'And how many hours a day did you do lessons?'
said Alice, in a hurry to change the subject.
'Ten hours the first day,' said the Mock Turtle,
'nine the next day, and so on.'
'What a curious plan!' exclaimed Alice.
'That's the reason they're called lessons,' the Gryphon remarked:
'because they lessen from day to day.'
— (Lewis Carroll, Alice in Wonderland).

## 1. Introduction.

In any class of tutorial system, Computational Natural Language Processing (CNLP) is one of the most friendly methods available for interaction with a user. This paper describes the history of Computer Aided Instruction (CAI) before and after the eighties AICAI "coup". In particular, Natural Language Interfaces (NLIs) which were implemented, or those that are currently under development are studied. It is important to note that while NLIs and their application in Artificially Intelligent Computer Aided Instruction (AICAI) are described, they are not confined within this role. They are also used in information retrieval systems, operating systems and command execution environments.

Section two explores the advantages of CAI, and some of the problems associated with this field. Hopefully AI will be useful in solving these problems. Following this, section three involves a discussion on work in CAI before the AICAI coup. It is noted that AI did not play a significant role in CAI until right into the eighties. The fourth section discusses the arrival of AICAI and the rise of NLIs. In section five the advantage of NLIs is advocated. In particular, it is shown how natural language dialogue and beliefs may be integrated in intelligent NLIs to produce powerful interfaces for tackling Hard Core Natural Language (HCNL) problems. Finally, we close by describing the highlights of the paper.

## 2. Why do we need computer aided instruction?

There are many advantages to having computers that can teach. These advantages are listed by Loftus (1982). In short, the computer has the ability to be patient; it can help students who have fallen behind; it stimulates gifted pupils; and, it provides the opportunity to review and practice current or old material. Kearsley, et al. (1983) reviewed fifty major computer based instruction projects in terms of theoretical and practical significance. The review uncovered ample evidence that computers make instruction more efficient and effective and that computer based instruction has spurred research through the entire field of education. However, according to Kearsley et al. we have just scratched the surface of possible accomplishments with computers in education.

Others have talked about the role CAI can play in remedial education. Kidd and Holmes (1982) list advantages associated with the computer and language remediation. One advantage is that remedial students have a negative self-image thus the impersonality of the computer offers a solution to this problem. However, theirs may be a dangerous argument — the machine is impersonal, therefore it cannot be as delicate or considerate as a teacher might be. Kidd and Holmes also point out that there is not the same stigma attached to working with a computer as there is when appearing for special help classes. In fact, the association with the machine involves an element of prestige. Another advantage of CAI is that, computers are usually available twenty-four hours a day — teachers are not. As a result, computers can supply individualized learning and provide precisely selected lessons for certain individuals. It is interesting to note here that Kearsley, et al. (1983) found we know little about how to individualize instruction. Like Loftus (1982), Kidd and Holmes describe the possibility of backtracking, reviewing and verifying with computers while also agreeing about the endless patience and willingness of the machine:

> Most non-human aids to instruction reveal distinct limitations. Textbooks are confined to a pictorial/textbook approach. Tape recorders demand a purely aural presentation of material. Recorded film, movie, television or video is probably the most versatile and yet it cannot react to student input. A CRT terminal with graphic capabilities, or microprocessor, can command all these features. The element to be learned can be presented pictorially, aurally, textually, with the added

features of movement and color, and it reacts directly to student input. (p. 235)

The important point brought up in this extract is that a machine, "...reacts directly to student input." It is this interaction or dialogue that makes CAI significant and furthermore, which separates it from other technology involved in instruction.

NLIs also have applications in helping the 'at-risk' learner. Goldenberg, et al. (1984) describe a pupil called Michael who cannot spell, and finds reading difficult. Michael is also interested in electronics. They believe that a program called SOPHIE built to instruct students in electronics, will be useful to him.

Although the advantages of CAI are numerous, there are also problems associated with building machines which can teach. Loftus (1982) lists the following human problems of CAI: teachers must tailor their classes to the requirements of particular programs; students have a fear of machines, and machines often do not tolerate mistakes. He states that, "... elimination of their *[students]* perceived right to be sloppy causes many of them to experience high levels of frustration as the machine repeatedly rejects answers, which are almost, but not actually, correct" (p. 119). Other problems are that students complain they do not want to take typing courses, and teachers fear that they may have to program machines.

