Skill Learning Using A Bottom-Up Hybrid Model

Ron Sun, Edward Merrill, Todd Peterson
The University of Alabama
Tuscaloosa, AL 35487, USA
rsun@cs.ua.edu

Abstract

This paper presents a skill learning model CLARION. Different from existing models of mostly high-level skill learning that use a top-down approach (that is, turning declarative knowledge into procedural knowledge), we adopt a bottom-up approach toward low-level skill learning, where procedural knowledge develops first and declarative knowledge develops from it. CLARION which follows this approach is formed by integrating connectionist, reinforcement, and symbolic learning methods to perform on-line learning. We compare the model with human data in a minefield navigation task. A match between the model and human data is observed in several comparisons.

1 Introduction

Skills vary in complexity and the degree of cognitive involvement. They range from simple motor movements and other routine tasks in everyday activities to high-level intellectual skills. We want to study "lower-level" cognitive skills, which have not received sufficient research attention. One type of task that exemplifies what we call low-level cognitive skill is reactive sequential decision making (Sun and Peterson 1995). It involves an agent selecting and performing a sequence of actions to accomplish an objective on the basis of moment-to-moment information (hence the term "reactive"). An example of this kind of task is the minefield navigation task developed at The Naval Research Lab (see Gordon et al. 1994). This kind of task setting appears to tap into real-world skills associated with decision making under conditions of time pressure and limited information. Thus, the results we obtain from human experiments will likely be transferable to real-world skill learning situations. Yet this kind of task is suitable for computational modeling given the recent development of machine learning techniques (Sun et al 1996, Watkins 1989).

The distinction between procedural knowledge and declarative knowledge has been made in many theories of learning and cognition (for example, Anderson 1982, 1993, Keil 1989, Damasio et al. 1994, and Sun 1995). It is believed that both procedural and declarative knowledge are essential to cognitive agents in complex environments. Anderson (1982) originally proposed the distinction based on data from a variety of skill learning studies, ranging from arithmetic to geometric theorem proving, to account for changes resulting from extensive practice. Similar distinctions have been made by other researchers based on different sets of data, in the areas of skill learning, concept formation, and verbal informal reasoning (e.g., Fitts and Posner, 1967; Keil, 1989; Sun, 1995).

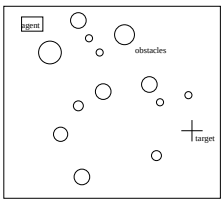

Figure 1: Navigating Through Mines

Most of the work in skill learning that makes the declarative/procedural distinction assumes a top-down approach; that is, learners first acquire a great deal of explicit declarative knowledge in a domain and then through practice, turn this knowledge into a procedural form ("proceduralization"), which leads to skilled performance. However, these models were not developed to account for skill learning in the absence of, or independent from, prexisting explicit domain knowledge. Several lines of research demonstrate that individuals can learn to perform complex skills without first obtaining a large amount of explicit declarative knowledge (e.g., Berry and Broadbent 1988, Stanley et al 1989, Lewicki et al 1992, Willingham et al 1992, Reber 1989, Karmiloff-Smith 1986, Schacter 1987, and Schraagen 1993). In research on *implicit learning*, Berry and Broadbent (1988), Willingham et al (1992), and Reber (1989) expressly demonstrate a *dissociation* between explicit knowledge and skilled performance in a variety of tasks including dynamic decision tasks (Berry and Broadbent 1988), artificial grammar learning tasks (Reber 1989), and serial reaction tasks (Willingham et al 1992). Berry and Broadbent (1988) argue that the psychological data in dynamic decision tasks are not consistent with exclusively top-down learning

models, because subjects can learn to perform the task without being provided a priori declarative knowledge and without being able to verbalize the rules they used to perform the task. This indicates that procedural skills are not necessarily accompanied by explicit declarative knowledge, which would not be the case if top-down learning is the only way to acquire skill. Willingham et al (1989) similarly demonstrate that procedural knowledge is not *always* preceded by declarative knowledge in human learning, and show that declarative and procedural learning are not necessarily correlated. There are even indications that explicit knowledge may arise from procedural skills in some circumstances (see Stanley et al 1989). Using a dynamic decision task, Stanley et al. (1989) found that the development of declarative knowledge paralleled but lagged behind the development of procedural knowledge.

