2 Analogy-Making, Learning, and Generalization

Consider that I laboured not for myself only, but for all them that seek learning.

-Ecclesiastics, Old Testament

This chapter defines and examines the concepts of analogy-making, learning, and generalization and their relationship to each other.

2.1 Analogy-Making

To see similarity between two things that might not appear to have much in common on the surface is to make an analogy. In ordinary, everyday thought, analogy-making is a very natural and spontaneous process. When asked to think of a bird and an airplane, one naturally associates the head of the bird with the cockpit, the tail feathers with the rudders, the bird's wings with the plane's wings, the bird's legs with the plane's landing gear, and so on. The human penchant for analogy-making has fascinated philosophers of mind for millennia, psychologists for a century, and cognitive scientists working in the field of artificial intelligence (AI) since the first fledgling analogy programs were developed back in the 1960's. Analogy-making has been recognized as being of central importance in such diverse areas as linguistics (Harris, 1994), creativity and design (Hofstadter *et al.*, 1995), problem solving (Hellman, 1988), and scientific discovery (Holyoak and Thagard, 1995), to name just a few.

For the artificial intelligence researcher, the potential rewards of creating flexible computer programs capable of making analogies are enormous. How much easier, say, programming would be if one could just edit a section of code and then instruct the computer to "do the same thing" to another section, and have the computer adapt the operations in very clever, analogous ways. Unfortunately, current computer programs, even those that learn, tend to be very "literal minded." The hope is that computer programs can be made more intelligent and useful by incorporating an ability to make analogies.

To the cognitive scientist, the prospect of creating programs capable of making analogies is tantamount to understanding the cognitive phenomena of analogy-making itself. Analogy-making takes many forms, variously termed by cognitive scientists: recognition, categorization, induction, generalization, analogical reminding, analogy-making, analogical problem solving, and analogical reasoning. Each form of analogy-making is slightly different from the others, however all involve the *perception of similarity*. Humans are incredibly flexible at perceiving similarity in a multitude of ways. For example, consider the following vignettes:

1. A woman goes to a potluck picnic with a frisbee and some potato salad. After throwing the disc for awhile, she gets in line to eat dinner. She soon realizes that there aren't any plates left. Hungrily, she looks down and sees the frisbee, but now she sees it as a plate. She scoops the food onto her frisbee-plate and enjoys her meal.

2. Two grandparents walking along a trail get tired. They wish they had a couple of chairs, but then they see two large, flat boulders. They wearily walk over to the rocks, sit down, and have a nice rest.

3. Two graduate students are complaining about how much paperwork they are required to fill out and turn in to the administration when one student remembers that he forgot to send his evil step-mom a mother's day card.

4. John Rogers, a comedian, likens graduate school to "the snooze button on the clock radio of life," and his audience laughs.

5. A physics student begins to understand the properties of light by comparing and contrasting the behavior of water waves and marbles.

6. A little girl points to the moon and says "ball."

7. A doctor recognizes a set of symptoms and diagnoses a disease.

8. A presidential hopeful attempting to remind voters of past events, refers to another candidate's activities as "Whitewatergate", "Travelgate", and "Filegate".

Although some of these examples are more abstract than others, I believe that each of these represents an act of analogy-making. It is a wide spectrum ranging from mundane, low-level recognition and categorization to sophisticated, abstract analogy and metaphor. As the above examples illustrate, analogy-making appears both spontaneously and deliberately in the thought processes of people of all ages in many different kinds of situations. Understanding this wonderful, pervasive ability we call analogy-making would represent a giant leap forward in understanding the full breadth of human thinking.

2.2 Learning

Although everyone seems to know what it is, learning is actually very difficult to precisely define. Roughly, any system that improves its performance in response to internal changes caused by experience can be said to learn. However, this brief definition is too broad and too vague to be of much use. For instance, when we oil a squeaky door it may have had an "internal change", but we do not mean that it has learned anything about being quiet. Wine and cheese in the cellar may be improving their "performance" due to "experience", but we would not say that they are learning. Tempering a sword could be described as "improving its performance in response to internal changes caused by experience," and yet we do not mean that it has learned anything. Flowers interactively adjust themselves to face the sun, but, again, no one would claim that they have learned about the position of the sun. Then, what *do* we mean we mean when we say a system has learned? To better illustrate the learning phenomenon, consider these vignettes:

1. Within a few months, a baby has acquired the ability to recognize his mother.

2. After just a couple of years, a young toddler discovers how to get the undivided attention of her parents in many ways.

