

University of Memphis

## University of Memphis Digital Commons

---

CCRG Papers

Cognitive Computing Research Group

---

1999

### Behavioral learning for adaptive software agents

A. Negatu

S. Franklin

Follow this and additional works at: [https://digitalcommons.memphis.edu/ccrg\\_papers](https://digitalcommons.memphis.edu/ccrg_papers)

---

#### Recommended Citation

Negatu, A., & Franklin, S. (1999). Behavioral learning for adaptive software agents. Retrieved from [https://digitalcommons.memphis.edu/ccrg\\_papers/146](https://digitalcommons.memphis.edu/ccrg_papers/146)

This Document is brought to you for free and open access by the Cognitive Computing Research Group at University of Memphis Digital Commons. It has been accepted for inclusion in CCRG Papers by an authorized administrator of University of Memphis Digital Commons. For more information, please contact [khggerty@memphis.edu](mailto:khggerty@memphis.edu).

# Behavioral Learning for Adaptive Software Agent

Aregahegn S. Negatu & Stan Franklin

Conscious Software Research Group

Institute for Intelligent Systems

The University of Memphis

Memphis, TN 38152

Tel: (901)678-2482, Fax: (901)678-2480

E-mail: {asnegatu, stan.franklin}@memphis.edu

## Abstract

In this paper we describe an adaptable action selection mechanism for a software agent called CMattie which implements the global workspace theory of consciousness [Baars88, Baars97]. CMattie gathers seminar information by corresponding with humans, maintains a mailing list, composes a weekly announcement of seminars and mails it to that mailing list. Her interaction with humans is via e-mail using natural language, and is restricted to a narrow domain. Moreover, the agent learns to deal with variations in her domain by conversing with human seminar organizers. Such a variation in the domain might be the introduction of the concept “colloquia,” which differs from the known concept “seminar.” CMattie’s behavioral learning process has two phases: first, she must learn to conceptualize the domain, and second, she must learn to produce actions appropriate to the newly acquired concept. The first type of learning happens in the perception module of CMattie [Ramamurthy98a, Ramamurthy98b]. Here we focus on the second type of learning, the learning of appropriate new behaviors. CMattie’s action selection mechanism is based on an extension of Maes’ behavior network [Franklin95, Maes90]. The behavior net is composed of behavior streams whose execution satisfies a single drive of the agent. It is these behavior streams that must be learned. Thinking of behaviors as operators we can view a behavior stream as a partially ordered plan. Learning a new behavior stream is then a problem of learning a partial order plan. For this we use case-based planners [Monoz-Avila98, Veloso94b] which support learning.

**Key words:** Learning, Planning, Action Selection, Case-based reasoning, Consciousness

## 1. Introduction

An agent perceives what is going on around it to have some idea of the state of its world and then uses its knowledge to produce appropriate action(s) to attain its goals.

Our software agent, CMattie, as a clerical agent, announces weekly seminars in an educational environment. She gathers information about the seminars, keeps a mailing list, remembers facts about seminars, composes a weekly seminar schedule, and emails this schedule to her list. Within her domain, she communicates using a natural language.

For an agent, which “lives” in a relatively complex environment, it is difficult for its designer to predict and prepare for all the various situations the agent will encounter. Thus, an agent’s ability to adapt to changes in its environment, and to improve its performance through experience is a significant factor. In CMattie’s environment, novel situations can occur due to the introduction of new concepts such as colloquia that differ from seminars.

CMattie learns new concepts like “colloquium” based on her existing knowledge of the concept “seminar” [15,16]. Once a new concept is learned, CMattie must then learn to produce the appropriate actions relative to that concept, i.e. to adapt her action-selection mechanism to attain goals in the modified domain.

