Conscious Learning In An Adaptive Software Agent

Uma Ramamurthy[1], Myles Bogner[2], and Stan Franklin[3]

Conscious Software Research Group
Institute of Intelligent Systems
The University of Memphis
Memphis TN 38152
Tel: (901)678-2482, Fax: (901)522-8843
{urmmrthy,mbogner,stan.franklin}@memphis.edu

1 Abstract

This paper describes the "conscious" learning mechanism of an adaptive software agent, Conscious Mattie (CMattie). This conscious software agent takes the role of a departmental seminar organizer. The agent gathers seminar information via email from humans, composes an announcement of the next week's seminars, and mails it to members of a mailing list which she maintains. Through further interaction with seminar organizers, CMattie learns to handle new variations of seminars, such as colloquia. In this paper, the agent's perception and consciousness modules are described. Using a limited capacity global workspace that coordinates with the perception module, CMattie communicates with human seminar organizers. Through such interaction, CMattie gathers missing information and learns new concepts for use in composing the announcements.[4]

2 Introduction

An autonomous agent [7] is a system situated in, and part of, an environment, which senses that environment, and acts on it, over time, in pursuit of its own agenda. It acts in such a way as to possibly influence what it senses in the future. CMattie is an autonomous cognitive agent, having multiple senses, perception, concept formation, attention, problem-solving, decision making, short and long-term memory, learning, and emotions. This agent is intended to be the first implementation of global workspace theory, a psychological theory of consciousness [2, 3]. Global workspace theory postulates that human cognition is implemented by a multitude of relatively small, special purpose processes, almost always unconscious. Coalitions of such processes, when aroused by novel and/or problematic situations, find their way into consciousness, the global workspace. This limited capacity global workspace serves to broadcast the message of the coalition to all the unconscious processors. This allows other processors to be recruited to join in handling the current novel situation, or in solving the current problem.

Conscious software agents [5] adapt and learn by reacting to the changes in their domain, and through their interaction with other agents in their domains, be they human or artificial. Due to this extensive interaction, conscious software agents tend to be social creatures. CMattie functions in an academic setting and "lives" in a UNIX-based system. She gathers information from humans regarding seminars and seminar-like events such as colloquia, theses defense, etc. Using this information, she composes an announcement of the next week's seminars, and mails this announcement weekly to members of a mailing list that she maintains.

CMattie is designed using a multiplicity of artificial intelligence mechanisms. Her modular architecture [4, 14] implements and integrates these several diverse mechanisms. They include behavior networks [12] for action selection, sparse distributed memory [10] for long-term, associative memory, pandemonium theory [9]

[1] Supported in part by Cancer Center Support CORE Grant, P30 CA 21765 and by American Lebanese Syrian Associated Charities (ALSAC)
[2] Supported in part by NSF grant SBR-9720314
[3] Supported in part by NSF grant SBR-9720314 and by ONR grant N00014-98-1-0332
[4] **Keywords:** conceptual learning, consciousness, global workspace, memory, perception

for agent grouping, Copycat architecture [8] and natural language understanding [1] for email comprehension, and case-based memory [11] for intermediate term, episodic memory. Each of these mechanisms has been significantly extended in order to merge with the others, and to meet the needs of this domain.

This article focuses on conscious conceptual learning. This learning arises from what the agent perceives. As discussed below, everything that CMattie perceives becomes conscious. Therefore, to understand this type of learning, the agent's perception and consciousness are first described.

3 Perception

The perception module in CMattie was inspired by and can be thought of as an extension of the Copycat architecture [13]. Copycat is based on the premise that analogy making is a process of high-level perception, and that analogy making lies at the core of understanding. Copycat makes and interprets analogies between situations in a microworld of letter-string analogy problems. Copycat's domain is predefined and fixed; therefore, there is no learning. Since CMattie "lives" in a dynamic domain, her conceptual learning mechanism enables her to perceive this dynamism. The agent's perception involves building instances of known concepts in her domain; learning, detecting and creating new concepts; and making appropriate relations between those concepts. CMattie's perception module, shown in Figure 1, consists of Mail Input and Output, the Slipnet, Working Memory and Case-Based Memory.

Mail Input and Output. This unit provides CMattie's interface to her domain. Using this module, she receives and sends out email messages related to seminars, seminar-like events such as colloquia, and maintenance of the recipient mailing list. Mail input and output can process more than one email message at a time, enabling the perception module to understand emergency events such as a system shutdown message. This aids in maintaining her sense of self-preservation as she proactively reacts to her changing resource needs.

