PROCEEDINGS OF THE INTERNATIONAL CONFERENCE
ON FIFTH GENERATION COMPUTER SYSTEMS 1988,
edited by ICOT. © ICOT, 1988

KNOWLEDGE ACQUISITION BY OBSERVATION

Hirokazu TAKI

ICOT Research Center, Institute for New Generation Computer Technology,
4-28, Mita 1-Chome, Minato-ku, Tokyo 108 Japan
Phone: +81-3-456-3192, Csnet:htaki%icot.jp@relay.cs.net
Junet: htaki@icot.junet

ABSTRACT

This paper describes a learning method for building knowledge bases. There are two types of knowledge acquisition systems that extract knowledge from human experts: interactive and non-interactive. This paper describes a non-interactive knowledge acquisition system that acquires knowledge from a human expert by observation. It learns strategies that the human expert uses to solve problems and makes logical rules from temporal sequential data. The learning method of the knowledge acquisition system is interpretation based learning (IBL), which uses advance knowledge in the learning process. The IBL has two subsystems: an interpretation system and a learning system. The interpretation system translates real world information to internal rule form. The rule maintenance system generalizes and specializes knowledge. In this paper, the interpretation system and pre-processes of the rule generalization are introduced.

1 INTRODUCTION

One major problem in building expert systems is removal of the knowledge acquisition bottleneck. Knowledge acquisition systems, usually interactive, have been developed to solve this problem. Each interactive knowledge acquisition systems has an interview sub-system that can access a human expert directly to ask for necessary information about the job. This type of system is called an active knowledge acquisition system (AKAS)[Boose 84][Boose 87][Taki 87][Kahn 85]. There are many cases or situations in knowledge acquisition environments. Sometimes, the human expert is too busy to answer questions that are asked by the interview system. In this case, knowledge is acquired by observation only, by a passive knowledge acquisition system (PKAS)[Taki 88a]. This type of system cannot ask the human expert any questions. The AKAS obtains symbolic data interactively from the human expert, data that can be translated into internal form easily. But the PKAS obtains both symbolic data and numerical data, so it must extract the numerical data and translate it into symbolic data. The PKAS must build a knowledge base inductively from observations only. Most inductive learning systems treat examples, which are represented in their internal symbolic form. The systems require many positive and negative examples. However, most examples that can be obtained from observation of human expert operations are positive. One learning system acquires knowledge from positive examples only: the explanation based learning (EBL) system [Mitchell 85][Mitchell 86], which extracts knowledge effectively using advance knowledge: domain theory and operationality. The EBL learns goal concepts, which are constructed according to the domain theory. However, the EBL treats symbolic examples which are deduced from a domain theory, but it does not treat real world information directly. The PKAS must treat real world information and have an effective learning mechanism. We are developing a PKAS, called interpretation based learning (IBL) that uses advance knowledge. The IBL has two subsystems: an interpretation system and a learning system. The interpretation system translates real world information to internal rule form. The learning system generalizes and specializes knowledge. The following sections discuss the interpretation system, its advance knowledge, and pre-processes for the rule generalization. This paper does not deal with details of the rule maintenance system[Taki 88c].

2 IBL OVERVIEW

In this section, the characteristics of the observation and the IBL framework are shown.

2.1 Observation Environment

The IBL can observe the actions of human experts and the situations in which those actions occur as shown in Figure 1. Normally, this symbolic information is translated from data extracted by sensors. Therefore, the IBL must be able to interpret the sensed data as internal symbolic representation data. Knowledge of an expert is formed into rules from situation and action information.

