Proceedings of the IJCAI-03 Workshop on Mixed-Initiative Intelligent Systems, Acapulco, Mexico, August 2003.

Toward a Disciple-based Mixed-Initiative Cognitive Assistant

Gheorghe Tecuci[1,2], Mihai Boicu[1], Dorin Marcu[1,2]
[1] Learning Agents Laboratory, George Mason University
4400 University Dr., Fairfax, VA 22030, USA
[2] Center for Strategic Leadership, US Army War College
Carlisle Barracks, PA 17013, USA
{tecuci, mboicu, dmarcu}@gmu.edu

Abstract

This paper discusses research issues in the development of a mixed-initiative cognitive assistant which is initially trained doctrinal correct problem solving by a subject matter expert and it is afterwards handed over to other users to assist them in problem solving, to learn, and to adapt to them.

1 Introduction

We envision a future where people are assisted by personal cognitive agents that effectively increase not only their recall, speed and accuracy, but even their creativity and flexibility. The user will act as the orchestrator of the problem solving process, guiding the high-level exploration, while the agent will implement this guidance, dealing with the associated details. The resulting mixed-initiative reasoning process will interleave routine and innovative reasoning steps proposed by the assistant with inventive and creative ones initiated by the human. In the same time, the assistant will continuously learn from this joint problem solving experience, and will adapt to its user, becoming a better collaborator that is aware of its user's preferences, biases and assumptions. This mixed-initiative problem solving and learning will require the user and the assistant to share representations, to communicate naturally, to properly divide their tasks and responsibilities, to coordinate their actions, to take initiative and to relinquish control.

For many years we have worked on developing the Disciple multistrategy learning approach to agent teaching by a subject matter expert [Tecuci 1988, 1998; Boicu et al. 2003]. In this approach a subject matter expert teaches his or her problem solving expertise to the agent using examples and explanations, in a way that resembles how a person teaches another person. Disciple is a highly interactive approach where the expert helps the agent to learn and the agent helps the expert to teach it. For this and other reasons Disciple provides a suitable base for developing the new generation of intelligent agents described above.

This paper discusses some of the complex challenges of building and using such a mixed-initiative cognitive assistant, called Disciple-MI. The next section presents the life cycle of the proposed Disciple-MI assistant. Then each of the follow-on section defines a main research problem that needs to be addressed in order to build the proposed mixed-initiative cognitive assistant.

2 The life cycle of the assistant

Figure 1 presents the main phases in the development and use of Disciple-MI. The starting point in Figure 1 is the development of the Disciple-MI learning agent shell and its customization for a particular application domain. This learning agent shell has the required knowledge representation, learning and problem solving capabilities, but no specific domain knowledge in its knowledge base. Its customization for a particular application domain consists primarily in the development of an initial object ontology, usually through import from previously developed knowledge repositories, such as CYC [Lenat, 1995], SUMO [Niles and Pease, 2001], or DARPA Agent Markup Language (DAML) based web ontologies [Hendler and McGuinness, 2000].

In the second phase of the life cycle the agent receives doctrinal training from a subject matter expert who teaches it to perform certain classes of tasks, according to the general doctrine of the discipline. The goal of this phase is to create an agent that can act as competent collaborators of both experts and non-experts.

Copies of the doctrinally trained agent are then provided to other users, to be used as their intelligent assistants. The user may be a subject matter expert who uses the agent as a problem solving and learning assistant. The user may also be a less-experienced person. In such a case the agent will first take the role of an intelligent tutor, teaching the user in a way that is similar to how it was taught by the subject matter expert (see phase 3 in Figure 1). Then the user will use the agent as an expert collaborator, improving his or her decision-making abilities.

During the agent's normal use (represented as phase 4 in Figure 1) the agent and the user will encounter novel situations which are new opportunities for learning.

