Designing Expert Systems

7. Multistrategy Rule Learning

Gheorghe Tecuci
tecuci@gmu.edu
http://lac.gmu.edu/

Learning Agents Center
and Computer Science Department
George Mason University
Ed Feigenbaum (AAAI Address, 1993): *Rarely does a technology arise that offers such a wide range of important benefits of this magnitude. Yet as the technology moved through the phase of early adoption to general industry adoption, the response has been cautious, slow, and “linear” (rather than exponential).*

Another approach: Agent training directly by the subject matter expert

Bill Gates (NYT, 1 March 2004): *If you invent a breakthrough in artificial intelligence, so machines can learn, that is worth 10 Microsofts.*
Disciple Approach to Agent Development

Develop learning and problem solving agents that can be taught by subject matter experts to become knowledge-based assistants.

The expert teaches the agent how to solve problems in a way that resembles how the expert would teach a student, an apprentice or a collaborator.

The agent continuously develops and refines its knowledge base to capture and better represent expert’s knowledge and problem solving strategies.

There is no longer a clear distinction between knowledge base development and its maintenance.
Vision: Evolution of Software Development and Use

- **Mainframe Computers**: Software systems developed and used by computer experts.
- **Personal Computers**: Software systems developed by computer experts and used by persons who are not computer experts.
- **Learning Assistants**: Software systems developed and used by persons who are not computer experts.

**DISCIPLE**
Multidisciplinarity and Integration in Disciple

Intelligence analysis, Center of gravity determination, Course of action critiquing, Emergency response planning, Workaround reasoning, PhD advisor selection, Teaching higher order thinking skills.

Development of systematic approach to expert problem solving

Working closely with subject matter experts to model their reasoning

Development and application of Disciple agents

Working closely with end users to receive crucial and timely feedback

Development of the Disciple theory for agents teaching by non-computer experts

Army War College
Air War College
George Mason University
Knowledge Base Development Activities

Define domain model

Create ontology

Define rules

Verify and update rules

Traditionally

With Disciple

Define initial model

Import and create initial ontology

Define and explain examples

Critique examples

Extend domain model

Specify instances

Learn ontological elements

Learn rules

Explain critiques

Refine rules

Specify instances

Learn ontological elements

Learn rules

Explain critiques

Refine rules

Generate initial model

Import and create initial ontology

Define and explain examples

Critique examples

Extend domain model

Specify instances

Learn ontological elements

Learn rules

Explain critiques

Refine rules

Generate initial model

Import and create initial ontology

Define and explain examples

Critique examples

Extend domain model

Specify instances

Learn ontological elements

Learn rules

Explain critiques

Refine rules

Generate initial model

Import and create initial ontology

Define and explain examples

Critique examples

Extend domain model

Specify instances

Learn ontological elements

Learn rules

Explain critiques

Refine rules

Generate initial model

Import and create initial ontology

Define and explain examples

Critique examples

Extend domain model

Specify instances

Learn ontological elements

Learn rules

Explain critiques

Refine rules
Control of Modeling, Learning and Problem Solving

Problem 

Mixed-Initiative Problem Solving 

Reasoning Tree 

Extend Reasoning Tree 

Accept Reasoning Step 

Reject Reasoning Steps 

Rules Refinement 

Explain Examples 

Explain Examples 

Rules Learning 

Ontology + Rules 

Learned Rules 

Refined Rules 

Refined Ontology
The expert makes explicit how to solve a problem.

1. Modeling

- Assess whether John Doe is a potential PhD advisor for Bob Sharp.
  - Is Bob Sharp interested in the area of expertise of John Doe?
    - Yes, because Bob Sharp is interested in Artificial Intelligence which is the area of expertise of John Doe.
  - Assess whether John Doe is a potential PhD advisor for Bob Sharp in Artificial Intelligence.
    - Is John Doe likely to stay on the faculty of George Mason University for the duration of Bob Sharp's dissertation?
      - Yes, because John Doe has a tenured position which is a long term position.

2. Learning

- Assess whether John Doe would be a good PhD advisor for Bob Sharp in Artificial Intelligence.
  - Which is a PhD advisor quality criterion?
    - professional reputation
  - Assess whether John Doe would be a good PhD advisor for Bob Sharp with respect to professional reputation.