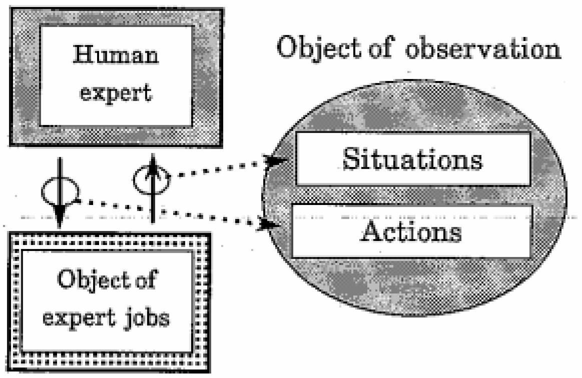


Figure 1 Observation overview

2.2 Interpretation and Learning System

The IBL system consists of two subsystems: an interpretation system and a learning system. The interpretation system interpretes real world information into an internal knowledge form according to advance knowledge. As a natural language processing system uses its dictionary to understand natural language sentences, so the interpretation system uses advance knowledge to translate real world knowledge into its knowledge representation. The learning system consists of a pre-generalization system and a rule maintenance system. The pre-generalization system eliminates noisy information from acquired knowledge according to advance knowledge. The rule maintenance system specializes rules in order to eliminate rule contradictions, and generalizes rules by induction. Figure 2 shows the interpretation system and the learning system.

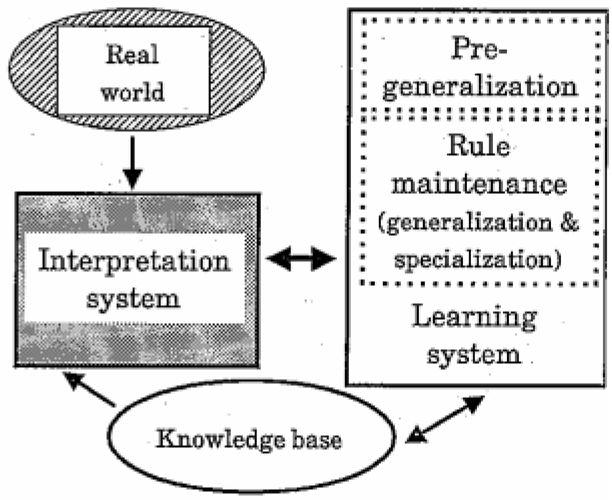


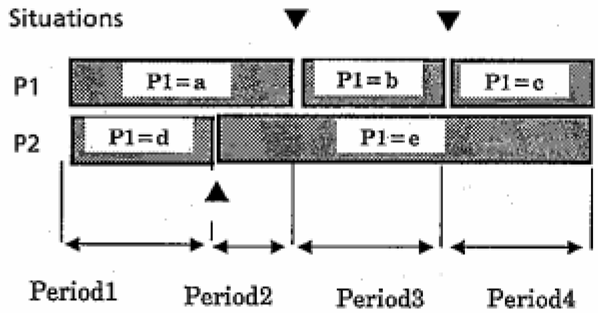
Figure 2 IBL system overview

3 OBSERVATION PROBLEMS

3.1 Interpretation Problems

(1) Problem of dividing sensed data

Sensed data is continuously collected at every sampling or when a sampling trigger is detected. Sensed data is temporal sequential data. To symbolize series data, the IBL divides the data into parts. If sensed data contains some ambiguity, there are many ways of dividing it. Therefore, the IBL must have knowledge to reduce the number of alternatives. The results of dividing data must be matched with internal symbolic concepts. The cause of ambiguity in sensed data is sensor capacity. The sensor has a limited capacity to detect and it detects unwanted noise. Figure 3 shows how to make situation data. In Figure 3, parameter 1 is divided into three parts. Parameter 1 has three values (a, b and c). If parameter 1 changes critically in these three values, it is easy to divide it. However, generally, parameter 1 does not always change in steps (it can be a middle value between a and b) but slides from one value to another continuously. Therefore, it is difficult to decide the points of change of parameter 1. If more detailed changes are considered, parameter 1 is divided into more parts, and the IBL obtains more detailed situation information. In this case, the IBL must be able to handle many concepts related to dividing criteria; it must have knowledge that divides sensed data into useful level granules corresponding to internal concepts.