Similar claims concerning the development of procedural knowledge prior to the development of declarative knowledge have surfaced in a number of research areas outside the skill learning literature and provided additional support for the bottom-up approach. *Implicit memory* research (e.g., Schacter 1987) demonstrates a dissociation between explicit and implicit knowledge/memories in that an individual's performance can improve by virtue of implicit "retrieval" from memory and the individual can be unaware of the process. This is not amenable to the exclusively top-down approach. *Instrumental conditioning* also reflects a learning process that differs from the top-down approach, because the process is typically non-verbal and involves the formation of action sequences without requiring a priori explicit knowledge. It may be applied to simple organisms as well as humans (Gluck and Bower 1988). In *developmental psychology*, Karmiloff-Smith (1986) proposed the idea of "representational redescription". During development, low-level implicit representations are transformed into more abstract and explicit representations and thereby made more accessible. This process is not top-down either, but in the opposite direction.

2 The Model

The difference between declarative and procedural knowledge leads naturally to "two-level" architectures (Sun 1995). We thus developed the model CLARION, which stands for *Connectionist Learning with Adaptive Rule Induction ON-line* (Sun et al 1996). It embodies the distinction of declarative and procedural knowledge (or, conceptual and subconceptual knowledge), and it performs learning in a bottom-up direction. It consists of two main components: the top level encodes explicit declarative knowledge in the form of propositional rules, and the bottom level encodes implicit procedural knowledge in neural networks. In addition, there is an episodic memory, which stores recent experiences in the form of "input, output, result" (i.e., stimulus, response, and consequence).

A high-level pseudo-code algorithm that describes CLARION is as follows:

1. Observe the current state x.
2. Compute in the bottom level the Q-value of each of the possible actions (a_i's) associated with the perceptual state x: $Q(x, a_1)$, $Q(x, a_2)$, ..., $Q(x, a_n)$.
3. Find out all the possible actions (b_1, b_2, ..., b_m) at the top level, based on the the perceptual information x and other available information (which goes up from the bottom level) and the rules in place at the top level.
4. Compare the values of a_i's with those of b_j's (which are sent down from the top level), and choose an appropriate action a.
5. Perform the action a, and observe the next state y and (possibly) the reinforcement r.
6. Update the bottom level in accordance with the *Q-Learning-Backpropagation* algorithm, based on the feedback information.
7. Update the top level using the *Rule-Extraction-Refinement* algorithm.
8. Go back to Step 1.

In the bottom level, a Q-value is an evaluation of the "quality" of an action in a given state: $Q(x, a)$ indicates how desirable action a is in state x. We can choose an action based on Q-values. To acquire the Q-values, supervised and/or reinforcement learning methods may be applied. A widely applicable option is the *Q-learning* algorithm (Watkins 1989), a reinforcement learning algorithm. In the algorithm, $Q(x, a)$ estimates the maximum discounted cumulative reinforcement that the agent will receive from the current state x on. The updating of $Q(x, a)$ is based on minimizing $r + \gamma e(y) - Q(x, a)$, where γ is a discount factor and $e(y) = \max_a Q(y, a)$. Thus, the updating is based on the *temporal difference* in evaluating the current state and the action chosen: In the above formula, $Q(x, a)$ estimates, before action a is performed, the (discounted) cumulative reinforcement to be received if action a is performed, and $r + \gamma e(y)$ estimates the (discounted) cumulative reinforcement that the agent will receive, after action a is performed; so their difference (the temporal difference in evaluating an action) enables the learning of Q-values that approximate the (discounted) cumulative reinforcement. Using Q-learning allows sequential behavior to emerge in an agent. Through successive updates of the Q function, the agent can learn to take into account future steps in longer and longer sequences.