Rule 1

Rule 2

Rule 3

The agent learns reduction rules.
3. Solving
Applies learned rules to solve new problems

4. Critiquing
Accepts or rejects individual reductions

5. Refining
Refines learned rules with the obtained positive and negative examples

---

3. Solving
Assess whether Bridget Jones is a potential PhD advisor for Bob Sharp.

- Is Bob Sharp interested in the area of expertise of Bridget Jones?
  - Yes, because Bob Sharp is interested in Artificial Intelligence which is the area of expertise of Bridget Jones.

- Assess whether Bridget Jones is a potential PhD advisor for Bob Sharp in Artificial Intelligence.

- Is Bridget Jones likely to stay on the faculty of George Mason University for the duration of Bob Sharp's dissertation?
  - Yes, because Bridget Jones has a tenured position which is a long term position.

- Assess whether Bridget Jones would be a good PhD advisor for Bob Sharp.

4. Critiquing

- Which is a PhD advisor quality criterion?
  - support for students

- Assess whether Bridget Jones would be a good PhD advisor for Bob Sharp with respect to support for students.

- Which is a PhD advisor quality criterion?
  - students learning experience

- Assess whether Bridget Jones would be a good PhD advisor for Bob Sharp with respect to students learning experience.

- Which is a PhD advisor quality criterion?
  - responsiveness to students

- Assess whether Bridget Jones would be a good PhD advisor for Bob Sharp with respect to responsiveness to students.

- Which is a PhD advisor quality criterion?
  - quality of student results

- Assess whether Bridget Jones would be a good PhD advisor for Bob Sharp with respect to quality of student results.

- Which is a PhD advisor quality criterion?
  - professional reputation

- Assess whether Bridget Jones would be a good PhD advisor for Bob Sharp with respect to professional reputation.

- Which is a PhD advisor quality criterion?
  - personality and compatibility with student

- Assess whether Bridget Jones would be a good PhD advisor for Bob Sharp with respect to personality and compatibility with student.
Overview

Introduction

Multistategy Rule Learning

Strategies for Explanation Generation

Demo and Hands-on

Explanations with Comparisons

Explanations with Functions

Reading
Control of Modeling, Learning and Problem Solving

Problem

Mixed-Initiative Problem Solving

Reasoning Tree

Extend Reasoning Tree

Accept Reasoning Step

Reject Reasoning Steps

Explain Reasoning Steps

Rules Refinement

Explain Examples

Learned Rules

Refined Rules

Refined Ontology

Ontology + Rules

Rules Learning

Explain Examples

I can be taught to help you.
GIVEN:

• an example of a problem reduction;

• a knowledge base that includes an object ontology and a set of problem reduction rules;

• an expert that understands why the given example is correct and may answer agent’s questions.

DETERMINE:

• a plausible version space rule that is a plausible generalization of the specific problem reduction.
Rule Learning

LEARNED REDUCTION RULE

REDUCTION EXAMPLE

Rule Viewer

DECOMPOSITION RULE DDR.00000 FORMAL DESCRIPTION

IF: Assess whether ?O1 is a potential PhD advisor for ?O2.

Q: Is ?O2 interested in the area of expertise of ?O1?
A: Yes, because ?O2 is interested in ?O3 which is the area of expertise of ?O1.

MAIN CONDITION

<table>
<thead>
<tr>
<th>Var</th>
<th>Lower Bound</th>
<th>Upper Bound</th>
</tr>
</thead>
<tbody>
<tr>
<td>?O1</td>
<td>(PhD advisor, associate professor)</td>
<td>(person)</td>
</tr>
<tr>
<td>?O2</td>
<td>(PhD student)</td>
<td>(person)</td>
</tr>
<tr>
<td>?O3</td>
<td>(computer science)</td>
<td>(PhD research area)</td>
</tr>
</tbody>
</table>

Var | Relationship | Var
---|--------------|---
?O2 | is interested in | ?O3
?O1 | is expert in   | ?O3

Basic Steps of the Rule Learning Method

1. Find a formal explanation of why the example is correct. This explanation is an approximation of the question and the answer, in the object ontology.

