Computers Math. Applic. Vol. 23, No. 6-9, pp. 733-745, 1992
Printed in Great Britain.
0097-4943/92 $5.00 + 0.00

QUEST: A MODEL OF QUESTION ANSWERING

ARTHUR C. GRAESSER
Department of Psychology, Memphis State University, Memphis, TN 38152 U.S.A.
SALLIE E. GORDON
Department of Psychology, University of Idaho, Moscow, ID 83843 U.S.A.
LAWRENCE E. BRAINERD
Department of Mathematical Sciences, Memphis State University, Memphis, TN 38152 U.S.A.

Abstract—QUEST is a computer model of question answering that simulates answers that adults produce when they answer open-class questions (e.g., why, how, what-if) and closed-class questions (e.g., is X true or false?). QUEST has four major procedural components: (1) question interpretation, (2) identification of relevant information sources, (3) pragmatics, and (4) convergence mechanisms. The procedures operate on information sources which are represented as conceptual graph structures. These structures contain goal/plan hierarchies, causal networks, taxonomic hierarchies, spatial region hierarchies, and other forms of knowledge. This article describes how knowledge is represented by QUEST's conceptual graph structures and how the procedural mechanisms operate on the knowledge structures during question answering. The primary focus is on convergence mechanisms, which identify the small subset of nodes in the information sources that serve as relevant answers to a particular question. An important convergence mechanism is the arc search procedures, which identify legal answers to the question by pursuing particular paths of arcs in each information source.

We have developed a computer model of human question answering, called QUEST. QUEST simulates the answers that people produce when they answer different types of questions, such as why, how, when, where, what-if, and yes/no verification questions. When QUEST answers a particular question, the model identifies relevant information sources and taps information within each source. Each information source is a package of world knowledge that is organized in the form of a "conceptual graph structure" containing nodes and relational arcs. The question answering (Q/A) procedures operate on these structures systematically, pursuing some paths of arcs, but not others, depending on the question category. The success of QUEST in simulating human question answering depends critically on an appropriate organization of world knowledge structures as well as an appropriate specification of the Q/A procedures that operate on the structures. The computational foundations of QUEST were inspired by models of question answering in artificial intelligence and computational linguistics [1-8]. In these models, text and world knowledge are organized as structured databases, such as semantic networks [9-11], conceptual dependency theory graphs [12], or conceptual graphs [13]. The Q/A procedures access these information sources and search through the structures systematically by traversing particular categories of arcs. Such models of question answering in AI or computational linguistics are regarded as computationally sufficient if they can generate all nodes from the information sources that are relevant to particular questions. QUEST is similar to these models in that it aspires to be a computationally sufficient model of question answering.

QUEST was intended to be a psychological model of question answering. It was therefore designed to be psychologically plausible in addition to being computationally sufficient. That is, it was developed under the additional constraint that the answers that QUEST produces should be the same as the answers that adults typically produce. Previous studies have reported the extent

This research was funded by grants awarded to the first author by the Office of Naval Research (N00014-88-K-0110 and N00014-90-J-1492) and a grant awarded to the second author by the Air Force Office of Scientific Research (88-0063). Requests for reprints should be sent to Arthur C. Graesser, Department of Psychology, Memphis State University, Memphis, TN 38152 U.S.A.

Typeset by $\mathcal{A}_{\mathcal{M}}\mathcal{S}$-TEX

to which QUEST can account for psychological data when questions are asked in the context of stories [14-16], expository texts on scientific mechanisms [17], and naturalistic conversation [18]. It is beyond the scope of this article to discuss the psychological validity of QUEST. It suffices to say that many of the theoretical components of QUEST have been supported in psychological experiments on question answering.

This article begins with a brief overview of the QUEST model of question answering [19-21]. We subsequently describe the conceptual graph structures and Q/A procedures that are associated with four types of knowledge: taxonomic hierarchies, spatial region hierarchies, goal/plan hierarchies, and causal networks. For each type of knowledge structure, we show how QUEST's Q/A procedures converge on a small number of answer nodes among hundreds of nodes in relevant information sources. The primary focus in this article is to describe the conceptual graph structures and the convergence mechanisms.

OVERVIEW OF QUEST

It is convenient to segregate QUEST into four procedural components: question interpretation, identification of relevant information sources, pragmatics, and convergence mechanisms. We acknowledge that an adequate Q/A model integrates these components in a highly interactive fashion [5,15,18], but it is beyond the scope of this article to elucidate how these interactions are accomplished.

Question Interpretation

The question is assigned to one of several question categories and translated into a standard form. QUEST assumes that each question category has a unique Q/A procedure. For example, "How did the video tape break?" is a how-event question. How-event questions have a Q/A procedure that elicits causal antecedents to the queried event (i.e., "the video tape broke"). During question interpretation, the question is translated into an expression with three elements, as illustrated below.

QUESTION (question category, queried node, information source)

Example: How did the tape break?

QUESTION (how-event question, the tape broke, ⟨Information source⟩)

Identification of Relevant Information Sources

The second component of QUEST identifies the information sources that are relevant to the question. An information source is a structured database that furnishes answers to a question. At least one information source must be accessed before a question can be interpreted. Without an information source, many questions are ambiguous, vague, or impossible to interpret. Several information sources are often relevant to a particular question. One class of information sources are "generic knowledge structures" (GKS), which are packages of generic knowledge which summarize the typical elements and relationships within a concept (e.g., the general concept of a VCR) [19]. When questions are answered, the relevant information sources accessed by a question normally include the GKS's associated with the content words. For example, the information sources for the question "How do you start the VCR?" would be the GKS for STARTING and the GKS for VCR. Of course, there would also be many other GKS's which are triggered by patterns of contextual information that accumulate in working memory. In addition to GKS's, there are "episodic" structures which are created from specific experiences (e.g., a video tape breaking on a particular day). Therefore, the information sources for a particular question consist of a family of episodic and generic knowledge structures. Given that each information source is a structured database with potentially hundreds of nodes, there is a wealth of information available when a question is answered. For example, if a question accesses 5 information sources and each source has 100 nodes, then 500 nodes would be available as candidate answers to the question.