There are technical problems in CAI augmenting the human problems: a given keyboard may not be appropriate for a student of a particular nationality; students experience numerous situations where there is a lack of terminals, and even if they get a terminal the load may be so high it is impractical to use the machine anyway; with regard to security there is the possibility that students can erase records; furthermore, Kearsley, et al. (1983) believe we do not have a good understanding of the effects of what they call "instructional variables". These are variables such as graphics, speech, motion or humor.

Molnar (1982) believes that the information explosion **requires** that CAI be developed. In his article on the search for new intellectual technologies he states that the proportion of knowable knowledge is growing smaller and we are experiencing an "ignorance explosion". Molnar talks of the knowledge worker who uses and applies knowledge at his job whether he is an engineering technician or a bank clerk. On asking what new demands will be placed on the educational process, Molnar's solution is the development of new intellectual technologies with the substitution of algorithms for intuitive judgements in automatic machines. For example, Control Data corporation has developed the program PLATO for learning environments. IBM expects its "learning centers" to cut the need for technical instructors from the field by greater than thirty percent. Although Molnar's fears of an ignorance explosion may be premature it is true that the quantity of information available to people is increasing rapidly and that the proportion of it we know is getting smaller. However, the efficiency of instruction and the number of instructors may not be the only limiting factors on the ignorance explosion. After all, we can only assimilate a limited amount of knowledge.

## 3. The situation before the AICAI coup.

Before the storm, interfaces in CAI were quite limited. Most interfaces used menu-driven front-ends or keyword systems containing trivial subsets of natural language. Thus few programs accepted natural language sentences or phrases. Wyer (1984) describes the picture for us in her paper on AI and CAI where she explains the limitations of "controlled learning" and "Ad Hoc Frame-Oriented CAI (AHFO CAI)". She states that students had little or no initiative in their own learning; that they couldn't use natural language with the system and rigid computer programs led to frustration.

Lewis (1983) believes that early attempts to provide flexibility in these programs suffered from two defects: users were given little guidance and often got lost in a forest of technical commands and parameters; and, menu-driven programs which provided varying degrees of both flexibility and structure displayed distracting options which could affect students' line of thought:

> The design of pupil material for use in a particular CAL [*Computer Aided Learning*] package is clearly going to be dependent on the design of the program. It is here that most inflexibility may occur as it cannot be assumed that teachers are able to modify software. (p. 82)

He also quotes the following introduction to keyword-controlled software taken from course notes prepared at St. Martin's College, Lancaster:

> An important criterion for a program to be used by non-specialist users of computers is to ensure that the user is in control, knowing clearly how to make the program perform in the desired way. The phrase 'user friendly' has been coined to describe this criterion. Few programs meet this requirement, yet it is crucial that they should do so if they are intended to be used by novice or timid users. (p. 82)

Lewis believes that CAI programs meet the above criterion if the first-time user is guided by, and prompted by, a dialogue in natural language. He also states that the program must allow flexibility to cater for the novice, yet provide the advanced learner with the freedom to explore all features of the program at will.

Researchers in CAI kept away from AI and continued to develop their own solutions for building better interfaces. Two attempts to keep AI at bay were generative approaches and authoring languages. However, it was soon realized that these were not adequate substitutes for NLIs. In Sleeman and Brown's (1982) book on intelligent tutoring systems generative CAI is described as drill and practice techniques. The idea of generative CAI is to select problems at a level of difficulty appropriate to the students overall performance. The obvious limitation of these systems is noted by Sleeman and Brown and we will see later on how belief models may be used to tackle such limitations.

Kearsley, et al. (1983) observe that significant progress or development of authoring tools or techniques have occurred to augment computer based instruction as they describe authoring languages and systems:

> One major area of research unique to CBI *[Computer Based Instruction]* has been the development of languages and systems for creating interactive programs. The purpose of both CBI author languages and authoring systems is to make the development of instructional lessons easier. Authoring languages provide a set of features especially suited to the kind of programming required by CBI (i.e. screen formatting, answer processing, data collection and analysis). (p. 90)

Systems such as PLATO and TACCIT distinguish authoring languages and authoring systems. For PLATO it is necessary to learn to program in Tutor whereas with TACCIT one can specify content and instructional strategies. Today, PLATO is menu-driven and TACCIT has an authoring language called TAL. Alas, Kearsley et al. found that "One drawback of authoring languages or authoring systems is that they have sometimes constrained the author to relatively narrow models of instruction" (p. 91).