2. Generalize the example and the explanation into a plausible version space rule.
The Rule Learning Method: Details

1. Identify a formal explanation EX of why the example E is correct, through mixed-initiative interaction with the subject matter expert. The explanation is an approximation of the meaning of the question and answer, expressed with the objects and the features from the object ontology. During the explanation generation process, new objects and features may be elicited from the expert and added to the object ontology.

2. Generate a variable for each instance, number and string that appears in the example and its explanation. Then use these variables, the example, and the explanation, to create an instance IC of the concept representing the applicability condition of the rule to be learned. This is the concept to be learned as part of rule learning.

3. Generate the problems, question, and answer of the rule by replacing each instance or constant from the example E with the corresponding variable generated in step 2. Then generate the plausible version space of the applicability condition of the rule. The concept represented by this condition is the set of instances and constants that produce correct instantiations of the rule. The plausible lower bound of this version space is the minimally general generalization of IC determined in step 2, generalization which does not contain any instance. The plausible upper bound of this version space is the set of the maximally general generalizations of IC.

5. If there is any variable from the THEN part of a rule which is not linked to some variable from the IF part of the rule, or if the rule has too many instances in the knowledge base, then interact with the expert to extend the explanation of the example and update the rule if new explanation pieces are found. Otherwise end the rule learning process.
The Rule Learning Method

Analogy and Hint Guided Explanation

Example of a problem reduction step

Guidance, hints
analogy
plausible explanations
Knowledge Base

Analogy-based Generalization

Plausible version space rule
PUB
PLB
Incomplete justification

Analogy and Hint Guided Explanation

Guidance, hints
analogy
plausible explanations
Knowledge Base
Find an Explanation of Why the Example Is Correct

The explanation is an approximate representation of the question and its answer, in the object ontology.

Bob Sharp — is interested in —> Artificial Intelligence
John Doe — is expert in —> Artificial Intelligence
Explanation Generation

The expert can guide the agent in explanation generation by selecting the objects from the example for which explanation pieces will be proposed.

Plausible explanation pieces proposed by the agent. The expert has to select the correct ones.
Generate Rule’s Condition

Task
Assess whether ?O1 is a potential PhD advisor for ?O2.

Is ?O2 interested in the area of expertise of ?O1?
Yes, because ?O2 is interested in ?O3 which is the area of expertise of ?O1.

Sub-task

Rewrite the objects from the example as an applicability condition

?O1 is John Doe
is expert in ?O3

?O2 is Bob Sharp
is interested in ?O3

?O3 is Artificial Intelligence
Generate Rule’s Condition

- **Object**
  - **Agent**
    - **Person**
      - **Employee**
        - **Faculty Member**
          - **Professor**
            - **Associate Professor**
              - **John Doe**
              - **Bob Sharp**
        - **Student**
          - **Graduate Student**
            - **PhD Advisor**
              - **PhD Student**

- **Relationships**
  - **is**
  - **is a subconcept of**
  - **is an instance of**

- **Most specific generalization**
  - LB (Lower Bound)
  - UB (Upper Bound)

- **Var | Lower Bound | Upper Bound**
  - ?O1 (PhD advisor, associate professor) (person)
  - ?O2 (PhD student) (person)
  - ?O3 (computer science) (PhD research area)

- **Var | Relationship | Var**
  - ?O2 is interested in ?O3
  - ?O1 is expert in ?O3

**Examples**

- ?O1 is John Doe is expert in ?O3
- ?O2 is Bob Sharp is interested in ?O3
- ?O3 is Artificial Intelligence

**Sentences**

- Person is expert in PhD research area
- Person is interested in PhD research area
- ?O2 is interested in ?O3
Notice that the explanation is first re-written as a condition, and then two generalizations of this condition are created: a most conservative one (the plausible lower bound condition) and a most aggressive one (the plausible upper bound condition).

The plausible lower bound is the minimal generalization of the condition from the left hand side of the slide. Similarly, the most general generalization of the condition is the plausible upper bound.

The agent uses various constraints from the knowledge base to restrict the values that the variables could take.
Rule Learned from an Example and its Explanation

LEARNED REDUCTION RULE

REDUCTION EXAMPLE

DECOMPOSITION RULE DDR.00000 FORMAL DESCRIPTION

IF: Assess whether ?O1 is a potential PhD advisor for ?O2.

