Intelligent agents extend knowledge-based systems feasibility

by G. Elofson

Episodic classification problems are the transient responsibilities of the knowledge worker—growing and then receding in importance over time. Typically, episodic classification problems do not conform to traditional expert system solutions, and they require specialized architectures to offer decision support and increased span of control for those individuals whose task is expediting the problems. This paper describes a system for addressing episodic classification problems, giving appropriate functional and technical detail, and continues by illustrating the effectiveness of the system through a case study of use of the system.

For reasons other than a lack of technical feasibility, often having to do with prohibitive time and cost constraints, a variety of knowledge-intensive problems are unsuitable for expert system approaches. This paper presents a system for solving a relatively overlooked subclass of these kinds of problems—episodic classification problems (ECPs). These problems require knowledge-intensive solutions that also tend to be transient in nature—not existing long enough to justify the time and cost of building an expert system solution. Additionally, with ECPs, experts are typically not available as sources of knowledge because the demand for them is prohibitively high. Moreover, simply knowing a method of inquiry and what questions to ask helps in solving ECPs.

This paper begins by describing in detail several examples of episodic classification problems. It continues by describing the functional characteristics of a prototype, the knowledge cache, for solving ECPs. This explanation provides the groundwork for understanding how a knowledge cache aids the decision-maker by increasing span of control. Then the functional and necessary technical descriptions of this architecture are explained. In addition to specifying the system used, detailed attention is given to the learning algorithm as well as to the opportunistic search heuristics used. This paper concludes with an illustrative example of the way in which the knowledge cache solves an ECP, and several assertions about system considerations for the future.

Episodic classification problems

The first example of an episodic classification problem comes from a recent Harvard case study.[1] A major software vendor planned to ease the workload on its customer service representatives with the introduction of expert system technology. The vendor received over 330 000 calls annually and wished to create an on-line service that allowed customers to get software support. The plan included having customers interact with an expert system instead of a service representative. The system would require the customer to report the nature of the problem (e.g., "I've got an abend."); ask questions of the customer that would have been asked by the service repre-

 0018-8670/95/$3.00 © 1995 IBM

sentative (e.g., "Was it a system abend? A user abend? Was there a console message? None of the above?"); and provide, when possible, a classification or solution to the problem. In those instances where the expert system was unable to provide such a classification, the service representative would make the decision.

The firm believed that the expert system would differentiate the firm's services from those of its competitors, help reduce a service representative's workload, and encourage the service representatives to remain at their jobs longer. (Their average length of job service was 20 months, and the cost of retraining was viewed as significant.) Thus, the plan found encouragement from many individuals within the company. Unfortunately, the project failed because of resistance on the part of the service representatives to spend additional time helping to build the system, coupled with the large knowledge acquisition effort required to bring the project on line.

Expert system technology is inappropriate for solving many episodic classification problems.

The second example of an episodic classification problem comes from the literature on environmental scanning.[2] To supply the strategic planning function of a business entity with information regarding threats and opportunities, managers and analysts combine their efforts in monitoring and searching the environment external to their enterprise.[3–5] The managers are experts in some aspect of the external environment such as political events, regulatory measures, and competitor financial status and decide to monitor sets of qualitative indicators that might provide insight into various threats and opportunities to the organization. Once the indicators are chosen, the managers request estimates of the values of the indicators from analysts. An analyst finds, interprets, and forwards information that explains the indicators in question.

The manager continually generates a list of information requirements and sends it via electronic mail to the analyst. Once the sought-after information is made available to the manager, inferences and classifications over the content of the data are made. For example, the manager may look for patterns over variables such as bidding behavior, research and development expenditures or hiring, new manufacturing methods, and suppliers. The manager may use his or her expertise to infer that a very low bid on the competitor's part may indicate several conditions: (1) the competitor's backlog is very low, (2) the competitor has made a leap in manufacturing methods and can reasonably meet the bid, (3) the competitor has made a gross error in judgment, or (4) the competitor is using a new supplier that can provide materials at a much lower cost. The manager would use the other variables to decide which of these explanations is most likely. If it is known that research and development expenditures have recently been cut and that there has been a hiring freeze, the manager will likely infer that either the competitor's backlog is low or there was a gross error in judgment. Conversely, if hiring for research and development has recently increased and the competitor has invested in a new manufacturing site, it may be that technological innovation is the best explanation for the very low bid.

Two types of knowledge are needed here: pattern classification knowledge and knowledge of what indicators are needed. Not only is it important to provide an assessment of information once the right questions have been asked, but it is also necessary to know just what those "right questions" are. Additionally, as in the case of the customer service representatives, the managers are rarely available for knowledge engineering interviews. Moreover, the problems that they are responsible for classifying tend to change over time as the goals of the organization change. An expert system solution is not a feasible approach for these two reasons.[6]

The third example of an episodic classification problem comes from the process of interpreting large volumes of radar images. Digital Synthetic Aperture Radar (SAR) imagery is being produced in ever-increasing quantities. One of its uses is as a relatively low-cost intelligence platform to track

Table 1 Examples of ECPs

Problem	Time Limitations	Expertise Scarce	Questions Needed to Focus Search
Workload of customer service representatives	Release of new software	High personnel turnover and no available time	How to narrow problem focus to save time
Gathering of political intelligence	Short-term interest in country	Shortage of experts	How to reduce analysis time for answers
Interpretation of radar imagery	Changes in force deployment	Limited personnel and some turnover	How to prescreen targets to shorten interpretation time

the activities and locations of military targets. Looking similar to a star map, low-resolution SAR imagery is "read" by image analysts as they comb through a given area in search of tactical deployments. An example of a typical pattern found in SAR imagery is a SAM (surface-to-air missile) site. In SAR imagery a SAM site often appears to have a "five-on-a-die" pattern. Another typical series of returns is the tank company—appearing as a line of "stars" with one slightly out of line (usually the commander). As one might expect, the force deployments change with time and terrain. So, new configurations must be classified by an image analyst with some regularity. Thus, positive contributions are provided from a computer-based system that assists the image analyst in reducing the amount of imagery that must be processed.[7]

Image analysts are in short supply. This scarcity of expertise, coupled with the changing nature of the problems to be classified and additions of new weaponry and force deployment, marks similarities between the SAR image exploitation problem, the monitoring problem, and the software service problem.

These episodic classification problems have been evaluated from an expert system point of view. In each case, expert system technology was found to be an inappropriate solution for the following reasons:

1. The problem is not recurrent for an extended period of time.
2. An expert is not available to help populate the knowledge base.
3. The goal of the process is not only to find correct answers but also to find the correct questions to ask.

Table 1 summarizes these points for ECPs. In the first example, the customer service representative answers questions about software problems, but those problems may not last after the next new release. In the second example, the manager scans for particular threats and opportunities until the next strategy meeting—when the change in direction of the firm will redefine what constitutes a threat or opportunity. Third, the image analyst identifies particular force deployments until new ones are created to suit changing terrain and tactics.

Exacerbating these difficulties is the fact that the customer service representatives remain in their positions for only a short period of time and are typically overburdened to the point of being unavailable for knowledge engineering sessions. The managers fall into the same category. For example, specialists in identifying political conditions in foreign countries are in short supply—when their expertise is applied to underwriting policies for the insurance industry, policies stop being created the moment that the individual is away. Similarly, the number of image analysts available to interpret SAR imagery is small in comparison to the demands placed on them. They are also unlikely candidates for lengthy knowledge acquisition sessions.

