LEARNING FLEXIBLE CONCEPTS: FUNDAMENTAL IDEAS AND A METHOD BASED ON TWO-TIERED REPRESENTATION

by

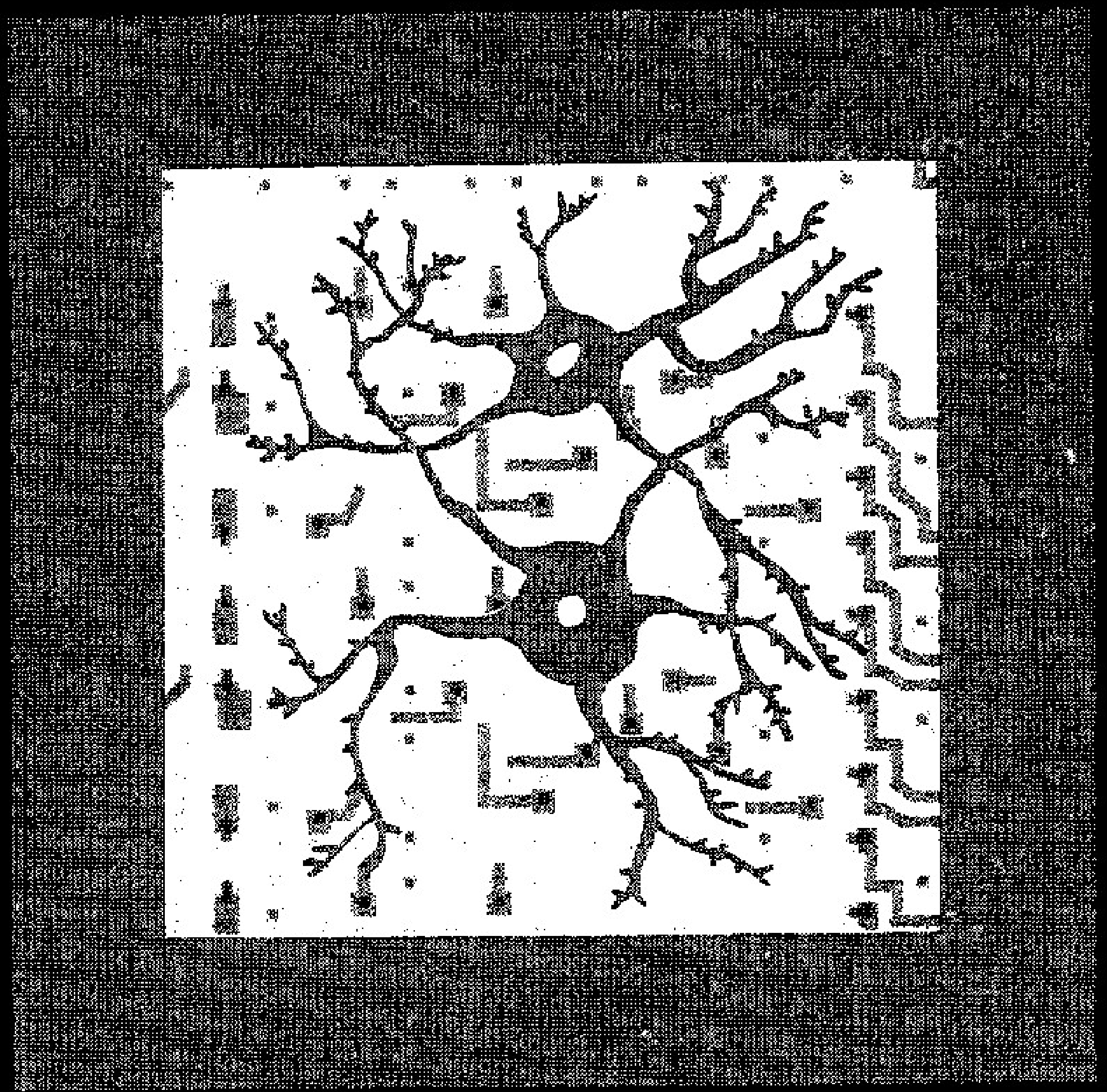
R. S. Michalski

Machine Learning: An Artificial Intelligence Approach Vol. III, Y. Kodratoff and
R. S. Michalski (Eds.), San Mateo, CA, pp. 63-111, Morgan Kaufmann Publishers,
June 1990.

MACHINE LEARNING

An Artificial Intelligence Approach

Volume III



Yves Kodratoff
Ryszard Michalski

MACHINE LEARNING

An Artificial Intelligence Approach
Volume III

Contributors:

E. Ray Bareiss
Francesco Bergadano
Pavel B. Brazdil
Jaime G. Carbonell
Gerald F. DeJong
Kenneth A. De Jong
Brian C. Falkenhainer
Jean-Gabriel Ganascia
Yolanda Gil
Attilio Giordana
Stephen J. Hanson
David Haussler
Geoffrey I. Hinton
Haym Hirsh
Robert C. Holte
Alex Kass
Yves Kodratoff
Michael Lebowitz
Sridhar Mahadevan
Ryszard S. Michalski

Tom M. Mitchell
Michael J. Pazzani
Bruce W. Porter
Armand E. Prieditis
J. Ross Quinlan
Larry A. Rendell
Ronald L. Rivest
Roger Schank
Robert E. Schapire
Jude W. Shavlik
Pawel A. Stefanski
Louis I. Stenberg
Robert E. Stepp
Gheorghe Tecuci
Christel Vrain
Craig C. Wier
David C. Wilkins
Janusz Wnek
Zianping Zhang

Editors:

Yves Kodratoff
French National Research Center
and George Mason University

Ryszard S. Michalski
George Mason University

MORGAN KAUFMANN PUBLISHERS, INC.
SAN MATEO, CALIFORNIA

Editor and Publisher: *Michael B. Morgan*
Production Manager: *Shirley Jowell*
Project Management: *Jennifer Ballentine*
Cover Design: *Andrea Hendricks*
Electronic Composition: *Ocean View Technical Publications*

The publisher gratefully acknowledges permission to reproduce the following figures: 20–4, 20–5, 20–6 reprinted with permission from "Learning Internal Representations by Error Propagation," *Nature*, Vol. 323, pp. 533–536. Copyright © 1986, MacMillan Magazines, Ltd.; Figure 23–3, adapted from *The Little Prince*, by Antoine de Saint-Exupéry, © 1943 and renewed 1971, by Harcourt, Brace, Jovanovich, Inc., reprinted with permission of the publisher.

Library of Congress Cataloging-in-Publication Data
(Revised for vol. 3)

Machine learning.

Vol. [1] previously published: Palo Alto, Calif. : Tioga Pub. Co., c1983.
Vol. 3- edited by Yves Kodratoff and Ryszard Michalski.
Includes bibliographies and indexes.
1. Machine learning. 2. Artificial intelligence.
1. Anderson, John R. (John Robert), 1947–
II. Michalski, Ryszard Stanislaw, 1937–
III. Carbonell, Jaime G. (Jaime Guillermo)
IV. Mitchell, Tom M. (Tom Mitchell), 1951–
Q325.M32 1983b 006.3'1 86-2953
ISBN 0-934613-09-5 (v. 1)
Vol. 3 ISBN 1-55860-119-8

Printed in the United States of America

10 9 8 7 6 5 4 3 2 1

MORGAN KAUFMANN PUBLISHERS, INC.
Editorial Office
2929 Campus Drive, Suite 260
San Mateo, CA 94403
415-578-9911

Order From
P.O. Box 50490
Palo Alto, CA 94303-9953
415-965-4081

3

LEARNING FLEXIBLE CONCEPTS:

Fundamental Ideas and a Method Based on Two-Tiered Representation

Ryszard S. Michalski
(George Mason University)

Abstract

Most human concepts elude precise definition—they have fluid boundaries and context-dependent meaning. We call such concepts *flexible*, in contrast to *crisp* concepts, which are well defined and context independent. As machine learning research has concentrated primarily on learning crisp concepts, learning flexible concepts emerges as a new challenge to the field and an important research direction.

This chapter describes an approach to learning flexible concepts based on a *two-tiered* concept representation. In such a representation, the concept meaning is defined by two components: the *base concept representation* (BCR), and the *inferential concept interpretation* (ICI). The BCR (the first tier) is an *explicit* description of basic concept properties, while the ICI (the second tier) characterizes allowed modifications of the concept meaning and its possible variations in different contexts. Thus, the ICI defines concept boundaries *implicitly*, by the results of matching procedures and inference processes. The latter can be deductive, analogical or inductive.

In the method described, the initial BCR is a complete and consistent concept description, induced from concept examples by a conventional AQ inductive learning program (AQ15). This description is then simplified by the so-called TRUNC procedure, to maximize a *description quality measure*. The so-obtained BCR is usually much simpler than the initial description, but in a strict, logical sense is incomplete with regard to the training examples. The ICI is implemented in the form of a procedure for *flexible matching*, which determines a degree to which instance matches different candidate concepts and chooses the concept that makes the best

match. Due to this procedure, training examples that have been "uncovered" durir the description-reduction process may still be classified correctly.

The method has been implemented in the learning system AQTT-15, and e: perimentally applied to learning diagnostic rules in a sample of medical domain Experiments have shown that the method may produce more significantly reduce concept representations than the traditional approach and that these representatio may also perform better in recognizing new concept examples. This surprising ar potentially significant result calls for further research and new experiments. In pa ticular, the method should be tested on other problems and in different domain Other interesting topics for future work include the development of a "direc method for learning two-tiered representations, an extension of the form of such re resentations, acquiring the second tier of descriptions through examples, and the d velopment of techniques for learning hierarchically organized two-tiered represent tions.

> *We have no sound notions either in logic or physics; substance, quality, actic passion, and existence are not clear notions...*
>
> Sir Francis Bacon
> *Novum Organum*, First Book, Chapter 15, 1620

3.1 INTRODUCTION

Most machine learning research done so far has focused primarily on learni *crisp* concepts, by which we mean concepts that have precise and context-indepe dent meaning. Such concepts are usually represented by explicit descriptions, whi are either satisfied or not satisfied by any given instance. Popular representations crisp concepts include logical expressions, production rules, semantic networks, c cision trees, and frames. For relevant references see, for example, Volumes I and of *Machine Learning* [Michalski, Carbonell, and Mitchell, 1983 and 1986]. The te dency to use crisp concepts is characteristic of not only machine learning resear but every scientific activity. The clarity and precision of concepts and of their int relationships has traditionally been, and remains, a mark of quality of scientific the ries. Crisp concepts enable us to reason precisely and derive strong conclusions.

Yet, most human concepts used for characterizing real-world objects and e ployed in communication are *flexible*—as they have fluid and modifiable bour aries, and their meaning often depends on the context of discourse. Note how dil cult it is to define precisely and in a context-independent fashion such concepts "chair," "music," "key," "space," "game," "freedom," or "mechanism," which : frequently used in conversations. To make machine learning programs more applic ble to real-world problems, it is crucial to make them able to learn flexible concep The key problem in learning such concepts lies in the difficulty of accounting, for their possible forms, permissible modifications and context dependence. Developi methods for representing and learning flexible concepts thus represents a fundamental new challenge to the field.

Some researchers view the imprecision and flexibility of human concepts as some fault of our language or an imperfection of our mind. In our view, these properties are a consequence of the necessity to cope efficiently with the complexity of our world. As discussed in [Michalski, 1988b], flexible concepts are a powerful means of increasing cognitive economy of our descriptions.

One evidence of this is that in an abstract, simplified world created by our imagination, concepts typically have a precise, well-defined meaning. But once they leave this abstract world and are applied to the real world, these concepts acquire a flexible and context-dependent meaning. Consider, for example, the concept of a triangle. It has a well-defined meaning in geometry. But outside of geometry, the concept "triangle" becomes imprecise and highly context dependent. For example, it can be used to characterize a configuration of streets, a relationship among people, or the shape of a musical instrument. In all these usages certain core properties of the ideal, geometrical concept are preserved, but the specific meaning depends on the context in which it is used.

Moreover, even in the context of geometry, one can distinguish between more or less *typical* triangles. This means that there is a perceived distinction in the *representativeness* of different instances of a given concept. Consider, for example, the concept of a bird. A cardinal is viewed in the U.S. as a more typical bird than, say, an ostrich or a kiwi. In most machine learning programs, however, the distinction between degrees of typicality of different concept examples has been largely ignored. Among the few early exceptions from this general rule is, for example, the idea of *near miss* [Winston, 1975] or the method of *outstanding representatives* for selecting "best" learning examples [Michalski and Larson, 1978].

A related and also relatively unexplored issue involves the degree of precision and accuracy with which individual examples are presented to a learning system. For example, a triangle can be drawn in many different ways: with dotted lines, lines made of other shapes, or to appear as a shadow on an uneven surface. In all these cases, the form may still be recognizable as a triangle. Thus, concept examples may vary greatly in the ways they are presented and may be strongly distorted or modified. Nevertheless, they represent the same concept.

