17 Learning by Explaining Experience

In this chapter, you learn how it is possible to *learn from experience* by working exercises. In particular, you learn about MACBETH, a procedure that works on précis of Shakespeare's plays. MACBETH *matches* an exercise description to a precedent description, then *transfers explanations* from the precedent to the exercise. The transferred explanation enables MACBETH to deal with the exercise and to construct an *antecedent–consequent rule*, thus performing a sort of *explanation-based learning*.

By way of illustration, you see how MACBETH learns about *causes*, learning, for example, that a weak character and a greedy wife can lead a noble to want to be king. You also see how MACBETH learns to relate *form and function*: given a functional definition for cups, MACBETH uses prior knowledge of other objects, together with physical descriptions of a particular cup, to construct a variety of possible physical descriptions, enabling cup identification.

Once you have finished this chapter, you will see that knowing a lot helps you to learn more.

LEARNING ABOUT WHY PEOPLE ACT THE WAY THEY DO

Teachers constantly supply precedents, give exercises, and expect students to do the exercises and to discover principles that apply to everyday situations. Students must find the correspondence between the precedents and

the exercises, use the precedents to deal with the exercises, generalize their work to form principles, and store the principles so that the principles can be retrieved when appropriate. This type of teaching and learning pervades subjects such as management, economics, law, medicine, and engineering, as well as aiding in the development of commonsense knowledge about life in general.

Consider, for example, the following exercise and précis of *Macbeth*:

Greed

This is an exercise about a weak noble and a greedy woman. The noble is married to the woman. Explain why the noble is likely to want to be king.

Macbeth

This is a story about Macbeth, Lady Macbeth, Duncan, and Macduff. Macbeth is an evil noble, and Lady Macbeth is a greedy, ambitious woman. Duncan is a king, and Macduff is a noble. Lady Macbeth persuades Macbeth to want to be king because she is greedy. She is able to influence him because he is married to her and because he is weak. Macbeth murders Duncan with a knife. Macbeth murders Duncan because Macbeth wants to be king, because Macbeth is evil, and because Duncan is king. Then Lady Macbeth kills herself. Finally Macduff gets angry and kills Macbeth because Macbeth murdered Duncan and because Macduff is loyal to Duncan.

Told by a teacher that *Macbeth* is to be considered a precedent, a human student should establish that the noble corresponds to Macbeth and the woman corresponds to Lady Macbeth. Next, the student should note that the characteristics of the noble and the woman are those that lead to Macbeth's desire to want to be king. From these correspondences, the student should form some sort of rule suggesting that the weakness of a noble and the greed of his wife can cause the noble to want to be king.

In the rest of this section, you learn how a program, MACBETH, can form explanations and learn new rules as well.

Reification and the Vocabulary of Thematic-Role Frames Capture Sentence-Level Meaning

The first step in using a precedent is to prepare a suitable description, cast in a good representation. For the *Macbeth* story, as given, an ordinary semantic net can represent certain characteristics of the people, their motivations, and their consequent actions. For example, Macbeth and Duncan can be denoted by nodes, with a Murder link between, as shown at the top of figure 17.1.

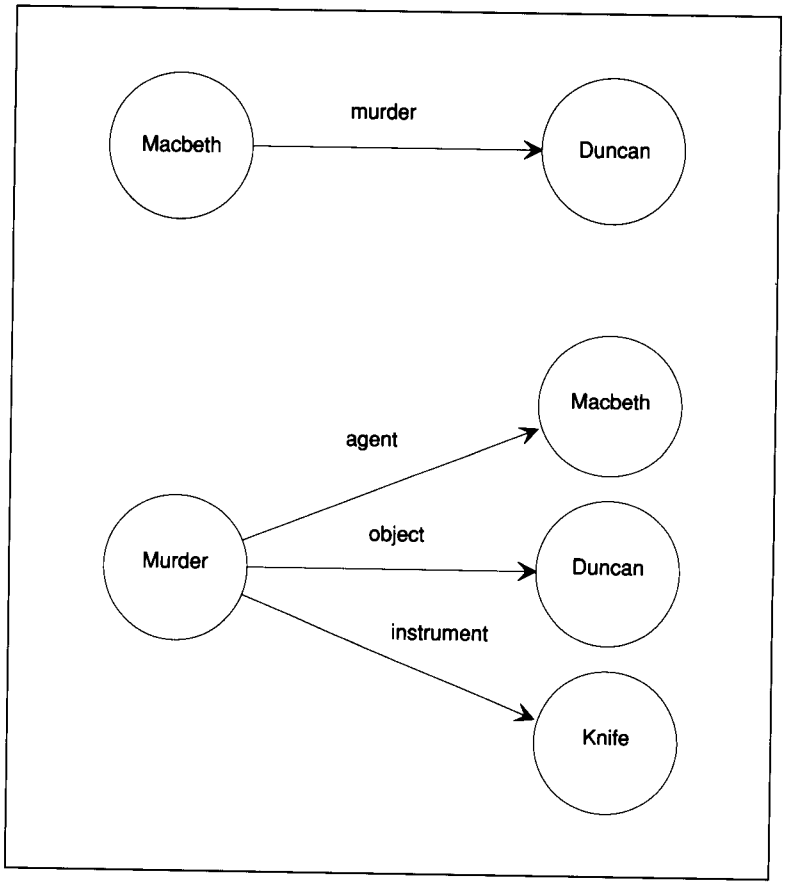

Figure 17.1 In the top part of the figure, Macbeth murders Duncan, as represented in the nodes and links of an ordinary semantic net. In the bottom part of the figure, the reified Murder link is treated as though it were a node with its own descriptive links.

There is more to be said about the murder, however, because Macbeth used a knife. Accordingly, you can think of the Murder link as though it were a node, with descriptive links of its own, as shown at the bottom of figure 17.1.

