SEMANTIC NETWORK REPRESENTATIONS
IN RULE-BASED INFERENCE SYSTEMS

TECHNICAL NOTE 136

MAY 1977

By: Richard O. Duda
      Peter E. Hart
      Nils J. Nilsson
      Georgia L. Sutherland
      Artificial Intelligence Center

SRI Project 5821

ABSTRACT

Rule-based inference systems allow judgmental knowledge about a specific problem domain to be represented as a collection of discrete rules. Each rule states that if certain premises are known, then certain conclusions can be inferred. An important design issue concerns the representational form for the premises and conclusions of the rules. We describe a rule-based system that uses a partitioned semantic network representation for the premises and conclusions.

Several advantages can be cited for the semantic network representation. The most important of these concern the ability to represent subset and element taxonomic information, the ability to include the same relation in several different premises and conclusions, and the potential for smooth interface with natural language subsystems. This representation is being used in a system currently under development at SRI to aid a geologist in the evaluation of the mineral potential of exploration sites. The principles behind this system and its current implementation are described in the paper.

CONTENTS

ACKNOWLEDGMENTS

I   Introduction

The use of production rules for knowledge representation has led to systems that have achieved impressive levels of performance in their particular domains [1, 12].   The advantages of this approach stem from the fact that the representation is modular and declarative. This provides conceptual clarity, encourages incremental development, and makes the knowledge base directly accessible, so that, for example, the program can explain its own reasoning processes.

Davis and King have observed that the production system formalism is more appropriate for some domains than others, being particularly natural when the knowledge can be expressed as a more or less independent set of 'recognize-act' pairs [3]. In particular, it might be natural to represent judgmental knowledge by a set of production rules, but unnatural to use the same mechanism to represent other relevant knowledge, such as taxonomic (subset/element) relations among objects in the domain.

In this paper we describe a way to use semantic network representations in rule-based inference systems.   This combination allows a designer to retain the desirable modularity of a rule-based approach, while permitting an explicit, structured description of the

semantics of the problem domain. Since semantic nets are among the leading internal representations used in computational linguistics, their use should also simplify the development of a natural language interface between the system and its users.

Throughout this paper we shall use examples drawn from a geological consultant system currently being developed at SRI. This system is intended to help geologists in evaluating the mineral potential of exploration sites. Our approach to designing the system has been influenced by various developments in artificial intelligence. Shortliffe's MYCIN program for diagnosing and treating bacterial infections has been the dominant influence, primarily through its use of production rules to represent judgmental knowledge, and its inclusion of formal mechanisms for handling uncertainty [10, 11, 12]. We share with Pople's INTERNIST (nee DIALOG) program an exploitation of taxonomic structures, a concern for the use of volunteered information (event-driven, bottom-up, forward-chaining, or antecedent reasoning), and a need for more flexible control strategies [9]. There are also parallels between our work and that of Trigoboff, who has developed different but related methods for propagating measures of uncertainty through a semantic network [13]. Finally, we have been influenced by the generality and power of Hendrix's partitioned semantic networks [6, 7], and have employed this approach in our system.

The following sections in this paper present the basic principles behind our system. Section II briefly reviews the rudiments of

2

partitioned semantic networks, and shows how individual rules are represented. Section III describes our use of Bayesian procedures to propagate information through a network of rules. Section IV describes our current implementation, and gives an example of the operation of the existing system. Finally, Section V discusses a number of unresolved design issues, whose presence should forewarn the reader that the ideas presented are not yet fully mature.

II   Partitioned Semantic Networks

A.   Background

We are using a semantic-network formalism (proposed by Hendrix) that uses 'partitions' as a way of grouping parts of the net into meaningful units [6, 7].   Partitioned semantic networks possess all of the expressive power of predicate calculus.   Quantification, implication, negation, disjunction, and conjunction are easily represented in partitioned semantic networks.   As compared with most computer implementations of predicate calculus representations, however, semantic networks have the additional advantages of two-way indexing, direct set-subset-element representations, classification of variables according to type, and the ability to represent modal statements.   In this section we give a brief overview of partitioned semantic nets and how we use them.

B.   Elements of Semantic Networks

A semantic network consists of nodes linked together by arcs.   We distinguish two main types of nodes: object nodes and relation nodes. These two types play roughly the same role as do terms and predicates, respectively, in the predicate calculus.   For example, the statement "Entity-1 is composed of rhyolite," could be represented by the net

structure shown in Figure 1. Here the 'composed-of' relation node has
two arguments, entity and value. In this net, these arguments are
filled by the object nodes, ENTITY-1 and RHYOLITE, respectively. Arcs
in the network are used to connect relation nodes to other nodes; they
are labeled with the name of the argument they represent.
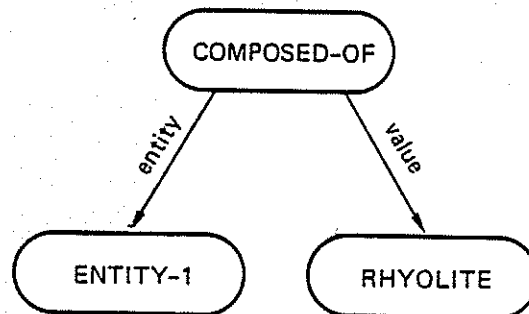
FIGURE 1   REPRESENTATION OF THE STATEMENT
            "ENTITY-1 IS COMPOSED OF RHYOLITE"