Pragmatics

This component evaluates the pragmatic features of the communicative interaction within which the questioner and answerer are situated. This includes the mutual knowledge of the speech participants, that is, the knowledge that they believe each other shares. Another pragmatic consideration is the set of goals of the speech participants. For example, does the questioner genuinely seek an answer to the question or is the questioner merely monitoring the flow of conversation? Although the pragmatic component is essential for a theory of question answering [1,18,22], we do not focus on this aspect of QUEST in this article.

Convergence Mechanisms

These mechanisms compute the subset of nodes within the identified information sources that are good answers to a question. These convergence mechanisms narrow the "node space" from hundreds of nodes (as in the above example that had 500 nodes) to 10 or fewer good answers to a question. Convergence is accomplished by three mechanisms: (1) an intersecting node identifier, (2) an arc search procedure, and (3) constraint satisfaction. Although all three mechanisms predict good answers to questions, this article concentrates primarily on the arc search procedures.

The intersecting node identifier isolates those statement nodes from different information sources that intersect (i.e., match, overlap). For example, the node "X push power button" would be an intersecting node if it was stored in the GKS for STARTING and the GKS for VCR. These nodes have a special status for two reasons. First, psychological studies have shown that intersecting nodes have a higher likelihood of being produced as answers than do nonintersecting nodes [23]. Second, the likelihood of a node being produced as an answer decreases exponentially as a function of its "structural distance" (i.e., number of arcs) from the nearest intersecting node [17,22,23].

Each question category has its own arc search procedure. The arc search procedure generates answers by pursuing legal paths of arcs and avoiding illegal paths (as will be discussed in a later section). The legal paths of arcs are defined according to the types of directed arcs that are accepted by the question category.

The constraint satisfaction mechanism insures that the conceptual content of the answer is not incompatible with the content of the queried node. Candidate nodes are discarded if they are incompatible with the conceptual content of the queried node. For example, the candidate answer should not involve a direct contradiction or have a time frame that is incompatible with the queried node.

CONCEPTUAL GRAPH STRUCTURES IN QUEST

Each information source is represented as a conceptual graph structure (or "knowledge structure" for short). A knowledge structure contains a set of categorized nodes that are connected by categorized, directed arcs. Figure 1 shows an example conceptual graph structure that is associated with the concept of a VCR. This structure contains a taxonomic hierarchy (nodes 1-11), a spatial region hierarchy (nodes 11-16), a goal hierarchy (nodes 8,14,17-20,24-25), and a causal network (nodes 19-25).

Each node is either a concept or a statement. A concept is normally expressed as a noun or noun-phrase (e.g., VCR, electronic device). A statement is a proposition-like expression which contains a predicate (i.e., verb, adjective) and one or more arguments (i.e., noun, embedded proposition). Each argument also has a thematic role, such as agent, object, location, or time [10,23]. When a node is a statement, it is assigned to one of four categories: state, event, goal, or style specification. A state is an ongoing characteristic which remains unchanged within the time frame that is presupposed. An event is a state change that occurs within the time frame. A goal refers to a state or event that an agent desires. A style specification conveys the speed, intensity, force, or qualitative manner in which an event unfolds (e.g., an event occurs quickly, in circles, quietly). In principle, it is possible to include additional node categories in QUEST, but these categories have been satisfactory in previous studies [14-21]. At times, we have defined an intentional "action" as an amalgamation of a goal node that is linked via an Outcome arc to an event or state that achieves the goal. For example, nodes 17 and 25 in Figure 1 represent the