Slipnet. The Slipnet is a network of nodes and links representing CMattie's permanent perceptual concepts. A concept has a core and a set of features representing its basic characteristics. In a given context, a feature might have a specific value. In CMattie, concepts are often defined by a region of nodes and links in the Slipnet. Each of the concepts in the agent may be an individual node or group of nodes. The various nodes are connected to each other through weighted links.

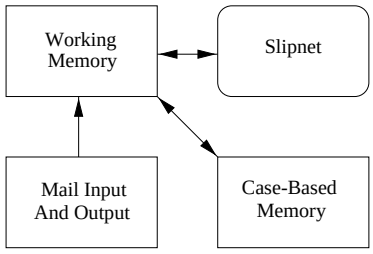

Fig. 1. CMattie's Perception Module

One of the built-in concepts in the perception module is the **Seminar** concept with the following features:

Name of the Seminar
Location where the Seminar is to be held
Day of the week of the Seminar
Speaker of the Seminar
Periodicity of the Seminar
Organizer of the Seminar
Date of the Seminar
Time at which the Seminar is to be held
Title of Talk for the Seminar

Name and **Day** are features of the **Seminar** concept and are concepts themselves, each with a sepa-

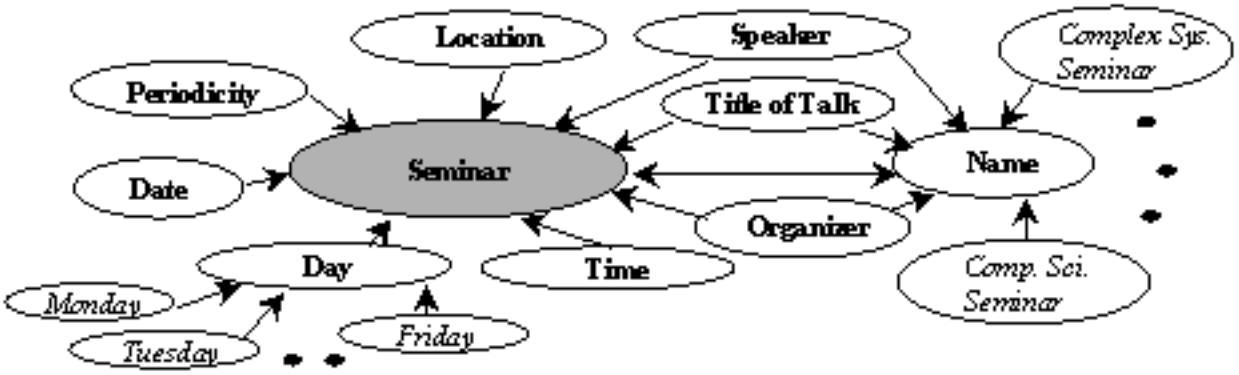

Fig. 2. Segment of the Slipnet in the Perception Module

rate set of features. The **Seminar** concept is deeper than the **Name** and **Day** concepts and has a higher depth value than those two concepts. Depth values aid in the assignment of node activation level.

Codelets can be thought of as small distinct agents designed to perform one task [4, 18]. Each node in the slipnet has one or more codelets associated with it. When an email message is received by mail input, these codelets aid in understanding the message, which is written in natural language. They recognize relevant words and phrases in the received message, and send activation to the appropriate slipnet nodes. A corpus of email messages collected for two years contributed to the building of the slipnet.

Working Memory. This memory holds the contents of the incoming email message. It also holds the perception process' intermediate results, as codelets associated with slipnet nodes operate inside working memory to understand the received email message. The most significant inference made in this process is the categorization of the **type** of the incoming message.

Case-Based Memory. Case-Based memory constitutes CMattie's episodic memory. In it she stores every email message received for a relatively short duration. By looking at the episodes formed from sequences of email messages with organizers, CMattie is able to relate new events to similar past events. She understands these past events using her built-in domain knowledge. Case-Based memory aids her in learning new slipnet concepts through case-based reasoning. This memory acts as an intermediate term memory, and the information stored here is used to learn domain knowledge.

Perception Process. An incoming message is understood when every significant word or phrase has been processed and the type of the email message has been inferred. This information is then transferred by perceptual codelets to the perception registers in the focus, described in the next section. Some of the perception registers are Name, Organizer, Location, Date, Day, Time, Speaker, Title-of-Talk, Periodicity, and Message Type. Other perception registers hold previously unencountered words and phrases that occur in the received email messages and that might be relevant. The perception process is complete when the type of the received message has been inferred and the understood information regarding the received message has been transferred to the perception registers.