$$Situation1 = (P1 = \mathbf{a}, P2 = \mathbf{d})$$
$$Situation2 = (P1 = \mathbf{a}, P2 = \mathbf{e})$$
$$Situation3 = (P1 = \mathbf{b}, P2 = \mathbf{e})$$
$$Situation4 = (P1 = \mathbf{c}, P2 = \mathbf{e})$$
$$Situation\mathbf{i} = (P1 = p1\mathbf{i}, P2 = p2\mathbf{i}, \partial P1 = \partial p1\mathbf{i}, \int P1 = \int p1\mathbf{i}, \ldots..)$$

Figure 3 Data division and situation generation

(2) Problem of symbolizing divided data

Normally, fragments of sensed data are translated into two kinds of information : symbols and parameters with values. A parameter consists of a parameter name and its value. Generalized parametric representation has a range of its value instead of an instance value.

Example 1: The human expert measured register 5 with voltage-tester 1. The tester detected 3 mV. The expert changed register 5.

Symbolic data: Voltage-tester = voltage-tester 1,
Register = register 5
Parametric data: Voltage = 3 mV

Rule-Expression :
use(Voltage-tester = voltage-tester1),
detect(Voltage = 3mV, Register = register5)
→ change(Register = register5).

To symbolize the sensed data, the IBL must match real data with internal symbolic concepts. In example 1, voltage-tester1 matches the concept "Voltage-tester", register5 matches the concept "Register", and the real voltage matches "Voltage". In this case, the IBL contains concepts of "Voltage-tester", "Register" and "Voltage". If the IBL has only concepts of "Tester", "Device" and "no-voltage(-5 mV $\leqq$ no-voltage $\leqq$ 5 mV)", the symbolic expression is changed as follows:

Example 2:
Symbolic data: Tester = voltage-tester 1,
Device = register 5, no-voltage (= 3 mV)

Rule-Expression :
use(Tester = voltage-tester1),
detect(no-voltage, Device = register 5)
→ change(Device = register5).

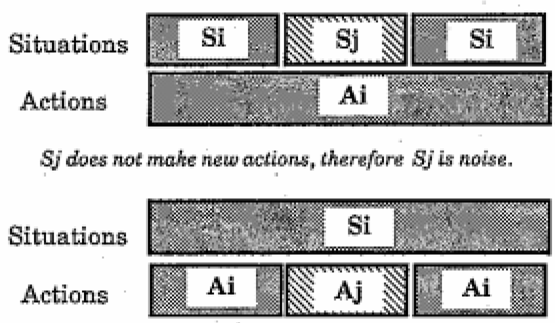
Examples 1 and 2 show different interpretation results under different concept sets. The IBL must have appropriate concept sets of the target domain. Generally, a concept consists of some sub-concepts. In Figure 3, a situation contains two parameters. There is other information in this example, combination information of temporal variable data, which can be thought of as differentiation and integration information. The necessity for higher-order differentiation depends on the target domain. The IBL must have internal symbolic concepts, internal concept sets, and internal parametric definitions as advance knowledge.

3.2 Rule Generalization Problems

Sometimes, there are ambiguities and noise (useless information) in the sensed data. The IBL must be able to handle various meanings in the ambiguities in building a knowledge base. Generally, this noise is very harmful. It makes acquired knowledge too specific.

(1) Problem of combining situations and actions

At certain times, there is some causality between situations and actions in the human expert's tasks. The IBL makes rules from these situations and actions. However, there is some noise in these rules. The IBL must select appropriate situations and actions, and must combine them carefully. Because normally there is a time delay in the causality. Figure 4 shows noise reduction examples. The first example has situation noise. The situation changed, but the action did not change, and the situation returned to its original state. Therefore, Sj must be noise. In the same way, the second example shows action noise. Aj may be noise. The IBL must have a noise reduction mechanism using noise detection heruristics.



Sj does not make new actions, therefore Sj is noise.

Aj may be noise, because Si is the same and Aj returns to Ai.

Figure 4 IBL system noise reduction

(2) Problem of eliminating unnecessary information
The IBL observes all situations at the same time, so it has special rules for all observed situations. The IBL must have a function which chooses only suitable situations related to actions. Example 3 is a special rule with an unnecessary situation.