Finally, the complete concept meaning perceived by a person depends on the amount of knowledge this person possesses about it. Clearly, the conceptualization of a triangle by a layperson is different from that of a mathematician specializing in geometry. The difference lies in the number of facts they know about and in the depth of their understanding of the concept and its properties. Such background knowledge-dependency in understanding a concept indicates that human concepts are personalized, living and growing constructs, rather than fixed and stable entities that mean exactly the same thing to all those using them. As the meaning of concepts may change from individual to individual and evolve in time, such concepts cannot be defined precisely as objective impersonal entities with a context-independent meaning. Note, that even fundamental scientific concepts, such as energy, force, light, gravitation, atom and electricity, have been changing and evolving over time. Nonscientific concepts are almost universally flexible, rather than crisp. Thus, in general, human concepts are very different entities than the well-defined and context-independent structures we use to represent concepts in today's computer systems. Determining representations of human concepts that would account for all their possible manifestations, allowable modifications and a change of meaning in different contexts is a fundamentally difficult and unresolved problem.

This problem is not new, however, and there have been many attempts to solve it. One of the most widely known is the work on fuzzy sets by Zadeh and his collaborators and many followers (e.g., [Zadeh, 1965; 1976; 1978; Mumdani and Gaines, 1981]). This approach has concentrated primarily on representing the imprecision of concept boundaries and has proposed to associate with an imprecise concept a *set membership function* that defines the degree to which an instance represents the concept. This is usually a continuous numeric function, which expresses a subjective view of a person about the concept variability. One way to interpret the set membership function is to view it as a representation of the typicality of instances. It has been shown that such a function is useful for computationally representing the influence of linguistic modifiers, such as "very," "more or less," "slightly" on the meaning of concepts. The fuzzy set approach has been widely studied and has found a number of applications, in particular, in the control of complex systems.

This approach does not address, however, several issues relevant to representing flexible concepts. The membership function must be defined by a person and for every context; the approach does not offer methods for automatically deriving such a function. The membership function is usually defined as one argument function; it is difficult to characterize in this way concepts whose boundaries depend on many arguments. For example, it is relatively easy to define the membership function for the concept "tall," whose meaning depends on one argument, the numerical height (and on the context). It is much more difficult to define the membership function for *multiargument* concepts, such as "chair," or "heart condition." The fuzzy set approach does not seem to provide adequate mechanisms for capturing concept extensions, representing multiple but interrelated meanings of a concept, reasoning about the context dependence, or employing background knowledge for interpreting a concept. A set membership function is not sufficient for handling such problems.

In the cognitive science literature, the inadequacy of representing human concepts by context-independent, logic-style definitions (the *classical* view), has been widely recognized (e.g., [Wittgenstein, 1922; McCloskey and Glucksberg, 1978; Barsalou and Medin, 1986; and Lakoff, 1987]). There have been other views advanced, such as the *probabilistic view* and the *exemplar view* (e.g., [Smith and Medin, 1981; Medin and Smith, 1984; Nosofsky, 1987; Allen, *et al.*, 1988]).

The probabilistic view represents concepts by prototypes and uses the so-called *family resemblance principle* (e.g, [Rosch and Mervis, 1975]), while the exemplar view claims that concepts are represented by means of examples (e.g., [Smith and Medin, 1981; Bareiss, Porter, and Craig, 1990—Chapter 4, this volume]). Both views can be criticized on various grounds. The prototype view, which formally is based on the idea of linear separability, disregards the existence of correlations between the attributes, the context dependence, and other information that has been shown to be relevant to human concept understanding (e.g., surprisingly, [Kempler-Nelson, 1984; Estes, 1986; Flannagan, Fried, and Holyoke, 1986]).

The exemplar view promotes the idea of using similarity-based and context-sensitive matching; a view that has received support in the cognitive science literature. It ignores, however, the importance of general concept descriptions, that clearly play a role in human concept formation. Such general descriptions are useful, for example, for comparing different concepts, for recognizing them from partial information, for identifying exceptions, handling context dependence, recording concept changes or for efficiently storing invariant information about concepts. The above operations are difficult to perform, if concepts are represented only by examples. In some work using the exemplar view, general aspects of concepts are captured under the idea of *category structure*, which is a network of domain knowledge that specifies the relevance of exemplars to the concept they define [Bareiss, Porter, and Craig, 1990—Chapter 4, this volume]. Some recent work has advocated a *knowledge-based view*, which emphasizes the need to define concepts through their role in theories in which they exist as interelated components [Carey, 1985; Hofstadter, 1985; Schank, Collins, and Hunter, 1986; Medin, 1989].

The *two-tiered representation* (TT), employed in this chapter, constitutes a significant departure from the existing approaches, although it has a relationship to most of them. The TT approach assumes that concepts have a certain central tendency and proposes to describe this tendency explicitly, as the "first approximation" of the concept. On the other hand, it assumes that concepts' variability and context dependence are best represented implicitly, by appropriate matching methods and context- and background knowledge-dependent rules of inference.

Thus, in the sense that it recognizes that concepts have a central tendency, and that there are typical and less typical concept examples, the TT approach is similar to the probabilistic view and the fuzzy set representation. It has also a relationship to the exemplar view, as it postulates the use of sophisticated matching procedures and inference rules in classifying new instances, and recognizes the usefulness of storing individual examples (by advocating a full or partial memory learning [Reinke and Michalski, 1988]). The TT approach is also closely related to the knowledge-based view, as it stresses the role of background knowledge (and inference) in matching concept with instances, especially, nontypical or borderline instances.

The TT approach was originally proposed by the author in [Michalski, 1986] and was motivated by an observation that although a given individual human concepts may lack precise definition when used alone in a context-independent sense, it acquires precise meaning when used in a combination with other concepts and in a specific context. Consider, for example, the statement: "This tall man in the group in the corner of the room." If there is only one man visibly taller than other people in the indicated group, the statement above precisely specifies the man of interest. The concept "tall," although by itself and/or without context is imprecise (as are the concepts "group," "corner," or "room"), in the given context it conveys a precise meaning—the height of the man pointed out in the group.

Thus, the TT approach views flexible concepts as inherently and intentionally imprecise when they are considered alone and outside of a specific context. Consequently, it does not try to give them a complete and precise meaning in an explicit and context-independent sense. Instead, this approach attempts to describe precisely only the central tendency and to use inference rules and matching procedures to implicitly characterize the complete concept meaning and context dependence.

As mentioned in [Michalski, 1986], the underlying supposition for the TT approach is that the imprecision of human concepts stems not from an undesirable vagueness of our concept definitions, but rather from the universal need for cognitive economy. By allowing concepts to have a context-modifiable meaning, and making them precise only to the extent to which a given situation and/or context requires them to be precise, the expressive power of concepts is greatly enhanced. This means that one can employ fewer concepts for expressing more meanings and helps us simplify our descriptions of our immensely complex universe. The experiments reported here seem to confirm this idea in a microworld to which it was applied.

The following sections describe various aspects of the proposed approach to learning flexible concepts using TT representation, present a simple computational method, and report early experimental results. The learning method employs the inductive learning program AQ15, which is also briefly described. For more details about AQ15, see [Michalski, *et al.*, 1986]. Various improvements to the method and a number of new experimental results with two-tiered representations are reported in [Bergadano, *et al.*, 1988b; 1988c; 1990].

3.2 TWO-TIERED CONCEPT REPRESENTATION

In order to develop a computational method for learning concepts one needs to make assumptions about the meaning of "concepts" in the method. We assume that concepts are named representations of classes of entities, whose borderlines can be imprecise and context dependent. The entities are assumed to have central tendencies within the concept classes, and therefore different concept instances may be characterized by different *typicality*. A concept representation can take a wide range of forms: an explicit description of observable properties of the entities in the class, an abstract description of the function of the entities and their relation to other concepts, a complete or partial listing of the entities, an implied concept characterization by the concept usage, or as a combination of the above. There can be an enormous variation in the specific instantiation of some concepts. Consider, for example, the concepts such as "object" or "set." Because of the assumed central tendencies, context dependence and other previously mentioned properties, a concept representation should allow a varying degree match with concept instances, and use of context-dependent inference rules in performing such matches (examples below illustrate this point in more detail). The problem then is how a concept with such central tendencies and context-dependent meaning can be efficiently represented and learned?

As mentioned earlier, the proposed approach to this problem is based on the idea of two-tiered (TT) concept representation. In the TT representation, a concept is defined by two components: the *base concept representation* (BCR) and the *inferential concept interpretation* (ICI). The BCR (the first tier) is an explicit characterization of a concept, stored directly in the learner's memory. The ICI (the second tier) is a set of matching procedures and inference rules that characterize the allowed modifications and possible variations of the concept meaning in different contexts. Thus, the ICI determines the meaning of a concept by executing these matching procedures and inference rules in the given context, and thus only *implicitly* defines concept boundaries.

In the general theory, the "distribution" of the meaning between the BCR and the ICI is not assumed to be fixed, but is modifiable based on a *criterion of description quality*. One extreme of such a distribution is when the BCR represents explicitly all possible concept variations in different contexts, and the ICI is just direct match. Another extreme is when the BCR is empty, and all concept meaning resides in context-dependent inference rules. The description quality criterion reflects computational properties of the learner and requirements of the problem domain. The former ones include, e.g., the relative costs of remembering concept properties versus deriving them through inference.

In an important, cognitively oriented special case of the TT representation, the BCR is assumed to express the general unifying idea, the typical function of the concept, and/or common measurable properties implied by or correlated with this idea or function. Such a BCR can be viewed as representing the "first approximation of the concept." The ICI, in this case, defines the matching procedures and inference rules for handling less typical instances and context dependence. This type of distribution of the concept meaning between the two tiers facilitates an efficient concept recognition and is related to the idea of censored production rules [Michalski and Winston, 1986].

In matching an instance with the BCR of a concept, the ICI may employ deductive, analogical, or inductive inference. A deductive inference is involved when the instance is a logical consequence of the BCR. An analogical inference is employed when the instance is similar to the BCR in a context-dependent sense. Finally, an inductive inference is employed when in order to match the instance with the BCR the latter needs to be generalized. Illustrative examples of such inference processes are given in Section 3.3. Performing these inferences may involve concept metaknowledge, e.g., the *importance* of concept attributes and frequencies of concept occurrence, the relation to other concepts and other relevant domain knowledge.

An advantage of distributing the concept meaning between the BCR and the ICI is that it permits a learner to flexibly modify or extend the concept meaning by varying matching procedures and inference rules and/or by changing the context of discourse. The concept meaning can thus be changed without having to alter the base concept representation. By evaluating the type and the amount of inference involved in matching a concept with an instance, one may produce a qualitative or quantitative estimation of the strength of such a match. The ability to produce a measure of the strength of match indicates one principal difference between this approach and the fuzzy set approach (e.g., [Zadeh, 1978]). In the fuzzy set approach, a set defining a concept is associated with a membership function, which needs to be *defined* to the learner by a person. The influence of the context is hidden in the definition of this membership function. In the proposed approach, a concept is associated with interpretation procedures and context-dependent inference rules, which implicitly define the membership of an instance in a concept. These rules and procedures can be used to *compute* the membership function in different contexts.

Figure 3–1 illustrates the relationship between the BCR and the ICI in a TT concept representation. It shows that the ICI can, in general, extend the concept meaning beyond the BCR in one area of the description space and reduce the meaning in another area.

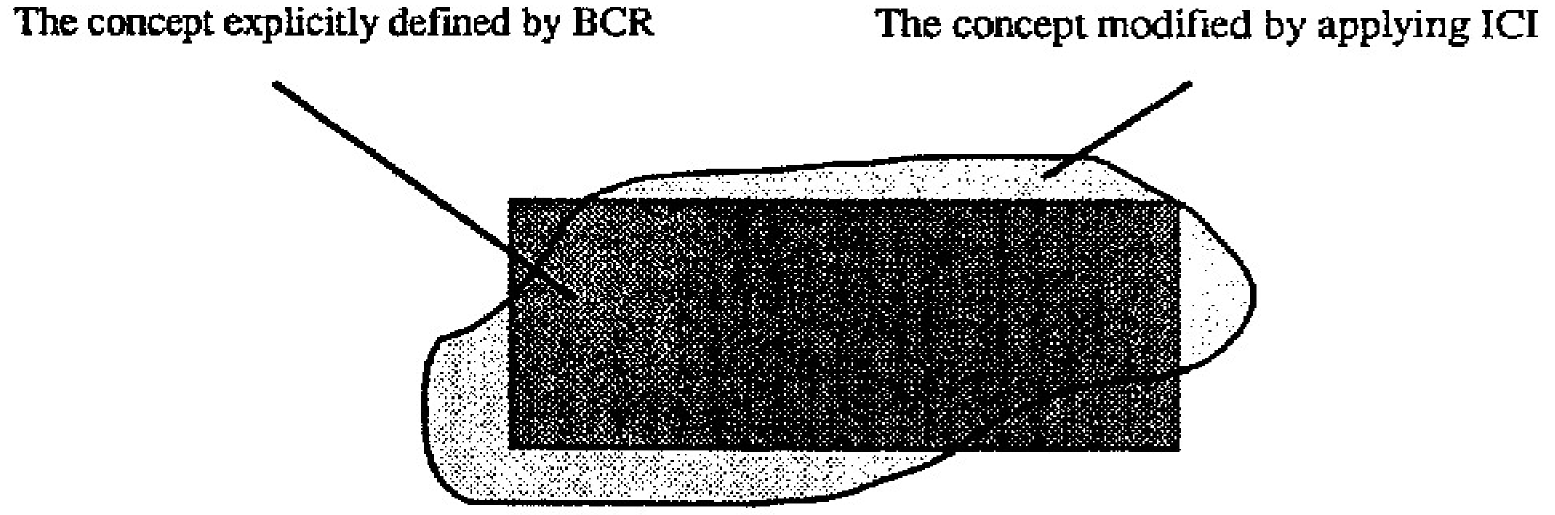

The concept explicitly defined by BCR
The concept modified by applying ICI

Figure 3–1: An illustration of the relationship between the Base Concept Representation (BCR) and the Inferential Concept Interpretation (ICI)

Learning a two-tiered representation of a concept consists thus of two parts:

1. acquiring the base representation, and
2. acquiring the inferential concept interpretation (i.e., matching procedures and inference rules for various contexts).

