning is common in animals, as we all know; most of us have had pets of one kind or another. Much learning occurs in other lower organisms, such as bees that dance to tell the other bees where the nectar is. There has recently been substantial work in the communications and intelligence of swarms.<sup>17</sup> Even certain solitary insects can communicate with others.<sup>18</sup>

The kinds of intelligence and learning I am discussing will usually require a shared vocabulary, including terms describing experiences and so on. Research on multiple agents is beginning to study these questions.

**Researchers: Be wary of traps**

There are several traps that machine learning could fall into.

**Learning in order**

What order should we attempt to use in developing sophisticated aspects of AI?

For example, some aspects of intelligence arise only from others that must be there first. A good example is consciousness: there are good reasons to suppose that consciousness is an emergent factor that cannot be attained without rich experiences in other factors, like cognition, motivation, behavior, and so on. It's also perhaps tempting for researchers to try to add assorted emotions to AI software. But although some forms of emotion arise early in the phylogenetic scale (like panic, which can be found even in many invertebrates), some emotions occur only in much older and more experienced humans—good examples are awe, respect, and other such feelings.

We should also remember that such studies must also consider the other changing and adapting modules and the changing environment.

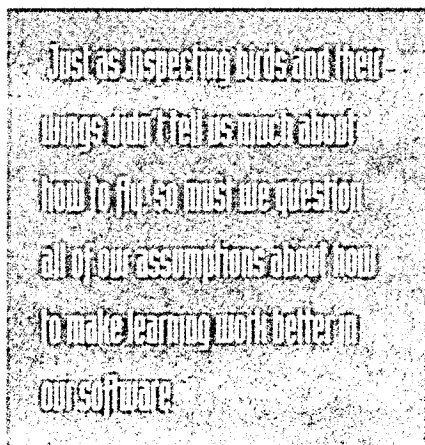
So, what aspects of learning and intelli-

gence are best considered as emergent (that is, not worth trying to analyze before the underpinnings have been developed)? In some sense, AI itself is emergent from elementary computer theory à la von Neumann. That is, in considering the precise operations of computer binary arithmetic, there is no hint that they can encode the adaptive powers that shape cognition and planning. We must change the way we talk about circuits and memory units.

I have already discussed something related to that in presenting the program COUNT.

**Modifying high-level goals**

When should we consider modifying the overall goals assigned to a powerful piece of



software? I presume that those goals are the basis of continuing adaptation. It is probably the case that the lower-level goals are often selected or modified by the goals upstairs, so to speak. I have a couple of examples here.

**Traffic lights.** Consider a system of adaptation that my friends and I worked on a few years ago. We proposed building a software agent to be part of a traffic light system. Our first plan for each individual agent included the high-level goal of minimizing the average delay of the vehicles passing through the intersection. The initial simulations seemed quite satisfactory, and we began to integrate the system into networks of similar agents.

That seemed to be satisfactory until I explored a certain possibility. Namely, when the traffic was very high in one direction and low in the other, the system might keep a single car waiting for 600 seconds (10 minutes) so that 300 cars wouldn't have to wait an average of one second longer. Not acceptable! So, we had to modify the higher-level

goal—perhaps so as to minimize the average of the squares of the delays.

**Sorting.** Something similar occurred on another project when I was working as an adjunct professor at the University of Massachusetts. It was about 30 years ago, and we were looking at a common application for the UMass computer system (those were the days when time-sharing systems were the common access for most users). The system had a powerful, widely used sorter, and we were interested in making it more responsive and cheaper to use. So we modeled an adaptive system whose goal was to minimize the length of time and the amount of memory the sorter used. (Some of you will remember when amount of memory still mattered—try telling that today to Microsoft!)

Of course, as memory's price and availability improved enormously, we had to change most of our early conclusions because the goal became wrong—memory is now essentially free.

**People aren't the best at everything**

Although we nearly always start a machine learning study by asking what people might do, we should not accept that as an absolute requirement. In the same way, we do not require our infants to learn adult attitudes, thoughts, or behavior—those can come later!

Computers have very large powers, which are going to grow, and they will eventually challenge ours in every intellectual respect, not counting the possible symbiosis of embedding nanochips and so on into human flesh. Just as inspecting birds and their wings didn't tell us much about how to fly, so must we question all of our assumptions about how to make learning work better in our software.

**M**uch recent work in machine learning has been brilliant, both in individual tasks and in more general approaches, like genetic programming and neural networks. But we must examine some broad questions and approaches (such as the ones I've discussed in this article) much more thoroughly; some of these have hardly been tackled at all.

Ideally, I should like to see a workshop put together to evaluate some of those ideas and others that researchers might feel have been neglected. I would hope such a workshop

would produce a proposal for a research program that would start examining some of those problems.

Those problems are all essentially basic research, but some have much more immediately useful applications than others. For example, purpose structures in ambitious pieces of software would enable adaptation at many levels simultaneously, making it far easier for programmers to design that software and to understand and correct errors. A more powerful and flexible learning system, able to adapt and integrate numerous cognitive approaches without having to be programmed for each one individually, might have wide military and commercial applicabilities. Perhaps communication languages for separate learning software agents could be tackled later.

My guess is that one of the first problems to be tackled is learning how to develop and apply purpose structures widely in the design and building of software. And I hope that these new software research fields can thrust AI technology into new learning capabilities and universes. ■

## References

1. A.L. Samuel, "Some Studies in Machine Learning Using the Game of Checkers," *IBM J. Research and Development*, July 1959, pp. 211-229.
2. O.G. Selfridge, "The Gardens of Learning: A Vision for Artificial Intelligence," *AI Magazine*, vol. 14, no. 2, 1993, pp. 36-48.
3. M. Davidson, *Helen Keller*, Scholastic, 1997.
4. R. Feynman, "Cargo Cult Science," commencement address, California Inst. of Technology, 1974.
5. I.M. Pepperburg, *The Alex Studies: Cognitive and Communicative Abilities of Grey Parrots*, Harvard Univ. Press, 2002.
6. W.S. McCulloch and W.H. Pitts, "A Logical Calculus of the Ideas Immanent in Nervous Activity," *Bull. of Mathematical Biophysics*, vol. 5, 1943, pp. 115-133.
7. O.G. Selfridge, "Pattern Recognition and Learning," *Proc. 3rd Symp. Information Theory*, Academic Press, 1956.
8. J. Shavlik and T. Dietterich, *Readings in Machine Learning*, Morgan Kaufmann, 1990.
9. M.L. Minsky and O.G. Selfridge, "Learning in Random Nets," *Proc. 4th London Symp. Information Theory*, Butterworths, 1961, pp. 335-347.
10. J.H. Holland, *Adaptation in Natural and Artificial Systems*, Univ. of Michigan Press, 1975.
11. C. Sammut and R.B. Banerji, "Learning Concepts by Asking Questions," *Machine Learning: An Artificial Intelligence Approach*, vol. 2, R.S. Michalski, J.G. Carbonell, and T.M. Mitchell, eds., Morgan Kaufmann, 1986, pp. 167-191.
12. L.G. Valiant, "A Theory of the Learnable," *Comm. ACM*, vol. 27, no. 11, 1984, pp. 1134-1142.
13. H.C. Berg and D.A. Brown, "Chemotaxis in *Escherichia coli* Analyzed by Three-Dimensional Tracking," *Nature*, vol. 239, no. 5374, 1972, pp. 500-504.
14. O.G. Selfridge and W. Feurzeig, "Adaptive Processes and Control," *Proc. 7th World Multi-Conference Systems, Cybernetics, and Information (SCI 2003)*, IIIS, 2003.
15. M. Minsky, *The Emotion Machine*, to be published by Simon and Schuster, 2006.
16. M. Ruhlen, *On the Origin of Languages*, Stanford Univ. Press, 1996.
17. E. Bonabeau, M. Dorigo, and G. Theraulaz, *Swarm Intelligence: From Natural to Artificial Systems*, Oxford Univ. Press, 1999.
18. I. Coolen, O. Dangles, and J. Casas, "Social Learning in Noncolonial Insects?" *Current Biology*, vol. 15, no. 21, 2005, pp. 1931-1935.

## The Author



**Oliver G. Selfridge** is a senior lecturer at the Massachusetts Institute of Technology Media Lab and a visiting scientist at BBN Technologies. He studied mathematics at MIT, doing graduate work under Norbert Wiener. He has always been interested in AI, especially machine learning. In February 1990, he was elected a fellow of the American Association for the Advancement of Science. In May 1990, he delivered the Distinguished lecture at the Computer and Information Sciences Department of the University of Massachusetts. In June 1991 he was elected a fellow of the AAAI. Contact him at 753 Concord Ave., Belmont, MA 02478; ogs@media.mit.edu and oselfrid@bbn.com.

18. I. Coolen, O. Dangles, and J. Casas, "Social Learning in Noncolonial Insects?" *Current Biology*, vol. 15, no. 21, 2005, pp. 1931-1935.

For more information on this or any other computing topic, please visit our Digital Library at [www.computer.org/publications/dlib](http://www.computer.org/publications/dlib).



**IEEE**  
**distributed systems**  
Expert-authored articles and resources **ONLINE**

**IEEE Distributed Systems Online** brings you peer-reviewed articles, detailed tutorials, expert-managed topic areas, and diverse departments covering the latest news and developments in this fast-growing field.