To implement Q functions, we chose to use a four-layered network (see Figure 2), in which the first three layers form a (either recurrent or feedforward) backpropagation network for computing Q-values and the fourth layer (with only one node) performs stochastic decision making. The output of the third layer (i.e., the output layer of the backpropagation network) indicates the Q-value of each action (represented by an individual node), and the node in the fourth layer determines probabilistically the action to be performed based on a Boltzmann distribution (i.e., Luce's choice axiom; Watkins 1989). This learning process performs both structural credit assignment (with backpropaga-

tion), so that the agent knows which element in a state should be assigned credit/blame, as well as temporal credit assignment, so that the agent knows which action leads to success or failure. This learning process enables the development of procedural skills potentially solely based on the agent independently exploring a particular world on a continuous and on-going basis.

In the top level, declarative knowledge is captured in a simple propositional rule form. To facilitate correspondence with the bottom level and to encourage uniformity and integration (Clark and Karmiloff-Smith 1993), we chose to use a localist connectionist model for implementing these rules (e.g., Sun 1992, Towell and Shavlik 1993). Basically, we translate the structure of a set of rules into that of a network. For each rule, a set of links are established, each of which connects a node representing a concept in the condition of a rule to the node representing the conclusion of the rule. For more complex rule forms including predicate rules and variable binding, see Sun (1992).

To fully capture bottom-up learning processes, we devised an algorithm for learning declarative knowledge (rules) using information in the bottom level (the *Rule-Extraction-Refinement* algorithm). The basic idea is as follows: if an action decided by the bottom level is successful then the agent extracts a rule (with its action corresponding to that selected by the bottom level and with its conditions corresponding to the current sensory state), and adds the rule to the top-level rule network. Then, in subsequent interactions with the world, the agent refines the extracted rule by considering the outcome of applying the rule: if the outcome is successful, the agent may try to generalize the conditions of the rule to make it more universal; if the outcome is not successful, then the conditions of the rule should be made more specific and exclusive of the current case.

We perform rule extraction at each step, based on the following information: (x, y, r, a), where x is the state before action a is performed, y is the new state entered after an action a is performed, and r is the reinforcement received after action a. Rules are in the following form: *conditions* $\longrightarrow$ *action*, where the left-hand side is a conjunction of individual conditions each of which refers to the value of an element in the (sensory) input state. Three different criteria can be used for rule learning at each step: (1) direct reinforcement received at a step, (2) temporal difference (as used in updating Q-values), and (3) maximum Q-values in a state. We adopt a three-phase approach, with each phase lasting for a certain number of episodes. Phase transition can be automatically determined based on the current performance level of the model. At each step, we apply the current-phase criterion to determine whether we should construct a rule. If so, a rule is wired up in the rule network. After rules are extracted, at each step, the algorithm reexamines the rules matching the current step to decide if each of them should be kept, revised, or discarded. See Sun et al. 1996 for the full details of rule learning.

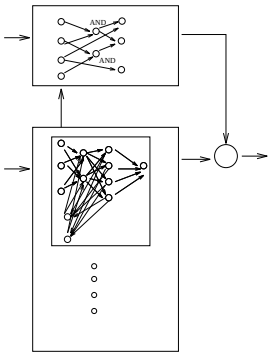

Figure 2: The implementation of CLARION.

Step 4 is for making the final decision on which action to take by incorporating outcomes from both levels. We combine the corresponding values for an action from the two levels by a weighted sum; that is, if the top level indicates that action a has an activation value v (which should be 0 or 1 as rules are binary) and the bottom level indicates that a has an activation value q (the Q-value), then the final outcome is $w_1 * v + w_2 * q$. Stochastic decision making with Boltzmann distribution (based on the weighted sums) is then performed. Figure 2 shows the two levels of the model.