The ICI can be completely or partially shared by concepts in the same class or inherited from a superclass. By sharing ICI, a significant economy of the concept representation can be achieved. As mentioned earlier, the distribution of the concept meaning between BCR and ICI can vary, in order to optimize the obtained description according to an assumed criterion of *description quality*. Such a criterion depends on the use of the concept and the properties of the learner. This implies that the BCR may be in several forms. For example, it can be in the form of an abstract definition, capturing the general principle and the central function of the concept. Such a description can be short, but inefficient for concept recognition. Alternatively, it can be in the form of a specification of easy-to-measure properties of concept instances. The latter description facilitates an efficient concept recognition, but it may be complex and too restrictive. In general, a BCR can be a combination of such forms.

3.3 EXAMPLES ILLUSTRATING TWO-TIERED REPRESENTATION

Let us consider a few examples illustrating the idea of two-tiered representation.

Example 1. Concept of Sugar Maple Our prototypical image of a sugar maple is that it is a tree with three- to five-lobed leaves that have V-shaped clefts. Some of us may also remember that the teeth on the leaves are coarser than those of red maple, that slender twigs turn brown, and the buds are brown and sharp pointed. As a tree, of course, a maple has roots, a trunk, and branches.

Suppose that while strolling on a nice winter day someone tells you that a particular tree is a sugar maple. A simple introspection tells you that the fact that the tree does not have leaves would not strike you as a contradiction of what you know about sugar maples. Yet, clearly, the presence of leaves of a particular type is deeply embedded in your typical image of a maple tree. The two-tiered theory explains this phenomenon simply: The inferential concept interpretation associated with the general concept of deciduous trees evokes a rule, "In winter deciduous trees lose leaves." Since a maple is deciduous tree, the rule would apply to the maple tree. The result of this inference would override the stored standard information about maple trees, and the inconsistency would be resolved. In this case, matching an instance with the concept requires deductive reasoning from the knowledge associated with a more general concept.

Example 2. Concept of an Abstract Tree Structure Suppose that a student is reading his first book on computer data structures and encounters a drawing of a graph structure, which the author calls a "tree." Calling such a structure a tree will likely not evoke any objection in the student, because he can see in this structure some abstracted and modified features (e.g., upside-down direction) of a biological tree. In this case, matching the graph structure with the concept of a tree involves a generalization operation on the base representation.

Example 3. Concept of a Triangle Let us go back to the concept of a triangle. Formally, a triangle can be described as a geometrical figure consisting of three noncolinear points connected by straight lines. Using the notation of annotated predicate calculus (APC), which is equivalent to predicate calculus but permits one to write logical expressions in a more compact form [Michalski, 1983], one can write:

$$\begin{aligned}\text{Triangle(T, P1, P2, P3)} \Leftarrow\ & \text{Consists(T, P1 \& P2 \& P3) \& Type(P1 \& P2 \& P3)} \\ & = \text{point \& Connected\_by} \\ & \text{(P1,P2 \& P1,P3 \& P2,P3)} = \text{straight\_line \&} \\ & \text{RelationAmong(P1, P2, P3)} = \text{noncolinear} \qquad (1)\end{aligned}$$

In (1), the symbol "&" is used in two related meanings: one, to denote an ordinary (external) conjunction connecting predicates; and second, to denote an *internal conjunction*, i.e., conjunction of terms, treated as a *compound* argument of a predicate. For example, the predicate "Consists(T, P1 & P2 & P3)" states that the triangle T consists of points P1 and P2 and P3.

Suppose that someone tells us that the towers in his hometown form a big triangle. Obviously, the meaning of the triangle in this statement differs from that in the formal geometrical description. To match the two, one needs to make the following assumptions and transformations:

a. In the context of describing a configuration of physical objects such as towers, the individual objects play the role of nodes. Thus, the statement implies that there are three towers in the town. The matching operation involves drawing an analogy between the abstract nodes and the towers, which can be characterized as consisting of one step of generalization (GEN):

   Point —GEN → Object

   and one step of specialization (SPEC):

   Object —SPEC → Tower

b. In the context of towers, the presence of a "straight line" is imaginary, i.e., there is no physical connection, but one could imagine a straight line between the objects (towers). The condition "Connected By" is satisfied in such an abstract sense. This is an operation of generalization. Thus, matching the statement about a triangular arrangement of towers with the formal definition of a triangle involves here both generalization and specialization.

The examples above show that relating a concept instance to a concept representation is not just a straightforward comparison of attribute values in an instance with those in the concept representation, as done in various mechanized decision processes. They show that such a process may involve different forms of inference.

3.4 TRADING BCR FOR ICI

As mentioned earlier, the TT representation does not assume that the distribution of the concept meaning between BCR and ICI is fixed, but that it can change to reflect the goals or the properties of the learning agent. To illustrate the interrelationship between the BCR and the ICI, let us consider an imaginary concept, which we call the "R-ball." Suppose that the complete meaning of this concept is defined by the diagram in Figure 3–2.

Each "1" in the diagram describes an instance of the R-ball by specifying values of attributes for this instance. The set of all instances of the R-ball depicted in the diagram defines precisely the concept of an R-ball. A complete and consistent (CC) description of the concept (i.e., one that covers all "1"s, and does not cover any empty cells) is:

SHAPE = round & BOUNCES = yes
or
SHAPE = round & SIZE = medium or large
or
BOUNCES = yes & SIZE = medium or large (2)

Any instance that strictly matches any of the above rules is recognized as an R-ball (It is assumed that "&" is interpreted as a logical conjunction, the "or" linking conjuncts as a logical disjunction; and the "or" linking attribute values as an internal disjunction [Michalski, 1983]). Assuming that satisfied conditions give the degree of match equal 1, and unsatisfied conditions give the degree of match 0, such an interpretation is equivalent to treating a conjunction as the minimum function (MIN), and a disjunction as the maximum function (MAX) of the degrees of match.

Let us now consider the diagram in Figure 3–3, which presents only four examples of the R-ball (the four "1"s). A CC description of these examples is:

SHAPE = round & BOUNCES = yes & SIZE = medium or large (3)

If interpreted the same way as above, this description covers only the indicated four R-balls, and thus is an incomplete concept description. Suppose, however, that we interpret "&" not as the minimum function but as the *average* function. Suppose

SHAPE	BOUNCES	small		medium		large		SIZE
		light	dark	light	dark	light	dark	COLOR
round	no			1	1	1	1	
	yes	1	1	1	1	1	1	
not round	no							
	yes			1	1	1	1	

Representation:

SHAPE = round & BOUNCES = yes or
SHAPE = round & SIZE = medium or large or
BOUNCES = yes & SIZE = medium or large

Interpretation:

& = MIN (conjunction)
or = MAX (disjunction)

Concept membership:

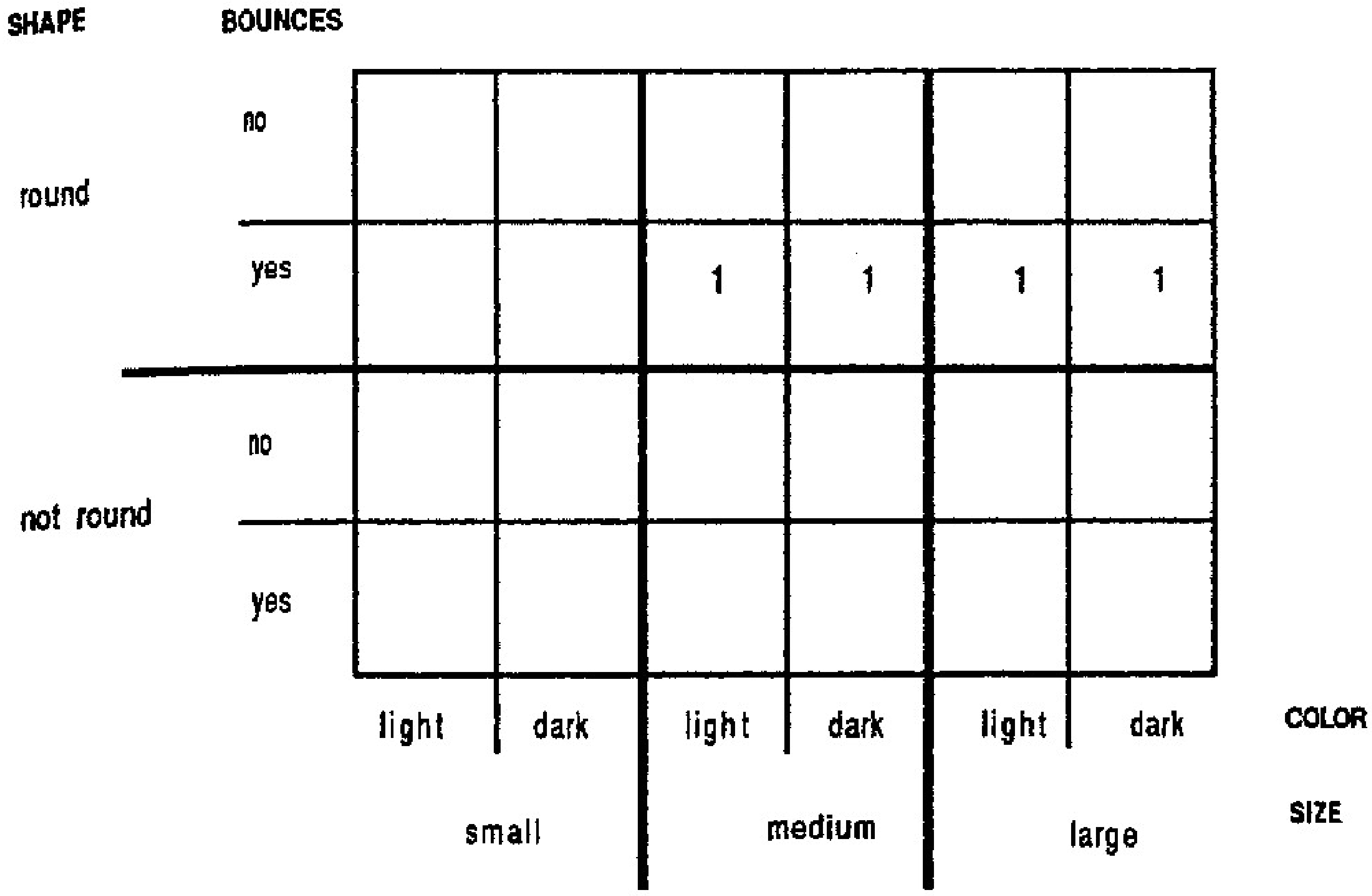
Yes, if degree of match = 1
No, otherwise

Figure 3–2: A diagram illustrating the concept of R-ball

also that we assume that an instance is classified as an R-ball, if it gives a degree of match with the description equal to or greater than ⅔.

The above interpretation, as it can be easily verified, gives a classification of instances into R-balls and not-R-balls exactly the same as the description in equation (2). Thus, we have two logically equivalent representations of R-balls: one, that explicitly describes all concept instances; and the second, that describes explicitly only a subset of the instances, and takes care of the remaining examples implicitly, by the matching procedure. Table 3–1 summarizes information about the two representations, denoted as CR1 and CR2.

In Table 3–1, "rules" are single conjunctions of conditions associated with a given concept. Although representations CR1 and CR2 are logically equivalent, they are pragmatically different. The first representation, CR1, is significantly more com-



SHAPE	BOUNCES	small		medium		large	
		light	dark	light	dark	light	dark
round	no						
	yes			1	1	1	1
not round	no						
	yes						

COLOR

SIZE

Representation:

SHAPE = round & BOUNCES = yes & SIZE = medium or large

Interpretation:

& = AVG

Concept membership:

Yes, if degree of match ≥ 2/3
No, otherwise

Figure 3–3: A subset of examples of the R-ball

plex than CR2. The BCR of the first representation consists of three rules, while the BCR of the second representation consists of only one rule. To compute the degree of match (DM), the ICI of the first representation uses a conventional interpretation of logical connectives (applicable also to multiple-valued conditions); while the ICI of the second representation uses a less conventional interpretation.

