Verification, The Key to AI

11/15/2001

It is a bit unseemly for an AI researcher to claim to have a special insight or plan for how his field should proceed. If he has such, why doesn't he just pursue it and, if he is right, exhibit its special fruits? Without denying that, there is still a role for assessing and analyzing the field as a whole, for diagnosing the ills that repeatedly plague it, and to suggest general solutions.

The insight that I would claim to have is that the key to a successful AI is that it can tell for itself whether or not it is working correctly. At one level this is a pragmatic issue. If the AI can't tell for itself whether it is working properly, then some person has to make that assessment and make any necessary modifications. An AI that can assess itself may be able to make the modifications itself.

The Verification Principle:

> An AI system can create and maintain knowledge only to the extent that it can verify that knowledge itself.

Successful verification occurs in all search-based AI systems, such as planners, game-players, even genetic algorithms. Deep Blue, for example, produces a score for each of its possible moves through an extensive search. Its belief that a particular move is a good one is verified by the search tree that shows its inevitable production of a good position. These systems don't have to be told what choices to make; they can tell for themselves. Image trying to program a chess machine by telling it what kinds of moves to make in each kind of position. Many early chess programs were constructed in this way. The problem, of course, was that there were many different kinds of chess positions. And the more advice and rules for move selection given by programmers, the more complex the system became and the more unexpected interactions there were between rules. The programs became brittle and unreliable, requiring constant maintainence, and before long this whole approach lost out to the "brute force" searchers.

Although search-based planners verify at the move selection level, they typically cannot verify at other levels. For example, they often take their state-evaluation scoring function as given. Even Deep Blue cannot search to the end of the game and relies on a human-tuned position-scoring function that it does not assess on its own. A major strength of the champion backgammon program, TD-Gammon, is that it does assess and improve its own scoring function.

Another important level at which search-based planners are almost never subject to verification is that which specifies the outcomes of the moves, actions, or operators. In games such as chess with a limited number of legal moves we can easily imagine programming in the consequences of all of them accurately. But if we imagine planning in a broader AI context, then many of the allowed actions will not have their outcomes completely known. If I take the bagel to Leslie's office, will she be there? How long will it take to drive to work? Will I finish this report today? So many of the decisions we take every day have uncertain and changing effects. Nevertheless, modern AI systems almost never take this into account. They assume that all the action models will be entered accurately by hand, even though these may be most of the knowledge in or ever produced by the system.

Finally, let us make the same point about knowledge in general. Consider any AI system and the knowledge that it has. It may be an expert system or a large database like CYC. Or it may be a robot with knowledge of a building's layout, or knowledge about how to react in various situations. In all these cases we can ask if the AI system can verify its own knowledge, or whether it requires people to intervene to detect errors and unforeseen interactions, and make corrections. As long as the latter is the case we will never be able to build really large knowledge systems. They will always be brittle and unreliable, and limited in size to what people can monitor and understand themselves.

> "Never program anything bigger than your head"

And yet it is overwhelmingly the case that today's AI systems are *not* able to verify their own knowledge. Large ontologies and knowledge bases are built that are totally reliant on human construction and maintainence. "Birds have wings" they say, but of course they have no way of verifying this.