Treating something abstract and difficult to talk about as though it were concrete and easy to talk about is called **reification**. Thus, the elevation of a link to the status of a describable node is a kind of reification; when a link is so elevated, it is said to be a **reified link**. One notation for reified links is shown at the top of figure 17.2. To prevent illustrations from becoming cluttered, however, it is convenient to drop the circles, along with the redundant Agent and Object links, as shown at the bottom of figure 17.2.

Note that many links denote actions. Accordingly, the vocabulary used to describe links is reminiscent of the thematic-role-frame vocabulary introduced in Chapter 10.

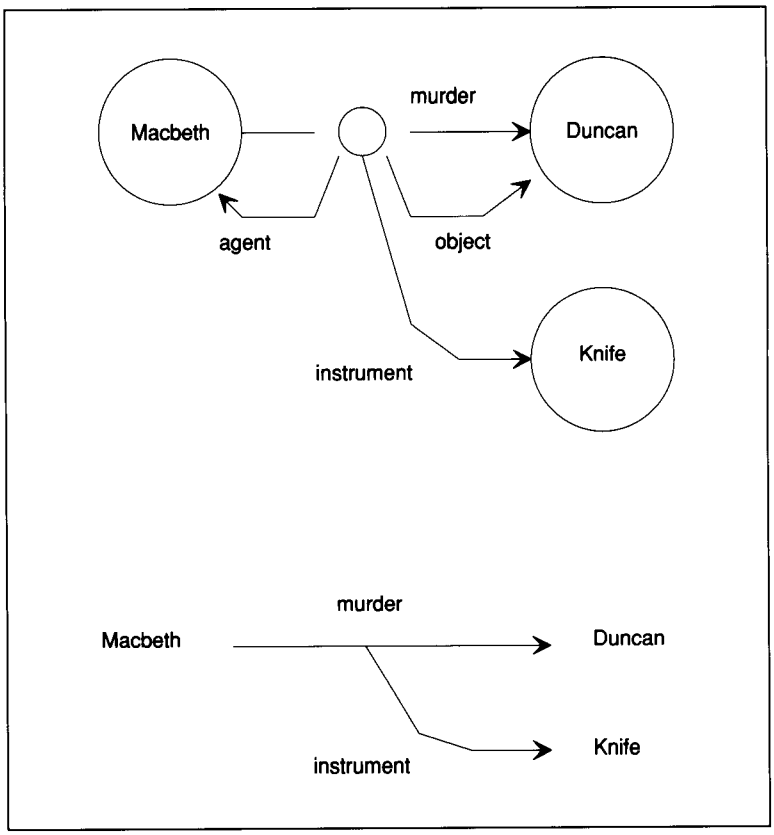

Figure 17.2 Two notations for a reified link describing Macbeth's murder of Duncan.

Explanation Transfer Solves Problems Using Analogy

How is it possible to know why an action has occurred or whether an action is likely to occur? How is it possible to know why a relation has come to be or whether it is likely to come to be? The answer to both questions is that Cause links in a precedent situation are likely to identify links that can serve as a basis for justification or prediction. Ultimately, of course, this answer rests on the assumption that, if two situations are similar in some respects, then they are likely to be similar in other respects as well.

To see how MACBETH makes use of precedents, consider the *Macbeth* precedent, shown in semantic-net form in figure 17.3, and the accompanying exercise, *Greed*, shown in figure 17.4.

MACBETH creates an **explanation template** consisting of the Cause links in the precedent and all the links that the Cause links tie together. Thus, the explanation template is a record of how links involving marriage, greed, and weakness have led to a want-to-be-king link in *Macbeth*.

To use the explanation template, MACBETH must determine how the people in the exercise correspond to the people in the precedent. In the

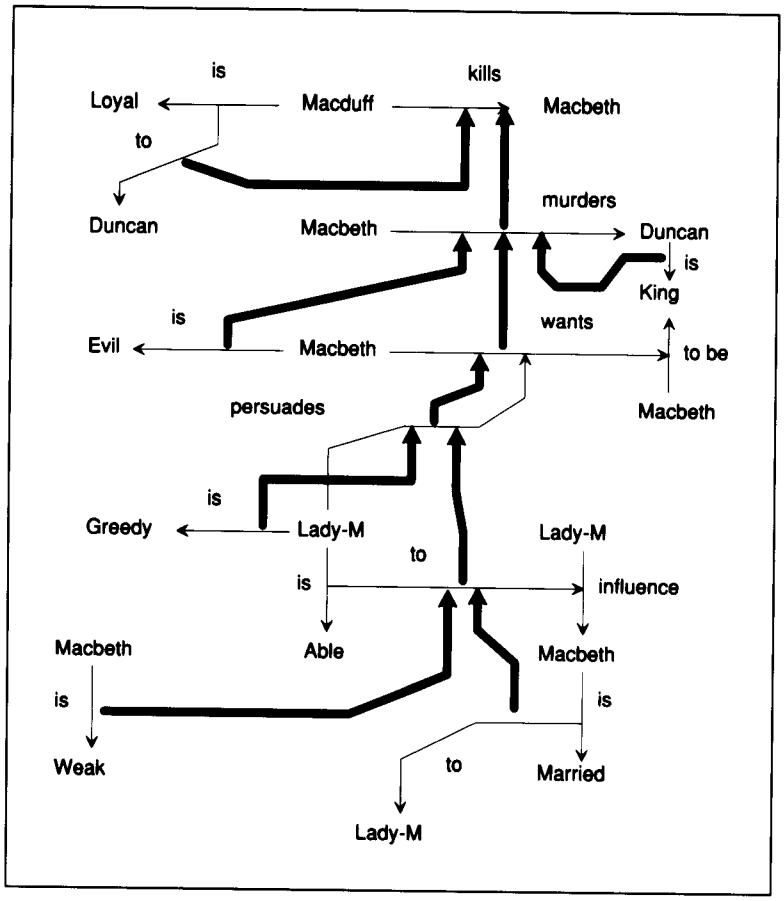

Figure 17.3 The *Macbeth* precedent with Cause links shown in bold.

exercise, the noble is weak, and his wife is greedy. In *Macbeth*, Macbeth is weak and his wife, Lady Macbeth, is greedy. Noting these characteristics, MACBETH concludes that the noble corresponds to Macbeth, and that his wife corresponds to Lady Macbeth.