With further development there was a move towards AI and natural language interfaces. A startling fact, however, is that in 1982 people were building systems with little or no AI in them. It may be the case that there were hesitations to apply AI techniques in education or that the AI sector didn't advertise enough. It seems that even in 1982, AI and education were only united in a minimum number of programs. Kidd & Holmes (1982) describe six systems under development at various universities where it appears as if AI wasn't even considered. These programs included direct word matching strategies and error detection for incorrect characters or words in the input — wholly syntactic, yes, but no AI.

For example at Dartmouth and NYSU, programs stored grammatical components, stems and endings of irregular verbs, and rules. These programs indicated where students went wrong. The UWO (University of West Ontario) project is outlined:

> ...when the error does not appear to be due to a lack of linguistic competence but is rather a minor slip in performance — omitted or inverted letters, spelling errors, and the like — mere detection is usually sufficient to allow the student to correct his mistakes. The location of errors is indicated by underlining the incorrect letters or inserting an underlined space where a letter is missing. (p. 238)

In 1982, simple syntax was being used for CAI while researchers in computational natural language processing such as Wilks (1975, 1978) were showing the advantage of semantics for processing language years before. Kidd & Holmes (1982) associate these syntax checking programs with a description on the advantages of feedback, which is surely an AI problem:

> Feedback that is both pertinent and helpful is needed in response to all student answers, but it is obviously more vital when dealing with incorrect attempts. Any drills or exercises done by the remedial student should not function as a means of testing but as a learning process. Information given to him should help him understand where he went wrong and why, if he is ultimately going

to produce the correct answer and comprehend the reasons for its correctness. This type of feedback demands a medium that can provide very sensitive response evaluation. When intelligently programmed the computer can fulfill this function admirably. (p. 236)

The above extract describes AI techniques exactly. However, although Kidd and Holmes define their associated examples as being "intelligently programmed" these examples are certainly not intelligence programming. Even more serious is the following quote from Higgins and Johns (1984): "Similar work [on spelling correction and input errors] has been done by Hanno Martin for German, although there the focus has been on syntax, using automatic parsing techniques. This is one of the many areas in which computer-assisted learning may expect to borrow in future from theoretical work in artificial intelligence" (p. 59). Do they not realize that CAI can already borrow these techniques and does not have to wait for the future? I believe one of the reasons that AI didn't catch on in the education sector was because of these misconceptions. Only a few people in the education sector seemed to know about developments in the AI field. Schank (1982) sums up the situation for us in his book *Reading and Understanding* when he states, "It is not surprising that our research has had no effect, since people who work in Artificial Intelligence are in the field of Computer Science, and usually have little contact with educators or people whose research is on how to teach reading to children" (preface).

## 4. The coming of AICAI and natural language interfaces.

It was soon realized that AI had a role to play in education. Wyer (1984) tells us that a true departure was Carbonell's SCHOLAR program. SCHOLAR departed from the Ad-Hoc Frame Oriented approach (mentioned earlier) as its structure was information oriented. It is important to note here that the meaning of "frame" within the CAI field has little relation to the meaning of "frame" in the AI sense. SCHOLAR taught the geography of South America through a 'mixed initiative' system rather than through frame oriented 'page turning'. Thus the student could query the system using natural language and vice versa. SCHOLAR's generation and comprehension of natural language is based on a semantic network — a theory developed within the AI community. The teacher must build the network and the significance of SCHOLAR was the in the fact that it made **some** attempt to utilize restricted natural language for input and output.

Wyer (1984) attributes several advantages to intelligent CAI. One advantage is that problem-solving techniques are instilled and attempts to model the student are introduced. In particular, tutoring strategies are developed. The idea of intelligent CAI is to separate teaching strategies from subject matter.

Higgins and Johns (1984) also noted the prospects associated with AICAI. They distinguish unintelligent and intelligent analysis. Unintelligent analysis is defined as that analysis where canned language is stored in memory. The machine operates in an automatic or random way on the forms of language and has no sensitivity to meaning. On the other hand, intelligent analysis involves the synthesis of language using a stored vocabulary and a set of grammar rules. The machine then relates the language to some form of knowledge structure such as the semantic networks mentioned earlier. Higgins and Johns go on to explore the limitations of intelligent synthesis. They talk of programs which *understand* the languages they process. We are told that Artificial Intelligence research since the sixties has shown that the computer can be programmed to generate and respond to language appropriately, provided that the language relates to knowledge of a restricted world. The classic program was Winograd's SHRDLU which 'knew' about a table-top world of three-dimensional objects, blocks, cones and pyramids. SHRDLU analyzed natural language and could manipulate objects in response to instructions in English. In a chapter on the computer outside the classroom two points about AI and language learning were noted. First, there are a large number of practical spinoffs to be gained; examples are various NLIs and expert systems for different applications. We have seen systems which offer to check and correct grammar of text such as IBM's EPISTLE program. Second, the theoretical work in AI should throw light on questions of general interest in language teaching.

