DOCUMENT RESUME

ED 308 693 FL 018 031

AUTHOR Psotka, Joseph; And Others

TITLE Cognitive Models of Students' Language Structure: The View from Intelligent Computer Assisted Instruction.

PUB DATE 15 Sep 88

NOTE 34p.; Paper presented at the Interagency Language Roundtable Invitational Symposium on Language Aptitude Testing (Rosslyn, VA, September 14-16, 1988). Some graphics may not reproduce clearly.

PUB TYPE Reports - Evaluative/Feasibility (142) -- Speeches/Conference Papers (150)

EDRS PRICE MF01/PC02 Plus Postage.

DESCRIPTORS *Artificial Intelligence; Cognitive Ability; Communicative Competence (Languages); Computational Linguistics; *Computer Assisted Instruction; Computer Software; Grammar; Higher Education; *Individualized Instruction; Language Aptitude; *Language Processing; Language Research; Language Tests; Models; *Programing; Second Language Instruction; Technological Advancement

IDENTIFIERS *Army Research Institute; *Hypertext; Parsing; Semantic Mapping

ABSTRACT

Methods and approaches used at the U.S. Army Research Institute to investigate intelligent computer-assisted foreign language instruction are discussed. The research described explores the use of hypertext and natural language processing for improving language training by articulating student knowledge structures and for providing, incidentally, a new basis for aptitude testing. The techniques being implemented model the cognitive skills underlying foreign language competence by using computational linguistic models and semantic networks built into hypertext systems. It is proposed that these techniques could be useful for assessing basic language competence. Aspects of the research and related technology are examined: intelligent tutoring systems, the expert model and second language learning, student (trainee) models and language aptitude testing, the pedagogical model, the knowledge base, the interface, hypertext systems and command menus, smart computer-assisted language learning environments, grammars and grammar representations, parsing strategies, using grammars for computer-assisted language learning, use and adaptation of specific software packages, immersion techniques, and semantic mapping. (MSE)

Cognitive Models of Students' Language Structure: The View from Intelligent Computer Assisted Instruction

Joseph Psotka
Melissa Holland
Merryanna L.Swartz
Seymour Hanfling
Khanh Bui
U. S. Army Research Institute
ATTN: PERI-ICC
5001 Eisenhower Avenue
Alexandria, VA 22333-5600

(202)274-5540/5545/5569

Our work is aimed at developing job and training aids for military machine translation and military Intelligent Computer - Aided Instruction (ICAI) for foreign language training. This talk will try to link these matters clearly to aptitude testing. It is not our intention to create a new set of tests: we see that as an enormous undertaking; nor to improve existing tests: again, an enormous task. Instead we should like to propose a theoretical framework explicitly grounded in empirical research that offers a real possibility for doing both. In order to make our position clear, we will have to explain certain new technologies that we find useful. Two technologies that show great promise for improving training are hypertext and natural language processing. Hypertext provides a text based system that goes beyond text to include graphics, video, and sound (hypermedia) as well as links, cross-references, and network or lattice structures. Natural language processing refers to the use of parsers, grammars, and dictionaries to provide computer-based language facilities. In order to individualize instruction to make it as effective as one-on-one tutoring, we must create effective student models that capture students' knowledge structures and skills. This provides the clearest connection to aptitude testing for second language learning skills and abilities, since student modelling is a directly parallel activity to student aptitude assessment. Our goal, therefore, is to investigate these new technologies to see how they may best be used to articulate student knowledge structures, and incidentally provide a new basis for aptitude testing. However, this is not the only connection we want to forge. In particular, we are interested in the web of links and nodes that relate students' knowledge in one language with their similar semantic and syntactic structures in the language they are learning. We are investigating ways to improve upon these technologies of hypertext and parsers to increase the effectiveness of training for the acquisition, sustainment, and automatic assessment of foreign language skills. Particularly, we are implementing techniques for modelling the cognitive skills underlying foreign language competence using computational linguistic models and semantic networks built in hypertext systems. These techniques may prove useful for assessing basic competencies.

INTELLIGENT TUTORING SYSTEMS (ITS)

To meet the needs of the high technology Army of the future, soldiers will have to learn more, in less time, about operating and maintaining complex systems. Current training technology is inadequate to meet these needs, but fortunately, a dramatic new approach is at hand. Artificial intelligence (AI) technology, in the form of intelligent tutoring systems (ITS), will revolutionize training much as autonomous vehicles are expected to revolutionize combat.
Over the past five years ARI has accumulated a great deal of practical knowledge about the use of expert systems for instruction. We have supported research in leading academic centers on the major topics necessary for the effective construction of ITS. We have carried out research and development efforts that have created working ITS in the domains of electronic troubleshooting, computer programming, technical training, and foreign language instruction (Simutis and Psotka, 1987, Gray, Pliske, and Psotka, 1985). From these efforts, it is safe to say that several of the technology base domains are well developed and ready to apply in broad Army mission areas (Fletcher and Psotka, 1986). Other domains have lagged behind, and are only beginning to show the benefits that have long been promised.

As ITS become more practical and effective, their applications are moving from well understood and structured academic domains to fuzzier, informal everyday situations. At the same time they are beginning to deal with ever more complex topics and broader areas of world knowledge. (Psotka, Massey, and Mutter, 1988) Techniques that are well suited to simpler topics (such as mathematical modelling and hierarchical decomposition) founder on the complexity of these new domains. Other techniques for solving many of the problems that have stubbornly resisted easy resolution are only beginning to be developed (Lawler and Yazdani, 1987). Natural Language Processing (NLP), promising the ideal human interface for many purposes, falls into this category. In order to be clearer about its role in ITS, the next section provides an introductory overview of ITS and their principle components.

Components of ITS

A good human tutor is an expert in the domain being taught and an expert pedagogue. Being a good pedagogue implies a number of sophisticated skills. A good tutor is able to accurately monitor and model each trainees' knowledge state. That is, the tutor is able to determine what each trainee knows, does not know, and knows incorrectly. The third part of that skill, knowing where trainees typically go wrong, has turned out to be the key to developing good ITS. It is also the principal basis for our optimism that ITS techniques may be able to shed new light on aptitude testing. In real life, the tutor's knowledge of the trainee's level of knowledge extends in time allowing the tutor to make predictions about instructional level and pacing. The tutor has an extensive set of knowledge resources:
books, graphics, films, carefully prepared courses of instruction; and the skill to use these appropriately. Finally, the tutor is able to marshall all these resources and skills in a carefully coordinated interaction with the trainee. These five aspects of a good human tutor are analogous in ITS to the expert model, the student model, the pedagogical model, the knowledge base, and the interface, respectively. Each is described in greater detail below.

The Expert Model

Expert systems are the most widely recognized application of artificial intelligence technology. It will therefore be useful to consider how the expert model in an ITS is and is not like an expert system. (For a more detailed discussion, see Clancey, 1986).

As an expert system, the expert model of an ITS contains both facts and procedures relevant to a domain. Both expert systems and ITS components should be capable of using these facts and procedures to solve problems that could be solved by a human expert. Moreover, both must be able to explain to a human why or how they reached their conclusions.

Expert systems and the expert model of an ITS differ mainly in the degree to which their procedures resemble those used by human experts. Useful expert systems simply organize their knowledge base into an arbitrary, but very efficient, collection of "if ... then" rules. In contrast, an expert system for training must organize its rules into cause and effect sequences or hierarchical structures that more closely resemble the organization of concepts by expert humans. There are many reasons why this is the case, but primarily this structure is adopted so that the other components can fulfill their function by engaging in a communicative dialog with the expert model.

Significantly, the role of cognitive science and AI tools and environments has been largely to reify the abstract connectivity of conceptual structures to make them amenable to public inspection. This is also the focus of hypertext systems (as will be discussed in greater detail later). Work on reifying declarative conceptual structures is proceeding apace and can in some sense be seen as the special focus of the IDE hypertext environment (Russell, Moran and Jordan, 1988; see also Lenat et al., 1985). As this paper will try to clarify, this is one of the fundamental strengths of computational grammars and tools connected to them: they can make the grammatical relations among words visible and manipulable. Once directly visible, these mental skills and structures can be modified in much the same way as athletic and music coaches influence the growth of performance skills.

The Expert Model and Second Language Learning

Technologies that will be described in greater detail later: Prolog and Unification Grammars, make it possible to create expert systems that can parse and to some extent understand natural languages, especially related languages like English and German. Their degree of expertise varies considerably with the amount of time and effort expended on their development, but for beginning learners of foreign languages, these systems are surprisingly competent. To summarize for the moment, computational systems can formulate some of the understanding skills of human language users by creating symbolic expressions of the grammatical rules of syntax and lexicon. These rule systems provide an objective and detailed description of language skills. How cognitively valid they are is a matter of some debate, but there is every reason to believe that it is worthwhile to examine the question experimentally; and even if it turns out that these representation systems are not good cognitive models of human skills, they can nevertheless be used to create learning environments of considerable power (Yazdani, 1987).

The Student Model

Student models are built in real time during the trainee's interaction with the ITS tutor. Each trainee's model is dynamic in that it is modified by the most recent trainee- computer interaction. Often it uses as its source a library of bugs, typical errors, and inconsistent models of the world that correspond to common misconceptions about the causal structure of the world held by many trainees. The structure of these resources depends heavily on explicit knowledge acquisition research in the best tradition of cognitive science. The student model guides the tutor in pacing and sequencing the instruction. Thus the ability to build an accurate trainee model is a critical component of any ITS.

Sleeman & Brown (1982) have discussed three approaches to the represention of trainee knowledge. The overlay approach represents the trainee's knowledge as a subset of the expert model. The differential approach "abstracts how the trainee's behavior is critically different from that of an expert". The coach approach minimizes interaction and focuses on a prearranged group of issues that are individualized to a trainee's particular stage of development. Other approaches such as perturbation and mal- rules attempt to represent the trainee's knowledge as misconceptions and deviations in the procedural structure associated with correct skills. It should be evident that the student model is not simply a reduced subset of the expert model. Nevertheless, an accurate analysis of the expert model is usually an excellent foundation and precondition for developing an approach to the student model.