Because of the time limitations faced by these experts, small gains can provide a significant advantage. For instance, simply asking a series of routine questions of an individual needing software support can offer meaningful advantages for the service representatives, giving them additional time to concentrate on more challenging and unique problems. So too with the managers: much time can be spent on the activities of recalling their monitoring strategies for a particular threat or opportunity and asking for that information from an information analyst. Again, much of the image analyst's time is spent in combing through an image. Given this constraint, simply locating potentially meaningful clusters is a boon to productivity.

These problems, called episodic classification problems (ECPs), are transient, emerging and receding in importance over time. They appear in environments where knowledge-intensive attention is highly scarce and the cost of searching for confirming and clarifying data is high.

Knowledge technologies for ECPs

The product and process attributes of an expert system architecture make it difficult to use for solving ECPs. The lengthy time requirements of knowledge acquisition, the changing nature of high-priority problems, and the monolithic structure of traditional knowledge bases[8] make the expert system architecture an unfeasible design alternative.

The system necessary for solving these kinds of problems must be able to assimilate the expert's knowledge, organizing it by classes or categories. Also, the necessary system must have the ability to deliver the expert's knowledge to others in a usable form. It must unobtrusively learn the concepts that the expert uses in solving ECPs and distribute that knowledge to others in the organization.

Providing information technology support to solve ECPs can be done through knowledge caching. Central to the concept of knowledge caching is the intelligent agent. Here, the intelligent agent acts as *apprentice* to an expert, learning relevant facets of the expert's problem-solving abilities through machine induction. That is, an apprentice's orientation is problem-specific. For example, when scanning the environment of an organization for threats and opportunities, one apprentice would be concerned only with political violence problems, whereas another would be concerned only with a competitor's increase in distribution channels.

An apprentice's functionality can be described by an organization doing a particular monitoring activity. For example, when an organization monitors the political climate of a foreign country, the attributes a manager considers include the following: proregime and antiregime sense of relative deprivation; proregime and antiregime belief in violence; coercive force available to both proregime and antiregime actors; and institutional support for both proregime and antiregime actors.[9] To acquire values for these attributes, the manager creates a new apprentice that carries the following information to the analyst:

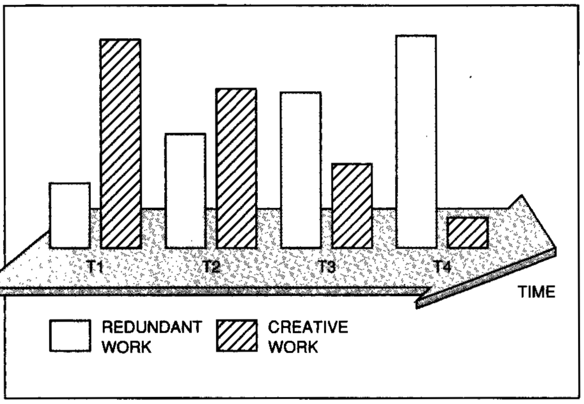
- A list of attributes for which the manager requires values, along with the name of the apprentice—corresponding to the threat or opportunity being monitored
- Explanations of the attributes to better clarify the nature of each listed attribute
- Likert-type scaling information that specifies which values are acceptable as answers to the attribute requests

This apprentice—something of a structured message having questions about attribute values, together with the explanations and scaling information—is sent to the analyst. To each of the attribute questions the analyst responds with a value corresponding to one of the scaled values provided by the apprentice. The analyst may also provide written explanations of why a particular scaled value was chosen.

Following the analyst's response to its query, the apprentice returns the new information to the manager. Here, the apprentice shows the specialist the answers to the specified questions and asks (in effect), "What does it mean?" In reply to this query, the manager may ask for an explanation of one of the analyst's answers or give the apprentice an assessment, or a classification, of the information provided. The apprentice forms an initial concept with the given classification and represents the concept as a classification and collection of attribute-value pairs.

Following this initial creation of an apprentice, restating the questions, explanations, and scaling information is unnecessary. The next time the manager needs information about the same threat or opportunity, the manager need only send the apprentice to do its work; it already has the questions and other requisite information. From that point, the apprentice proceeds to the analyst, as before, and asks the same questions and receives a new set of answers. With these new answers the

Figure 1 Change in work over time



apprentice returns to the manager, and if the apprentice has a concept that matches the answers provided by the analyst, it reports its classification to the manager. Otherwise, it again asks the manager, "What does it mean?," and the manager provides another classification that the apprentice uses to augment its concepts and make further generalizations.

Under these circumstances the apprentice is gathering information from the analyst and knowledge from the manager. The apprentice asks questions and receives answers from the analyst and, in doing so, gathers information. When the apprentice asks the manager what a particular set of attribute-value pairs means and receives a classification with which to form a concept, the apprentice gathers knowledge from the manager.

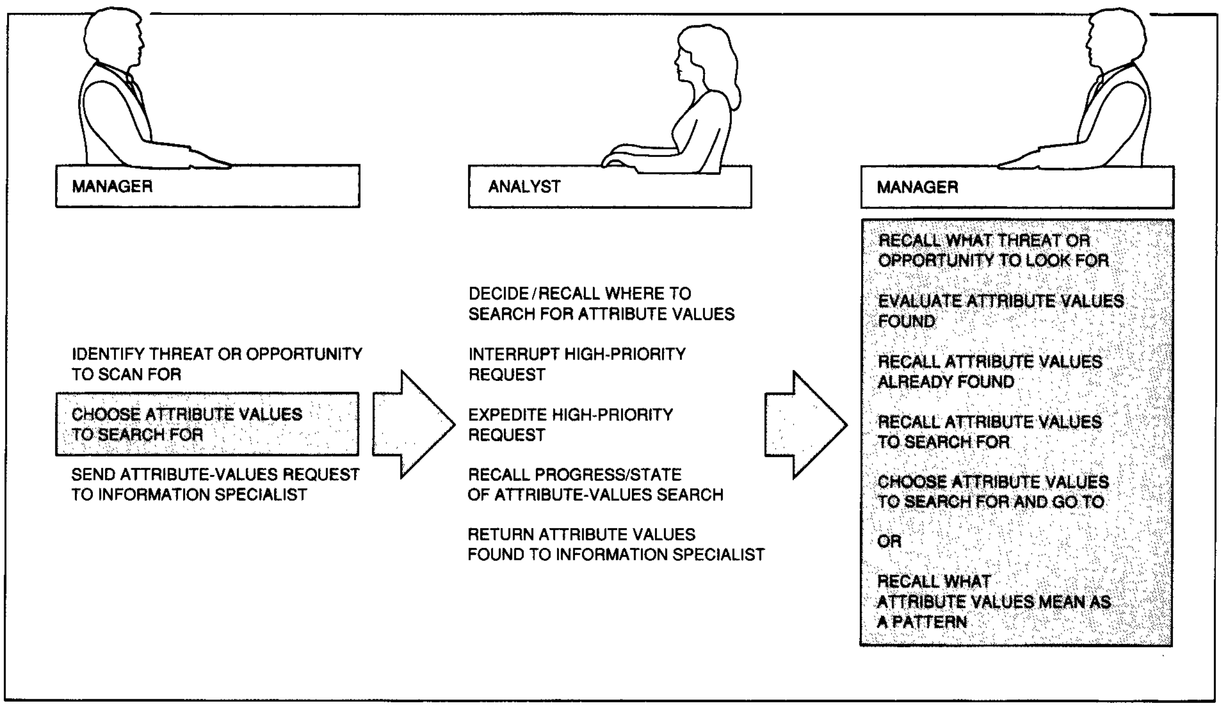
Increasing span of attention

Although an apprentice caches the knowledge an expert uses in solving problems, it also increases the expert's span of attention. That is, the expert who reapportions his or her knowledge to an apprentice is better able to attend to more challenging problems. For example, when the goals and objectives of the organization are first communicated to the environmental scanning group, a lot of time is spent identifying just what to look for, where to look for it, and how to look for it.[10] In other words, decisions are made as to what the possible threats and opportunities might be and how best to monitor them. Environmental scanners refer to this kind of work as "creative work."[11] As time passes and the scanners' tasks become more structured, an increasing amount of time is spent doing those tasks that have been structured and that become redundant, with the result that less time is available for generating new ideas for the problem of identifying threats and opportunities.