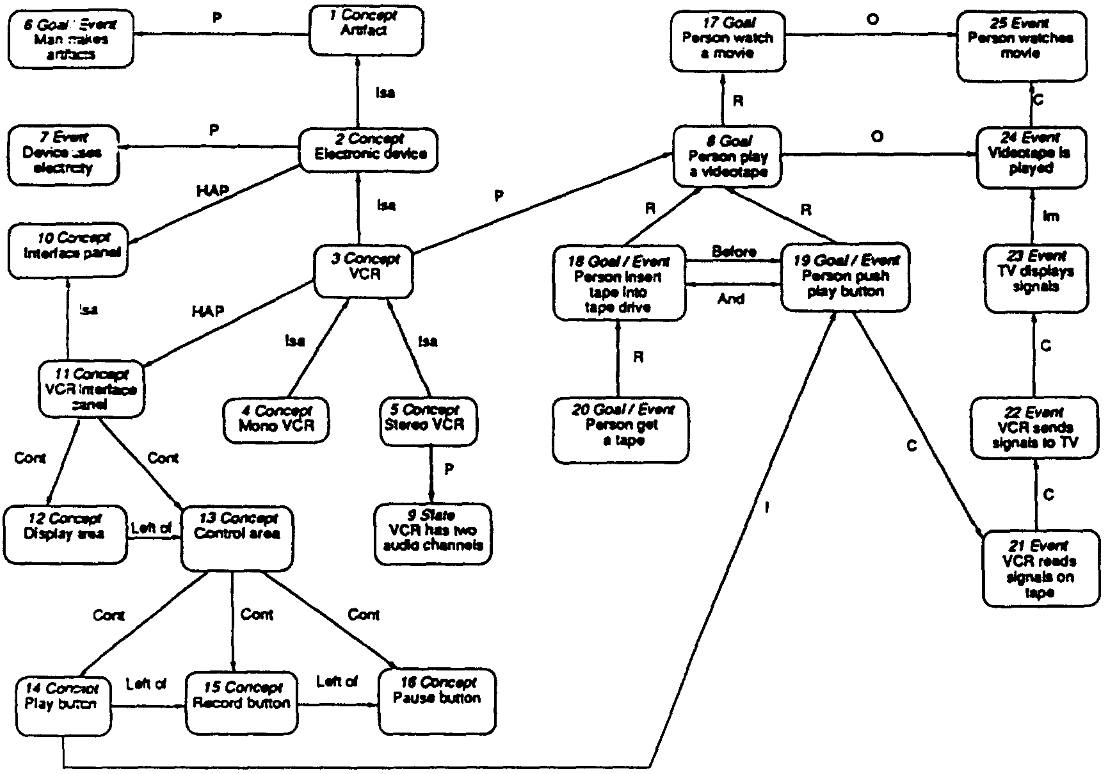

Figure 1. A conceptual graph structure that contains knowledge about VCR's.

action "person watches movie." It should be noted that each statement node in Figure 1 could be expressed more completely by specifying the predicate, the argument, and the thematic role of each argument. However, the verbal descriptions of the nodes in Figure 1 are adequate for the present article.

The nodes in a structure are interrelated by categorized arcs. The arc categories in the current version of QUEST are presented in Table 1. Table 1 includes the abbreviation of each arc category, its definition and constraints, its rules of composition, and an example. The composition rule specifies which node categories can be connected by a particular arc (e.g., Reason arcs can connect goal nodes but not other node categories). Most of the arc categories are directed, such that the end node is connected to the head of the arc and the source node is connected to the tail. Many arc categories also have an inverse form but Table 1 does not include inverse arcs. For example, the inverse of "before" is "after"; the inverse of "contains" is "is-in."

In principle, QUEST could be expanded with additional arc categories (e.g., X is equivalent to Y, X interferes with Y) and by subdividing some of the current arcs so that finer distinctions could be made. For example, the Consequence arc could be subdivided into "enables," "results-in," versus "directly causes." The Property arc could be subdivided into "function," "physical property," "setting," and so on. Such distinctions might indeed be necessary for certain applications. Nevertheless, we are satisfied with the current set of arcs because they are useful, if not necessary, for simulating human question answering in the context of taxonomic, spatial, goal-oriented, and causal knowledge structures. Moreover, the categories are sufficiently discriminable that they can be used by other researchers without getting bogged down into excessively subtle decisions. Aside from these functional and practical considerations, the arc categories have theoretical roots in semantic networks [9-11], conceptual dependency theory [4,12], and discourse analysis [3,24].

The arc categories vary among the taxonomic, spatial, goal-oriented, and causal structures. Each of these types of knowledge structures have a number of characteristics that are briefly elaborated below.

Table 1. Definitions and Composition Rules for Fourteen Categories of Arc

<table>
<tr><th>Arc Category</th><th>Definition and Temporal Constraints</th><th>Composition Rule</th><th>Example</th></tr>
<tr><td>Isa</td><td>A is a kind | type | instance of B</td><td>(concept)-Isa→(concept)</td><td>(Concept: robin-Isa→ (Concept: bird)</td></tr>
<tr><td>Has As Part (HAP)</td><td>A has as a part B</td><td>(concept)-HAP→(concept)</td><td>(Concept: robin)-HAP→(Concept: beak)</td></tr>
<tr><td>Property(P)</td><td>A has a property B</td><td>(concept)-P→ (state | event | goal)</td><td>(Concept:bird)-P→(State: bird has wings)</td></tr>
<tr><td>Referential Pointer (ref)</td><td>An argument of A refers to a concept B</td><td>(event | state | goal)-ref→ (concept)</td><td>Noun argument of (Event:The man died)-ref→ (Concept: man)</td></tr>
<tr><td>Spatial Relationship
left-of | right-of
top-of | bottom-of
east-of | west-of
north-of | south-of
touch | connect</td><td>A has a spatial relationship with B</td><td>(concept)-east-of→ (concept)</td><td>(Concept: Nevada)-east-of → (Concept: California)</td></tr>
<tr><td>Temporal Relationship
before | after
during</td><td>A has a temporal relationship with B</td><td>(goal)-before→(goal)
(event | state)-before → (event | state)</td><td>(Event:John died)-before→ (Event:Bob got married)</td></tr>
<tr><td>And | or</td><td>Both A and B exist | occur
Either A or B exists | occurs</td><td>(concept)-or → (concept)
(goal) -or →(goal)
(event | state)-or →(event | state)</td><td>(Event: John bought a car)-or→ (Event:John stole a car)</td></tr>
<tr><td>Consequence (C)</td><td>A causes or enables B
A precedes B in time</td><td>(event | state | style) -C → (event | state | style)</td><td>(Event: The daughters cried)-C → (Event: The heroes heard the cries)</td></tr>
<tr><td>Implies (Im)</td><td>A implies B
A and B overlap in time</td><td>(event | state | style)-Im → (event | state | style)</td><td>(State: The man was strong) (State: The man was powerful)</td></tr>
<tr><td>Reason (R)</td><td>B is a reason or motive for A
B is a superordinate goal of A
A is achieved before B is achieved</td><td>(goal)-R → (goal)</td><td>(Goal: The man wanted to buy a sandwich)-R → (Goal: The man wanted to eat)</td></tr>
<tr><td>Outcome (O)</td><td>B specifies whether or not the goal A is achieved</td><td>(goal)-O → (event | state | style)</td><td>(Goal: The man wanted to eat)-O → Event: The man ate food)</td></tr>
<tr><td>Initiate (I)</td><td>A initiates or triggers the goal in B
A precedes B in time</td><td>(event | state | style) -I→(goal)</td><td>(State: The man was hungry)-I→ (Goal: The man wanted to eat)</td></tr>
<tr><td>Manner (M)</td><td>B specifies the manner in which A occurs
A and B overlap in time</td><td>(goal)-M→ (goal | style)
(style)-M→(style)
(event)-M→(event | style)</td><td>(Goal: Heroes want to go to dragon)-M→ (Goal: Heroes want to run to dragon)
(Event: The daughters cried)-M→ (Style: The cries were loud)</td></tr>
</table>