However, the user will be primarily concerned with the current problem solving process and will have neither the time nor the incentive to support the learning of his/her agent at that time. Moreover, the agent will have to use many of its computational resources to support its user in the current problem solving process. Therefore, the agent will have to learn from this mixed-initiative problem solving process by employing resource bounded, non-disruptive learning techniques, storing relevant experiences for mixed-initiative learning to occur during periodic after-action reviews, represented as phase 5 in Figure 1. During the after-action reviews, the agent will learn not only new reasoning patterns, but also a model of its user, incorporating his/her preferences, biases, and assumptions. This will allow the agent to behave more and more like its user, but also to challenge the user's assumptions and biases when warranted by the current situation. After the completion of the fifth phase, the agent will reenter the fourth phase of its life cycle.

During its use, the agent will acquire a significant amount of problem solving knowledge. However, because its user is not a knowledge engineer, this knowledge will not always be optimally represented and organized. A knowledge engineer could now interact with the agent to optimize its knowledge base, with very limited or no assistance from a subject matter expert. This optimized agent is then returned to its subject matter expert (see the dotted line from phase 6 to phase 5). Notice also that the knowledge bases of different agents will incorporate new collective expertise. A comparative analysis of these knowledge bases (performed during phase 6) will reveal valuable new knowledge to be integrated into an improved doctrinal agent, and therefore injected back into the process. The developed knowledge bases could also be kept to represent the knowledge of various users/experts, allowing the preservation and access to this knowledge. This phase will also include an improvement of the learning agent shell, based on the lessons learned in the previous phases.

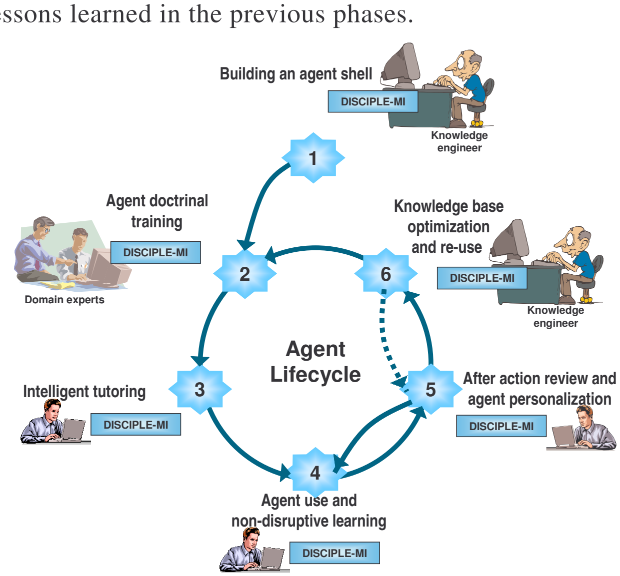

Figure 1: Disciple-MI life cycle

3 Modeling expert's reasoning

As indicated in the previous section, a subject matter expert has to teach Disciple-MI his/her problem solving expertise by showing it how to solve specific problems, and explaining his/her reasoning process. A critical research issue, (that builds on both the cognitive science and the artificial intelligence research on problem solving) is the development of a multistrategy modeling framework that will help the subject matter experts to express their reasoning in a natural way. An appropriate modeling framework should allow the use of natural language, to avoid placing additional burden on an already difficult expert task. On the other hand, the modeling framework should be precise enough to form the basis for hybrid and automated reasoning.

We have long-term experience with the task reduction modeling framework. In this framework a complex problem solving task is successively reduced to simpler tasks, the solutions of the simplest tasks are found, and these solutions are successively combined to synthesize a solution to the initial task. We have used several versions of the task reduction paradigm to model action planning, course of action critiquing, and the identification and testing of strategic center of gravity candidates in major theater of war scenarios [Bowman, 2002; Tecuci and Boicu, 2002]. Experiments performed by us at the Army War College demonstrated that subject matter experts found this type of modeling natural and easy to use [Tecuci et al, 2002]. Therefore, for Disciple-MI we plan to further extend our current modeling approach.