3 Experiments

In all of the human experiments, subjects were seated in front of a computer monitor that displayed an instrument panel containing several gauges that provided current information (see Figure 3). The following instruction was given to explain the setting:

> I. Imagine yourself navigating an underwater submarine that has to go through a minefield to reach a target location. The readings from the following instruments are available:
>
> (1) Sonar gauges show you how close the mines are to the submarine. This information is presented in 8 equal areas that range from 45 degrees to your left, to directly in front of you and then to 45 degrees to your right. Mines are detected by the sonars and the sonar readings in each of these directions are shown as circles in these boxes. A circle becomes larger as you approach mines in that direction.
>
> (2) A fuel gauge shows you how much time you have left before you run out fuels. Obviously, you must reach the target before you run out of fuel to successfully complete the task.
>
> (3) A bearing gauge shows you the direction of the target from your present direction; that is, the angle from your current direction of motion to the direction of the target.
>
> (4) A range gauge shows you how far your current location is from the target.
>
> II. At the beginning of each episode you are located on one side of the minefield and the target is on the other side of the minefield. You task is to navigate through the minefield to get to the target before you run out of fuel. An episode ends when: (a) you get to the goal (success); (b)

you hit a mine (failure); (c) you run out of fuel (failure).

A random mine layout was generated for each episode. This setting is *stochastic* and *non-Markovian*. Five training conditions were used:

- The standard training condition. Subjects received five blocks of 20 episodes on each of five consecutive days (100 episodes per day). In each episode the minefield contained 60 mines. The subjects were allowed 200 steps.
- The verbalization training condition. This condition was identical to the standard training condition except that subjects were asked to step through slow replays of selected episodes and to verbalize what they were thinking during the episode. Subjects received replays on the first, third, and fifth days of training. The subjects were replayed five episodes after the first block of 20 episodes and five episodes after the fifth block of 20 episodes on these days.
- The over-verbalization training condition. In this condition subjects were presented replays of 15 of their first 25 episodes, and asked to verbalize during the slow playback. Replay of an episode occurred immediately after the subject finished the episode.
- The 30-to-60 transfer condition. This condition was also identical to the standard training condition except that subjects performed the task with 30 mines on the first two days of training and switched to 60 mines starting the third day.
- The mixed training condition. "Mixed" refers to the fact that mine density was manipulated during training. Subjects performed the task with 30, 50, 70, or 90 mines. Subjects received eight blocks of 10 episodes per day over five days, two at each mine density. Order of presentation was randomized.

In CLARION each gauge was represented by a set of nodes that corresponded to what human subjects would see on screen. This input setup yielded a total of 43 primary perceptual inputs. Thus, there were more than 10^{12} possible input states. Thus the model had to deal with the problem of high dimensionality. As a result, a lookup table implementation for Q-learning at the bottom level was not possible (Tesauro 1992, Lin 1992). To deal with the situation. a functional approximator such as backpropagation networks must be used. Also in correspondence to the human experimental setting, the action outputs consisted of two clusters of nodes representing turn and speed.

The model started out with no more a priori knowledge about the task than a typical human subject, so that bottom-up learning can be captured. The bottom level contained randomly initialized weights (with a pre-chosen, fixed topology). The top level started empty and contained no a priori knowledge

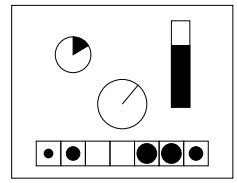

Figure 3: The Navigation Input
The display at the upper left corner is the fuel gauge; the vertical one at the upper right corner is the range gauge; the round one in the middle is the bearing gauge; the 7 sonar gauges are at the bottom.

about the task, either in the form of instructions or instances. The episodic memory was empty at the beginning. There was no supervised learning (i.e., no teacher input). The reinforcement signals embodied some a priori notions regarding getting close to target and avoiding explosion that were also provided to human subjects through instructions. The learning algorithm with all the requisite parameters was pre-set, presumably reflecting the learning mechanisms in humans.

The results of the experiments are analyzed as follows.

The standard training condition. We obtained performance data over 500 episodes per subject. We averaged the data over 10 human subjects. We did the same with the model: Each model run was initialized with different random number sequences and thus produced different results; we averaged 10 such runs in exact correspondence with human experiments (i.e., we did not tune the random number sequences to generate a match, but randomly set seeds for random number generators, analogous to random selection of human subjects in this experiment). We compared *average* success rates because in this way we can eliminate the uninteresting impact of individual differences and instead focus on essential features of learning in this task. These data are presented in Figure 4. Both sets of data were best fit by power functions (for failure rate). The degree of similarity is evident. A Pearson product moment correlation coefficient was calculated (treating blocks as individuals and human versus model as the X and Y variables). The analysis yielded a high positive correlation ($r = .82$), indicating a high degree of similarity between human subjects and model runs.