Novak (1976) believes the English language is a remarkably efficient and effective means of communication. However, he notes that the English sentence is only a set of clues to the message it wishes to convey. He makes the point that the clues or code can be interpreted by comparing these with what one already knows. In fact, this is the heuristic many NLIs use to process natural language in limited domains.

Kearsley, et al. (1983) define intelligent CAI exactly as they state, “Most importantly, intelligent CAI programs have sophisticated student models which allow them to understand what the student is/is not learning. It is this capability to understand what is being taught and why a student makes a mistake that makes these programs ‘intelligent’” (p. 91). They list a number of intelligent CAI programs including SCHOLAR (Carbonell & Collins), SOPHIE a tutor for electronics troubleshooting (Brown, Burton & deKleer), WHY for tutoring in meteorology (Stevens & Collins), WEST, a coach for game playing (Brown & Burton) and BUGGY, a diagnostic program for arithmetic (Burton & Brown). Alas, there is one reservation expressed by Kearsley et al. — despite the promise of intelligent CAI, none of these programs have had any real impact on educational practice to date. It is important to note that except for SCHOLAR which originally was not intelligent CAI anyway, all of above programs were developed in the late seventies and eighties.

The WEST program was theoretically useful as it demonstrated that intelligent CAI could not be done with traditional computer based instruction authoring languages such as Tutor. Their basic structure is not suitable. WEST is a popular game which was originally developed on the PLATO system. Kearsley et al. (1983) also argue that one of the outcomes from their research was that we do not know a lot about interaction. They believe that the more interaction and student involvement with the program the better the computer based instruction. Again, I would argue that NLIs offer the way forward in this domain.

A very detailed combination of papers on AICAI is compiled in Sleeman & Brown’s (1982) book, *Intelligent Tutoring Systems*. Many of the AICAI programs in these papers include intelligent natural language interfaces. In the introduction they single out three issues that may be of special interest to AI researchers. These issues are: the implementation of friendly interfaces and conversational systems; the contribution of the student modelling work to techniques for instruction; and, special purpose inference and deduction techniques. Sleeman and Brown discuss what they call “habitable(friendly) natural language systems” (p. 5). They propose that, “...limited domains of inquiry can be handled robustly and efficiently by formalizing the semantic structure of the domain directly as a grammar in its own right” (p. 5). Burton explored this technique in SOPHIE as do many other NLIs in areas of application other than CAI. Since we have now moved into the higher reaches of AICAI so suddenly it’s time to consider the possible integration of natural language dialogue and beliefs within NLIs.

**5. Hard Core Natural Language, dialogue and applied beliefs.**

In Sleeman and Brown (1982), the problems of, modelling discourse between the machine and the user, handling deletions, ellipsis, and pronominal reference are discussed. We will term the latter topics Hard Core Natural Language (HCNL). The handling of context on preceding dialogue and focus of conversation are also HCNL. Realistic dialogues with the user must include some conceptualization of the domain and the current focus of conversation. Much work has been done already in the domain of goals and plans in order to solve HCNL. However, another dimension is becoming more important. Belief models were discussed back in the seventies by people such as Bruce and Schmidt (1974) and Brown (1974). Recently, studies on beliefs have returned to the limelight, especially with research in Natural Language Processing. Wilks (1985) has advocated work which can represent the notion of an individual’s beliefs and has a way of distinguishing between what one participant in a dialogue believes and what another does, where those two may be different, even though the two people communicate very well. He describes a dialogue which takes place, “between a (human) USER and a SYSTEM, that may or may not be human...” (p. 10). Later on, discussing the limitations on default belief, Wilks proposes an environment for expert and self knowledge.

Wilks & Bien (1983) discuss beliefs, points of view and multiple environments. This paper considers the phenomena of *assumed beliefs*. Assumed beliefs are those beliefs which we do not really hold, but which we attribute to our hearers. What is the significance of all this information on beliefs of various people? Even if we do model the beliefs in conversations why are these beliefs useful?