The above two concept representations illustrate two different "distributions" of the concept meaning between the BCR and the ICI. In general, there can be a range of logically equivalent concept descriptions that differ in the distribution of the meaning between the BCR and the ICI. The ICIs presented here are just an illustration. A more elaborate interpretation method, called *flexible matching*, is discussed in Section 3.6.

Table 3–1: A comparison between two representations of the concept of R-ball

CR1		CR2	
BCR	ICI	BCR	ICI

3 rules
6 conditions
	& = MIN
	V = MAX
	DM = 1
		1 rule
		3 conditions
			& = AVG
			V = MAX
			DM ≥ 2/3

3.5 LEARNING TWO-TIERED REPRESENTATIONS

The method of learning two-tiered concept representations from examples proposed below utilizes our previous work on inductive concept learning. In the method, learning the BCR of a concept consists of two phases. In the first phase, a complete and consistent (CC) concept description is induced from a set of positive and negative examples of the concept. This phase is performed by using a standard AQ inductive learning methodology, such as implemented in the program AQ15 (see below).

The second phase reduces the so-obtained CC description to a simpler description that maximizes a criterion of description quality. Such a criterion evaluates various properties of the description from the viewpoint of the goals of learning (Section 3.5.2). The description reduction is done using the so-called TRUNC method, which iteratively removes components of the description, from the least "important" to the most "important" (Section 3.5.2). A reduced description that scores best on the assumed quality measure is taken as the base concept representation (BCR).

To determine the ICI, one needs to develop a matching procedure that would handle all positive examples of the concept that do not match BCR, and all negative examples that match BCR. In general, such a procedure needs to involve various context-dependent inference rules. In the method described here, we limit ourselves only to a very simple ICI, based on a *flexible matching* procedure (Section 3.6).

The above method of learning TT descriptions has been implemented in the learning system AQTT-15. The system integrates the AQ15 learning program with the TRUNC procedure and flexible matching. The next two sections give a brief description of the AQ15 module and the TRUNC procedure, respectively.

3.5.1 An Overview of the AQ15 Module

AQ15 is a descendant of the AQ family of inductive learning programs (e.g., [Michalski, 1972; Michalski and Larson, 1975; Mozetic and Hong, 1984]). From the viewpoint of its capabilities, AQ15 is a highly advanced program for learning attributional descriptions from examples, which can serve as a mini-laboratory for exploring various aspects of inductive learning.

Different versions of the AQ family were experimentally applied to a variety of practical tasks, such as learning criteria for discriminating between cancer of the pancreas and cancer of the liver [Michalski, 1973], defining provably correct general characterizations of the "win" and "draw" positions in a chess endgame [Negri and Michalski, 1977], determining rules for plant disease diagnosis [Michalski and Chilausky, 1980], and automatically creating a knowledge base for diagnosing cardiac arrhythmias from electrocardiograms [Mozetic, 1986; Bratko, Mozetic, and Lavrac, 1989]. The latter work is one of the most advanced and most interesting applications of machine learning to a practical problem.

The AQ programs are based on the AQ algorithm for a quasi-optimal solution of the general covering problem, originally developed by the author [Michalski, 1969]. (For a more tutorial presentation, see [Michalski and McCormick, 1971].) The algorithm generates the near-minimum or minimum number of general rules distinguishing between a set of positive examples and a set of negative examples. The complete version of the algorithm also produces an upper bound on the maximal difference in the number of the rules between the obtained solution and the minimal one. This upper bound allows the algorithm to produce a provably minimal solution for some classes of covering problems (which are NP-complete) in the polynomial time. While the complete version of the algorithm is more interesting from a theoretical viewpoint, for machine learning problems a simplified version seems to be more useful. The simplified version does not produce the upper bound on the complexity of the solution, but it is easier to implement and faster to run. Here is the basic structure of a simplified version of the AQ algorithm:

1. A single positive example, called a *seed*, is selected (randomly or by design) from the available positive examples, and a set of alternative, most general rules (conjunctive descriptions) characterizing this example is computed. The limits to which the rules are generalized are defined by negative examples. The obtained set is called a *star* for the seed.
2. The most preferred rule is selected from the star according to a *rule preference criterion* (see below). If this rule, jointly with any previously generated rules, covers all positive examples, the algorithm stops.
3. Otherwise, a new seed is selected among the examples uncovered so far, and steps 1 and 2 are repeated until all examples are covered.

The ruleset assembled from rules selected in each step constitutes a complete and consistent concept description and optimizes the assumed *description preference criterion*.

Typically, supplied examples are insufficient for uniquely defining a concept description. Therefore, one needs a criterion that would enable one to choose among alternatives, which represent different generalizations of positive examples. In the AQ approach, such a criterion is not viewed as a "bias," which might imply that the choice is arbitrary, but is supposed to reflect the requirements of the problem domain. For example, if costs of measuring attributes vary significantly in the given problem domain, it may be desirable to choose a description that is more complex (e.g., has more components), but which involves "inexpensive" attributes, and thus is less costly overall. If input examples are "noisy" and/or the overall efficiency and accuracy of concept recognition is the primary goal, then it may be desirable to chose a description that is incomplete and/or inconsistent with regard to the training examples (see Section 3.5.2, and [Bergadano, *et al.*, 1990]).

The *rule preference criterion* used in selecting a rule from a star is assumed to produce a concept description that will score high on a *description quality criterion*. That is, the rule preference criterion should reflect the desirable properties of the goal concept description, according to the requirements of the problem domain. For example, if the description quality criterion requires descriptions to have the minimum number of rules, then the rule preference criterion might rank high the rules that individually cover the maximum number of examples. If each rule covers many examples, then it is likely that fewer rules will be needed to cover all examples.

The rule preference criterion is defined by a list of elementary criteria assembled by a user from a set of predefined criteria. In AQ15, the predefined criteria relate to various measurable properties of a rule, such as the coverage (the number of positive examples covered by a rule), the simplicity (measured by the number of attributes involved in the rule), the cost (the sum of the measurement costs of individual variables), an estimate of generality (such as the ratio of the number of possible examples to the number of actually observed examples covered by the rule), and others.

To determine the "best" rule, the elementary criteria on the list are applied one by one to individual rules in the star, and the rule that best satisfies all criteria within a certain tolerance range is selected. Such a multicriterion measure for selecting the best alternative from a set of alternatives is called a *lexicographic evaluation functional* or LEF [Michalski, 1973; 1983].

AQ programs express concept descriptions using the *variable-valued logic system 1* (VL$_1$), which is a multiple-valued logic extension of the proposition calculus with typed variables [Michalski, 1974]. It is an easy-to-interpret, highly concise and powerful language for expressing any relationship among multivalued and multitype attributes.

The simplest expression in VL$_1$ is an elementary condition, called *selector*, which relates a variable or an attribute to a value or an *internal disjunction* of values, for example, [color = blue or red] or [height = 3 inches]. A conjunction of such conditions forms a *complex*, which can be viewed as a rule for partially or completely defining a concept. For example, suppose that the complex [weight = high] & [length = 2..5 meters] describes (some or all) examples of the class "big objects." Such a complex can be viewed as a rule [weight = high] & [length = 2..5 meters] ⇒

[class = big objects]. A set of complexes (rules) can be expressed as a disjunctive normal expression, in which individual complexes are linked by disjunction.

From now on, by *rules* we will mean VL_1 complexes associated with an appropriate decision class, and by *conditions* (briefly, conds) we will mean VL_1 selectors. By a *concept (class) description* we will mean a set of rules (a ruleset) whose right-hand side points to that concept. A ruleset that describes all positive examples and none of the negative examples is a *complete and consistent* (CC) concept description (such a ruleset is also called a *cover*).

The AQ15 module is capable of *incremental learning with full memory* of past examples. In this type of incremental learning, the process of modifying the current hypothesis to accommodate new facts takes into consideration all past examples [Reinke and Michalski, 1988]. This way, it can be guaranteed that any so-modified concept description (hypothesis) is always complete and consistent with regard to all examples. Such a method can therefore produce higher quality concept descriptions than incremental methods *with no memory* of past examples (e.g., [Winston, 1970; Michalski and Larson, 1976; Gross, 1988; Iba, Woogulis, and Langley, 1988]). Because the costs of computer memories are decreasing, the need for storing past examples is not considered a strong disadvantage of such incremental learning.

In the sense that this method keeps past concept examples, it is related to exemplar-based learning, in which concepts are represented by positive examples (e.g., [Bareiss, Porter, and Craig, 1990—Chapter 4, this volume]). The principal difference between the two approaches is, however, that the method represents concepts by "optimized" general descriptions rather than by examples. As mentioned earlier, keeping a general concept description facilitates a number of operations, such as determining the relationship between the concept and other concepts, incrementally modifying the concept, etc.

The program also has a "generality parameter," which enables it to generate descriptions of different generality from the same input examples. Depending on the setting of this parameter, the generated concept description may be maximally general, maximally specific or intermediate. The default value of the parameter produces a maximally general description, which covers the maximum number of instances, observed or hypothetical, without covering any negative examples. By specializing such a description, one can produce another extreme; i.e., a maximally specific description, which cannot be more specialized without "uncovering" some positive examples.

Because the program can learn incrementally, it allows a user to supply some initial, partially correct decision rules, which are then improved in the process of applying them to new examples. The program can also perform *constructive induction*, in which domain background knowledge is used to generate new concepts (e.g., attributes) that are not initially specified, but may produce a better final description. Such domain knowledge can be expressed using two types of rules: *L-rules*, which are in the form of logical assertions or if–then rules, and *A-rules*, which are in the form of arithmetic functions or term-rewriting rules. A more detailed description of AQ15 can be found in [Michalski, *et al.*, 1989b].

3.5.2 The TRUNC Procedure

The purpose of the TRUNC procedure is to determine the "best distribution" of the concept description between the explicit base concept representation (BCR) and the inferential concept interpretation (ICI) [Michalski, 1986a; 1986b]. The procedure starts with a complete and consistent concept description, whose rules have been ordered on the basis of their "importance" from the most important to the least important. This description is then reduced, by truncating one rule at each step, starting with the least important rules. After each step, a "quality" of the reduced description is measured. The description that has the highest quality is chosen as the base concept representation (BCR). To measure the description quality, the method takes into consideration the description complexity and its performance on testing examples. In general, a number of other factors can be taken into consideration in evaluating a description (e.g., [Zhang and Michalski, 1989]).

To determine the importance of a rule, each rule in the description is associated with a pair of weights, *t* and *u*, representing the *total* number of training examples covered by the rule, and the number of training examples covered *uniquely* by that rule, respectively. Obviously, the t-weight is always greater or equal to the u-weight of a rule, and the difference between the two indicates the degree of overlap between the rule and other rules in the description. The t-weight of a rule can be interpreted as a measure of its representativeness as a concept description, and the u-weight as a measure of its interrelationship with other rules. The rule with the highest t-weight may be viewed as characterizing the most typical concept properties, and thus serve as its prototypical description. The rules with the low t-weight describe rare, exceptional cases. If training examples are noisy, such "light" rules are indicative of errors in the data. A rule with a large u-weight (and consequently large t-weight) is a highly representative and irreplaceable component of the concept description. A rule with zero u-weight is redundant. A rule with a large t-weight and a small u-weight is a good candidate for a merger with another rule.

Let us now describe the TRUNC procedure in more detail and illustrate it with an example. The procedure starts with a complete and consistent (CC) description obtained from the AQ15 module. The rules in the CC description are linearly ordered from those with the highest t-weight to those with the lowest t-weight. (If two rules have the same t-weight, the one with the higher u-weight has precedence; if they also have the same u-weight, the order is arbitrary.)

Figure 3–4 illustrates such an ordering. A consistent and complete description consists of four rules, depicted as rectangles. The rectangles overlap because rules may logically intersect; i.e., some training examples may be covered by more than

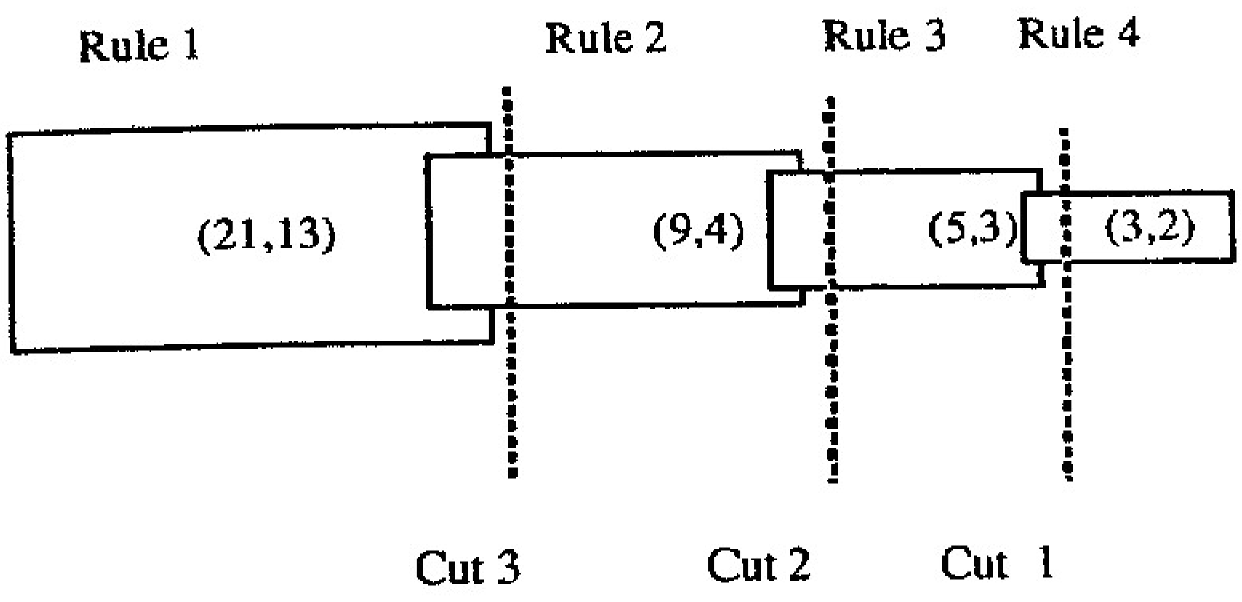


Figure 3–4: An illustration of a t-ordered concept description

one rule. In each pair (x,y), x denotes the t-weight, and y denotes the u-weight of the corresponding rule.