    Some relations, such as set membership, are so common that as a
shorthand we represent them by special arcs instead of by their own
relation nodes and arguments. The net structure shown in Figure 2, with
its 's' and 'e' arcs, includes a representation of the statement "Galena
is an element of the lead sulfides which is a subset of the sulfide
minerals which is a subset of the minerals." Net structures of this
sort are obviously useful in representing taxonomic hierarchies.
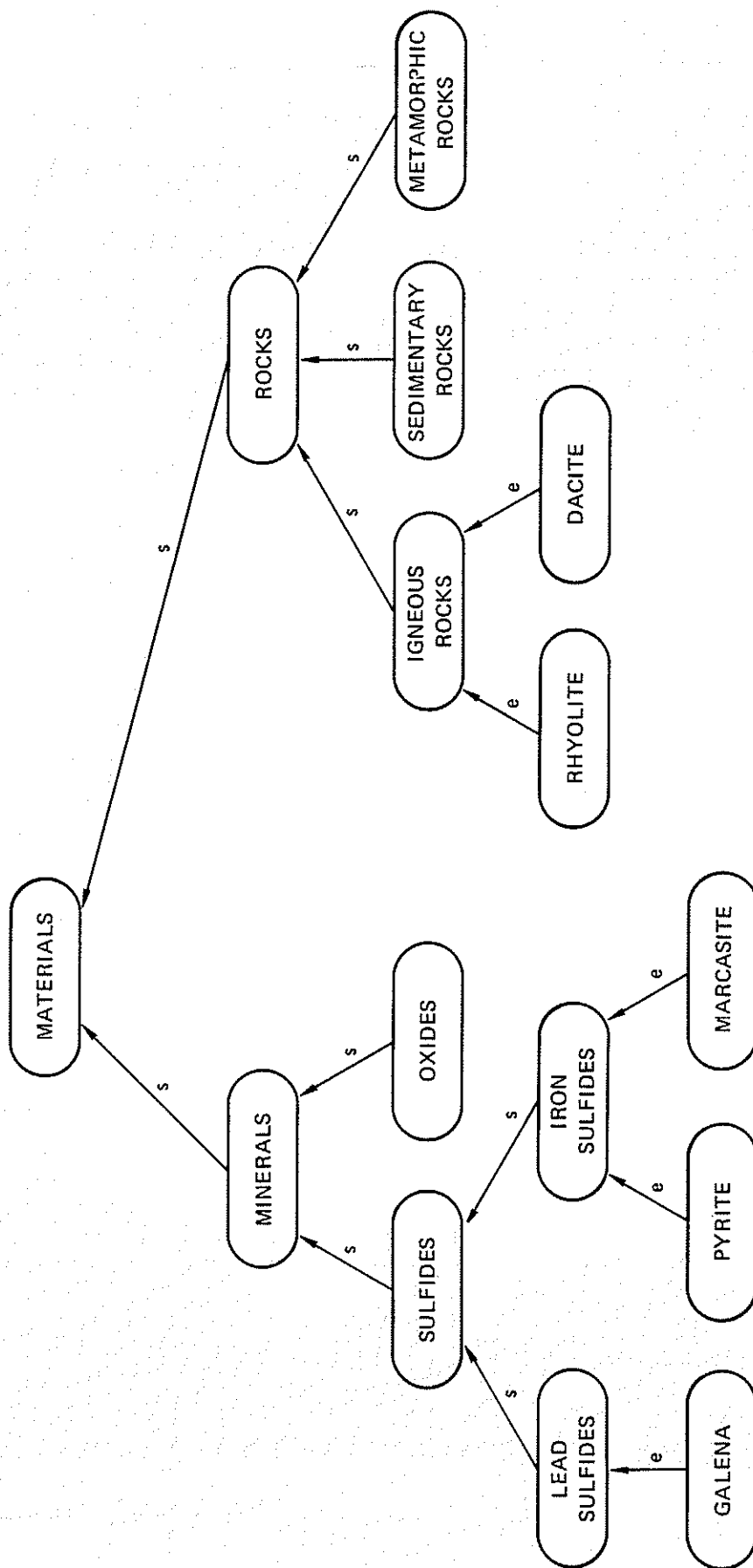
FIGURE 2   A TAXONOMY OF MATERIALS

Each particular instance of a relation is an element of the set of all relations of that type. Thus, the net shown in Figure 1 depicts just one instance of all 'composed-of' relations. If we labeled this particular instance by C1, we would have the structure shown in Figure 3.
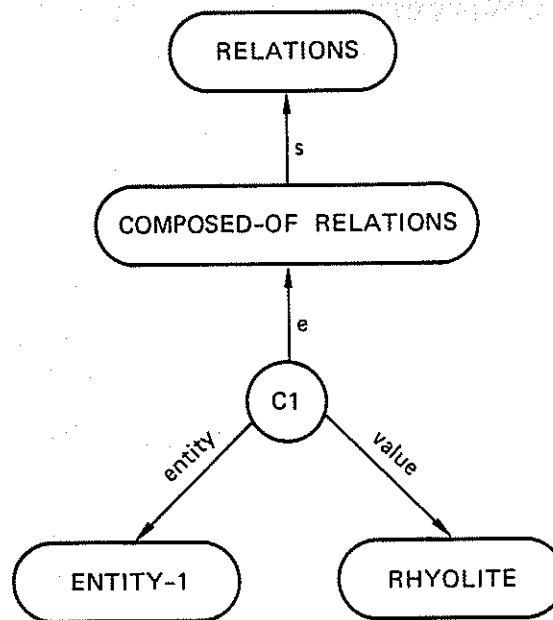
FIGURE 3   C1 AS AN INSTANCE OF A 'COMPOSED-OF' RELATION

The set of all 'composed-of' relations forms a relation family. Our present system uses several different types of relation families to express concepts such as composition, form, physical location, distance, and special properties. Each relation family is represented in the network by a structure called a 'delineation' in which the types of the arguments are explicitly shown. For example, the delineation for the 'composed-of' relation is shown in part in Figure 4.