Language Aptitude Testing and Student Modeling

The computational techniques used to create parsers and grammars also provide a way of describing students' language skills in the form of rules and data. Lytinen and Moon (1988) present a thoughtful analysis that provides a provocative case for the argument we are proposing here. They delineated a system (IMMIGRANT) that analyzed real teachers' instructions to students of second languages in English and wrote new rules for German grammar based on these instructions. For instance, an example instruction might be : "In German, verbs come at the end of relative clauses." or, "In German, the case of a prepositional object must match the case required by the preposition." Obviously, these are simple rules for relatively novice students.

For each instruction the system analyzed the terms in the instructions and created a proposed set of rules to add the English rule base: e.g.;

PP
(1) = PREP
(2)=NP
(1 RIBORD) = (2 LEFBORD)
(LEFBORD) = (1 LEFBORD)
(RIBORD) = (2 RIBORD)

(HEAD) = (1 HEAD)
(HEAD PREP-OBJ) = (2 HEAD)

When unified with the total English grammar database, IMMIGRANT adds:

(1 CASE) = (HEAD PREP-OBJ)

This then defines how the English grammar is inadequate as a model of German grammar. One can also think of this as a technique that can be applied to the analysis of any particular student's skills to define the inadequacies of the student's grammar.

Imagine, furthermore, a systematic set of unifications to generate a more or less complete German grammar as derived from the English. How different would this look from a German grammar created explicitly for German? How different would the actual sequence of linguistic utterances be when generated by these two hypothetical systems, one derived from English and the other created specifically for German? Also, could one use the difficulty of transforming particular English rules to predict the difficulty that English speakers have learning those differences? These are questions that have no answer yet, but await a detailed computational analysis of these sorts of student models.

For the first time, we are able to make specific predictions about second language production by novice language learners on the basis of their first language using this explicit theoretical framework. For now, we must expect limited success from such a development, since these techniques are only beginning to develop, but a bold new enterprise lies before us.

The Pedagogical Model

The pedagogical model is the curriculum part of an ITS. This component generates instruction based upon the trainee's instructional history. The ability to generate instruction is best understood by way of contrast to more traditional forms of computer-based training (CBT). In traditional CBT, a limited number of instructional sequences are built into the design of the lesson. Most CBT presents the trainee with a fairly well-defined, hierarchical path from start to finish. Additional prestored sequences may be provided to trainees who need extra help on a point, but the trainee is quickly shuttled back to the main sequence. Feedback for the trainee's answers is also prestored and is typically based only upon the trainee's immediate answer to the question. The trainee's history of prior successes and failures seldom affects this feedback. In contrast, in ITS both the instructional sequence and the feedback for answers is generated by the tutor in light of the trainee's unique instructional history. ITS thus have the potential to provide each trainee with an individualized instructional sequence.

The pedagogical model is not just more complex, it is principled in ways that earlier forms of CBT could not be. The pedagogical model is rule-based and able to adopt both strategic and tactical approaches to instruction. Given a student model that may be a huge semantic network of the appropriate knowledge, the pedagogical model may analyze an individual trainee's knowledge network into specific trees or grammars to decide which nodes are ripe to be used to attach new information to make the structure grow in a predefined way. Similarly, the model may use this structure to decide which resources in the knowledge base to use to have the greatest effect

Second Language Learning and the Pedagogical Model

The grammatical formalisms the specify the expert and student models clearly have specific implications for the order and content on instruction. A careful analysis of the first language compared to a similar analysis for the second language should make specific predictions about the order of sequencing the new rules to be learned. This order also constrains the content of instruction. Since, it is unlikely that the best instruction will focus on the grammatical rules themselves, the more important impact may well be on the selection of representative and illuminating examples that make the underlying rule easily and intuitively learnable.

There is another important offshoot of these new technologies: they emphasize the interconnectedness of lexical structure, and they offer a means of concretely representing these connections and making them directly visible and interactively inspectable. The structure of lexical interconnections can be rendered as a visible semantic network, tree, or directed acyclic graph (DAG). This structure can be displayed graphically on a computer. It makes the connections among words transparent and graspable at a glance. It provides a way of systematically organizing and sequencing the learning of vocabulary in structures that have not really been available before.

The Knowledge Base

The knowledge base is potentially the most important and extensive component. Knowledge of every kind needs to be made available for instructional purposes, and it needs to be in the form for most effective training. The knowledge base constitutes more than the curriculum or program of instruction (POI). It also needs to be the environment where training takes place. It is usually an enormous set of resources (graphics, simulations, text, and animations) that must be carefully structured and organized so that it can be properly presented through the human interface. Convenient knowledge representation techniques include graphic simulations, graphic browsers, qualitative models, and hypertext.

Too little work has been aimed at developing new representations for information and relationships. There are several important categories: representing justifications, consequences, and causal mechanisms.

A representation is a structure of objects, relationships, and operations together with a mapping that places this structure in some explicit relationship (correspondence, transformation). Of most obvious importance for language instruction is the development of structures that encode and make explicit connections among words, concepts, images and other mental structures. Hypertext systems provide the most general and powerful solution to these problems. Graphic browsers are the best developed of these representation systems and underlie most useful hypertext systems. Qualitative models, viewed generically, provide additional interactive and programmatic possibilities. The graphics packages used in MACH-III (developed out of the STEAMER projects) make inspections of the causal relationships among components much more direct than any physical device can allow.

Qualitative simulations provide convenient vehicles for creating systems with both meaningful structural and functional components. Instead of a textual description of terminology and its interrelationships, the structure is described visually. Functional relations can also be described by using animation, color, arrows, and textual descriptions. However, visual descriptions of both structure and function are notoriously concrete: it is difficult to obtain the right level of abstraction without the use of conceptual descriptions in a textual and hierarchical form. Furthermore, it is difficult to create visual descriptions at different levels of abstractions in a truly hierarchical format. Functional and structural relations must be compressed or eliminated in arbitrary ways that defy accurate concrete representations. So, it is only when conceptual (textual) and concrete (visual) representations complement each other that adequately faithful models can be created. These problems must be resolved at the point where tutor and trainee meet, the human interface.

The Interface

The resources of the knowledge base are all employed in the course of instruction. Access to this large database can often be unwieldy, time consuming, and cumbersome. A natural language interface would resolve many problems, but it remains a dream quite a distance beyond our grasp for most applications. Because of the complexity of the materials available, unique interfaces have been designed to manage the potential explosion of possibilities. The most important recent development has been the exploration of hypertext structures. Hypertext provides a systematic approach to structuring and delivering knowledge organized in complex networks and presented in multi-media formats. It is uniquely suited to training design and development. Some common and important features of hypertext systems that can be exploited for training will be described in the following sections.

Vannevar Bush (1945) is generally credited with the basic ideas of hypertext systems. However, it is only recently that computers with high bandwidth graphical displays have been able to implement these ideas in an acceptable way (Forsdick, 1985, Halasz et al., 1985, McCracken and Akscyn, 1984, McMath, Tamaru, and Rada, 1987). The central features of a hypertext, multimedia system appear to be hotspots that appear on the interface of the screen/text/graphic surface that can be activated to bring up further selections or the materials that have been linked to the hotspot. The rich interlinkages of graphic and textual materials that can be created with these mechanisms are only beginning to be explored. The unique facilities for linking text and simulations in very flexible but controlled ways is the hallmark of hypertext systems.

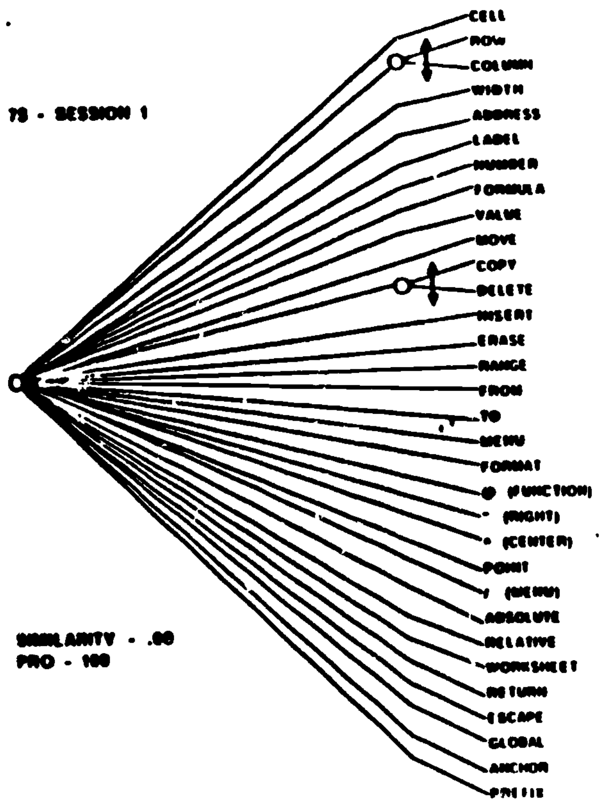
Hypertext and Command Menus

Hypertext systems are integrally linked to command menu systems, such as those popularized so dramatically in modern electronic spreadsheets. Texts written in hypertext act like implicit command menus. However, command menus are usually

simple trees or hierarchies. When one is at the bottom of the tree, it is usually necessary to follow the tree back up to the top before taking another track to get to a different part of the tree. Hypertext opens the real possibility of converting these trees into networks. This raises the very real question of how to structure the networks. The suggestion has been made by many researchers that the proper structure to use can be determined by comparison with the structure of semantic memory.
The structure of semantic memory has been analyzed in various terms as a system of nodes and links, much like a hypertext system (Anderson, 1983; Collins & Quillian, 1969; Rosch & Mervis, 1975; Smith & Medin, 1981). Within this framework the structure of concepts and the links between them are dependent on the relationship of features and default values stored at each concept node. In order to elucidate this structure we have carried out a series of experiments on the differences between novices' and experts' understanding of LOTUS commands (Mutter et al., In Press).