Example 3:
Situation information:
{It rained, and
the output voltage of the amplifier was too low.}
Action information:
{An expert changed an output transistor.}

Generated rule:
(Weather = rain)
& (Amp-output-voltage = low)
→ (Change Amp-output-transistor)

This result is too specific to be used in real amplifier maintenance, because the weather is not related to amplifier maintenance. Therefore, the IBL must choose situations related to actions. It has to make the following rule (example 4).

Example 4:
Generated rule:
(Amp-output-voltage = low)
→ (Change Amp-output-transistor)

Generated rules are checked with domain knowledge which contains relations between situations and actions in a target domain. This method is a sort of pre-generalization from the point of view of generalizing situations. The IBL must have symbolic concept relations to make appropriate rules.

(3) Problem of rule maintenence
Generally, a learning system obtains general knowledge from more special instances by induction. The rule maintenance system controls the generalization level to keep rules from over-generalizing. It also maintains conflicting rule pairs that implicate inconsistent results. If the concept set as advance knowledge is not enough to interprete real world information correctly, the rule maintenance system cannot maintain rules. In this case, the rule maintenance system must reconstruct a new concept set and new translation knowledge for interpretation. The interpretation system interprets old instances again.

4 INTERPRETATION & LEARNING

This section describes an interpretation-based learning system and explains the learning flow and mechanism.

4.1 Learning Input and Output

Examples are given as samples of an expert's jobs. They are temporal sequential data. They contain the problem-solving strategy knowledge of the expert. The IBL learns problem-solving rules. The following examples show input and output.

Example 5: *Input contents*
Sensing parameters at time t0: p1(t0), p2(t0), ..., pn(t0)
Values of the parameter: numerical data, symbol or logical values.

Example 6: *Output contents*
Implication rules: S1&S2&....Sj → a1&a2&...am
An expression Si(i=1, ..., j) is a variable with a

range.
An expression ai(i = 1, ..., m) is a function with one or more variables.
The variables of the action part are shown as Ai(i = 1, ..., k).

Variable and range:

The values of variables (Si/Aj) are numerical values, symbols, or logical values. The variation of the range of a variable, V, is shown as follows:

Equality: V = number/symbol/logical values
(For examples, true/false)
Upper limit: $V \leqq$ number-1
Lower limit: $V \geqq$ number-2
Upper and lower limits:
number-2 $\leqq V \leqq$ number-1
A sub-set: $V \subseteq$ {symbol-1, symbol-2, ...}

4.2 Learning Strategy

There are six learning steps in the IBL. Figure 5 shows an overview of this flow and advance knowledge.

Step 1: Dividing sensed data
Sensed data consists of many parameters. Each parameter has temporal variable values. The IBL checks the value change of each parameter, and divides data in the time scale.

Step 2: Matching sensed data with internal concepts
Here, data is separated into symbolic concepts and parameter instances. Symbolic concepts are set as situations and actions.

Step 3: Reducing noise in situations and actions
There are some relations between situations and actions. Therefore, action data that is independent of situations must be noise. In the same way, situation data that is independent of actions must also be noise.

Step 4: Making symbolic rules
Rules are made to combine situations and actions. A rule consists of an "if-part" and a "then-part". Situations match the if-part, and actions match the then-part. Sometimes, generated rules also have useless information as example 1 shows. So the relationships between situations and actions in all rules must be checked, and unnecessary situations or actions must be removed.

Step 5: Optimizing values of parameters
A parameter has an instance value and a range of its value. This instance value is collected from sensed data. It is only one example, so it must be generalized and optimized to chang it into the mean or typical value.