Taxonomic Hierarchies

These structures have roots in semantic network theories and are widely recognized [9-11]. A taxonomic hierarchy contains a hierarchical structure of concept nodes, which are interrelated by isa-arcs. In addition, each concept node (C) has a number of distinctive properties (via the Property-arc) which distinguish C from the sibling nodes of C. When a concept C has a property P, then P is typically true about C but is not typically a property of the sibling nodes of C. For example, a bird can fly whereas reptiles, mammals, fish, and amphibians rarely fly. Of course, the quantifier "typically" is more appropriate than is "never"; some mammals do fly, such as bats. This "sibling node constraint" is a critical consideration because it prevents the researcher from haphazardly assigning properties to concepts. One other arc category that frequently emerges in taxonomic hierarchies is "has-as-part" (HAP), which identifies the parts of a concept [25]. There is no sibling node constraint associated with the HAP-arc because many parts of a concept are not distinctive to that concept.

It is widely acknowledged that taxonomic hierarchies are economical in the sense that many inferences can be generated from a structure that contains a small set of isa-arcs and Property-arcs. Some isa-expressions are directly stored in the taxonomic hierarchy, such as "A VCR is an electronic device" in Figure 1. Other isa-expressions are inferred by virtue of a "transitivity operator": If A isa B and B isa C, then it follows that A isa C. We would infer that "A VCR is an artifact" by virtue of the transitivity operator. There are 4 isa-expressions that explicitly connect nodes 1-5 in Figure 1, and 5 other isa-expressions that would be inferred from these five nodes via the transitivity operator. In addition to the transitivity operator, there is an "inheritance operator" which states that a concept C inherits the properties of concept nodes that are superordinate to C via the forward isa-arc (e.g., a VCR uses electricity); a superordinate property is not inherited if it contradicts any property of node C. There are 4 property-expressions directly stored in the taxonomic hierarchy in Figure 1 whereas 9 property-expressions would be inferred by an inheritance operator. When considering both the transitivity and inheritance operators, there are 14 inferences derived from the example taxonomic hierarchy that contains 8 explicit arcs. Therefore, the ratio of inferences per explicit arc is nearly 2 to 1. This ratio would be even higher if "inverse" arcs were considered. A knowledge structure is economical to the extent that there is a high ratio of inferences per explicit arc.

Spatial Region Hierarchies

These structures capture the spatial layout of regions and objects in regions [26]. There is a containment hierarchy of regions, with concept nodes related by contains-arcs. For example, the western United States contains California and Nevada; California contains San Diego and Los Angeles; and Nevada contains Reno and Las Vegas. These structures also contain spatial direction arcs that designate the relative spatial locations of regions (e.g., right-of/left-of, top-of/bottom-of, east-of/west-of, north-of/south-of). There is a sibling node constraint which specifies that only sibling nodes can be connected by a spatial direction arc. For example, cities within a state can be connected by the north/south/east/west relations but cities between states normally are *not* directly connected by these arcs.

The psychological representation of spatiality is compatible with a spatial containment hierarchy that has sibling node constraints on spatial direction arcs [20,21,26]. With such an organization and constraints, inferences must usually be made when determining whether Los Angeles is west or east of Reno, for example. Given that LA is in California, that Reno is in Nevada, and that California is west of Nevada, it follows that LA is west of Reno. This inference is made by adults even though the inference is false; LA is actually east of Reno on a map. The proposed region hierarchy is a more valid psychological representation of spatiality than is a Cartesian coordinate system.

The region hierarchy is very economical with respect to the ratio of inferences per explicit arc. A transitivity operator generates inferred containment expressions (e.g., a VCR interface panel contains a record button) from contains-arcs that are directly stored (e.g., a VCR interface panel contains a control area). Nodes 11-16 generate 5 explicit expressions about containment whereas 3 expressions would be derived by a transitivity operator. Similarly, inheritance and transitivity operators permit inferences of spatial directions that are not directly stored. There are 3 left-arcs directly stored in Figure 1 whereas there are 4 inferred expressions denoting spatial direction: the display area is left of the play button, the display area is left of the record button, the display area is left of the pause button, and the play button is left of the pause button.