The verbalization training condition. Obviously, we could not require verbalization from the model. However, we posited that much of the effect of verbalization on learning was associated with rehearsing previous steps and episodes (although there may be additional factors involved). Thus for the model, we used episode memory playback (Lin 1992) in a first attempt to capture this effect. Episode memory playback involves training the model with previously performed episodes between blocks of actual trial episodes in exactly the same manner as in human experiments. In this case, the data from 5 human subjects was com-

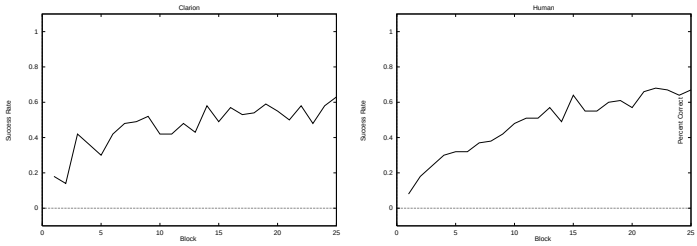

Figure 4: The learning curves in terms of success rates in the standard condition. The right side is the human data and the left side is the model data.

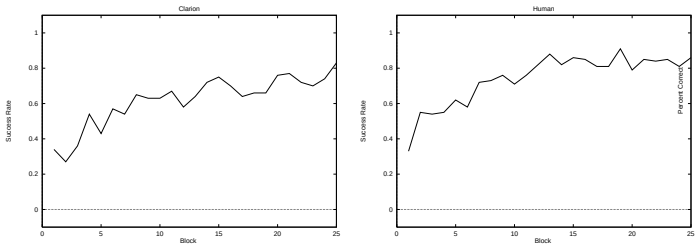

Figure 5: The learning curves in terms of success rates in the verbalization condition.

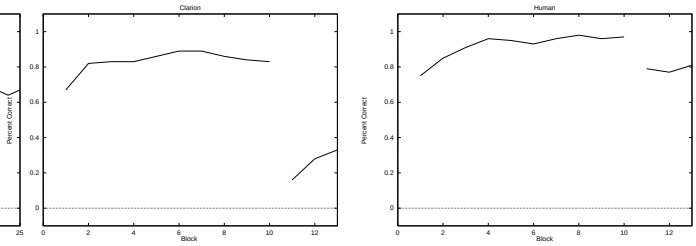

Figure 6: The 30-to-60 transfer data in terms of success rates.

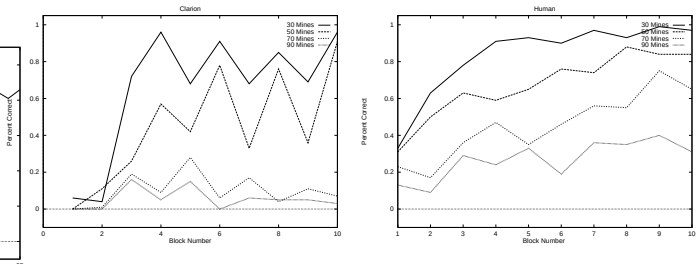

Figure 7: Average success rates for each mine densities in the mixed condition.

pared to that of 5 model runs. Data was averaged for each of 25 blocks (see Figure 5). Again, both sets of data were highly similar and both were best fit by power functions. We also calculated a Pearson product moment correlation coefficient, which yielded a high positive correlation (r = .84).

We subsequently compared the changes in performance due to verbalization for the human subjects and the model runs. This was done by averaging failure rates across blocks separately for each human subject and for each model run and subjecting that data to a 2×2 ANOVA. The analysis of these data indicated the both groups exhibited a significant increase in performance due to verbalization ($p < .01$), and that the changes due to verbalization for the two groups were not significantly different (52 to 25 percent failure rate for the human subjects versus 53 to 38 percent failure rate for the model runs). The effect of explication of implicit knowledge which likely results from verbalization was captured through the usual rule learning process, which was also at work during episode replay.