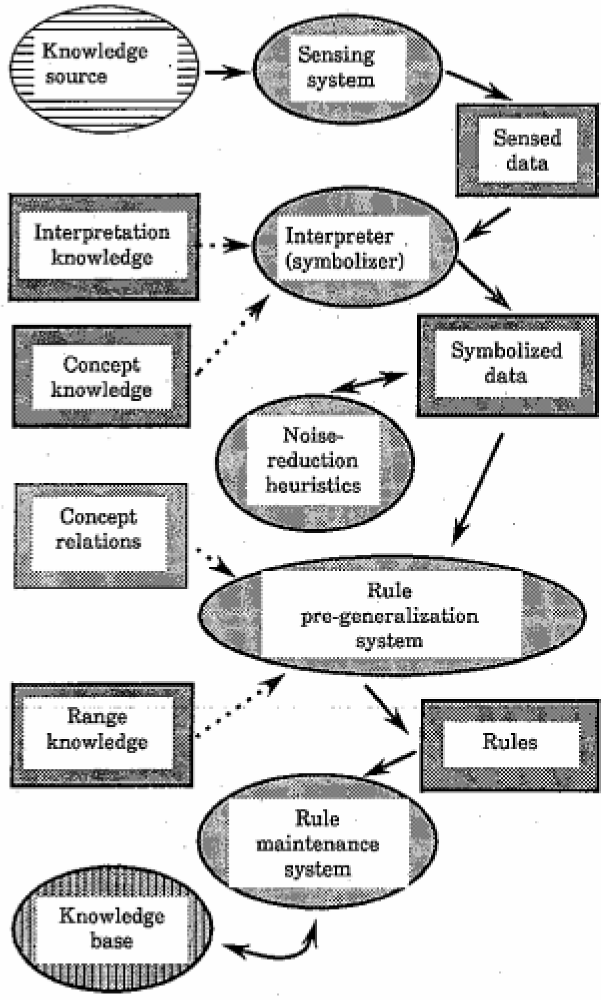


Figure 5 IBL system structure

Step 6: Generalizing rules and parameter data by multiple examples
The IBL learns rules and parameter ranges from step 1 to step 5, at which it obtains one example. The IBL acquires other knowledge from other examples, then checks and compares rules with the same form. If their actions are the same, the two situations are reformed into a more general situation. The ranges of parameters are also generalized. For example, a parameter consists of

"Voltage" as a symbolic name and "0 V $\leqq$ Voltage $\leqq$ 15 V" as the range of its value. A new example brings the IBL a new range of its value, that is, "3 V $\leqq$ Voltage $\leqq$ 20 V". IBL makes a new parameter which contains "Voltage" as the name and "0 V $\leqq$ Voltage $\leqq$ 20 V" as the range of its value.

5 ADVANCE KNOWLEDGE

One of the most important components of learning systems is advance knowledge as a concept bias and a background theory. Advance knowledge controls the learning flow; it limits and stimulates the knowledge acquisition system to induce knowledge from examples. In the EBL, there are two types of advance knowledge: domain theory and operationality criteria. Domain knowledge attempts to explain the examples. If an example is implied from the domain knowledge, it is explained and the EBL recognizes it as a positive example. An example is given to the EBL system as a goal concept, so it learns how to construct the goal concept from domain knowledge. Operationality knowledge controls the generalization level of explained knowledge. It limits generalization of that knowledge. There are two learning steps in the EBL. The first step is the explanation step to check whether an example is positive or not, and the second step is the generalization step to generalize knowledge. The IBL uses advance knowledge as a dictionary to translate real world knowledge, to generalize acquired rules , and to specialize ranges of values in these rules.

5.1 Domain Concept Knowledge

One type of domain concept knowledge is atom level concepts and relations between them. Atom level concepts mean symbolized situations and actions, parameter expressions, and ranges of the parameter values. Another type of domain concept knowledge is relation knowledge, which contains relations between symbolic concepts. Each concept has a range of its value. This information is used for parameter generalization and optimization, it is a generalization limit. The range depends on the target domain.

Example 7: *Symbolic concepts*
Symbolic concepts: register5, capacitor3
Parameter expressions: Voltage, Time-delay

Example 8: *Range knowledge of concepts*
Parameter range: 0 V $\leqq$ Voltage $\leqq$ 12 V,
3 mA $\leqq$ Ampere 8 $\leqq$ 1 A.
In logic circuits, the voltage range is from 0 V to 5 V. This range is 0 V or 5 V in the logical meaning.