Patrick (1982) discusses problem diagnosis performance in relation to SOPHIE, the electronics troubleshooter:

> In some studies, which have concentrated on the representations of knowledge of a trainee and an expert troubleshooter, the difference between the student and expert model can be used to remove any misconceptions which the trainee might have about the system and its states.” (p. 319)

We can see that users have assumed beliefs pertaining to various AICAI programs. Although Patrick describes the Adaptive Computerized Training System (ACTS) under development at the Army Research Institute since 1974 as providing, “an adaptive computer program which can develop and compare the decision structure of a trainee with that of an expert during the diagnostic process” (p. 320), I do not think they are applying any such strategy as beliefs or points of view.

Wyer (1984) tells us that SCHOLAR makes no attempt to evaluate the student’s incorrect responses or to use a diagnosis of what may be wrong to help the student overcome misunderstandings or misconceptions. Belief models concerned with false beliefs could be used to recognize misconceptions. Higgins & Johns (1984) provide an analysis of discourse and describe the false impression that programs such as ELIZA and DOCTOR produced to make it seem as if they were intelligent. Today we are trying to do the opposite. Sleeman & Brown (1982) tell us that pre-AICAI research centered on models of the student which were based more on parametric summaries of his behavior than explicit representations of his knowledge. Surely, we can represent this knowledge using belief models.

Sleeman and Brown describe three major acknowledged shortcomings in AICAI: instructional material produced in response to students queries or mistakes is often at the wrong level of detail — this might not happen if the system had a set of beliefs about what the user believes; the system assumes particular conceptualization of the domain pushing the students performance into one conceptual framework — this might not happen if the system knew what the user believed about it, also they say tutoring and critiquing strategies used by these systems are ad hoc and unprincipled. Moreover, Sleeman and Brown refer directly to beliefs, “Discovering consistent principles would be facilitated by constructing better theories of learning and mislearning — a task requiring detailed psychological theories of knowledge representation and belief revision” (p. 3). Furthermore they talk of the *assignment-of-credit* problem, i.e., the problem of how to rationally allocate blame (credit for having caused failure) when more than one elementary step is necessary for success. If the failure of a task occurs, which requires multiple skills, then which skill is it that the student does not have? We are more likely to find out if we have derived a model of the beliefs of the student. The following are examples of where points of view can be applied:

> The problems of failing to discuss important aspects of physical processes and failing to use important ways of describing physical processes arise because the script-subscript structure is limited in the types of knowledge it can represent. We believe that representing knowledge about physical processes requires multiple “representational viewpoints”. (p. 15)

> The availability of multiple viewpoints raises the possibility that some errors may turn out to reflect the fact that some perspective is missing in the students knowledge. Alternatively, some bugs might be described as a failure to integrate several different viewpoints, a failure to recognize contradictions between two different perspectives or to update one viewpoint when relevant changes are made in another. (p. 24)

Wilks (1986) describes a multi-actor system for cooperation and planning which would implement the “representational viewpoints” described above and how belief models could be applied to a problem arising in an NLI called GIFT. Similar ideas are discussed in Wilks & Bien (1979).

In Sleeman & Brown, Goldstein describes how *belief measures* can be defined on genetic graphs:

> This corresponds perhaps, to the psychological observation that a student does not always employ a skill which has just been explained. While the student may be able to repeat the explanation, and even describe implications of the new knowledge, he may not actually use the skill when solving problems... (p. 73)

> A formal representation for this learning conservatism can be added to our learning model by introducing a belief measure. (p. 73)

However, Goldstein uses a numerical representation for beliefs. Wilks & Bien’s scheme or model is a better representation than Goldstein’s model which can be criticized on the basis that emotions or feelings cannot be realistically represented using numerical quantities. Beliefs are also described in Clancey’s GUIDON program, which uses certainty factors for beliefs. Overlay student models are utilized to determine whether the tutor believes that the student knows how to achieve subgoals. Clancey decides that change in belief is interesting. There are numerous examples where belief models can be applied to solve HCNL problems. Furthermore, the integration of belief models within natural language interfaces is a powerful mechanism which may subsume theories on plans and goals.

## 6. Conclusion.

This paper has been a study on the history of AI in Computer Aided Instruction with particular emphasis on Natural Language Interfaces. Before the eighties the existence of AI in CAI was quite limited. However, this was soon to change, and after the AICAI coup there was an explosion of AICAI programs. Due to this explosion there was a greater insight into the problems of AI. Fortunately, the investigation of AICAI led to a better understanding of instruction itself.