4 Bringing the Focus to Consciousness

Many of CMattie's components use information from the focus. This section describes how the focus is used to bring perceived information into consciousness. As shown in Figure 3, the focus includes five vectors: the perception registers, the output of sparse distributed memory, the input to sparse distributed memory, the output of case-based memory, and the input to case-based memory. The perception module places the components of the understood email message into the perception registers. That constitutes the current percept. Next, sparse distributed memory is read with the current percept as the address. Also, case-based

memory is read with the same address. These reads are designed to gather the past information most relevant to what was just perceived.

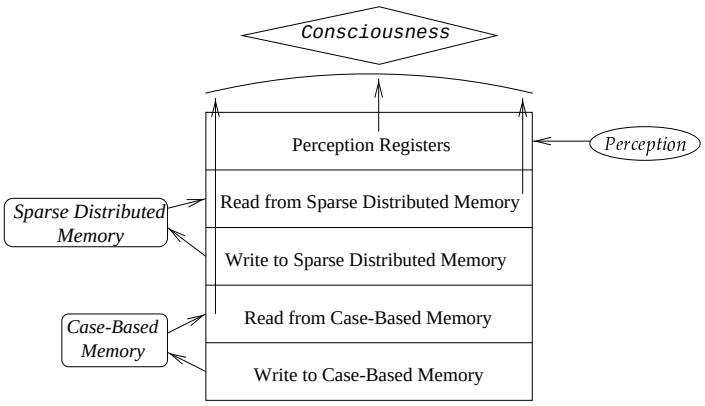

Fig. 3. CMattie's Focus

A Consciousness Codelet is one whose function is to bring specific information to consciousness [4]. In this case, after the memory reads, perceptual consciousness codelets bring information from the focus to consciousness. One such consciousness codelet is associated with each of the perception registers and carries the specific piece of perceived information from that register. For example, one codelet carries the speaker's name, and another carries the seminar's time. In addition, some consciousness codelets check for conflicts amongst the relevant items returned from the percept and the memory reads. For example, a conflict occurs if the perceived place, time and date for the Cognitive Science Seminar are the same as case-based memory's output of these same features for the Graph Theory Seminar. The consciousness codelet recognizing the conflict raises its activation level in an attempt to reach consciousness. When the spotlight shines on this and the other perceptual consciousness codelets, the perceptual information along with the conflict is broadcast to the entire system, that is, to every codelet.

5 Consciousness

CMattie's global workspace gives the agent several important performance features. It allows for coalitions of codelets to gain attention. Information about these codelets is broadcast to all of the agent's other codelets. Recipients of this broadcast become active themselves if enough of the information is understood, and if it is applicable. In this way, the broadcast recipients have the potential to contribute towards solving the problem raised by the conscious coalition. Learning also uses the information in consciousness to learn to associate codelets as a coalition. In addition to consciousness codelets, the consciousness module consists of four major components: the Playing Field, Coalition Manager, Spotlight Controller, and Broadcast Manager.

Playing Field. *Artificial Minds* [6] contains a detailed summary of pandemonium theory first described by Oliver Selfridge in 1959 for perceptual uses and extended by John Jackson in 1987 to an "idea for a mind" [9]. Pandemonium theory's components interact like people in a sports arena. Both the fans and players are known as demons. Demons can cause external actions, they can act on other internal demons, and they are involved in perception. The vast majority of demons are the audience in the stands. There are a small number of demons on the playing field. These demons are attempting to excite the fans. Audience members respond in varying degrees to these attempts to excite them, with the more excited fans yelling louder. The loudest fan goes down to the playing field and joins the players, often causing one of the players to return to the stands. The louder fans are those who are most closely linked to the players. There are initial links in the system. Links are created and strengthened by the amount of time demons spend together on the playing field and by the system's overall motivational level at the time.

CMattie uses pandemonium theory's notion of a playing field. A collection of codelets which act as demons are instantiated when the program first runs. Each of these is a generator codelet of a specific codelet type such as a Slipnet codelet. Generator codelets themselves cannot become conscious. Instead, they instantiate copies of themselves containing the appropriate information received from a relevant conscious broadcast. This allows for multiple codelets of the same codelet type to run in parallel, each working with different information. The generator codelets can be considered fans in pandemonium theory's arena.