7
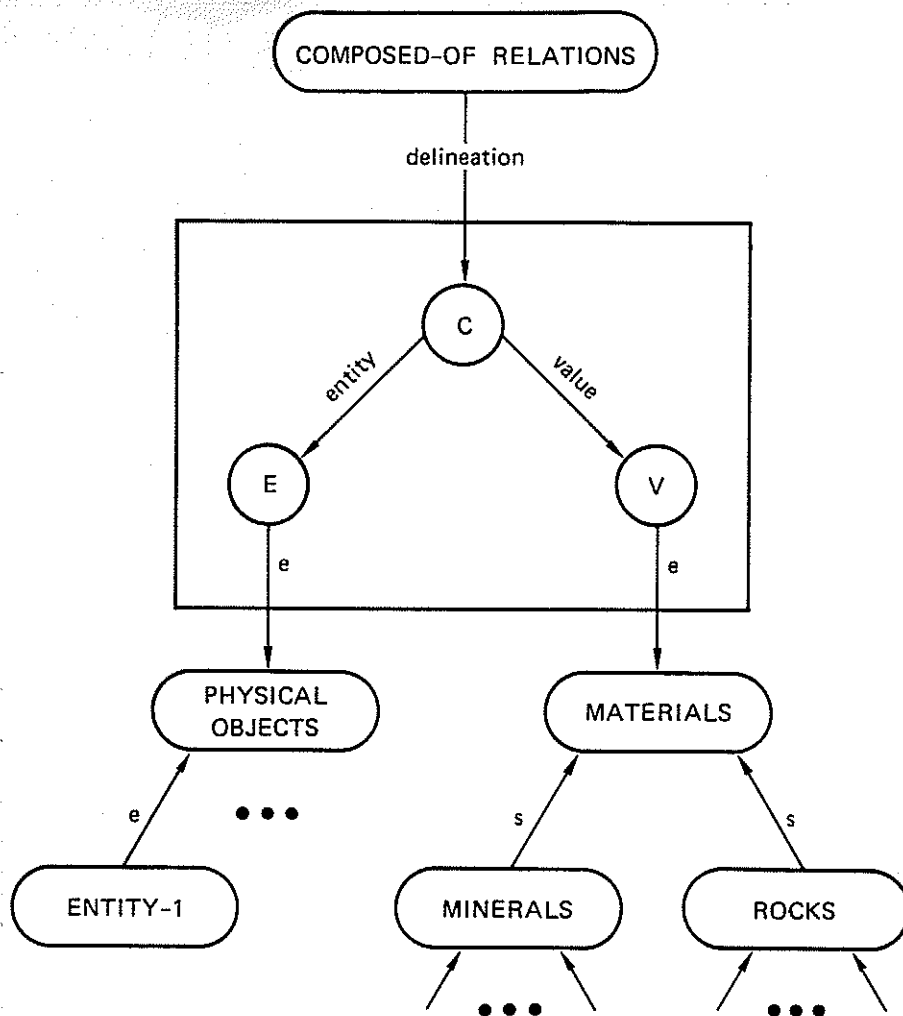
FIGURE 4   DELINEATION FOR THE 'COMPOSED-OF' RELATION

This structure shows our first use of partitions.   The structure

within the partition (box) delineates the composed-of relation, saying

that it has two arguments.   One argument is an 'entity' E drawn from the

set PHYSICAL OBJECTS (of which ENTITY-1 is an example); the other

argument is a 'value' V drawn from the set MATERIALS.   The partition

serves to isolate the delineation, which is really like a definition,

from other actual instances of relations in the network.   Structures

8

within a partition are treated specially and do not have the same 'existential' character as unpartitioned nodes. In particular, we use nodes within partitions as variables which can be bound in various ways to constants outside the partition.

A complete network characterization of a large body of knowledge would interconnect many structures of the type we have mentioned. The separate occurrence of several relation nodes in a partition represents their logical conjunction. Partitions are also used to isolate the components of disjunctions, implications, and negations. For details on these and related topics, such as quantification, see [6, 7].

C.   Rules

Much judgmental knowledge about mineral exploration can be represented in the form of 'rules' such as:

"Limonite casts suggest the probable presence of pyrite"

or

"Barite overlying sulfides suggests the possible presence of a massive sulfide deposit."

These rules are in the form of simple implicational statements such as E1 & E2 & ... & EN => H, where the Ei are individual pieces of evidence and H is a hypothesis suggested by the evidence. Seldom can any of the implications be made with absolute certainty; usually the English versions of the rules contain phrases such as "strongly suggest" or "is mildly important for."

9

To represent rules of this sort in our semantic net formalism, we
must be able to represent the individual pieces of evidence, the
hypothesis, and the implication and its strength. To represent the
implication, we use separate partitions for the antecedent and the
consequent. Each rule is represented by a structure having the form
shown in Figure 5. The individual partitions for antecedent and
consequent contain the appropriate network structures. A property list
attached to the rule node includes a measure of the strength of the
implication. (This actually requires the specification of two numbers,
as will be discussed in the next section.)
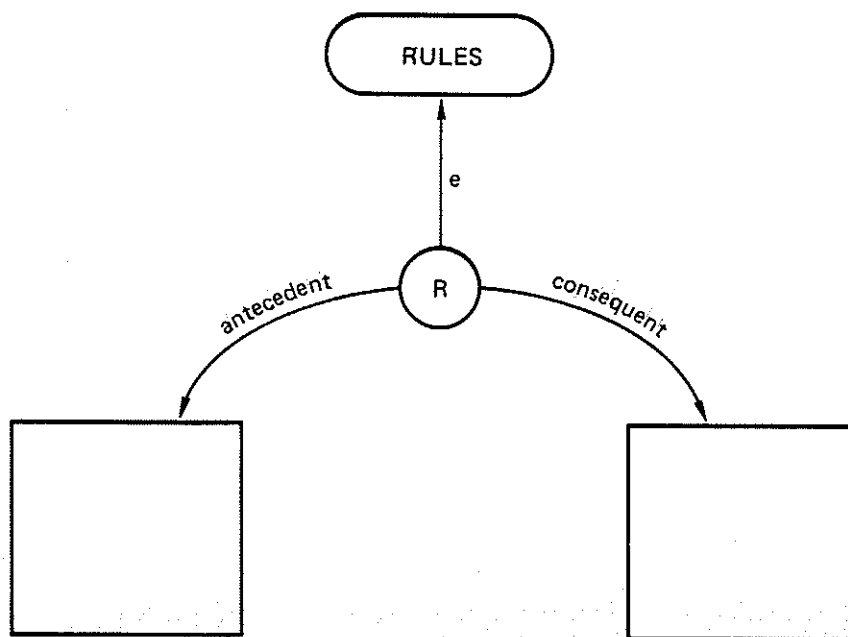
FIGURE 5   GENERAL FORM OF THE REPRESENTATION OF A RULE