Figure 1 illustrates the relative change from creative to redundant work over time. As time passes for the manager, the creative work process becomes more infrequent as the amount of redundant work increases. Both tasks are necessary to properly fulfill the environmental scanning group's responsibilities.[12,13] Little can be done to increase the amount of time available for creative work, although the manager is viewed as having a fixed span of attention. The problems are not around long enough, and the expert's time is too scarce, to build an expert system. But, caching the expert's knowledge increases his or her span of attention through the delegation of redundant tasks. Knowledge caching allows the execution of the redundant tasks by an apprentice. It provides the manager with an opportunity to continually augment the search for new and unexplored sources of intelligence.

In Figure 2 the way in which a manager can delegate redundant work to an apprentice is shown. During this sequence of events, the analyst will often be asked to pursue other data requests by other managers. The initial request takes some time to expedite, and other requests, perhaps of a higher priority, present themselves to the analyst in the course of expediting the original request. Consequently, several hours can elapse between the time of the initial request and the response to that request. The manager too has other tasks to perform while waiting for the answer to his or her first data request. Other threats and opportunities to the organization must be scanned for; hence, other data requests must be made. When the manager does receive an answer to the initial request for data, the manager must reorient himself or herself to the goals of the scanning. What was being searched for must be recalled. The strategy for deciding how to proceed must also be recalled. What the data mean must be decided, and what to ask next must also be decided. If the requested information is only one part of a line of questions that have already been asked and answered, that previous information regarding the classification must also be recalled

Figure 2 Tasks performed by managers and analysts



and taken into consideration for the final classification.

The shaded activities of Figure 2 show what responsibilities an apprentice can assume after an initial inquiry is made by the manager. With the threat or opportunity to be investigated as the basis, a chosen apprentice will already have the correct data values to search for and conduct a question-answer session with the analyst. It will attempt to evaluate all attributes found and continue searching until it has either made a classification or exhausted its problem-solving knowledge. If the latter occurs, the chosen apprentice returns to the manager with the attribute values and asks for a classification. The apprentice can immediately take over the query responsibilities of the specialist in every instance, and over time it can expedite greater portions of the classification responsibilities.

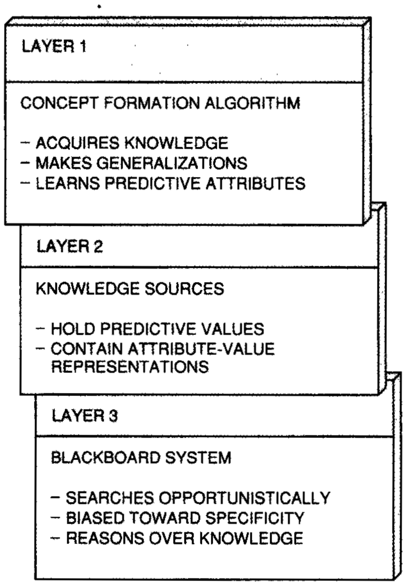
Agent architecture

An apprentice is made up of three layers (Figure 3), which together perform the tasks of: (1) acquiring, classifying, and distributing the information needed to classify the ECP, and (2) acquiring, classifying, and distributing the knowledge used by experts in diagnosing the ECP. The three layers are the concept formation layer, the knowledge source layer, and the blackboard layer.[14–19] (A prototype of the apprentice has been developed on an IBM Personal System/2* [PS/2*] with a 386 processor using Turbo Pascal.)

The central responsibility of the concept formation layer involves the gathering of groups of attribute-value pairs to form generalizations based on regularities in those groups. The knowledge source layer receives knowledge from the output of the concept formation layer. That is, the tree generated by the concept formation layer is parsed to yield a distinct set of concepts that become the knowledge sources for the apprentice. The inference engine for the apprentice is the blackboard layer. It uses the search information provided by the knowledge sources to intelligently query the data source for specific values of chosen attributes.

The concept formation layer. Unimem[20–22] was the chosen algorithm for the concept formation layer.

Figure 3 Apprentice architecture



Unimem is an incremental conceptual clustering algorithm that creates a hierarchy of feature-vectors from inputs of labeled sets of attribute-value pairs. Additionally, Unimem updates its hierarchy with each new piece of data it receives.

Within the hierarchy created by Unimem, feature-vectors found closer to the root are more general (having fewer attribute-value pairs and more labels) than those near the leaves. Also, the arcs pointing to the nodes in the hierarchy of Unimem have predictive attribute values associated with them. These predictive attribute values provide heuristic search information to the blackboard, as the node pointed to incorporates the predictive attribute-value pair, whereas the other nodes probably do not. Also, the classifications of the feature-vectors are identified by either a single label or a disjunction of labels.

A description of the Unimem algorithm, adapted from Reference 23, follows:

Step 1. An object is presented to be incorporated into the hierarchical clustering. Unimem first considers what children of the root of the clustering might serve to incorporate the object.

Step 2. A collection is made of all children of the root node that are indexed (predicted) by at least one value of the object.

Step 3. The object is incorporated into the clustering based on one of the following rules:

a. If no children of the root were collected in Step 2, then make the object a child of the root by directing arcs from the root to the object. All variable values of the object are considered predictive of the object in this situation. This may cause some values to be predictive of more than an acceptable number of clusters, thus causing these values to be removed as predictive of any cluster.

b. If some number of children of the root were collected in Step 2, then for each child perform the following:
   - Increment all of the child's predictable values that are present in the object.
   - If the object has a different value than the child along any variable, *then* do each of the following:
     - Decrement the weight of each predictable value that is not present in the object.
     - If the threshold of the decremented value falls below a user-specified threshold, then drop this value from the set of predictable values and remove this value as a predictive value of the cluster if the value is, in fact, predictive. Removing predictable values of a cluster (and thus the arcs labeled from these predictive values) may cause a cluster to be removed from the hierarchy if all such predictive values are removed.
     - If dropping values results in a concept with too few values (according to some user-specified limit), then remove the concept from the hierarchy by removing arcs to it from its parent.

   Else attempt to incorporate the object into one of the children of the cluster by treating each child as the root of a subordinate, clustering and recursively applying Steps 1 through 3.

c. If the object could not be incorporated into any cluster in steps a or b above, then make the object a child of the root by the same process as given in step a.

Table 2 Attributes, values, and their classification

	Product Innovation Classifications	Change in Distribution Channels	Cash Flow Status	Impending Plant Sale	Relative Market Share
1.	Vulnerable to innovations	Abandoned convenience stores	Highly leveraged	Yes	High
2.	No substantial change	No change	Highly leveraged	Yes	High
3.	No substantial change	Added convenience stores	Fairly liquid	No	High
4.	Possible product improvement	Product shift to drug stores	Highly liquid	No	High
5.	Possible new product	Product shift to drug stores	Highly liquid	No	Very high

Unimem gathers its new inputs of classified attribute-value pairs from the data source. (This source could be SAR image metrics, analyst's answers, or software user responses.) The classification of this information, the judgment call made by the expert about the returned information, becomes the label of the attribute-value data.