5.2 Concept Relation Knowledge

Relations between concepts may be positive (such as same class concepts and positive relativity), negative (such as contrary relativity), no relations, or equations.

Example 9: . *Concept relations* (about force feedback robot control)
Positive relations:
pair (Movement direction, Velocity vector) in position control
Negative relations:
pair (X-axis velocity, X-axis pressure)

Note: If the robot's grip touches a wall, a tactile sensor detects pressure in the opposite direction to which the sensor is moving.

No relations: pair (X-axis velocity, Y-axis pressure)
Equations: Velocity 3 = Velocity 5 / 2.0

5.3 Interpretation Knowledge

Interpretation knowledge is used for translating sensed data into symbolic concepts and parameters. It also contains dividing knowledge for sensed data because divided data must be matched with internal concepts.

Example 10: *Dividing knowledge:*
IF $|$ p1(ti)-p1(ti+1) $| \geqq$ e1,
THEN divide parameter p1 at ti.
IF $|$ p1(ti)-p1(tj) $| \geqq$ g1,
THEN divide parameter p1 at tj-1.
e1 and g1 are special knowledge for dividing data.

Symbolizing knowledge (translation knowledge):
IF f1 $\geqq$ p1(from ti to tj) $\geqq$ f2,
THEN p1(from ti to tj) is a concept, "X".
IF p1(from ti to tj) = f3,
THEN p1(from ti to tj) is a concept, "Y".
The range of "X" is from f1 to f2. The value of "Y" is f3.

6 PRE-GENERALIZATION

This section describes how to make and optimize rules for pre-processing of learning. It describes the induction method, noise reduction and relation check.

6.1 Rule Generation

Situations and actions are extracted each time. They are represented by symbolic expressions and parameters. The IBL makes implication rule sets from situations and actions to select a good set of situations as Figure 6 shows. Temporal information shows a sequential rule evaluation flow. Rule(ti), which is made from situations and actions that occurred at time ti, makes a new environment which matches situations of rule(ti+1). Therefore, the IBL adds situations made by actions of rule(ti) to the situations of rule(ti+1) shown in example 11.

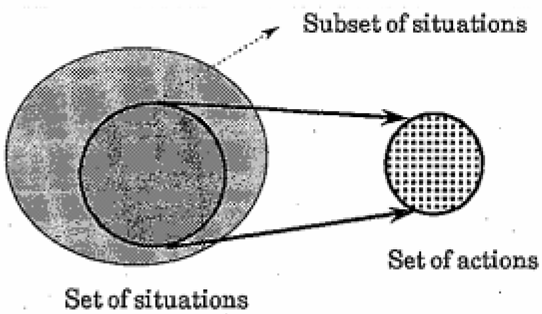


Figure 6 IBL induction

Example 11: . *Rule generation considering temporal information*

Situations: S1, S2 and S3 are observed at time ti+1.

S1 = symbol-1, 0 ≦ S2 ≦ 15 and S3 = symbol-2.

Actions: a1 is done by the human expert at time ti+1.

The parameter of "a1" is A1, and A1 = 20.

Action of rule(ti): a2 is done.

Generated rule(ti+1):

S1& S2& S3 & side-effect of a2 → a1(A1 = 20),

in context (S1 = symbol-1, 0 ≦ S2 ≦ 15 and S3 = symbol-2)

6.2 Noise Reduction

Real noise is caused by sensors and errors made by human experts. This noise must be removed as it is unnecessary data in expert jobs. For example, in spite of a sensor detecting a situation, a human expert sometimes does not react to that situation. That situation information is useless data. The IBL detects this noise as shown in Figure 4. Situations and actions are causally connected, so data that have no causality must be removed.

6.3 Concept Relation Check

As shown in example 3, sensed data contains most concepts of the target domain. Therefore, generated rules contain unnecessary situations in their "if-part". Each situation must have some causality which depends on the target domain; this causality is dealt with as the concept relation knowledge. The IBL uses this concept relation knowledge to reduce the amount of unnecessary information.