The procedure proceeds by removing at each step a rule that in the currently considered description has the smallest t-weight. In Figure 3–4, first rule 4 is removed, then rule 3, etc., until only one, rule 1, is left. The description consisting of the last remaining rule, i.e., the rule with the highest t-weight, is called the "top rule" description.

In Figure 3–4, cuts 1, 2 and 3 mark consecutive truncations, producing descriptions with the number of rules equal to 3, 2, and 1, respectively.

Removing a rule from a description is equivalent to removing a disjunctively linked condition from a description. Such a process *specializes* the knowledge representation [Michalski, 1983] and produces an *incomplete* concept description (one that does not cover some positive training examples).

All so-reduced descriptions are evaluated according to a *description quality criterion*. A simple form of such a criterion is to require the description to perform well on testing examples and to have low complexity. Indicators of complexity are the number of rules in the description and the total number of conditions in these rules. The description that best satisfies the assumed description quality criterion is taken as the BCR.

The criterion of description quality should reflect the needs of the problem at hand and can depend on many factors. For more details about this topic, see [Bergadano, *et al.*, 1988a; Zhang and Michalski, 1989].

Intuitively, one might expect a trade-off between the simplicity of a description (a reciprocal of complexity) and its performance. Such an expectation is justified because simplifying the description (by removing a rule) uncovers some training examples. To test this hypothesis, we have performed a series of experiments with data from the area of medical diagnosis. Results of these experiments have been quite surprising. They are described in Section 3.7. The problems of other potential tradeoffs characterizing concept descriptions are studied in *variable precision logic* [Michalski and Winston, 1986].

In summary, the TRUNC method reduces the initial CC concept description to the "best" description, which is used as the BCR of the concept.

3.6 RELATING INSTANCES TO CONCEPTS: FLEXIBLE MATCHING

We now turn to the topic of inferential concept interpretation (ICI) of a concept description. In order to determine the identity of an unknown instance, the instance needs to be matched against a set of candidate concept descriptions. One can distinguish between two basic methods for matching an instance with a set of descriptions: the *sufficient* match and the *best* match.

In the sufficient match method, the properties of the instance are matched against conditions in the candidate descriptions to determine which description is satisfied. An instance may satisfy a description either completely (a *crisp match*) or "sufficiently" (a *satisficing match*). Assuming that an instance can belong to only one candidate concept and that the descriptions are logically disjoint, then any description that is found to be satisfied determines the instance identity. In such a situation, there is no need to test other candidate descriptions. This property has been explored in *dynamic recognition*, which tries to achieve an instance recognition with the minimum number of operations and without actually matching individual rules [Michalski, 1989].

In the best match approach, one determines a degree of "fit" or "similarity" between the instance and candidate descriptions, and selects the description that provides the closest match. Determining a "similarity" between a description and an instance can be accomplished in a variety of ways, ranging from an approximate matching of feature values to "conceptual cohesiveness" [Michalski and Stepp, 1983].

In the two-tiered approach, an instance is matched against the BCR (base concept representation) of candidate concepts, using the ICI (inferential concept interpretation). In the method described here, the ICI consists of a *flexible matching* procedure, which applies the best match approach and does not involve any explicit rules of inference. If there is no crisp match with the BCR of just one description, the procedure measures the fit between an instance and the candidate BCRs, and chooses the concept that provides the best fit. A more advanced ICI is considered in [Bergadano, *et al.*, 1990].

As described before, the BCR of a concept is a logic-style description consisting of one or more rules (a ruleset). When matching a new example against such a ruleset, three outcomes are possible: There may be only one match (one ruleset is satisfied), more than one, or there is no match. These three types of outcomes are called *single-match, multiple-match*, and *no-match*, respectively (Figure 3–5).

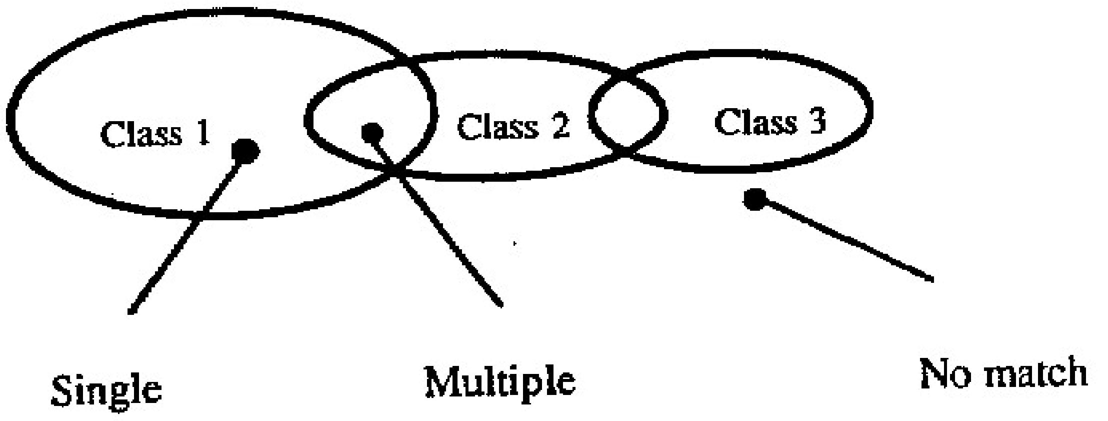


Figure 3–5: An illustration of single-match, multiple-match, and no-match

In the implemented flexible matching procedure, each type of match involves a different decision assignment procedure. When there is a single-match case, the instance is immediately assigned the concept associated with the matched BCR. The decision is counted as correct, if it is equal to the expert-given classification of the testing example, and as incorrect otherwise. If there is multiple match or no-match, the system makes the decision on the basis of the best match. This decision is compared with an expert decision and evaluated as correct or incorrect.

There are potentially many ways to define a measure of fit between an instance and a description. Below are simple heuristic measures, one for the multiple-match case, and the other for the no-match case.

Let $C_1,\ldots,C_m$ denote concepts (decision classes) and e denote an event (instance) to be recognized. For each concept C_i, we have a BCR consisting of one or more rules. Each rule is a conjunction of conditions (Conds). For generality, it is not assumed that BCRs of different concepts are logically disjoint.

The Multiple-Match Case When an event matches the BCR of more than one concept, the system selects the concept whose BCR provides the highest *degree of fit* with the event. To determine such a degree, we first define the degree of fit, F(e, Rule_j), between an instance e and a Rule_j. If the instance satisfies the Rule_j, then F(e, Rule_j) is equal to the *significance* of the rule, otherwise 0. The rule significance is defined as the ratio of the t-weight(Rule_j); i.e., the number of training examples covered by the Rule_j, by the total number of training examples (#examples). Thus, we have:

$$F(e, \text{Rule}_j) = \begin{cases} \text{t-weight}(\text{Rule}_j)/\text{\#examples, if } e \text{ satisfies Rule}_j \\ 0\text{, otherwise.} \end{cases} \tag{4}$$

A justification for this measure is that if an event satisfies a rule that describes a large proportion of training examples, then it is likely that this event belongs to the class implied by the rule.

The degree of fit, $F(e, C_i)$, between the instance e and the concept description C_i, is the *probabilistic sum* of the degrees of fit between e and rules in C_i. If C_i consists of just two rules, $Rule_1$ and $Rule_2$, we have:

$$F(e, C_i) = F(e, Rule_1) + F(e, Rule_2) - F(e, Rule_1) \times F(e, Rule_2) \qquad (5)$$

If C_i consists of more than two rules, equation (5) is iteratively applied. The reason for using the probabilistic sum is that individual rules may logically intersect. The most plausible concept is defined as the one with the largest degree of fit.

The No-Match Case If an event does not satisfy the BCR of any concept (ruleset) under consideration, but it is assumed that it belongs to one of the candidate concepts, the degree of fit between the event and a ruleset depends on the degree of "closeness" between attribute values in the event and those in the ruleset, and on the prior probability of the concept.

For illustration, we will use here a measure of fit described in the study by [Michalski, *et al.*, 1986]. First, we define the degree of fit, $F(e, Cond_k)$, between event e and condition $Cond_k$. This measure takes value 1, if the condition is satisfied; otherwise, it is proportional to the relative size of the attribute's domain covered by the condition:

$$F(e, Cond_k) = \begin{cases} 1 & \text{if condition } Cond_k \text{ is satisfied by e} \\ \#values/DomainSize, \text{ otherwise.} & \end{cases} \qquad (6)$$

In (6), #values is the number of alternative attribute values in the condition, and DomainSize is the total number of attribute's possible values. For example, if $Cond_k$ is [attribute = 2..5], the #values is 4.

A justification for this measure is that conditions in which an attribute can take many values are viewed as weaker than conditions in which it can take only one or few values. For example, if an event does not satisfy condition [blood type = A v O], than this should cause a lower loss in confidence than if it does not satisfy condition [blood type = A].

The degree of fit, $F(e, Rule_j)$, between an event e and the $Rule_j$ is the product of degrees of fit between e and conditions in the rule, weighted by the ratio of t-weight($Rule_j$) over the total number of training examples:

$$F(e, Rule_j) = \prod_k F(e, Cond_k) \times (\text{t-weight}(Rule_j)\#examples) \qquad (7)$$

The above measure is based on the assumption that individual conditions in a rule are independent, which is justified, because the induction algorithm tends to form rules with independent conditions. This measure can be viewed as a combination of a "closeness" of the event to a rule and an estimate of the prior probability of the rule in the description. This measure could be further refined by replacing a discrete degree of satisfaction of a condition by a continuous degree, such as described in [Michalski and Chilausky, 1980].

The measure of fit, $F(e, C_i)$, between the event e and a concept C_i is defined as the probabilistic sum of the Fs of rules in the concept description, the same way as in the multiple match case (equation (5)).

3.7 EXPERIMENTS WITH AQTT-15

The presented method of learning concept descriptions from examples combines a conventional inductive learning approach with ideas of TT representation. It has been implemented in the system AQTT-15, whose major components include the AQ15 inductive learning program, the TRUNC method of description reduction and a procedure for flexible matching.

To illustrate the performance of the system, this section presents results from its experimental application to learning diagnostic rules in three medical domains: lymphatic cancer, prognosis of breast cancer recurrence, and location of primary tumor. All the data were obtained from the Institute of Oncology of the University Medical Center in Ljubljana, Yugoslavia [Kononenko, Bratko, and Roskar, 1984].

Lymphatic Cancer In this domain there were four possible diagnoses; i.e., decision classes. The available data consisted of descriptions of about 148 patients and their diagnosis. Each patient was described in terms of 18 multivalued attributes. The diagnoses were not verified independently. A specialist's estimation is that internists diagnose this kind of disease correctly in about 60% of the cases, and specialists in about 85% of the cases.

Prognosis of Breast Cancer Recurrence There were two decision classes. The available data described 286 patients with known diagnostic status five years after the operation. Each patient was characterized by nine multivalued attributes. The set of attributes was incomplete; i.e., the measured attributes were insufficient to always completely discriminate between patients with different diseases. Diagnosing on the basis of these attributes therefore has to produce a certain amount of error. Five oncologists of the Institute of Oncology in Ljubljana were tested, and they gave correct prognoses in 64% of the cases. There is no estimate of the performance of internists in this domain.

Location of Primary Tumor Physicians distinguish between 22 locations of a primary tumor. The available data characterized 339 patients with known locations of primary tumors. Each patient was described in terms of 17 attributes. As in the prognosis of breast cancer recurrence, the set of attributes was also incomplete. Four internists (who were tested) determined the correct location of the primary tumor in 32% of the cases, and four oncologists in 42% of the cases.

All the data used in the experiments are summarized in Table 3-2. Individual columns represent respectively: the disease type; the number of available examples for this disease type; the number of different diseases of the given type (decision classes); the number of attributes used to characterize a patient, and the average number of values per attribute for each of the disease types.