Using this formalism, the rule "Barite overlying sulfides suggests
the possible presence of a massive sulfide deposit" is represented as
shown in Figure 6. A literal English statement of the antecedent might

10
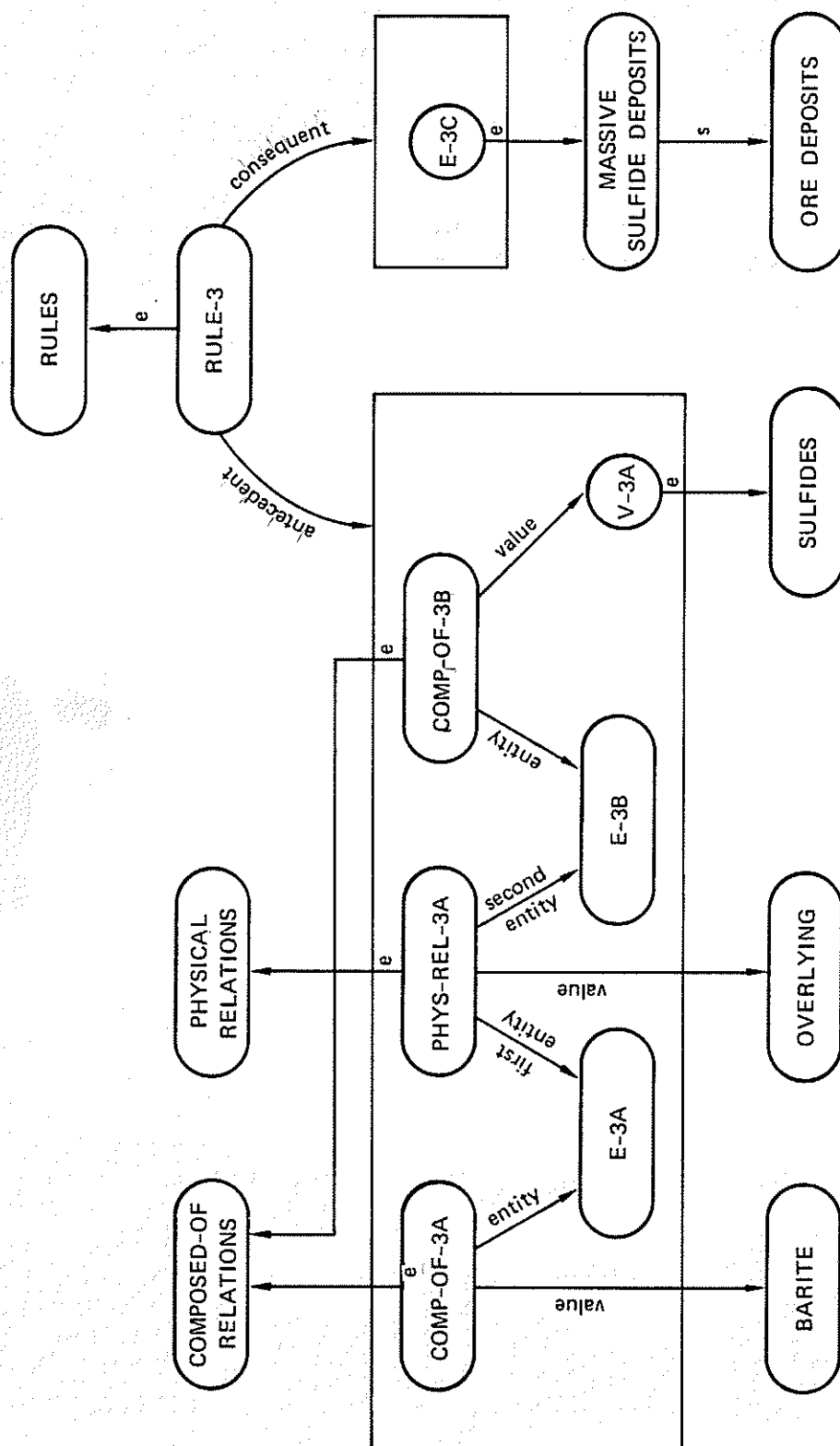
FIGURE 6   REPRESENTATION OF THE RULE "BARITE OVERLYING SULFIDES SUGGESTS
THE POSSIBLE PRESENCE OF A MASSIVE SULFIDE DEPOSIT"

11

be something like "There is some entity, which we call E-3A internally, that participates in an overlying relation (PHYS-REL-3A) with some other entity, E-3B. Furthermore, E-3A is composed of barite, and E-3B is composed of some material, V-3A, that is a member of the sulfides." Note that the nodes for BARITE and SULFIDES lie outside of the partitions; these concepts 'have existence' in their own right, independent of the example rule.

III   Information Propagation in Inference Networks

A.   Inference Networks

The production rules used to represent judgmental knowledge typically are not independent, but link together in various ways to form what we call an inference network. Explicit links occur when the hypothesis (consequent) of one rule is the evidence (antecedent) for another. Several examples of this appear in Figure 7, which is a simplified representation of seven of the thirty-four rules currently used to draw conclusions about a possible Kuroko-type massive sulfide deposit. For example, the observation of bleached rocks would suggest the possibility of a reduction process (Rule 27), which in turn suggests the existence of clay minerals (Rule 28), which are often associated with a Kuroko-type massive sulfide deposit (Rule 24).

