MOLE: A Knowledge Acquisition Tool That Uses Its Head

Larry Eshelman and John McDermott
Department of Computer Science
Carnegie-Mellon University
Pittsburgh, PA 15213

Abstract

MOLE can help domain experts build a heuristic classification problem-solver by working with them to generate an initial knowledge base and then detect and remedy deficiencies in it. By exploiting several heuristic assumptions about the world, MOLE is able to minimize the information it needs to elicit from the domain expert. In particular, by using static techniques of analysis, MOLE is able to infer support values and fill in gaps when a knowledge base is under-specified. And by using dynamic techniques of analysis, MOLE is able to interactively refine the knowledge base.

1. Introduction

MOLE assists domain experts in building expert systems that do heuristic classification [Clancey 84, Clancey 85, Buchanan 84]. MOLE is useful in domains in which the expert can pre-enumerate a set of candidate hypotheses (e.g., faults, diseases, components) and in which hypotheses can be evaluated on the basis of weighted evidential considerations (e.g., symptoms, requirements). MOLE is the successor to MORE [Kahn 85a, Kahn 85b] and, more generally, follows in the footsteps of systems like TEIRESIAS [Davis 82] and ETS [Boose 84]. Like these other knowledge acquisition tools, MOLE elicits knowledge from the domain expert and builds a knowledge base. The knowledge base can then be interpreted by an inference engine to perform some heuristic classification task. In all such knowledge acquisition tools the inference engines make certain assumptions about the nature of the world. MOLE differs from these other systems in that its heuristic assumptions are made explicit and are exploited in the knowledge acquisition process. We are trying to make MOLE smart -- which in this case means asking as few questions of the expert as possible while still being able to build a reasonable knowledge base for performing a task. MOLE's approach to knowledge acquisition is to use its heuristic assumptions about the world and assumptions about how domain experts express themselves to disambiguate the knowledge elicited from the expert.

In Section 2 we describe MOLE's inference engine and how it depends upon MOLE's heuristic assumptions about the world. Unlike most other knowledge acquisition tools, MOLE is both a knowledge acquisition system and a performance system. The knowledge base built by MOLE's knowledge acquisition tool is interpreted by MOLE's inference engine to perform the given task. In Section 3 we show how MOLE's heuristic assumptions guide its knowledge acquisition process. This section is divided into two subsections which reflect the two modes of analysis used by MOLE when guiding the knowledge acquisition process: static and dynamic. Static analysis looks at the structure of the dormant knowledge base. Dynamic analysis focuses on certain parts of the knowledge base in the context of feedback provided by the expert during test diagnoses.

2. The Inference Engine

MOLE's power as a knowledge acquisition tool comes from its understanding of its problem-solving method. In MOLE's case this means selecting or classifying hypotheses on the basis of evidential considerations. To the extent that a problem-solving method makes weak presuppositions about the world, the method may give only the most limited leverage to a knowledge acquisition tool. MYCIN, for example, makes very weak presuppositions; it views its rules essentially as arbitrary implications among arbitrary facts about the world [Szolovits 78]. Other classification systems such as INTERNIST [Miller 82, Pople 82] and CASNET [Weiss 78] provide a much more specific interpretation -- a causal interpretation -- of the network of rules or links connecting its "facts". MOLE in more like INTERNIST and CASNET in this respect.

MOLE's current strength is principally in the area of assisting in the development of diagnostic systems (as opposed to other types of classification systems). For MOLE a hypothesis is the cause or explanation of the problem being diagnosed. There are three types of associations supporting hypotheses:

1. symptoms
2. prior-conditions
3. qualifying conditions

A symptom is any event or state that is a causal manifestation of a hypothesis. A prior-condition is any event or state that occurs prior to or simultaneous with the hypothesis and makes the hypothesis more or less likely to be true. A qualifying condition is any background or distinguishing condition that qualifies the support of a symptom or prior-condition for a hypothesis.

We will illustrate these various types of associations with an example from a knowledge base that allows MOLE to diagnose steel rolling mill problems. One problem that can arise in a rolling mill is that the sheet of steel being rolled is too narrow coming out of the mill. This symptom has three potential causes: (1) a roll is worn out; (2) there is excessive tension between the various rolls; (3) the sheet of steel was too narrow going into the rolling mill. These are the hypotheses which could explain the symptom. The hypothesis that the roll is worn out has several other symptoms -- for example, an oscillating looper roll. In addition, the worn out roll hypothesis has several prior-conditions which might affect the likelihood that it is worn out -- for example, its installation date. Note that the symptoms of the hypothesis, unlike the prior-conditions, are explained by the hypothesis. The association between a hypothesis and a symptom or prior-condition may need to be qualified; for example, if the looper roll fails to oscillate, this tends to rule out the hypothesis that the roll is worn out unless the steel being rolled is a soft alloy.