Q: Is ?O2 interested in the area of expertise of ?O1?
A: Yes, because ?O2 is interested in ?O3 which is the area of expertise of ?O1.

MAIN CONDITION

<table>
<thead>
<tr>
<th>Var</th>
<th>Lower Bound</th>
<th>Upper Bound</th>
</tr>
</thead>
<tbody>
<tr>
<td>?O1</td>
<td>(PhD advisor, associate professor)</td>
<td>(person)</td>
</tr>
<tr>
<td>?O2</td>
<td>(PhD student)</td>
<td>(person)</td>
</tr>
<tr>
<td>?O3</td>
<td>(computer science)</td>
<td>(PhD research area)</td>
</tr>
</tbody>
</table>


Bob Sharp — is interested in — Artificial Intelligence — is expert in — John Doe
Analogical Reasoning

I need to
Assess whether John Doe is a potential PhD advisor for Bob Sharp.

Therefore I need to
Assess whether John Doe is a potential PhD advisor for Bob Sharp in Artificial Intelligence.

I need to
Assess whether Dan Smith is a potential PhD advisor for Peter Jones.

Therefore I need to
Assess whether Dan Smith is a potential PhD advisor for Peter Jones in Information Security.
The agent uses analogical reasoning to generalize the example and its explanation into a plausible version space rule. This slide provides a justification for the generalization procedure used by the agent.

Let us consider that the expert has provided to the agent the problem reduction example from the bottom left of this slide. This reduction is correct because

Now let us consider

Using the same logic as above, one can create the problem reduction example from the bottom right of the slide.

This is a type of analogical reasoning that the agent performs. The explanation from the left hand side of this slide explains the problem reduction from the left hand side. This explanation is similar with the explanation from the right hand side of this slide (they have the same structure, being both less general than the analogy criterion from the top of this slide). Therefore one could expect that this explanation from the right hand side of the slide would explain an example that would be similar with the initial example. This example is the one from the right hand side of the slide.

To summarize: The expert provided the example from the left hand side of this slide and helped the agent to find its explanation. Using analogical reasoning the agent can perform by itself the reasoning from the bottom right hand side of the slide.
Assess whether John Doe is a potential PhD advisor for Bob Sharp.

Therefore I need to:
Assess whether John Doe is a potential PhD advisor for Bob Sharp in Artificial Intelligence.

I need to:

Therefore I need to:
Notice that in the previous illustration we could have used any other entities \(?O1, \?O2\) and \(?O3\) instead of Bob Sharp, Artificial Intelligence and John Doe. As long as \(?O1\) is interested in \(?O3\) and \(?O2\) is expert in \(?O3\), the agent would hypothesize that, in order to “Assess whether \(?O1\) is a potential PhD advisor for \(?O2\)” then one would need to “Assess whether \(?O1\) is a potential PhD advisor for \(?O2\) in \(?O3\).”

The agent uses various constraints from the knowledge base to restrict the values that the variables \(?O1, \?O2\) and \(?O3\) could take. For instance, \(?O1\) should have the feature “is interested in” and the domain of this feature (i.e. the set of objects that may have this feature) is person. Therefore \(?O1\) should be a person.

Using this kind of reasoning, the agent generalizes the example from the left hand side of this slide to the expression from the right hand side of this slide.
I need to
Assess whether John Doe is a potential PhD advisor for Bob Sharp.

Therefore I need to
Assess whether John Doe is a potential PhD advisor for Bob Sharp in Artificial Intelligence.

I need to
Assess whether ?O1 is a potential PhD advisor for ?O2.

Therefore I need to

Knowledge-base constraints on the generalization:
Any value of ?O1 should be an instance of: 
DOMAIN(is interested in) = person
Any value of ?O3 should be an instance of: 
RANGE(is interested in) = PhD research area

Generalization by Analogy
Learning with an Evolving Representation Language

IF
<task>

Plausible Upper Bound Condition
<PUB condition>

Plausible Lower Bound Condition
<PLB condition>

THEN
<subtask 1>
...
<subtask m>

Plausible version space

Universe of Instances

Exact Condition

Plausible Lower Bound Condition
<PLB condition>

Plausible Upper Bound Condition
<PUB condition>

Universe of Instances

IF
<task>

THEN
<subtask 1>
...
<subtask m>
Characterization of the Learned Rule

Universe of Instances

$E_h$

Plausible Upper Bound Condition

Plausible Lower Bound Condition
The plausible upper bound condition of the learned rule is an analogy criterion that allows the agent to solve problems by analogy with the example from which the rule was learned. Because analogy is only a plausible reasoning process, some of the examples covered by the rule may be wrong. The plausible upper bound of the rule is therefore only an approximation of a hypothetical exact condition that will cover only positive examples of the rule. That is why it is called plausible upper bound.