An example is the way in which the Unimem function could involve a manager looking for marketing vulnerabilities in a competitor's product line. In this case, a manager would monitor changes in the competitor's distribution strategies, cash position, plant usage, and market share while an apprentice captured knowledge about how to act on behalf of that same manager. A description of these attributes follows:

- Change in distribution status—an indication of the competitor preparing to introduce additional products by moving existing ones "down" retail outlets. For example, moving a line of men's cologne from department stores to drug stores, making way for a new product.
- Cash flow status—an indication of how leveraged a competitor is and, consequently, how capable of investing in new product development
- Impending plant sale—an indication of a desire to improve production, to start production on a new product, or to acquire more cash
- Market status—an indication of the relative market share of a given product line

With these variables, the manager infers whether a competitor is vulnerable to new product innovations. Table 2 gives five scenarios of a competitor's status in this regard, together with a classification of the information provided.

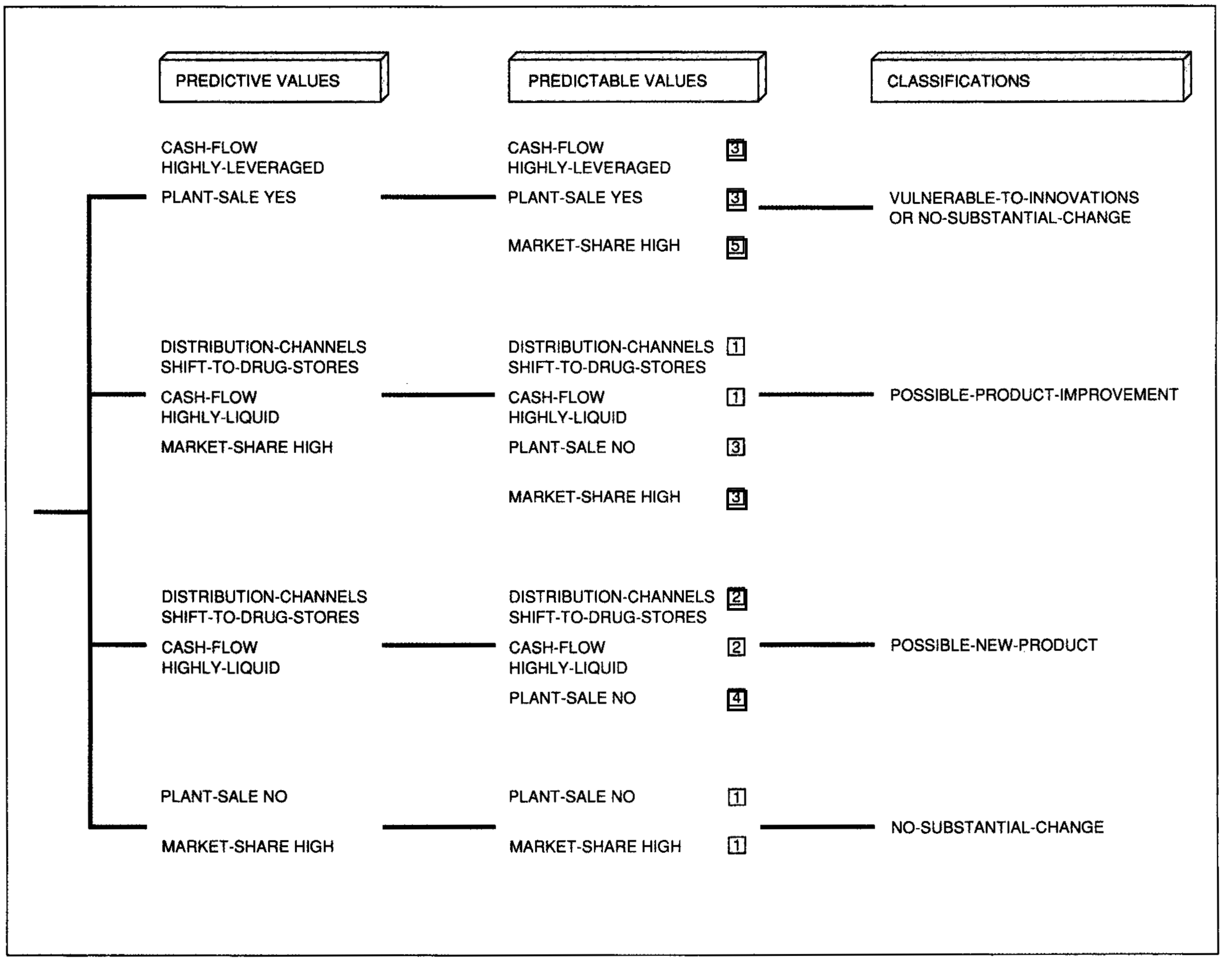
These lists of attribute values, together with their classifications (numbered 1 through 5 in the table), were submitted to an apprentice in the following order: 1, 2, 3, 4, 5, 4, 1, 3, 2, 5. That is, the attribute-value pairs corresponding to the classification "vulnerable to innovations" were presented first, those corresponding to "no substantial change" were presented second, with the rest presented according to the order given. The resulting hierarchy created by the concept formation layer is shown in Figure 4 (the numbers in boxes are the weights explained in the algorithm).

Knowledge source layer. The hierarchy formed by Unimem is parsed to create the knowledge sources in the three-layered architecture. The result is a number of hypotheses that correspond to each hierarchy node of Unimem, together with the arc labels pointing to them. Thus, a given hypothesis will contain information about both its predictive and predictable attribute values, as well as the classification given to it. A simple example of a hypothesis is as follows:

(plant-sale yes, cash-flow highly-leveraged) (market-share high) (vulnerable-to-innovations or no-substantial-change)

From left to right, three groups are in this hypothesis: predictive values, predictable values, and a classification label. In this example, if the issue in question has the characteristics "plant-sale yes" and "cash-flow highly-leveraged," it is likely that it also has "market-share high," more

Figure 4 Knowledge base generated by an apprentice

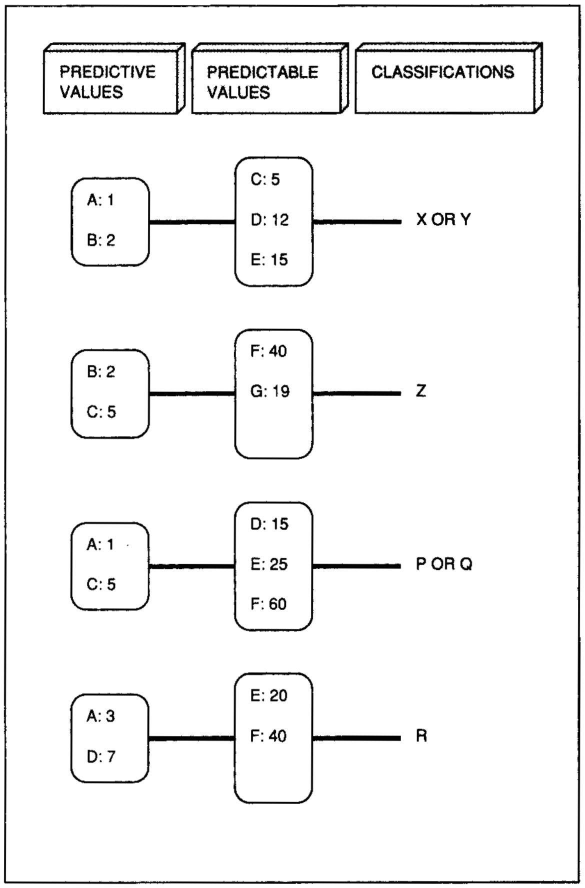


particularly "vulnerable-to-innovations or no-substantial-change." The values "plant-sale yes" and "cash-flow highly-leveraged" are the predictive values; their presence suggests the presence of the predictable value "market-share high." According to this hypothesis, if all three attribute-value pairs are matched, the conclusion is that the classification is "vulnerable-to-innovations or no-substantial-change."