Even with an appropriate modeling framework in natural language that captures a wide variety of reasoning patterns, making explicit their reasoning will remain a challenging task for a subject matter expert who does not have knowledge engineering experience. Therefore a critical research issue is the development of plausible reasoning methods that will support the experts in expressing their reasoning. Some methods will be based on analogical reasoning with similar models developed in the past. Other methods will use knowledge engineering principles to suggest corrections of an initial model formulated by the expert. Initial such methods associated with the task reduction paradigm from the Disciple approach are presented in [Boicu, 2002].

4 Learnable knowledge representation

To facilitate effective mixed-initiative reasoning, the knowledge representation has to be appropriate both for a human (who uses natural language and visual representations), and for an automated agent (which requires a formal representation). Building on the current representation of Disciple [Tecuci *et al.,* 2002], the learnable representation of Disciple-MI will have two main components, an object ontology that defines the terms (the concepts) from an application domain, and a set of problem solving methods expressed with these terms. The object ontology consists of hierarchical descriptions of the types of objects from the application domain, specifying their properties and relationships. The problem solving methods are general representations of the reasoning models used by experts.

The object ontology has a critical role from several points of view. First, it represents the basis for communication with the expert and with other agents. They should all share many of the concepts represented in this ontology. Second, the representation of concepts is used by the agent as a generalization hierarchy for learning. However, any object ontology is only an approximate representation of a complex and dynamic world. For these reasons, we will address the problem of learning within an evolving representation space. This involves not only continuously refining the object ontology, but also reformulating all the previously learned problem solving methods, which are based on the ontology.

In order to satisfy the dual requirement of understandability by the human expert and automatic processing by the agent, the ontological terms and the inference methods are represented at multiple levels of abstraction and formalization, from natural language to formal logic, allowing both the expert and the agent to manipulate its representation, while maintaining their equivalence.

5 Multistrategy teaching and learning

A critical research issue is the development of improved methods by which a subject matter expert can teach Disciple-MI his/her problem solving expertise. The expert will demonstrate to the agent how to solve a specific task and the agent will learn general reasoning methods and will refine its ontology, to be able to solve similar tasks in a similar fashion, as demonstrated by the Disciple family of systems.

A key notion that we have introduced in the past, and plan to generalize and further explore is that of a plausible version space of a concept [Tecuci, 1998; Boicu 2002]. It represents the space of the best approximations of a concept, learned from a limited number of examples and explanations. We plan to develop new incremental learning methods which will rapidly learn and refine such a plausible version space, even if the concept to be learned is not representable in the language of the agent. An analysis of the limits of these approximations will also facilitate the learning of the necessary extensions of the representation language, in the form of new concepts to be introduced in the object ontology.

Initially, when the agent has no reasoning methods or rules, the expert teaches it how to solve problems and the agent learns reasoning methods. As the agent learns from the expert, the interaction between the expert and the agent evolves from a teacher-student interaction, toward an interaction where both collaborate in solving a problem. During this mixed-initiative problem solving process, the agent learns not only from the contributions of the expert, but also from its own successful or unsuccessful problem solving attempts, which are based on the applicability of the partially learned methods/rules.

6 Acquisition of expert's language

The current learning methods of Disciple evolve toward a type of understanding-based learning, which involves a mixed-initiative process of language to knowledge transformation. However, the agent cannot have powerful natural language processing capabilities from the very beginning. Therefore, a critical research issue is the development of methods that will allow the agent to acquire the language of the expert as part of the same process in which it acquires the expert's problem solving knowledge. In this context, partial understanding of the expert's phrases should help the agent in learning object concepts and reasoning methods. On the other hand, the agent's previously learned knowledge should help it in better understanding the expert's current phrases.

We plan to explore mixed-initiative methods for language acquisition, understanding and generation. These methods will be based on the complementary communication capabilities of humans and automated agents in language generation and understanding. In particular, we will use the fact that it is easier for an expert to understand a formal expression than it is to generate one. Conversely, it is easier for an agent to generate a natural language expression than to understand one [Grishman, 1986]. For instance, when the expert expresses some natural language sentence, the agent may need to generate alternative interpretations, asking the expert which of them corresponds best to the input sentence. If the input sentence contains some unknown word, the agent will use the contextual information to guide the expert in defining a new concept or a new relation that corresponds to that word, and appropriately place it in the object ontology. If the agent needs an answer to some question, it may formulate the question and may hypothesize possible answers, asking the expert to select the right one. If the agent cannot generate an answer, then the expert may provide some helpful hints to generate it.

