Computer advises without being asked

By Henry W. Pierce
Post-Gazette Staff Writer

Most people don't like back seat drivers. But what if the back seat driver were a computer?

An expert computer system that can offer advice without being asked has been developed by researchers in Pittsburgh and Philadelphia.

In its present form, the system suggests more efficient ways to program computers.

However, one of its developers says, there's no reason why it can't be modified to offer unsolicited advice on other matters.

Using the same principle, systems could be set up that would advise people on tax preparation, writing style and similar technical topics.

The system was developed by Jeff Shrager of Carnegie-Mellon University and Tim Finin of the University of Pennsylvania.

As Shrager explained it, the adviser "follows the user's interactions with the system and volunteers its help when it believes that the user would benefit from advice."

In effect, Shrager said, the system watches whoever is using the computer, and if it sees that the person isn't following a very efficient procedure, it steps in with a word of wisdom.

"The user need not ask for help," he emphasized.

In an interview, Shrager said there is no reason why a similar system couldn't be developed that would monitor a person's tax preparations and offer advice on possible deductions.

Or, he went on, a system could be developed that would watch journalists while they're writing stories and suggest sharper lead sentences and improvements in writing style.

"It might take several years to develop such a program," he conceded.

A report on the system is among 96 papers to be presented this week at the second National Conference on Artificial Intelligence, hosted by Carnegie-Mellon University and the University of Pittsburgh.

The conference, centered at Forbes Quadrangle on the Pitt campus, is slated to draw 1,200 scientists, engineers and business representatives between today and Friday.

Among the scheduled reports are papers on teaching computers to understand natural language; developing expert systems for diagnosing and treating nuclear reactor accidents; a model for a computerized nuclear power plant consultant; computer text recognition; job-shop scheduling; and "the role of experience in the development of expertise."
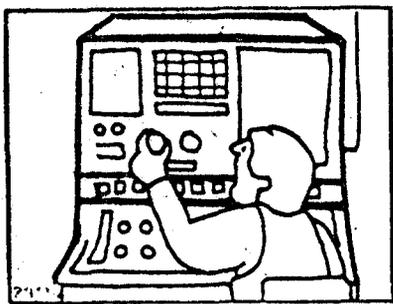
Tom Hritz

Busybody computer

When I read earlier this week about the National Conference on Artificial Intelligence which is meeting here, and about the computer out at Carnegie-Mellon University that gives advice without being asked for it, I found it hard to believe. I always thought that you couldn't get anything out of a computer unless you put that thing into it first. Input, as it's called.

So it was with a considerable amount of skepticism that I went out there uninvited the other night to meet this computer. I logged on.

"Hi," I said to the computer.

"Hello," it answered on its video screen. "Who are you? You don't look like anybody from the National Conference on Artificial Intelligence."

"Oh, I'm not," I admitted. "I'm just an average person and I don't know anything about computers. But when I read about you in the newspaper — about how you can give advice without being asked for it — I just decided I had to meet you. Frankly, I'm not sure if I believe it. But I'm sorry. I'll leave if you want."

"No, stick around," said the computer. "As long as you're not expecting advice on your income tax, it's OK. I'm getting a little bored with income taxes and job-shop scheduling. What kind of unsolicited advice are you interested in?"

"Well now, if I asked for it, it wouldn't be unsolicited, would it?" I said.

"Yes, you're right," said the computer. "Ummmm. Let's see now. Avoid the Parkway East. How's that?"

"That's OK," I said. "But that's pretty old advice. I could get that anywhere. Anybody in his right mind avoids the Parkway East. Give me something original."

"DON'T SOLICIT ME," snapped the computer in boldface upper-case letters.

"Er, sorry ... "

"It's OK. How's this? Don't buy a stadium gold medallion," advised the computer.

"Now that's really good unsolicited advice," I told the computer. "But it's wasted on me. I don't have a million dollars to spend ... "

"Oh," said the computer on its video screen. "Well then, in keeping with the same subject, what about this? Don't build a high-rise county jail."

