A Model for Intelligent Computer Assisted Language Instruction (MICALI)

Ali Farghaly

Kuwait University, English Department, P.O. Box 23558, Safat, Code 13096, Kuwait

Abstract: This paper presents the view that Computer Assisted Language Instruction (CALI) software should be developed as a natural language processing system that offers an interactive environment for language learners. A description of Artificial Intelligence tools and techniques, such as parsing, knowledge representation and expert systems is presented. Their capabilities and limitations are discussed and a model for intelligent CALI software (MICALI) is proposed. MICALI is highly interactive and communicative and can initiate conversation with a student or respond to questions on a previously defined domain of knowledge. In the present state of the art, MICALI can only operate in limited parsing and domain-specific knowledge representation.

Key Words: foreign language teaching — educational technology — intelligent software — parsing — syntax — natural language understanding — grammar formalisms — knowledge representation — man-machine interaction — tutoring systems.

Introduction

The advent of the personal computer in the late seventies and its availability in schools and colleges led educators to think of ways to make use of this new technology for instructional purposes. Since educators knew little about this new technology they had to rely on programmers who knew how to manipulate the machine.

The first generation of Computer Assisted Instruction (CAI) was characterized by two important properties. First it came into being as a result of the emerging new technology. The question educators had was: "We have a new technology, what can we do with it?" rather than: "We have a problem, how can technology help us with it?" Second, almost all CAI software was developed by programmers who knew very little about the principles and techniques of teaching methodology.

Not surprisingly, many researchers and ESL educators expressed disappointment over the quality of most ESL Computer Assisted Language Instruction (CALI) software. The (1982) believes that "the most fundamental problem is that most educational software is written by programmers who know nothing about pedagogy." Stevens (1983) suggests that "professional educators must be integrally involved in its production." For this purpose, books have already appeared (Ahmed, Corbett, Rogers, and Sussex, 1985; Higgins and Johns, 1984; Hope, Taylor and Pusack, 1984; Underwood, 1984; Jones and Fortescue, 1987) to introduce ESL teachers to educational software design and evaluation.

A second generation of CALI software came as a result of serious involvement of language teachers and educators in the production of CALI. Many teachers were involved in research over CALI software design and evaluation (Phinney, 1987; Both, 1987; Hill, 1987; Stevens and Thrush, 1987). This second generation was pedagogically sound, and based on more com-
municative approaches to language learning. It also benefited from close observation and evaluation of student response to different types of CALI software.

Most current CALI programs still have built-in solutions to the problems presented to the learners, but very limited reasoning power. Such programs cannot solve the problems themselves. In many cases the computer rejects students' answers simply because they were not listed in the program among the acceptable ones. In a review of CALI software Johnson (1987) reports on his experience with a good software package for teaching English idioms: “The only problem I had with Essential Idioms is one that is common to much of the language arts software, and that is, the program’s inflexibility in accepting meaningful responses.” At advanced level, it is almost impossible to list all acceptable responses expected from students. Program developers had to restrict their questions to the type that have a limited number of possible answers. This imposes severe restrictions on the types of questions that can be included in CALI programs. Most questions in these programs are objective questions which are more appropriate to testing than to teaching. Open-ended questions and communicative interaction are missing from most CALI software.

Third generation CALI software should be organized around expert systems (Bonnet, 1984), and should be able to solve the problems they give to students, as well as evaluate and criticize student performance. In language learning this would mean that principles and strategies for natural language parsing must be an integral part of CALI programs. The design of efficient parsers has been one of the central goals in Computational Linguistics. The parsing operation (Karttunen and Zwicky, 1985) can be viewed as the application of language-specific rules of a sentence (like phrase structure rules $S \rightarrow NP VP$) in order to obtain all grammatical descriptions of that sentence. Parsing a sentence with a grammar (Grishman, 1986) means finding a derivation, usually represented as a parse tree of that sentence. If the sentence is ambiguous, there will be several derivations corresponding to the different interpretations of that sentence, unless the ambiguity of the sentence can be resolved by the context in which it occurs. The derivation of a sentence is obtained through the application of the rules in a given order. We may start building up the tree, performing lower applications before those higher up (Bottom-Up parsing) until we reach the “$S$” node. Alternatively, we may begin with the start symbol, working down from the top of the tree (Top-Down parsing). The specific rules are neutral between the TD and BU approaches, as well as between analysis and generation. They can be viewed either as instructions for the generation of the sentence or as instructions for assigning a structural description to a given sentence.

A parser is a tool used by computational linguists and Artificial Intelligence scientists for a variety of purposes such as:

- (1) testing the adequacy of grammars;
- (2) parsing the source text in a machine translation system;
- (3) analyzing the input string in a man/machine interface or an advanced information retrieval system.

I will argue that AI tools and techniques such as parsing, knowledge representation and expert systems can improve the quality of existing Computer Assisted Language Learning software. However, to achieve this goal we need to recognize both the limitations and capabilities of these tools, and to define their function in CALI programs as opposed to natural language processing systems. We will also need to develop the system that can take only what it needs from these tools and at the same time supplement them with modules that provide the learner with what he needs.

AI tools may help us overcome some of the major drawbacks in CALI software. Programs could be more interactive, departing form the rigid objective questions, and allowing for freer communication between students and the computer. It will be easier to introduce variety to the programs, thus they will attract students' interest and minimize boredom. Teaching materials for CALI will be easier to write and material writers will be able to use their creativity to the full since they no longer have to write their materials in the strict format required for present CALI programs.

**Computational Linguistics**

Language may be viewed as a knowledge-based