It is not necessary that all parts of a consequent and a related antecedent match. For example, rhyolite appears as part of the antecedent for Rule 14: "Galena, sphalerite, or chalcopyrite filled cracks in rhyolite or dacite is very suggestive of a massive sulfide deposit." Rules can also be linked implicitly through so-called 'e-s' (element-subset) chains. For example, suppose that an entity composed of galena is observed. Since galena is an element of the lead sulfides which in turn is a subset of the sulfide minerals, this observation is
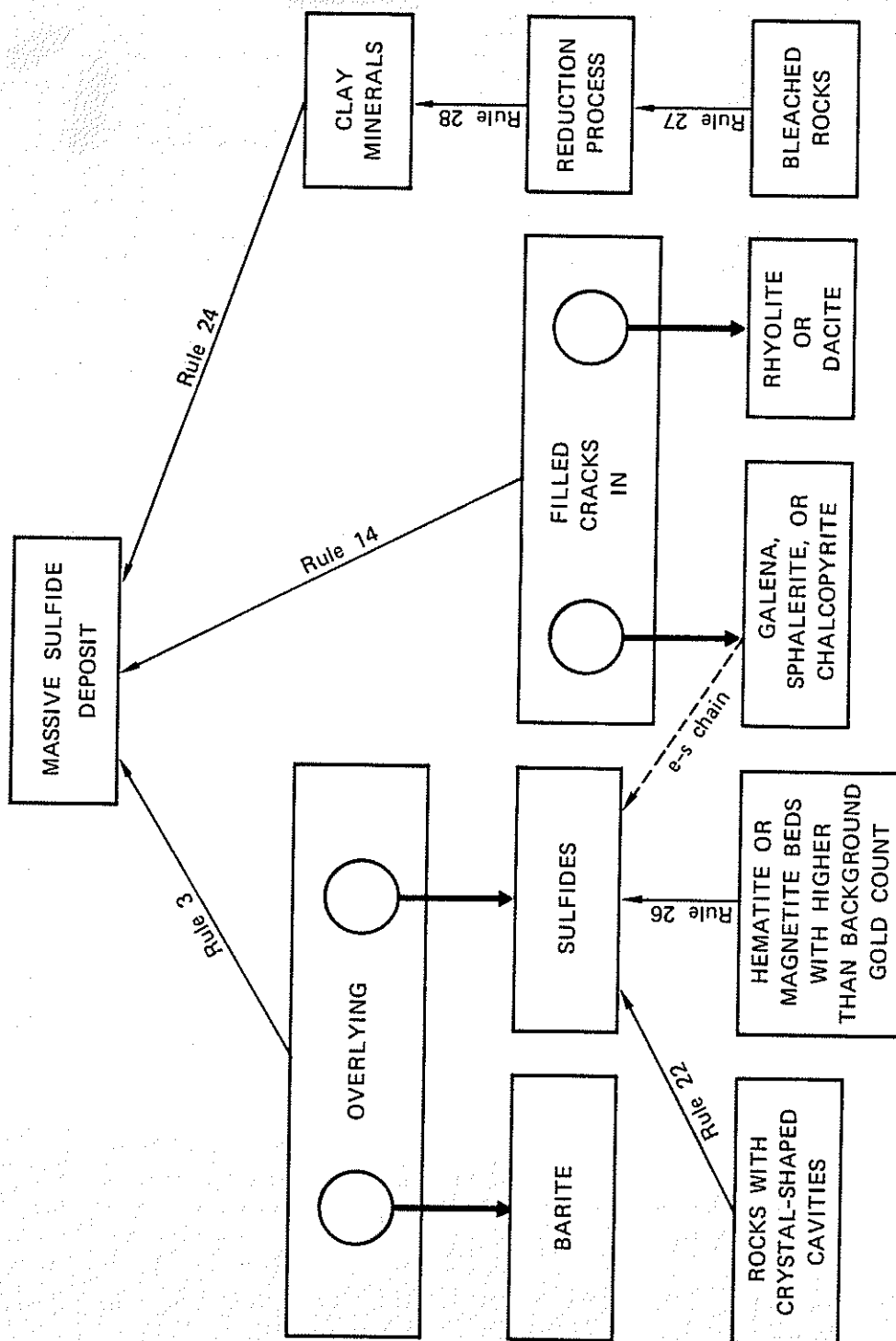
13

FIGURE 7   SIMPLIFIED REPRESENTATION OF PART OF THE MASSIVE SULFIDE DEPOSIT
            INFERENCE NET

14

relevant to Rule 3: "Barite overlying sulfides is mildly suggestive of a massive sulfide deposit." Thus, information can propagate through the network in two ways, either directly through chained rules, or indirectly through e-s chains.

B.   Variables

In the general case, the links shown between rules in Figure 7 should be thought of as potential rather than actual links. The object and relation nodes within any partition are variables, and can be bound in various ways. Thus, for example, the particular entity composed of galena used in Rule 14 might be different from the physical entity composed of galena used to reach sulfides in Rule 3; one might satisfy one set of relations, the other another.

As a result, the rules cannot be linked statically, but must be connected by pattern matching at run time. While this situation is simplified by the fact that the potential matches are relatively few in number and can be precomputed, it still gives rise to a number of complications. As a temporary expedient, our current implementation tacitly assumes that any variable can be bound in only one way, so that, for example, only one entity composed of galena would be allowed. However, the representation used is general, permitting the unrestricted use of variables that ultimately will be needed.

C.   Uncertainty

To account for uncertainty in both the evidence and the rules, we associate a subjective probability with every relation and i pair of strength values with every rule.   Thus, rather than saying definitely that "Entity-1 is composed of rhyolite," we would say that "Entity-1 is composed of rhyolite with probability P1," and would associate P1 with the composed-of relation, rather than with the entity or the value. The interpretation is subjective, meaning that we interpret P1 as a measure of degree of belief rather than as the long-run relative frequency of occurrence [5].

In general, the antecedent of a rule is a logical function of the relations involved. Rule 3 illustrates the typical case of logical conjunction; for the antecedent to be true, we must have an entity, E-3A, composed of barite, and an entity, E-3B, composed of one of the sulfide minerals, and an overlying relation between E-3A and E-3B (see Figure 6).   To compute the probability of the antecedent we make recursive use of Zadeh's fuzzy-set formulas [14]:

$$Pr(A \ \& \ B) = \min \ \{Pr(A), \ Pr(B)\}$$
$$Pr(A \ v \ B) = \max \ \{Pr(A), \ Pr(B)\}$$
$$Pr(\tilde{\ }A) = 1 - Pr(A).$$

Given the probability associated with an antecedent, we use a form of Bayes' rule, modified to accommodate possible inconsistencies between subjectively determined probabilities, to determine the probability of

the consequent. Unlike the mechanism used in MYCIN, this procedure does not require separate treatment of belief and disbelief, nor does it require the attainment of a given level of certainty before a rule can be used. Any time the probability of the antecedent changes, the rule can be 'applied' again to update the probability of the consequent. This feature is particularly valuable if the user modifies previously given information, thereby forcing a reevaluation of the situation.