Goal Hierarchies

Goal hierarchies underlie planned action sequences that are executed by agents [10,12,19,27,28]. Each goal node refers to a state or event that is desired by the agent. Nodes 8, 17, 18, 19, and 20 are example goal nodes in Figure 1. Node 17 is the most superordinate goal in the hierarchy and node 20 is the most subordinate. When a goal is achieved, there is an event or state that designates such an outcome via an Outcome-arc. As discussed earlier, an intentional action is an amalgamation of a goal and a successful outcome. Nodes 18, 19, and 20 in Figure 1 are categorized as "goal/event"; this is a shorthand notation for the goal and its successful outcome (i.e., an intentional action).

Goal hierarchies are hierarchical with respect to Reason and Manner arcs. There also are a number of arc categories that interrelate sibling nodes in the goal hierarchy. First, sibling nodes are interrelated by bidirectional and-arcs and or-arcs. Second, sibling nodes are related by before-arcs when temporal information needs to be conveyed. There is a sibling node constraint that states that only sibling nodes can be related by before-arcs. There also is an implicit temporal relation which specifies that a subgoal must be achieved before its superordinate goal. For example, a person must get the tape before putting the tape in the tape drive (see Figure 1). Between the directly stored before-arcs and the implicit temporal relations, it is possible to generate temporal inferences via a transitivity operator (e.g., the person gets a tape before the person pushes the play button). Although there is only one before-arc explicitly represented in Figure 1, there are 9 temporal inferences that would be derived from nodes 8/24, 17/25, 18, 19, and 20.

In addition to Reason, Manner, and Outcome arcs, there is one other arc category that frequently exists in goal hierarchies. Goals are prompted by states and events in the world by virtue of Initiate-arcs. For example, the state of being hungry initiates the goal of eating food.

Causal Networks

Causal networks underlie the event chains in physical, biological, and technological systems, e.g., tornadoes, mitosis, and nuclear power, respectively [17,19,29,30]. Nodes 19-25 form an event chain in Figure 1. Some of these events are inspired by goals of agents (nodes 19,24, and 25) whereas other events are entirely products of mechanistic systems (nodes 21,22, and 23). The events and states in a causal system are related by Consequence-arcs (which convey a weak sense of causality), Implies-arcs, and Manner-arcs.

A simple way of representing a causally driven set of events is by a chain of nodes, connected by Consequence, Implies, and Manner arcs. Additional complexity exists when there are structural loops. For example, rainfall involves a cycle of events rather than a linear chain. Complexity is also added if a particular event requires (a) a set of enabling states and (b) multiple, simultaneous, antecedent events.

It should be noted that the human mind cannot handle the level of complexity and sophistication that a scientist or engineer might need to describe a causal system. Therefore, the representation of causality in the cognitive system is somewhat different from that in science and technology [17,19,24]. A hierarchical structure must be constructed in the human mind when there are hundreds of nodes in the causal network. The mind chunks substructures into natural packages of information. This chunking imposes a hierarchical organization on the physical components and events in a network. In addition, adults frequently impose a teleological interpretation on scientific mechanisms. A teleological interpretation consists of a goal hierarchy that is superimposed on the events and states in the causal system. That is, an event E occurs for the purpose of achieving subsequent events. In technological systems the engineers clearly design artifacts in a way that satisfies specific goals.

Referential Pointers

One other arc category that was not mentioned above is the "referential pointer" (ref-arc). The argument of a statement node may be linked to a concept node by a ref-arc. For example, the statement node "the videotape broke" has the argument *videotape* which would have a ref-arc to the concept node for "videotape." The grouping of a set of nodes is also accomplished by a ref-arc. There is a group node (G) that is linked to a large set of nodes by ref-arcs. This occurs whenever a group of nodes is organized into a natural package of information.

We have not entirely resolved the extent to which ref-arcs should be incorporated in conceptual graph structures. At one extreme, ref-arcs may be extensively used in conceptual graph structures in order to explicitly capture (a) referents of arguments and (b) groups of nodes in natural packages of information. At the other extreme, ref-arcs may be used sparingly. Instead, conceptual and semantic procedures may be responsible for the binding of referents to arguments and for clustering nodes into natural groupings.

QUESTION ANSWERING PROCEDURES

This section concentrates primarily on the Q/A procedures of open-class questions rather than the closed-class verification questions. With regard to the latter, the previous section suggests how a variety of YES/NO verification questions would be answered by QUEST in the context of taxonomic, spatial, causal, and goal-oriented structures. Examples of these questions are listed below.

Is a VCR an electronic device?

Does a VCR use electricity?

Is the play button left of the pause button?

Does the person get the tape before he pushes the play button?

Answers to some of these questions would be YES because the information is directly stored in the information source. Other YES answers are derived inferentially by virtue of the transitivity and inheritance operators. NO answers are produced if the expression to be verified is not directly stored and not able to be derived inferentially. There are also conditions in which the appropriate answer is "maybe" or "don't know" but these answers are not addressed here.

ARC SEARCH PROCEDURES

As introduced earlier, an important feature of QUEST consists of three convergence mechanisms that narrow the node space from hundreds of nodes in the information sources to a handful of good answers to a question. These convergence mechanisms include an intersecting node identifier, an arc search procedure, and constraint satisfaction. We focus primarily on the arc search procedures in this section.

Each type of knowledge structure (i.e., taxonomic, spatial, causal, goal-oriented) has a set of question categories that is natural to ask. Each question category has a distinctive arc search procedure that pursues some paths of arcs but not others. Legal answers are on paths of arcs that are generated by the arc search procedure whereas illegal answers are not accessed by the arc search procedure.