Codelets, other than generator codelets, which are performing their respective functions are pandemonium theory's players on the playing field. The playing field is a shared space in memory; all active codelets exist in this shared memory space. Codelets on the playing field may be associated with one another. For example, codelets underlying the same higher level concept, such as a Slipnet node, are likely to be associated with one another. Codelet associations also develop and are strengthened when codelets are together in consciousness. This illustrates one point of difference with pandemonium theory. There, association arises or is strengthened from being together in the playing field. Here, it is from being in consciousness together.

Codelets have a two-part name. The first portion signifies from where a codelet on the playing field is derived, such as a particular behavior. Since there can be multiple codelets of the same type active, codelets also carry a unique identification number. Codelets on the playing field have an activation level, which may come from the higher level construct from which they were instantiated, for example from a behavior, a slipnet node or an emotion. Consciousness codelets provide their own activation. Activation normally decays over time. The activation level of codelets is an important factor in deciding which coalition gains conscious attention.

Figure 4 illustrates CMattie's playing field. Two components of her global workspace implementation, the coalition manager and the spotlight controller, play important roles on the playing field.

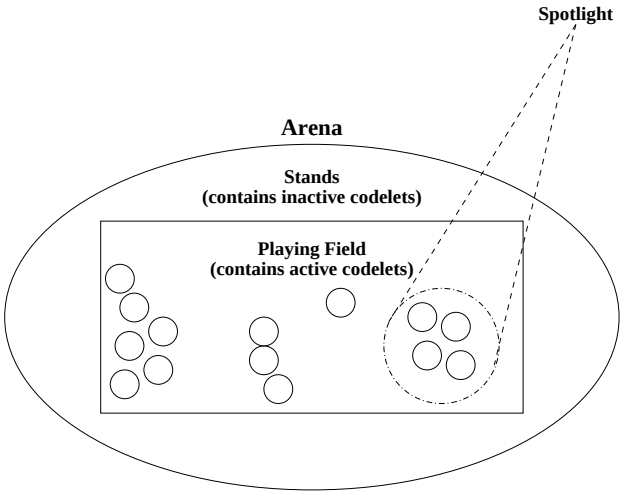

Fig. 4. CMattie's Playing Field

Coalition Manager. The coalition manager groups active codelets into coalitions, and keeps track of them. To make coalitions, the coalition manager groups codelets according to the strength of the associations between them. Only if a collection of codelets is associated above a certain threshold level are they considered to be in a coalition. In particular, the collection of codelets associated with a single higher level concept may or may not form a coalition. As a codelet can serve more than one higher level concept, so may it belong to more than one coalition.

The playing field provides an active dynamic environment. The activation level of codelets decay. Newly activated codelets join existing coalitions. Codelets leave one coalition and possibly join another. The strength of a codelet's association to its higher level concept may change. Due to this dynamic environment, the coalition manager must continually and efficiently survey the playing field to keep its record of coalitions up to date.

Spotlight Controller. The spotlight controller determines which coalition becomes conscious. It calculates the average activation level of each of the coalitions by averaging the activation levels of the coalition's codelets. The spotlight shines on the coalition with the highest average activation level. Average activation among a coalition's codelets, not the total activation, is taken to prevent larger coalitions from having an advantage over smaller ones. In the same way as the coalition manager's, the spotlight controller's domain is extremely dynamic. For example, activation level falls when codelets' work is complete. Also, codelets increase their activation while attempting to come to consciousness.

Broadcast Manager. Once the spotlight controller has determined a conscious coalition, it notifies the broadcast manager who is responsible for gathering information from the conscious coalition, and sending it to all of the codelets. As in global workspace theory, messages are small and understood by only some of the codelets. Specifically, from the conscious coalition the broadcast manager gathers objects labeled for broadcast, that is, those that contain information needed for specifying the current novelty or problem. This information is then broadcast to all of CMattie's codelets. This is accomplished via a blackboard model. Information gathered from a coalition is placed on the blackboard, implemented as a shared memory space. When each codelet polls this blackboard, it searches for parameters it understands. Information is left on the blackboard for a significant time period so that all codelets receive the broadcast as specified in global workspace theory.