MOLE's predecessor, MORE, evaluated candidate hypotheses by combining support values and comparing the resulting value to a threshold. Hypotheses whose combined support was above the accept threshold were accepted, and hypotheses whose combined support was below the reject threshold were rejected. Any hypothesis whose combined support was in between the reject and accept thresholds was classified as indeterminate. However, indeterminate candidates were rejected if they were not needed to explain any symptoms. This latter criterion for rejecting candidates meant that MORE had some rudimentary capability to reason about evidence. But for the most part, MORE's performance was dependent upon the expert assigning reasonable numeric support values to its evidential associations. This meant that the adequacy of the knowledge acquisition process depended upon the expert's ability to assign reliable support values. Since experts have trouble assigning these support values and often do so in a rather ad hoc fashion, this became the weakest link in MORE's knowledge acquisition process. Although experts could use MORE to build diagnostic knowledge bases, MORE was little more than a knowledge acceptor.

With MOLE, on the other hand, less emphasis has been placed on the numeric support values and more on reasoning about evidence. The user no longer has to supply support values. MOLE is able to assign reasonable support values because of certain heuristic assumptions it makes about the world. These assumptions also facilitate MOLE's ability to reason about evidence which, in turn, enables MOLE to be less reliant on its support values. We discuss how support values are determined in the next section. In the remainder of this section we discuss MOLE's heuristic assumptions and how they affect its ability to reason about evidence.

MOLE's heuristic assumptions about the world are similar to those made by INTERNIST. MOLE makes two basic assumptions about the world:

1. Exhaustivity: every abnormal finding has an explanation -- i.e, some candidate hypothesis will account for it.
2. Exclusivity: explanations should not be multiplied beyond necessity -- i.e., do not accept two hypotheses if one will do.

The exhaustivity heuristic enables MOLE to interpret the evidential links in its domain model causally. Every symptom is assumed to have a cause. If a symptom is not explained by one hypothesis, it must be explained by another. The exclusivity heuristic is based on Occam's razor. All other things being equal, parsimonious explanations should be favored. In addition, it captures the assumption that the types of events represented by hypotheses are fairly rare, so it is unlikely that several occur simultaneously. (Of course, two such events might be interrelated, but then this should be represented in the network.)

An important corollary follows from the exhaustivity and exclusivity heuristics: Accept the best candidate relative to its competitors -- i.e, a candidate may "win" by ruling out competing candidates. Because symptoms must be explained by some hypothesis (exhaustivity), one of the hypotheses must be true. And because only one hypothesis is likely to be true (exclusivity), we can drive up the support of one hypothesis by driving down the support of its competitors or vice versa.

For instance, the fact that the looper roll is not oscillating tends to rule out the hypothesis that a worn out roll is the cause of the steel being too narrow on exit. If we have already ruled out that the steel was too narrow on entry, then we are led to conclude that the only remaining hypothesis must be the cause -- i.e., there is excessive tension between the rolls. However, if we find that there is greater evidence against this hypothesis than the other two, the other two again become contenders. Even though there is evidence that would normally rule them out, they are still better than the only other alternative.

In order to show the important role MOLE's heuristic assumptions play in the evaluation process, we will briefly summarize its method of evaluation. The evaluation process begins by asking the user about a set of core symptoms. Depending upon the starting point within a given network of hypotheses and evidential associations, the inference engine can do either backward or forward chaining. The evaluation method consists of the following steps:

1. Ask about the **core** symptoms.
2. **Activate** those hypotheses that are needed to explain the the symptoms that are known to be present.
3. **Differentiate** active hypotheses
    - **Rule out:** Raise support for one hypothesis by lowering support for competing hypotheses by establishing that negative prior-conditions are satisfied.
    - **Raise prior probability:** Raise support for one hypothesis relative to its competitors by establishing that positive prior-conditions are satisfied.
    - **Symptom differentiation:** Establish that there are symptoms which support one hypothesis more than its competitors; go to 2.
4. **Combine the support** provided by the evidence for each hypothesis using the Bernoulli combination.
5. **Accept** those hypotheses whose evaluation is above some threshold.
    - Accept those hypotheses which explain a single symptom better than any of their competitors.
    - Accept those hypotheses whose combined support from symptoms is greater than any of their competitors.
6. If there are some symptoms which are not explained by an accepted hypothesis and there are potential queries which might be relevant, go to 3.
7. **Reject** those hypotheses that are not needed to explain the known symptoms.
    - Reject those hypotheses that are not accepted and which are not needed to explain known symptoms.
    - Reject those hypotheses that are accepted because they explain a particular symptom, provided this symptom is very likely to follow from a hypothesis that is needed to explain other symptoms.