This type of communication is a generalization and extension of the types of interactions used by the current version of Disciple [Tecuci et al., 2002]. Using its learning capabilities, the agent will continuously learn the language of the expert. In particular, the agent will acquire lexical knowledge related to the ontology that characterizes a domain, as well as syntactic-semantic knowledge associated with rules specific to the various applications within the domain. The learning process will judiciously incorporate the constraints on subject matter, dialogue structure, vocabulary and syntax, to improve the effectiveness of the language learning process. One idea is to develop incremental understanding methods, by analogy with incremental learning [Mitchell, 1997]. This means that the agent will maintain representations of partially understood phrases, and will improve its understanding as it encounters similar phrases, or as it improves the learning of the phrase's constituents. Therefore, the developed knowledge representation will have to allow for the representation of general language knowledge (such as lexicon and grammar rules), as well as the representation of partially understood phrases, both as part of the object ontology, and as part of the reasoning methods. In addition, as mentioned in section 4, each knowledge piece from the knowledge base will have both an external informal representation and an equivalent internal formal representation. Therefore the expert can easily interact and update the external form, and the agent will accordingly update the internal form.

7 Mixed-initiative problem solving

Mixed-initiative problem solving and learning are intimately interleaved. Previously learned knowledge will allow the agent to contribute to the mixed-initiative problem solving process. On the other hand, the human's contribution to this process is a learning opportunity for the agent. An important research issue is the development of methods that synergistically integrate human's experience, flexibility and creativity, with the agent's speed, recall, accuracy and consistency. In essence, the expert should act as the orchestrator of the problem solving process, guiding the high-level exploration, while the agent should implement this guidance, dealing with the associated details. The resulting mixed-initiative reasoning process will interleave routine and innovative reasoning steps proposed by the agent, with inventive and creative ones initiated by the expert, based on the ability of the agent to reason not only deductively, but also analogically, abductively, inductively, and abstractly.

Let us assume that the user needs to perform some reasoning task. The agent may immediately generate the corresponding reasoning tree, by applying its reasoning rules, rapidly providing a solution and its justification. There will be, however, situations where the learned rules will not be applicable. In such a case we would still like the agent to propose a plausible solution or, at least, to help the expert to find the solution. This is precisely what the agent will do by using partially learned rules in conjunction with plausible reasoning strategies. As mentioned above, the agent will learn rules from single examples and explanations. Each such rule will have two applicability conditions. The plausible lower bound condition is based on a most conservative generalization of the example. Therefore, when this condition will be satisfied, the solution provided by the rule will be expected to be correct. We call such a solution a routine one because it is very similar to a previously encountered solution (namely, the example solution from which the rule was learned). However, the rule also has a plausible upper bound condition. This condition is based on a more aggressive generalization of the example. Therefore, when it will be satisfied, the solution provided by the rule may or may not be correct. We call such a solution an innovative solution. In such a case the agent would alert the expert of the uncertainty of the solution. This type of reasoning will lead to a significant increase in the number of problem solving tasks for which the agent will be able to provide (partially correct) solutions and justifications. We would like the agent to be able to contribute two other types of solutions, associated with increased degrees of novelty. We call them inventive solutions and creative solutions. An inventive solution will be obtained from the solutions of several rules, which have the conditions only partially satisfied. Such an inventive solution will generally be composed through an interactive process with the subject matter expert. Finally, a creative solution can only be composed in collaboration with the subject matter expert. The difference between an inventive solution and a creative solution is that the creative solution involves using elements (such as new concepts or features) that are not yet represented in the knowledge base of the agent. In this way the agent can help the expert not only when confronted with familiar situations, but also when confronted with novel and unfamiliar situations. Notice, however, that once an appropriate inventive or creative solution was found, a new rule will be learned from it, and this type of situation will no longer be unfamiliar to the agent. Thus the agent will continuously expand its knowledge, and will become a more useful assistant to the human user.