A derivation and justification for our procedure is given in [4]; for completeness, we summarize the final results briefly. Let E denote the antecedent and H the consequent of a rule. Let O(E) be the prior odds on E, and let O(H) be the prior odds on H, where odds O are uniquely related to probabilities P by

$$O = \frac{P}{1 - P} \quad .$$

Let E' denote all of the evidence we have for believing E to be true (or false). Through the rule, this evidence affects the posterior odds on H, O(H|E'). Of all of the possible situations regarding our knowledge of the truth of E, three are particularly interesting: E known true, E known false, and E believed true with the prior probability P(E). The last case is trivial, in that it leaves the odds on H unchanged at O(H). For the other two cases, Bayes' rule yields

$$O(H|E) = \lambda * O(H)$$

and

$$O(H|{\sim}E) = \bar{\lambda} * O(H) \quad .$$

17

Here λ is the likelihood ratio for E true, and $\bar{\lambda}$ is the likelihood ratio for E false. We sometimes say that λ measures the degree of sufficiency, since a very large value for λ means that E is sufficient for H. Similarly, $\bar{\lambda}$ measures the degree of necessity, since a very small value for $\bar{\lambda}$ means that E is necessary for H. The values of λ and $\bar{\lambda}$ taken together define the strength of the rule.
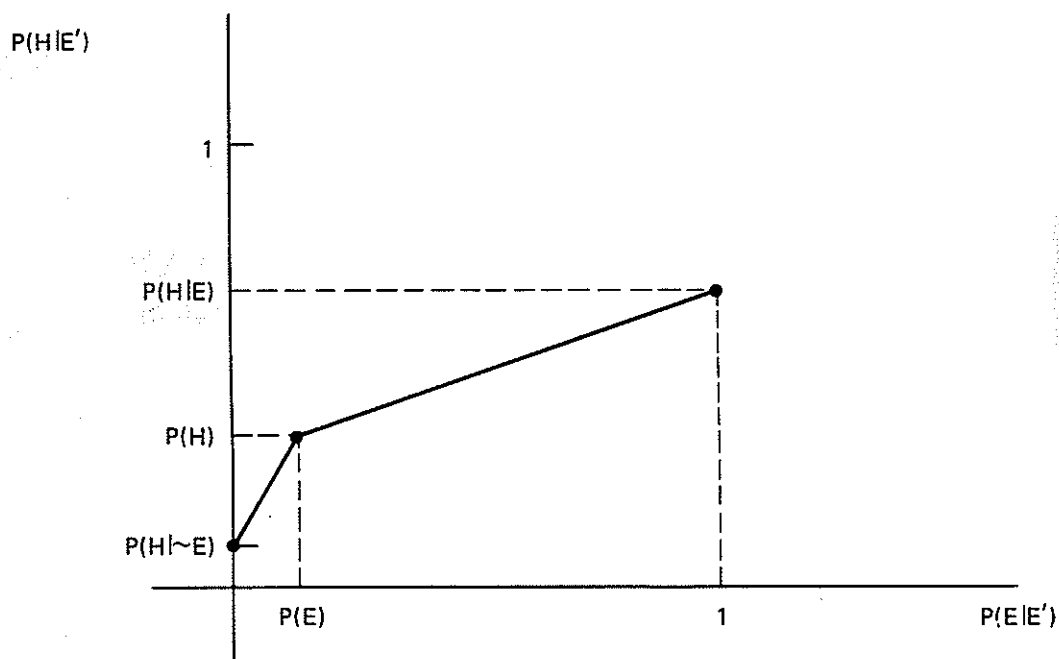

FIGURE 8   THE FUNCTION USED TO DETERMINE THE POSTERIOR
PROBABILITY OF THE CONSEQUENT FOR A SINGLE
RULE

For the general case, let P(E|E') denote our present degree of belief in E based on E'. We compute P(H|E') as the piecewise-linear function of P(E|E') shown in Figure 8, which amounts to interpolating linearly between the three special cases just described. This in turn

18

defines the desired posterior odds O(H|E'), and an effective likelihood ratio λ' defined by

$$\lambda' = \frac{O(H|E')}{O(H)} \quad .$$

An effective likelihood ratio is associated with every rule. Unlike $\bar{\lambda}$ and $\bar{\lambda}$, it varies in value as information is gained, starting initially at unity (indifference) and approaching either λ if E is determined to be true, or $\bar{\lambda}$ if E is determined to be false. If several rules all bear on the same hypothesis, the effective likelihood ratio provides the mechanism for combining their effects. Assuming that the separate rules bear on H independently, we compute the posterior odds O(H|E') by multiplying the prior odds O(H) by the product of all of the incoming effective likelihood ratios. Repeated application of this computationally simple procedure allows the effects of the alteration of any probability to propagate through the network.

IV   Current Implementation

The current implementation of our system is called PROSPECTOR. It
is coded in INTERLISP, and makes direct use of Hendrix and Slocum's
semantic net package.   In addition to providing the data structures
described in Section II, PROSPECTOR contains an executive program and
facilities (or hooks for facilities) for the following tasks:

   (1) Accepting volunteered information

   (2) Propagating consequences

   (3) Determining needed information

   (4) Asking Questions

   (5) Answering questions

   (6) Augmenting the knowledge base.

While some of these facilities are fairly sophisticated, the
current implementation is still rather new, and little attention has
been devoted to the important topic of handling English input and
output. This means that the interactions involved in Topics 1, 4, and 5
are currently rather clumsy. In particular, to volunteer information to
the current system, one must know the internal net representations and
evaluate the appropriate LISP functions.   A more convenient but still
rudimentary question-answering facility based on Hendrix's LIFER package

allows the user to use constrained English to ask certain kinds of questions about either the taxonomy or the rules. Answering questions posed by the system is straightforward, although even here no attempt has been made to have PROSPECTOR pose the questions in graceful English. Thus, the human engineering features so important for computer-naive users are currently minimal.