Table 3-2: A characterization of three problem domains

Disease type	#Examples	#Classes	#Attrs	#Vals/Attr
Lymphatic cancer	148	4	18	3.3
Breast cancer	286	2	9	5.8
Primary tumor	339	22	17	2.2

For all three domains (disease types), 70% of the examples were selected for learning diagnostic descriptions of diseases, and the remaining 30% were used for testing the learned descriptions.

The first phase of the experiment was to induce complete and consistent (CC) descriptions from training examples for all decision classes (diseases) in each domain. The results are summarized in Table 3-3.

Table 3-3: The average complexity of complete and consistent descriptions(i.e., before truncation)

Disease type	#Rules/Class	#Conds/Rule	#Values/Attr	#Examples/Rule
Lymphatic cancer	3.0	3.1	1.8	8.0
Breast cancer	20.0	3.9	1.7	5.0
Primary tumor	5.2	5.3	1.0	2.3

Individual columns list, respectively: the disease type, the average number of rules in the description of each decision class, the average number of conditions per rule, the average number of attribute values in a condition (i.e., values linked by the internal disjunction), and finally, the average number of training examples covered by one rule. One can see that in the domain of lymphatic cancer, rules cover on the average eight examples, which indicates the presence of relatively strong patterns.

On the other hand, in the domain of primary tumor, the rules cover on the average only slightly more than two examples, which suggests an absence of strong patterns.

The second part of the experiment was to apply the TRUNC method to reduce the above CC descriptions and to determine the best candidate for a BCR of each decision class. The quality of initial and reduced descriptions was evaluated in terms of their complexity and their performance on testing examples. The description complexity was measured by the number of rules in the description and the total number of conditions in it.

Results reported here compare three types of descriptions. The first type were the initial CC descriptions induced from training examples of each decision class. The second type descriptions consisted of only those rules in CC descriptions that covered uniquely more than one training example (the "unique>1" case); all other rules were removed. (Notice, that the removed rules could cover several training examples, because the removal condition relates only to uniquely covered examples). The third type descriptions consisted of only "top rules" for each class; i.e., rules that cover the largest number of training examples in each class. Such descriptions can be viewed as covering only the most "typical" examples.

The experiment was performed four times, using randomly chosen training and testing examples. The results describing the average of the four experiments are presented in Table 3–4. A more detailed description of the experiments is in [Michalski, *et al.*, 1986].

Table 3–4: Results of testing three types of diagnostic rules generated by AQTT-15

Disease type	**Description type**	**Complexity**		**Diagnostic accuracy**	**Experts/ internists**	**Random decision**
		#Rules	**#Conds**			
Lymphatic cancer	Complete	12	37	81%	85/60%	25%
	Unique>1	10	34	80%		
	Top rule	4	10	82%		
Breast cancer	Complete	41	160	66%	64%	50%
	Unique>1	32	128	68%		
	Top rule	2	7	68%		
Primary tumor	Complete	104	551	39%	42/32%	5%
	Unique>1	42	257	41%		
	Top rule	22	112	29%		

The column "Description Type" indicates the type of the description used in testing. The types are:

"Complete"	Original CC description of the each decision class
"Unique>1"	Description with rules that cover uniquely more than one training example
"Top rule"	Description with only one rule covering the largest number of examples.

The bold description represents a suggested candidate for the BCR of each decision class, assuming that the diagnostic accuracy takes precendence over the simplicity.

The column "Complexity" gives a characterization of the complexity of a concept description in terms of the number of rules in it and the total number of conditions in all rules. The column "Diagnostic Accuracy" specifies the percentage of correct diagnoses made by the descriptions for testing cases (where the "correctness" is defined as the agreement of the rule-based diagnosis with the diagnosis stated in the data). The column "Experts/Internists" gives an estimate of the percentage of correct diagnoses made by specialists in the given domain and internists, respectively [Kononenko, Bratko, and Roskar, 1986]. The column "Random Decision" indicates the probability that a decision taken at random is correct.

Some results shown in Table 3–4 seem to be rather surprising. One striking case concerns the diagnosis in the domain of lymphatic cancer. The "top rule" description gave the highest diagnostic accuracy (82%) among all descriptions, although it was the simplest (it had three times fewer rules than the CC description). A similar phenomenon occurred in the breast cancer domain, where the diagnostic accuracy of the "top rule" description was 68% versus 66% of the "complete" description; while it had about 20 times fewer rules (two rules versus 41). Thus, in both cases, the "top rule" description is the clear candidate for the BCR of the concepts, as it gives both the highest diagnostic accuracy and the simplest concept representation.

In the domain of the location of primary tumor, the diagnostic accuracy of all learned descriptions was significantly lower than in the previous two cases, ranging from 29% to 41%. The best performance was achieved by the "Unique>1" description (41%), which has on the average about two rules per disease (42/22). Although this is a low performance, it is comparable with that of specialists (42%). The performance of the "top rule" description (29%) was significantly lower than both, the "Unique>1" and the "Complete" descriptions. This indicates that truncating the description below two rules per class goes too far. As the "top rule" description is, of course, the simplest (22 rules; one rule per class), there is a trade-off between the diagnostic accuracy and the complexity. Assuming the precedence of the diagnostic accuracy over the simplicity, the "Unique>1" description is chosen as the BCR.

Why did the "top rule" descriptions give a better diagnostic accuracy than the CC descriptions in the first two domains? Clearly, these descriptions do not cover examples that were uniquely covered by the truncated rules (the total eight rules were removed from CC descriptions in the domain of lymphatic cancer, and 82 rules in the domain of breast cancer).

One reason for this behavior seems to be the use of flexible matching. Due to such matching, events that are not covered by a description are still correctly classified, if they have the "best fit" with the description of the correct class (recall the example with the R-ball). Since the "top rule" can be viewed as representing the "central tendency" of examples from the given class, then even examples not covered by it are likely to fit better to it than to the "top rule" of other classes. Another reason may be the well-known phenomenon of "overfitting" [Watanabe, 1969]. It has been observed, that in the presence of noise in the data, a simpler description, although giving a greater error rate on the observed data, may be a better representation of the true relationship than a complicated one with a lower error rate on the observed data.

In diagnosing the location of primary tumor, the results were generally poor, which may be attributed to several factors. This domain has significantly more decision classes than the other two domains (22 versus four in lymphatic cancer and two in breast cancer), and relatively few examples per class were available (about 15 versus 37 in lymphatic cancer and 143 in breast cancer). The set of available attributes was relatively small (17) for such a large number of classes, and incomplete (i.e., the attributes were insufficient to discriminate completely between different classes). The attributes were mostly binary, and thus less informative than those in the other two domains (the average number of possible attribute values was over three in the domain of lymphatic cancer, and about six in breast cancer). The available data can then be classified as being of a substantially lower quality than the data in the other two domains. Individual rules in the CC description covered only few examples (2.3 on the average), in contrast to rules in the other two domains (in lymphatic cancer, eight examples per rule, and in breast cancer, five examples per rule). Thus, the rules have covered only small portions of the examples in the description space.

The above indicates an absence of a single, strong pattern in this domain, which explains why the "top rule" description gave a poor diagnostic accuracy. The relatively high performance of the "Unique>1" descriptions (with two rules per class) indicates that there were on the average two relatively important patterns in this domain.

3.8 A COMPARISON WITH THE ASSISTANT PROGRAM

A popular approach to empirical learning from examples is based on building a decision-tree representation of a group of related concepts (e.g., [Quinlan, 1983; Chapter 5, this volume]). This section discusses representational issues of the rule-based method implemented in AQTT-15, and the decision-tree-based method, implemented in ASSISTANT [Cestnik, *et al.*, 1986], a descendant of ID3 (e.g., [Quinlan, 1983]). It also presents results from applying ASSISTANT to the same medical problems as above [Kononenko, Bratko, and Roskar, 1986].

In the decision-tree representation, individual nodes correspond to single attributes, the branches from the nodes to the values of the attributes, and the leaves to individual concepts (decision classes). The process of creating a decision tree involves an iterative application of an attribute-selection technique. At each step, the "best" attribute (e.g., the most predictive as to the identity of examples) is selected from a given set of attributes and assigned to a node of the generated tree, until the leaves of the tree give a unique classification of the training examples. Such a process is simple to implement, since it does not involve any complex reasoning or taking into consideration an explicitly defined domain knowledge.

Like AQTT-15, ASSISTANT creates first a description (here, a decision tree) that gives a complete and consistent classification of training examples for all concepts (decision classes). This tree is then reduced by a *tree-pruning* technique, in order to maximize the classification accuracy on testing examples (see also Chapter 5 of this book).

The tree-pruning technique removes certain subtrees from the given decision tree, and replaces them with leaves. Each leaf so created is assigned the most dominant concept among the concepts associated with the leaves of the removed subtree. For this dominant concept, such pruning is equivalent to removing conjunctively linked conditions from a concept description, and thus represents a generalization operation [Michalski, 1983].

For other concepts associated with leaves of the pruned subtree, the pruning is equivalent to removing disjointly linked conditions from a description, and thus represents a specialization operation. Notice, that the training examples of these other concepts have no longer any representation in the tree. Therefore, the so-pruned tree will necessarily misclassify some of the training examples. From the standpoint of the TT representation, a pruned tree can be viewed as a special case of the BCR of a class of concepts represented by the tree.

As indicated above, the tree-pruning technique performs an *interdependent* generalization and specialization of the initial knowledge representation. It moves the boundaries of a partition of the whole description space, but it cannot independently modify the boundaries of individual concepts; i.e., to independently generalize or specialize individual concepts. Therefore, pruning a subtree may improve performance for one decision class, but may decrease it for other classes.

In the rule-based representation used in AQTT-15, concept descriptions can be independently modified. The TRUNC procedure removes individual rules from a description, which is a specialization operation. This operation is done independently for each decision class. The union of the conditions of the rules does not have to cover the whole description space. Instances that do not match any rule to a sufficient degree can be assigned the "undecided" decision. In contract, a decision tree always partitions the whole description space. Thus, if there are undecided instances, one needs to introduce an explicit "undecided" class, which will be associated with some leaves. Therefore, the decision tree representation may be overly complex in decision problems where there are many undecided cases. Also, for a similar reason, it is usually not possible to add a new concept to a decision tree without building a new tree from scratch. Adding new concepts using a rule representation, may be done by adding new rules.

When there are many concepts differing only slightly from each other, a decision tree may produce a very compact and efficient representation, because classes will share many of the same nodes. On the other hand, if there are few concepts, but each described by somewhat different sets of attributes, the decision tree may be very complex. The complexity of rule representation depends directly on the number of concepts (unless rules are organized into a hierarchy).

In a decision tree, a classification decision is determined by a sequential testing of attributes assigned to the nodes from the root to the leaves. Each node represents a clear-cut test, which results in the selection of one specific branch to follow. Such a process can be quite efficient, if the values of all relevant attributes are known. If some attribute value is unknown, however (e.g., the value of the attribute corresponding to the root), it is difficult to derive a classification decision. In the rule-based representation, the evaluation order of attributes is unimportant. Because rules are independent units of knowledge, a decision can often be reached without knowing the values of many attributes.

From the logical viewpoint, rulesets and trees are equivalent representations. A ruleset can be represented as a logically equivalent decision tree, and vice versa. (This is true, of course, only in the case of *attributional* rules; i.e., rules that involve only attributes, as opposed to multiargument predicates or relations. *Structural rules*, which involve predicates and quantifiers, cannot be represented as a decision tree.)

Given a decision tree, by tracing paths from the root to individual leaves, one can determine an equivalent set of rules. Each path generates one rule. Given a set of rules, one can also determine a logically equivalent decision tree. The latter process can be done simply by using *decision diagrams* (e.g., [Michalski, 1978a; 1978b]).

From the pragmatic viewpoint, however, the rule representation has greater expressive power than the decision tree representation. This means, that a ruleset may be significantly simpler than a logically equivalent decision tree. As a simple illustration of this, consider representing the following ruleset as a decision tree:

a & b => Class 1
c & d => Class 1
~a & ~c & d => Class 2;
Otherwise, the class is U. (8)

The simplest decision tree (i.e., with the smallest number of nodes; there are few equivalent such trees), which is logically equivalent to this ruleset is shown in Figure 3–6. The left branch stemming from a node denotes the "false" value, and the right branch, the "true" value of the attribute assigned to this node.

Reexpressing this tree as a ruleset by tracing different paths from the root to the leaves gives the following rules:

a & b => Class 1
a & ~b & c & d => Class 1
~a & d & c => Class 1
~a & d & ~c => Class 2
Otherwise, the class is U. (9)

Comparing (8) with (9), one can see that the ruleset produced from the tree has more rules than the original ruleset, and that some rules in it have more conditions than corresponding rules in (8). For example, the second rule for Class 1 in (9) involves four conditions. Out of these four conditions, two are redundant (a and ~b). This means, the Class 1 decision can be assigned in some cases without knowing values of attributes a and b, contrary to what is stated by the second rule in (9). Thus, a tree representation may be misleading as to the logical dependence of a decision class on some attributes.

