Disciple-RKF/COG: Agent Teaching by Subject Matter Experts

Mihai Boicu[1], Gheorghe Tecuci[1,2], Dorin Marcu[1], Bogdan Stanescu[1], Cristina Boicu[1], Catalin Balan[1], Marcel Barbulescu[1] and Xianjun Hao[1]

[1]Learning Agents Laboratory, Department of Computer Science, MSN 4A5, George Mason University, Fairfax, VA 22030
{mboicu, tecuci, dmarcu, bstanesc, ccascava, gbalan, mbard, xhao}@gmu.edu
[2]Center for Strategic Leadership, US Army War College, 650 Wright Ave, Carlisle Barracks, PA 17013

Introduction

We are addressing the knowledge acquisition bottleneck in the development of knowledge-based systems by elaborating the Disciple theory and methodology that enables subject matter experts to build such systems by themselves, with limited assistance from knowledge engineers (Tecuci 1998). The investigated solution consists of developing a very capable learning agent shell that can perform many of the functions of a knowledge engineer. As an expert system shell, the learning agent shell includes a general problem solving engine that can be reused for multiple applications. In addition, it includes a multistrategy learning engine for building its knowledge base (KB) which has two main components: an object ontology that defines the concepts from a specific application domain, and a set of task reduction rules expressed with these concepts. The subject matter expert and the agent engage into a mixed-initiative reasoning process during which the expert is teaching the agent his problem solving expertise, and the agent learns from the expert, building, verifying, and improving its KB.

Over the years we have developed a series of increasingly more capable learning agent shells from the Disciple family. The most recent family member, Disciple-RKF/COG, represents a significant advancement over its predecessors. It implements a more powerful plausible version space representation that allows all the types of knowledge from the KB (not only the rules, but also the objects and the tasks) to be learned with similar methods. Moreover, the partially learned knowledge pieces are represented at several levels of formalization, from natural language to formal logic, facilitating expert-agent communication, mixed-initiative problem solving, and learning. As a consequence, Disciple-RKF/COG incorporates new tools that allow a subject matter expert to perform additional knowledge engineering tasks, such as scenario specification, modeling of his problem solving process, and task formalization.

Disciple-RKF/COG was used and evaluated in several courses at the US Army War College, with very promising results, being made part of their regular syllabi.

Mixed-Initiative KB Development

The top part of Figure 1 shows the complex knowledge engineering activities that are generally required to build a KB. The knowledge engineer (KE) has to develop a model of the application domain that makes explicit the way the subject matter expert (SME) solves problems. Then the knowledge engineer has to develop the object ontology. He also needs to define general problem solving rules and to debug them.

The main idea of the theory implemented in the Disciple-RKF/COG learning agent shell is to replace these complex KB development activities performed by a knowledge engineer and a subject matter expert, with equivalent ones performed by the expert and a learning agent (Agent), through mixed-initiative reasoning, as shown in the lower part of Figure 1. The knowledge engineer is still needed to help the subject matter expert to define an initial domain model and to develop an initial object ontology. After that, however, the domain model and the ontology can be extended and refined by the expert and the Agent, with limited assistance from the knowledge engineer. For the complex activities of defining, verifying and updating the problem solving rules, the assistance needed from the knowledge engineer is much more limited. The subject matter expert can teach the Agent how to solve problems, through examples and explanations, and the Agent can learn and refine the rules by itself.

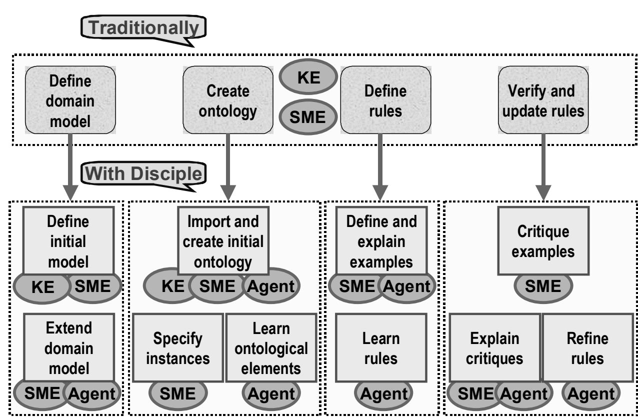

Figure 1: Complex knowledge engineering activities replaced with simpler mixed-initiative activities.

Disciple teaching by a subject matter expert

An important feature of the Disciple agent development approach is that it distinguishes very clearly the phases where the knowledge engineer plays a critical role, from those that are primarily performed by the subject matter expert.