The hybrid blackboard layer. Although explicit links are employed in the blackboard, it functions to exhibit global data to the various knowledge sources and provide the facility for opportunistic searching. The hybrid blackboard is made up of several data structures: the knowledge sources, the blackboard, the focus-of-attention list, and the hypothesis candidate list. The knowledge sources correspond directly to the concepts developed by the concept formation level of the apprentice. By using the predictive information provided by the concept formation level, along with the varying degrees of specificity of the knowledge sources (the more attribute-value pairs that describe the concept, the more specific it is), the control or search method is opportunistic. The blackboard holds the classifications of hypotheses under consideration, as well as their attributes and values. The focus-of-attention list is a stack of the hypotheses under consideration, the top of the stack being the currently considered hypothesis. The hypothesis candidate list holds the collected hypotheses to be chosen as a focus of attention.

The search method is opportunistic[24,25] because the cost to acquire individual data values is often nontrivial. For example, when conducting a search for a classification, it is first necessary to begin with some piece of information. Therefore, it is necessary to choose a particular attribute for which a value must be found. In making this choice, the preferred result will lead quickly to a conclusion, taking the search in a direction that requires fewer data requests of, say, an analyst rather than more data requests. In achieving this end, two constraints are placed on the selection of an attribute to investigate: (1) it must be a predictive attribute, as the presence of such an attribute-value match tends to rapidly decrease the number of alternative hypotheses that may be considered, and (2) the attribute chosen for inquiry must be the most frequently occurring predictive attribute across all hypotheses. The reason for this, too, is to prune the list of alternatives as quickly as possible from the list of competing hypotheses. Figure 5 contains an example.

Figure 5 A simplified knowledge base of four hypotheses



The heuristic used for choosing which predictive attribute to investigate first is a function of the greatest number of times each appears. That is, the total number of times each attribute appears is counted. Here, "A" occurs three times, "B" and "C" occur two times, and "D" occurs once. Thus, "A" is chosen to be investigated first. If, however, "C" had also appeared three times as a predictive value, the conflict would have been broken by choosing the attribute that varies most. For example, if "C" had three distinct values across the hypotheses shown (say "7," "8," and "9"), "C" would have been chosen instead of "A," as "A" has only the two distinct values of "1" and "3." Note here the implicit assumption that the apprentice will find a matching hypothesis for the incoming values. That is, if it is assumed that the value being sought will most likely match one of the values found in the hypotheses, the tree is most quickly pruned by choosing the most varying value. Conversely, if it assumed that the value being sought is only as likely to occur as any other possible value for the attribute in question, the tree is most quickly pruned by choosing the least varying value. From a search efficiency standpoint, this approach closely resembles a *branch and bound* approach, sometimes approaching the efficiency of A^* when the initial look at the predictive values serves to quickly reduce the search space. The code for this heuristic search is as follows:

```
New (q); P := q; S := q;
Read q^.att, test_val;
q^.pcount := 1; q^.rcount := 1;
While not (end_of_list) Do
  Begin
    Read att, val;
    If att <> q^.att then Begin
      New (q); S^.next := q; S := q;
      q^.att := att; q^.pcount := 0; Q^.rcount := 0;
      test_val := "000";
  End;
q^.pcount := q^.pcount + 1;
If val <> test_val then
begin q^.rcount := Q^.rcount + 1; test_val := val;
end;
End: {while}
```

Table 3 Blackboard flow of control

Step 1	Choose predictive attribute and find its value.
Step 2	Post predictive attribute value to the blackboard.
Step 3	Put eligible hypotheses on candidate list.
Step 4	Select hypothesis to process and place on focus of attention list.
Step 5	Find value for next attribute of focus of attention hypothesis, first predictive and next predictable, and post to blackboard.
Step 6	Based on attribute value found, use heuristic to choose current focus of attention. If new focus of attention chosen, go to step 7.
Step 7	More hypotheses left on candidate list? If yes, go to step 4. If no, report results.

This code creates a list of attributes and their number of occurrences as both predictive and predictable.

The next algorithm uses the previously described heuristics to choose an attribute for consideration:

```
Current_att := p^.att; Current_pcount := p^.pcount;
Current_vcount := p^.vcount; q := p^.next;
While q <> nil Do Begin
  if q^.pcount > p^.pcount then current_att := q^.att;
  if q^.pcount = p^.pcount AND
  q^.vcount > p^.vcount then current_att := q^.att;
  q := q^.next;
  p := p^.next;
End; {while}
```

Once an attribute is chosen and a value is assigned to that attribute by the analyst, a list of candidate hypotheses is selected from the knowledge base and placed on a list with other candidates (see Table 3). They are chosen simply according to whether or not they have the attribute just identified and whether their value for that attribute matches the value supplied by the analyst. Next, given that more than one hypothesis becomes a candidate, the selection of which of these hypotheses is to be the focus of attention is made. It is done by selecting the competing hypothesis with the most specific classification. For example, if one of the competing hypotheses would make the classification "A OR B," while another of the hypotheses would make the final classification of just "A," the hypothesis that was the more specific of the two, the one concluding "A," would become the focus of attention. To further complicate matters in this process of conflict resolution, there is the problem of deciding over two competing hypotheses when both are most specific. That is, what does the system do when both competing hypotheses have only one conclusion? The answer here is indicative of a bias toward predictiveness. The hypothesis with fewer unanswered predictive values is opted for. Should these also be equal, the hypothesis with fewer unanswered predictable values will be opted for. And if these are also equal, the choice is simply arbitrary—the first hypothesis on the list is chosen.

Once chosen, the focus of attention remains in place during the process of identifying values for, first, its predictive values and, second, its predictable values. It loses its position as the focus of attention when there is either no match or when another hypothesis becomes a more desirable candidate (using the heuristic described above).

Overall, these three layers effect the required functionality of the apprentice. The concept formation layer is put to use when the apprentice "asks" the expert for a classification of attribute-value pairs. The classification given is used, along with the attribute values provided by the data source, to augment the concepts via the learning algorithm. The output of Unimem is put into the knowledge sources, which are used by the blackboard to conduct a search of the data source. Thus, different parts of the apprentice's architecture are used according to the individual with which the apprentice is interacting. The blackboard and knowledge sources are used with the data source, and the concept formation layer is used with the manager.

Case study

To show a real-world application of this apprentice prototype, an archival case study[26–28] was performed. The subject of the case study concerned a multinational corporation[29,30] monitoring the political climate of Poland in the summer of 1980.[31] Here, a manager monitored the political climate of Poland in the summer of 1980 while an apprentice captured knowledge about how to act in the absence of that same manager.