3. A young boy burns his hand on the stove. It never happens again.

4. A man trusts a stranger, and the stranger takes his wallet. It never happens again.

5. After years of practicing, a young woman develops an ability to play the tuba.

6. A man goes to a potluck picnic with a frisbee and some coleslaw. He soon realizes that there aren't any plates left. Hungrily, he watches a woman scoop food onto her frisbee-plate and enjoy her meal. He makes a mental note, and does the same.

7. A physics teacher explains the properties of light to a student by comparing and contrasting the behavior of water waves and marbles, and the student passes a test.

8. A robot's sonar sensor registers 0.85, it moves forward, and then its bumper sensor is activated. Later, after some internal adjustments, its sonar sensor again registers 0.85, it moves *backward*, and its bumper sensor remains inactive.

9. A man goes to medical school and develops the ability to diagnose a disease by examining many patients with the disease.

10. After studying a math textbook and practicing for many days, a boy is able to recite from memory the first 100 digits of π.

11. After watching Jeopardy, a woman in New Jersey now knows which mammal runs the fastest.

12. A man knows that if he receives a promotion then he will get a bonus. Later, he finds out that he will be promoted. Therefore, he deduces that he will get a bonus.

These examples demonstrate the wide range of phenomenon that we call learning. Cognitive scientists have given these forms of learning various names: learning by deduction, learning by abduction, learning by induction (also called learning from examples), learning by memorization, learning from a single example (also called one-shot learning), learning by analogy, rote learning, learning by instruction (also called learning by being told), and learning by observation, to name just a few too many. Although each of these forms of learning emphasizes a different aspect of learning, they all involve a change to an internal, persistent *memory* of the system. Other than this single fact, the above examples have little else in common.

As can be seen, 'systems that learn' is a diverse category. It also appears that the category is getting larger. Carbonell, Michalski, and Mitchell (1983) define a type of learning that involves the "direct implanting of new knowledge":

> …Variants of this knowledge acquisition method include: Learning by being programmed, constructed, or modified by an external entity, requiring no effort on the part of the learner (for example, the usual style of programming).
>
> (Carbonell, Michalski, and Mitchell, 1983)

This type of 'learning' is much too inclusive, encompassing not only sword tempering, and door oiling, but program writing, and building construction. For this dissertation, we will limit our discussion to the category ‘learning by example’. Furthermore, we will limit that category by considering only learning methods capable of *generalization*.

2.3 Generalization

The word ‘generalization’, as we normally use it, refers to the ability to infer the general from the particulars. For instance, if a robot with a visual system successfully learns to avoid bumping into a trash can, and can avoid the trash can even though it might be in a position never before seen, then we can say that the robot has learned to generalize from previous particular instances of trash can images. Generalizations of this type are often based on statistical regularities found directly in the sensory stimuli.

However, generalizations need not be tied to low-level perception. Recall the man from example 12 from above who got a promotion and then a bonus. One could say that he is simply using *modus ponens* to deduce the previously-unknown fact that he will get a bonus.[2] However, one could also argue that he is generalizing from examples he has seen before. Imagine that, instead of using *modus ponens*, he had made a generalization based on the following story he had read:

> A woman knows that if she gets caught dating a fellow employee then she will get fired. Later, she finds out she has been caught. Therefore, she predicts that she will get fired, and she does.

In effect, the man can be said to have learned *modus ponens* by example. Having generalized the previous story, he concludes that he will get a bonus. Notice that this type

[2] Recall that *modus ponens* is the logical rule of inference that states “if ‘A implies B’ is true, and ‘A’ is true, then ‘B’ necessarily follows.”

of generalization appears quite different from the generalization made by the robot viewing trash cans. The robot was able to generalize based on common properties found directly in its low-level visual system. Although, the generalization of the notion of *modus ponens* may be suggested by low-level perception (i.e., the examples had to enter the man's head through his senses), it is a generalization of abstractions, such as things known to be true, consequences of actions, etc.

Abstractions, such as 'truth' and 'consequences', are concepts. The word 'concept' has probably been used in more varieties of ways than 'learning'. Informally, a concept is a general 'idea' inferred from specific instances or occurrences. So, to the robot, 'trash can' is also a concept. In effect, concepts are themselves generalizations.

Therefore, the robot made generalizations based on similarity found directly (or nearly directly) in its low-level sensors, while the notion of *modus ponens* was a generalization based on similarity found in higher-level descriptions. Following Hofstadter's terminology, I shall call the process of perceiving from sensory stimuli *low-level perception*, and the process of perceiving from abstractions *high-level perception* (Hofstadter *et al.*, 1995).