Taxonomic Structures

Taxonomic hierarchies provide a natural organization for answering *definition* questions (i.e., What does X mean?, What is an X?). QUEST adopts a "genus-differentiae" procedure for answering definition questions, which is adopted in most dictionary definitions. This procedure produces the immediate superordinate node of concept X (via the forward isa-arc) and the properties directly linked to X (via the forward Property-arc), as illustrated below.

QUESTION: What is an X?

ANSWER: An X is a ⟨superordinate node via isa-arc⟩

that ⟨property-1 via Property-arc⟩, ⟨property-2 via Property-arc⟩,

... and ⟨property-n via Property-arc⟩

For example, the questions and answers below would be produced when Figure 1 is the information source.

What is a VCR?

A VCR is an electronic device that plays videotapes.

What is an electronic device?

An electronic device is an artifact that uses electricity.

Although there are 25 nodes in Figure 1, the answer to each question converges on only 2 nodes.

A second question category consists of class inclusion questions (i.e., What are some examples of X?, What are some types of X?). These answers tap subordinate nodes in the taxonomic hierarchy, on paths that radiate from the queried node via backward isa-arcs. Most answers are only one arc from the queried node but the arc search procedure permits answers that are many arcs away, as illustrated in the example below.

What is an example of an electronic device?

A VCR.

A mono VCR.

A stereo VCR.

A third category is a *contrast* question, i.e., What are the differences between X and Y? The arc search procedure for a contrast question is systematic, but more complex than the above question categories. Step 1 of the arc search procedure identifies the superordinate concepts of X and the superordinate concepts of Y (on paths of forward isa-arcs). Step 2 computes overlapping superordinate nodes from the two sets. Step 3 identifies the most subordinate node from the set of overlapping nodes; we refer to this node as the proximate superordinate node S. Step 4 identifies the child node of S (via the backward isa-arc) that is also either X or a superordinate concept of X; the properties of this child node are produced as properties of X. Similarly, step 5 identifies the child node of S (via the backward isa-arc) that is also either Y or a superordinate concept of Y; the properties of this child node are produced as properties of Y. Some example questions and answers are presented below.

What is the difference between a VCR and a radio?

People play videotapes on a VCR whereas radios broadcast radio signals.

What is the difference between a mono VCR and a stereo VCR?

A mono VCR has one audio channel whereas a stereo VCR has two audio channels.

The arc search procedure for *similarity* questions (i.e., How is X similar to Y?) is straightforward but will not be specified in this article.

Spatial Region Hierarchies

The obvious open-class question associated with spatial hierarchies is "Where is X?". When asked this question, QUEST produces nodes that (a) are superordinate to X in the region hierarchy via paths of backward contains-arcs and (b) spatial direction arcs that radiate from X. The superordinate region is normally one arc away from X but regions several arcs away are occasionally produced [22]. The question and answers below illustrate the arc search procedure for two where-questions that are asked in the context of Figure 1.

Where is the play button?

In the control area, to the left of the record button.

Where is the control area?

On the VCR interface panel, to the right of the display area.

Subordinate concepts are produced as answers to *containment* questions, i.e., What does X contain? The answers are on paths of nodes that radiate from X via forward contains-arcs, as shown in the example below.

What does the VCR interface contain?

A display area and a control area.

Play, record, and pause buttons.

According to QUEST, nodes that are one arc from X would be produced as answers more often than nodes that are several arcs away from X.

Goal Hierarchies

Three categories of questions are frequently asked about the goals and actions in goal hierarchies: *why, how, and what are the consequences.* It should be noted that the queried nodes for these questions are statement nodes rather than concept nodes.

Four sets of nodes are produced as answers to a why question when a goal node G is probed. First, there are superordinate goals that radiate from G via paths of forward Reason-arcs and backward Manner-arcs. Second, there are sibling nodes of G that radiate from G via paths of forward before-arcs. Third, there are goal initiators; these are connected by a backward Initiate-arc either to G or to any of G's superordinate goals. Fourth, there are causal antecedents to each goal initiator; these radiate from a goal initiator on paths of backward Consequence-arcs, Implies-arcs, backward Outcome-arcs, and backward Initiate-arcs. For the present purposes, we consider only the first and second sets of answers. The questions and answers below illustrate appropriate answers to two why questions when Figure 1 is the information source.

Why does a person insert a tape into the tape drive?

In order to play a video tape.

In order to watch a movie.

So that the person can push the play button.

Why does a person play a videotape?

In order to watch a movie.

In contrast to why questions, answers to how questions consist of subordinate achieved goals in the goal hierarchy. That is, if goal G is probed, then legal answers are on paths of backward Reason-arcs and forward Manner-arcs.

How does a person insert a tape into the tape drive?

The person gets a tape.

How does a person play a videotape?

The person gets a videotape,

inserts it into the tape drive, and pushes the play button.

Legal answers to consequence-questions include two sets of nodes. First there are achieved superordinate goals, via paths of forward Reason-arcs and backward Manner-arcs. Second, there are causal consequences of the queried node and the achieved superordinate goals (via paths of forward Consequence-arcs, Implies-arcs, forward Outcome-arcs, and forward Initiate-arcs). It should be noted that paths of causal antecedents constitute an inverse of paths of causal consequences.

What are the consequences of pushing the play button?

The VCR reads the signals on the tape.

The VCR sends signals to the TV.

The TV displays the signals.

The videotape is played.

The person watches the movie.

Causal Networks

As in the case of goal hierarchies, the natural questions for causal networks are why, how, and consequence questions. The arc search procedures are rather straightforward for these questions. Answers to why-event questions are on paths of causal antecedents to the queried event or state. Answers to how-questions are on paths of causal antecedents, with additional style embellishments via paths of forward Manner-arcs. Answers to consequence-event questions are on paths of causal consequences. Example Q&A protocols are presented below in the context of Figure 1.

