Mixed-Initiative Exception-Based Learning for Knowledge Base Refinement

Cristina Boicu, Gheorghe Tecuci, Mihai Boicu

Learning Agents Laboratory, Computer Science Department, MS 4A5, George Mason University, 4400 University Dr, Fairfax, VA 22030, Phone (703) 993-4669 {ccascava, tecuci, mboicu}@gmu.edu, http://lalab.gmu.edu, http://lalab.gmu.edu/cristina

Introduction

Over the years we have developed the Disciple approach for the rapid development of knowledge bases and knowledge-based agents, by subject matter experts, with limited assistance from knowledge engineers (Tecuci 1998). This approach relies on a Disciple learning agent that can be trained to solve problems by an expert. First, however, a knowledge engineer has to work with the expert to define the object ontology of Disciple. This ontology consists of hierarchical descriptions of objects and features from the application domain. Then, the expert can teach Disciple to solve problems in a way that resembles how the expert would teach a student. For instance, the expert defines a specific problem, helps the agent to understand each reasoning step toward the solution, and supervises and corrects the agent’s behavior, when it attempts to solve new problems. During such mixed-initiative interactions, the agent learns general problem solving rules from individual problem solving steps and their explanations of success or failure. A critical role in this multistrategy rule learning process is played by the object ontology, which is used as the generalization hierarchy.

Mixed-Initiative Exception-Based Learning

The Disciple approach was successfully used in an agent training experiment at the US Army War College, where experts succeeded to teach personal Disciple agents their own problem solving expertise in military center of gravity (COG) determination (Boicu et al. 2001). This experiment, however, revealed that the rules learned from subject matter experts have a significant number of negative exceptions. A negative exception is a negative example that is covered by the rule, because the current object ontology does not contain any object concept or feature-value pair that distinguishes between all the positive examples of the rule, on one side, and this negative example, on the other side (Wrobel 1989). Therefore, in the context of the current ontology, the rule cannot be specialized to uncover the negative example, which is kept as a negative exception.

Such rule exceptions provide valuable information on how the ontology should be extended to represent the subtle distinctions that real experts make in their domain.

We are developing a suite of mixed-initiative multistrategy methods for learning new object concepts and features that extend the object ontology, allowing the elimination of the rule’s exceptions. The first type of methods involves only the Disciple agent and the expert, and considers one rule with its exceptions at a time. The second class of methods considers again one rule with its exceptions at a time, but requires also the participation of a knowledge engineer in the mixed-initiative learning process. Finally, the third and most complex type of methods are global, considering all the exceptions from the knowledge base, and involving both the expert and the knowledge engineer. All the methods have four major phases: a candidates discovery phase, a selection phase, an ontology refinement phase, and a rule refinement phase. In the candidates discovery phase, the Disciple agent generates an ordered set of candidates that have the potential of removing the exceptions. Each candidate is a new ontology piece (for instance, a new value of an existing feature, a new object feature, or even a new object concept) that has the potential of distinguishing between the positive examples and the negative exceptions. To generate these candidates and to order them by their plausibility, Disciple uses analogical reasoning heuristics, ontology design principles, and hints from the user. In the candidate selection phase, Disciple interacts with the user to test the most plausible candidates, and to select one of them. In the ontology refinement phase, Disciple elicits additional knowledge from the expert, related to the selected candidate. For instance, if the selected candidate is a new type of feature, then Disciple will attempt to elicit from the expert which other objects from the knowledge base have that feature, and will also learn a general definition of the feature. This definition includes a domain concept (which represents the set of objects that can have that feature), and a range concept (which represents the set of possible values of that feature). Finally, in the rule refinement phase, the rule is updated based on the refined ontology. Because of the central role of the object ontology as the generalization hierarchy for learning, an ontology change may potentially affect any rule from the knowledge
base, not only those with exceptions. We have therefore
developed methods for rapid rule relearning in the context
of the updated ontology. These methods maintain the
relevant knowledge from which an individual rule was
learned, such as generalized explanations and prototypical
examples, and automatically regenerate the rule.

We will illustrate the first type of exception handling
methods, in which the expert collaborates with Disciple to
analyze a rule with a negative exception. Figure 1 shows an
example of a task reduction step from the Center of Gravity
analysis domain (Boicu et al. 2001). It consists of a
problem solving task, a question relevant to the reduction
of this task, the answer to the question, and the subtask
resulted from this answer.

```
IF the task is
Identify a strategic COG candidate for Japan_1944 with respect to
other sources of strength and power
Question: What is a source of strength and power of Japan_1944?
Answer: Japanese_army_forces_on_Luzon
THEN
Japanese_army_forces_on_Luzon is a strategic COG candidate for
Japan_1944
```

Figure 1: A problem solving episode

Based on this problem solving episode, Disciple learns a
general task reduction rule. This rule, however, generates
the wrong solution “Japanese_expectation_for_negotiation
is a strategic COG candidate for Japan_1944,” which is
rejected by the expert. Because the ontology does not
contain any element that distinguishes between
“Japanese_army_forces_on_Luzon” and “Japanese_
expectation_for_negotiation,” the incorrect reasoning step
is kept as a negative exception of the rule. The expert can
invoke the Exception-Based Learning module, attempting
to extend the ontology by himself. First, Disciple proposes
him candidate extensions that have the potential of
removing the exception. For instance, Disciple looks for an
existing feature that may be associated with
“Japanese_army_forces_on_Luzon” (the positive example),
without being associated with “Japanese_
expectation_for_negotiation” (the negative exception), and
finds “is_a_strategically_important_military_capability_for.”
The domain of this feature is “Military_factor”, which
includes the positive example without including the
negative exception. The expert accepts this feature and
specifies that its value for the positive example is
Japan_1944. Next, Disciple guides the expert to also
specify this feature for other instances of Military_factor,
such as “Japanese_concentration_of_naval_assets.” Then,
it refines the object ontology with this new knowledge
acquired from the expert, as shown in Figure 2. Disciple
refines also the rule based on this knowledge, transforming
the negative exception into a negative example that is no
longer covered by the rule.

As illustrated above, the first class of methods discovers
limited extensions of the ontology (such as an additional
feature of an object when the feature definition is already
present in the ontology). The second class of methods leads
to more complex refinements, such as the definition of new
types of objects or the restructuring of the object hierarchy.
For instance, Disciple may elicit from the expert an
explanation of why the negative exception of a rule is an
incorrect problem solving episode, explanation represented
by a new type of object that is placed in the object
hierarchy. The methods from the third and most complex
class first hypothesize knowledge pieces for all the rules
with exceptions. Then, they analyze these hypotheses to
define an ordered set of hypotheses, each one eliminating
or reducing the exceptions from more than one rule.

**Conclusions**

Some of the above methods are already implemented in the
Exception-Based Learning module of Disciple-RKF/COG.
With these methods we are proposing a solution to the
complex problem of learning with an evolving
representation language, as represented by the object
ontology.

**Acknowledgements.** 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, M.; Tecuci, G.; Stanescu, B.; Marcu, D.; and
Cascaval (now Boicu), C. 2001. Automatic Knowledge
Acquisition from Subject Matter Experts. In *Proceedings
of the Thirteenth International Conference on Tools with
Artificial Intelligence*, 69-78. Los Alamitos, California:
IEEE Computer Society.

Apprenticeship Multistrategy Learning Theory, Methodology,

Wrobel, S. 1989. Demand-driven concept formation. In
Morik, K. ed. *Knowledge Representation and Organization
in Machine Learning. Lecture Notes in Artificial