We have seen that analogy-making is the perception of similarity between two things that could be very dissimilar on the surface, and learning by example is the process of altering an internal memory so that a system can make generalizations. Also, we have seen that generalizations are inferences formed from perceived similarity. Therefore, analogy-making and learning-by-example are intimately connected to the perception of similarity via the process of generalization.

2.4 Analogy-Making and Learning

Considering the amount of overlap between analogy-making and learning, it is ironic that the interaction between these two areas of research has been so limited. That is not to say that there has not been much research associating these two processes, but that research has traditionally been restricted to two categories of models: those that learn low-level perceptual processes, and those that learn by making analogies between high-level descriptions. Analogator defines a third category of models – those that learn what it means to make an analogy. The Analogator model will be examine in detail in Chapter 4. These first two categories will be explored in more detail here.

2.4.1 Low-level Perceptual Processing

Recognition and categorization based on sensory stimuli are two tasks studied via low-level perceptual processing models. In the last fifteen years, programs designed to learn low-level perceptual processes have flourished. Artificial neural networks (ANNs) have become the mechanism of choice for studying low-level perceptual processes. In the mid-1980's, research in ANNs, also called connectionist networks, was revitalized (after a long period of inactivity) due to the creation of many new and interesting learning algorithms. We will examine connectionist networks in detail in Chapter 3; briefly, a connectionist network is a model that is based loosely on neurons and can learn by being exposed to examples. Many such connectionist learning procedures do quite well with low-level perceptual tasks, such as the categorization of printed or hand-written characters. Also in the 1980's, the field of machine learning exploded with many other types of learning schemes designed for other generalization tasks (see, for instance, Marr, 1982; Holland, 1975; and Holland *et al.*, 1986). However, connectionist systems by their very nature are better suited to learning low-level perceptual processes than are many other machine learning techniques. This is due to the fact that connectionist networks can be directly connected to low-level stimuli, and can learn to re-represent a problem.

For example, connectionist-based optical character recognition (OCR) systems can view digital photographs (or similar input) and produce category labels after learning is complete. For instance, many large cities in the United States now have connectionist OCR systems capable of reading hand-written ZIP codes on envelopes to help with the automatic routing of mail. Of course, people's hand-writing varies widely, yet these systems work very well. Typically, the programs in this category have very narrowly defined tasks; they are designed to take low-level stimuli, such as a picture of the number 4, and produce a category, such as "four". Currently, these models are only capable of solving very basic problems, and, therefore, have had little connection to more high-level tasks.

2.4.2 High-level Conceptual Processing

The second category of models, those that learn by making analogies between high-level descriptions, has also had a recent surge of active research. Models in this category attempt to solve a problem by comparing it's highly abstracted representation to other abstracted representations of problems that the system has previously seen (and solved).[3] The idea is designed to work as follows. Imagine a robot attempting to vacuum a carpet. The robot plugs the vacuum cleaner in and the vacuum cleaner works for a moment, but then it stops. Not knowing much about how vacuum cleaners work, the robot searches its memory and recalls a similar event that happened before when it was attempting to, say, make some toast. It remembers that the toaster stopped working, but

[3] 'Representation' will be formally defined in Chapter 3.

was fixed by jiggling the toaster's lever. The robot recalls this toaster-episode from memory, *maps* it to the current situation, and, in a clever fashion, jiggles the vacuum cleaner's *on/off switch* (the analogous part to the toaster's lever). The vacuum cleaner comes back on, and the robot is able to complete its chores. As this story illustrates, there are two main goals of such an analogical model: 1) the retrieval from memory of similar episodes, and 2) the application of the recalled solution to the current problem in an analogous fashion. Notice that both goals completely rely on the perception of similarity: the first must see the similarity (or not) between the current problem and all other episodes stored in memory, and the second must match similar pieces of the retrieved memory episode with the current problem.

The basic idea of using analogy as a computational tool to solve new problems is relatively old, dating back to at least the early 1960's (Minsky, 1963). Many researchers throughout the last three decades have attempted to construct AI programs in this spirit. One of the most successful AI research paradigms ever is based on these basic ideas: Schank's Case-Based Reasoning (CBR) approach (Schank, 1982).

All of the models in this category posit the existence of an analogy-making 'engine'. That is, the mechanism that is responsible for seeing similarity and finding correspondences is hard-coded and never changes. Learning, if involved at all, is limited to altering and adding memory episodes rather than developing or honing analogy-making abilities. The perception of similarity in these models operates quite differently than the low-level perceptual models, as the representations used are very different from sensory stimuli. Each representation is a highly abstract description of the problem's gist. Because the representations take a very different form from those of low-level stimuli, learning is hardly ever incorporated into the modeling of the perception of similarity between highly abstracted descriptions.