Information about the novice-to-expert transition is critical for developing training for the use of command languages. Our study examined the evolution in knowledge structures as the trainees pass from a novice to an intermediate level of expertise. These data enabled us to develop a student model for the acquisition of expertise in using command languages.

Our task involved multi-trial cued recall (MTCR) of words representing command language concepts (e.g. cell, column, address) or functions (e.g. edit, copy, move, save). This test yields a hierarchical tree-like structure of the trainee's organization of these words. By comparing the tree-like structures from various points during the ten-week period, we expected to obtain a picture of how the organization of these command language terms changes over time. Some preliminary data from this task are presented below.


7S - SESSION 1
CELL
ROW
COLUMN
WIDTH
ADDRESS
LABEL
NUMBER
FORMULA
VALUE
MOVE
COPY
DELETE
INSERT
ERASE
RANGE
FROM
TO
MENU
FORMAT
@ (FUNCTION)
" (RIGHT)
^ (CENTER)
POINT
/ (MENU)
ABSOLUTE
RELATIVE
WORKSHEET
RETURN
ESCAPE
GLOBAL
ANCHOR
PREFIX
SIMILARITY - .60
PRO - 100

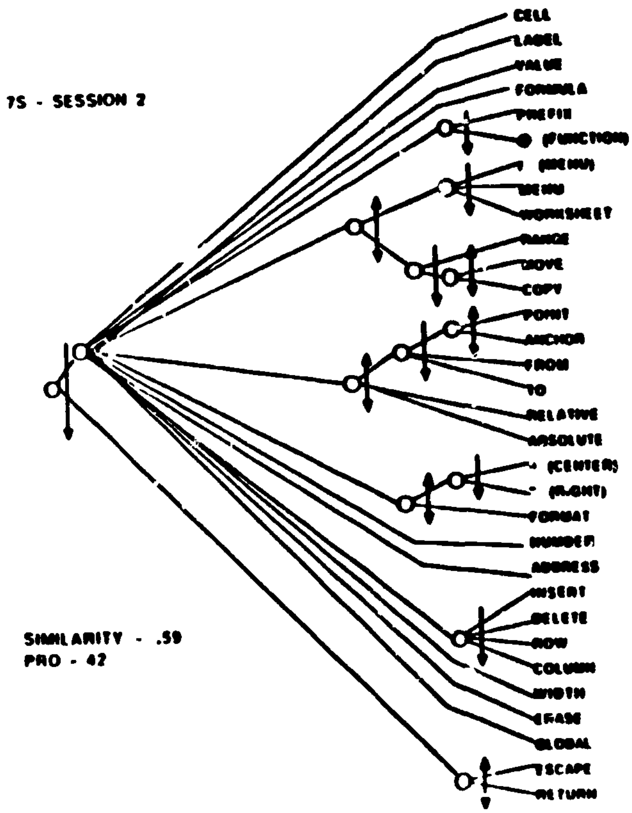

7S - SESSION 2
CELL
LABEL
VALUE
FORMULA
PREFIX
@ (FUNCTION)
/ (MENU)
MENU
WORKSHEET
RANGE
MOVE
COPY
POINT
ANCHOR
FROM
TO
RELATIVE
ABSOLUTE
^ (CENTER)
" (RIGHT)
FORMAT
NUMBER
ADDRESS
INSERT
DELETE
ROW
COLUMN
WIDTH
ERASE
GLOBAL
ESCAPE
RETURN
SIMILARITY - .59
PRO - 42

Figure 1: Before and After knowledge structures for a person learning LOTUS.

After three weeks of training, trainees show evidence of an organization that is becoming meaningful for LOTUS. Not only are the graphs describing their knowledge structures more like those of the experts, but other measures as well showing increasing similarity. There is also evidence that these structures change with different contexts and goals. What these studies imply is that simple trees can be used for particular contexts, plans, goals, and purposes. Changing the context requires a change in the command structure, to form a new hierarchy or tree. Since these new trees have many nodes in common with the old tree, it implies an overall structure that is a large network of concepts. Such a complex system can be fully described in a hypertext network.

Notecards

The Notecards (Halasz, Moran, and Trigg, 1985) system is a hypertext environment designed to aid in the collection, structuring, and analysis of textual and graphical information. The metaphor upon which Notecards is based is that of notecard (e.g., 3 by 5 cards) and fileboxes. A Notecard is intended to contain a small, single, idea-sized chunk of information, in either textual or graphic form. A collection of Notecards can be arbitrarily linked together to form networks that convey the relationships among the ideas stored in the various Notecards. Each link in the network has a type associated with it indicating the kind of relationship that exists between pairs of ideas. Multiple links in many directions are supported. The links are visible within each notecard as a hotspot title or icon that is mouse-sensitive and pops up into the full text or graphic when it is buttoned. There are many card types, each supporting one of the full range of functionalities available on the lisp machine.

NoteCards offers designers a tool with which to organize, manipulate, and structure their ideas. Furthermore, it permits the designers to analyze the structure of their ideas, revealing not just the content of their knowledge, but also the way in which facts interrelate. It gives the designer the ability to view the contents of a knowledge domain from any one of a number of perspectives. The multiple card types of Notecards - editor, text, graph, sketch, simulation, student model, instruction, etc. - organize and provide a common user interface to many functions. We envision that a Notecards design in a frame-like environment will lead to automatic construction of objects and simulations, and we are actively exploring this view in a series of design and development environments arising from Notecards. We have

ES

e of Action (ENCOA)

slations

German Report

Enemy Location We probably
an unkown MRD consisting
All three regiments are equip
T-64 Panzer and are reporte
mounted. They are at 75% st
morale is considered to be ve

Deutsche Bericht

Wahrscheinlich treten Wi. einen unbekannten Feind entgegen, der drei MRR hat. Alle drei Regimenten sind mit T-64 Panzer ausgerüstet.

Enemy Course of Action (ENCOA)

Enemy Location

Schiessender Panzer

TRANSLATION: German Report

Wahrscheinlich

adj. probable, likely,

adv. probably, likely,

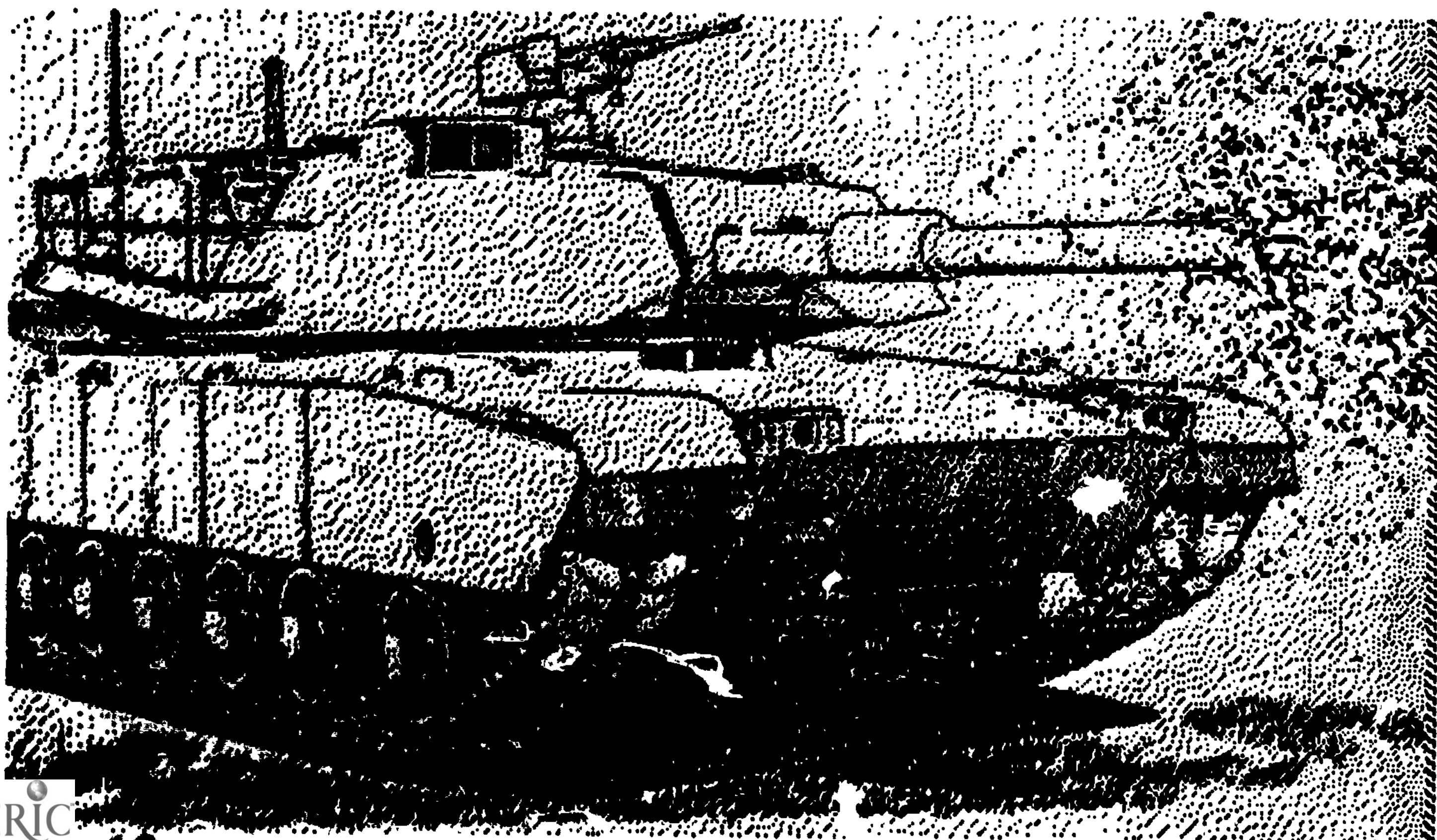
Enemy Course of Action (ENCOA)