CMattie is a cognitive agent [5] **NEW REFERENCE ADDED HERE** whose architecture implements the global workspace (GW) theory of consciousness [Baars]. GW theory is a psychological theory of consciousness, that provides a functional account and a high-level architecture. It postulates a global workspace (where conscious events reside) in a distributed system of processors, and the importance of contexts. *Processors* can be viewed as small autonomous, unconscious systems that are limited to one particular function. Processors typically have little or no direct communication among themselves. The *global workspace* is a shared memory area occupied by a coalition of unconscious processes whose contents are broadcast to all other processors. The broadcast from the global workspace effect the recruitment of unconscious processors to help deal with the current situation. The

global workspace has a limited capacity, and coalitions of processors compete for access to it. A broadcast message recruits appropriate unconscious processors to contribute to the handling of a novel situation or to the solution of a difficult problem. At any given time, consciousness deals with a single event, which is internally consistent. *Contexts* are coalition of processors that organize themselves as needed to influence and shape understanding. Though they consist of unconscious processors, contexts (goal contexts, dominant-goal contexts, perceptual contexts, etc.) constrain conscious content.

Behaviors executing in CMattie's action-selection mechanism correspond to dominant-goal contexts. A goal can be thought of as a representation of a future state that helps to select subgoals and/or actions to reach to the intended state. To achieve the state of a given goal, its corresponding behavior produces a collection of actions to reach at the state of the behavior.

CMattie's action selection module is based on the Maes' behavior network [11], which is a collection of behaviors connected by directed links, and includes a fixed set of global goals (drives) and an environmental interface. Each behavior by itself can be viewed as a production rule with preconditions and actions (delete & add lists) together with an activation level. The different links are constructed based on the shared conditions of behaviors in the precondition, delete and add lists. Activation spreads among behaviors via the links. The initial sources of the activation are the goals and relevant environmental inputs. There are global parameters to adjust the characteristics of the system as an action-selection mechanism by controlling the dynamics of the spreading of activation.

Maes' behavior network is modified in CMattie in an important way to handle variables [17], and as a result it is possible to have instantiation of behavior streams in CMattie's action-selection mechanism. Each behavior stream is a collection of behaviors that will serve to satisfy a single drive of the agent. The whole behavior network comprises a number of behavior streams and the relationship between drives of the agent and streams is one-to-many since a drive can have more than one stream that satisfies it.

CMattie uses e-mail communication with humans; for the acquisition of new domain knowledge and for evaluation of her performance. In CMattie's learning cycle, she interacts with seminar organizers to acquire necessary information or to get feedback on her performance.

In this paper we will describe the behavioral learning mechanism that allows the learning of a new behavioral stream (as a partial-order plan of action) using current knowledge. The behavioral learning module should be capable of fast learning (effective use of prior knowledge and of interacting with organizers during learning.

In section 2, we will elaborate on the need for behavioral learning in CMattie's domain, and give brief description of the components of the learning module. In section 3, we describe the learning process in action.

## 2. The Need for Behavioral Learning

CMattie's drives operate continuously. Her behaviors act in the service of these drives. CMattie wants to announce the weekly seminars in a timely fashion, to maintain the latest information about each of the ongoing seminars, to keep her mailing list updated, to acknowledge each incoming message, to keep herself safe (self-preservation). Drives influence the selection of behaviors by passing proportional to the urgency of the drive. A behavior stream is a subset of the behavior network whose execution serves to satisfy one of the drives.

When CMattie receives an email message, her perception module understands the message and posts the resulting percept on a special table called the focus. The focus contains both the percept and relevant information from associative and episodic memories. That is, the focus holds the percept in context as CMattie currently understands it. The contents of the focus come to "consciousness" [Bogner, et al][Ramamurthy98a] when the "consciousness" mechanism broadcasts it to all the unconscious processes, called codelets. Those codelets that find themselves relevant to the broadcast will collectively serve as an index to pick an appropriate behavior stream to be instantiated as an addition to the behavior net. With the rich contextual information of the percept in the focus, CMattie has enough knowledge to pick or index the most relevant behavior stream for instantiation. Each instantiated behavior stream will eventually execute its behaviors thereby satisfying its associated drive. Behaviors in a behavior stream execute in a partial order. We can view a behavior stream as a partial order plan which guides behaviors (plan operators) from the initial state (resulting from the broadcast) to the goal state that satisfies a particular drive.

CMattie's behavior network has built-in behavior streams that produce appropriate actions if the problem description as it is presented in the focus does not constitute a novel situation. In dealing with seminars, CMattie has built-in domain knowledge and operators. She can do a number of things such as (a) compose all weekly seminars and announce them together, (b) change seminar information and if necessary re-announce the updated information, and (c) remind seminar organizers to send weekly seminar information.