6.4 Generalization and Specification

In one learning process, only parameters are generalized or optimized. However, structures of rules are not generalized in one observation, but by multiple examples.

(1) Specialization for ranges of parameters

Range expressions are shown in example 12. They show the generalization criteria. Strictly speaking, range information contains a lower case and an upper case. The lower case is used for parameter generalization and the upper case for parameter specialization.

Example 12: *Range specialization*

Lower case (narrow range): 3 ≦ V ≦ 4
Upper case (wide range) : 1 ≦ V ≦ 5
Acquired range: 0.5 ≦ V ≦ 3.5
Optimum range: 1 ≦ V ≦ 4

The lower limit of value "V" must be more than 1 and less than 3; therefore, the acquired range is changed to "1 ≦ V ≦ 3.5". The higher limit of value

"V" must be more than 4 and less than 5, so the acquired range is translated into "$1 \leqq V \leqq 4$".

If an acquired range is within the limits of a lower case, it must be rewritten as a lower case. If it is beyond the limits of an upper case, it must be rewritten as an upper case. A range of an instance is generalized or specialized in order to fit it into a range between the upper case and the lower case. It becomes an optimized range as shown in Figure 7.

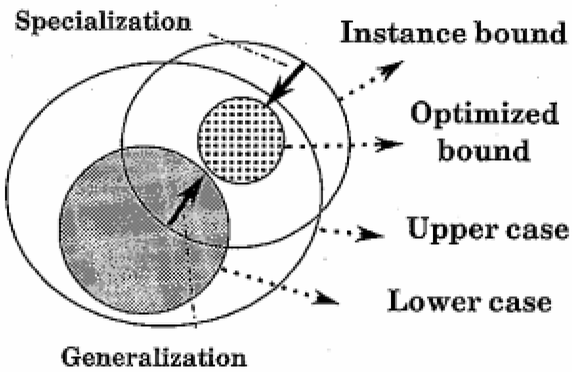


Figure 7 Range optimization

(2) Generalization by multiple examples
There are many rules in one expert task. However, general rules and special rules are mixed in the task. Taking other expert tasks into consideration, some of the same rules are extracted. Both old and new rules have some differences from each other. To use these differences, rules can be generalized. We explain the generalization of the IBL for each difference.

Case 1: There are no symbolic differences in the if-parts between new and old rules, and each then-part is the same. However, the values of the new parameters of the if-parts are different from old ones. The IBL generalizes the ranges of situation parameters.

Case 2: There are symbolic differences in the if-parts between new and old rules, but each if-part is the same. The IBL applies the logical-OR operator to these if-parts and makes a new rule.

Case 3: There is no difference in the if-parts but the new then-part is different from the old one. The IBL applies the logical-AND operator to these then-parts and makes a new rule.

(3) Rule maintenance
Some causes lead toinconsistent situations in a knowledge base. Sometimes, the human expert makes mis-operations. Local errors of these mis-operations can be eliminated by the noise reduction heuristics. In order to detect non-local errors include backtracking processes, the learning system searches the expert's recovering processes. Then it eliminates non-appropriate knowledge made from mis-operations. If advance knowledge is not enough to interpret the real world, then non-correct knowledge may be acquired. In this case, the rule maintenence system does not always maintain rules easily; it reforms the new concept set and new translation knowledge. The interpretation system must re-interpretat.

7 OBJECT MODEL

Human expert knowledge is learned from situations and actions inductively. A form of acquired knowledge is an implication rule (situations $\rightarrow$ actions). These actions are made by the human expert according to situations. These actions lead to next situations in an object of the expert job. Therefore, an implication form (actions $\rightarrow$ next situations) represents a sub-model of the object. The IBL can also obtain the sub-model of the object. If a detail model (such as deep knowledge) of the object is given, we can know a coverage of acquired knowledge to compare the detail model and the sub-model.