The 30-to-60 transfer condition. Subjects were first trained on 30-mine minefields, and then transferred to 60-mine minefields. The model was tested under the same condition. Both human and model data were averaged over 10 subjects. Comparing the human and model data (see Figure 6), we noticed that both learned well at 30 mines, although human data was slightly better. When transferred to 60 mines, both exhibited a significant drop in performance, although the model exhibited a deeper drop. Specifically, we compared performance of the last block before the change in mine density and the first block after the change. Success rates were 98% and 79% for the human subjects and 83% and 26% for the model runs respectively. The drops were both statistically significant. At first look, it might appear that the drop in performance for the model runs was much greater than that for the human subjects. However, this might not be a fair assessment in that we did not allow the model runs to reach the same performance as the human subjects before changing the mine density. Indeed, the 5 highest performing of the model runs before the change performed 8 times better after the change than did the 5 lowest performing ones.

The mixed training condition. We plotted learning curves in terms of success rates for each mine density separately. The data were averaged over 8 human subjects and 8 model runs, respectively. The average curves are shown in Figure 7. We calculated overall success rates for each of the mine densities. Both the human subjects and model runs performed best with the lowest mine density and performance decreased with each increase in the number of mines. Thus, we observed a similar pattern. The drop in performance was roughly the same for human subjects and model runs between the 30 and 50 mine densities (16% versus 13%, respectively). We do not know for sure what accounts for the failure of the model at the 70 and 90 mine densities. However, questionnaires completed by the human subjects indicated that they treated the higher density conditions as different from the lower density conditions. Because the model runs did not "start over" at each density, they were applying what was learned to conditions in which it did not work. In contrast, human subjects could sense the change in conditions and discard their old strategies.

The over-verbalization condition. Human subjects under the over-verbalization condition failed to learned. During the 25 episodes of training, their success rates were well below 10%, compared with the

33% performance for the subjects under the (sparse) verbalization condition. If we eliminate one subject who performed at 60%, the remaining subjects only achieved approximately 3% success rate. CLARION accounts for this phenomenon by positing that too much verbalization (e.g., verbalizing for more than half of the training episodes) caused the learner to switch to a completely explicit mode of learning; they tended to rely completely on the top-level learning mechanism and shut down the bottom level. This is consistent with the similar hypothesis by Stanley et al (1989), for explaining their findings regarding the difficulty their subjects had in learning a dynamic decision task after being given instructions that encouraged them to be explicit. Schooler et al (1993) also reported that requiring verbalization impaired subjects' ability to solve problems that require "insight", by forcing them to be overly explicit. CLARION explains the findings readily with the shut-down mechanism. The top-level learning mechanism when disconnected from the bottom level, clearly has trouble learning this kind of sequential task, because of its lack of a temporal credit assignment process (comparable in power to Q-learning) and its all-or-nothing learning process. On the other hand, in the bottom level, the distributed network representation and learning process that incorporates gradedness and temporal information handle complex sequences well.

Verbalization segments indicating bottom-up learning. The verbalization data we collected from the subjects (under the verbalization training condition) were consistent, in an informal sense, with our assumption of bottom-up learning being prominent in this task setting, as exemplified by the following segments.

> S: I thought about it after I started doing it. I said, look at me .... look what I'm doing. I didn't start thinking about it until I started doing it. I figured out that it started helping me and that's when I started doing it myself. (subj.38)

> S: When I started off ..... I didn't understand at all .... I couldn't grasp the whole sonar concept at all. (subj.38)

> S: So, basically what I do – not thinking about driving a submarine or mine. (subj.38)

> S: When you get in a situation like this, where there are gaps, it's purely instinctual. (subj.37)

> S: That's pretty much I've done the whole game [being instinctual], with the exception of a couple of patterns I've started to recognize. (subj.37)

In sum, the verbalization by the subjects suggested that some degree of bottom-level (implicit) learning/decision making and gradual bottom-up learning existed. This is the kind of learning CLARION was meant to capture.