MOLE's heuristic assumptions are the basis for steps 3 and 7 -- the differentiation and rejection steps. The exhaustivity heuristic implies that a hypothesis can be rejected only if it is not needed to explain any of the symptoms. The exclusivity heuristic also is relevant for determining when to reject a hypothesis. A tentatively accepted hypothesis H_1 is rejected if some other independently accepted hypothesis H_2 will explain those symptoms S_i which H_1 explains, and the S_i are more likely to follow from H_2 than H_1. The corollary which follows from the two heuristics is the basis for the differentiation process by which MOLE distinguishes the relative merits of the active hypotheses.

To return to our rolling mill example, if MOLE knows that the steel is too narrow on exit, three hypotheses are activated: (1) the roll is worn out, (2) there is excessive tension between the rolls, and (3) the steel was too narrow on entry. In order to determine which of these three hypotheses is the cause of the steel being too narrow on exit, MOLE looks for symptoms which only one of the hypotheses will explain and for circumstances which will rule out the other hypotheses. In this example, there are three symptoms which are explained by a worn out roll that are not explained by either of the other hypotheses. If one of them holds -- e.g., the looper roll is oscillating -- then MOLE concludes that since the worn out roll explains all known symtoms, it is the cause of the steel being too narrow on exit. The other two hypotheses are not needed and so are rejected. However, if there were another symptom which only the excessive tension hypothesis explained, MOLE would accept this hypothesis as well as the worn out roll hypothesis. Suppose, on the other hand, that none of the three symtoms which are only explained by the worn out roll hypothesis were present, but that some circumstance held which tended to rule out the worn out roll hypothesis -- e.g., there are no uneven surface problems. In this case MOLE would conclude that it is unlikely that there is a worn out roll, and would focus on the other two hypotheses. If MOLE could also rule out that the steel was too narrow on exit, then, by elimination, MOLE would conclude that the cause must be excessive tension between rolls.

MOLE's method of evaluation can be usefully compared to that of INTERNIST and CASNET. Like both these systems, MOLE attempts to select the hypothesis which accounts for the most data. And like both these systems more than one may be selected. There may not be a single hypothesis which covers all symptoms, so several hypotheses may need to be accepted. Although it is assumed that only a single hypothesis is needed to explain a particular symptom, another hypothesis may better explain some other symptom. MOLE's way of selecting the best hypothesis is similar to INTERNIST's. Like INTERNIST, MOLE picks the best hypothesis relative to its competitors instead of accepting a hypothesis only if its absolute score (numeric measure of belief) is above some fixed threshold. This method of selecting a hypothesis is a natural consequence of MOLE's heuristic assumptions about the world which, as we have noted, are similar to those made by INTERNIST. However, MOLE handles differentiation somewhat differently in that support is dynamically shifted from one hypothesis to another. When one hypothesis is ruled out, the support values for other hypotheses explaining the same symptom increase. MOLE sides with INTERNIST, and against CASNET, on one other important issue: observations and intermediate states are lumped together as manifestations of hypotheses. For MOLE this means that confidence in observations can be easily integrated into the differentiation process. If its confidence in a symptom is less than certain, MOLE treats the possibility that the observation might be mistaken as another hypothesis explaining the symptom. As evidence against the hypothesis explaining the symptom mounts, the likelihood that the observation is mistaken increases. Finally, MOLE has one very important property in common with CASNET: MOLE can reason both backwards and forwards within its network just as CASNET can in its network of pathophysiological states.

At the heart of MOLE's evaluation process is the distinction between evidence that needs to be explained or covered by some hypothesis and evidence that is circumstantial -- which is merely correlated with some hypothesis. By allowing MOLE's inference engine to be driven by "covering" evidence as opposed to circumstantial evidence, the emphasis is shifted from numeric support values to how well the covering evidence is explained. Only those hypotheses which are potentially needed to explain the symptoms are activated. Circumstantial evidence is used to differentiate the active candidates relative to some piece of evidence that must be covered. A hypothesis is accepted if it covers a piece of evidence better than its competitors.