Log on for **free access** to such topic areas as

**Grid Computing • Mobile & Pervasive  
Cluster Computing • Security • Peer-to-Peer  
and More!**

To receive monthly updates, email  
[dsonline@computer.org](mailto:dsonline@computer.org)

<http://dsonline.computer.org>

# AI and Similarity

Edwina L. Rissland, *University of Massachusetts*

**A**s AI moves into the second half of its first century, we certainly have much to cheer about. It has produced a wealth of solid results on many fronts, including machine learning and knowledge representation, for instance. More generally, the field has delivered impressive, reliable, and widely applicable techniques we couldn't have

dreamed of 50 years ago: constraint-satisfaction problem solving, probabilistic learning techniques, real-time planning, case-based reasoning, market-based models for societies of agents, and so on. AI has made significant headway on developing techniques, computational models, and systems that advance its synergistic twin goals of modeling cognition and building systems to get the job done. In fact, a lot of AI systems get the job done spectacularly.

Yet we still have much work to do on some topics. Among these are similarity-driven reasoning, analogy, learning, and explanation, especially as they concern open-textured, ever-changing, and exception-riddled concepts. Although some of these were recognized from its beginning, AI still cannot deal well enough (at least for me) with the inherent messiness that characterizes much of the world that humans and AI artifacts operate in.

There is no way to shrink from this challenge. Even though some subfields, such as my own disciplines of case-based reasoning (CBR) and AI and law, have made significant advances, abundant opportunities exist to push the envelope further. Doing so is necessary both to shed light on cognition and to advance the state of the art of performance systems. In the next half-century, AI can become robust enough not only to cope with our messy world but also to thrive in it. In this essay, I discuss a few aspects of the topics that I believe are important in order to realize truly robust AI.

## The enduring problem of similarity

From its earliest days, AI has been interested in similarity. For instance, in introducing the AI section of *Computers and Thought*,<sup>1</sup> Edward Feigen-

baum wrote, "A useful rule of thumb used by human beings in most of their problem-solving is this: Attack a new problem by methods that have solved similar problems in the past. The criteria for 'similarity' may themselves be heuristic." Marvin Minsky also addressed this in his 1961 essay "Steps toward Artificial Intelligence," reprinted in the same collection.<sup>2</sup> In discussing teleology and classification, he wrote that "objects grouped together in the classification should be 'similar' in a useful sense; they should depend on relevant or essential features." Earlier still in his paper for the 1958 Teddington conference, Minsky wrote, "Try methods that are similar 'in an appropriate sense' to those that have been successful in the past. Each problem domain has, of course, its own structure, and its own relevant kinds of similarity."<sup>3</sup> Indeed, yes.

Despite this spotlight on what we might call the CBR heuristic, researchers haven't answered this early call to arms on similarity-based problem solving—or on the very nature of similarity itself—nearly enough. Of course, these aren't small topics: they're fundamentally about analogy, cases, concepts, context, relevance, reuse, and more, and similarity is at the heart of much learning and teaching. And vice versa, since similarity is a Cheshire-cat type of notion that depends on "where you want to get to," particularly in explanation and argument.

Broadly construed, the problem of similarity has been attacked from all quarters over the years—for instance, we can even consider some work on game playing and character recognition in this light.<sup>4,5</sup> Of course, CBR is particularly concerned with reasoning with similar cases and modeling different notions of similarity (see the sidebar "The Case-Based Rea-

*For AI to become truly robust, we must further our understanding of similarity-driven reasoning, analogy, learning, and explanation. Here are some suggested research directions.*

## The Case-Based Reasoning Process

CBR is the process of solving new problems by using old cases. Generally speaking, there are two types of CBR: interpretive CBR, in which a new situation is interpreted in light of past ones; and problem-solving CBR, in which a new problem is solved by adapting an old solution. Classifying or arguing for a particular interpretation of a new fact situation in light of existing legal precedents is an example of the first type, and creating a new design or plan by adapting old ones is an example of the second.

In all types of CBR, the process proceeds as follows:

1. The new problem case is analyzed.
2. Based on the analysis, relevant cases are retrieved from a case base using various sorts of indices.
3. From these relevant cases, the most on-point or most relevant are selected using a method for assessing similarity (a so-called *similarity metric*).
4. The most similar or "best" cases (there might be several) are used to craft a solution for the new case.
5. This proposed solution is vetted in some way (for example, with a simulation, sensitivity analysis, or hypotheticals).
6. The new solution is possibly saved in the case base, and indices and similarity measures are possibly updated.

In some CBR systems, some of these steps (for instance, steps 2 and 3) happen together.<sup>1,2</sup> The CBR process is sometimes described in terms of Rs: for instance, retrieve, reuse, revise, and retain.<sup>3</sup>

### References

1. J.L. Kolodner, *Case-Based Reasoning*, Morgan Kaufmann, 1993.
2. D.B. Leake, ed., *Case-Based Reasoning: Experiences, Lessons, and Future Directions*, AAAI Press/MIT Press, 1996.
3. A. Aamodt and E. Plaza, "Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches," *Artificial Intelligence Comm.*, vol. 7, no. 1, 1994, pp. 39-59.

soning Process"). Investigations of analogical reasoning are also on target; while fewer in number, they've given us significant insights.

Although other lines of research also have embedded notions of similarity, they tend not to reason explicitly with individual cases. Rather, they encapsulate similarity in various sorts of generalizations (for example, weights); most throw away the cases and thus mask the similarity-based reasoning involved. In such systems, it is hard to unpack the reasoning to understand how and why similarity was used and so those interested in shedding light on human cognition will find it difficult to learn much about similarity from them.

In contrast, CBR and analogical-reasoning programs explicitly (and mostly symbolically) use individual cases, and the best present the reasoning behind their conclusions. This not only enhances the use of such systems as tools for cognitive studies of sim-

ilarity, but also makes their output more useful in domains such as law and mathematics where the reasoning is as highly valued as the conclusions.

Despite AI's advances, we need more insight and techniques for performing similarity-based reasoning. We must also understand different types and uses of similarity and the concepts and contexts for which they are useful or appropriate, and the relationship of similarity to explanation and learning, and by implication with a host of other core AI topics. These include methods for knowledge representation and techniques for dealing with concepts, context, and purpose.

### The persistent need for explanations

A key requirement for intelligent systems is that the reasoning they do should not be opaque. For instance, just classifying an instance isn't enough. We—humans and AI

systems—should be able to provide an explanation to back up the classification. Announcing an answer to a problem without an explanation, even if it is based on the best practices of the inference technique involved, isn't enough. At present, while a host of methods can perform concept learning and classification, for instance, fewer can provide explanations, say, via analogies with past exemplars. Fewer still can explain the model of similarity used to reach a conclusion and why it's the appropriate one to use in the given problem-solving context.

Explanation often involves the CBR heuristic—that is, reasoning about similar relevant cases. That's especially so in domains such as law and mathematics where a black-box or "because I say so" approach isn't acceptable. Although these two domains are at opposite ends of the messy-neat spectrum, both place a high premium on explanation. Examples and cases play a key role in explanations in both of them. This is obviously the case with Anglo-American law, which is based on precedent. However, even in mathematics, where the standard for justification is logical proof, examples can play a key role in explaining a proof and illustrating its nuances, coming up with conjectures and plans for proving them, reminding us of the importance of a theory's elements and the connections among them, and so on.<sup>6,7</sup>

Explanation involving exemplars can foster concept learning in a wide variety of learners, from toddlers to graduate students. For instance, Dedre Gentner and her colleagues have shown that when young children engage in explicit comparison of exemplars from a concept class during categorization, their learning is deeper in the sense that they move away from perceptual features (such as shape) to deeper, more conceptual ones (such as functional roles).<sup>8</sup> They are also more able to ignore perceptually seductive features in favor of relational ones. In a study of business school students, researchers found that students who explicitly compared negotiation scenarios were more apt to use the appropriate strategy in actual problem scenarios than those who read the same two cases but didn't compare them.<sup>9</sup>

Explanation and argument are at the heart of the case method used in many professional schools. Introduced in 1870 by the Dean of Harvard Law School, Christopher Columbus Langdell, it is now also established practice in business schools and to some degree in medical schools (teaching rounds can be con-

sidered akin to a CBR experience). The case method focuses on close readings of cases, often presents conflicting cases and principles, engages students in pointed Socratic dialogue, and makes strong use of hypotheticals. In essence, it focuses on hard, borderline cases. As Elizabeth Warren, a Harvard law professor, says, "You know the difference between daylight and dark? Well, we spend all of our time at the Law School on dawn and dusk."<sup>10</sup> And, I might add, all the umbral shades in between.

The need for explanation is especially critical when the conclusion will be used in making decisions, giving advice, or other contexts with potentially significant impact. Situations involving health, finances, or legal issues come to mind. For instance, consider the slippery concept of creditworthiness and the task of classifying a consumer as creditworthy. No one—neither credit applicant nor banker—should be satisfied with a system that doesn't explain its conclusions.

Imagine that you've just been classified as not creditworthy. An "explanation" that is only a raw score would be particularly unsatisfying, especially if this decision matters to you—for instance, if it means you won't be approved for an important loan or mortgage.

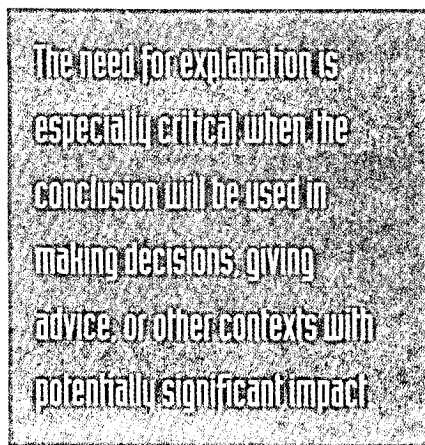
To understand why, you'd probably want a detailed explanation. You might want to know how the score was computed, what the theory behind that computation is, where the theory came from, and what alternative theories and thus scores and outcomes there might be. You might want to know what data of yours was used and its provenance. You might want to know what training data gave rise to the theory, or the theory's biases and track record.