Assuming that similarities among causes in two analogous situations lead to similarities among the consequents of those causes, MACBETH explains the want-to-be-king link in *Greed* by transferring the explanation template from *Macbeth* to *Greed*, as shown in figure 17.5. Once the explanation template taken from *Macbeth* is in place, a chain of overlaid Cause links connects known relations to the want-to-be-king relation, which is the one to be explained.

In this first example, there is only one precedent involved. When a single precedent cannot supply the total explanation template needed, MACBETH attempts to chain together several explanation templates. In the following exercise, for example, it is not known that the noble is weak, as is required for the *Macbeth* precedent to be completely successful:

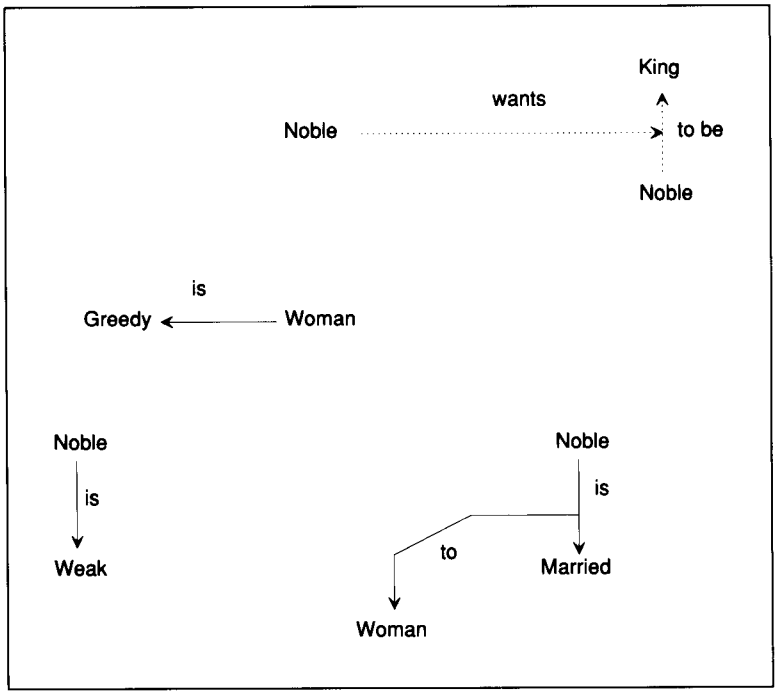

Figure 17.4 The *Greed* exercise, with dotted links identifying the relation to be explained.

Domination

This is an exercise about a noble and a domineering, greedy woman. The noble is married to the woman. Show that the noble is likely to want to be king.

A second precedent, the *Linda and Dick* story, can help MACBETH to establish that this new noble is also weak, thus working in support of the *Macbeth* precedent:

Linda and Dick

This is a story about Linda and Dick. Linda is a woman, and Dick is a man. Dick is married to Linda. Dick is weak because he is tired. He is tired because Linda is domineering.

Figure 17.6 shows what the *Linda and Dick* story looks like. Figure 17.7 shows how *Linda and Dick* works in support of *Macbeth*: *Macbeth* handles the top part; *Linda and Dick*, handles the bottom; and the two are joined at the link denoting the noble's weakness.

Thus, explanation templates taken from precedents serve to guide MACBETH to the possible causes of the link to be explained. Here, in summary, is the MACBETH procedure expressed in procedural English:

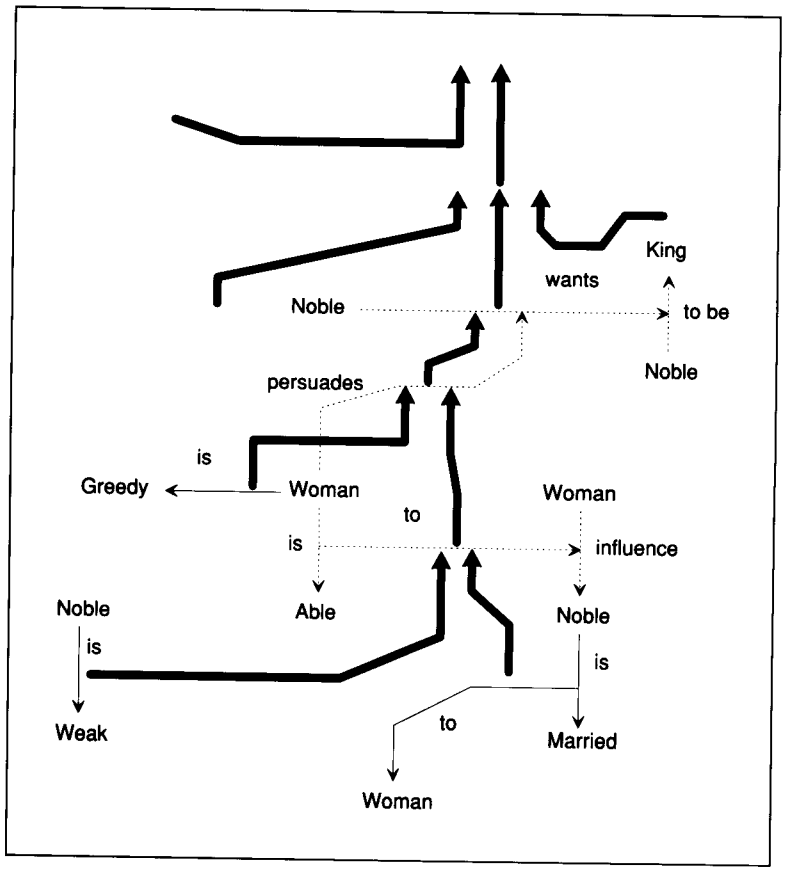

Figure 17.5 The exercise, with an explanation template, taken from *Macbeth*, shown with heavy lines.