A composite of attributes, values, and rules employed by political analysts in identifying the likelihood of political turmoil were used in place of an

Table 4 Attributes used for identifying political turmoil

Attributes	Explanations
Success of antiregime movements outside of country	A measure of the extent to which antiregime members outside of the country have succeeded in achieving their own political objectives
Institutional support for proregime	A measure of the extent to which proregime members are supported, in terms of organizational cohesion and the size and geographic location of their resources, as well as psychological, economic, and political support short of coercive force, in achieving their objectives
Institutional support for antiregime	A measure of the extent to which antiregime members are supported, in terms of organizational cohesion and the size and geographic location of their resources, as well as psychological, economic, and political support short of coercive force, in achieving their objectives
Proregime relative deprivation	A measure of the extent to which members characterized as proregime feel frustrated regarding their economic condition and general welfare
Antiregime relative deprivation	A measure of the extent to which members characterized as antiregime feel frustrated regarding their economic condition and general welfare
Proregime belief in violence	A measure of the extent to which members characterized as proregime believe that, given the practical opportunities and limitations of the current political situation, violence is justified on either pragmatic grounds, or on moral, doctrinal, and historical grounds
Antiregime belief in violence	A measure of the extent to which members characterized as antiregime believe that, given the practical opportunities and limitations of the current political situation, violence is justified on either pragmatic grounds, or on moral, doctrinal, and historical grounds
Coercive support for proregime	A measure of the extent to which proregime members are supported, in terms of equipment, training, size, strategic location, and loyalty of armed manpower from within and without the country
Coercive support for antiregime	A measure of the extent to which antiregime members are supported, in terms of equipment, training, size, strategic location, and loyalty of armed manpower from within and without the country

actual expert. These were based on a well-known approach, developed by Gurr,[32] to analyze the political climate of a country. His model takes into account a number of qualitative indicators to assess the well-being of the political health of a country or the likelihood of an outbreak of political violence. These attributes were used in illustrating how a manager would seek and classify information about Poland. They are explained in Table 4.

The manager in this case study used these variables to project the likelihood and type of violence that might have occurred in Poland. The events that are relevant to this case study take place in a short period of time—two months. The "high points" of this period of time are chronicled in Table 5.

Together with information concerning general trends and conditions in Poland, these "high points" provide the substance of the case study that follows.

With this information, the monitoring activity begins on July 7. First, the manager creates an apprentice and sends it to the analyst who answers its questions and returns it. With the July 7 information, the manager makes a classification of the attribute values he receives. On July 21, after two weeks, the manager again sends the apprentice to do its job and classifies the attribute values upon its return. Because the July 21 attribute values held the potential for escalating violence, the manager sends the apprentice again after nine days instead of two weeks. Now he finds that his suspicions were correct and that the situation in Poland had become somewhat more volatile, but not alarmingly so. Thus, he sends the apprentice again, one week later, and finds that the situation has made no dramatic change. Another week passes and he sends the apprentice again. It is

Table 5 Highlights of Polish conflict in the summer of 1980*

Date	Event
July 1	The Central Committee announces a large hike in meat prices.
July 2	The first strike breaks out in the Ursus plant near Warsaw. The strike was for higher wages to cover the increased meat prices.
July 9	Central Committee First Secretary Gierek makes public statement that no broader wage increases will be allowed, but workers continue striking.
July 16	Strikes spread to nearby Lublin.
July 27	Gierek goes on holiday to Moscow and does not return until August 14.
August 15	Telephone lines to Gdansk are cut.
August 16	Representatives from 21 enterprises convene in Gdansk to form the interfactory strike committee (MKS), a group formed to coordinate strike action, promote solidarity, and begin broadening demands to include political concessions.
August 18	The MKS continues to grow while Gierek makes a television speech stating that no political concessions will be made with the strikers.
August 19	Dissidents are arrested on a widespread basis.
August 23	Barcikowski, Deputy Prime Minister, begins talks with the MKS on national television.
August 24	Four top Central Committee members are relieved of their posts.
August 30	The Gdansk agreement is signed, guaranteeing the right to strike and self-governing trade unions.

*Note that this information was widely available during this time period. For example, from July to November *The New York Times* printed a front-page story on Poland every 2.5 days.

now August 16, and the manager finds that the Central Committee has chosen a destabilizing course of action—cutting telephone lines into Gdansk. Now, the possibility for violence is strong. To more closely monitor the situation, the analyst increases the frequency of checking the situation, sending the apprentice out again after five days, and finds that tensions in Poland still remain very high. Three days later the manager sends out the apprentice again. Now, tensions have somewhat lessened, but the possibility of political violence is still present.

Applying the variables from Table 4 to the events in Table 5, the apprentice generated the knowledge base shown in Figure 6 (as of August 23). There are three possible classifications of the likelihood of political turmoil in this knowledge base thus far: somewhat likely, likely, and highly likely. The attribute values in the top layer of the tree are the predictive values used by the blackboard in selecting a search strategy. The values in the middle layer of the tree are the predictable values that follow from the occurrence of the predictive values. If all values match the available information, the appropriate classification at the bottom of the tree is chosen.

Using this knowledge base, the apprentice was allowed to do its own analysis of the events up to and including August 24 (the assumption being that the "manager" had taken a vacation). In this case, when it was the apprentice's "turn" to classify the political information (see Table 6), the apprentice noted that the information fit two classifications: "PT: somewhat likely" and "PT: likely." Had the manager been present, undoubtedly only one classification would have been provided. This fact supports what we would intuitively believe of an apprentice; that is, after existing for only six weeks and having seven learning experiences, it is not going to be as skilled at classifications as an established expert. Nevertheless, the apprentice was able to provide a useful indication of what the pattern of qualitative political data meant. The strategy of an apprentice is conservative. It identifies only what it has already observed or some subset of what it has already