In so far as the underlying heuristic assumptions can be given a suitable interpretation, MOLE's method can be applied to non-diagnostic domains. The exhaustivity heuristic simply says that there is information associated with some of the hypotheses which, when it holds, must be covered by one of these hypotheses. If the domain involves component selection, for example, then the hypotheses would be components and the relevant information might be requirements that must be met. Exhaustivity is then interpreted to mean that given some requirement, some member from a set of components which meets this requirement must be selected. The exclusivity assumption is interpreted to mean that only one component should be selected from this set. If no single hypothesis will cover all the requirements, then there must either be a missing hypothesis (component) that would cover all the requirements, or some of the requirements must be relaxed. It should be noted that the relaxing of requirements in a selection task parallels lowering the confidence in some of the symptoms in a diagnostic task.

We are not claiming that a suitable interpretation of these heuristics can be found for all heuristic classification tasks. Some classification tasks seem to be based primarily on circumstantial knowledge, with little or no role for covering knowledge. An example would be Grundy which recommends books based on the reader's personality [Rich 79]; the relevant knowledge is correlations between book traits and personality traits. No doubt there are many other classification tasks which provide little, if any, role for covering knowledge.

Early heuristic classification systems did not distinguish between covering and circumstantial knowledge [Shortliffe 76, Weiss 79]. In effect, they treated all evidential knowledge as circumstantial. This does not mean that their performance is inferior to MOLE's. If the expert can provide correct support values, they should perform as well as MOLE. The main advantage of MOLE lies elsewhere. As will be shown in the next two sections, by distinguishing covering knowledge from other types of associations, MOLE can provide more guidance to the knowledge acquisition process than would otherwise be possible.

3. Knowledge Acquisition

MOLE's knowledge acquisition process consists of two phases: (1) the gathering of information for constructing the initial knowledge base and (2) the iterative refinement of this knowledge base. In order to generate the initial knowledge base, MOLE asks the expert to list hypotheses and evidence that are commonly relevant in the expert's domain and to draw associations between the evidence and the hypotheses. The expert is encouraged to be as specific as possible. However, the expert is not required to specify anything more than the names of "events" and to indicate which events are associated. The resulting knowledge base can be viewed as an under-specified network of nodes and links. For the network to be fully specified three additional kinds of information are needed: (1) the type of each node, (2) the type of each link, and (3) each link's support value. A node's type indicates whether the method for determining its value is by directly asking the user or by infering its value from other nodes. A link's type indicates the type of evidential association it represents -- a covering association, a circumstantial association, or an association which qualifies the support of a covering or circumstantial association. The support value indicates how much positive or negative support a piece of evidence provides for a hypothesis.

MOLE understands that experts cannot always provide such information. This is a major difference between MOLE and its predecessor, MORE. MORE required the expert to specify the information in a form that reflected the knowledge structure presupposed by its knowledge base interpreter. The burden was on the expert to fit his knowledge into MORE rather than MORE being intelligent enough to make sense of whatever information the expert was willing to provide. MOLE, on the other hand, recognizes that experts often have difficulty coming up with a consistent set of support values, that they sometimes are uncertain about the type of evidential link, and that they occassionally are even unsure whether an event is observed or inferred. MOLE can tolerate such indeterminateness. MOLE is opportunistic and relies on its heuristics to mold the under-specified information provided by the expert into a consistent and unambiguous network and to discover missing or incorrect knowledge. Our research effort has been directed toward making MOLE smarter and less tedious to use. MOLE now asks less and infers more.

During the second phase of knowledge acquisition, MOLE and the expert interact in order to refine the knowledge base. The nature of this interaction is another major difference between MOLE and MORE. MORE used static analysis to try to discover weaknesses in the knowledge base. MORE had certain expectations about the structure of diagnostic networks, and prompted the user when the network did not meet these expectations. MOLE also uses static analysis, but it plays less of a role in discovering weaknesses in the knowledge base and more of role in disambiguating an under-specified network. Of MORE's eight strategies for improving diagnostic performance, only differentiation plays an important role during static analysis. Most of the burden of refining the knowledge base has been shifted to dynamic analysis. The expert supplies MOLE with feedback on how accurate its diagnosis is for some test case. If the diagnosis is incorrect, MOLE tries to determine the likely cause of the mistake and recommends possible remedies. The following two subsections discuss how static and dynamic analysis aid in the knowledge acquisition process.