You might benefit from a contrast-and-compare analysis with relevant past exemplars, both positive and negative. Illustrative examples might make the decision more understandable. You might want to know how you could improve your situation or imperil it further. That is, you might want to consider hypotheticals that shed light on your situation. You might like advice on what to do and perhaps a comparison of the courses of action open to you. You certainly wouldn't want to pursue a course of action that would make things worse. In short, you might want a highly reasoned explanation complete with actual precedents and illustrative hypothetical examples, and a bit of well-laid-out guidance. Of course, you might not want all of this at once, but you should be able to get it

if and when you want it—in the manner of *explanation on demand*. I know I would.

The same comments apply for those making the decisions. In arenas with more earth-shaking implications such as international relations, this should be part of decision-making best practices. Comparisons of courses of action (I call them COCOAS) involving precedents and hypotheticals aren't just a nicety; they're essential for making good decisions. Ignoring them can lead to spectacularly awful decisions.<sup>11</sup>

To provide informative explanations requires more research on CBR, analogy, and hypothetical reasoning. We must know more about what constitutes a good explanation and about the art of explaining and advice-giving.



### Messy concepts

What we might call "messy" concepts make excellent subjects for studying similarity, explanation, examples, argument, analogy, and so on. Conversely, these methods are key ways—perhaps the best, or only ways—of dealing with messy concepts.

Most real-world concepts are messy, so dealing with them is unavoidable. Only in the world of mathematics do we have the luxury of fixed concepts with crisp, black-and-white boundaries. Even these aren't so neat and clean when they're evolving.

In discussing concepts, a useful metaphor is that of a *set* of instances residing in a *space* of instances or examples. (The notion of *what space* to use is a complex one depending on context, available features, and so on. Given the space, *what topology* or *metric* to endow it with depends critically on what sort of similarity one wants to emphasize. These are not small issues.) Instances in the set's interior are "in" the concept class (that is, positive exemplars), and those that are "outside"

aren't (negative exemplars). As in mathematics, sets are often imagined or pictured as roundish areas sitting in a 2D plane. Of course, few concepts reside in such a low-dimensional space, and fewer still have such simple (for example, connected) topology. Nonetheless, the picture is useful because it facilitates metaphorically borrowing ideas such as boundary, interior, and the like from mathematics.

Messy concepts have three characteristics: they have gray areas of interpretation, they change, and they have exceptions. In other words, they are *open textured*, *nonstationary*, and *nonconvex*.

### Open-textured concepts

Open-textured concepts cannot be defined by universally valid, necessary and sufficient conditions. They don't have hard boundaries offering black-and-white distinctions between positive and negative instances. They are more like open sets in that their boundaries are soft and gray and always allow a bit of wiggle room.

The world is full of such concepts. For instance, if a regulation forbids taking a "vehicle" into a public park, what counts as a vehicle? A bicycle, kiddie-car, motorized wheelchair, park department work truck? A Segway? A rickshaw, pedicab, or palanquin? In discussing such concepts, the legal philosopher H.L.A. Hart introduced the "penumbra-core" distinction:

[T]hen the general words we use—like "vehicle" in the case I consider—must have some standard instance in which no doubts are felt about its application. There must be a core of settled meaning, but there will be, as well, a penumbra of debatable cases in which words are neither obviously applicable nor obviously ruled out. These cases will each have some features in common with the standard case: they will lack others or be accompanied by features not present in the standard case.<sup>12</sup>

While instances clearly situated in the interior region—the core—might be easy to classify, those in the boundary region—the *penumbra*—aren't. In the penumbra, there are often good reasons to call an instance "in" and equally good ones to call it "out."

### Nonstationary concepts

Nonstationary concepts change over time. The change might be gradual ("normal science" or "concept drift") or abrupt (a "paradigm change or shift").<sup>13</sup> Furthermore, change is ubiquitous. Sometimes concepts begin to change as soon as they come into being; at

other times, they occur after a long period of dormancy or settledness. Changes are often provoked by some anomaly or other disturbing example or circumstance that calls into question the concept's meaning or scope. Sometimes the catalyst comes from new data, and sometimes from new ways of looking at old data.

Real-world concepts can exhibit tremendous change. The legal domain provides numerous examples. A few familiar ones from American constitutional law are privacy, protected speech, due process, obscenity, and taxable income.

Even long-settled legal concepts can become unglued and undergo change so intense that the exceptions cause a complete flip in the rule. For example, the legal scholar Edward Levi described how the rise of the "inherently dangerous" exception and manufacturer liability in contract and commercial law reversed the once-supreme rule of privity of contract.<sup>14</sup> (Up until the mid-1800s, the privity rule required that to be compensated for a defect in a product, you had to be in a direct relationship with the manufacturer and not a "third party" or "remote purchaser." In a nutshell: "No privity, no liability.") Some concepts, such as "separate but equal," can even die.

The situation with mathematical and scientific concepts is similar. Change is often triggered by examples of a particularly unsettling nature: counterexamples. The history of mathematics is full of change in fundamental concepts (for example, natural numbers, irrational numbers, and functions).<sup>15</sup>

In his wonderful book *Proofs and Refutations*,<sup>16</sup> Imre Lakatos recounts how during the mathematical history of Euler's formula, changes in key concepts such as face, edge, and polyhedron were triggered by pesky examples (Kepler's "urchin," for example). He describes how such "monsters" catalyzed attempts to rejigger definitions in order to bar certain exceptional cases. He goes on to describe the push and pull between (counter)examples and (re)definition and the dialectic of refutation, reformulation, and rejustification. George Pólya called this ubiquitous dialectic the "alternating process."<sup>16</sup>

### Nonconvex concepts

Concepts can have "holes"—that is, negative examples residing in the concept's interior where (only) positive examples ought to be, and vice versa—and thus can fail to be

convex. Holes often arise from exceptional cases that come to light after a concept has been initially formulated.

Nonconvexity is intimately bound up with conceptual change involving counterexamples and methods of monster barring. Sometimes, a hole is caused by nothing more than a one-time anomaly; at other times, it's more sizable and contains a bevy of counterexamples. Sometimes the hole can be patched—for instance, by fixing an aspect of the concept's definition, monster barring, narrowing the discussion's setting, or redefining the instance space—and sometimes it can't. Often, the hole simply grows larger and larger in the face of a stream of challenging examples. Many areas of the law have bodies of

Even long-settled legal concepts can become unglued and undergo change so intense that the exceptions cause a complete flip in the rule.

exceptions that have grown over time, sometimes to the point of "swallowing the rule."

Nonstationary, nonconvex, open-textured concept can thus be likened to a pliable slice of Swiss cheese with blurry holes and morphing boundaries. While concepts more akin to well-machined designs cut into sheet metal are important, I think they're less interesting, even though the process of *how* they get hammered out so that they are clear-cut, fixed, and convex is exceedingly interesting. Perhaps more to the point for AI, the study of flexible, slippery concepts has great potential for shedding light on how we think and how to build robust systems capable of dealing with the evolving, messy world we and our systems need to thrive in.

AI has had a long, abiding interest in many aspects of messy concepts (see the sidebar "AI's Abiding Interest in Messy Concepts"). Core research in AI has contributed to our understanding of many foundational issues: commonsense concepts, default reasoning, exceptional cases, case memory, analogy, and classification, to name just a few.

### Synergies

The study of messy concepts isn't the sole province of AI, of course. It has been of long-standing interest in other disciplines such as philosophy, psychology, and law, and we ought to borrow more liberally from their insights. It was Ludwig Wittgenstein, for instance, who coined the term *family resemblance* and provided the famous example of a "game"; it's a classic example of an open-textured concept. Linguists and philosophers have long pursued natural concepts and their meaning.<sup>17,18</sup> Ethnographic studies—for instance, of concepts used by the Itzaj Maya of Guatemala and other cultures—have given us insights about concept formation far beyond those from our usual subject base (typically undergraduate psychology majors).<sup>19,20</sup>

Much of legal reasoning—whether in common, statutory, or constitutional law—focuses on formation and change in the meanings of concepts.<sup>14</sup> For instance, the problem of statutory interpretation focuses on the meaning of words in statutes and regulations.<sup>21,22</sup> In Anglo-American law, the doctrine of *stare decisis*—the very heart of our jurisprudence—is all about similarity: it dictates that similar cases should be decided similarly (even though similarity itself isn't spelled out in much detail). Basically it is reasoning by example.