To solve problems with precedents using MACBETH,

- ▷ Match the exercise to a precedent with a link corresponding to the relation or action to be explained.
- ▷ Transfer the explanation template from the precedent to the exercise.
- ▷ Trace through the transferred explanation template. Determine whether the link to be explained is supported by existing links.
  - ▷ If the desired conclusion is supported, announce success.
  - ▷ Otherwise, check whether MACBETH can justify needed, but missing, links, using other precedents. If it can, then announce success.
  - ▷ Otherwise, announce failure.

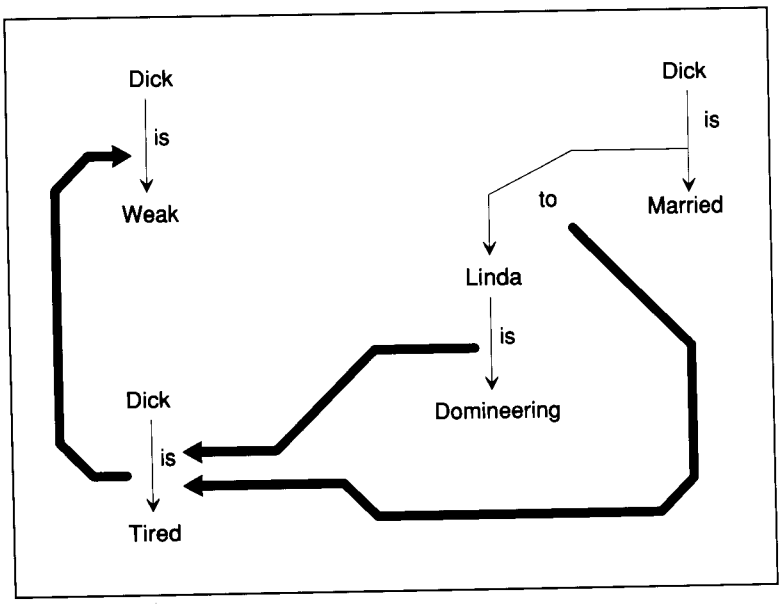

Figure 17.6 A precedent showing that a person's domineering nature can lead to a spouse's weakness.

Commonsense Problem Solving Can Generate Rulelike Principles

Whenever explanation templates are used to explain a relation or action, those explanation templates essentially supply an And tree that spans the gap between the relation or action to be explained and relations or actions that are already known. The root node of the And tree corresponds to the relation or action to be explained; the leaf nodes correspond to relations or actions that are already in place; the nodes in between correspond to the links joined by Cause relations in the explanation templates.

If the exercise is a good exercise, the And tree should be a good gap spanner in general, not just in the exercise. Accordingly, MACBETH uses the And tree to commit a new description to memory for future use:

To learn using MACBETH,

- ▷ Find the And tree implied by the explanation templates used to work the exercise.
- ▷ Using the climb-tree induction heuristic, generalize the agents and objects that appear in the And tree.
- ▷ Build a new description using the links that appear in the And tree and the generalized agents and objects.

Because the new description can involve bits of many explanation templates, it is said to be a **recollection**. The word *recollection* is intended to suggest that knowledge is collected in a new way—that is, knowledge is

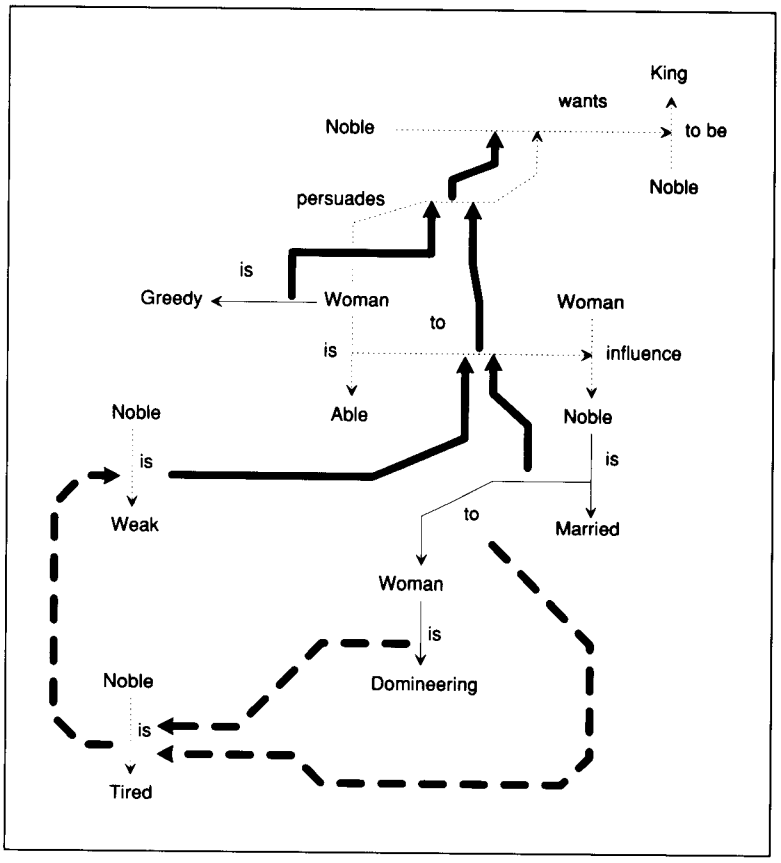

Figure 17.7 An example illustrating the use of two precedents working together. *Macbeth* establishes, via the heavy lines, that the noble is likely to want to be king, but only if it can be established that the noble is weak. *Linda and Dick* establishes, via the heavy dashed lines, that the noble is likely to be weak by pushing analysis further, ultimately connecting the want-to-be-king link to links that appear in the exercise.

re-collected. In the simple example involving just *Macbeth* and the *Greed* exercise, the recollection synthesized from the precedent and the exercise is illustrated in figure 17.8.