8 Resource-bounded learning

During mixed-initiative problem solving a user will encounter novel and unfamiliar situations, each being an important opportunity for the agent to learn. However, the user will be primarily concerned with the current critical problem solving process and will have neither the time nor the incentive to support the learning of his/her agent. A critical research issue is how to take advantage of these rich learning opportunities in the context of the limited availability of the user and of the computational resources. To address it, we plan to develop non-disruptive knowledge acquisition and learning methods that will involve a combination of autonomous learning with selective memorization for mixed-initiative learning to take place during after-action-review periods.

During time-critical problem solving, the agent may use autonomous learning techniques (such as empirical induction from examples), when they do not require the help from the expert. However, when faced with a complex learning problem that requires interaction with the user, the agent will simply record all the relevant information to be able to resume the learning during the after action review process. For instance, the agent may simply keep as negative exception a problem solving step that it attempted but was rejected by the user, and the agent does not understand the reason. Similarly, it may record the novel way in which the user composed some pieces of evidence, to learn a general method latter.

During periodic after action reviews, the agent will learn from previously encountered situations, as discussed in section 5. One of the issues that the user and the agent would need to deal with is the accumulation of exceptions of the rules. These exceptions are a result of the incompleteness of the agent's representation language, which cannot distinguish between some correct problem solving episodes and incorrect ones. Building on our previous research [Boicu et al., 2003], we plan to develop advanced exception-based methods for extending the object ontology of the agent to eliminate the exceptions. Notice however that any change in the ontology will require an adaptation of the previously learned knowledge pieces because the ontology is the agent's generalization hierarchy for learning. To optimize this costly process, we plan to develop methods for rapid relearning in the context of the updated ontology. These methods will be based on maintaining the relevant information from which an individual knowledge piece was learned, such as generalized explanations and prototypical examples.

9 Learning a model of the expert

Effective mixed-initiative inferencing requires the agent to exhibit different behaviors. On one hand, it has to be able to act as a seamless extension of the user, solving the problem in the same way the user would do it (only faster and more accurately, without being affected by stress or fatigue). On the other hand, the agent should be aware of the user's assumptions and biases, checking and challenging them, to minimize the chances of error. Therefore, a critical research issue is the development of methods for learning and continuously updating a model of the user, incorporating his/her factual and tacit problem solving knowledge, assumptions, preferences, and biases.

The user model will be built incrementally and interactively, as a by-product of mixed-initiative problem solving and learning. The agent will record the user's assumptions and preferences, which are elicited from the user, or are inferred from his/her behavior. The user, on the other hand, should be able to examine his/her model and to refine and modify it if necessary. One promising way to model the user's problem solving preferences and biases is to learn meta-rules that express user's preferences among competing applicable rules.

10 User-information interaction

A challenge of mixed-initiative reasoning consists in the quantity and diversity of information that the human has to interact with, and the significant number of different tasks that the human and the agent have to perform. In order to organize, control, and facilitate these tasks and their associated information, new interaction methods are needed. These methods should allow the human to collaborate with the agent in solving complex tasks, making explicit their assumptions, hypotheses, and the thread of logic, and exploring competing hypotheses and scenarios in parallel. They will include the capability to manipulate graphical structures that will allow the human both to have a global picture of the status of the current tasks, and a detailed view on a specific aspect, and to switch very easily between them.

For instance, a global view would show the entire problem solving tree at a high level of abstraction, easily identifying what high level tasks have been solved, and what is the status of the others, what are the assumptions made, and to what degree they have been challenged, which tasks are too difficult for the agent and require user's assistance, etc. The user could then choose the next item to concentrate on. For instance, he/she may decide to focus on an assumption challenged by the agent. Then a new working space will be created, containing a more detailed view of how the assumption was challenged and which is the evidence against it. An approach to facilitate this kind of interaction is to develop methods to assign priorities to the tasks that are currently in the solving process, taking into account those tasks' different starting points, their significance in comparison with the other parallel tasks, and their completion deadlines. For each such task one would maintain an agenda with the operations that need to be performed and a history of the operations already executed, to help the user in tracking the status of the task after he/she returns from the execution of another task.