Psychologists such as Eleanor Rosch,<sup>23,24</sup> Douglas Medin<sup>25–27</sup> and Gentner<sup>8,28</sup> have given us fundamental insights about such notions as family resemblance, typicality, basic categories, and analogy by structure mapping. Some have studied differences in novices and experts.<sup>29</sup> This foundational research is summarized in Gregory Murphy's *Big Book of Concepts*.<sup>30</sup>

Psychology has long since abandoned the view that concepts are defined by universally valid, necessary and sufficient features (the so-called *classical* or *definitional* view), replacing it with new paradigms (the *exemplar*, *prototype*, and *knowledge* views). In the prototype paradigm, categories have a graded structure, with some instances being more "typical" than others in that they share many features common in the category and share few features common to other categories. In some versions of this paradigm, a prototype is a maximally typical actual example; in others, it's more of a summary representation or ideal of what a most typical example would be. In either view, typical examples are closer to the prototype, and atypical examples and borderline examples—penumbral examples

AI has addressed fundamental questions about how to represent real-world concepts. Note, for instance, Marvin Minsky's well-known seminal work on frames and common sense and subsequent discussion about them.<sup>1,2</sup> CBR has addressed similarity issues head on. Roger Schank and his students have led the way on explanation and similarity-based problem solving.<sup>3-5</sup> Closely related work has explored example-based approaches.<sup>6-8</sup>

The subfield of AI and law has developed models of case-based argument, explanation, and hypothetical reasoning.<sup>9-11</sup> Projects have delved into how cases, examples, and explanations are intertwined,<sup>12,13</sup> how to represent complex concepts such as "ownership";<sup>14</sup> and the role of examples—hypothetical, prototypical, and real—in systems that do legal reasoning.<sup>15-18</sup>

From AI's earliest days to the present, researchers have studied analogy.<sup>19-23</sup> Thomas Evans' Analogy program, one of AI's earliest projects, dealt with it explicitly, and Patrick Winston's Arch program was one of the earliest to tackle learning and analogy together. Karl Branting's legal-reasoning system GREBE (Generator of Exemplar-Based Explanations)<sup>12</sup> adopted the structure-mapping approach developed by Dedre Gentner and Ken Forbus.<sup>24,25</sup> Manuela Veloso combined CBR and Jaime Carbonell's derivational analogy.<sup>26</sup> Recently Forbus and his group revisited the geometric analogy problems Evans studied, and developed a two-level analogical-processing scheme (involving first structure mapping and then reasoning with differences) that can produce human-like performance.<sup>23</sup>

References

1. R.J. Brachman, "I Lied about the Trees," *AI Magazine*, vol. 6, no. 3, 1985, pp. 80-93.
2. M. Minsky, P. Singh, and A. Sloman, "The St. Thomas Common Sense Symposium: Designing Architectures for Human-Level Intelligence," *AI Magazine*, vol. 25, no. 2, 2004, pp. 113-124.
3. R.C. Schank, *Explanation Patterns: Understanding Mechanically and Creatively*, Lawrence Erlbaum, 1986.
4. K.J. Hammond, *Case-Based Planning*, Academic Press, 1989.
5. D.B. Leake, *Evaluating Explanations: A Content Theory*, Lawrence Erlbaum, 1992.
6. E.L. Rissland, "Example Generation," *Proc. 3rd Nat'l Conf. Canadian Soc. Computational Studies of Intelligence*, Canadian Soc. for Computational Studies of Intelligence, 1980, pp. 280-288.
7. G. DeJong and R. Mooney, "Explanation-Based Learning: An Alternative View," *Machine Learning*, vol. 1, no. 2, 1986, pp. 145-176.
8. R. Bareiss, *Exemplar-Based Knowledge Acquisition: A Unified Approach to Concept Representation, Classification, and Learning*, Academic Press, 1989.
9. K.D. Ashley, *Modeling Legal Argument: Reasoning with Cases and Hypotheticals*, MIT Press, 1990.

10. L.K. Branting, *Reasoning with Rules and Precedents*, Kluwer, 2000.
11. E.L. Rissland, K.D. Ashley, and R.P. Loui, "AI and Law: A Fruitful Synergy," *Artificial Intelligence*, vol. 150, nos. 1-2, 2003, pp. 1-17.
12. L.K. Branting, "Building Explanations from Rules and Structured Cases," *Int'l J. Man-Machine Studies*, vol. 34, no. 6, 1991, pp. 797-837.
13. D.B. Skalak and E.L. Rissland, "Arguments and Cases: An Inevitable Intertwining," *Artificial Intelligence and Law*, vol. 1, no. 1, 1992, pp. 3-44.
14. L.T. McCarty, "Ownership: A Case Study in the Representation of Legal Concepts," *Artificial Intelligence and Law*, vol. 10, nos. 1-3, 2002, pp. 135-161.
15. L.T. McCarty and N.S. Sridharan, "The Representation of an Evolving System of Legal Concepts: I. Logical Templates," *Proc. 3rd Nat'l Conf. Canadian Soc. for Computational Studies of Intelligence*, Canadian Soc. for Computational Studies of Intelligence, 1980, pp. 304-311.
16. E.L. Rissland, "Examples in Legal Reasoning: Legal Hypotheticals," *Proc. 8th Int'l Joint Conf. Artificial Intelligence (IJCAI 83)*, Morgan Kaufmann, 1983, pp. 90-93.
17. E.L. Rissland, E.M. Valcarce, and K.D. Ashley, "Explaining and Arguing with Examples," *Proc. 4th Nat'l Conf. Artificial Intelligence (AAAI 84)*, AAAI Press, 1984, pp. 288-294.
18. A.v.d.L. Gardner, *An Artificial Intelligence Approach to Legal Reasoning*, MIT Press, 1987.
19. T.G. Evans, "A Program for the Solution of Geometric-Analogy Intelligence Test Questions," *Semantic Information Processing*, M. Minsky, ed., MIT Press, 1968, pp. 271-351.
20. P.H. Winston, "Learning Structural Descriptions from Examples," *The Psychology of Computer Vision*, P.H. Winston, ed., MIT Press, 1975, pp. 157-209.
21. P.H. Winston, "Learning and Reasoning by Analogy," *Comm. ACM*, vol. 23, no. 12, 1980, pp. 689-703.
22. M.A. Finlayson and P.H. Winston, "Intermediate Features and Informational-Level Constraint on Analogical Retrieval," *Proc. 27th Ann. Meeting Cognitive Science Soc.*, Lawrence Erlbaum, 2005, pp. 666-671.
23. T. Emmett et al., "A Structure Mapping Model for Solving Geometric Analogy Problems," *Proc. 27th Ann. Conf. Cognitive Science Soc.*, 2005, pp. 2190-2195.
24. D. Gentner, "Structure-Mapping: A Theoretical Framework for Analogy," *Cognitive Science*, vol. 7, no. 2, 1983, pp. 155-170.
25. B. Falkenhainer, K.D. Forbus, and D. Gentner, "The Structure-Mapping Engine: Algorithm and Examples," *Artificial Intelligence*, vol. 41, no. 1, 1989, pp. 1-63.
26. M.M. Veloso and J.G. Carbonell, "Derivational Analogy in Prodigy: Automating Case Acquisition, Storage and Utilization," *Machine Learning*, vol. 10, no. 3, 1993, pp. 249-278.

in Hart's sense—are much further away. In the exemplar paradigm, categories are represented in toto by the known positive examples of the category; it is a completely extensional representation. The knowledge view captures information such as commonsense

knowledge and the purposes of the category. While there isn't universal agreement concerning views, there's no going back to the simplistic classical view.

Our allies in other fields have much to offer us, and we them. For instance, while

similarity is indeed the keystone of precedent-based reasoning, what jurisprudence has to say about similarity is nowhere concrete enough for AI purposes. When we build AI programs that reason with precedents, we cannot avoid taking a computational stance

on what similarity means, and this can't help but inform jurisprudential ideas about it.

Likewise for psychology, while ideas about similarity are central to much work on categories, they're sometimes woefully vague or limited, especially in the exemplar view. In the prototype view, the similarity measure, while sufficiently defined, is rather simplistic: it's essentially a threshold for a linear evaluation function. In both views, there's too little discussion of what constitutes an example, and how to index, retrieve, and (partially) match them. By comparison, CBR has offered some detailed investigations of these issues. Work on knowledge representation—for instance, on frames and common sense—can inform the prototype and knowledge views. A case in point is Michael Pazzani's research<sup>31</sup> on how prior expectations and background knowledge can enhance category learning. Veloso's current work situates ideas about concepts in a robot environment.<sup>32</sup>

Disappointingly, the categories examined in psychology experiments are often—natural categories aside—overly simple and impoverished. Furthermore, the context aspired to in psychology experiments can be pristine to the point of being empty, and subjects can be viewed as *tabulae rasae*. These approaches aren't, in my view, the way forward. Psychology seems reluctant to explore or espouse hybrid approaches—such as a mixture of prototype and exemplar views—even though evidence exists for them.<sup>30</sup> By contrast, AI has made significant use of hybrids, and researchers are currently even more inclined to do so.

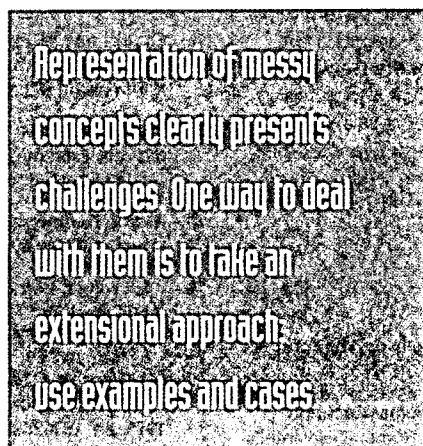
All these allied fields have much work to do. None addresses well enough the fact that notions of similarity are not static. Nonetheless, there's synergy to be exploited among such scientifically allied fields, and analogies across them might suggest new questions and new approaches. At the very least, both psychology and AI should broaden their investigations to include more research on messy concepts, and each should look to the other for cross-fertilization.

## Representing messy concepts

Representation of messy concepts clearly presents challenges. For instance, it's difficult, if not impossible, to represent them in the paradigm of mathematical logic with a fixed set of features or predicates and restrictions to sound inference rules. Probabilistic methods aren't the answer either. Although they can deal with some aspects of uncer-

tainty and classification, they're not particularly well suited for providing explanations, analogies, comparisons, contrasts, and so on.