First, the knowledge engineer has to work with the subject matter expert to develop an initial model of how the expert solves problems, based on the task reduction paradigm. This model identifies also the object concepts that need to be present in Disciple's ontology so that it can perform this type of reasoning. These object concepts represent a specification of the needed ontology, specification that guides the process of importing ontological knowledge from existing knowledge repositories. Then the knowledge engineer and the subject matter expert extend the imported ontology and define the scripts for elicitation of specific scenarios.

After the object ontology has been developed, the subject matter expert can teach the Disciple-RKF/COG agent how to solve problems, with very limited assistance from a knowledge engineer. Figure 2 shows the main steps of the agent teaching process. During Scenario specification Disciple guides the subject matter expert to describe a scenario and creates a formal representation of it consisting of instances in the object ontology. Then, in the modeling phase, the expert shows Disciple how to solve problems, by using the task reduction paradigm. The expert has to formulate an initial problem solving task. Then he has to successively reduce this task to simpler tasks, until a solution is found. This entire problem solving process is expressed in English. In the task and rule learning phase Disciple learns general tasks and rules from the task reduction steps defined in the modeling phase. In the refinement phase Disciple uses the partially learned tasks and rules in problem solving and refines them based on the expert's feedback. While this is the normal sequence of the teaching phases, there is also a need to return to a previous phase when, during problem solving, the expert needs to define a new reduction, thus performing modeling, task formalization and rule learning.

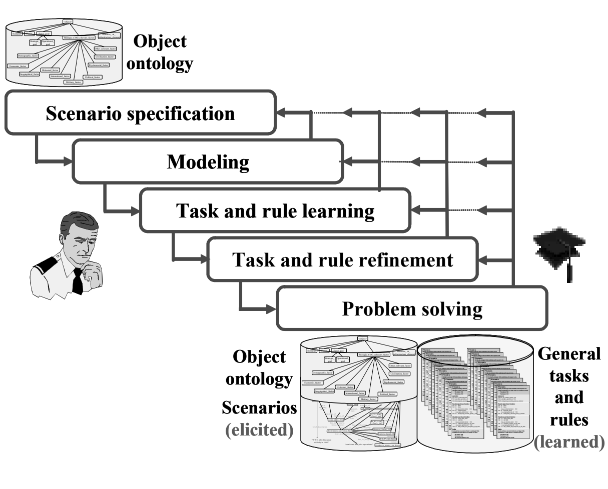

Figure 2: The main phases of the agent training

Final remarks

The Disciple-RKF/COG instructable agent is used in a sequence of two courses taught regularly at the US Army War College, "Case Studies in Center of Gravity Analysis," and "Military Applications of Artificial Intelligence" (Tecuci et al. 2002). In the first course the students use a Disciple agent that was already taught the expertise of the course's instructor in center of gravity analysis (Department of the Army 2001). During the course, the students become familiar with Disciple-RKF/COG as end-users, using it as an aid for learning about center of gravity analysis, and for developing a report containing a case study analysis. 9 of the 13 students in the Winter 2002 session of this course agreed, and the other 4 strongly agreed with the statement "The use of Disciple is an assignment that is well suited to the course's learning objectives."

In the "Military Applications of Artificial Intelligence" course, each student uses a Disciple-RKF/COG agent that does not contain any reasoning rule, and teaches it his own problem solving expertise in center of gravity analysis. The Spring 2001 session of this course ended with a final agent teaching experiment. At the end of the experiment 7 out of the 10 experts (which are high ranking military officers) agreed, 1 expert strongly agreed and 2 experts were neutral with respect to the statement: "I think that a subject matter expert can use Disciple to build an agent, with limited assistance from a knowledge engineer." To our knowledge, this is the first time that subject matter experts have trained an agent their own problem solving expertise, with very limited assistance from a knowledge engineer. This experimental result supports our long term vision of developing a capability that will allow typical computer users to build and maintain their own assistants, as easily as they now use personal computers for text processing.

Acknowledgments. This research was sponsored by DARPA, AFRL, AFMC, USAF, under agreement number F30602-00-2-0546, by AFOSR under grant no. F49620-00-1-0072, and by the US Army War College. Jerry Comello, Mike Bowman, Chip Cleckner, Jim Donlon, and Tony Lopez have contributed to the application of Disciple RKF/COG at the US Army War College.

References

Department of the Army 2001. *Field Manual 3-0, Operations*. Washington, D.C.: U.S. Gov. Printing Office.

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

Tecuci G.; Boicu M.; Marcu D.; Stanescu B.; Boicu C.; Comello J.; Lopez T.; Donlon J.; and Cleckner C. 2002. Development and Deployment of a Disciple Agent for Center of Gravity Analysis. In *Proceedings of the Fourteenth Annual Conference on Innovative Applications of Artificial Intelligence.* Menlo Park, Calif.: AAAI Press.