In practice, a problem's representation cannot be too different from those of the solutions stored in memory or the program would be unable to see any similarity. Because of this limitation, much effort is spent attempting to get representations into the correct form. Also, these models generally suffer from a boot-strapping problem: the model needs to have a solution in memory to solve a new problem, but where do the initial memories come from? Typically, the researcher must supply them. Although the role of learning is restricted to altering memory rather than being used in the perception of similarity, the analogy-making process is of central importance. Generally, these types of models have little interaction with low-level sensory stimuli.

The goal of Analogator is to explore the large gap between those systems that learn the very process of perceiving similarity and those that are hard-coded to make analogies between abstractions. There are many issues that can be explored in this chasm: Can a single mechanism perform both low-level and high-level generalizations? Since they are generalizations themselves, can concepts be created by this same generalizing mechanism? What is the role of concepts in such a system? How could a system learn this general generalization mechanism? These are core questions in AI and cognitive science.

In order to explore these questions more fully, the traditional approach based on making analogies between high-level descriptions is described and examined in the following section.

2.5 A sketch of the traditional approach to analogy-making

Intuition has been the mother of invention, at least in AI. Many high-level cognitive tasks have been modeled by AI researchers implementing their intuitions. Planning, for instance, is a task that has been traditionally modeled in terms of "goals", "problem spaces", "goal states", etc. Intuitive concepts, such as goals, represent high-level descriptions that, it is hoped, mirror the actual cognitive processes and mental representations that people actually use to solve problems. Goals, problem spaces, and their like, have been suggested by introspection. That is, as people attempt to solve a problem, they report their self-perception of what is going on in their mind; people report in terms of goals and problem spaces. For instance, someone attempting to fix a car will say statements such as: “Before I can check the battery, I must open the hood. But I remember that the hood is stuck, so that idea leads to a dead end. I’m now going back to the drawing board…” In this scenario, there is a goal to check the battery, and a plan that involves opening the hood. It makes sense, it appears, to model planning using goals, goal states, etc.

Like planning, analogy-making has been traditionally modeled with similar introspective concepts. For analogy-making, the introspective concepts are items such as *correspondences, objects, attributes, relations*, and *mapping processes*. When expressed in these terms, one imagines analogy-making as creating explicit links via a “mapping process” between corresponding elements. So, if one were thinking of a bird and an airplane again, then one might model the analogy-making process by creating connections between the bird’s and the airplane’s representations. The two objects could be symbolized by high-level descriptions, such as those shown in Figure 2-1.

Mappings can now be made in a straightforward manner by matching the abstracted part names (i.e., CONTROL-SOURCE) in the two representations. So, for instance, *head* would map to the *cockpit*, the *tail-feathers* would map to *rudders*, etc. If one then asked, “What corresponds to the bird’s legs?” one could search through the representation of the bird, find *legs*, and simply follow the *SUPPORT* link to the plane’s representation to find *landing-gear*.

To learn to make analogies between birds and airplanes seems to require a very different mechanism that that of learning to generalize over different views of trash cans. Yet, combining these two types of generalization into a single framework is ultimately my goal. To examine this possibility further, we will need to see exactly how connectionist networks generalize.

2.6 Generalizing in connectionist networks

Connectionist learning networks, as we will see in Chapter 3, have the power to generalize over novel patterns. However, in order for this to happen, one of the following three conditions must be met:

```
(bird
     (CONTROL-SOURCE head)
     (LIFT-MECHANISM wings)
     (SUPPORT legs)
     (STEERING-MECHANISM tail-feathers))

(plane
     (CONTROL-SOURCE cockpit)
     (LIFT-MECHANISM wings)
     (SUPPORT landing-gear)
     (STEERING-MECHANISM rudders))
```

Figure 2-1. Abstract representations for comparing a bird with an airplane.

1. There must exist statistical regularities in the representations.

2. There must exist statistical regularities in the way that representations are used.

3. A combination of 1 and 2.

In this sense, generalization is merely *interpolation*. That is, networks must have experience with similar patterns in order to make useful generalizations with new ones. Unfortunately, Analogy-making is often seen as *extrapolation*. This implies that analogy-making is, in some way, more powerful than the generalization abilities of a network would allow. However, we shall see in Chapter 4 that there are some techniques to reduce extrapolation to interpolation. First, we must examine the connectionist mechanisms in more detail.