For the most part, AI has focused more on working around messy concepts rather than on accepting them at face value. Researchers seldom exploit open-texture or conceptual change or, dare I say, rejoice in it. AI for the most part treats such characteristics as difficulties. I feel quite the opposite: they're features, not bugs. They permit a degree of flexibility that is an immeasurably valuable asset in domains such as law where concepts hardened in cement would be unacceptably brittle. By the way, it's not for lack of trying or diligence that fields such as law can't tame the open-textured, nonstationary, nonconvex



nature of their concepts; rather, I believe it is in their very nature. Instead of trying to cure them, perhaps we should simply embrace the fact that they're open-textured "all the way down," so to speak—that the one constant is change, and exceptions aren't exceptional.

One way to deal with messy concepts is to take an extensional approach: use examples and cases. Obviously, one can use an extensional approach in concert with others, such as logic, prototypes, frames, or statistics. We can sum up the example-based approach with Oliver Wendell Holmes's well-known aphorism that "the life of the law has not been logic: it has been experience." Adding experience—that is, examples—to the mix, in my mind, can only help.

Even in mathematics, it is exceedingly difficult to make progress on hard problems without the aid of examples and cases. Working mathematicians need them for checking conjectures and for coming up with them in the first place.<sup>6</sup> They play a key role in developing understanding. Mathematical reason-

ing involves a highly interconnected network of many types of knowledge, including theorems, definitions, proofs, heuristics, and examples, and there are many types of each—particularly examples.<sup>7</sup>

A problem for most of us with the extensional approach is that the only examples we've ever encountered are typical ones (in the core); we've never had to grapple with strange examples (in the penumbra). Intuitively—to paraphrase Justice Potter Stewart—we might know what to do with an easy example when we see one, but we really don't know how to deal with hard ones. Much of professional education in disciplines such as mathematics and law is about penumbral examples. The examples we use for induction often come from the core; only later on, when we begin to explore our ideas' ramifications, do we tackle penumbral examples. We've also been given contradictory messages about examples: Use examples to perform induction, but be wary of them because no example ever proves a rule (recall what your high school geometry teacher told you).

In the probabilistic sense, we tend not to be well acquainted with the rare instances in the tails of distributions. This paucity of examples often forces us to create hypotheticals or otherwise populate the little-encountered parts of the instance space to give us added experience with atypical situations. Instances and concepts are inevitably intertwined in the continuing cycle of conceptualization, testing, and refinement.

In putting together a hybrid approach, we can take a lead from the core-penumbra metaphor and, for instance, use rules, prototypes, or models for the core and cases or examples for the penumbra. We can view examples as annotations on the rules or models. Instead of a rule and its constituent predicates needing to be revised at every turn, cases can serve as updates to their meaning and scope. They are extensional patches. When (and if) the rules are revised, the cases can be taken into consideration. Over the years, various CBR systems have successfully used rule-case hybrids.<sup>33-35</sup> Cases have also been used to extensionally define principles in ethics.<sup>36</sup>

Another hybrid approach would be to use probabilistic approaches in concert with symbolic ones—for instance, statistical methods to propose concepts and classifications, and symbolic methods to provide precedent-based explanations and hypotheticals to test them out. Perhaps a three-way approach using cases and examples, rules and prototypes, and

probabilistic methods would combine the best of many worlds.

To get a toehold on representation, we often make hefty assumptions that really don't hold with real-world concepts. One is that the features, predicates, or descriptors in which a concept is couched don't change. Another is that features aren't open-textured or hard to evaluate. In reality, features often change while the concept is changing, and they can be as tricky to identify and work with as the concept they represent.

As Murphy says, "category learning itself may be needed to identify features." Robert Collins and I discussed this same problem in the context of Levi's well-known legal example using a version space approach.<sup>37</sup> It's a central problem in statutory law, where rules tend to change in the same time frame as their ingredient terms do. For instance, in American personal (Chapter 13) bankruptcy law, the concept of a "good plan" changed at the same time that the underlying predicate of "good faith" (and, in turn, its underlying feature of "disposable income") changed. In the Lakatos example, the substrate of concepts about faces, edges, and so on changed at the same time that the concept of Eulerian polyhedra did.

Such issues have implications for all manner of human and machine learning, in that the representational framework in which the learning is supposed to take place often changes simultaneously with the concepts, generalizations, rules, prototypes, and so on being learned. Although pragmatically we need to "live" in some space to get something started, we'll need to revisit these deep problems. We can't sweep them under the rug forever.

### Example-based probing

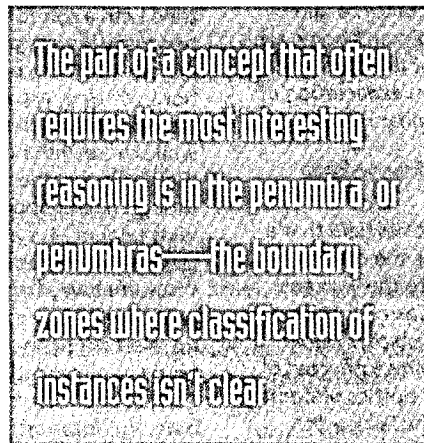
The part of a concept that often requires the most interesting reasoning is in the penumbra, or penumbras—the boundary zones where classification of instances isn't clear. One way to probe the penumbra is through hypotheticals—that is, made-up instances. (See the sidebar "An Example from Law.") In the penumbra, actual experience can be sparse (otherwise, the issues would probably have been fleshed out), and hypotheticals enrich the store of experience.

There are two aspects of probing with hypotheticals or examples:

1. *Specification*—deciding, out of a vast instance space, what (few) artful examples to use.

2. *Generation*—producing examples that satisfy the specifications. I call this *constrained example generation*, or CEG.

The constraints for probing examples come from the reasoning context. For instance, in mathematics, if you suspect that a conjecture (say  $A \Rightarrow B$ ) is false and want to disprove it, you need a counterexample—that is, an instance having property A and not property B. If you believe the conjecture is true, you want a supportive, positive example—that is, an example having both properties. Teachers often put a lot of care into specifying examples to motivate, illustrate, refine, and limit the concepts they're teaching. They take into account not only domain-



specific specifications but also those specific to their students and their own tastes, capabilities, and so on. Examples and hypotheticals are often used sequentially.<sup>7,38</sup> For adults, children, and machine learners, the examples' order can be important.<sup>39-41</sup>

To explore the limits of a line of reasoning, an appellate judge or law school professor might pose a hypothetical that is more extreme than the case at hand in order to press the advocate or student about whether the proffered line of reasoning should extend to the part of the instance space exemplified by the hypothetical. A series of hypotheticals might be constructed in the manner of a "slippery slope" sequence. Hypotheticals can include facts that support two conflicting lines of reasoning (for example, a house on wheels). Many argument strategies and tactics involve hypotheticals.<sup>42</sup>

CEG uses a retrieval-plus-modify strategy: satisfy as many constraints as possible through retrieval from a corpus of examples and then modify the best candidates with

domain-specific methods to try to satisfy the other constraints.<sup>43-45</sup> This is essentially a case-based approach, so CEG can be considered an early CBR precursor.

The HYPO project, which grew out of CEG, initially set out to model the creation of hypotheticals in law. It culminated, in Kevin Ashley's 1987 doctoral dissertation, in a full-scale computational model for case-based legal argument encompassing point-counterpoint thrusts and parries, methods to compare cases, and methods for creating hypotheticals.<sup>46-48</sup> HYPO was one of the earliest "interpretive" CBR systems: its goal was to interpret a new fact situation in light of existing precedents. HYPO has since had many progeny, including a mixed-paradigm cases-and-rules system called CABARET (for *case-based reasoning tool*) for the income tax law domain<sup>33</sup> and a case-based tutoring system called CATO to help law school students learn how to argue.<sup>49,50</sup>

A key mechanism in HYPO-style systems is a *dimension*.<sup>46,48,51</sup> Dimensions capture the knowledge that different ways exist to argue about an issue. They're an intermediate level of representation that encodes the knowledge that certain sets of facts enable a legal dispute to be approached in a particular way, and that changes in these facts tend to strengthen or weaken the argument of one side or the other. Dimensions focus attention on important aspects of cases. For instance, in HYPO's domain of misappropriation of trade secrets, the dimension called "secrets voluntarily disclosed" captures the idea that the more disclosures you (the plaintiff) have made of your putative secret, the less convincing is your argument that a particular party (the defendant) is responsible for letting the cat out of the bag (the secret). Dimensions are a hallmark of HYPO-style systems. There are several ways to use dimensions to generate hypotheticals, including making a case weaker or stronger, making a case extreme, enabling a near-miss, disabling a near-hit, and making a conflict hypothetical.<sup>52</sup> HYPO used such heuristics to generate hypotheticals in the context of an argument.<sup>48</sup>

Of course, there are other ways to generate hypotheticals that don't rely on dimensions. For instance, if you have metricized the instance space, you can create an instance lying midway between the centers of clusters of positive and negative examples, between the center of the core and the nearest unlike neighbor, between a particular case



The sort of reasoning done in Anglo-American appellate law provides good examples of what we (and our colleagues in psychology) should consider in future forays into similarity-based reasoning, explanation, analogy, learning, and teaching with nonstationary, nonconvex, open-textured concepts (see main article for more details). The law is rife with such examples.

This example is taken from the body of Fourth Amendment law concerning the warrant requirement. The Fourth Amendment states: "The right of the people to be secure in their persons, houses, papers, and effects, against unreasonable searches and seizures, shall not be violated and no Warrants shall issue, but upon probable cause, supported by Oath or affirmation, and particularly describing the place to be searched, and the persons or things to be seized." The US Supreme Court has interpreted this to mean that to execute a search the police must have a warrant except in a few circumstances (for example, incident to an arrest). With respect to a person's home, there's a longstanding reluctance to allow exceptions to the warrant requirement—"a man's home is his castle" sums up the default position. One of the primary rationales for this doctrine is that people have an *expectation of privacy* in their homes, hotel rooms, and similar places.