As mentioned above, using a tree for decision making requires a sequential testing of attributes. Because of this, the decision-tree approach makes it difficult to determine a fit (or similarity) between an instance and the whole concept description

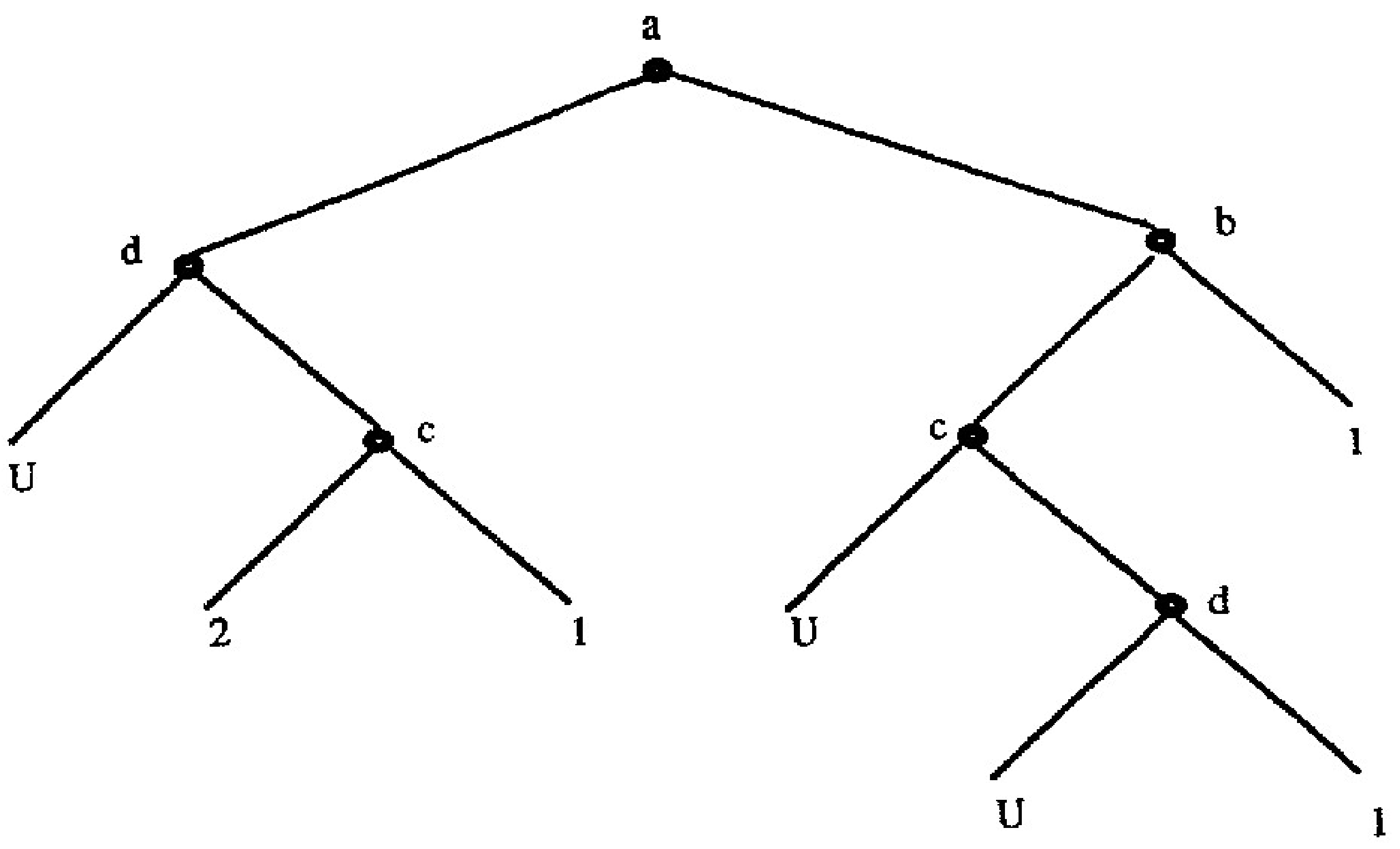


Figure 3–6: A decision tree logically equivalent to the ruleset in equation (8)

(a path through the tree). Therefore, it seems difficult to use a flexible matching technique (like the one described here) with a decision-tree representation (unless one transfers the tree to a ruleset).

To decrease the inflexibility of "clear-cut" tests of decision trees, Quinlan (Chapter 5 of this book) describes a method of "soft thresholds," which is somewhat similar to the method of determining a "degree of consonance" between an attribute value and a test [Michalski and Chilausky, 1980]. In a decision tree, the evaluation considers only one attribute at a time, and therefore the decision-tree approach is inherently more sensitive to small variations of the attribute values than a rule-based approach.

The technique used in ASSISTANT nevertheless often leads to an improvement of accuracy in classifying testing examples, which can be attributed to the previously mentioned phenomenon of "overfitting." More general descriptions (due to pruning) may avoid misclassifications produced by overly specific descriptions.

Table 3–5 presents the complexity and diagnostic accuracy of decision trees generated by ASSISTANT, with and without the tree-pruning mechanism, from the same data as used in the experiments with AQTT-15 [Kononenko, Bratko, and Roskar, 1986]. It should be noted, however, that although the training and testing examples were drawn from the same data, the specific examples used for training and testing by AQTT-15 and ASSISTANT were different, and therefore the results listed in Table 3–5 are not totally comparable with those in Table 3–4.

Table 3–5: Results from the ASSISTANT program

Disease type	Tree type	Complexity		Diagnostic accuracy
		#Leaves	#Nodes	
Lymphatic cancer	Complete	22	38	76%
	Pruned	14	25	77%
Breast cancer	Complete	63	120	67%
	Pruned	9	16	72%
Primary tumor	Complete	90	188	41%
	Pruned	18	35	46%

To compare the relative complexity of rulesets generated by AQTT-15 with that of the corresponding decision tree, one may compare the total number of rules with the number of leaves in the tree. (As mentioned before, a decision tree can be turned into a ruleset by tracing paths from the root to individual leaves, and each such path corresponds to one rule. Notice, however, that comparing the number of conditions in a collection of rulesets with the number of nodes in a tree is not very meaningful, because an attribute assigned to a node in the tree will be repeated several times in the corresponding ruleset).

Comparing results in Table 3–5 with those in Table 3–4, in the first two domains the diagnostic accuracy of the selected rule representation ("top rule") and the pruned decision tree is roughly similar (82% versus 77% for lymphatic cancer, and 68% versus 72% for breast cancer). The striking difference is in terms of complexity. In both domains, the complexity of the rule representation is substantially lower than that of the pruned decision tree (four rules versus 14 leaves for lymphatic cancer; two rules versus nine leaves for breast cancer). In the domain of primary tumor, the pruned decision tree and the chosen rule representation (unique>1) were relatively close in terms of accuracy (46% versus 41%), and comparable to the performance of specialists (42%), but in terms of complexity, the pruned decision tree was considerably simpler (18 leaves versus 42 rules). Notice, however, that the pruned tree has only 18 leaves, while the number of decision classes is 22. This means, that for four diagnostic classes there are no corresponding leaves; i.e., these diagnostic classes have no representation, and instances from these classes cannot be recognized by the decision tree.

The discussion above explored some of the aspects of the rule-based and decision-tree-based approaches. A number of other issues were not considered, such as learning and testing efficiency, the extensibility and modifiability of representations, their cognitive comprehensibility, or the use of background knowledge in learning. While the analysis of representational aspects of the two approaches does not depend on the specific application domain, and represents a valid finding, experimental results should be viewed only as a few datapoints. Further experiments are needed for making a more conclusive evaluation.

3.9 CONCLUSION AND TOPICS FOR FUTURE RESEARCH

Unlike conventional representations, which try to describe all concept instances in one explicit structure, the proposed two-tiered (TT) representation describes explicitly only the "first approximation" of a concept. Finer aspects, less typical instances and context dependence are defined implicitly, through a matching procedure and inference rules. The explicit part of a concept description is called the base concept representation (BCR); and the implicit part is called the inferential concept interpretation (ICI).

The learning method described is the first and limited implementation of the idea of TT representation. To determine a concept's BCR, it first employs a conventional program to learn a consistent and complete (CC) concept description. This CC description is then reduced by removing from it rules in the increasing order of their importance (the TRUNC procedure). The truncated description that scores best on a description quality measure is selected as the BCR.

The rule truncation is a specialization operation, and therefore all truncated descriptions are incomplete (with regard to training instances). An opposite method would be to remove individual conditions from the rules and merge identical or closely related rules, which is a generalization operation. Such a process would produce inconsistent concept descriptions. One might expect that an application of both, the specialization and generalization operations, may lead to a better concept representation than when only one type of operation is used. To test this hypothesis, such a method has recently been developed. The experimental results have demonstrated that the resulting descriptions are indeed simpler and give better performance [Bergadano, *et al.*, 1988b; 1988c; 1990]. Future research might explore different methods of applying specialization and generalization operators to a CC description, and also address the problem of directly determining the BCR from examples. There is also a need for applying more advanced description quality measures (e.g., [Zhang and Michalski, 1989]).

In the method, the ICI consists of a procedure for flexible matching, which measures a "fit" between an instance and candidate descriptions. Due to this procedure, an incomplete concept description may still classify correctly training examples. The current method, however, does not address the issue of employing inference rules for reasoning about concept boundaries and handling context dependence. These problems are important tasks for future research. Further work may employ more advanced procedures for flexible matching and may investigate the problem of automatically determining the "best" interpretation method.

The system AQTT-15, implementing the current method, was experimentally applied to learning diagnostic rules in three medical domains. Concept descriptions obtained by the method were substantially simpler than the original CC descriptions, and at the same time performed better in diagnosing new cases. In all three domains, the diagnostic accuracy was comparable with that of specialists in these domains.

The rule-based method employed in AQTT-15 was compared with the decision-tree-based method used in ASSISTANT, a descendant of ID3. Although a ruleset can be converted to a logically equivalent decision tree, and vice versa, it has been shown that the rule representation has pragmatically greater representational power than decision trees. For some problems, a ruleset can be significantly simpler than the equivalent decision tree. It has been also shown that a decision tree may suggest a nonexisting dependence of the concept description on some attributes.

The flexible matching procedure may allow an incomplete rule representation to classify correctly training examples that were "uncovered" by the truncation process. Such a procedure does not apply to the decision tree representation, because it is difficult to measure a fit between an instance and the whole concept description in a decision tree (unless the tree is transformed to a ruleset). Consequently, truncated trees cannot avoid producing errors on some training examples.

Both programs have been experimentally applied to the same problems in three medical domains. In domains, in which training examples had relatively high quality and concept descriptions had strong patterns (lymphatic cancer and breast cancer), the diagnostic accuracy of both representations was high, but the AQTT-15's rules were significantly simpler than the ASSISTANT's decision tree. In the domain where data were of poor quality and there were no strong patterns (location of primary tumor), the diagnostic accuracy of rules and decision trees was quite low (trees performed somewhat better than rules), although comparable with that of humans specialists. The pruned decision tree was considerably simpler than the rule representation. It did not provide, however, any representation for four out of 22 diagnostic classes.

One general conclusion from experimental results seems to be that the proposed method offers significant advantages over conventional methods that use complete and consistent concept representations. Concept descriptions that it produces may be much simpler, while their performance on classifying new examples may also be higher. More research is needed to test these conclusions in other domains and across different application areas.

Knowledge representation used in AQTT-15 is limited to attributional descriptions. To extend the proposed method to learning structural descriptions, one could replace the AQ15 module by the INDUCE 3 learning program [Hoff, Michalski, and Stepp, 1983], or its incremental learning version, INDUCE 4 [Mehler, Bentrup, and Riedesel, 1986]. There would also be a need to develop a flexible matching procedure for structural descriptions and to implement a corresponding TRUNC procedure. The current method has concentrated on problems of learning TT representations of a relatively small class of concepts. Future work might address the problem of learning TT representations of a large system of concepts, and the related issue of the inheritance and sharing parts of the BCR and the ICI among different concepts.

Concluding, we would like to emphasize the importance to artificial intelligence and cognitive science of the problem of learning flexible concepts. As most human concepts are flexible, the issues of their representation, learning, and use in reasoning constitute a major part of the agenda for future research in these fields.

ACKNOWLEDGMENTS

The author thanks members of the Advanced Machine Learning Seminar and the Center for Artificial Intelligence at George Mason University for the discussion and criticism of ideas presented here. Special thanks go to Jianping Zhang, Gheorghe Tecuci, and Pawel Stefanski for their careful reading of the manuscript and many insightful comments, and Dr. Elizabeth Marchut-Michalski for important stylistic suggestions that improved the final version of this chapter.

The work described was done partially in the Laboratory for Artificial Intelligence at the University of Illinois at Urbana-Champaign, and partially in the Center for Artificial Intelligence at George Mason University. Various aspects of research were supported by the National Science Foundation under grant No. DCR 084-06801, the Office of Naval Research under grants N00014-88-K-0226 and N00014-88-K-0397, and the Defense Advanced Research Projects Agency under the grant administered by the Office of Naval Research No. N00014-87-K-0874.