When CMattie learns a new concept like "colloquium" [Ramamurthy98a, Ramamurthy98b], she

learns the characteristics that distinguish the new concept “colloquium” from the known concept “seminar.” CMattie must also learn to produce appropriate responses to messages concerning colloquia. The feature “periodicity” is one characteristic that distinguishes seminars from colloquia. In the department CMattie serves, seminars are held weekly while colloquia are held irregularly. CMattie must learn that there is no need to send reminders to organizers about colloquia. There are other differences also. For example, she should learn to announce colloquia as the proper number of days in advance, to announce colloquia separately from seminars, to include colloquia in the seminar announcements also, and to include the abstract of a colloquium in the seminar announcement when available. Other items also need to be adapted for the colloquia concept. CMattie acknowledges every received message. The message composition template for this acknowledgement needs to be adapted to accommodate the needs of this expanded domain. All this illustrates how CMattie needs a mechanism to acquire domain knowledge and learn new behavior streams (action plans) to produce correct responses to such changes in her domain.

Figure 1 shows the components of CMattie’s behavioral learning module. This module is designed on two premises. (a) CMattie should use past experience to learn new plans or behavior streams by adapting old plans that worked in similar situations. By doing so a new solution can be found relatively quickly. (b) She must carry on conversations with humans to acquire new domain knowledge. This also allows her to get feedback on the accuracy of new plans and revise them if

necessary. We go on to describe each of the pieces of this learning module.

**Behavior Network (BN) System:** CMattie’s action selection module is implemented as a behavior net. The original behavior network system [Franklin95, Maes90] has been enhanced in several different ways [Song98], for example, allowing it to support variables. This module is a critical component in CMattie’s overall architecture (see Bogner, Ramamurthy & Franklin 1999). The behavior network works along with the perception module, the different memory mechanisms, and several processes and coalition of processes which are not shown in Figure 1. As it is pointed out above, a behavior stream is a connected partially ordered set of behaviors (plan operators) that serve to satisfy a single drive. A stream is typically instantiated as a result of the receipt of a message. Several behavior streams can be instantiated at the same time, and all run in parallel. Activation originating with the drives and from the environment (incoming messages) is passed between behaviors. At any given time CMattie is controlled by a single behavior which could come from any one of the instantiated behavior streams. This executing behavior is the one with the highest activation among those whose preconditions are satisfied.

**KB:** This knowledge base contains all domain related knowledge including the behaviors (plan operators). When new facts about the domain are acquired via conversations with humans, they are stored in the KB. The information in the KB is needed for the learning/adaptation process.

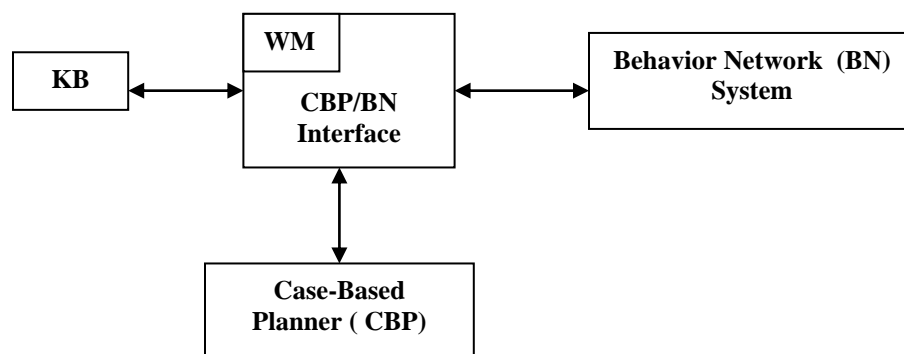


Figure 1. CMattie’s Behavioral Learning Module

**Case-Base Planner (CBP):** The case-based planner is a case-based reasoning (CBR) system [Kolodner93] that can generate plans. In our case, the CBP must have a flexible plan learning/adaptation mechanism. In general a CBR system is a paradigm that solves new problems by adapting solutions to old problems, i.e., using past experience. There are case-based planning systems that can be used in CMattie’s domain [Monoz-

