RESEARCH IN MACHINE LEARNING: RECENT PROGRESS, CLASSIFICATION OF METHODS AND FUTURE DIRECTIONS

by

R. S. Michalski
Y. Kodratoff

Machine Learning: An AI Approach, Vol. III, Y. Kodratoff and R.S. Michalski (Eds)
San Mateo, CA. Morgan Kauffman Publ., pp. 3-30, June 1990.

Γ 7(⌐ –

1

RESEARCH IN MACHINE LEARNING: Recent Progress, Classification of Methods and Future Directions

Ryszard S. Michalski
George Mason University

Yves Kodratoff
French National Research Center (CNRS)
and *George Mason University*

Abstract

The last few years have witnessed a remarkable expansion of research in machine learning. The field has gained an unprecedented popularity, several new areas have developed, and some previously established areas have gained new momentum. While symbolic methods, both empirical and knowledge-intensive, in particular, inductive concept learning and explanation-based methods, continued to be exceedingly active (Parts 2 and 3 of the book, respectively), sub-symbolic approaches, especially neural networks, have experienced tremendous growth (Part 5). Unlike past efforts that concentrated on single learning strategies, the new trend has been to integrate different strategies, and to develop cognitive learning architectures (Part 4). There has been an increasing interest in experimental comparisons of various methods, and in theoretical analyses of learning algorithms. Researchers have been sharing the same data sets, and have applied their techniques to the same problems in order to understand relative merits of different methods. Theoretical investigations have brought new insights into the complexity of learning processes (Part 6).

This chapter gives a brief account of the recent progress and prospective research directions in the field, attempts to clarify some basic concepts, proposes a multicriteria classification of learning methods, and concludes with a brief description of each chapter.

1. Introduction

One of the most striking differences between how people and computers work is that humans, while performing any kind of activity, usually simultaneously expend efforts to improve the way they perform it. This is to say, that human performance of any task is inseparably intertwined with a learning process, while current computers are strictly executors of procedures supplied to them. They may execute very efficiently, but they do not self-improve with experience.

Research in machine learning has been concerned with building computer programs such that by executing them, externally supplied information will be generalized and/or improved in some sense. So far, this input information (examples, facts, descriptions, etc.) has been typically typed in by a human instructor. Future machine learning programs will undoubtedly be able to receive their inputs directly from the environment, through a variety of sensory devices.

The great appeal of this field to its practitioners is that machine learning offers an immense diversity of research tasks and testing grounds. This diversity is due to the fact that learning can accompany any kind of problem solving or process, and thus can be studied in many different contexts, such as decision making, classification, sensory signal recognition, problem solving, task execution, control, or planning.

This continual appeal of the field has been enhanced recently by the fact that progress in machine learning has become central to the development of the field of artificial intelligence as a whole, and affects almost all of its subareas. In particular, the work in this field has importance for expert systems development, problem solving, computer vision, speech understanding, autonomous robotics, conceptual analysis of databases, and intelligent tutoring systems. Consequently, the development of powerful learning systems may ultimately open an unprecedented range of new applications (e.g., Michalski, 1986).

Research on building learning programs goes back to almost the beginning of the computer area. After the first significant burst of activities on perceptrons and self-organizing systems in the fifties and the first few years of the sixties, the field has been growing slowly but steadily. Some early successes include, for example, the Samuel's checker's program (Samuel, 1959), Winston's program for learning structural descriptions (Winston, 1970, 1975), the Meta-DENDRAL program for heuristic rule formation (Buchanan, Feigenbaum and Sridharan, 1972), the AM and EURISKO discovery programs (Lenat, 1977, 1983), AQ-11 for diagnostic rule learning (Michalski & Chilausky, 1980), LEX for learning symbolic integration (Mitchell, Utgoff and Banerji, 1983), and CLUSTER for conceptual clustering (Michalski and Stepp, 1983).

These successes and the ever present challenge to build powerful learning systems, have exerted strong pressure to expand the activities in this field. The first machine learning workshop was held at Carnegie Mellon University in 1980. This workshop and the publication in 1983 of the first volume of Machine Learning (Michalski, Carbonell and Mitchell, 1983) have marked a breaking point. These two events have given the field a clear identity and a sense of direction, which have stimulated the rapid growth that has continued unabated since then.

There have been subsequent workshops and conferences: at the University of Illinois at Champaign-Urbana in 1983; at Rutgers University in 1985; at the University of California at Irvine in 1987; at the University of Michigan in 1988, and at Cornell University in 1989. In 1986, Machine Learning Volume II has appeared (Michalski, Carbonell and Mitchell, 1986). In response to the growing need for an adequate forum for presenting research progress, *Machine Learning Journal* was established in 1986.

There have also been numerous workshops and meetings on special topics, such as computational learning theory (COLT 88 and 89), explanation-based learning (AAAI workshop at Stanford U., 1988), connectionist models of learning (e.g., summer schools at CMU in 1986 and 1988, and a number of international conferences), and knowledge discovery in databases (IJCAI-89 workshop in Detroit).