3.1. Static Analysis

Static analysis concentrates on the structure of the dormant knowledge base. MOLE uses static analysis (1) to disambiguate an under-specified nework, (2) to assign support values, and (3) to recognize structural inadequacies in the network.

The expert may specify the initial knowledge base at any of several levels of abstraction. If the expert is not able to say whether an association is a covering or a circumstantial link, for example, he can specify the temporal relation of the association. This will create some ambiguity for MOLE. For instance, event E_1 could be prior to event E_2 because either E_1 is a prior-condition for hypothesis E_2 or E_1 is a hypothesis explaining symptom E_2. If the expert is unable to specify the temporal direction of a link, then he can minimally specify that two events are associated with no indication of the type of association or the temporal direction.

In this case, there is even more ambiguity in the network.

Because the network can be layered, with some hypotheses serving as symptoms for other hypotheses, there are often many possible interpretations of an under-specified network. MOLE currently has a number heuristics for helping it interpret such a network. Some of these heuristics rely on the nature of the types of associations understood by MOLE's evaluation method. Others make assumptions about how an expert's style of specifying the network should be interpreted. The following is an example of a heuristic based on the nature of associations.

If event E_2 leads to event E_1 and
event E_1 (when false) rules out event E_2
then E_1 is a symptom for E_2 rather than a prior-condition

MOLE assumes that although symptoms may provide negative as well as positive support, prior-conditions tend to be either positive or negative but not both. The following is an example of a heuristic based on how experts express themselves:

If event E_2 is inferred to be a symptom of event E_1 and
event E_3 is input as a sibling of event E_1
then E_3 is inferred to be a symptom of E_1

If the specification of the network is so under-determined that MOLE is not able to make any reasonable guesses about its shape, then MOLE asks the expert for additional information. Of course, even here MOLE does not simply ask for undirected guidance. MOLE ask for information which it expects will be the most effective in helping it determine the structure of the network. For example, asking about the role of an association with many siblings usually provides more information than asking about the role of an association with only a few siblings.

So far, nothing has been said about qualifying conditions. This is because MOLE initially assumes that each piece of information is either a symptom or a prior-condition and not some background qualifying condition. Symptoms and prior-conditions are assumed to provide independent evidence for hypotheses. This is a default assumption which expresses a lack of knowledge on MOLE's part. Once MOLE gets some feedback about the network's performance, MOLE can adjust this assumption during dynamic analysis if it needs to. This is done by adding conditions that qualify the support of a symptom or prior-condition for the hypothesis. Although qualifying conditions are typically extraneous background conditions, the interdependence of two symptoms can be represented by treating them as qualifying conditions for each other.

The rolling mill example illustrates some of these heuristics for disambiguating an under-specified network. MOLE was told that a worn out roll *leads to* a number of events such as the steel being too narrow on exit and the looper roll oscillating. Because these events were leaf nodes that follow from a worn out roll, MOLE assumed that the worn out roll was a hypothesis explaining these leaf nodes. For the same reason it concluded that excessive tension between rolls was a hypothesis. The excessive tension hypothesis, in turn, can be explained by one of two second level hypotheses -- i.e., either there is an overload or the looper is not working. MOLE assumed these were hypotheses because it was told that they *lead to* excessive tension and that there were other events that *lead to* them. On the other hand, MOLE was told that the roll being installed before a certain date was *linked* to the worn out roll. Because this association was less specific than the other types of specifications, MOLE assumed that it was probably a different type of an association -- i.e., a prior-condition. One of the events that MOLE was told leads to the looper not working is that there is a regulator malfunction. MOLE was uncertain whether this was a third level hypothesis explaining, or a positive prior-condition affecting the likelihood of, the looper not working. When it learned that a regulator malfunction leads to the looper meter resting on zero and that this is a leaf node, it concluded that the regulator malfunction must be a third level hypothesis.

Static analysis is also used to assign default support values. The method for assigning support values for covering evidence follows directly from MOLE's heuristic assumptions about the world. The exhaustivity heuristic, which assumes that every symptom can be explained by some hypothesis, in conjunction with the rule out corollary, which assumes that best is relative, insures that the positive support provided by a piece of evidence must be distributed among the hypotheses. And these two assumptions, along with the exclusivity heuristic, insure that the positive support from some piece of evidence to various candidates must sum to 1.0. MOLE makes the default assumption that the support values for any symptom are equally divided among the hypotheses that explain it.