FILE BOXES

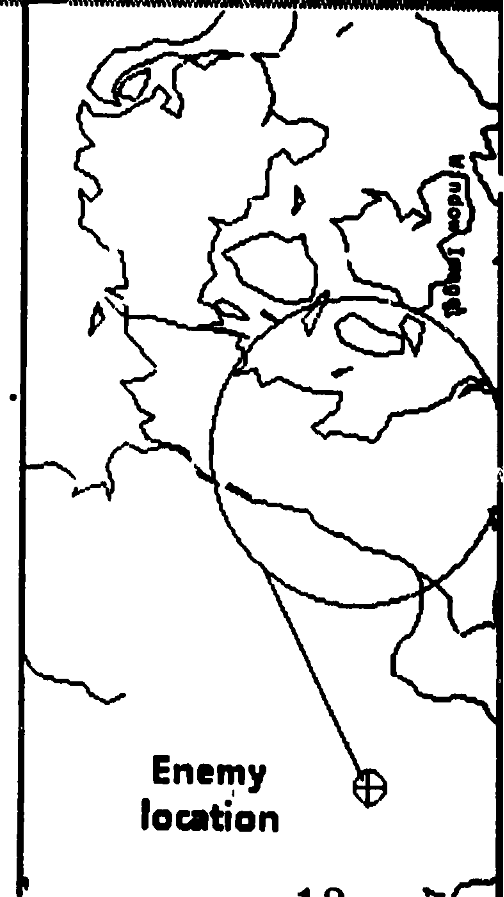
NOTE CARDS

mplete.

anzer



on



called this environment IDE (Instructional Design Environment) and it is available on a research basis (Russell, Moran and Trigg; 1988). We are currently developing multilingual courses for the 97E military occupational specialty (MOS) in English, French, German, Spanish, Russian, and Korean to be taught interactively using this environment. An example of this structure for German is given in Figure 2.

FIGURE 2: A sample IDE screen for German, showing hypertext hotspots, graphics, andanimations.

Smart Computer Assisted Language Learning Environments

Training newly assigned soldiers and officers in relevant languages imposes a continuing problem of maintaining productivity in the schoolhouse and on the job. Junior personnel arriving from a training school often have good language skills but little background in the specific job assignment. Subject matters, such as preparation of the battlefield, operational orders, and interrogation are usually not covered by language instruction; so for the new personnel there is an imposing array of forbidding terminology, unusual grammatical usage, and new semantic structures; even in his or her native language. As a result, senior and more experienced personnel must devote a great deal of their time to on-the-job training repeatedly with each new individual. Hardcopy working aids, to be effective, must be complex, and they are thus difficult to develop. They are prepared sporadically, but are difficult to maintain and update.

Several compelling technologies are reaching the state of ready application to solve these problems. Chief among them is the computer: fast, new processors are becoming readily available. Computer technology is quickly advancing on a number of frontiers: VHLSI, portability, graphics, C D - ROM, videodisc, speech synthesis and understanding, and high density memory. In addition to the computer, Cognitive science is establishing methods for making explicit poorly understood processes such as natural language processing, second language learning, making inferences, decision-making, problem- solving, intuition, and pattern recognition. These are powerful techniques for creating general learning strategies and advanced cognitive skills. Finally, the technologies of Natural Language Processing (NLP) are beginning to show themselves particularly suitable to the task.

Student Modelling and Grammars

The problem of describing linguistic competence is as old as mental testing, and, given the venerable traditions of this field of research, many alternatives for how to proceed in many promising directions have been proposed. But understanding language competence lies at the heart of understanding cognitive processes of all sorts, so one should not be too surprised that novel approaches are even yet under development, for this is an enterprise of great scope and complexity, whose ultimate paths are almost certainly still unexplored. In this paper we offer some evidence and more rationales for yet another opportunity to explore the heart of language. We propose that computational linguistic formalisms have reached sufficient maturity that they offer an unparalleled precision for describing the structure of students' linguistic knowledge, and we suggest some directions for incorporating these student modelling techniques into standardized tests.

Over the years the corpus of available theories to account for grammatical skills and language learning have increased steadily. Yet, it has turned out that it is nearly impossible to select which grammar best fits American language as we know it. Some investigators (e.g. MacWhinney, 1982) have argued that a relatively large number of very different mechanisms are needed to describe language competence. Others, (e.g. Pinker, 1984) have proposed that one formalism, indeed any formalism selected by sepcific enough constraints, should be nominated for the task of describing linguistic competence in a cognitively verifiable way, and the enterprise of justification and competitive argumentation and experimentation to substantiate these decisions begun in earnest. This latter view exposes the spirit of the present paper.

Instructional Constraints

Our conception of the enterprise is based on the view that language formalisms offer a tremendous opportunity for improving foreign language instruction. We begin with the idea that the language knowledge and skills that a student possesses can be modelled by a carefully selected grammar formalism. Where that model differs substantially from a more complete and consistent description of the base language, one can begin to focus instructional interventions. The effectiveness of this intervention then reflects on the validity of the grammar formalism.

We call this view the instructional constraint in analogy to Pinker's (1984) learnability constraint. Whereas his constraint operated on the adequacy of a formalism to express the graduations in language competence as a learner progressed in a developmental sequence, our constraint operates on the adequacy of a grammar to capture the misconceptions and bugs a learner brings to the process of learning a second language. Our constraint also highlights the perspicuity of the formalism for designing instruction. If the notation for describing a learner's state is so opaque that one can make no adequate descriptions upon which to base instruction, then a better formalism must be sought. Again, if the constituents of the formalism cannot be meaningfully employed in designing remediating instruction, then a better formalism must be sought out. However, given that instruction is implemented based on the grammar's articulation of the language, and it results in effective instruction, then one can argue that the formalism provides a psychologically meaningful interpretation of the learner's linguistic knowledge. It is of course possible that the instruction is effective for many unrelated or spurious reasons: there are many similar rationales that might be constructed from fundamentally different evidence. For instance, the grammar's rules may simply capture and formalize intuitions that are superficially obvious, such as : a noun phrase begins a declarative sentence. Conflict with a rules such as that would result in fundamentally incoherent speech; yet practically any grammar would make the prediction. As is usual in this experimental, cognitive enterprise it is important to find meaningful predictions that are not obvious and perhaps even counterintuitive. At the moment, this an untested hypothesis at many levels.

In general, we have proceeded within a framework created by unification grammars. As described in the ensuing sections , these grammars use unification as the fundamental mechanism for deducing a grammatical pattern, and have been implemented in various forms call FUG, LFG, GPSG, DCG, and PATR. These grammars are generally acknowledged to be the most promising linguistically for capturing the rich variation of language, and they have all generated very useful computational environments that may be used for creating instruction.

Grammar Representations

Technological advances in this discipline have provided powerful formalisms for representing natural languages in computer systems. These representations promise to address the issues of lexical, morphological, semantic, and syntactic analysis required by an intelligent CALL environment. Parsing strategies which access and manipulate the language representations will address syntactic, semantic and pragmatic (to some degree) analysis. An important question is which combination of grammar formalisms and parsing strategies will provide a suitable learning environment in intelligent CALL? We review several current grammar formalisms and parsing strategies below.

Semantic Grammars. A semantic grammar is based on a simplified conceptual dependency construct (Brown and Burton, 1979). The grammar is organized as a set of attribute specific semantic categories for a given domain. This construct has been used in early natural language application for handling question asking about some specific topic. In NLP the grammar defines the allowable patterns that can be matched with a given input string. These legal patterns restrict the form of the natural language which cuts down on ambiguity and provides surprisingly good language coverage. A sample semantic grammar and a query to parse might look like the following:

<s> - - - > What is <ship-property> of <ship>?]

<ship-property> - - - > the <ship-prop>|<ship|prop>

<ship-prop> - - - > speed|length|type

<ship> - - - > <ship-name>

<ship-name> - - - > Kennedy|Kitty Hawk

Query: What is the length of the Kitty Hawk?

A semantic grammar is context-free and can be parsed by any of the several existing context-free parsers. The contextual semantics of a given input string are derived from the categories in the grammar. Key-word and pattern-watching strategies are used to parse and answer queries. While this mechanism allows for natural language interaction, there is no flexibility for syntactic deviations in question form. Lexical variations will also be constrained to the vocabulary represented in the semantic grammar.

In order to get more flexibility in natural language input and parsing strategies, separate representations will need to be encoded to represent the lexicon, morphological tables, phrase structure rules and grammar. This brings our discussion to another set of natural language processing representations. While the unification formalisms to be discussed next allow for rich natural language interaction, they pose a different kind of problem in parsing in regard to semantic and pragmatic language contexts.

Unification Formalisms.

Grammar formalisms are representations used to describe a language, its set of sentences, the syntactic features of the sentences, and their semantics. These formalisms have been constructed to help linguists understand universal linguistic theory as well as provide a means for computational interpretations of the grammar being studied when it is represented in some computer system. A unification formalism is one that prescribes a given operation (pattern-matching) between an input string (a sentence) and the grammar. Schreiber (1987) discusses the essential assumptions of a unification formalism:

1) it is surface- based with a direct relation to the surface order of the sentential elements

2) it associates the sentence string elements with some relevant information (knowledge) domain

3) it is inductive, defining the association between an element and the information domain recursively and according to a specified inference procedure

4) it is declarative, defining permissible associations, not how the association is computed

5) they are context- feature- based in that associations between features and values are taken from a well defined, structured set Examples of unification- based grammar formalisms include: Functional Unification Grammar (FUG), Lexical- Functional Grammar (LFG), General Phrase Structure Grammar (GPSG), and Definite Clause Grammar (DCG).