We also compared the verbalizations of good performers (subjects) vs. poor performers. Our analysis indicated a lack of difference: we failed to notice any significant difference across a variety of measures (such as length of verbalization, detailedness, and types of statements uttered). We suggest that this is one more piece of evidence that indicates the importance/prominence of bottom-level (implicit) learning: The performance is mostly determined by implicit procedural learning, which cannot be easily verbalized, while verbalized explicit knowledge is nonspecific and has relatively minor impact during learning.

4 Conclusions

In sum, we discussed a hybrid connectionist model CLARION as a demonstration of the approach of bottom-up skill learning, which consists of two levels for capturing both procedural and declarative knowledge and performing bottom-up learning. Some degree of match with human data was found across a number of different experimental conditions.

Acknowledgements

This work is supported in part by Office of Naval Research grant N00014-95-1-0440. The simulator was provided by NRL.

References

J. R. Anderson, (1982). Acquisition of cognitive skill. *Psychological Review.* Vol.89, pp.369-406.

D. Berry and D. Broadbent, (1988). Interactive tasks and the implicit-explicit distinction. *British Journal of Psychology.* 79, 251-272.

A. Clark and A. Karmiloff-Smith, (1993). The cognizer's innards: a psychological and philosophical perspective on the development of thought. *Mind and Language.* 8 (4), 487-519.

A. Damasio, (1994). *Decartes' Error.* Grosset/Putnam, NY.

D. Gordon, et al. (1994). *User's Guide to the Navigation and Collision Avoidance Task.* Naval Research Lab. DC.

H. Dreyfus and S. Dreyfus, (1987). *Mind Over Machine: The Power of Human Intuition*, The Free Press, New York, NY.

P. Fitts and M. Posner, (1967). *Human Performance.* Brooks/Cole, Monterey, CA.

M. Gluck and G. Bower, (1988). From conditioning to category learning. *Journal of Experimental Psychology: General.* 117 (3), 227-247.

W. James, (1890). *The Principles of Psychology.* Dover, NY.

A. Karmiloff-Smith, (1986). From meta-processes to conscious access. *Cognition.* 23. 95-147.

F. Keil, (1989). *Concepts, Kinds, and Cognitive Development.* MIT Press. Cambridge, MA.

P. Lewicki, et al. (1992). Nonconscious acquisition of information. *American Psychologist.* 47, 796-801.

L. Lin, (1992). Self-improving reactive agents based on reinforcement learning, planning, and teaching. *Machine Learning.* Vol.8, pp.293-321.

A. Reber, (1989). Implicit learning and tacit knowledge. *Journal of Exp Psychology: General.* 118 (3), 219-235.

J. Schooler, S. Ohlsson, and K. Brooks, (1993). Thoughts beyond words: when language overshadows insight. *Journal of Experimental Psychology: General.* 122 (2). 166-183.

D. Schacter, (1987). Implicit memory: History and current status. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 13, 501-518.

R. Shiffrin and W. Schneider, (1977). Controlled and automatic human information processing II. *Psychological Review.* 84. 127-190.

P. Smolensky, (1988). On the proper treatment of connectionism. *Behavioral and Brain Sciences*, 11(1):1–74.

W. Stanley, et al (1989). Insight without awareness *Quarterly Journal of Experimental Psychology.* 41A (3), 553-577.

R. Sun, (1992). On Variable Binding in Connectionist Networks, *Connection Science*, Vol.4, No.2, pp.93-124.

R. Sun, (1995). Robust reasoning: integrating rule-based and similarity-based reasoning. *Artificial Intelligence.* 75, 2. 241-296.

R. Sun, T. Peterson, and E. Merrill, (1996). Bottom-up skill learning. Technical Report TR-CS-96-0021, University of Alabama. shortened version in *Proc.of 18th Cognitive Science Society Conference*,

T. Tesauro, (1992). Practical issues in temporal difference learning. *Machine Learning.* Vol.8, 257-277.

C. Watkins, (1989). *Learning with Delayed Rewards.* Ph.D Thesis, Cambridge University, Cambridge, UK.

D. Willingham, et al (1989). On the development of procedural knowledge. *Journal of Experimental Psychology: Learning, Memory, and Cognition.* 15, 1047-1060.