### **Carroll and the rise of "The Automobile Exception"**

On the other hand, the US Supreme Court has held that a person's car is not nearly as sacrosanct. Quite the contrary—it established the "automobile exception" to the warrant requirement in the prohibition-era case of *Carroll v. United States*.<sup>1</sup> This case involved a warrantless search on a public highway of a bootlegger's car, which the police suspected of transporting whiskey. The police did find 68 bottles of bootleg whiskey, but not before they ripped apart the back seat.

In *Carroll*, the Supreme Court held that when the police have probable cause to believe that a moving or temporarily stationed vehicle contains the fruits or instrumentalities of a crime, they may search it without a warrant. To justify the exception, the *Carroll* opinion relied in part on the long tradition of warrantless search of ships, especially for smuggled or contraband goods. It reasoned that a vehicle's mobility creates such an *exigent* situation that an exception to the warrant requirement is constitutionally valid. It recognized

*a necessary difference between search of a store, dwelling house or other structure in respect of which a proper official warrant readily may be obtained, and a search of a ship, motor boat, wagon, or automobile, for contraband goods, where it is not practicable to secure a warrant because the vehicle can be quickly moved out of the locality or jurisdiction in which the warrant must be sought. (p. 153)*

However, it also wrote:

*In cases where the securing of a warrant is reasonably practicable, it must be used.*

In carving out the automobile exception, the *Carroll* case triggered a stunning conceptual change.

### **From Carroll to Carney**

In the years following *Carroll*, the vehicle exception was considered many times. For the most part, the dichotomy in the

class of places for Fourth Amendment purposes remained neatly cleaved into "homes" and "vehicles." In some cases, the Court questioned the exigency rationale of *Carroll* and used a competing rationale based on a *lesser expectation of privacy*—for instance, in *Cardwell v. Lewis*, it wrote:

*One has a lesser expectation of privacy in a motor vehicle because its function is transportation and it seldom serves as one's residence or as the repository of personal effects. A car has little capacity for escaping public scrutiny. It travels public thoroughfares where both its occupants and its contents are in plain view.<sup>2</sup> (p. 590)*

The Court also tried to keep the exception in check. For instance, in *Coolidge v. New Hampshire*, it remarked that

*The word "automobile" is not a talisman in whose presence the Fourth Amendment fades away and disappears.<sup>3</sup>*

Nonetheless, the vehicle exception became well-settled doctrine with a comfortable bifurcation—houses and cars—and two distinct ways of looking at the problem—exigency and privacy—with each having a gradation of circumstances. Situations with low expectation of privacy or high exigency (cars) are candidates for constitutionally valid warrantless searches, while those with high expectation of privacy and low exigency (houses) enjoy the warrant requirement's full protection. Although what happens otherwise—say, high expectation and high exigency—is arguable, at least what obtains in these prototypical cases appears clear.

Another way to picture the situation is to think of the concept of valid Fourth Amendment searches and seizures as an open set sitting in some (high-dimensional) instance space. A house is in the settled core. A hole exists in this set owing to *Carroll*, and a car is clearly a typical case in the core of this hole. An (interior) penumbral case—in the boundary of the *Carroll* hole—would be a Winnebago. An (exterior) penumbral case would be a camper's tent or a homeless person's cardboard box shelter.

Over the years, the search issue in the automobile context has been revisited again and again. There have been innumerable cases about searching containers (for example, suitcases and packages) in cars searched without a warrant. The Court created and even closed up exceptions. In a 1979 case (*Arkansas v. Sanders*), the Court held that an unlocked suitcase in a cab's trunk could not be searched without a warrant.<sup>4</sup> This created a hole in the hole of the automobile exception. Then in 1982 (*United States v. Ross*), it partially closed up the hole when it held that "if probable cause justifies the search of a lawfully stopped vehicle, it justifies search of every part of the vehicle and its contents."<sup>5</sup> In 1991, the Supreme Court completely overturned *Sanders* in *California v. Acevedo*.<sup>6</sup>

Thus, we don't have just a stoma in the concept, we have islands (of positive examples) within it, and holes within these. Visually, this is akin to a coral atoll with interior islands or even whole archipelagos within them. There are exceptions in the "main" home-as-castle part of the concept as well.

### **Carney**

Difficulties with the automobile exception were brought to the fore in *Carney v. California*, a perfect storm of a case.<sup>7</sup> *Carney* involved a warrantless search of a motor home for drugs.

A sequence posed to the attorney for the State of California:

- J: Well, what if the vehicle is in one of these mobile home parks and hooked up to water and electricity but still has its wheels on?  
A: If it still has its wheels and it still has its engine, it is capable of movement ... very quickly.  
J: Even though the people are living in it as a home, are paying rent for the trailer space, and so forth?  
J: Well, there are places where people can plug into water, and electricity, and do ... where people go and spend the winter in a mobile home. And do you think there would be no expectation of privacy in such circumstances?  
A: Well, I am not suggesting that there is no expectation ...  
J: May I inquire, just so I understand your position? Is it that the vehicle have wheels? Could a trailer without a tractor in front of it qualify?  
A: No. I don't think it would ... if the officer looks ... and determines that it has the objective indicia of mobility ...  
J: It has to be self-propelled?  
J: But then what about a self-propelled vehicle that's plugged into the plumbing and the electricity?  
J: And ... even if it had been parked there three months or so ...  
J: What about a camper's tent, if the camper takes his things out of the motor home and pitches a tent next to it?  
A: The motor home would be subject to search ... Not the tent ...  
J: Why wouldn't the tent be just as mobile as a self-propelled vehicle? I gather you can pull it down pretty fast -  
J: It doesn't have wheels, right?

Some hypotheticals posed to Carney's attorney:

- J: We're getting closer to your case. Suppose somebody drives a great big stretch Cadillac down and puts it in a parking lot, and pulls all the curtains around it, including the one over the windshield and around all the rest ...  
A: It does come closer to a home ...  
J: Well it has everything in the back of it that your [house] has ...  
A: Does it have a bed?  
J: Yes, yes.  
...  
J: [W]hat about a van. You see thousands of them on the road.  
A: A van ordinarily would not be subject to the motor home exception.  
J: Well, I've seen some with all the chairs, all the upholstered chairs in them.  
J: So you would say that if there is a car parked alongside the motor home ... they could enter the car but not the motor home?

**Figure A. Excerpts from the oral arguments in the US Supreme Court case *Carney v. California*. "J" indicates a hypothetical question from a Justice, and "A" indicates a reply by an advocate, either Carney's or the one for the State of California.**

The motor home was parked in a public parking lot, just a few blocks from the courthouse. A drug enforcement agent and others had it under surveillance for over an hour. They questioned a youth who had emerged from the motor home, and he told them that he had received marijuana in exchange for sex. The police then knocked on Carney's door. Carney stepped outside, and without his express consent or a warrant, the police entered and spotted marijuana. This led to his conviction on drug charges.

*Carney* presents a head-on clash between a citizen's expectation of privacy and the desires of the police to investigate unfettered where there is exigency. In the oral argument, the Justices probed with many hypotheticals (figure A offers some excerpts). As for the case's outcome, despite a strong dissent by Justices Stevens, Brennan, and Marshall, the Court reasoned that because Carney's motor home was "readily mobile," was parked in a public place "not regularly used for residential purposes," and was subject to many motor vehicle state regulations, it had a lesser expectation of privacy.

### Lessons learned

The *Carney* oral-argument example in figure A illustrates how hypotheticals can be used to probe a concept's penumbral regions. This extended example also illustrates how difficult it would be to craft a definition for a concept like constitutionally valid warrantless search. Heuristic rules of thumb do exist, but they're not hard and fast. Even if we could feed the

corpus of Supreme Court cases into a concept-formation engine and then use the induced rule (or prototype) to classify a new case, this still would not be a useful model of legal reasoning. In particular, judicial reasoning requires consideration of precedents—both pro and con—to justify a decision. Giving a high probability or a nice hyperplane isn't sufficient.

The *Carroll-Carney* example also illustrates the role of exceptional cases in concept change. One critical issue about concept change is identification of episodes of change, particularly their onset. While the beginning of a change that's provoked by a single, stunning example such as *Carroll* isn't that hard to spot, gradual change is usually harder to identify other than in retrospect, because a court might initially dismiss an early harbingering case as an anomaly or even wrongly decided.

### References

1. *Carroll v. United States*, 267 U.S. 132 (1925).
2. *Cardwell v. Lewis*, 417 U.S. 583 (1974).
3. *Coolidge v. New Hampshire*, 403 U.S. 443 (1971).
4. *Arkansas v. Sanders*, 442 U.S. 753 (1979).
5. *United States v. Ross*, 456 U.S. 798 (1982).
6. *California v. Acevedo*, 500 U.S. 565 (1991).
7. *Carney v. California*, 471 U.S. 386 (1985).

The Author



**Edwina Rissland** is a professor at the University of Massachusetts. Since 2003, she has been serving as program director for the Artificial Intelligence and Cognitive Science Program in the Division of Information and Intelligence Systems within the CISE Directorate of the US National Science Foundation. Currently she is leading the Robust Intelligence cluster. She also has taught at Harvard Law School. She received her PhD in mathematics from MIT. Her research interests include case-based reasoning, AI and legal reasoning, and cyberlaw. She is a Fellow of the American Association for Artificial Intelligence. She coauthored *Cognitive Science: An Integrated Approach* and was coeditor of the MIT Press series on AI and Law. She is also a founding member of the editorial board of *Artificial Intelligence and Law*. Contact her at the Dept. of Computer Science, Univ. of Massachusetts, Amherst, MA 01003; rissland@cs.umass.edu.

and its nearest like neighbor, and so on.<sup>53</sup> You could use hypotheticals on either side of a boundary—say, close pairs of like and unlike neighbors that straddle a known flip in classification.