The FUG representation uses functional structures as generalized feature/value pairs to denote linguistic information. Patterns and constituent sets are used in the matching or unifying of the sentence elements with the grammar. FUG also introduces disjunction into the grammar which allows for getting around any potential constraints imposed by the unification rules. In contrast to the LFG formalism below, FUG deliberately blurs the distinction between constituent and functionalstructures . This formalism emphasizes the functional description of the language with the constituent structure specified by a means of patterns rather than phrase structure rules.

LFG was developed primarily as a linguistic theory (Bresnan, 1982) and then became recognized as a useful formalism for represent- ing natural languages in computer systems. LFG is based on a model of syntax that is not completely structurally- based. The formalism describes grammatical functions which are represented by functional (F -) structures. An F - structure for a given sentence describes the appropriate grammatical functions or role for the sentence constituents. Constituent (C -) structures are part of the formalism that represents the syntax of a sentence (see Figure 4). The lexical component of the theory emphasizes the commitment of LFG to characterize semantic relations. LFG can thus analyze grammatically at the lexical level. This formalism more than the others described here has been used to investigate psycholinguistic issues in relation to language processing systems.

GPSG also came into being from a linguistic analysis motivation. This formalism attempts to retain a formally restrictive system of feature structures that can handle a wide variety of semantic and syntactic phenomenon. Unification procedures are also highly restructured. A given phrase structure tree for a sentence will satisfy various rules in the GPSG grammar based upon specific principles (control agreement, head feature) in the formalism.

DCG arose from work in the PROLOG programming language (the other formalisms mentioned above are generally written in LISP). The formalism uses term structures rather than feature structures for a given sentence. Terms are the informational elements in DCG with the unification principles derived from the theorem- proving mechanism inherent in PROLOG. Examples of terms are:

s (head (subjHead, form)))

np (subjhead)

np (head (subjHead, form)))

Terms differ from the feature structures in FUG, LFG and GPSG in that they identify sentence constituent values by their order in the term structure rather than their association with the feature.

Parsing Strategies

Parsing is "searching a space of possibilities" in an input string (Kay, 1985) as the string is matched with the grammar. A language is a set of strings made up of finite symbols. These are referred to as terminal symbols. A context-free grammar is a formal device that specifies the strings in a given set. This grammar also uses non-terminals that represent syntactic classes A sentence has a form consistency of a series of terminal symbols. When parsing occurs, an algorithm to search the string and make matches between its elements and the grammar is performed. There are many strategies one can use, transition network parsing, top-down parsing, bottom-up parsing, deterministic parsing and so on. Which approach is used depends on the domain and the goal of the natural language application.

An augmented transition network (ATN) (Woods, 1970) is one way to parse language going from an initial state (the first word in a string) to a final state (the last word in a string). ATN's use the nodes in a network of language elements to represent parts of speech or phrase names. The arcs between nodes are used to implement certain actions when they are transversed (see Figure 5).

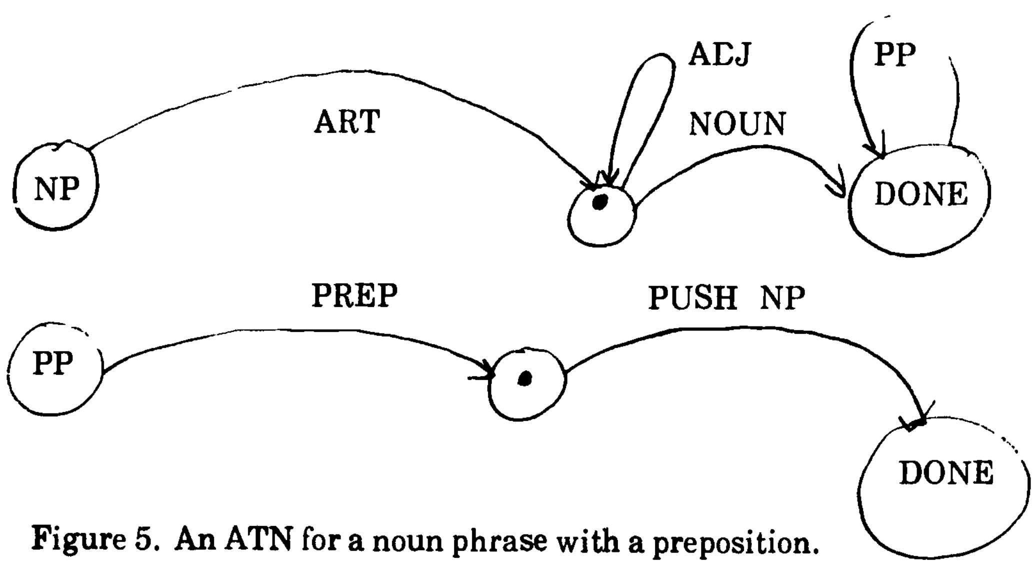


Figure 5. An ATN for a noun phrase with a preposition.

Augmenting such a network allows for three kinds of transitions:

1) Adding arbitrary tests to the arcs such as agreement of a word and its modifier.

2) Adding structure- building actions to arcs to be used by the parser to determine semantic analysis or active- passive sentence transformations

3) Allowing subroutines to be called on arcs. Forexample, in Figure 2, a preposition (PP) is attached to a noun phrase by a subroutine call to PP.

Using Grammars for CALL

Our first steps in developing these ideas and intuitions into an environment for second language learning began with a system developed by researchers at Xerox PARC and AIS using the LFG environment created by Ron Kaplan. In this system, called CALLE (Computer Assisted Language Learning Environment), a simple Spanish grammar was created to provide an environment in which student and computer could interact by holding a realistic topical conversation in Spanish about occupations and foreign places (Feuerman et al., 1987). The focus of the embedded lessons was on the distinctions between two Spanish verbs of being: ser and estar.

Within the dialogue window provided by the system, the student and computer can interact within the limits of the underlying grammar. If the LFG system can parse the student's responses the dialogue continues gracefully. However, where no complete parse is available (according to the built- in definitions of LFG) a control system built in LOOPS (an object- oriented programming language specific to Xerox) tries to determine the errant rule (e.g. subject- verb number agreement) and provides some student feedback. The feedback is not detailed nor carefully articulated, but it is a beginning of the sort of capability that we foresee as possible within these environments and in need of experimentally rigorous analysis.

The system as it stand merits further detailed description so that its implications and possibilities will become clearer. At this time we can only offer a very rudimentary estimate of these kinds of possibilities. Consider for instance, an application of this system to the teaching of verb conjugation (much in the spirit of Fum, Giangrandi, and Tasso's (1988) system for teaching English verb tenses, ET (English Tutor)). As an exercise generator, we can provide students with sentences like:

" Las muchachas de la clase (are) todas muy bonitas."

This example tests an understanding of the indicative use of "ser". But because we are using a general parser (LFG) to examine the correctness of the sentence, we could provide the English replacement of all or any of the words and the parser would then find any and all errors a student made in the translation exercise.

A fair amount of fundamental research has been deployed with the goal of using NLP techniques for language instruction. Yazdani (1986) provides a short review of the topic.

A brief description of the CALLE is given in Figure 4.

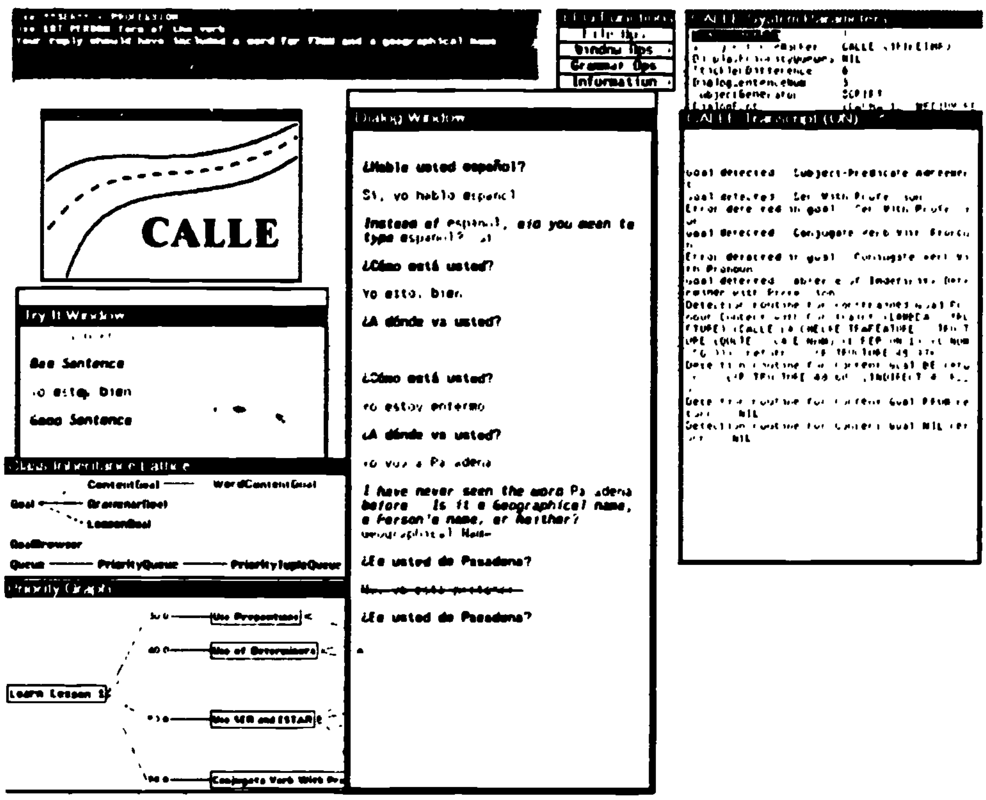


Figure 4: A sample CALLE screen showing the main windows.