The plausible lower bound condition of the rule covers the input example that is known to be correct. However, the bound is a minimal generalization performed in the context of an incomplete ontology (generalization hierarchy). Therefore it is also a plausible bound.

The previous slide shows the most likely relation between the plausible lower bound, the plausible upper bound and the hypothetical exact condition of the rule. Notice that there are instances of the plausible upper bound that are not instances of the hypothetical exact condition of the rule. This means that the learned rule could also generate wrong solutions to some problems, as already mentioned. Also, there are instances of the hypothetical exact condition that are not instances of the plausible upper bound. This means that the plausible upper bound does not cover all the cases in which the solution provided by the rule would be correct.

Similarly, there may be cases that are covered by the plausible lower bound, without being covered by the hypothetical exact condition. All these situations are a consequence of the fact that the explanation of the initial example might be incomplete, and that the representation language for learning (which is based on the object ontology) might also be incomplete. These results are consistent with what one would expect from an agent performing analogical reasoning.
Overview

- Introduction
- Multistartegy Rule Learning
- Strategies for Explanation Generation
- Demo and Hands-on
- Explanations with Comparisons
- Explanations with Functions
- Reading
General Heuristics for Explanation Generation

Look for the relationships between the objects from the question and the answer.

Look for the relationships between an object from the IF problem and an object from the question or the answer.
The expert selects an object from the example. The agent generates a list of plausible explanations containing that object. The expert selects the correct explanation(s).
1. Look for a rule $R_k$ that reduces the current problem $P_1$.
2. Extract the explanations $E_g$ from the rule $R_k$.
3. Look for explanations of the current problem reduction that are similar with $E_g$.

Example to be explained:

$$\text{IF the problem to solve is } P_1$$
$$\text{THEN solve } P_{1a}, \ldots, P_{1d}$$

Look for explanations that are similar with $E_g$

Previously learned rule $R_k$:

$$\text{IF the problem to solve is } P_{1g}$$
$$\text{Explanation } E_g$$
$$\text{PUB condition}$$
$$\text{PLB condition}$$
$$\text{THEN accomplish } P_{11g}, \ldots, P_{1ng}$$
This heuristic is based on the observation that the explanations of the alternative reductions of a problem tend to have similar structures. The same factors are considered, but the relationships between them are different.

Question:
Answers:
Explanations:
Another Analogical Reasoning Heuristic

1. Look for a rule $R_k$ that reduces a similar problem to similar subproblems.

2. Extract the explanations $E_g$ from the rule $R_k$.

3. Look for explanations of the current problem reduction that are similar with $E_g$. 
This heuristic is based on the observation that similar problem solving episodes tend to have similar explanations:

\[ E \rightarrow \text{similar} \rightarrow E' \]
\[ \text{explains} \rightarrow \text{similar} \rightarrow \text{explains?} \]

Justification of the Heuristic
Yet Another Analogical Reasoning Heuristic

1. Look for a rule $R_k$ that reduces a problem that is similar to the current problem even if the subproblems are not similar.

2. Extract the explanations $E_g$ from the rule $R_k$.

3. Look for explanations of the current problem reduction that are similar with $E_g$.

The plausible explanations found by the agent can be ordered by their plausibility (based on the heuristics used).
Sometimes no formal explanation is necessary, as in the above example.

We need to invoke Rule Learning, but then quit it without selecting any explanation. The agent will generalize this example to a rule.
Select Reasoning ➔ Mixed-Initiative Reasoning
1. Click on a problem

2. Click on “Select”
Disciple will select the first modeling node, in a depth-first search, to learn a rule from the corresponding reasoning step. Alternatively, you can browse the reasoning tree, click on a modeling node (usually a question/answer node) of a modeling step, and then click on “Continue Learning” to learn a rule from that step. The modeling nodes have yellow borders to distinguish them from the nodes generated by learned rules, which have grey borders.
2. Click on "Start Learning"

1. The first node of the reasoning tree is selected

Assess whether John Doe is a potential PhD advisor for Bob Sharp.