The main parts of PROSPECTOR are domain independent. Domain-specific information is kept in separate taxonomy and rule files. Special facilities are available for reading and writing taxonomy files, acquiring taxonomic information interactively, and constructing semantic net structure corresponding to a taxonomy. A formal description language has been designed for representing the rules in the rule files. Facilities are also available for reading and writing the rule files, and automatically constructing the corresponding semantic-net representations. The present rule base contains 34 rules for Kuroko-type massive sulfide deposits and 16 rules for Mississippi-Valley-type lead/zinc deposits. These rules have been entered manually, although we have experimented with and recognize the ultimate usefulness of automatic rule acquisition [2].

Whenever new information is entered, the procedures described in Section III are used to propagate the consequences. In particular, this will usually affect the probabilities associated with certain 'top-level' nodes that correspond to important hypotheses. After the user has finished volunteering information and propagation has terminated,

the system must determine what additional information will be most effective in establishing the top-level hypotheses with greater certainty. This is the so-called control problem, and it raises many unsolved problems.

Our current strategy resembles the depth-first strategy used by MYCIN, with two important exceptions: we allow for volunteered information at any time, and we use a simple evaluation function for dynamic rule ordering. The hypothesis selected initially is the top-level hypothesis having the highest current probability. Every untried rule having that hypothesis as a consequent is scored according to the function

$$\left| \log \frac{\lambda}{\lambda_t} \right| P(E|E') + \left| \log \frac{\lambda'}{\lambda} \right| [1 - P(E|E')] \quad .$$

and the highest scoring rule is selected. If its antecedent is askable (and has not been asked about before), the user is asked about it. If the user can provide even a partial answer, those results are propagated and the rules are rescored. However, if the antecedent is not askable, or if the user has no information to offer, that antecedent becomes the new hypothesis, and the same procedure is applied recursively.

The particular scoring function used computes the expected change in $| \log O(H|E') |$ under the assumption that E will be found to be true with probability $P(E|E')$ or false with probability $1 - P(E|E')$. When E is unlikely a priori, this criterion initially favors rules with small

values of $\overline{\lambda}$ (necessary conditions), and as E becomes more likely it favors rules with large values of λ (sufficient conditions). However, low scores are assigned if the truth or falsity of E becomes well established, so that there is little to gain in trying to make knowledge about the antecedent more certain.

The following example, edited for the sake of brevity, illustrates the operation of the present system. The interaction was designed to display the various aspects of the program, including favorable and unfavorable evidence, linked rules, subquestions, etc, and does not correspond to an actual exploration problem.

. . . .

PROSPECTOR, an experimental computer based consulting system, is designed for use both in searching for ore deposits and in evaluating the mineral potential of large geographic areas. Being in the early research stages, PROSPECTOR knows only one exploration model, specifically, Park's model for Kuroko-type massive sulfide deposits.

In using PROSPECTOR, you will be asked questions, or you may volunteer information about a particular mineral prospect. The program will use your information and the rules it contains to draw conclusions about possible ore deposits on the prospect.

Indicate your answers as follows:

```
 2 -- VIRTUALLY CERTAINLY PRESENT
 1 -- PROBABLY PRESENT
 0 -- NO OPINION ONE WAY OR THE OTHER
-1 -- PROBABLY ABSENT
-2 -- VIRTUALLY CERTAINLY ABSENT
```

23

................Program execution is now starting....................

Do you want to volunteer any evidence? YES

A.   Space name of evidence: SPACE-25L

        New likelihood of (Widespread Igneous Rocks) : 2

B.   Space name of evidence: NIL

Proceeding to establish the likelihood of (Massive Sulfide Deposit):

1.   Do you have anything to say about (Volcanic province and major

fault zone) ? 1

2.   Do you have anything to say about (Mineralization) ? 0

3.   Do you have anything to say about (Near shore depositional

sequences of andesites, rhyolites or dacites) ? 0

        4.   Do you have anything to say about (Pillow structures) ? 2

5.   Do you have anything to say about (Breccia) ? NO

6.   Do you have anything to say about (Rhyolite or dacite plug) ? YES

     6a.   Have you anything to say about an entity with the composition

     of (OR RHYOLITE DACITE) ? 1

     This entity will hereafter be referred to as ENTITY-1

     6b.   Have you anything to say about whether ENTITY-1 has the form

     of PLUG ? 2

7. Do you have anything to say about (Galena, sphalerite, or chalcopyrite filled cracks in rhyolite or dacite) ? YES

   7a. Have you anything to say about an entity with the composition of (OR GALENA SPHALERITE CHALCOPYRITE) ? 2
   This entity will hereafter be referred to as ENTITY-2

   7b. Have you anything to say about whether ENTITY-1 has the property of CONTAINING-CRACKS ? 2

   7c. Have you anything to say about whether ENTITY-2 is CONTAINED-IN ENTITY-1 ? 1
                   .
                   .
                   .

10. Do you have anything to say about (Olivine or alkaline andesite) ? 2

11. Do you have anything to say about (Calc-alkaline andesite) ? NO
                   .
                   .
                   .

14. Do you have anything to say about (Clay Minerals) ? 0

       15. Do you have anything to say about (Reduction process) ? 0

           16. Do you have anything to say about (Bleaching of rocks) ? YES

16a. Have you anything to say about an entity

with the composition of ROCKS ? 2

This entity will hereafter be referred to as

ENTITY-4

16b. Have you anything to say about whether

ENTITY-4 has the property of BLEACHED ? 1
.
.
.

22. Do you have anything to say about (Prospect within a few

miles of known MSD) ? NO

All the rules which bear on (Massive Sulfide Deposit) have been

considered. Current likelihood of (Massive Sulfide Deposit) is .01465.

.............................DONE...................................

This example represents a typical interaction between the current

program and a user. After the initial description and instructions have

been given, the system permits the user to input relevant knowledge he

may wish the program to use.   Since language understanding is absent,

internal names must be used. In this run, the user asserts the

existence of widespread igneous rocks by knowing   that SPACE-25L

represents this concept; the system responds with an appropriate

translation, and then accepts the probability assignment.