Labeling the root node of the recollection with *then*, and labeling the leaf nodes with *if*, you can see that recollections can be viewed as though they were antecedent–consequent rules of the sort used in rule-based deduction systems:

The Want-To-Be-King Rule
If ?noble is weak
?noble is married to ?woman
?woman is greedy
Then ?noble is likely to want to be king

Because recollections are actually represented in the same semantic-net representation as are precedents, they are treated as ordinary precedents by MACBETH. Thus, recollections can contribute to new recollections, just as precedents can.

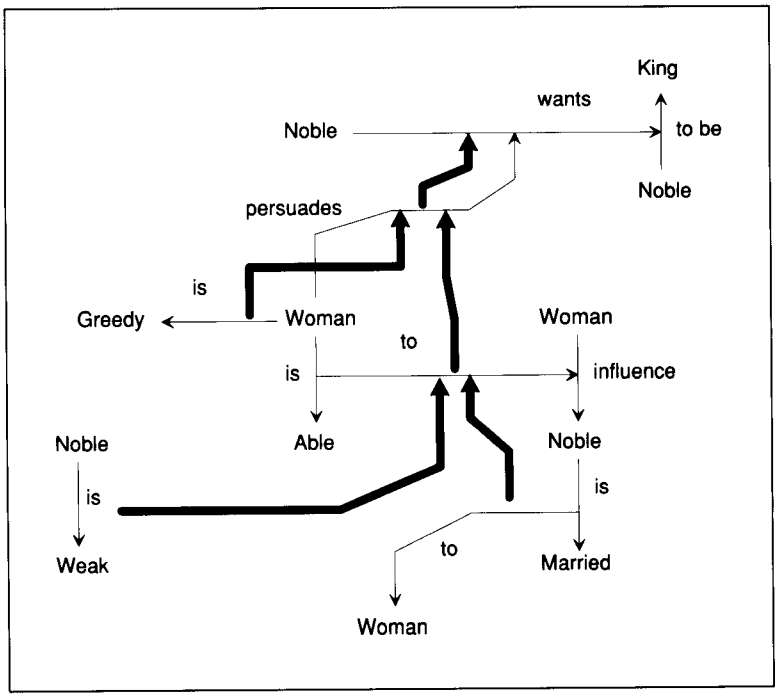

Figure 17.8 An exercise causes part of the explanation template of a precedent to be extracted, leading to a recollection. The recollection is in the same representation used for the precedent and the exercise. Heavy links are causal links.

The MACBETH Procedure Illustrates the Explanation Principle

From the problem-solving perspective, MACBETH's job is to explain how an action or relation can be justified or predicted on the basis of precedents. From the learning perspective, however, MACBETH's job is to build on its own explanations, assembling new recollections. This idea is, of course, a good one for people to follow as well:

The **explanation principle**:

▷ If you want to understand a concept, try explaining it to someone else.

The Macbeth Procedure Can Use Causal Chains to Establish Common Context

The MACBETH procedure, as described, happily thinks, by virtue of the *Linda and Dick* story, that a man married to a domineering woman will be weak. But what if the man does not live with the woman? What if the woman is only domineering with respect to her siblings? What if the man is even more domineering?

The usual answer is that precedents such as *Linda and Dick* should be used only in the right **context**, which raises the question of how you can tell whether the context is right. One way—a weak way—is to look for

points of general similarity between two situations. It is possible, however, for two situations to be similar in many respects without being similar in the right respects. Worse yet, the analogy-justifying contextual similarities between two situations may be implicit, and may not even be visible in their descriptions.

There is, nevertheless, a way to increase your confidence that a precedent is applicable to an exercise even if many of the Cause links in its explanation template are flimsy and depend on the precedent and the exercise having the same context.

The idea is to turn the problem into an asset. If two situations have different contexts, few matching links lead via Cause links to common consequences. Conversely, if many matching links lead via Cause links to common consequences, you can be increasingly confident that the situations are similar in important ways. If they are similar in important ways, then one is likely to be a good precedent for the other, as expressed in the following principle:

Winston's principle of parallel evolution:

▷ The longer two situations have been evolving in the same way, the more likely they are to continue to evolve in the same way.

To see how MACBETH can use this principle, consider the following exercise:

Domination and Fatigue

This is an exercise about a noble and a greedy woman. The noble is married to the woman. The noble is tired because the woman is domineering. Show that the noble is likely to want to be king.

As shown in figure 17.9, *Domination and Fatigue* is just like *Domination*, except that the noble is said to be tired, and he is tired because his wife is domineering.

When MACBETH tries to show that this noble is likely to want to be king, MACBETH must once again depend on the *Macbeth* precedent, which leads MACBETH to look for a reason why the noble is weak. Once again, MACBETH must use the *Linda and Dick* precedent. This time, however, *Domination and Fatigue* and *Linda and Dick* both have a Cause link that ties the woman's domineering nature to the man's fatigue. Thus, the explanation template taken from *Linda and Dick* not only spans a gap in *Domination and Fatigue*, but also matches a Cause link that lies just before the gap.

Evidently, the contexts of *Domination and Fatigue* and *Linda and Dick* are the same insofar as the contexts influence the matching Cause links. If