Why/how does the TV display signals?

The VCR sends signals to the TV.

The VCR reads the signals on the tape.

The person pushes a play button.

A play button exists. (node 14)

What are the consequences of the TV displaying signals?

The videotape is played.

The person watches a movie.

Additional questions are frequently asked in the context of causal networks, such as "what enabled X to occur?" and "when did X occur?". The arc search procedures for these question categories are specified in other reports [17,19-21].

Convergence Scores

A convergence score can be computed for each Q/A procedure as an index of the extent to which the procedure narrows down the node space to good answers. The convergence score is simply the proportion of nodes in the relevant information sources that would be generated by QUEST as good answers to a particular question. An adequate model would have a very *low* convergence score (close to 0), signifying that very few nodes are good answers to a particular question. An entirely separate criteria for evaluating the success of a Q/A procedure is that of psychological validity. A perfectly valid Q/A procedure would produce answers that perfectly correspond to answers that humans produce. The psychological validity of the Q/A procedures is quite impressive [14-21], but this article does not directly address this issue.

We performed computer simulations that computed convergence scores for why, how, and consequence questions. The simulation included the three convergence mechanisms in QUEST: the arc search procedure, dampening via structural distance between question and answer, and constraint satisfaction.

The information sources consisted of eight generic knowledge structures that spanned many knowledge domains. The GKSs were hero, child, home, tree, time, fighting, walking, and crying. Each GKS was represented as a conceptual graph structure; the content and structure of each GKS was reported in a book by Graesser and Clark [19]. The number of nodes per GKS varied from 102 (for hero) to 259 (for tree), with a mean of 174 nodes.

Eight actions and events were randomly selected from each GKS and were probed with a why, how, and consequence question. Therefore, there were 64 why questions, 64 how questions, and 64 consequence questions. We simulated answers to these 192 questions by submitting each question to our computer model of QUEST. The computer model implements the arc search procedures of each question category in LISP on a Texas Instrument EXPLORER II computer. The user first declares the relevant information source (i.e., a GKS) and then asks QUEST a question; the program subsequently prints out all legal answers based on the appropriate arc search procedure. A convergence score was computed for each particular question and mean convergence scores were collected for each question category. The mean convergence score was .11, .10, and .09 for why, how, and consequence questions, respectively, yielding an overall mean of .10. This estimate of convergence incorporated the arc search procedure but not the other two convergence mechanisms.

The second convergence mechanism of QUEST is the intersecting node identifier and the structural distance gradient. Given that the simulation considered only one structure, a single GKS for any particular question, the only intersecting node in the structure was the node that matched the queried node. Nodes that were structurally close to the queried node should be better answers than nodes that are many arcs away from the queried node. According to QUEST, the probability of a node being produced as an answer dampens exponentially as a function of the structural distance from the queried node [17,19]. This dampening function is applied to the legal answers that pass the arc search procedures, not the illegal answers. Given a distance d and the likelihood of traversing a single arc t, the probability that a legal node is produced as an answer is t^d. That is, legal answers that are 1, 2, versus 3 arcs away from the queried node have answer production scores of t, t^2, and t^3, respectively. An empirical estimate of t is .67 [17], which would yield answer production scores of .67, .45, and .30 for distances of 1, 2, and 3 arcs, respectively. Using this estimate of t, we computed the proportion of QUEST's simulated legal answers that would pass the structural distance dampening function. These proportion scores were .38, .43, and .34 for why, how, and consequence questions, respectively, yielding an overall mean of .38. Therefore, given that .10 of the nodes in an information source are legal answers that pass arc search, and given that .38 of these legal answers pass the structural distance dampening function, then .04 (i.e., .10 × .38) of the nodes in the information source pass both mechanisms of convergence.

The third mechanism of convergence was constraint satisfaction. The likelihood that a legal answer passes constraint satisfaction has been estimated empirically at .55. Using this estimate, an overall convergence score would be .02 (i.e., .10 × .38 × .55). Therefore, 2% of the answers in an information source would be produced as an answer to a particular question, on the average. The average GKS had 174 nodes, so approximately 3.5 answers would be produced when a why, how, or consequence question is asked in the context of a generic knowledge structure. This estimate is quantitatively close to the results of studies on human question answering [15,18,19].

CLOSING COMMENTS

There are a number of practical uses of a theory of human question answering that is computationally sufficient. We can design human-computer interfaces that seriously incorporate question answering processes. In such an interface, the user would ask the computer questions and the computer would produce answers according to the Q/A procedures in QUEST. Another area of application is the design of expert systems. One of the critical bottlenecks in the design of these systems is "knowledge elicitation," the process of extracting knowledge from topic experts and translating this knowledge into structured databases. Gordon and Gill [31] have proposed a "question probe technique" which systematically asks the topic experts questions in order to elicit important knowledge about a topic. Now that we have some understanding of the Q/A procedures and the organization of world knowledge, it is possible to offer specific recommendations on what questions to ask in the question probe technique [32]. By grounding knowledge elicitation methods in a question answering theory, we can select those questions that maximize the acquisition of useful information and that minimize interview time.