6 Conscious Conceptual Learning

The conscious broadcast recruits codelets that understand the message and for which it is relevant. This causes their activation to increase, motivating them to begin performing their respective tasks. These tasks might include activating their overlying higher construct, say a behavior, an emotion, a slipnet node, a learning mechanism. Figure 5 illustrates the significant role of consciousness in perceptual learning, highlighting global workspace theory's premise that consciousness is sufficient for learning. This section focuses on the perceptual learning that results from the conscious broadcasts. CMattie has a limited number of seminars

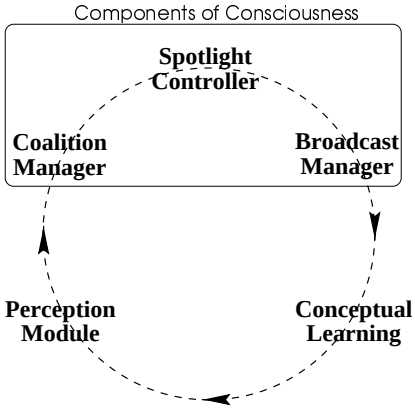

Fig. 5. Consciousness As A Facilitator For Conceptual Learning

already defined in her slipnet. She "knows" about these seminars through the built-in **seminar** concept and its features. In particular, she knows that:

- A seminar is held once a week
- It has an organizer and a name
- Each week, there might be a different speaker
- It has a different title-of-talk

– It is usually held at the same location, on the same day of the week, and at the same time.

Suppose a seminar organizer sends her a message announcing a seminar with a seminar name that she has never seen before. CMattie attempts to treat such a message in a way similar to seminars that she already knows. The learning mechanism described here is based on the premise that any agent, including humans, learns based on what it already knows. When the message understanding mechanism attempts to understand this message, the agent recognizes that it is an **initiate-seminar-message** for a seminar, but that the **name** of this seminar is not part of the built-in knowledge. This information is placed into the perception registers, brought to consciousness, and broadcast. CMattie has codelets that understand this broadcast and can activate different behaviors that act to converse with the sender of the message to determine if the sender wishes to initialize a new seminar. She sends an acknowledgement to the sender stating that a new seminar with that seminar name will be initialized, with the sender as its organizer and requesting confirmation. Reinforcement of a sort is provided to CMattie by the response she might or might not get. Based on the conscious broadcast of this feedback, if any, a new slipnet node is created for this seminar name, and it is linked to the **name** node, which is also a feature of the **seminar** concept. When this new node is generated, the associated codelets for it are generated as well, a quite straightforward process. The new codelets are based on similar, existing codelets for the other name nodes. Once the process is complete, CMattie has understood the incoming confirmation message, and the perception module sends the relevant fields to the perception registers.

The second type of learning that takes place in the perception module occurs when CMattie learns concepts which are not completely identical to the built-in **seminar** concept, but slightly different from it. In her domain, colloquia, dissertation defenses, dissertation committee meetings, and faculty meetings, all fall into this category. This second learning mechanism is based on viewing every new situation in terms of a previously solved problem (analogy making). When CMattie receives a message about such a non-seminar event, say a dissertation defense, she treats it as a **speaker-topic message** for a seminar. This understanding is disseminated through consciousness. The agent sends an acknowledgement to the sender stating that she is initializing a new seminar by the name "Dissertation Defense Seminar" with the sender as organizer. This misunderstanding can be expected to result in events such as:

– The acknowledgement elicits a negative response from the sender, starting an episode. The resulting "conversation" between CMattie and the sender is stored in case-based memory. This episode provides information that allows CMattie, even with her limited natural language understanding, to learn that a dissertation defense is similar to the seminar concept, but with slightly different features. In this case, the **periodicity** feature has a different value. CMattie learns this through case-based reasoning.
– CMattie includes the Dissertation Defense Seminar in seminar announcement. This action is likely to elicit a negative response from the sender, starting an interaction with CMattie. This episode again is stored in case-based memory to aid her in learning what a Dissertation Defense is.

Due to such interactions with seminar organizers and case-based reasoning, CMattie learns a new concept called dissertation defense that is closely related to the seminar concept. Note that this conceptual learning takes place through the internal interaction between consciousness and perception. All such conversations become conscious before any changes to the perception module occur.

CMattie has codelets in her slipnet that look for words and phrases that she hasn't previously encountered. The perception module tracks such new words and phrases that occur with any regularity by keeping statistics and recognizing novelty. This aids in CMattie's natural language understanding. Even with her limited natural language understanding, CMattie can understand messages from organizers that have negative connotations. Her slipnet has nodes and codelets that detect words and phrases with such negative connotations.

