Artificial Intelligence: The Human-Computer Interface?

James D. Basile
Department of Computer Science
Long Island University, C. W. Post Campus
Greenvale, NY 11548

ABSTRACT

This paper explores some applications of artificial intellegence as a front end interface for computer systems. It is intended to acquaint the reader with possible areas where such an interface has been used and to provide a summary of some relevant sources for further reading.

Over the past decade tremendous advances have been made in the area of hardware development. At the same time, new developments in software have been comparitively scarce. We are now at a point at which desktop computers are common in the workplace. Yet, because of the inability of the average user to take access the computer's resources, much of this computing power is underutilized.

Many people are now familiar with the idea of artificial intelligence. Nonetheless, it is often regarded as an esoteric, theoretical tool of the mad scientist seeking to build a machine that "thinks". In spite of this misconception, artificial intelligence techniques have begun to creep into the marketplace aiding the ability of humans to use the computer more effectively. By allowing humans to communicate more easily with the computer, these techniques remove roadblocks to using computer-based tools. In addition, such techniques often eliminate the need for the user to specify paths for the computer to follow in routine problem solving. Thus, through use of artificial intelligence interfaces we can successfully improve the user's abilty to use software and perhaps more fully realize the potential of the computer.

There are currently three main areas in which artificial intelligence techniques can be brought into play in every day situations. Perhaps the most common is the idea of the "expert system", more properly called a knowledge-based system. Another area in which such techniques are useful is the realm of natural languages and interpretation of meaning. Finally, the computer can be instructed to "think" its way through routine tasks that ordinarily occupy much human time.

In many fields, the number of individuals qualified to make reasonable decisions has been far outstripped be the need for such individuals. Engineers, physicians, finiancial analysts, and even programmers fit this mold. The knowledge-based system is intended to fill this gap by providing computer resources with the ability to generate decision scenarios.

The knowledge based system includes a data base of facts (factual knowledge) used in the decison process as well as rules (heuristic knowledge) associated with it. For example, we can think of an expert system to assist in patient diagnosis as consisting of facts about the patients, symptoms of various disorders, and rules that associate symptoms with disorders. By using the facts provided and "following the rules" the computer can provide some possible diagnoses for the disorder. Moreover, the system can provide the rationale for the diagnosis. For examples of such systems see [KUL84] and [TRE86].

It is this dual function that is so important. In whatever application , the computer can be given a decision to analyze. Then possible paths along with the method used to arrive at them can be documented. This enables the user to utilize the computer to search through a jungle of facts and rules and yet leave the final interpretation to the human. This can maximize the human resources available to solve a problem while maintaining an overview of the decision process.

Another example of such knowledge based applications is in the area of VSLI design. Because of the complexity of the design process, utilization of human resources can be optimized by having a computer based "design assistant" that can assist with routine tasks such as routing and placement of design elements as well as test out the efficacy of a particular design. This leaves the human engineer more time to actual conceptual design which the computer cannot yet accomplish effectively (see [NES85]). Other examples of such expert systems can be found in [BON83], [KUN84], or [MCD80].

Another major application of artificial intelligence is in system interfaces for interpreting natural languages [HAR84]. While such situations occur in virtually every area, they are most heavily concentrated in the area of database query facilities. With more and more corporations turning to computerized databases for information regarding everything from personnel to design specifications, the need is more widespread to develop non-technical interfaces for these databases.

The object is to create a user interface that is capable of "conversing" with the user in a non-technical, "human" way. Instead of formulating a query in some programming language (which is what most query languages are), or even instead of working through a series of pictures (icons) of the desired action, it would be beneficial for the non-technical database user to submit a request like "Give me the names of the employees in department 2045." Even more important is the ability of the system to recognize and process requests such as "And give me their addresses" where "their" is context dependent (here it means the employees').

What is involved in building such an interface is two-fold: first, the ability to decipher the syntax of the request and, second, the ability to determine what each of the syntactic elements means in the context of the query. Thus, the system interface must first be designed to "know" the elements of the user's language. Even once this is accomplished, however, the system must still be made capable of interpreting each element, recognizing ambiguous ones, and prompting the user to clarify the ambiguity. Database systems such as Savvy and Knowledge-Man are attempting to utilize some of these techniques.

The natural language interface can be extended to other applications as well such as process control, instrumentation interfaces, and even expert systems. Perhaps the culmination of all of this is a voice recognition system that allows the user to "converse" with the computer. While this is not yet a reality for everyday application, these tools lead us in that direction.

The third major area of usage for artificial intelligence in practical applications is in providing the computer with sufficient "intelligence" to allow it to perform routine tasks. In many practical situations, it is unwieldy for a human or humans to monitor the large number of variables that might exist in a manufacturing process or complex control system. To do so would to be a waste of resources and actually inefficient, even leading to inaccuracies. If the number of variables is large, then it is easy to see how a human monitor might miss a significant change, whereas the computer would not. In these situations, then, it is practical to not only let computers monitor the process, but also give the computer the ability to make decisions regarding the control of the outcome.

This kind of system is a natural choice for say an on-board navigation system, a circuit-board production, or even a nuclear power plant. When given the ability to use internal factual and heuristic knowledge of the situation to control it, the computer can monitor and control a complicated system more effectively than a human. Of course, the capacity to question the computer about its rationale as well as override it must be present since the computer's knowledge is limited.

The ultimate extension of this knowledge-based process control is of course robotics. Industrial robots that perform routine tasks have become a given in the manufacturing process. Nonetheless, these robots are but a beginning. As we give robots more ability to perform according to heuristic knowledge, we will open new horizons for utilizing them in practical situations as well as technical ones. Thus, we can envision robots that not only perform routine tasks but also "perceive" their environemnt and "learn" from it. An important element of this kind of system will no doubt be visual perception and audio recognition. Already such inroads are being explored.

What is the role of Forth in all this? Forth has long been utilized for process control applications. Forth has been more recently investigated for its possibilities in artificial intelligence. It would seem perfectly suited - compact code, fast execution, easily reconfigurable, extensible data structures - but has not yet attained wide acceptance. The use of Expert-2 [PAR83] as a Forth-based artificial intelligence environment demonstrates that it can be so utilized. Even more is its use to implement OPS5 [DRE85] or LISP [CAR85], popular artificial intelligence environments.

Some of the examples in this paper have been developed in Forth-based systems. If its potential in this market is to be realized, we must move forward on developing and providing universal Forth tools for artificial intelligence.

References:

Bonissone, Piero and Johnson, Harold, "Expert System for Diesel Electric Motor Repair", Journal of Forth Application and Research, Vol. 1, No. 4, Sep. 1983.

Carr, Harold and Kessler, Robert, "A Forth-Based Object File Format and Relocating Loader Used to Bootstrap Portable Standard LISP", Journal of Forth Application and Research, Vol. 3, No. 2, Jul. 1985.

Dress, William, "A Forth Implementation of the Heap Data Structure", Journal of Forth Application and Research, Vol. 3, No. 2, Jul. 1985.

Harris, Larry, "Natural Language Simplifies Computer Access", Systems & Software, Jan. 1984.

Kowalski, Thaddeus, and Thomas, Donald, "The VLSI Design Automation Assistant", IEEE Design and Test, Vol. 1, No. 1, Feb. 1984.

Kulikowski, Caismir, "Expert Medical Consultation Systems", Journal of Medical Systems, Vol. 7, No. 3, 1983.

Nester, Eric, "Symbolic Processor Aids Design of Complex Chips", Computer Design, Jan. 1985.

Trealase, Robert, "Experimental Medical Diagnosis System using Expert-2", Journal of Forth Application and Research, to be published.