REFERENCES

1. J. Allen, Recognizing intentions from natural language utterances, In *Computational Models of Discourse*, (Edited by M. Brady and R.C. Berwick), MIT Press, Cambridge, MA, (1983).
2. B.C. Bruce, Natural communication between person and computer, In *Strategies for Natural Language Processing*, (Edited by W.G. Lehnert and M.H. Ringle), Erlbaum, Hillsdale, NJ, (1982).
3. K. Dahlgren, *Naive Semantics for Natural Language Understanding*, Kluwer Academic Publishers, Boston, (1988).
4. W.G. Lehnert, *The Process of Question Answering*, Erlbaum, Hillsdale, NJ, (1978).
5. W.G. Lehnert, M.G. Dyer, P.N. Johnson, C.J. Young and S. Harley, BORIS—An experiment in in-depth understanding of narratives, *Artificial Intelligence* **20** (1), 15–62 (1983).
6. K.R. McKeown, Discourse strategies for generating natural-language text, *Artificial Intelligence* **27** (1), 1–41 (1985).

7. A. Souther, L. Acker, J. Lester and B. Porter, Using view types to generate explanations in intelligent tutoring systems, In *Proceedings of the Eleventh Annual Conference of the Cognitive Science Society*, pp. 123–130, Erlbaum, Hillsdale, New Jersey, (1989).
8. W.A. Woods, Lunar rocks in natural English: Explorations in natural language question answering, In *Linguistic Structures Processing*, (Edited by A. Zampoli), Elsevier/North-Holland, New York, (1977).
9. A. Collins, E.H. Warnock, N. Aiello and. M.L. Miller, Reasoning from incomplete knowledge, In *Representation and Understanding*, (Edited by D.G. Bobrow and A. Collins), Academic Press, New York, (1975).
10. D.A. Norman and D.E. Rumelhart, *Explorations in Cognition*, Freeman, San Francisco, (1975).
11. R.J. Brachman, What IS-A is and isn't: An analysis of taxonomic links in semantic networks, *Computer* **16** (1), 30–36 (1983).
12. R.C. Schank and R. Abelson, *Scripts, Plans, Goals, and Understanding*, Erlbaum, Hillsdale, NJ, (1977).
13. J.F. Sowa, *Conceptual Structures: Information Processing in Mind and Machine*, Addison-Wesley, Reading MA, (1983).
14. J.M. Golding, A.C. Graesser and K.K. Millis, What makes a good answer to a question?: Testing a psychological model of question answering, *Discourse Processes* **13** (3), 305–325 (1990).
15. A.C. Graesser and T. Murachver, Symbolic procedures of question answering, In*The Psychology of Questions*, (Edited by A.C. Graesser, and J.B. Black), Erlbaum, Hillsdale, NJ, (1985).
16. A.C. Graesser, S.P. Robertson and P.A. Anderson, Incorporating inferences in narrative representations: A study of how and why, *Cognitive Psychology* **13** (1), 1–26 (1981).
17. A.C. Graesser, D.D. Hemphill, and L.E. Brainerd, Question answering in the context of causal mechanisms, In *Proceedings of the 11th Annual Conference of the Cognitive Science Society*, pp. 621–626, Erlbaum, Hillsdale, NJ, (1989).
18. A.C. Graesser, R.M. Roberts and C. Hackett-Renner, Question answering in the context of telephone surveys, business interactions, and interviews, *Discourse Processes* **13** (3), 327–348 (1990).
19. A.C. Graesser and L.F. Clark, *Structures and Procedures of Implicit Knowledge*, Ablex, Norwood, NJ, (1985).
20. A.C. Graesser and S.E. Gordon, Question answering and the organization of world knowledge, In *Essays in Honor of George Mandler*, (Edited by G. Craik, A. Ortony and W. Kessen), Erlbaum, Hillsdale, NJ, (in press).
21. A.C. Graesser and S.P. Franklin, QUEST: A cognitive model of question answering, *Discourse Processes* **13** (3), 279–303 (1990).
22. B. Shanon, Answers to where-questions, *Discourse Processes* **6** (4), 319–352 (1983).
23. J.F. Allen, *Natural Language Understanding*, Benjamin/Cummings, Menlo Park, CA, (1987).
24. T. Trabasso, P. van den Broek and L. Lui, A model for generating questions that assess and promote comprehension, *Questioning Exchange* **2** (1), 25–38 (1988).
25. B. Tversky and K. Hemenway, Categories of environmental scenes, *Cognitive Psychology* **15** (2), 121–149 (1983).
26. A. Stevens and P. Coupe, Distortions in judged spatial relations, *Cognitive Psychology* **10** (4), 422–437 (1978).
27. S.K. Card, T.P. Moran and A. Newell, *The Psychology of Human-Computer Interaction*, Erlbaum, Hillsdale, NJ, (1983).
28. G.A. Miller, E. Gallanter and K.H. Pribram, *Plans and the Structure of Behavior*, Holt, Rinehart, Winston, New York, (1960).
29. J.S. Brown, R.R. Burton and J. De Kleer, Pedagogical, natural language and knowledge engineering techniques in SOPHIE I, II, and III, In *Intelligent Tutoring Systems*, (Edited by D. Sleeman and J. S. Brown), Academic Press, New York, (1982).
30. B. Kuipers, Commonsense reasoning about causality: Deriving behavior from structure, *Artificial Intelligence* **24** (1–3), 169–203 (1984).
31. S.E. Gordon and R. Gill, Question probes: A structured method for eliciting declarative knowledge, *AI Applications in Natural Resource Management* **3**, 13–20 (1989).
32. S.E. Gordon, Knowledge engineering: the use of question probes with conceptual graph structures, Paper presented at the *Question Generation and Applications to Information Systems Conference*, Oakland University, (June 1990).