These methods will be supported by research in mixed-initiative interaction and control, user task agendas, user modeling, and intelligent user interfaces.

11 User-agent interaction

Effective mixed-initiative problem solving and learning activities, in which both the agent and the human contribute what they do best at the appropriate moment, requires evolvable interaction and user models.

The interaction model describes the tasks that both the user and the agent may perform in different contexts, how these tasks may succeed one after another, what are the conditions for a task to become performable and what are the consequences resulting from it being performed. The model of these tasks may be a hierarchical model similar to GOMS [John and Kieras, 1996], with two dimensions: a generalization hierarchy in which tasks are grouped based on their type (for example, defining the example of a reasoning step informally and formalizing an informally defined example of a reasoning step are both teaching tasks), and a decomposition hierarchy in which each general task is decomposed in more specific tasks that need to be performed in order to complete the general task until elementary tasks are reached that can be performed directly by the user or by the agent (for instance, the task to define an informal example of a reasoning step in Disciple can be decomposed in the more specific tasks of informally defining the task, question, answer and subtasks). Each elementary task described by the interaction model will be associated from the very beginning with all the participants that can perform it (i.e. the user and/or the agent), this association being refined as the capabilities and the actual performances of the participants evolve and are continuously evaluated.

The mixed-initiative interaction framework will provide the agent direct access to the real time trace of the performed tasks through the inference engine that will maintain that trace, while helping the user to visualize and interact with it through a dedicated tool.

Many different tasks can take place at any given time in a mixed-initiative interaction, and while the actions performed by the agent can easily and precisely be disclosed to the user, the actions performed by the user may not always be predictable or clear to the agent. Therefore, the agent will require advanced methods to identify the intentions of the user and place them in the general context of the interaction model.

The user model should include information relevant to the mixed-initiative interaction process. Such information refers to the capability and effectiveness of the user in performing different tasks, therefore the user model and the interaction model will be closely related to one another and will evolve as such. The user model should also include the preferences of the user with respect to the inner workings of the initiative switching process (for example, when the user may be interrupted by the agent and when it should not be, whether the interruption should be made in an intrusive way or more gently based on the gravity of the situation or other factors, etc.). This information must also be updated as the interaction between the user and the agent evolves, therefore requiring specific learning and refinement methods.

12 KB optimization and integration

During its use, the agent will acquire a significant amount of problem solving expertise. However, because the user does not have prior knowledge engineering experience, this knowledge will not be always optimally represented and organized. Examples of expected problems with the knowledge base are: semantic inconsistencies between the different levels of abstractions of a learned knowledge piece, existence of redundant, semantically equivalent knowledge pieces, and violation of knowledge engineering principles. A critical research issue is the development of mixed-initiative knowledge base reformulation and optimization methods for a knowledge engineer and an agent. These methods will have to require no or very limited assistance from a subject matter expert.

We plan to develop methods for the improvement of individual knowledge pieces. An example of a poorly learned task reduction rule is one that has too many except-when conditions (i.e. conditions that should not be satisfied in order for the rule to be applicable). One may improve such a rule by replacing several except-when conditions with their generalization. Yet another example of a poorly learned rule is one containing too many subtasks. Such a rule may need to be replaced with several simpler rules.

We also plan to develop methods for improving the organization of the entire knowledge base. For instance, the agent may have learned too many rules for reducing the same problem solving task. This will require a reorganization of the rules into different strategy-related clusters and learning of intermediate rules to connect the clusters. The high flexibility of the natural language leads to the proliferation of learned tasks that correspond to different paraphrases of the same expression. We plan to develop methods for detecting such similarities and unifying the equivalent tasks. These methods will integrate natural language processing of the informal representation of the tasks with analogical reasoning based on their formal representation.