References

Alen, S.W., Brooks, L.R., and Norman, G.R. 1988. "Effect of Prior Examples on Rule-based Diagnostic Performance," *Proceedings of the 29th Annual Meeting of the Psychologic Society*, November, 1988.

Bareiss, E.R., Porter, B.W., and Craig, C.W. 1990. "Protos: An Exemplar-based Learning Apprentice," in *Machine Learning: An Artificial Intelligence Approach, Volume III*, Y. Kodratoff and R.S. Michalski (eds.), Morgan Kaufmann Publishers.

Barsalou, L.W. and Medin, D.L. 1986. "Concepts: Fixed Definitions or Dynamic Context-dependent Representations," *Cahiers de Psychologie Cognitive*, 6, pp. 187–202.

Bergadano, F., Matwin, S., Michalski, R.S., and Zhang, J. 1988a. "Measuring Quality of Concept Descriptions," *Proceedings of the Third European Working Session on Learning*, D. Sleeman, pp. 1–14, Pitman, London, (an extended version was published under the title "A Measure of Quality of Descriptions," *Reports of the Machine Learning and Inference Laboratory*, No. 88–3, Artificial Intelligence Center, George Mason University, 1988.)

———, 1988b. "Learning Flexible Concept Descriptions Using a Two-tiered Knowledge Representation: Ideas and a Method," *Reports of the Machine Learning and Inference Laboratory*, No. 88–4, Artificial Intelligence Center, George Mason University.

———, 1988c. "Learning Flexible Concept Descriptions Using a Two-tiered Knowledge Representation: Implementation and Experiments," *Reports of the Machine Learning and Inference Laboratory*, No. 88–35, Artificial Intelligence Center, George Mason University.

———, 1990. "Learning Two-tiered Descriptions of Flexible Concepts," submitted to *Machine Learning*.

Bratko, I., Mozetic, I., and Nada, L. 1989. *Kardio: A Study in Deep and Qualitative Knowledge for Expert Systems*, MIT Press, Cambridge, MA.

Carey, S. 1985. *Conceptual Change in Childhood*, MIT Press, Cambridge, MA.

Cestnik, B., Kononenko, I., and Bratko, I. 1987. "ASSISTANT 86: A Knowledge Elicitation Tool for Sophisticated Users," *Proceedings of the second European Working Session on Learning*, I. Bratko and N. Lavrac (ed.), Bled, Yugoslavia, May 1987.

Estes, W.K. 1986. "Memory Storage and retrieval Processes in Category Learning," *Journal of Experimental Psychology: General*, 115, pp. 155–175.

Flannagan, M.J., Fried, L.S., and Holyoke, K.J. 1986. "Distributional Expectations and the Induction of Category Structure," *Journal of Experimental Psychology: Learning, Memory and Cognition*, No. 12, pp. 241–256.

Gentner, D. and Landers, R. 1985. *Analogical Reminding: A Good Match is Hard to Find*, paper prepared for the Panel on Commonsense Reasoning at the International Conference on Systems, Man and Cybernetics, Tucson, Arizona.

Gross, P.K. 1988. "Incremental Multiple Concept Learning Using Experiments," *Proceedings of the Fifth International Conference on Machine Learning*, Ann Arbor, June 12–14.

Hoff, W., Michalski, R.S., and Stepp, R.E. 1983. "INDUCE. 3: A Program for Learning Structural Descriptions from Examples," Report ISG 83-4, UIUCDCS-F-83–904, Dept. of Computer Science, University of Illinois, Urbana.

Hofstadter, D.R. 1985. "Analogies and Roles in Human and Machine Thinking," in *Metamagical Themas: Questing for the Essence of Mind and Pattern*, Basic Books, Inc.

Holyoke, K.J. 1985. "The Pragmatics of Analogical Transfer," *The Psychology of Learning and Motivation*, 19.

Hong, J., Mozetic, I., and Michalski, R.S. 1986. "AQ15: Incremental Learning of Attribute-based Descriptions from Examples, The Method and User's Guide," Report ISG 86–5, UIUCDCS-F-86-949, Dept. of Computer Science, University of Illinois, Urbana.

Iba, W., Woogulis, J., and Langley, P. 1988. "Trading Simplicity and Coverage in Incremental Concept Learning," *Proceedings of the Fifth Intern. Conference on Machine Learning*, Ann Arbor, June 12–14, 1988.

Kempler-Nelson, D.G. 1984. "The Effect of Intention on What Concepts are Acquired," *Journal of Verbal Learning and Verbal Behavior*, No. 23, pp. 734–759.

Lakoff, G. 1987. *Women, Fire and Dangerous Things: What Categories Tell Us About the Nature of Thought*, Chicago University Press.

Lenat, D.B., Hayes-Roth, F., and Klahr, P. 1979. "Cognitive Economy in Artificial Intelligence Systems," *Proc. IJCAI*, Tokyo, Japan.

McCloskey, M. and Glucksberg, S. 1978. "Natural Categories: Well-defined or Fuzzy Sets?," *Memory and Cognition*, No. 6, pp. 462–472.

Medin, D.L. 1989. "Concepts and Conceptual Structure," *American Psychologist*, vol. 44, pp. 1469–1481.

Medin, D.L. and Smith, E.E. 1984. "Concepts and Concept Formation," in *Annual Review of Psychology*, M.R. Rosenzweig and L.W. Porter (Eds.), No. 35, pp. 113–118.

Mehler, G., Bentrup, J., and Riedesel, J. 1986. "INDUCE. 4: A Program for Incrementally Learning Structural Descriptions from Examples," *Reports of Intelligent Systems Group 86*, Department of Computer Science, University of Illinois, Urbana.

Michalski, R.S. 1969. "On the Quasi-Minimal Solution of the General Covering Problem," *Proceedings of the V International Symposium on Information Processing* (FCIP 69), Vol. A3 (Switching Circuits), Bled, Yugoslavia, pp. 125–128.

——— , 1973. "AQVAL/1—Computer Implementation of a Variable-valued Logic System VL1 and Examples of its Application to Pattern Recognition," *Proceedings of the First International Joint Conference on Pattern Recognition*, Washington, DC, pp. 3–17, October 30–November 1, 1973.

——— , 1974. "Variable-valued Logic: System VL1," *Proceedings of the 1974 International Symposium on Multiple-Valued Logic*, West Virginia University, Morgantown, pp. 323–346, May 29–31.

——— , 1978a. "A Planar Geometrical Model for Representing Multidimensional Discrete Spaces and Multiple-Valued Logic Functions," Report No. 897, Department of Computer Science, University of Illinois, Urbana, January.

——— , 1978b. "Designing Extended Entry Decision Tables and Optimal Decision Trees Using Decision Diagrams," Report No. 898, Department of Computer Science, University of Illinois, Urbana, March 1978.

——— , 1983. "Theory and Methodology of Inductive Learning." In R.S. Michalski, J.G. Carbonell, T.M. Mitchell (Eds.), *Machine Learning: An Artificial Intelligence Approach*, Morgan Kaufmann, San Mateo.

——— , 1986. "Concept Learning," *Encyclopedia of Artificial Intelligence*, John Wiley & Sons.

——— , 1986. "Two-tiered Concept Representation, Analogical Matching and Conceptual Cohesiveness," Invited paper for the Workshop on Similarity and Analogy, Allerton House, University of Illinois, June 12–14, 1986. [An extended and improved version is in *Analogy and Similarity*, S. Vosniadou and A. Ortony (eds.), Cambridge University Press, 1989.]

——— , 1987. "How to Learning Imprecise Concepts: A Method for Employing a Two-tiered Knowledge Representation in Learning," *Proceedings of the Fourth International Workshop on Machine Learning*, University of California at Irvine, June 22–25, 1987.

——— , 1989. "Dynamic Recognition: An Outline of a Theory on How to Recognize Concepts without Matching Rules," *Reports of Machine Learning and Inference Laboratory*, Center for Artificial Intelligence, George Mason University, June 1989.

Michalski, R.S. and Chilausky, R.L. 1980. "Learning by Being Told and Learning from Examples: An Experimental Comparison of the Two Methods of Knowledge Acquisition in the Context of Developing an Expert System for Soybean Disease Diagnosis," *International Journal of Policy Analysis and Information Systems*, Vol. 4, No. 2, pp. 125–160.

Michalski, R.S. and Larson, J. 1975. "AQVAL/1 (AQ7) User's Guide and Program Description," Report No. 731, Dept. of Computer Science, University of Illinois, Urbana.

——— , 1978. "Selection of Most Representative Training Examples and Incremental Generation of VL1 Hypotheses: The Underlying Methodology and the Description of Programs ESEL and AQ11," *Reports of the Department of Computer Science*, University of Illinois, No. 867, Urbana, May 1978.

Michalski, R.S. and McCormick, B.H. 1971. "Interval Generalization of Switching Theory," *Reports of the Department of Computer Science*, No. 442, University of Illinois, Urbana.

Michalski, R.S. and Negri, P. 1977. "An Experiment on Inductive Learning in Chess End Games," *Machine Representation of Knowledge, Machine Intelligence 8*, E.W. Elcock and D. Michie (eds.), Ellis Horwood Ltd., New York, pp. 175–192.

Michalski, R.S. and Stepp, R.E. 1983. "Learning from Observations: Conceptual Clustering," in *Machine Learning: An Artificial Intelligence Approach*, R.S. Michalski, J.G. Carbonell, T.M. Mitchell (eds.), San Mateo, Morgan Kaufmann, 1983.

Michalski, R.S. and Winston, P.H. 1986. "Variable Precision Logic." *Artificial Intelligence Journal*, No. 29.

Michalski, R.S., Mozetic, I., Hong, J., and Lavrac, N. 1986. "The AQ15 Inductive Learning System: An Overview and Experiments," *Proceedings of the American Association for Artificial intelligence Conference (AAAI).*

Mitchell, T.M. 1982. "Generalization as a Search," *Artificial Intelligence*, 1, 203–226.

Mozetic, I. 1986. "Knowledge Extraction through Learning from Examples" In T.M. Mitchell, J.G. Carbonell, R.S. Michalski (eds.), *Machine Learning: A Guide to Current Research*, Kluwer Academic Publishers.

Mumdani, E.H. and Gaines, B.R. (eds.). 1981. *Fuzzy Reasoning and its Applications*, Academic Press.

Murphy, G.L. and Medin, D.L. 1985. "The Role of Theories in Conceptual Coherence," *Psychological Review*, No. 92.

Nosofsky, R.M. 1988. "Exemplar-based Accounts of Relations between Classification, Recognition and Typicality," *Journal of Experimental Psychology: Learning, Memory and Cognition.*

Popper, K.R. 1979. *Objective Knowledge: An Evolutionary Approach*, Oxford at the Clarendon Press.

Quinlan, R.J. 1983. "Learning Efficient Classification Procedures and their Application to Chess End Games," in *Machine Learning: An Artificial Intelligence Approach*, R.S. Michalski, J.G. Carbonell, T.M. Mitchell (eds.), Morgan Kaufmann, San Mateo.

——— , 1990. "Probabilistic Decision Trees," in *Machine Learning: An Artificial Intelligence Approach, Volume III*, Y. Kodratoff and R.S. Michalski (eds.), Morgan Kaufmann Publishers.

Reinke, R.E. and Michalski, R.S. 1988. "Incremental Learning of Decision Rules: A Method and Experimental Results," in J.E. Hayes, D. Michie, J. Richards (eds.), *Machine Intelligence*, Oxford University Press.

Rosch, E. and Mervis, C.B. 1975. "Family Resemblances: Studies in the Structure of Categories," *Cognitive Psychology*, No. 7.

Schank, R.C., Collins, G.C., and Hunter, L.E. 1986. "Transcending Induction Category Formation in Learning," *The Behavioral and Brain Sciences*.

Smith, E.E. and Medin, D.L. 1981. *Categories and Concepts*, Harvard University Press, Cambridge, MA.

Sowa, J.F. 1984. *Conceptual Structures: Information Processing in Mind and Machine*. Addison-Wesley Publishing Company.

Watanbe, S. 1969. *Knowing and Guessing: A Formal and Quantitative Study*, Wiley Pub. Co.

Wittgenstein, L. 1922. *Tractatus Logico-Philosophicus*, London: Routledge and Kegan.

Winston. P.H. 1975. "Learning Structural Descriptions from Examples," *The Psychology of Computer Vision*, Winston P.H. (Ed.), McGraw Hill, New York, chapter 5.

Zhang, J. and Michalski, R.S. 1989. "A Description Preference Criterion in Constructive Learning," *Proceedings of the Sixth International Workshop on Machine Learning*, Cornell University, Ithaca, New York, June 26–27, 1989.

Zadeh, L.A. 1965. "Fuzzy Sets," *Information and Control*, 8.

——— , 1976. "A Fuzzy-Algorithmic Approach to the Definition of Complex or Imprecise Concepts," *International Journal of Man-Machine Studies*, 8.

——— , 1978. "Fuzzy Sets as a Basis for a Theory of Possibility," *Fuzzy Sets and Systems 1*.