In the next few years, I hope we pay more attention to dealing with the problems of concepts and problems of analogy and explanation. Concepts must be dealt with in their own right, particularly those that are open-textured, changing, and exception ridden. Our systems must learn to keep their versions up-to-date and responsive to changes in their application and their contexts.

We must understand better how one entity—human or not—can instruct, show, teach, advise, or argue with another. We must know more about explanation and justification—for instance, how one agent can convince or dissuade others about an interpretation or a course of action. We must better understand how explanation, learning, and concept change are inextricably intertwined.

In summary:

- Our systems must be able to deal with messy concepts, including learning their changes and detecting exceptions and anomalies. At the beginning, of course, we'll have to program in such dealings, but they must be widely alterable and adaptable at all levels.
- A key process is to be able to work with similarities (and differences) in the system's experiences, often at many levels at once. That is vital for handling analogies, so that lessons learned can be explained and applied suitably.
- Work on CBR, example-based reasoning,

and explanation can help us break through current boundaries to give AI a vaster arsenal of methods for similarity-based problem solving. We need more of it.

- As our systems work through different experiences, they should learn to discover new similarity measures and new techniques.

Right after discussing the CBR heuristic in *Computers and Thought*, Feigenbaum and Feldman suggest that one problem "ripe for attack" is the learning of heuristics, particularly those dealing with concepts. Perhaps in the next few decades, AI systems will be able to discover new approaches to similarity and new, powerful reasoning methods for analogy. Perhaps one day, as the great mathematician Stefan Banach was said to remark, they'll even be able to "see analogies between analogies."<sup>54</sup> ■

**Acknowledgments**

I thank Jake Beal, Paul Cohen, Ken Forbus, Dedre Gentner, Jim Hendler, Roger Hurwitz, David Waltz, Patrick Winston, and especially Oliver Selfridge for all their critical discussions and generous feedback on this article. I acknowledge the support of the US National Science Foundation in preparing this article. Any opinions, findings, conclusions, or recommendations expressed in this material are my own and do not necessarily reflect the views of the NSF.

**References**

1. E.A. Feigenbaum and J. Feldman, *Computers and Thought*, McGraw-Hill, 1963.
2. M.L. Minsky, "Steps toward Artificial Intelligence," *Proc. Inst. Radio Engineers*, vol. 49, 1961, pp. 8-30.
3. M.L. Minsky, "Some Methods of Artificial Intelligence and Heuristic Programming," *Mechanisation of Thought Processes*. Her

Majesty's Stationery Office, vol. 1, 1959, pp. 3-27.

4. A.L. Samuel, "Some Studies in Machine Learning Using the Game of Checkers," *Computers and Thought*, E.A. Feigenbaum and J. Feldman, eds., McGraw-Hill, 1963, pp. 71-105.
5. O.G. Selfridge, "Pandemonium: A Paradigm for Learning," *Mechanisation of Thought Processes*, Her Majesty's Stationery Office, vol. 1, 1959, pp. 511-526.
6. G. Pólya, *Mathematical Discovery: Understanding, Learning, and Teaching Problem Solving*, vol. 2, John Wiley & Sons, 1965.
7. E. Rissland Michener, "Understanding Understanding Mathematics," *Cognitive Science*, vol. 2, no. 4, 1978, pp. 361-383.
8. L.L. Namy and D. Gentner, "Making a Silk Purse Out of Two Sows' Ears: Young Children's Use of Comparison in Category Learning," *J. Experimental Psychology*, vol. 131, no. 1, 2002, pp. 5-15.
9. J. Loewenstein, L. Thompson, and D. Gentner, "Analogical Encoding Facilitates Knowledge Transfer in Negotiation," *Psychonomic Bulletin & Rev.*, vol. 6, 1999, pp. 586-597.
10. D.A. Garvin, "Making the Case: Professional Education for the World of Practice," *Harvard Magazine*, vol. 85, pp. 56-65.
11. R.E. Neustadt and E.R. May, *Thinking in Time*, Free Press, 1986.
12. H.L.A. Hart, "Positivism and the Separation of Law and Morals," *Harvard Law Rev.*, vol. 71, 1958, pp. 593-629.
13. T.S. Kuhn, *The Structure of Scientific Revolutions*, 2nd ed., enlarged, Univ. of Chicago Press, 1962.
14. E. Levi, *An Introduction to Legal Reasoning*, Univ. of Chicago Press, 1949.
15. M. Kline, *Mathematical Thought from Ancient to Modern Times*, Oxford Univ. Press, 1972.
16. I. Lakatos, *Proofs and Refutations: The Logic of Mathematical Discovery*, Cambridge Univ. Press, 1976.
17. G. Lakoff, *Women, Fire, and Dangerous Things: What Categories Reveal about the Mind*, Univ. of Chicago Press, 1987.
18. H. Putnam, "The Meaning of 'Meaning,'" *Mind, Language, and Reality*, vol. 2, Cambridge Univ. Press, 1975, pp. 215-271.
19. S. Atran et al., "Generic Species and Basic Levels: Essence and Appearance in Folk Biology," *J. Ethnobiology*, vol. 17, 1997, pp. 22-45.

20. B. Berlin, D.E. Breedlove, and P.H. Raven, "General Principles of Classification and Nomenclature in Folk Biology," *Am. Anthropologist*, vol. 75, 1973, pp. 214-242.
21. K.N. Llewellyn, *The Common Law Tradition: Deciding Appeals*, Little-Brown, 1960.
22. W. Twining and D. Miers, *How to Do Things with Rules*, 2nd ed., Weidenfeld and Nicolson, 1982.
23. E. Rosch and C.B. Mervis, "Family Resemblance: Studies in the Internal Structure of Categories," *Cognitive Psychology*, vol. 7, 1975, pp. 573-605.
24. E. Rosch et al., "Basic Objects in Natural Categories," *Cognitive Psychology*, vol. 8, no. 3, 1976, pp. 382-439.
25. D.L. Medin and M.M. Shaffer, "Context Theory of Classification Learning," *Psychological Rev.*, vol. 85, 1978, pp. 207-238.
26. E.E. Smith and D.L. Medin, *Categories and Concepts*, Harvard Univ. Press, 1981.
27. D. Medin, "Concepts and Conceptual Structure," *Am. Psychologist*, vol. 44, no. 12, 1989, pp. 1469-1481.
28. D. Gentner, "Structure-Mapping: A Theoretical Framework for Analogy," *Cognitive Science*, vol. 7, no. 2, 1983, pp. 155-170.
29. G.E.A. Solomon, "Conceptual Change and Wine Expertise," *J. Learning Sciences*, vol. 6, 1997, pp. 41-60.
30. G.L. Murphy, *The Big Book of Concepts*, MIT Press, 2002.
31. M.J. Pazzani, "Influence of Prior Knowledge on Concept Acquisition: Experimental and Computational Results," *J. Experimental Psychology: Learning, Memory, and Cognition*, vol. 17, no. 3, 1991, pp. 416-432.
32. M.M. Veloso, P.E. Rybski, and F. von Hundelshausen, "FOCUS: A Generalized Method for Object Discovery for Robots That Observe and Interact with Humans," *Proc. 2006 ACM Conf. Human-Robot Interaction*, ACM Press, 2006, pp. 102-109.
33. E.L. Rissland and D.B. Skalak, "CABARET: Statutory Interpretation in a Hybrid Architecture," *Int'l J. Man-Machine Studies*, vol. 34, no. 6, 1991, pp. 839-887.
34. L.K. Branting, "Building Explanations from Rules and Structured Cases," *Int'l J. Man-Machine Studies*, vol. 34, no. 6, 1991, pp. 797-837.
35. J. Hastings, L.K. Branting, and J. Lockwood, "CARMA: A Case-Based Rangeland Management Advisor," *AI Magazine*, vol. 23, no. 2, 2002, pp. 49-62.
36. B.M. McLaren, "Extensionally Defining Principles and Cases in Ethics: An AI Model," *Artificial Intelligence*, vol. 150, nos. 1-2, 2003, pp. 145-181.
37. E.L. Rissland and R.C. Collins, "The Law as a Learning System," *Proc. 8th Ann. Cognitive Science Soc. Conf.*, Lawrence Erlbaum Associates, 1986, pp. 500-513.
38. E.L. Rissland, "Examples in Legal Reasoning: Legal Hypotheticals," *Proc. 8th Int'l Joint Conf. Artificial Intelligence (IJCAI 83)*, Morgan Kaufmann, 1983, pp. 90-93.
39. S.E. Kuehne et al., "SEQL: Category Learning as Progressive Abstraction Using Structure Mapping," *Proc. 22nd Ann. Conf. Cognitive Science Soc.*, 2000, pp. 770-775.
40. L. Kotovsky and D. Gentner, "Comparison and Categorization in the Development of Relational Similarity," *Child Development*, vol. 67, 1996, pp. 2797-2822.
41. P.H. Winston, "Learning Structural Descriptions from Examples," *The Psychology of Computer Vision*, P.H. Winston, ed., MIT Press, 1975, pp. 157-209.
42. D.B. Skalak and E.L. Rissland, "Arguments and Cases: An Inevitable Intertwining," *Artificial Intelligence and Law*, vol. 1, no. 1, 1992, pp. 3-44.
43. E.L. Rissland, "Example Generation," *Proc. 3rd Nat'l Conf. Canadian Soc. Computational Studies of Intelligence*, Canadian Soc. for Computational Studies of Intelligence, 1980, pp. 280-288.
44. E.L. Rissland, *Constrained Example Generation*, COINS tech. report 81-24, Computer and Information Science Dept., Univ. of Massachusetts, 1981.
45. E.L. Rissland and E.M. Soloway, "Overview of an Example Generation System," *Proc. 1st Nat'l Conf. Artificial Intelligence (AAAI 80)*, AAAI Press, 1980, pp. 256-258.
46. E.L. Rissland, E.M. Valcarce, and K.D. Ashley, "Explaining and Arguing with Examples," *Proc. 4th Nat'l Conf. Artificial Intelligence (AAAI 84)*, AAAI Press, 1984, pp. 288-294.
47. K.D. Ashley and E.L. Rissland, "A Case-Based Approach to Modeling Legal Expertise," *IEEE Expert*, vol. 3, no. 3, 1988, pp. 70-77.
48. K.D. Ashley, *Modeling Legal Argument: Reasoning with Cases and Hypotheticals*, MIT Press, 1990.
49. V. Alevan and K.D. Ashley, "Automated Generation of Examples for a Tutorial in Case Based Argumentation," *Proc. 2nd Int'l Conf. Intelligent Tutoring Systems (ITS 92)*, LNCS 608, Springer, 1992, pp. 575-584.
50. V. Alevan, "Using Background Knowledge in Case-Based Legal Reasoning: A Computational Model and an Intelligent Learning Environment," *Artificial Intelligence*, vol. 150, nos. 1-2, 2003, pp. 183-237.
51. E.L. Rissland and K.D. Ashley, "A Note on Dimensions and Factors," *J. Artificial Intelligence and Law*, vol. 10, nos. 1-3, 2002, pp. 65-77.
52. E.L. Rissland and K.D. Ashley, "Hypotheticals as Heuristic Device," *Proc. 5th Nat'l Conf. Artificial Intelligence (AAAI 86)*, AAAI Press, 1986, pp. 289-297.
53. S.J. Delany et al., "Generating Estimates of Classification Confidence for a Case-Based Spam Filter," *Case-Based Reasoning Research and Development: Proc. 6th Int'l Conf. Case-Based Reasoning (ICCBR 05)*, H. Muñoz-Avila and F. Ricci, eds., Springer, 2005, pp. 177-190.
54. S.M. Ulam, *Analogies between Analogies: The Mathematical Reports of S.M. Ulam and His Los Alamos Collaborators*, Univ. of Calif. Press, 1990.