Is Bob Sharp interested in the area of expertise of John Doe? Yes, because Bob Sharp is interested in Artificial Intelligence which is the area of expertise of John Doe.
Example Explanation Interface

Reduction example

Objects from the example

Click on an object to deselect or select it

Search explanation pieces

Accept selected explanation piece

See more explanation pieces

User defines an explanation piece

Explanation pieces related to the selected objects

Finish the explanation process

Task
Assess whether John Doe is a potential PhD advisor for Bob Sharp.

Question
Is Bob Sharp interested in the area of expertise of John Doe?

Answer
Yes, because Bob Sharp is interested in Artificial Intelligence which is the area of expertise of John Doe.

Explanations

Sub-task
Assess whether John Doe is a potential PhD advisor for Bob Sharp in Artificial Intelligence.
1. Select a relevant piece of explanation

2. Click on “Accept”
1. The selected explanation is added to the example

2. Select another relevant piece of explanation

3. Click on “Accept”
1. Select a previously accepted piece of explanation, if you would like to remove it

2. Click on “Remove”
Click on “Finish” to end the rule learning process
1. Disciple has learned a rule and has applied it to generate this step.

2. Click on “Decomposition/Reduction Rule” to see the learned rule (this button is visible when a question/answer node is selected in the Reasoning Hierarchy panel).
1. Learned rule

2. Click on “x” to close the window
Click on “Continue Learning”
2. Click on “Start Learning”

1. The next modeling node of the reasoning tree is selected
1. Select a relevant piece of explanation

2. Click on “Accept”

3. Click on Finish because no other explanation pieces are needed.
1. Disciple has learned a rule and has applied it to generate this step.

2. Click on “Decomposition Rule” to see the learned rule (this button is visible when a question/answer node is selected in the Reasoning Hierarchy panel).
1. Learned rule

DECOMPOSITION RULE DDR.00001 FORMAL DESCRIPTION

**IF:** Assess whether ?O1 is a potential PhD advisor for ?O2 in ?O3.

**Q:** Is ?O1 likely to stay on the faculty of ?O4 for the duration of ?O2's dissertation?

**A:** Yes, because ?O1 has ?O5 which is a long term position.

**MAIN CONDITION**

<table>
<thead>
<tr>
<th>Var</th>
<th>Lower Bound</th>
<th>Upper Bound</th>
</tr>
</thead>
<tbody>
<tr>
<td>?O1</td>
<td>(PhD advisor, associate professor)</td>
<td>(person)</td>
</tr>
<tr>
<td>?O2</td>
<td>(PhD student)</td>
<td>(agent)</td>
</tr>
<tr>
<td>?O3</td>
<td>(computer science)</td>
<td>(research area)</td>
</tr>
<tr>
<td>?O4</td>
<td>(university)</td>
<td>(employer)</td>
</tr>
<tr>
<td>?O5</td>
<td>(long term position)</td>
<td>(long term position)</td>
</tr>
</tbody>
</table>

**THEN:** Assess whether ?O1 would be a good PhD advisor for ?O2 in ?O3.

2. Click on “x” to close the window
1. Browse the tree and select the next modeling (question/answer) node to learn a rule from the corresponding reasoning step.

2. Click on “Continue Learning”
2. Click on “Start Learning”
1. Select the relevant piece of explanation

2. Click on “Accept”

3. Click on Finish because no other explanation pieces are needed for this example.
1. Disciple has learned a rule and has applied it to generate all these steps.

2. The rule may be refined by indicating which of the generated reductions are correct or incorrect or by modifying the explanation.
1. Browse the tree and select the next modeling (question/answer) node to learn a rule from the corresponding reasoning step.

2. Click on “Continue Learning” and then click on “Start Learning” on the follow-on screen.
1. Select the relevant piece of explanation

2. Click on “Accept”

3. Click on Finish because no other explanation pieces are needed for this example.
1. Disciple has learned a rule and has applied it to generate all these steps.