After the volunteered evidence phase, the system's control strategy

selects the most likely top-level hypothesis, scores all incoming rules, and selects the highest scoring rule in order to obtain evidence about the hypothesis. Questions 1, 2, 3, 5, 6, 7, 10, 11, 14 and 22 all represent rules which give evidence for the top-level hypothesis. Questions 4, 15 and 16 represent rules at a deeper level in the net. Questions 6a, 6b, 7a, 7b, 7c, 16a and 16b are subquestions which establish the likelihoods of the different relations which make up an antecedent.

The user has a variety of possible responses to any question. A 'YES' answer causes the system to pursue the question further, whereas a 'NO' answer terminates interest in that evidence. The answer '0' indicates "no opinion" or "don't know", and causes the system to pursue the question if it has deeper rules which can be used to infer the desired evidence. A non-zero numerical answer is used directly to assign probabilities and to propagate inferences through the net.

The propagations are not shown in the example, but each non-zero numerical response triggered a propagation, many of which had an effect on the likelihood of the top hypothesis. The sequence of changes to the top hypothesis' probability value was .001 -> .003985 -> .004312 -> .03241 -> .711 -> .002968 -> .01465. In particular, the response to Question 7 was highly favorable, while the response to Question 10 was highly unfavorable for a Kuroko-type massive sulfide deposit. The numerical value of the final probability assignment indicates more than the prior likelihood of a deposit, though, of course, a final evaluation requires more than this single number.

27

V   Discussion

As the example of the preceeding section showed, we have made some
progress in developing a system which both represents   and uses
judgmental knowledge about a specific problem domain.   Most of the
remaining technical problems are shared with other production-system
approaches to consultant systems, and the most important of these
problems deserve at least brief mention.

A major problem concerns the acceptance of volunteered information,
which could ultimately include diagrams and maps as well as   text. Our
present procedure, which requires the user to know the internal naming
conventions, is obviously a temporary expedient, and the use of
unrestricted natural language is not technically feasible. Our decision
to use semantic net representations was at least partly motivated by the
hope for fairly flexible English input/output, but the development of
such an interface is a major unfinished task.

A second major research area concerns control strategies. Any
attempt to determine strategies that are optimal in a decision-theoretic
sense is probably computationally infeasible, particularly when the
networks are large and when unrestricted use of variables is allowed.
Pople's ideas on focusing are very attractive here [9], and further work
along these lines is needed.

28

Another important part of any consultant program is the explanation
system. As MYCIN has demonstrated, quite informative explanations of
conclusions can be produced merely by doing a backtrace of the applied
rules. However, the rules themselves are often the consequence of more
fundamental considerations (such as the effects of certain underlying
ore genesis processes in our geology example), and are not always
satisfactory as explanations. Kulikowski's work on causal networks is
evidently relevant in this regard [8], and the intermixed use of
procedural and declarative models is a natural and intriguing extension
of this kind of work.

The use of production rules to encode the judgmental knowledge and
the use of partitioned semantic networks to represent the structured
knowledge about a domain provide a general and potentially powerful
framework for building a consultant system. This combination has clear
advantages and has provided us with what we consider to be a strong base
for further development.

REFERENCES

1.   Buchanan, B. G., Sutherland, G., and Feigenbaum, E. Heuristic
     Dendral: A program for generating explanatory hypotheses in organic
     chemistry. Machine Intelligence 4, Meltzer, B., and Michie, D.
     (Eds.), Elsevier, New York, 1969, 209-254.

2.   Davis, R. Application of meta level knowledge to the construction,
     maintenance and use of large knowledge bases. Memo AIM-283,
     Artificial Intelligence Laboratory, Stanford University, Stanford,
     CA, July 1976, 291 pp.

3.   Davis, R., and King, J. An overview of production systems. Memo
     AIM-271, Artificial Intelligence Laboratory, Stanford University,
     Stanford, CA, October 1975, 38 pp.

4.   Duda, R. O., Hart, P. E., and Nilsson, N. J. Subjective Bayesian
     methods for rule-based inference systems. Proc. 1976 National
     Computer Conference (AFIPS Conference Proceedings), 45, 1976, 1075-
     1082.

5.   Fine, T. L. Theories of Probability. Academic Press, New York,
     1973, 263 pp.

6.   Hendrix, G. G. Expanding the utility of semantic networks through
     partitioning. Proc. Fourth International Joint Conference on
     Artificial Intelligence, Tbilisi, Georgia, USSR, 1975, 115-121.

7.   Hendrix, G. G. The representation of semantic knowledge. Chapter 5
     in Speech understanding research, Walker, D. (Ed.), Final Report,
     SRI Project 4762, Stanford Research Institute, Menlo Park, CA,
     October 1976.

8.   Kulikowski, C. A., and Weiss, S. M. Strategies of data base
     utilization in sequential pattern recognition. Proc. IEEE
     Conference on Decision and Control and 11th Symposium on Adaptive
     Processes, New York, 1972, 103-105.

9.   Pople, H. E., Jr., Myers, J. D., and Miller, R. A. DIALOG: A model
     of diagnostic logic for internal medicine. Proc. Fourth
     International Joint Conference on Artificial Intelligence, Tbilisi,
     Georgia, USSR, 1975, 848-855.

10. Shortliffe, E. H., et al. An artificial intelligence program to

30

advise physicians regarding antimicrobial therapy. Computers and Biomedical Research, 6, 1973, 544- 560.

11. Shortliffe, E. H. MYCIN: A rule-based computer program for advising physicians regarding antimicrobial therapy selection. Memo AIM-251, Artificial Intelligence Laboratory, Stanford University, Stanford, CA, 1974, 395 pp.

12. Shortliffe, E. H. Computer-Based Medical Consultations: MYCIN. Elsevier, New York, 1976, 264 pp.

13. Trigoboff, M. Propagation of information in a semantic net. Proc. AISB Summer Conference, Edinburgh, Scotland, 1976, 334-343.