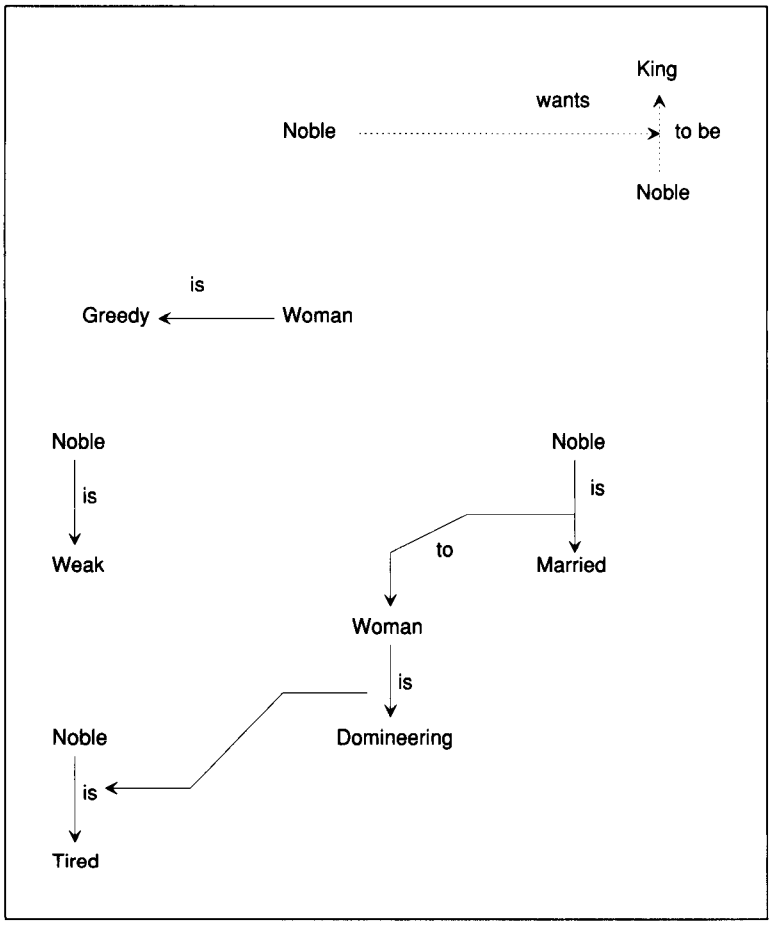

Figure 17.9 This exercise description, for *Domination and Fatigue*, is like the one for *Domination*, except that there are three extra links indicating that the noble is fatigued because the woman is domineering.

the context were radically different, you would expect no matching Cause links.

Of course, one matching pair of Cause links is not substantial evidence, but it is more evidence than none at all. Furthermore, if the number of parallel Cause links is larger, your expectation that the situations are likely to evolve in parallel increases in proportion.

Accordingly, MACBETH can make use of the following heuristic corollary of Winston's principle of parallel evolution: Given a set of useful-looking precedents, use the one with an explanation template that matches the largest number of Cause links in the exercise.

LEARNING ABOUT FORM AND FUNCTION

Interestingly, the MACBETH procedure also learns what objects look like from functional descriptions and descriptions of particular examples. In

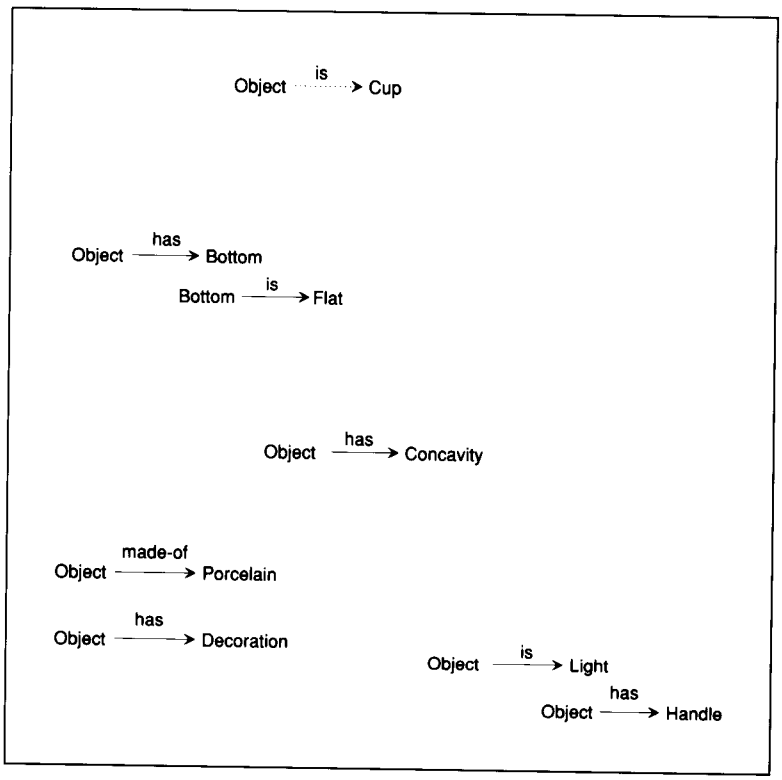

Figure 17.10 A particular object that is likely to be a cup.

this section, you see how MACBETH can learn what cups look like once it knows what cups are for.

Examples and Precedents Help Each Other

The first step in relating form to function is to describe the function. For MACBETH's benefit, a cup is defined by way of the following English sentences:

A Cup Description

This is a description of an object. The object is a cup because it is stable and because it enables drinking.

In contrast to the functional description, the description of a particular cup, such as the one described below and shown in figure 17.10, concentrates on physical qualities, not functional ones:

A Particular Object

This is an exercise about a light object that is made of porcelain. The object has a decoration, a concavity, and a handle. The object's bottom is flat. Show that the object is a cup.

Now it is time to demonstrate that the functional requirements are met by the physical description. The demonstration requires precedents that relate the cup's functional description to the description of the particular cup. Four precedents are needed. One explains how an object can be stable; another explains how drinking is related to carrying liquids and being liftable; another explains how being liftable is related to being of light weight and having a handle; and still another explains how carrying liquids is related to having a concavity. All contain at least one relation that is irrelevant with respect to dealing with cups; these irrelevant relations are representative of the detritus that can accompany the useful material:

A Brick

This is a description of a brick. The brick is stable because the brick's bottom is flat. The brick is heavy.

A Glass

This is a description of a glass. The glass enables drinking because the glass carries liquids and because the glass is liftable. The glass is pretty.

A Briefcase

This is a description of a briefcase. The briefcase is liftable because it is has a handle and because it is light. The briefcase is useful because it is a portable container for papers.

A Bowl

This is a description of a bowl. The bowl carries liquids because it has a concavity. The bowl contains cherry soup.