8 CONCLUSIONS

The IBL learns the human expert's problem solving knowledge by observation. It consists of the interpretation system and the learning system. In this paper, this interpretation system, its advance knowledge and the pre-generalization mechanism as a sub-system of the learning system are described. The IBL system acquires knowledge in logical form and the range information of the values in logical rules. It cannot obtain general rules from one observation, but it has a function which optimizes parameter ranges. In order to acquire general knowledge, multiple task examples are given to this system. A subset of its functions was developed for a robot skill acquisition system [Taki 85], and it was proved that the major functions of this system are useful for skill acquisition by observation in that system. We believe that it is also useful to extract not only the skills but also the knowledge of human experts. This paper does not deal with the treatment of alternative interpretations (translations) and the rule maintenance mechanism. The TMS [Doyle 79] and ATMS[de Kleer 86] mechanisms are useful to maintain the acquired rule base. Acquired knowledge is a logical form, so the partial evaluation techniques in logic programming [Fujita 87] are useful for these rules to reform effective rule sets.

ACKNOWLEDGEMENTS

I would like to thank Mr. Yuichi Fujii, the chief of the fifth laboratory at ICOT, who helped me to further knowledge acquisition research. I wish to express my thanks to Dr. Kazuhiro Fuchi, Director of the ICOT Research Center, who provided me with the opportunity of doing this research in the Fifth Generation Computer Systems Project.

REFERENCES

[Boose 84] Boose, J., Personal Construct Theory and the Transfer of Human Expertise, *Proceedings of the National Conference on Artificial Intelligence, Austin, Texas*, 1984

[Boose 87] Boose, J., Expertise transfer and complex problems: using AQUINAS as a knowledge-acquisition workbench for knowledge-based systems, *Int. J. Man-Machine Studies 26*, pp3-28, 1987

[de Kleer 86] de Kleer, J., An Assumption-based TMS, *Artificial Intelligence 28*, pp127-162, 1986

[Doyle 79] Doyle, J., A Truth Maintenance System, *Artificial Intelligence 12*, pp231-272, 1979

[Fujita 87] Fujita, H. and Furukawa, K., A Self-Applicable Partial Evaluator and Its Use in Incremental Compilation, *New Generation Computing*, Vol.6, No.2, 1988

[Ginsberg 87] Ginsberg, A., A New Approach to Checking Knowledge Bases for Inconsistency and redundancy, *Proceedings of IEEE 3rd Annual Expert Systems in Government Conference, Washington, D.C.*, pp102-111, 1987

[Kahn 85] Kahn, G., Nowlan, S., and McDermott, J., Strategies for Knowledge Acquiqsition, *IEEE Transactions on Pattern Analysis and Machine Intelligence 7(5)*, 1985

[Mitchell 85] Mitchell, T., Mahadevan, S., and Steinberg, L., LEAP: A Learning Apprentice for VLSI Design, *Proceedings of the 9th IJCAI, Los Angeles*, 1985

[Mitchell 86] Mitchell, T., Keller, R. and Kedar-Cabelli, S., Explanation-Based Generalization: A Unifying View, *Machine Learning 1*, January 1986

[Taki 85] Taki, H. and Sakaue, Y., A Programming System for Intelligent Robots with Force Feedback, *ROBOTS 9 Conference Proceedings (SME)*, Vol. 1, 1985

[Taki 87] Taki, H., Tsubaki, K., and Iwashita, Y., EXPERT MODEL for Knowledge Acquisition, *Proceedings of IEEE 3rd Annual Expert Systems in Government Conference, Washington, D.C.*, pp117-124, 1987

[Taki 88a] Taki, H., Interpretation-Based Knowledge Acquisition, *Report of Information Processing Society of Japan, 88-AI-56-5*, 1988 (in Japanese)

[Taki 88b] Taki, H., Knowledge Acquisition for Protocol Analysis, *Report of Information Processing Society of Japan, 88-AI-59-11*, 1988

[Taki 88c] Taki, H., Rule Maintenance = Generalization + Specialization, *ICOT Technical Reports , 1988 (to appear)*