Most of the AICAI programs developed after the coup included intelligent NLIs which utilized new theories in Computational Natural Language Processing (CNLP). However, the application of intelligent NLIs led to higher level problems in Natural Language Processing, i.e. Hard Core Natural Language (HCNL). HCNL includes problems in modelling both novice and expert users, and explicit language processing difficulties in interpreting dialogue. Researchers in natural language discovered the stubbornness of user input involving contextual reference, ellipsis, and deletions. Wilks and Bien's theory of belief models may be used to tackle numerous examples of HCNL.

We can only guess at the reasons why AI had to wait for such a long time before it was applied to education. It could be that AI people did not advertise their theories enough outside the AI field and education people did not look to AI for solutions. Even in 1982 people in CAI were either unaware, or sceptical of progress in AI, but we can be satisfied that AI and education are finally beginning to join forces and we should try to unite these two areas more in the future. Intelligent Natural Language Interfaces will certainly enhance the marriage of both Artificial Intelligence and education.

## REFERENCES


Brown, Geoffrey (1974) *The believer system*. Department of Computer Science, Rutgers University, New Brunswick, NJ 08903.

Bruce, Bertram & Charles F. Schmidt (1974) *Episode understanding and belief guided parsing*. Computer Science Dept. (Bruce) and Psychology Dept. (Schmidt), Rutgers University, New Brunswick, NJ 08903.

Goldenberg, E. Paul, Susan Jo Russell, Cynthia J. Carter, Shari Stokes, Mary Jane Sylvester & Peter Kelman (1984) *Chapter on Communication and language arts. In Computers, education and special needs*. Addison-Wesley: London.

Higgins, John & Tim Johns (1984) *Computers in language learning*. Addison-Wesley and Collins ELT: London.

Kearsley, G.; B. Hunter & R.J. Seidel (1983) *Two decades of computer based instruction projects: what have we learned?*. Technological Horizons in Education (T.H.E.) Journal, Vol. 10, No. 3, January, 90-94 & T.H.E. Journal, Vol. 10, No. 4, February, 88-96.

Kidd, Marilyn E. & Glyn Holmes (1982) *The computer and language remediation*. Programmed Learning and Educational Technology (PLET), Journal of AETT, Vol. 19, No. 3, August, 234-239.

Lewis, R. (1983) *Teachers, pupils and microcomputers*. Technological Horizons in Education (T.H.E.) Journal, Vol. 10, No. 4, February, 81-87.

Loftus, William J. (1982) *A user's balance sheet on computer assisted learning*. Technological Horizons in Education (T.H.E.) Journal, Vol. 10, No. 1, September, 118-123.

Molnar, Andrew R. (1982) *The search for new intellectual technologies*. Technological Horizons in Education (T.H.E.) Journal, Vol. 10, No. 1, September, 104-112.

Novak, Gordon S. (1976) *Computer understanding of physical problems stated in natural language*. Department of Computer Science, University of Texas at Austin, March.

Patrick, J. (1982) *Some issues and developments in computer-based training (CBT). Programmed Learning & Educational Technology (PLET), Journal of AETT, Vol. 19, No. 4, November, 317-321.*

Schank, Roger (1982) *Reading and Understanding: Teaching from the perspective of Artificial Intelligence*. Lawrence Erlbaum associates, publishers: Hillsdale, New Jersey.

Sleeman D. and J.S. Brown (Eds.) (1982) *Intelligent tutoring systems*. Computers and people series edited by B. R. Gaines, Academic Press: London.

Wilks, Y. (1975) *A preferential pattern-seeking semantics for natural language inference*. Artificial Intelligence, Vol. 6, 88-111.

Wilks, Y. (1978) *Making preferences more active*. Artificial Intelligence, Vol. 8, 75-97.

Wilks, Y. & J. Bien (1979) *Speech acts and multiple environments*. Proc. IJCAI-6, Tokyo, Japan, 451-455.

Wilks, Y. & J. Bien (1983) *Beliefs, environments and points of view*. Cognitive Science, Vol. 8, 120-146.

Wilks, Y. (1985) *Relevance, Points of View and Speech Acts: An Artificial Intelligence View*. Memorandum MCCS-85-25, Rio Grande Research Corridor, Computing Research Laboratory, New Mexico State University, NM 88003.

Wilks, Y. (1986) *Projects at CRL in Natural Language Processing*. Memorandum MCCS-86-58, Rio Grande Research Corridor, Computing Research laboratory, New Mexico State University, NM 88003.

Wyer, Jo-Anne (1984) *New bird on the branch: Artificial Intelligence and computer-assisted instruction*. Programmed Learning and Educational Technology (PLET), Journal of the AETT, Vol. 21, No. 3, August, 185-191.