Avila98, Veloso94b]. In general, a CBR system supports the following main processes: retrieval, adaptation, and retention. *Retrieval:* Given a problem description, select a set of stored cases which have problem descriptions similar to the given problem description. Indexing of cases is necessary for efficient retrieval. *Adaptation:* The retrieved cases are adapted, as needed, in an attempt to solve the new problem. The outcome of the new solution

must be evaluated. If it is not acceptable, the solution must be revised. A good feedback mechanism is necessary for proper evaluation and consequently to improve the retrieval precision and the adaptation control strategy. *Retention*: As part of accumulating experience, a newly found case (solution) can be stored so that it can be used in the future.

In CMattie's domain, cases are represented as triplets,  $\langle \text{problem description, solution, outcome} \rangle$ . A *problem description* includes the initial state of the problem situation (the contents of the focus, relevant coalition of codelets, and feature values of relevant concepts) and the goals that satisfy a drive (from the relevant behavior stream). A *solution* is a plan (behavior stream) whose execution beginning at the initial state of the problem satisfies its stated drive. An *outcome* is the expected result (feedback from a human) when the solution plan is applied in the initial state.

**CBP/BN-Interface:** This module connects the CBP and BN systems and serves both the behavioral learning and the domain knowledge acquisition processes. It is used to (a) store newly acquired domain knowledge into the KB, (b) compile the problem description (from the BN side) in the format the CBP can use, (c) format a newly obtained plan (from the CBP) so that it can be integrated into the BN system and (d) facilitate effective conversation with human (via the BN) by providing information available in the CBP and/or the KB. The CBP/BN-interface uses its own working memory (WM).

### 3. How Behavioral Learning Happens

When a situation new to CMattie arises, a resulting instantiated behavior stream may fail to produce an acceptable response. This failure to do the right thing starts the learning process. For CMattie, a failure signal is a negative e-mail message from a person in reply to her inappropriate message. CMattie's perception module [Ramamurthy98a] recognizes this negative feedback message and tries to learn from it. As an example, suppose CMattie has learned the colloquium concept. She may send a reminder message to the colloquium organizer, as she does to seminar organizers, saying "Please send me the weekly information for the colloquium". But colloquia occur irregularly, and sending a reminder is not the right thing to do. As long as CMattie does not get a response to her reminder, she may continue to send them until the announcement deadline passes. Now suppose the colloquium organizer sends CMattie a message saying "Do not send reminders for a colloquium." The perception module has enough vocabulary and reasoning capacity to understand this message as feedback to a past action. This activates the behavioral learning module. It stores the newly acquired

domain knowledge "do not send reminders for a colloquium" into her knowledge-base, appropriately represented. It stores a case in the case-memory of the CBP to note the failure of the outcome for "sending reminder to colloquium organizer." And finally, it modifies the behavior stream that sends the "seminar reminder" be consistent with this newly acquired domain knowledge. The knowledge-base update is made permanent after the proper evaluation (via more conversation with human and/or the success of plans that are generated by this new knowledge). Next time the "send reminder to colloquium organizer" action sequence becomes relevant by an indexing coalition of codelets, the CBP will generate a plan that will satisfy the goal "do not send reminders to colloquium organizers". If the outcome of the new plan succeeds, the case of the plan will be stored in the CBP and the plan itself will become a behavior stream serving one of the relevant drives of the agent.

It is possible that one feedback message may not be enough to get the required domain information and produce the correct response (plan). The interaction with the human may continue until the adaptation process succeeds in producing the correct response. CMattie's interaction with a seminar organizer and her sequence of reasoning decisions made to acquire a particular piece of knowledge will be stored in the case-based memory as an episode. The effectiveness of her conversation depends on her past experience and on her ability to comprehend the new situation based on this past experience. So, as CMattie gets more experienced, she can be expected to learn to do the right things faster. Learning speed is a critical feature for an agent like CMattie that gathers information via interaction with humans.

Similarly, domain knowledge acquisition and learning of new behavior streams (plans) is done for the new domain facts like "announce colloquia separately from seminars".

