Self-Programming through Imitation

J Storrs Hall

June 30, 2011

Abstract

The processes by which humans extend their own programming differ significantly from the way in which, for example, a human would program a robot. Imitation is a process of self-programming which seems subjectively simple; but, as with many other cognitive abilities, this apparent simplicity is deceptive. We attempt to elicit some of the architectural capabilities a system needs to imitate in a general and robust fashion.

Programming

Learning a new skill can be considered to be constructing a program to perform an action, recognize an object or concept, or predict the consequence of an action or situation. However, the construction of such a program differs in every essential from the kind of programming which is currently done to construct AIs and robots.

- Programming in sequential codes is one of the more intellectually challenging tasks people do. Many can never learn the skill at all.
- Yet virtually everybody learns skills that have yet to be programmed by the cleverest of programmers: fluency in natural language and ethics, recognizing people, walking into random houses and making coffee.
- Thus people are programming themselves by some other method.
- Much of what we learn is by imitation. A "program" in this sense is simply the memory of what is being imitated.
- People also learn by practice and experience. Practicing an observed skill can optimize the imitation program.

Architecture

There have been many attempts over the course of AI research to imitate the higher mammals' ability to imitate. None has had notable success (*cf.* "case-based reasoning"). It is the contention of this position paper that robust, general imitation requires a number of architectural capabilities that are not present in most current systems:

- An abstraction hierarchy. One never sees *exactly* the action one wants to imitate. Thus one is always matching something at some level of abstraction above the concrete present, and backfilling from the part of the memory trace that matches to the actual situation.
- Analogical quadrature. In situation C, find D such that C:D :: A:B, where A:B is a similar situation and the resulting action. The classic implementation of this in mainstream AI is Mitchell and Hofstadter's Copycat.
- Plastic representations. Abstraction can be enhanced by inventing new categories under the pressure of the situation, which retain the salient aspects of the memory to be imitated but exclude those which do not fit the new situation.
- Metric representations. One of the main reasons case-based reasoning fell short was that strictly symbol-based representations too often fail to capture salient similarities.
- Feedback in the hierarchy. The key to Copycat's success was its ability to use "pressures" from the target situation to guide the interpretation of the example situation. Expansion of the ontological framework must continue to allow these pressures to operate properly.
- Massively parallel associative processing, preferably in hardware but could be simulated with enough general-purpose processing power. The brain, at a low level, can be thought of as a massively parallel collection of hardware pattern matchers.

Example

For concreteness, let us consider the apparently simple blocks-world task of watching a human place one block on another, and attempting to imitate the feat. The following obstacles arise:

- The robot does not have access to the identical blocks the human used, or even if it does, those blocks are already stacked. The robot must find appropriate blocks of its own to stack.
- The robot's point of view of the stacking is different. The actual pictures of the human doing the task returned by the robot's visual system will be drastically different from those returned when the robot is itself attempting it. Substantial translation must occur. Must its gripper approach the block from the same direction? Must it use its right hand if the human did?

- The human has stacked a small green block onto a large red one. The robot has only a large green block and a small red one. Is size or color the more salient property?
- The human stacks the blocks, points at the robot's blocks, says "Now do this with those," puts his hands in his pockets, and whistles a tune. How much of this behavior should the robot attempt to imitate? What are the boundaries along which you cut an imitation program out of an extended observation?

Architecture redux

An abstraction hierarchy

Minsky and Papert's Society of Mind architecture was inspired by the architecture developed for robot control. It features a hierarchy of ontologically separate concerns, with higher-level plans driving subgoals driving specific action sequences driving low-level motor actions. Later efforts ranging through Albus, Brooks, and Ng have adopted similar abstraction hierarchies. It would not be going too far to say they had been forced to adopt them by the nature of the task.

The sequence of reaching for, grasping, lifting, placing, and releasing a block should be the same in the robot's own action as in its exegesis of the human's example. None of the underlying motor operations, however, refer to the same positions, blocks, or even arms.

When the robot uses experience to guide the lower-level motor sequence, it is essentially imitating itself. In this case the arms will be the same and the backfill will take place at the level of specific positions, sizes, weights, and so forth.

One particularly important aspect of the hierarchy is that it allows the robot to be imitating several different actions or memories at once, at different levels.

Analogical quadrature

It should now be clear that analogical quadrature is the fundamental driver for the imitative engine. In particular, an observed or remembered situation A is matched with the present situation C and the same mappings are used to derive the action D from the example B. However, it should also be noted that the same mechanism can be used simply to predict or simulate: find the best match in memory to the situation, with an unspecified outcome rather than action.

This implies that the memory should be organized as associative triples

$$\text{situation} \Longrightarrow \text{action} \Longrightarrow \text{resultant situation}$$

with the standard associative ability to match on any field(s) and retrieve the unspecified one(s).

This ability to generate a causal semantics from the memory enables the system to parse observed actions in a manner reminiscent of "explanation-based generalization."

Plastic representations

In the example above, the robot is faced with blocks which do not match the example, and must make a choice of whether to ignore size or color. The capability necessary is to have a representation scheme which is more like a grammar than a record with fixed slots. The grammar can generate a variety of parses for the object to be represented. The representation process is guided by context – just as it must be in the case of natural language.

Metric representations

Copycat used clouds of nodes in a semantic network, each with a continuously variable activation level, to represent "fluid concepts." The major drawback to this or any similar scheme is that the basic network, and all the "codelets" which operated on it, had to be hand-crafted.

We have taken the experimental path of forming representations using numeric vectors. These can represent such activated clouds, among many other things. They have a built-in metric which allows the use of nearest-neighbor and other weak methods when nothing more appropriate is available. There exists a substantial corpus of standard statistical and regression techniques which has been found useful by physical scientists for decades or even centuries. These include well-understood techniques for abstraction, such as principal components.

Feedback in the hierarchy

The pressures or context signals guiding the process of variable representation must propagate not only within an ontological level but between them, and must flow in both directions. Context signals serve the function referred to as priming in human cognitive psychology.

Massively parallel associative processing

The question of associative processing hardware is clearly a practical rather than a theoretical issue. Any associative algorithm could easily be simulated in a conventional architecture. However, besides the practical issue of real-time operation, there is a an insidious pressure on the algorithmicist due to the paucity of processing power. This typically takes the form of substituting indexing schemes and arbitrary cutoffs for full associative searches.

For example, consider a Copycat-like matching scheme between a given situation and memories of previous and observed actions. The proper retrieval action is to perform the full match between the situation and each memory, allowing each memory to be potentially perceived with an interpretation most appropriate to the current case. With constrained processing power, the temptation is high to pre-process the memories, biasing the interpretations, and ultimately preventing potential matches. This in turn produces exactly the brittleness that has hampered existing attempts at memory-based planning and reasoning.

Process

We claim that the imitation process works in roughly the following way:

- An action is observed.
- It is parsed with an "action grammar" into abstraction levels, subgoals, etc.
- The parsing process is strongly influenced by the necessity of matching the resultant representation to the situation in which the robot will be attempting to imitate the action.
- To imitate the action, the superstructure (higher-level portions of the parse tree) can be adopted more or less unchanged.
- Lower-level portions must be adapted to the differing circumstances of the execution environment.
- Analogical quadrature is used to adapt intermediate-level actions.
- Where AQ founders, search among all the robot's action memories for ones with matchable situations and goals.
- At the lowest levels, physical trial and error is employed. Practice is valuable because it fills in the space of memories, allowing imitation of one's own successful trials, and interpolation, using metric representations, from near misses.