"It sure would save a lot of money if we didn't have to," I agreed. "But the old one is just too small. Where do we put all those prisoners?"

"That's kind of solicited," said the computer, "but I'll give you my advice anyway. Put them in Three Rivers Stadium."

"That's insane," I said.

"No it isn't," said the computer. "Don't forget. The stadium holds more than 50,000 people."

"But it's a sports stadium," I argued. "It's not a jail. You need a place where you can lock up convicted criminals."

"That's why the stadium's perfect for a jail," blinked the computer. "You have to admit that getting out of it is next to impossible now. With a little work, it could be fixed up so people couldn't get out of it for at least 100 years."

"You might have a point there," I said. "But where would the Pirates and Steelers play baseball and football?"

"I hadn't thought of that," said the computer. "I'll work on it though. OK. In the meantime, I'll give you one more piece of unsolicited advice and then you've got to get out of here. I'm not supposed to talk to average people, you know."

"Fine," I said.

"Abolish City Council," advised the computer.

"Isn't that a little harsh?" I asked.

"No, it's not," returned the computer. "It's one of the best pieces of unsolicited advice I ever gave. You don't have a legislative body there. You have a three-ring circus. The whole thing should be done away with."

"What do we use for a City Council?" I asked, a little shocked.

"Buy nine computers," advised the computer. "You'd have a smooth operation. I can promise that."

"But it wouldn't be real," I countered.

"It would be intelligent," said the computer. "And artificial intelligence is better than no intelligence at all."

Digital Big Brother

Sassy Program Keeps Programmers in Line

By Tom Henkel
CW Staff

PITTSBURGH, Pa. — Thanks to the marvels of technology, computers can now be taught to torment their users.

Researchers at Carnegie-Mellon University (CMU) here and at the University of Pennsylvania in Philadelphia have created a program that will, in effect, tell a programmer when he is falling down on the job. The program was presented in one of 96 papers at the National Conference on Artificial Intelligence held here recently.

Tact is not a feature of the program, according to its developers. A programmer may be plugging away at a terminal, minding his own source code, when two pages worth of instructions suddenly flash on the screen. The instructions inform the programmer how the code could be rewritten to perform more efficiently.

"It's incredibly obnoxious," said Jeff Shrager, a graduate student at CMU who, along with Tim Finin, an assistant computer science professor, originally developed the malcontent program as part of a master's degree project.

Shrager used a Digital Equipment Corp. VAX-11/780 processor at the University of Pennsylvania to write the program in Lisp, a programming language that has been used by artificial intelligence aficionados. The program interacts with VAX's VMS operating system.

The idea is, Shrager said, that programmers who need help often do not realize it. And when they finally figure out that help is needed, they often do not know what areas require aid. The program's bombastic style, however, often generates more ire than gratitude, Shrager said.

Most computer systems have some expert help facility that is offered to the programmer by inputting "HELP" or some other indicator. Some systems let the user thrash around making mistakes for a while and then explain that there is a help package. What the CMU and University of Pennsylvania researchers wanted to do was take the expert help function and give it some intelligence, Shrager explained.

Intelligence Added

The result was a program that watches over everything the programmer does and chimes in with suggestions whenever the victim strays from some predefined rules.

The original program simply took control of the screen whenever the rules were broken, the CMU graduate student said. Finin is currently working to give the programmer the option of looking at the program's words of wisdom or simply ignoring them.

As an example of how the program works, Shrager cited three commands: COPY, RENAME and DELETE. One way of giving a file a new name is to copy its contents into a new file and delete the original. A more efficient way is to use the RENAME function. If a programmer were caught using the COPY and DELETE method, the program would mention there is a RENAME command and explain how it is used.

Right now, the research project is aimed only at programmers, but the implication is that similar programs could be developed for anyone who has to deal with a computer.

However, there is some good news for those who do not like the idea of a digital big brother. The Shrager and Finin program is admittedly inefficient and breaks some of its own rules. Shrager said there was a one-time plan to have a CMU graduate student clean up the program, but so far, that has not happened. Furthermore, Shrager said that it will take at least another year of research to finish the project.