Consider a possible path CMattie might take during her conceptual learning. CMattie misunderstands the first dissertation defense message, and sends an acknowledgement to the organizer for a Dissertation Defense Seminar. Suppose the organizer responds with a negative message saying, "It is not a seminar, but a dissertation defense". CMattie understands the negative connotation in "not a seminar" and the repeated occurrence of the phrase "dissertation defense" activates her questioning capability to send a message to the organizer with the question, "What is a dissertation defense?" The organizer might reply with a simple

explanation such as, "A dissertation defense is like a seminar, but it might not occur regularly" or "Dissertation defenses do not ordinarily occur every week". CMattie understands the negative connotations in relation to the words "regularly," "every week" or "weekly". These are keywords in the slipnet related to the periodicity feature of the seminar concept. CMattie uses her case-based memory and natural language understanding to reason that a dissertation defense has a periodicity different from that of a seminar. This interaction with the organizer and her reasoning effects the creation of a new concept, dissertation defense, in the slipnet with related codelets that search for it in future messages. This conscious learning enables her to correctly perceive and understand a dissertation defense message when she encounters one next time. CMattie must also learn to behave differently when faced with a dissertation defense message than when she does with a seminar message.

A trace of this learning stored in her case-based memory serves to enhance her case-based reasoning capabilities. Later, CMattie might encounter a colloquium message, and in response to her incorrect acknowledgement of a Colloquium Seminar, be told that "It is a colloquium, not a Colloquium Seminar". CMattie's case based reasoning depends on (a) past experiences she has had, and (b) her ability to understand new situations in terms of her past experiences. She recalls her experience with the first dissertation defense message from her case-based memory, and reasons that colloquium might be similar to dissertation defense. She sends a message to the organizer with the question, "Is a colloquium similar to a dissertation defense?" Her understanding and reasoning, based on the organizer's reply to her query, aid her in learning about colloquia.

7 Conclusions

This paper presents CMattie's perception, consciousness, and their unique integration fostering conceptual learning. Perceptual output enters the focus, which is brought to consciousness. The global workspace broadcast allows for conscious conceptual learning, completing the cycle. CMattie's design and development have been ongoing for two years. Her predecessor, VMattie, performed successfully with a subset of CMattie's functions in the same domain[15]. As of this writing, the CMattie is in the implementation stage. In future stages, extensions such as unlearning are planned.

References

1. James Allen. *Natural Language Understanding.* The Benjamin/Cummings Publishing Company, Inc., Redwood City, CA, 1995.
2. Bernard Baars. *A Cognitive Theory of Consciousness.* Cambridge University Press, New York, 1988.
3. Bernard Baars. *In The Theater Of Consciousness.* Oxford University Press, New York, 1997.
4. Myles Bogner. Creating a conscious agent. Master's thesis, The University of Memphis, May 1998.
5. Myles Bogner, Uma Ramamurthy, and Stan Franklin. Consciousness and conceptual learning in a socially situated agent. In Kerstin Dautenhahn, editor, *Human Cognition and Social Agent Technology*, Forthcoming.
6. Stan Franklin. *Artificial Minds.* The MIT Press, 1995.
7. Stan Franklin and Art Graesser. Is it an agent, or just a program?: A taxonomy for autonomous agents. In *Intelligent Agents III*, pages 21–35. Springer-Verlag, 1997.
8. Douglas Hofstadter and Melanie Mitchell. The copycat project: A model of mental fluidity and analogy-making. In K.J. Holyoak and J.A. Barnden, editors, *Advances In Connectionist And Neural Computation Theory*, volume 2: Analogical Connections, Norwood, NJ, 1994. Ablex.
9. John V. Jackson. Idea for a mind. *SIGGART Newsletter*, 101:23–26, July 1987.
10. Pentti Kanerva. *Sparse Distributed Memory.* MIT Press, Cambridge, MA, 1988.
11. Janet Kolodner. *Case-Based Reasoning.* Morgan Kaufmann Publishers, 1993.
12. Pattie Maes. How to do the right thing. *Connection Science*, 1990.
13. Melanie Mitchell. *Analogy-Making As Perception.* MIT Press, 1993.
14. Uma Ramamurthy, Stan Franklin, and Aregahegn Negatu. Learning concepts in software agents. In *Fifth International Conference of the Society for Adaptive Behavior-SAB98*, Zurich, Switzerland, 1998. MIT Press.
15. Hongjung Song. *A Control Structure For Autonomous Agents.* PhD thesis, The University of Memphis, May 1998.