### 4. Conclusion

In this paper we have described an adaptive action-selection mechanism in the domain of a clerical software agent, CMattie. We presented a behavioral learning module that can produce new behavior streams (action plans). Each such stream can execute a series of actions appropriate to a newly conceptualized novel situation. Behavioral learning follows the detection and conceptualization of a change in the domain which takes place in the CMattie's perception module [15,16].

The adaptation process uses available knowledge (both domain and control knowledge) in the behavioral learning module, and feedback obtained from interaction with seminar organizers.

Maes has developed a learnable behavior network system [12] which is more appropriate for robotic learning. But, her mechanism can not be used in CMattie's domain since it does not support delayed feedback during learning (it is a reactive system) and, being reinforcement learning, it would learn too slowly using interactions with humans. CMattie's behavioral learning supports learning via delayed feedback and fast learning is produced by an effective use of prior knowledge using case-based reasoning [10, 14, 18].

At the time of writing of this paper, CMattie is under implementation.

## References

1. Bernard Baars. A Cognitive Theory of Consciousness. Cambridge University Press, New York, 1988
2. Bernard Baars. In The Theater of Consciousness. Oxford University Press, New York, 1997.
3. Bogner, Myles, Ramamurthy, Uma, & Franklin, Stan. (1999). "Consciousness" and conceptual learning in a socially situated agent. Dautenhahn, Kerstin. (Ed.). Human cognition and social agent technology. Amsterdam: John Benjamins Publishing Company.
4. Stan Franklin. Artificial Minds. Cambridge, MA:MIT Press, 1995.
5. Franklin, S. (1997) Autonomous Agents as Embodied AI, *Cybernetics and Systems*. 28: 499-520.
6. Stan Franklin, Art Graesser, Brent Olde, Hongjun Song, and Aregahegn Negatu, "Virtual Mattie: An Intelligent Clerical Agent", AAAI Symposium on Embodied Cognition and Action, Cambridge, MA, November, 1996.
7. Kristian Hammond. "Learning to Anticipate and Avoid Planning Problems through the Explanation of Failures" Proceedings of the Fifth Conference on Artificial Intelligence, AAAI, Philadelphia, 1986.
8. John V. Jackson, "Idea for a Mind", SIGART Newsletter, No. 181, July 1987, p. 23-26.
9. Leslie Pack Kaelbling, Michael L. Littman, Andrew W. Moore, "Reinforcement Learning: A Survey", *Journal of Artificial Intelligence Research* 4 (1996) p. 237-285.
10. Pentti Kanerva, "Sparse Distributed Memory", MIT Press, 1988.
11. Janet Kolodner, "Case-based Reasoning", Morgan Kaufmann Publishers, 1993.
12. Pattie Maes. How to do the right thing. *Connection Science*. 1990.
13. Pattie Maes and Rodney A. Brooks, "Learning to Coordinate Behaviors", Proceedings of the Eighth National Conference on Artificial Intelligence, AAAI Press/MIT Press, 1990, p. 796-802.
14. Melanie Mitchell, "Analogy-Making as Perception", MIT Press, 1993.
15. H. Monoz-Avila. "Integrating twofold Case Retrieval and Complete Decision Replay in CAPlan/CbC", Doctoral dissertation, Dept. of Computer Science, University of Kaiserslautern, 1998.
16. Uma Ramamurthy, Myles Bogner, and Stan Franklin. Conscious Learning In An Adaptive Software Agent. Proceedings of The Second Asia Pacific Conference on Simulated Evolution and Learning (Canberra, Australia - November 1998).
17. Uma Ramamurthy, Stan Franklin, and Aregahegn Negatu. Learning Concepts in Software Agents. From Animals to Animat 5: Proceedings of The Fifth International Conference on Simulation of Adaptive Behavior (Zurich, Switzerland - August 1998).
18. Hongjung Song. "A Control Structure For Autonomous Agents", Doctoral thesis, The University of Memphis, May, 1998.
19. M. Veloso, "Prodigy/Analogy: Analogical Reasoning in General Problem Solving", Topics on Case-Based Reasoning, Springer Verlag, pps. 33-50, 1994.
20. M. Veloso, "Planning and Learning by Analogical Reasoning", Lecture Notes in Artificial Intelligence, SpringerVerlag, 1994.