Overview of CALLE

CALLE interacts with trainees through a number of functional windows visible in Figure 4. The main Dialogue Window provides the key functionality: on-line exercises in which trainees can engage in realistic foreign language dialogues with the machine. These dialogues are monitored by special rules to determine which immediate lesson goals have been exercised, and which remain to be tested. The machine deliberately generates dialogue contexts to exercise the remaining goals as the conversation proceeds. Other windows allow trainees to use the foreign language without being directed by the machine's specific dialogue goals (the Try It Window) but with diagnostic comments. It is this window that is used when exercise examples such as the one above are given.

Integrating CALLE and IDE

The hypertext system of IDE provides the perfect complement to CALLE for training systems. IDE can create the structural linkages and detailed, sometimes graphic explications of structural, declarative knowledge; and CALLE offers the environment for exercising and refining the skills that use that knowledge. It is our intent to integrate these two kinds of environments seamlessly into a coherent and efficient instrument for language training. In particular, we plan to focus our efforts on creating large semantic networks of words in different languages, according to the taxonomic structures natural to those languages. These taxonomic structures can provide the base for translating automatically from one language to another. They can also point out the places where knowledge of one language is likely to interfere with learning the semantic structure of another language. In the long run, these semantic structures will be necessary for a fully functional natural language processing machine translation system.

This environment continues to offer us a picture of how to use technology for second language instruction that points out its strengths and weaknesses. Monitoring students to try to build a genereal student model of their learning s..ategies and skills so that their learning can be individualized and strengthened has been particularly difficult. Our present understanding does not allow us to decipher the particular skills and prerequisites to use for managing the course of instruction and the particular techniques that might be most effective. It appears that the current system does not begin to deal with the complexities of either language or learning. It offers a particularly astute beginning that must be extended greatly.

One of the salient shortcomings of this approach is that explanations are very cryptic and generally canned. Although the dialogue is fairly natural (if somewhat restricted to narrow topics) it is difficult to generate feedback that is particularly meaningful to sudents. There is no general body of declarative knowledge to which the system can direct or offer a student for remediation. A simple technique would be to tie this system as a set of exercises that complement materials presented in a specific text. This is the route that Anderson and his colleagues (1985) have chosen with the Lisp tutor. The tutor exercises procedural skills, but declarative knowledge structures are built up with a standard text.

Although the system contextualizes its comments and creates a meaninful body of discourse and communication, no way is provided for expanding the context to include pictures and animations that may clarify the meaning and reduce ambiguities. In general, the system offers an arena for making use of one's knowledge in order to eliminate the grammatical bugs and inconsistencies that may have arisen; but it offers no real support for extending the knowledge of grammatical and lexical items. It is for this particular contextualizing of knowledge that hypertext sytems may be better suited.

HyperLexicon and LEXNET-INSITU

We have continued our exploration of Hypertext as an instructionally useful environment for second language learning with some protoypical environments built using Apple's Hypercard environment. The first attempt (Bui, 1988) created

HyperLexicon, a hypermedia- based lexicon for vocabulary acquisition in a foreign language. It presented a conceptual design and several examples of coupled semantic / hypermedia links and their use. Starting from the basic question of what it means to learn the meaning of a word, and asking how this learning process could be improved, we were led to a system which represents a generalization of the standard vocabulary acquisition tools, the dictionary and thesaurus. It appears that "learning the meaning of a word" involves not only knowing when the word applies but when another word is more appropriate. For example a guard in front of Buckingham Palace is more correctly said to be "marching" than "walking".

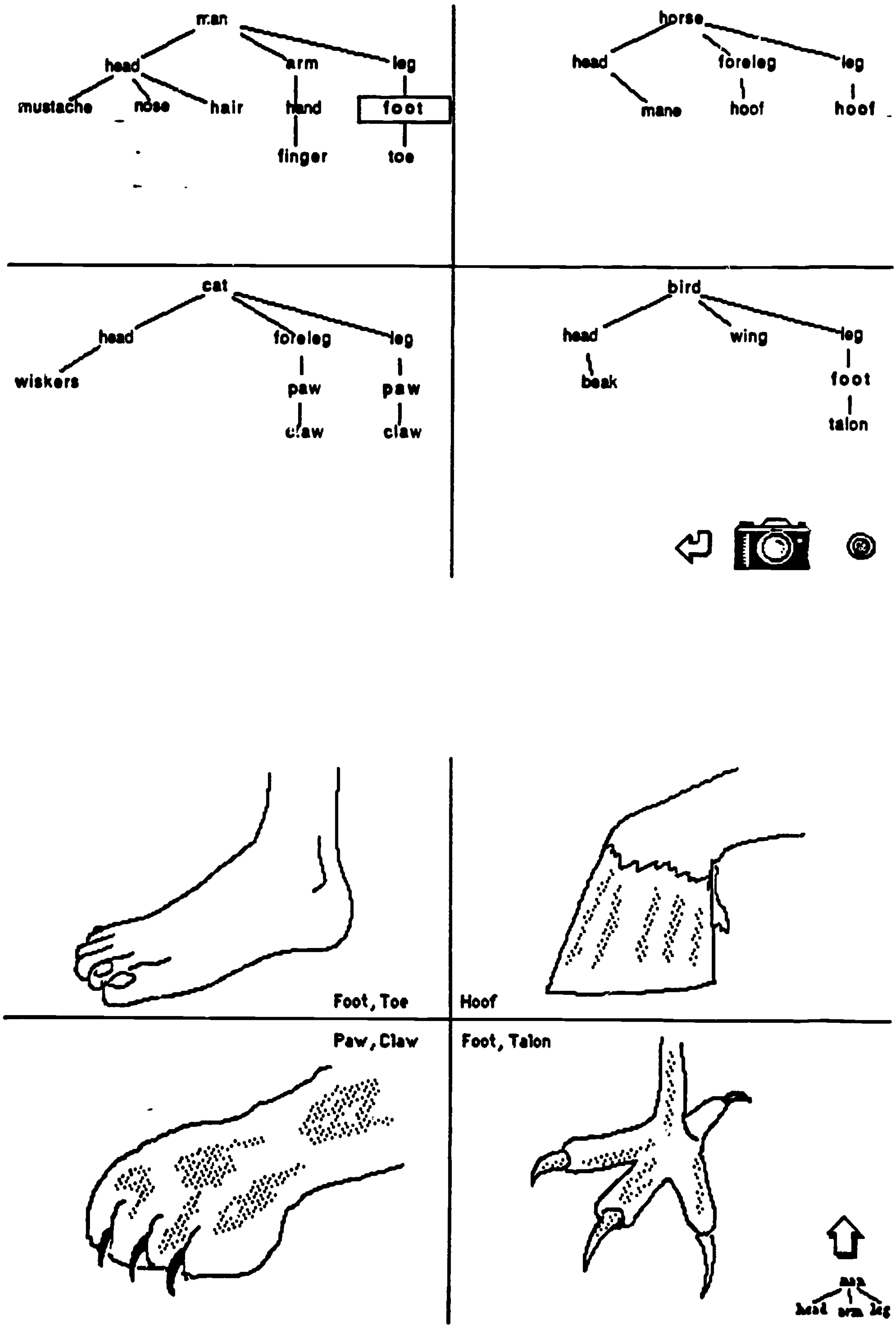
In learning the meaning of the words, we drew on the extensive research carried out by George Miller and his associates (Miller, Fellbaum, Kegl, and Miller, 1988) and the implementation of his theory in an on - line lexical reference system, Word' , which attempts to mirror the organization of human lexical knowledge.

WordNet is a computer- based lexicon that supports conceptual relations of hyponymy, meronymy and antonymy, i.e. ISA and Is - Part and Opposites. Lexical information (hyponymy and meronymy) are represented by a semantic hierarchy extracted from a collection of dictionaries and thesauruses, liberally seasoned with linguistic intuition. In the future, this kind of information may be extracted from a machine- readable dictionary by automatic and semi - automatic procedures (Amsler 1980). HyperLexicon's goal is to utilize these kind of semantic hierarchies to organize lexical knowledge and to provide an environment where students can access and aquire that knowledge.

We constructed the system to minimize, but not eliminate, reliance on a student's native language skills. While we recognized that there are instances when cross-language instruction can be useful we tried to reduce the amount of duplication of HyperLexicon data, and so we relied heavily on more universal graphic and aural communication. In Peirce's (1965) terms we focused on icons and signs instead of the symbolic relationships. In addition, many of the interfaces we implemented in the demonstration prototype display semantic relationships graphically and so depend on the student's understanding of these easily learned graphic conventions.

Using the several interactive features of this environment (see Figure 7) students can explore audio/visual components which are related through Hypermedia links corresponding to the ISA and Is - Part relations. The student can also explore the definition of concepts lending themselves to graphic or aural depiction. Although this is illustrated in the demonstration through painstaking hand drawings, the technology exists to easily digitize existing hardcopy imagery, or video signals as well as store vast amounts of visual or aural data on CD ROM.

Another potentially useful method for learning words is to understand analogous concepts. For example: "Hand is to man as ?????? is to horse?" This type of learning is aided in the HyperLexicon by presenting analogous concepts in a manner the student can grasp easily. Examples of this include graphic presentation as in Figure 8, taken from the demo. A somewhat different type of analogy which HyperLexicon supports is native language dependent; namely, analogies between the native and target language. These are depicted graphically in a manner which uses geometric proximity to indicate semantic closeness.



man
head
arm
leg
mustache
nose
hair
hand
foot
finger
toe
horse
head
foreleg
leg
mane
hoof
hoof
cat
head
foreleg
leg
wiskers
paw
paw
claw
claw
bird
head
wing
leg
beak
foot
talon
Foot, Toe
Hoof
Paw, Claw
Foot, Talon
man
head
arm
leg

Figure 7.

The teaching aid which we originally envisioned when thinking about HyperLexicon functioned like a generalized thesaurus, using the semantic relationships discussed earlier, in addition to the concept of semantic "nearness" used in a standard thesaurus. Two ISA trees of concepts to be learned are presented graphically and the student can use Hypermedia links as described earlier to interrogate the environment and receive answers in graphic dialogues.