Figure 6 Knowledge base of an apprentice

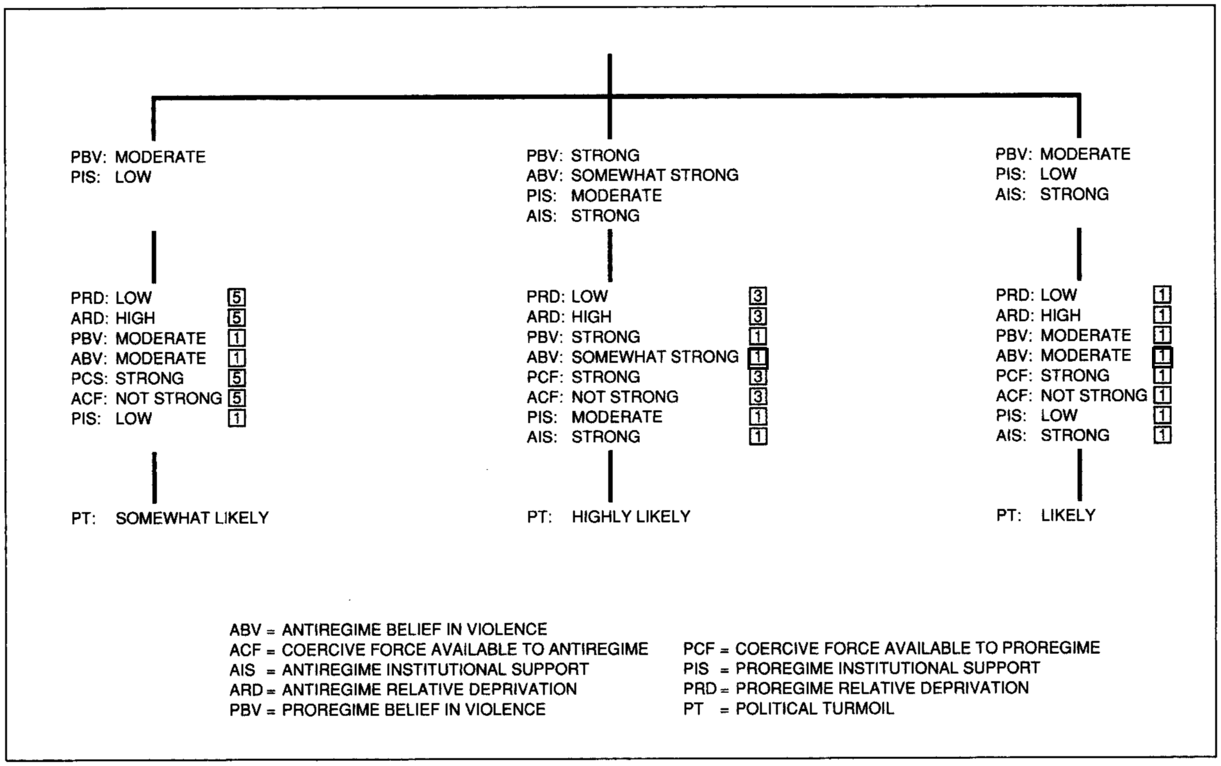


Table 6 Attributes, values, and explanations for political events on August 24

Attributes	Values	Explanations
Proregime relative deprivation	Low	The proregime party has enjoyed special privileges for some time.
Antiregime relative deprivation	High	The economic conditions of the workers have not improved, but declined over the past decade.
Proregime belief in violence	Moderate	Gierek agreed to unions holding secret elections on the 25th—indicating continued efforts at cooperation.
Antiregime belief in violence	Moderate	Continued success of MKS in getting secret elections will further convince strikers of soundness of organized approach.
Proregime coercive force available	Strong	The Polish army and police are supportive of the party and the Central Committee.
Antiregime coercive force available	Not strong	The Polish workers are not armed and little armed support for them exists outside of their country.
Proregime institutional support	Low	Gierek's concessions to the unions will only further polarize the already divided party and Central Committee.
Antiregime institutional support	Strong	Achievement of political gains in terms of secret elections will further strengthen the resolve of the strikers.

Table 7 Knowledge sources used in blackboard search

Predictive Values	Predictable Values	Classifications
PBV: moderate PIS: low	PRD: low [5] ARD: high [5] PBV: moderate [1] ABV: moderate [1] ACF: not strong [5] PIS: low [1]	PT: somewhat likely or likely
PBV: strong ABV: somewhat strong PIS: moderate AIS: strong	AIS: strong [1] PRD: low [3] ARD: high [3] PBV: strong [1] ABV: somewhat strong [1] PCF: strong [3] ACF: not strong [3] PIS: moderate [1]	PT: highly likely
PBV: moderate PIS: low	AIS: moderate [1] PRD: low [1] ARD: high [1] PBV: moderate [1] ABV: moderate [1] PCF: strong [1] ACF: not strong [1] PIS: low [1]	PT: likely
PBV: moderate PIS: low AIS: moderate	AIS: moderate [2] PCF: very strong [2] PRD: low [5] ARD: high [5] PBV: moderate [1] ABV: moderate [1] PIS: low [1]	PT: somewhat likely

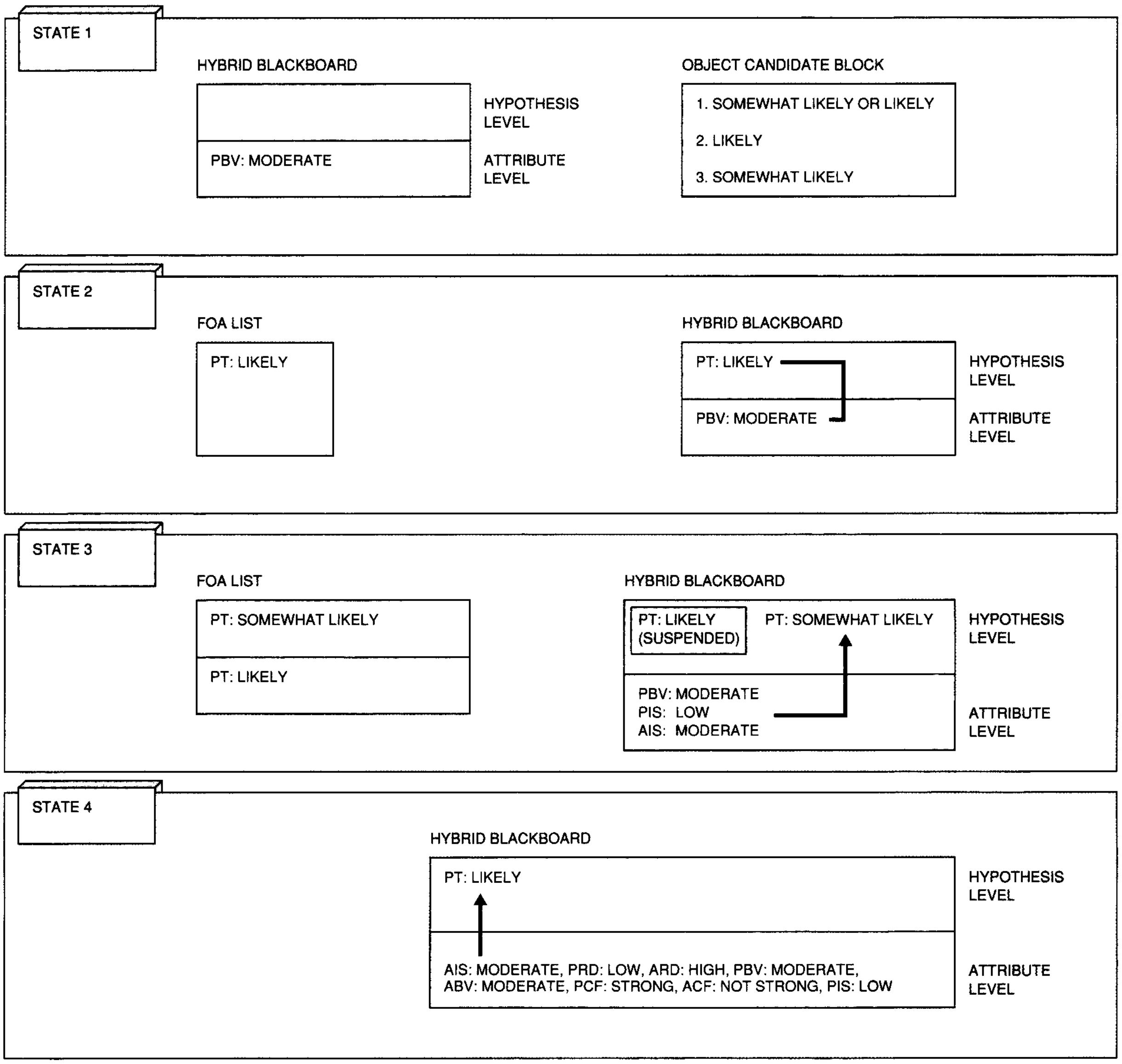
observed. If it is confronted with a pattern completely unlike anything it has come across before, it defers judgment and asks for a classification.

A simple example of how the hybrid blackboard of the knowledge cache solves the classification of recognizing the likelihood of political turmoil completes our description of the system.[33] Using the hypotheses from Table 7, we present an example of one hypothesis first being considered by the system (likely chance of political turmoil). This hypothesis is then pushed onto the focus-of-attention list while another hypothesis is considered (somewhat likely chance of political turmoil). Finally, the original hypothesis is opted for.

In Figure 7 we show how the data structures of the blackboard change as the controller goes through its algorithm. First is the initialization step. Here, with the use of the search heuristics described above, the variable PBV is chosen and is found to be "moderate." This value is posted to the attribute level of the blackboard, and a series of candidate hypotheses are identified (State 1). The hypothesis "likely" is chosen from the candidate list, becomes the focus of attention, and is placed on the blackboard (State 2). After two more attributes are identified, the focus of attention is changed to the "somewhat likely" hypothesis (State 3). This hypothesis is rejected after additional attribute information is found, and the system finally concludes that the hypothesis "likely" is the correct one (State 4).