The method for assigning support values for circumstantial evidence relies on a heuristic concerning how experts express themselves. MOLE assumes that experts initially mention a positive or negative prior-condition only if it has a significant impact; thus a fairly high support value is assigned in all cases. These values, like the support values for covering knowledge, can subsequently be changed by MOLE during dynamic analysis.

So far we have focused on semantic inadequacies of the initial network. Another source of problems is structural inadequacies. The expert typically forgets to add certain basic associations. Sometimes the resulting structure makes little sense from a diagnostic point of view. MOLE is able to recognize certain structural inadequacies and prompt the expert for likely remedies. For example, there may be no way to differentiate two hypotheses on the basis of the evidential associations provided by the expert. The expert may have forgotten to specify that there is some positive piece of evidence which supports one hypothesis but not the other or that when a positive piece of evidence which supports both hypotheses fails to hold, it tends to rule out one of the hypotheses. In the case of the rolling mill, the expert indicated that both excessive top speed and a wrong speed set up could explain an overload. MOLE reasoned that there is no point in specifying alternative explanations of an event unless these explanations can somehow be differentiated. MOLE asked the expert if there was any event that followed from one of the hypotheses and not the other. In this case, there was one such event for excessive top speed and two for the wrong speed set up.

Although static analysis plays an important role in locating structural inadequacies, its greatest value is in disambiguating and completing an under-specified network. Because MOLE does not need to elicit a lot of information from the expert in order to build a reasonable knowledge base, the expert is able to use MOLE to quickly generate a proto-type that performs the diagnostic task. The expert can then experiment with this proto-type and use MOLE's dynamic analysis capabilities to interatively refine the knowledge base.

3.2. Dynamic Analysis

Dynamic analysis is done in conjunction with test diagnoses. The expert gives MOLE a test case and tells MOLE the correct diagnosis. If MOLE the performance program comes to an incorrect conclusion, MOLE the knowledge acquisition tool tries to determine the source of the error and recommends possible remedies.

MOLE's predecessor, MORE, only did static analysis of its knowledge base. MORE analyzed the network staticly and suggested what types of knowledge might be missing. For instance, if MORE discovered that a hypothesis had no symptoms providing strong positive support, it would ask whether there were any features of the symptom which, when true, increased the support for the hypothesis. The problem is that there are potentially too many places where knowledge may be missing. In the rolling mill example, MORE discovered eighteen cases where distinguishing features might be needed, but only in one case could the expert provide any such features. This may be because the expert cannot think of the missing knowledge or because there is none. In either case, with the static approach, analysis of the network for missing knowledge was often cumbersome and not very helpful.

As was indicated in the previous subsection, MOLE does use static analysis. However, MOLE limits it to a few special cases. Generally, what is needed is some way to focus the analysis on the relevant parts of the network. MOLE uses feedback from diagnostic sessions to help it focus its attention on parts of the network with missing or incorrect knowledge. After MOLE has provided its diagnosis for some test case, the expert has the option of telling MOLE what he thinks is the correct diagnosis. This enables MOLE to focus on the part of the network where there is likely to be missing knowledge and to do so in a context in which the expert is more likely to notice that some knowledge is missing. If, for example, MOLE cannot distinguish between the hypotheses that would explain the looper not working, but the expert has told it that it should be able to, then it will occur to MOLE that it may be missing some distinguishing condition. In other words, MOLE does not ask for a specific type of knowledge until it makes an incorrect diagnosis where that type of knowledge could make a difference.

MOLE uses dynamic analysis to help (1) discover missing knowledge, (2) guide in the revision of support values, and (3) further disambiguate the network. The conditions for these actions are closely intertwined.

Given MOLE's diagnosis and a target diagnosis supplied by the expert, MOLE first determines whether the targeted diagnosis is reachable by shifting support within the existing network of symptoms and hypotheses. If this is possible, MOLE does one of the following:

- If a hypothesis' support needs to be driven down, and it does not have strong negative support, MOLE asks for information that would tend to rule it out.
- If a hypothesis's support needs to be driven up, and it has strong negative support, MOLE asks for background conditions that would mask negative support.
- If a hypothesis's support needs to be driven up, and it does not have strong positive prior-conditions, MOLE asks for positive prior-conditions.
- If a symptom's support needs to be shifted from one hypothesis to another, MOLE asks for distinguishing conditions.
- If the user provides no additional information, MOLE either revises support values or reinterprets parts of the network depending on its confidence in its interpretation and in its support values.