**IEEE**  
**Software**

**FUTURE TOPICS:**  
**Software Testing**  
**Global Software Development**  
**Software Engineering Challenges in Small Software Organizations**

[www.computer.org/software](http://www.computer.org/software)



# The 2006 IEEE International Conference on Information Reuse and Integration (IEEE IRI-2006) Heuristic Systems Engineering

September 16-18, 2006  
Hilton Waikoloa Village, Waikoloa, Hawaii, USA

<http://www.sis.pitt.edu/~iri06>



Sponsored by

The IEEE Systems, Man and Cybernetics Society & US Air Force Office of Scientific Research

## Honorary General Chair

Lotfi Zadeh  
University of California, Berkeley,  
USA  
zadeh@cs.berkeley.edu

## General Chairs

Stuart Rubin  
SPAWAR Systems Center, USA  
stuart.rubin@navy.mil  
Shu-Ching Chen  
Florida International University, USA  
chens@cs.fiu.edu

## Program Chairs

Du Zhang  
California State University, USA  
zhangd@ecs.csus.edu  
Taghi M. Khoshgoftaar  
Florida Atlantic University, USA  
taghi@cse.fau.edu  
James B.D. Joshi  
University of Pittsburgh, USA  
jjoshi@mail.sis.pitt.edu

## Workshop Chair

Eric Grégoire  
Université d'Artois, France  
gregoire@cril.univ-artois.fr

## Technical Chair and Treasurer

Willie Chang  
California State University, USA  
changw@ecs.csus.edu

## Publications Chair

V. Scott Gordon  
California State University, USA  
gordonvs@ecs.csus.edu

## Publicity Chair

Elena Ferrari  
University of Insubria, Como, Italy  
elena.ferrari@uninsubria.it

## Asian Liaison

Althea Liang Qianhui,  
Singapore Management University,  
Singapore,  
althealiang@smu.edu.sg

## Registration Chair

Ju-Yeon Jo  
California State University, USA  
joj@ecs.csus.edu

## Local Arrangements Chair

Louellen McCoy  
SPAWAR Systems Center, USA  
louellen.mccoy@navy.mil

## Webmaster

Saubhagya Ram Joshi  
University of Pittsburgh, USA  
srjoshi@mail.sis.pitt.edu

This year's conference theme addresses all aspects of **Heuristics** as they relate to the **Engineering** design and operation of scalable **Systems**. This theme was selected to reflect the ever-increasing role played by intelligent systems in the engineering design process as well as the ever-increasing need to integrate mathematical logics with heuristic search in complex industrial systems (e.g., heuristic solutions of the TSP). The common problem confronting such systems is the acquisition and evolution of heuristics. In particular, the evolution of heuristics can be formalized in the context of reuse and integration. The IEEE International Conference on Information Reuse and Integration will feature contributed as well as invited papers. Theoretical and applied papers are both included in this call. The conference program will include special sessions and open forum workshops. Several funding agency program directors – including NSF, ONR, et al. – will present an open panel discussion entitled, "Funding Opportunities in Information Reuse and Systems Engineering". The conference includes, **but is not limited to**, the areas listed below:

- Homeland Security
- Component-Based Design
- Data Mining and Knowledge Discovery
- AI & Decision Support Systems
- Heuristic Optimization
- Fuzzy and Neural Systems
- Heuristic Search
- Large Scale Data Integration
- Human-Machine Information Systems
- Reuse in Software Engineering
- Information Assurance
- Soft Computing
- Manufacturing Systems
- Knowledge Acquisition and Management
- Evolutionary Computing
- Case-Based Reasoning
- Modeling & Simulation
- Multimedia Reuse & Integration
- Internet Computing
- Knowledge Management and E-Government
- Agent-Based Systems
- Sensory and Information Fusion
- Natural Language Understanding
- Command & Control Systems (C4ISR)
- Space and Robotic Systems
- Biomedical Systems

## Instructions for Authors:

Papers reporting original and unpublished research results pertaining to the above and related topics are solicited. Full paper manuscripts must be in English of length 4 to 6 pages (using the IEEE two-column template). Submissions should include the title, author(s), affiliation(s), e-mail address(es), tel/fax numbers, abstract, and postal address(es) on the first page. Papers should be submitted at the conference web site: <http://www.sis.pitt.edu/~iri06>. If web submission is not possible, manuscripts should be sent as an attachment via email to one of the Program Chairs listed below on or before the deadline date of **June 6, 2006**.

### Du Zhang, Ph.D.

Department of Computer Science  
California State University  
6000 J Street  
Sacramento, CA 95819-6021,  
USA  
Email: [zhangd@ecs.csus.edu](mailto:zhangd@ecs.csus.edu)  
Phone: 1-916-278-7952  
Fax: 1-916-278-6774

### Taghi M. Khoshgoftaar, Ph.D.

Department of Computer Science  
and Engineering  
Florida Atlantic University  
777 Glades Road  
Boca Raton, Florida 33431, USA  
Email: [taghi@cse.fau.edu](mailto:taghi@cse.fau.edu)  
Phone: 1-561-297-3994  
Fax: 1-561-297-2800

### James B. D. Joshi, Ph.D.

Department of Information Science  
and Telecommunications  
University of Pittsburgh  
135 N. Bellefield Avenue  
Pittsburgh, PA 15260, USA  
Email: [jjoshi@mail.sis.pitt.edu](mailto:jjoshi@mail.sis.pitt.edu)  
Phone: 1-412-624-9982  
Fax: 1-412-624-2788

The attachment must be in .pdf (preferred) or word.doc format. The subject of the email must be "IEEE IRI 2006 Submission." Papers will be selected based on their originality, timeliness, significance, relevance, and clarity of presentation. Authors should certify that their papers represent substantially new work and are previously unpublished. Paper submission implies the intent of at least one of the authors to register and present the paper, if accepted. Authors of selected papers that are also presented at the conference will be invited to submit expanded versions of their papers for review for publication in an approved special issue of the IEEE SMC Transactions, part C, on IRI to be published in 2008.

## Important Dates:

- |                        |  |
|------------------------|--|
| <b>April 15, 2006</b>  | Workshop/Special session proposal ( <i>closed</i> )                    |
| <b>June 6, 2006</b>    | Paper submission deadline  |
| <b>June 16, 2006</b>   | Notification of acceptance   |
| <b>June 30, 2006</b>   | Camera-ready paper due   |
| <b>July 5, 2006</b>    | Presenting author registration due                                     |
| <b>July 30, 2006</b>   | Advance (discount) registration for general public and other co-author |
| <b>August 21, 2006</b> | Hotel reservation (special discount rate)                              |

## Keynote Speeches

- Dr. Mo Jamshidi**  
University of Texas, San Antonio, USA
- Lt. Col. Bill Nace**  
Technical Director, Asian Office of  
Aerospace Research and Development,  
US Air Force Office of Scientific Research
- Dr. Ljiljana Trajković**  
School of Engineering Science,  
Simon Fraser University, USA