2. The rule may be refined by indicating which of the generated reductions are correct or incorrect or by modifying the explanation.
1. In the **Refinement** tab

2. Click on **Continue Learning** button

3. Look at the context of the selected node

4. Click on **Start Learning** button

5. Select the explanation piece that best represents part of the question/answer pair.

6. Click on **Accept** button

7. Repeat steps 5 and 6 to accept all the needed explanation pieces.

8. Click on the **Finish** button to end explanation selection and learn the rule

9. Restart the process as long as the button **Continue Learning** is active
Hands-on: Rule Learning
Hands On: Rule Learning

Install the system from:

Load the “PAD-m2o\CS681” scenario KB.
Overview

Introduction

Multistategy Rule Learning

Strategies for Explanation Generation

Demo and Hands-on

Explanations with Comparisons

Explanations with Functions

Reading
An example with an explanation that requires a comparison between two numbers
Select the number in the text editor and right click over the selection.
Select "Create Number Constant" from the popup menu
This is now recognized by Disciple as being a number.

Repeat the same procedure for the other numbers.
After all numbers are correctly identified, start the learning process.
Accept the relevant explanations that link the numbers with other ontology elements.
Click on “Create New…” to define the explanation that compares the two numbers.
In the “Create explanation” dialog, type the numbers in the left and, respectively, right editors. Type the comparison operator in the middle editor.

The following comparison operators can be used: >, <, <=, >= and !=

Click “OK” when done.
The new explanation was automatically accepted.

Click “Finish” when done.
Disciple learned a rule with a general comparison explanation.
1. An example that requires a numeric computation.

2. The marked price was defined in the ontology, but the net price was computed outside Disciple and entered as a number in modeling.
During learning we need to explain how the net price was computed such that Disciple will be able to automatically do it in future situations.
Accept all the relevant explanations proposed by the system.
Click on “Edit Expression” to specify how the net price was computed.
In the left editor type the result of the computation (the net price).

retail store in Virginia is 449.19 dollars.
1. In the right editor type the expression used to compute the net price.

2. The following operators can be used: +, -, *, /, log(), log10(), sqrt(), exp() and pow()

3. Click “OK” when done.
1. The new explanation was automatically accepted.

2. Click “Finish” when done.
Disciple learned a rule with a general function in the condition.

<table>
<thead>
<tr>
<th>Var</th>
<th>Lower Bound</th>
<th>Upper Bound</th>
</tr>
</thead>
<tbody>
<tr>
<td>$?N1$</td>
<td>[449.19, 449.19]</td>
<td>(-infinity, +infinity)</td>
</tr>
<tr>
<td>$?N2$</td>
<td>[427.8, 427.8]</td>
<td>[0.0, +infinity)</td>
</tr>
<tr>
<td>$?N3$</td>
<td>[0.05, 0.05]</td>
<td>[0.0, +infinity)</td>
</tr>
<tr>
<td>$?O1$</td>
<td>(digital camera)</td>
<td>(digital camera)</td>
</tr>
<tr>
<td>$?O2$</td>
<td>(state)</td>
<td>(state)</td>
</tr>
</tbody>
</table>

**THEN:** The net price of a $?O1$ when bought at a retail store in $?O2$ is $?N1$ dollars.
Disciple will generalize the same number that appears in several places in a modeling example to a unique variable in the rule.

Therefore, an expression of the form $427.8 = 427.8 \times 0 + 427.8$ (when there is no applicable tax) will be generalized to $?N1 = ?N1 \times ?N2 + ?N1$

Such a rule will not be applicable when the tax is greater than 0!
These Lecture Notes *(required).*

Tecuci G., Boicu M., Boicu C., Marcu D., Stanescu B., Barbulescu M., The Disciple-RKF Learning and Reasoning Agent, Computational Intelligence, Volume 21, Number 4, 2005, pp 1-15 *(required).*

Tecuci G., Boicu M., Boicu C., Marcu D., Boicu C., Barbulescu M., Ayers C., Cammons D., Cognitive Assistants for Analysts, 2007 *(required).*


Tecuci, Building Intelligent Agents, Ch. 4 pp. 79-100 *(rule learning in Disciple).*