HyperLexicon is primarily a tool for learning by exploration. It is possible however when designing HyperLexicon courseware to create several "levels" of a lesson which could then be offered to students at their or the tutor's (man or machine) discretion.

Immersion

As Henry Hamburger (1988, working paper) points out, immersion techniques for SLA have the important property of blocking convenient but short- sighted translation strategies. From the perspective of this paper, translation strategies essentially prevent the transformation of rules internalized for L1 grammar into an appropriate form for L2 grammar (Lytinen and Moon, 1988). In order to modify these existing rules effectively the comprehension techniques of the L1 grammar must be invoked on L2 sentences in a problem solving context that facilitates their transformation.

Immersion demands explanations in context. using the target unknown words and phrases. It also demands instances as objects , not as symbols. One way of satisfying these contextual demands is with physical immersion. Another, technological way is through contextual enrichment using computer- based systems with videodisk and graphics. Focusing on context also makes it natural to produce words in consistent discourse structures: sentences, paragraphs, and especially conversational interactions to complement graphics and real- life sounds.
There is another way of getting at these same strengths for learning environments: semantic mapping.

SEMANTIC MAPPING.

As Rebecca Oxford (1988, in progress) so carefully and insightfully describes, semantic mapping is a technique that has sprung up with some popularity among language teachers, without the existence of a strong theoretical framework that explains its power. Yet, what does cognitive research tell us about nature of word knowledge? It takes at least partly a network form with semantic and associational links between words (Miller, 1988). What, by coincidence, has Foreign language pedagogy been using as a vocabulary teaching tool? Student generated "maps" that resemble associative networks of words. Given their potential usefulness and validity, how might maps be constrained so as to better reflect and instil the associations and relationships that language users will need when producing or interpreting words? In beginning to think about the issue, I feel more and more that the semantic mapping techniques act as a kind of bridge between the purely symbolic system of language with its grammar, syntax, morphology, etc.; and the more realistic (Peirce (1965) called it a sign system) of images and icons. In fact the semantic map could be thought of as an icon, a reduced, simplified image.

The frame of reference that seems best suited to understanding the relationships among these systems of language and thought, is that developed by C. S. Peirce (1965) in his theory of semiotics. He thought that there were three main components of thought:

1. a sign, such as icons, indexes (such as smoke for fire), and symbols;
2. an object, such as physical objects, but also relations and properties;
3. a meaning, such as the translation of a proposition into another form most directly applicable to self-control. In other words, it is a restructuring such as one finds in a semantic network of concepts.

Words as a symbolic system have their meanings inherent in their relationships, between themselves. They also have a referent in the outside world and images and icons can act as intermediaries between these symbols and objects.

The visual connections betwen words in a semantic map especially when accompanied by icons and images (but even without them), makes this transitional bridge easier to cross. Semantic maps foster and facilitate the connection between symbol and referent.

Another important point: semantic maps begin to offer a notation for describing semantic relations. As we all know, a notation is indispensible for communicating ideas; so semantic maps facilitate the communication of ideas about the structure of language.

Semantic maps are very rich structures, almost like image images can be parsed in a very large number of ways - - active, passives, different perspectives, etc.. Semantic maps have this richness about them too. By relating two nodes in a particular way, there are imbedded relations with all the other indirectly connected nodes. So for instance, in a graph of the parse of a sentence; the graph itself is not as ambiguous in terms of the actual relations among the words as the original sentence: The relationships have been made explicit, and the ambiguity reduced, hopefully in a way which is consistent with the way people actually parse or understand the sentence. So a parse of a scene more or less representative of the sentence is likewise less ambiguous than the scene itself. A semantic network of the words in the sentence, or the words describing referents in the scene, is less ambiguous than the sentence or the scene, but more ambiguous than the parse or a single understanding. Something of the structure of a parse can be imbedded in the semantic network: e.g.; component terms of the head noun phrase can be graphed to modify or have relations with each other and with terms in the verb phrase too; and all these relations can be captured by the graph in a very succinct form. So too can the part/whole or antonym or synonym relations that one can define in a semantic map of nouns and adjectives.

One can enrich a semantic map of nouns with relationships developed from verbs and prepositions. Prepositions in particular offer an almost purely relational vocabulary:

on, before . after, during, beside, opposite, to, from; etc.

Conjunctions as well provide a rich relational vocabulary: and, because, therefore, etc.

My second comments focus on why there is this impulse to create a semantic map? In part, as I haved just suggested above, a map presents an interesting compromise between the meanings that one paragraph could hold and the range of possible interconnections that the same words could have, and conventionally do have, with each other. But now, lets focus on the graphic interconnection. Using the perceptual system to bring these words together has some pronounced benefits, particularly when some of the words are also represented as pictures. Themeaning is grasped more or less directly (in a different form than if the words were presented only symbolically in sentences. Pictures represent an intermediate form of reality between symbols as words and direct perceptions of the objects of the world. This intermediate form of signs with their own logic of interconnection has a psychological reality. As recent studies of brain damaged individuals have shown, people can have semantic deficits with symbols that are removed by showing the pictures. For instance, a brain damaged individual can be asked what the word "elephant" means and respond with only "an animal" even after being prodded for more information. But when shown the picture of an elelphant; the same individual will respond with a description of its habitat, characteristic sounds, actions, and a wealth of semantic information. Just so, perhaps, the semantic map can trigger a host of associations and semantic structures within the cognitive lexicon that a simple word list or constrained paragraph could never elicit.

So the drive to create semantic maps may perhaps be seen as motivated by a desire for a representation that bridges reality and symbol with a notation that verges on the sign of semantic meaning.

My third focus is on the definition of semantic mapping within the constraints of second language learning, focusing on learning and teaching. Here I would like to place it in a continuum that stretches to other carefully refined formalisms structured to ease a student's task of learning a complex language and leveraging existing skills. Just as teachers and texts predigest the knowledge structures of the second language by e.g. grouping verb declension types; artificially dealing with tenses one at a time; grouping prepositions and conjunctions for easing learning; so too the general lexicon can be dealt with by grouping words according to a sekeletal outline that can absorb the entire range of communicative contexts. This emphasizes the great care that must be taken to group words into category structures that provide sufficient overlap and context so that inferences about word meanings become easier and more accurate.

For instance:

```
                bird
         /       |      \
   robin        owl    ostrich
    /            |          \
redbreast      hoots     flightless
```

may be more difficult to learn and remember than the more richly interconnected:

```
                bird
   /             |            \
robin        pigeon      sparrow
/                |              \
redbreast    coos         common
```

This kind of issue rises to particular prominence when the semantic map is created on a hypertext system, where not only teh terms and symbols are mapped; but each node and link is an interactive hotspot, whose selection can lead into ever-expanding domains of graphics, sounds, animations and even more detailed hierarchies. There is a naturalness to the type hierarchy of the structure of these concepts that is the focus of much concept learning research (see Fisher, 1988) and that too ought to be taken into account.

Selectional Constraints

Any recognizable object conveys a common noun that represents a natural type (Smith and Medin, 1981). Picking the proper representation for L_2 represents a problem whose difficulty depends on the similarity of the L_1 and L_2 type hierarchy. For many common nouns in related languages, this type hierarchy is very similar: e.g., "dog" is closely matched by "hund" except for relatively rare archaic, or colloquial relations. Verbs have type hierarchies too; with syntactic constraints explicitly interlinked with semantic constraints. For instance, coos can only be said of individual birds (mainly pigeons); pecks can be used with birds as a whole class; eats can be used with any mammal; ingests with many things; nourishes with many more; consumes with even more (such as fire). Both types of selectional constraints work in concert with the noun type hierarchies to create powerful differences going from one language to another. For isntance, the simple noun America, a proper place name has a host of selectional constraints that differ from one language to another:
In English:

I am going to America, is roughly translated

In German:

Ich fahre nach Amerika.

The word for the United States in the two languages is so similar, one could guess its counterpart with little understanding, but the syntactic and semantic selectional constraints in these two simple sentences are radically different. German presumes that the act of "going" when combined with a country place name presupposes travel with a conveyance of some sort and demands the use of "fahren" (roughly "travel") How does one learn these selectional constraints of words? They are basic to the words approptiate set of meanings. It seems that they arise naturally from an understanding of the context of use of a word; but how is that context most speedily and effectively conveyed to a new learner?

George Miller's research program on learning strategies for vocabulary offers a solid model for research on lexical acquisition in SLA. Gildea, Miller, and Wurtenberg (1988) created a very rich hypertext environment for exploring word meanings and how people learn them from context. Using fairly uncommon words with fifth- and sixth-graders, they examined how effective word definitions, pictures, scenes, and illustrative sentences were for learning novel words. Learning was assessed using multiple choice tests for the best definition and the best sentence using the novel word; and with production tests that asked students to generate definitions or illustractive sentences. Overall, they found that illustrative sentences produced superior learning , particularly when compared to dictionary definitions. It appears that student are unaware of their own learning abilities, since their first choice of explanation was always for dictionary definitions. Applying Miller's basic finding that definitions, context-preserving sentences, and pictures all serve their particular purposes for lexical acquisition to Second Language Acquisition (SLA) appears relatively straightforward, but there are some interesting wrinkles we may first want to play with. From his work it seems that definitions are the recourse of first choice for children who do not know a word (probably because they have been hit over the head so often about using dictionaries!) but they are not really effective for generating appropriate sentences (although they do help to select good definitions). Illustrative sentences are just as good for selecting definitions, and they are superior for using a word properly in a novel sentence. Illustrative sentences also appear to challenge and motivate students somewhat more to understand and perhaps even generate their own definitions. Pictures were not as cleanly useful; but they appear to clarify the meanings of words encountered for the first time - - - a likely frequent situation with Second Language Acquisition (SLA).