With the functional description in hand, together with relevant precedents, MACBETH is ready to go to work. First, MACBETH searches for precedents that are relevant to showing that the object in the exercise is a cup. The functional description is retrieved. Next, MACBETH determines the correspondence between parts of the exercise and the parts of the functional description—a straightforward task in this instance. Now, MACBETH overlays the explanation template of the functional description on the exercise. Tracing through the Cause links in the explanation template raises two questions: Is the observed object stable? Does it enable drinking?

Questioning whether the object is stable leads MACBETH to a second search for a precedent, and the brick precedent is retrieved. The object has a flat bottom, like a brick, so it is likely to be stable.

Next the glass precedent is retrieved to deal with drinking. MACBETH uses the glass precedent to translate the drinking question into a question of carrying liquids and liftability.

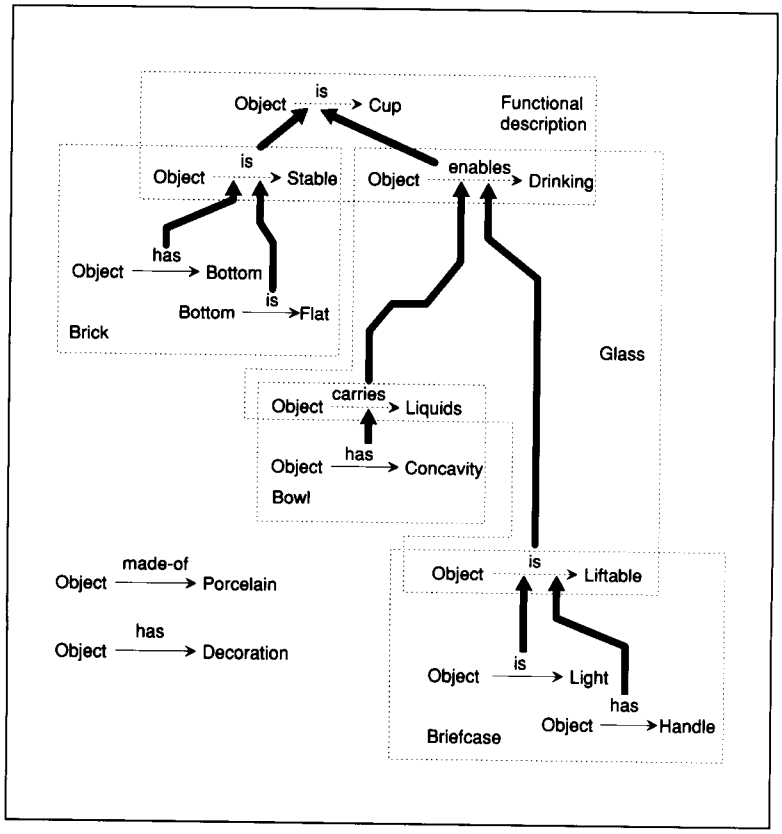

Figure 17.11 Evidence supporting the conclusion that a particular object is a cup is drawn from several precedents. Only a few of the links in the precedents supply supporting evidence; the rest are ignored.

To see whether the object can carry liquids, MACBETH uses the bowl precedent. MACBETH notes that a bowl carries liquids because it has a concavity, just as the example object does.

Next, MACBETH considers the liftability requirement, and retrieves the briefcase precedent. The briefcase precedent indicates that an object is liftable if it is light and has a handle. Because the object is light and has a handle, it must be liftable, completing the explanation of why it can function as a cup.

At this point, MACBETH has supporting evidence for the conclusion that the exercise object is a cup, all of which is summarized in figure 17.11.

Now it is time for MACBETH to build a recollection that deals with what cups look like. Because this new recollection is to be used for identification, rather than for justification or prediction, either the term *identification model* or the term *model* is more appropriate than *recollection.*

MACBETH builds a cup model as it builds justification and prediction explanations, by extracting an And tree from explanation templates. The resulting model, shown as an antecedent–consequent rule, is as follows:

The Cup Model
If ?object has a flat bottom
?object has a concavity
?object is light
?object has a handle
Then ?object is a cup

From this example, it is clear why learning form from function requires a physical example and some precedents, in addition to a functional description:

- The physical example is essential: otherwise, there would be no way to know which precedents are relevant.
- The precedents are essential: otherwise, there would be no way to know which aspects of the physical example are relevant.

Explanation-Based Learning Offers More than Speedup

At first glance, explanation-based learning appears to do no more than to take note of existing Cause links, repackaging them into more directly useful recollections, but not adding anything new. Thus, it would appear that explanation-based learning offers speedup only, for you could always go back to the original precedents in principle.

Looking more deeply, however, you can see that explanation-based learning offers more than mere speedup when the learner has imprecise or faulty knowledge of how individual precedents are allowed to form larger explanations. The reason is that particular exercises, often supplied by a knowledgeable teacher, provide heuristic evidence about which precedents can be stuck together usefully to determine what objects look like. Without that kind of heuristic evidence, a program can chain only through whatever precedents it has, indiscriminately. Such a program could reason that cows are cups on the grounds that they stand stably and enable drinking, thus satisfying the functional description provided for cups.

MATCHING

MACBETH cannot do its job unless it can identify how objects in an exercise correspond to the objects in a precedent. In the *Greed* example, for instance, MACBETH has to know that the noble corresponds to Macbeth and that the woman corresponds to Lady Macbeth. In this section, you learn how a matching procedure can establish that sort of correspondence with reasonable efficiency.

Stupid Matchers Are Slow and Easy to Fool

Consider *Macbeth*, as shown in figure 17.3, and *Greed*, as shown in figure 17.4. The desired correspondence of individuals is as follows:

Macbeth	Greed	Role
Macbeth	The noble	Victim
Lady Macbeth	The woman	Villain

Nothing explicitly identifies the villains and victims, however, so MACBETH must do matching by analyzing how all the individuals fit into the two situations.

The obvious approach, of course, is to try each possible way of pairing off the individuals, scoring each way according to the number of relations that are placed in congruence, and selecting the way with the highest score.