On the other hand, if the the targeted diagnosis is not reachable by shifting support within the current network of symptoms and hypotheses, MOLE tries to determine what part of the required structure might be missing:

- If a hypothesis cannot be rejected because it is needed to explain given symptoms, or if a hypothesis is accepted because it is the only explanation of a symptom:
    - MOLE asks for alternative explanations.
    - If no such hypotheses is provided, MOLE assumes that the observation of this symptom is not always reliable and adjusts the default confidence (initially 1.0) in the symptom downward.
- If a hypothesis was rejected but should not have been, then MOLE asks if there is some symptom which the hypothesis would explain, but which is not currently associated with it in the network.

When faced with a choice between revising support values and re-interpreting the network, MOLE bases its decision on its confidence in past decisions. In order to avoid thrashing, MOLE keeps a record of any revisions in support values that it makes. This enables it to know whether it has revised a support value in the opposite direction in the past. The source of a support value and its degree of stability are used to determine a weight which represents MOLE's confidence in the support value. Similarly, during static analysis MOLE records its confidence in any interpretations of the network that it makes. MOLE remembers whether its interpretation of a link or node was specified by the user or determined by its heuristics. If the interpretation is a reasoned guess based on its heuristics, MOLE assigns this guess a degree of confidence reflecting the strength of the heuristic used. MOLE changes those parts of the network in which it is the least confident.

It should be stressed that the static mode of analysis does not remove all ambiguities in the network. Some ambiguity may be inherent to the network and can only be disambiguated in the context of actual examples. When staticly disambiguating the network, certain associations are represented by several types of links. Some of these extra links need to be pruned. By examining which associations are needed in the context of diagnostic cases, MOLE is able to determine when it is possible to prune some of these associations. However, MOLE's performance system does not require that all ambiguity be resolved. Sometimes ambiguity is inherent to the problem and the associations can only be disambiguated in context. For example, a node which in some instances may serve as a hypothesis explaining a second node, may in other instances serve as circumstantial evidence for this second node. The interpretation will depend upon which node's value is discovered first.

An example from the rolling mill system will illustrate how MOLE uses dynamic analysis. Suppose the user has indicated that the steel is too narrow and that the looper roll is oscillating. Based on this information, MOLE would conclude that there is a worn out roll. This is the only hypothesis which would explain the oscillating. There are two other hypotheses -- i.e., excessive tension between rolls and too narrow on entry into the mill -- which would explain why the steel on exit from the mill is too narrow, but since the narrowness on exit can be explained by a hypothesis which is needed for independent reasons, these two alternative hypotheses are rejected.

Now suppose the expert indicates that MOLE should have accepted one of these two alternative hypotheses and rejected the worn out roll hypothesis. MOLE will ask the user to give an alternative explanation for why the looper roll is oscillating. Since every symptom must have an explanation, and the only explanation for the oscillation hypothesis that MOLE knows about is a hypothesis that it is told to reject, MOLE concludes that there must be an alternative explanation. If the expert says that there is no such alternative hypothesis, MOLE asks the expert how certain he is that the roll really is oscillating. If the expert says that he is certain, then MOLE will provide a "dummy" hypothesis for explaining the symptom. MOLE assumes that this dummy explanation is uninteresting because either it explains an event that occurs often in non-problematic situations or it explains an event the expert does not understand. There is one other alternative. If MOLE is not very certain that the oscillating roll observation is a symptom, MOLE will tentatively try treating it as a prior-condition so that it does not have to be explained by any hypothesis.

Suppose, on the other hand, MOLE is told that the steel is too narrow on exit, that it was not too narrow on entry, and that there is no oscillation problem. In this case, it would conclude that there must be excessive tension between rolls. If the expert indicates that he is undecided between this hypothesis and the worn out roll hypothesis, MOLE will first focus on why it ruled out the worn out roll hypothesis. It will discover that the reason is that the oscillation symptom failed to occur. MOLE will ask the expert whether there is any background condition which masks the negative effect of the failure of this symptom. It might be that MOLE does not yet know that worn out rolls do not typically lead to oscillation if the alloy is soft. If the expert fails to indicate that there is such a masking condition, MOLE will ask for positive prior-conditions that increase the likelihood of a worn out roll and offset the negative affects of the oscillation failing to occur. Ultimately, if the expert does not indicate additional information, MOLE will try revising the default support values by shifting them from the accepted hypothesis to the worn out roll hypothesis so that neither will be above the accept threshold.