These implications for SLA are rich but fraught with uncertainty. In learning a new foreign lexicon, are adults at an advantage over children, since they have already acquired the relevant semantic structures, or are they at a disadvantage because L1 and L2 carve up knowledge differently? If it depends on the relative similarity of the two languages, will we ever be able to fix this similarity quantitatively, using rule-based grammars?

Implications for Aptitude Testing:

Well, what was this all about?

1.) Aptitude depends on the measure of learning as a final validation. Obviously if we can radically change the way languages are learned, we may have a profound effect on aptitude tests.

2.) If aptitude testing is to be principled, it needs a theoretical understanding of the skills and knowledges needed for the task to be tested. I hope we have shown that there is the beginning of such an overarching theory.

3.) Student modelling is very much related to aptitude testing. As student models are improved they may provide an ongoing form of such an assessment in a rich environment that integrates validation with assessment in an ongoing spiral. What better way to assess the skills and abilities of a student than by presenting him or her with the next problem to solve or the next real life test of authentic language skills?

CONCLUSION

As these two powerful technologies, hypertext and natural language processing, develop they will support each other to create systems of unrivalled efficacy for Army use. Undoubtedly language training and support are only one small part of the fundamental transmutation that will reshape our working environments; but their effects are significant and provide a microcosm in which we can see the outline of even more profound changes to come.

References

Allen, R., and Psotka, J. (1985). Artificial intelligence for executives. Army Research, Development & Acquisition Magazine, November- December, 1- 3.

Anderson, J. R. (1983). The architecture of cognition. Cambridge, MA: Harvard University Press.

Anderson, J. R., and Reiser, B. (1985) The LISP Tutor. BYTE, 10, 4.

Ayuso, D.M., Shaked,V., Weischedel, R.M. (1987) An environment for acquiring semantic information. Proceedings of 25th ACL, 32-40.

Barchan, J., Woodmansee, B. J., and Yazdani, M. (1985) A Prolog -based tool for French grammar analysers. Instructional Science, 14, 22-66.

Bresnan, J. (Ed.) (1982) The Mental Representation of Grammatical Relations. Cambridge, MA: MIT Press.

Bui, Khanh. P. (1988) HyperLexicon, a Hypermedia-based Lexicon for Vocabulary Acquisition, MS Project, Computer Science Department, George Mason University. Also USARI Technical Report # , 1989.

Bush, V. (1945) As we may think. The Atlantic Monthly, July, 101-108.

Caramazza, A. and Brones, I. (1980) Semantic classification by bilinguals. Canadian Journal of Psychology, 34, 77-81.

Clancey, W. J. (1986) Qualitative student models. Annual Review of Computer Science, 1, 381-450.

Collins, A. and Quillian, M. R. (1969). Retrieval time from semantic memory. Journal of Verbal Learning and Verbal Behavior, 8, 240- 247.

Conklin, J. (1987) Hypertext: An Introduction and Survey, IEEE Computer, Sept 1987, 17-41.

Cruse, D. A. (1986) Lexical Semantics. Cambridge University Press.

Feuerman, K., Marshall, C., Newman, D. and Rypa, M. (1987) The CALLE Project. CALICO Journal, March, 25-34.

Fletcher, D. and Psotka, J. (1986) Intelligent training systems for maintenance. Signal, 89 - 95.

Forsdick, H. (1985) Explorations in real-time multimedia conferencing. Proceedings of the Second International Symposium on Computer Message Systems. Washington, DC: IFIP Technical Committee on Data Communications.

Fox, Jermy. (1988) From instructional to facilitative approaches to CALL. (THIS VOLUME)

Fum, D., Giangrandi, P., Tasso, C. (1988) The ET project: Artificial Intelligence in second language teaching. Proceedings of the European Conference on Computers in Education, Lausanne, Switzerland.

Gildea, P. M., Miller, G. A., and Wurtenberg, C. L. (1988) Contextual Enrichment by Videodisc: a First Report. Cognitive Science Laboratory Report No. 12, Princeton University.

Gray, W. D., Psotka, J., and Pliske, D. (1985) Smart Technology for training: Promise and current status. (RR 1412). Alexandria, Virginia: Army Research Institute.

Griselli, A. (1988) Towards linguistic competence in L2. (THIS VOLUME)

Halasz, F., Moran, T., and Trigg, R. (1985) NoteCards Release 1.2I Reference Manual. Palo Alto, California: Xerox Corporation.

Kay, M. (1985) Parsing in Functional Unification Grammar. In Natural Language Parsing. 251-278. Cambridge, England: Cambridge University Press.

Lawler, R. W. and Yazdani, M. (Eds.) (1987) Artificial Intelligence and Education: Learning Environments and Intelligent Tutoring Systems. Norwood, NJ: ABLEX Press.

Lenat, D., Prakash, M., and Shepherd, M. (1986) CYC: Using commons sense knowledge to overcome brittleness and knowledge acquisition bottlenecks. The AI Magazine, Winter, 65 - 85.

Lupker, S. J. (1988) Picture naming: an investigation of the nature of categorical priming. Journal of Experimental Psychology: Learning,Memory and Cognition, 14, 444-455.

MacWhinney, B. (1982) Basic processes in syntactic acquisition. In S. A. Kuczaj, (Ed.), Language Development. Hillsdale, New Jersey: Lawrence Erlbaum Associates, Inc.

Marchionini, G. & Shneiderman, B. (1988) Finding facts versus browsing: Knowledge in Hypertext systems. IEEE Computer, 1.

Massey, D., Kurland,L., Quayle, K. C., Roberts, B., Smith, E. E., and Tenney, Y. (In press) HAWK MACH - III Design Development Report. Technical Report, Alexandria, Virginia: Army Research Institute.

McCracken, D. and Akscyn, R. (1984) Experience with the ZOG human-computer interface system. International Journal of Man-Machine Studies, 21, 293-310.

McMath, C., Tamaru, B., and Rada, R. (In press) Graphical interface to thesaurus-based information retrieval system. International Journal of Man-Machine Studies.

Miller, George. A. and Johnson-Laird Philip N. (1976) Language and Perception. The Belknap Press of Harvard University Press, Cambridge, Massachusetts.

Miller, George. A. (1985) Dictionaries of the Mind, Proceedings, 23rd Annual Meeting of the Association for Computational Linguistics, Chicago, Illinois, 8-12 .

Miller,G.A. , Fellbaum C., Kegl J., and Miller K.(1988) WORDNET: An Electronic Lexical Reference System Based on Theories of Lexical Memory, CSL Techical Report #11, Office of Naval Research, Arlington, VA.

Minsky, Marvin (1985) A Framework for Representing Knowledge. In Readings in Knowledge Representation, Ronald J. Brachman and Hector J. Levesque (Eds), Morgan Kaufmann Publishers, Inc., Los Altos, California.

Mutter, S. A., Swartz, M.L., Psotka, J., Sneed, D. C., and Turner, J. O. (1988) Changes in knowledge representation with increasing expertise. Technical Report # 810, Alexandria, Virginia: U. S. Army Research Institute.

Nagao, M., Tsuji, J. I., and Nakamura J. I. (1986) Machine Translation from Japanese to English. Proceedings of the IEEE, vol. 74, 993-1012.

Nirenburg, S. (Ed.) (1987) Machine Translation. New York, N. Y.: Cambridge University Press.

Oxford, R. L. (1988) Semantic Mapping as a vocabulary learning toll for Second or Foreign Language. Alexandria, Va: U. S. Army Research Institute, in preparation.

Peirce, C. S. Collected Papers: Volume 2 (1965) Elements of Logic. Cambridge, MA: Harvard University Press.

Pellegrino, J. W., Rosinski, R. R., Chiesi, H. L., and Siegel, A. (1977) Picture-word differences in decision latency: An analysis of single and dual memory models. Memory and Cognition, 5, 383 - 396.

Pinker, S. (1984) Language Learnability and Language Development. Cambridge, MA: Harvard University Press.

Pliske, D. B. and Psotka, J. (1986) Exploratory programming environments for designing ICAI. Journal of Computer- Based Instruction, 13, 52 - 57.

Psotka, J., Massey, L.D., and Mutter, S. A. (Eds.) (1988) Intelligent Tutoring Systems: Lessons Learned. Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.

Rosch, E. and Mervis, C. B. (1975) Family resemblances: Studies in the internal structure of categories. In E. Rosch and B. B. Lloyd (Eds.), Cognition and Categorization. Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.

Russell, D. M., Moran, T. P., and Jordan, D. S. (1988) The Instructional Design Environment (IDE). In J. Psotka, L.D. Massey, and S. A. Mutter (Eds.) Intelligent Tutoring Systems: Lessons Learned. Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.

Schreiber S. M. (1987) An introduction to unification-based approaches to grammar. Stanford, CA: CSLI.

Simutis, Z. M., and Psotka, J. (1987) What we know that we didn't know five years ago: An overview of army research in intelligent computer-assisted instruction. Proceedings of the National Science and Communications Center, Fort Gordon, GA.

Sleeman, D., & Brown, J. S. (1982). Introduction: Intelligent tutoring systems. In D. Sleeman & J. S. Brown (Eds.), Intelligent tutoring systems. New York: Academic Press.

Smith, E. E. and Medin, D. L. (1981). Categories and Concepts. Cambridge: Harvard University Press.

Swartz, M. L. and Russell, D. M. (In press) FL-IDE: Hypertext for structuring a conceptual design for computer-assisted language learning. U.S. Army Research Institute and Xerox PARC.

Tulving, E. and Colotla, V. A. (1970) Free recall of trilingual lists. Cognitive Psychology, 1, 86-98.

VanLehn, K. (1988) Student modelling in Intelligent Tutoring Systems. In R. J. Richardson and M. Polson (Eds.) Foundations of Intelligent Tutoring Systems. Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.

Woods, W. (1970) Transition Network grammars for natural language analysis. Communications of the ACM, 13 (10), 591-606.

Yazdani, M. (1986) Intelligent tutoring systems: An overview. Expert Systems, 3, 154 - 162.