This case illustrates how an apprentice can be successfully used to capture knowledge about "real-world" information and established methods of classifying that information. At any time during the period discussed above, other managers would also have access to the apprentice monitoring political violence in Poland. For example,

Figure 7 States of the hybrid blackboard during the search for a classification



during the Solidarity movement the stock market fell over a period of time while the price of gold rose. Investment personnel may have chosen to use the apprentice as one of the inputs to their investment decisions, and they could continue using it in the originating manager's absence. Finally, had the manager who was responsible for the creation of the political violence apprentice left the firm instead of taking a vacation, his or her replacement would have had an immediate starting point, using the apprentice, from which to continue the work.

Conclusion

Knowledge-intensive problems may resist expert system solutions. Episodic classification problems—temporary in nature, very scarce in available expertise, benefiting even from appropriate query generation—represent one class of just these kinds of problems. We have introduced an approach for solving these kinds of problems that centers on caching knowledge. The application that has been discussed was in the area of environmental scanning. But we believe that relatively simple modifications to the basic architecture introduced would also result in positive results where other ECPs are encountered.

Conceptually, we have encouraged the notion of increasing a decision-maker's span of attention through the reapportionment of routine cognitive responsibilities to intelligent agents. Implicit in this concept is the idea that small wins can be had by relaxing the typical constraints that are brought to a problem-solving endeavor. That is, complete solutions are usually sought for software support, environmental scanning, or SAR image interpretation. The problem solver wants a final answer. By relaxing this constraint, by looking for utility in partial solutions, small but important gains can be had. This is the motivation for caching knowledge.

Further, it may be expected that additional such problems, equally resistant to traditional expert system approaches, will be recognized with the passage of time and additional advances in artificial intelligence. To prematurely reject technical approaches to these problems, simply because they cannot be solved "completely" by the existing technologies, would be to overlook many potential advantages.

*Trademark or registered trademark of International Business Machines Corporation.

Cited references and notes

1. L. Porter, *McDonald and Denny, Inc.: The Keeps Project*, Harvard Case Series, No. 9-187-001 (1986).
2. F. Aguilar, *Scanning the Business Environment*, Macmillan Publishing Co., New York (1967).
3. D. C. Hambrick, "Specialization of Environmental Scanning Activities Among Upper Level Executives," *Journal of Management Studies* **18**, No. 3, 299–320 (1981).
4. P. T. Terry, "Mechanisms for Environmental Scanning," *Long Range Planning* **10**, No. 3, 2–9 (June 1977).
5. H. I. Ansoff, "Managing Strategic Surprise by Response to Weak Signals," *California Management Review* **18**, No. 2, 36–48 (Winter 1975).
6. G. S. Elofson and B. R. Konsynski, "Supporting Knowledge Sharing in Environmental Scanning Units," *Proceedings of the 23rd Hawaii International Conference on Systems Sciences* (January 1990).
7. *SAR Tactical Interpretation and Reporting System (STIRS), Final Design Study*, Science Applications International, Contract No. DACA76-84-C-0001.
8. A. Newell, "Production Systems: Models of Control Structures," *Visual Information Processing*, W. G. Chase, Editor, Academic Press, New York (1973), Chapter 10, pp. 463–526.
9. R. J. Heuer, Jr., *Quantitative Approaches to Political Intelligence: The CIA Experience*, Westview Press, Boulder, CO (1978).
10. Personnel communication, Mike Scirola, Stanford Research Institute (SRI), Stanford, CA.
11. Personal communication, Cathy Rock, General Electric, Hackensack, NJ.
12. D. B. Montgomery and C. B. Weinberg, "Towards Strategic Intelligence Systems," *Journal of Marketing*, 41–52 (Fall 1979).
13. F. F. Neubauer and N. Soloman, "A Managerial Approach to Environmental Assessment," *Long Range Planning* **10**, No. 2, 13–20 (April 1977).
14. For more information about blackboard architectures, see References 15, 16, 17, 18, and 19.
15. H. P. Nii, "Blackboard Systems: Part One," *AI Magazine* **7**, No. 2, 38–64 (1986).
16. H. P. Nii, "Blackboard Systems: Part Two," *AI Magazine* **7**, No. 3, 82–106 (1986).
17. L. D. Erman, F. Hayes-Roth, V. R. Lesser, and D. R. Reddy, "The Hearsay-II Speech Understanding System: Integrating Knowledge to Resolve Uncertainty," *Computing Surveys* **12**, 213–253 (1980).
18. *Blackboard Systems*, R. Englemore and T. Morgan, Editors, Addison-Wesley Publishing Co., Reading, MA (1988).
19. D. D. Corkill, K. Q. Gallagher, and K. E. Murray, "GBB: A Generic Blackboard Development System," *Proceedings of the Fifth National Conference on Artificial Intelligence*, Philadelphia, PA (1986), pp. 1008–1014.
20. M. Lebowitz, "Generalization from Natural Language Text," *Cognitive Science* **7**, 1–40 (1983).
21. M. Lebowitz, "Experiments with Incremental Concept Formation: UNIMEM," *Machine Learning* **2**, No. 2, 103–137 (September 1987).
22. M. Lebowitz, "Classifying Numeric Information from Generalization," *Cognitive Science* **9**, 285–308 (1985).
23. D. Fischer, "Concept Learning Algorithms," *Artificial Intelligence and Statistics*, Wiley and Associates, New York (1986), pp. 139–171.
24. B. Hayes-Roth, "A Blackboard Architecture for Control," *Artificial Intelligence* **26**, 251–321 (1985).
25. B. Hayes-Roth and F. Hayes-Roth, "A Cognitive Model of Planning," *Cognitive Science* **3**, 275–310 (1979).
26. From a methodological viewpoint, archival case studies have been performed and accepted for some time. For example, Rosner[27] used hospital records to evaluate the willingness of the hospital to innovate. And some classic studies of organizational behavior have depended heavily on records, such as in Reference 28.
27. M. M. Rosner, "Administrative Controls and Innovation," *Behavioral Science* **13**, 36–43 (1968).
28. P. Selznik, *TVA and the Grass Roots*, University of California Press, Berkeley, CA (1949).
29. As early as 1973 nearly half of the U.S.-based multinational corporations were regularly monitoring the political climate of foreign countries as part of their day-to-day operations.[30]
30. D. Zink, *The Political Risks for Multinational Enterprises*

in Developing Countries, Praeger Publishers, New York (1973).
31. J. Weydenthal, *The Polish Drama: 1980–82*, Lexington Books, Lexington, MA (1983).
32. T. R. Gurr, *The Conditions of Civil Violence; First Tests of a Causal Model*, Research Monograph No. 28, Princeton University, Center of International Studies, Princeton, NJ (1967).
33. To illustrate the opportunistic search characteristics of the hybrid blackboard, a slightly larger set of knowledge sources was needed than was available in the previous example.

Accepted for publication August 15, 1994.

Greg Elofson *Emory University, Emory Business School, Decision and Information Analysis, 1602 Mizell Drive, Atlanta, Georgia 30322.* Dr. Elofson is an assistant professor at the Emory University School of Business. His research interests include the use and effects of intelligent information technologies in organization design, environmental scanning, and electronic markets. He has published many articles and proceedings in scholarly journals such as the *Journal of Management Information Systems*, *Decision Support Systems*, *Expert Systems with Applications*, and *Information and Software Technology*. Dr. Elofson has been a staff scientist in the artificial architectures group of Science Applications International, a faculty member with the University of Miami School of Business, and a consultant with Andersen Consulting. He received a Ph.D. and M.S. in management information systems from the University of Arizona and an A.B. from Occidental College.

Reprint Order No. G321-5559.