There are two problems, however, with the obvious approach. The first problem is that the number of ways of pairing off the individuals increases with the product of the number of individuals to be paired off. *Macbeth* has four: Macbeth, Lady Macbeth, Duncan, and Macduff; *Greed* has two: the noble and the woman; hence eight pairings would have to be scored, which is not too bad. If each situation has many individuals, however, *not bad* becomes *scary.*

A second, more serious problem, is that accidental, random correspondences can pull the matching process away from the correspondences that produce useful analogies. It might be, for example, that Macbeth in *Macbeth* and the woman in *Greed* share a large number of irrelevant, but mentioned, properties such as hair color and appetite size.

The matcher you learn about here, in contrast, is both more reliable and faster:

- It is driven by the purpose of the match, so it is immune to irrelevant properties.
- It only matches individuals that are relevant to the purpose of the match.

Matching Inexact Situations Reduces to Backward Chaining

The matcher works by using ideas you first learned about in connection with backward chaining through rules. Accordingly, you might find it useful to review Chapter 7 if the forthcoming discussion seems overly telegraphic.

Two key ideas enable the reduction of situation matching to backward chaining:

- You treat the precedent as though it were a source of *antecedent–consequent* rules.
- You use the exercise as though it were a database of assertions.

To see how these ideas help you, first consider the use of *Macbeth* as a source of *antecedent–consequent* rules. On examining figure 17.3, you see that there are five links that are explained by one or more Cause links that tie the first five links to other links. Each of the five explained links is readily transformed into an *antecedent–consequent* rule, with the explained link

constituting the consequent, the other links constituting the antecedents, and each individual viewed as a match variable:

R1 If *?x1* is loyal to *?x2*
?x3 murders *?x2*
Then *?x1* kills *?x3*

R2 If *?x3* is evil
?x3 wants to be king
?x2 is king
Then *?x3* murders *?x2*

R3 If *?x4* persuades *?x3* to want to be king
Then *?x3* wants to be king

R4 If *?x4* is greedy
?x4 is able to influence *?x3*
Then *?x4* persuades *?x3* to want to be king

R5 If *?x3* is weak
?x3 is married to *?x4*
Then *?x4* is able to influence *?x3*

The process of converting individual names into match variables is called **variablization**. The following table records how the variablization has been done:

Individual	Variables
Macduff	*?x1*
Duncan	*?x2*
Macbeth	*?x3*
Lady Macbeth	*?x4*

Next, consider the use of *Greed* as a source of assertions. On examining figure 17.5, you see that there are the following assertions only:

The woman is greedy
The noble is weak
The noble is married to the woman

With these assertions, and the five rules, you can now ask whether *Macbeth* supports the assertion, *the noble wants to be king.*

Of the five rules, only rule R3's consequent matches *the noble wants to be king*, creating the following variable bindings:

Variable	Binding
?x3	the noble

Having established the initial variable bindings, backward chaining then attempts to find assertions corresponding to the antecedent of R3 with the variable binding substituted for the variable—namely, *?x4 persuades the noble to want to be king.* Unfortunately, an assertion corresponding to the antecedent of R3 is not found. Accordingly, a rule establishing who persuades the noble to be king must be used. Rule R4 has the right kind of consequent, because that consequent matches the antecedent of R3 exactly.

Now it is time to consider rule R4's antecedents. The first of these is *?x4 is greedy*, which matches the assertion *the woman is greedy*, augmenting the variable bindings:

Variable	Binding
?x4	the woman
?x3	the noble

Alas, there is no assertion corresponding to the second antecedent in R4, with variables replaced by variable bindings, *the woman is able to influence the noble.* Accordingly, further chaining through another rule is necessary. There is, fortunately, a rule that does the job—namely, R5, because the consequent of R5 matches the antecedent of R4 exactly.

Turning to R5's antecedents, assertions are found for both, once variable bindings are substituted for variables, for *Greed* supplies both *the noble is weak* and *the noble is married to the woman.*

At this point, *Macbeth* has found support for the assertion that the noble wants to be king. Also, the pairing of individuals that enables problem solving is obvious, and is captured in the following table of variables, variable bindings, and variable origins:

Variable	Binding in Greed	Origin in Macbeth
?x3	the noble	Macbeth
?x4	the woman	Lady Macbeth

Thus, Macbeth is paired with the noble and Lady Macbeth is paired with the woman, as expected.

Matching Sheds Light on Analogical Problem Solving

From the way matching is done in conjunction with analogical problem solving, you can see that each precedent is, in effect, a repository of implicit antecedent–consequent rules, along with many useless details with respect to any particular purpose. The job of MACBETH, then, is to find and exploit such implicit rules of inference.

SUMMARY

- One way to learn is to piece together explanations from causal chains transferred from precedents to deal with a new problem.
- In general, to reify is to treat something abstract as though it were concrete. To reify a link is treat it as a node so that you can describe it. Reification and the vocabulary of thematic-role frames make it possible to record story plots for use as precedents.
- The MACBETH procedure uses causal chains found in story plots and object descriptions to solve problems and to generate rulelike principles.
- The MACBETH procedure takes overlapping causal chains as evidence of common context.
- The MACBETH procedure uses a matcher that reduces matching to a form of backward chaining. Thus, matching and problem solving are interdigitated.
- Explanation-based learning is particularly beneficial in that teachers can guide students toward appropriate precedent combinations. In the absence of such guidance, a weak student could put together the wrong things, making silly mistakes. Thus, explanation-based learning offers more than speedup.

BACKGROUND

The discussion of MACBETH is based on work by Patrick H. Winston [1980, 1982]. The discussion of MACBETH in the context of form and function is based on work by Winston and his associates [Winston 1980, 1982; Winston, Thomas O. Binford, Boris Katz, and Michael R. Lowry 1983].

The theory was shaped to a great degree by experiments that would have been extraordinarily tedious to perform without the English interface, developed by Boris Katz [1982], by means of which the experimental database was prepared, revised, and revised again. As artificial intelligence progresses beyond toy-world domains, it becomes obvious that databases prepared and accessed using English are necessary for research, not simply for research presentation.