As MOLE has evolved, dynamic analysis has become more critical. In the earlier versions in which the expert was required to describe the knowledge base in terms precisely understood by MOLE, dynamic analysis was only useful for finding missing knowledge and adjusting support values. Now dynamic analysis is also needed for correcting wrong guesses made during static analysis. In the earlier versions wrong guesses were made as well, but they were made by the expert who did not understand how to map his knowledge into the types of associations understood by MOLE. When doing dynamic analysis MOLE had little basis for distinguishing between those instances where the expert knew what he was doing and those where he was guessing. By allowing the expert to be unspecific about association types when he is unsure, MOLE has some basis during dynamic analysis for knowing what relations in the network are guesses and thus reasonable candidates for reinterpretation.

4. Conclusion

MORE, MOLE's predecessor, was used to build knowledge-based systems that diagnosed computer disk faults, computer network problems, and circuit board manufacturing problems. Experts were able to use MORE to build these systems only after they had acquired an understanding of how MORE worked. In each case, the initial sessions with MORE had to be treated as training sessions. The expert had to learn to "think" like MORE. Our subsequent efforts have been directed toward not bothering the expert with unnecessary questions and enabling MOLE to treat the expert's responses in a more tentative fashion. As a result less time is needed for the expert to familiarize himself or herself with the system. The current version of MOLE has been used to build systems that diagnose rolling mill problems and help with Micro-Vax tuning. MOLE is currently being used to build a system for doing power plant diagnosis. In addition, we are exploring its use in non-diagnostic domains. We are planning to use MOLE to build a system that selects computer components based on a set of generic specifications.

Acknowledgements

We want to thank Damien Ehret, Gary Kahn, Sandra Marcus, and Ming Tan for helpful suggestions in the development of MOLE.

References

[Boose 84] Boose, J. Personal construct theory and the transfer of human expertise. In *Proceedings of the National Conference on Artificial Intelligence*. Austin, Texas, 1984.

[Buchanan 84] Buchanan, B. and E. Shortliffe. *Rule-based Systems: the Mycin experiments of the Stanford Heuristic Programming Project.* Addison-Wesley, 1984.

[Clancey 84] Clancey, W. Classification problem solving. In *Proceedings of the National Conference on Artificial Intelligence*. Austin, Texas, 1984.

[Clancey 85] Clancey, W. Heuristic classification. *Artificial Intelligence* 27, 1985.

[Davis 82] Davis, R. and D. Lenat. *Knowledge-Based Systems in Artificial Intelligence.* McGraw-Hill, 1982.

[Kahn 85a] Kahn, G., S. Nowlan, and J. McDermott. Strategies for knowledge acquisition. *IEEE transactions on Pattern Analysis and Machine Intelligence* 7(5), 1985.

[Kahn 85b] Kahn, G., S. Nowlan, and J. McDermott. MORE: an intelligent knowledge acquisition tool. In *Proceedings of Ninth International Conference on Artificial Intelligence*. Los Angelos, California, 1985.

[Miller 82] Miller, R., H. Pople, and J. Myers. INTERNIST-1, an experimental computer-based diagnostic consultant for general internal medicine. *New England Journal of Medicine* 307, 1982.

[Pople 82] Pople, H. Heuristic methods for imposing structure on ill-structured problems. In Szolovits, P. (editor), *Artificial Intelligence in Medicine*. Westview Press, 1982.

[Rich 79] Rich, E. User modeling via stereotypes. *Cognitive Science* 3, 1979.

[Shortliffe 76] Shortliffe, E. *Computer-Based Medical Consultation: Mycin.* Elsevier, 1976.

[Szolovits 78] Szolovits, P. and R. Patil. Categorical and probablilistic reasoning in medical diagnosis. *Artificial Intelligence* 11, 1978.

[Weiss 78] Weiss, S., C.A. Kulikowski, S. Amarel, and A. Safir. A model-based method for computer-aided medical decision-making. *Artificial Intelligence* 11, 1978.

[Weiss 79] Weiss, S. and C. Kulikowski. EXPERT: a system for developing consultation models. In *Proceedings of the Sixth International Joint Conference on Artificial Intelligence.* Tokyo, Japan, 1979.