13 Intelligent tutoring

As indicated in Figure 1, after the agent has been trained by a subject matter expert (during phase 2), it can be handed-over to a less-experienced person. In such a case, the agent, which was taught by a subject matter expert, reverses its role, teaching its user in a way that is similar to how it was taught, through specific examples and explanations, clearly showing its thread of logic (see phase 3 in Figure 1). An important research issue is therefore the development of a novel approach to intelligent tutoring that takes advantage of our instruction-based approach to knowledge acquisition.

14 Conclusions

In the mainframe computers age, the software systems were both built and used by computer science experts. In the current age of personal computers, these systems are still being built by computer science experts, but many of them (such as text processors, email programs, or Internet browsers) are now used by persons that have no formal computer education. Continuing this trend, we think that the next age will be that of the personal agents, where typical computer users will be able to both develop and use special types of software agents. We think that the Disciple approach, and the future developments outlined above, contribute directly to this new age in the software systems development process.

Acknowledgments

This research was sponsored by DARPA, AFRL, AFMC, USAF, under agreement number F30602-00-2-0546, by the AFOSR under grant no. F49620-00-1-0072, and by the US Army War College.

References

[Boicu *et al.*, 2003] Cristina Boicu, Gheorghe Tecuci, Mihai Boicu, and Dorin Marcu. Improving the Representation Space through Exception-Based Learning. To appear in *Proceedings of the Sixteenth International Flairs Conference.* 2003.

[Boicu, 2002] Mihai Boicu. *Modeling and Learning with Incomplete Knowledge* (Doctoral Dissertation). Learning Agents Laboratory, George Mason University: Fairfax, Virginia, 2002.

[Bowman, 2002] Michael Bowman. *A Methodology for Modeling Expert Knowledge that Supports Teaching Based Development of Agents* (Doctoral Dissertation). Learning Agents Laboratory, George Mason University: Fairfax, Virginia, 2002.

[Grishman, 1986] Ralph Grishman. *Computational linguistics*. Cambridge University Press: Cambridge, UK, 1986.

[Hendler and McGuinness, 2000] James Hendler, and Deborah L. McGuinness. The DARPA Agent Markup Language. *IEEE Intelligent Systems*, 15(6): 67-73, 2000.

[John and Kieras, 1996] Bonnie E. John, and David E. Kieras. The GOMS Family of User Interface Analysis Techniques: Comparison and Contrast. *ACM Transactions on Computer-Human Interaction*, 3(4): 320-351, 1996.

[Lenat, 1995] Doug B. Lenat. CYC: A Large-scale Investment in Knowledge Infrastructure. *Comm. ACM,* 38(11): 33-38, 1995.

[Mitchell, 1997] Tom M. Mitchell. *Machine Learning*. McGraw-Hill: New York, NY, 1997.

[Niles and Pease, 2001] Ian Niles, and Adam Pease. Towards a Standard Upper Ontology. In Christopher A. Welty, and Barry Smith, (Eds.) *Formal Ontology in Information Systems: Collected Papers from the Second International Conference*, 2-9. ACM Press: New York, NY, 2001

[Tecuci, 1988] Gheorghe Tecuci. DISCIPLE: A Theory, Methodology and System for Learning Expert Knowledge. *Thése de Docteur en Science*, University of Paris-South, France, 1988.

[Tecuci, 1998] Gheorghe Tecuci. *Building Intelligent Agents: An Apprenticeship Multistrategy Learning Theory, Methodology, Tool and Case Studies*. Academic Press, London, 1998.

[Tecuci and Boicu, 2002] Gheorghe Tecuci, and Mihai Boicu. Military Applications of the Disciple Learning Agent. In Jain, L. (Ed.), *Advanced Information Systems in Defense and Related Applications*. Springer Verlag: Berlin, Germany, 2002.

[Tecuci *et al.*, 2002] Gheorghe Tecuci, Mihai Boicu, Dorin Marcu, Bogdan Stanescu, Cristina Boicu, and Jerome Comello. Training and Using Disciple Agents: A Case Study in the Military Center of Gravity Analysis Domain. *AI Magazine*, 23(4): 51–68. AAAI Press, Menlo Park, California, 2002.