In parallel, there has been a rapid increase of interest in machine learning in Europe, as signified by many activities, meetings and conferences. Among the most noteworthy were the European Working Sessions on Learning (Orsay 86, Bled 87, Glasgow 88, and Montpellier 89), the International Meeting on Advances in Learning in Les Arcs in 1986, the workshop on Knowledge Representation and Organization in Machine Learning (KROML 87), Workshop on Machine Learning, Meta-Reasoning and Logic (Sesimbra 88), and Summer Schools in Machine Learning (Les Arcs 88 and Urbino 89), International Schools for the Synthesis of Knowledge (ISSEK 87 and 89). To reflect these activities, this volume includes a significant number of non-US contributions.

2. Recent Developments

The last few years have witnessed both a continuation of the major traditional research approaches and a rapid increase of interest in several new methodologies. The most active research area in recent years has continued to be *symbolic empirical learning* (SEL). This area is concerned with creating and/or modifying general symbolic descriptions, whose structure is unknown *a priori*. This type of learning can be contrasted with, e.g., learning weights assigned to connections in a given neural net, or coefficients of equations in a predefined form. The descriptions are created on the basis of examples or specific facts. The word "empirical" signifies the fact that the learning process does not require much prior knowledge of the learner (if the process relies on a large amount of explicitly stated prior knowledge, then we have *knowledge-intensive symbolic learning*).

An important criterion underlying SEL methods is that knowledge created by a learning program is supposed to be easy to interpret and comprehend by humans. This means that there is a concern to make knowledge representations simple in terms of the structures used and the number of operators involved. It also means that the concepts employed in the descriptions should directly correspond to those used by human experts. This criterion is sometimes called the *comprehensibility principle* (Michalski, 1983). Typical knowledge structures employed in the SEL systems include commonly used symbolic representations, such as logic-based descriptions, rules, decision trees, semantic networks, equations, frames and grammars. Due to the comprehensibily criterion, the SEL systems can be particularly useful in applications in which people need to fully comprehend the results of learning, for example, in technical, medical or agricultural diagnosis, decision making, planning, economical or political analysis, discovery of knowledge in databases, prediction, etc.

The most common topic in SEL is developing concept descriptions from concept examples. The machine learning bibliography (MLB; the last chapter of the book), which contains 1050 entries covering the period 1985-89, lists about 190 publications on this topic. Other major topics in SEL include qualitative discovery, conceptual clustering and empirical sequence prediction. MLB lists another 130 publications on these topics, thus together, there are about 320 papers listed in MLB on symbolic empirical learning.

As mentioned above, empirical methods typically use relatively little background knowledge (BK), by which we mean the relevant domain-dependent knowledge, such as facts or rules characterizing the application domain, and domain-independent knowledge, such as general definitional knowledge, commonsense knowledge and explicit rules of inference, which learner can bring to bear in the process of learning. In SEL systems, the BK may include merely information about the value sets and types of attributes or terms (descriptors) used, the constraints on the attributes, preference criteria or biases for judging candidate solutions, etc. The domain-dependent information can be introduced to a program when it is applied to a particular problem, and therefore it is relatively easy to develop a general-purpose empirical learning program. The AQ family of rule learning programs (e.g., Michalski, 1973; and chapter 3), and the ID3-type decision tree learning programs (Quinlan, 1979; and chapter 5) are examples of such general-purpose SEL systems. The AQ programs generate rules by manipulating knowledge structures according to rules of inference and knowledge transformation. The ID3 type systems create a decision tree by a recursive selection of attributes from a given set. The attribute selection is based on statistical considerations (e.g., the minimum entropy rule), rather than on explicit rules of inference.

The primary inference type used by SEL methods is empirical induction. This form of induction (as other forms, such as constructive induction and abduction, see the next section) is a falsity-preserving, rather than truth-preserving inference. Therefore, the results of SEL methods are generally hypotheses, which need to be validated by further experiments. This feature is often viewed as an important weakness of the empirical methods. It reflects the intrinsic uncertainty of any process of creating new knowledge about the world, and therefore is unavoidable in principle. The only ways to circumvent it, is to restrict the learning process either to coping existing knowledge, or to strict deductions from knowledge that has been tested and assumed to be true. Such an *a priori* knowledge has to be encoded into the system before any learning can occur (see analytic methods in the next section).

Another major weakness of the SEL methods is that the knowledge learned by them represents relations expressed merely in terms of attributes or concepts either directly specified in the input data, or closely related to them (an empirical program may include procedures for transforming the initial description space). Because the methods rely primarily on the input information, rather than on background knowledge, they cannot discover complex relationships or causal dependencies, which require high level terms or concepts, not provided by the input.

The fact that symbolic empirical methods do not use/require much background knowledge is appealing to many researchers. Examples or observations are often easily available from existing databases, or can be measured by sensors. There is no need for debugging and handcrafting large amounts of knowledge into the system. Consequently, empirical learning systems are readily applicable to a wide spectrum of practical problems. In addition, because the results are usually easy to interpret (in contrast to sub-symbolic systems; see below), the methods are particularly attractive in application areas where understandability of the results is an important factor. A selection of research in symbolic empirical learning is presented in Part 2, chapters 3-9.

In recent years, there have been various efforts to extend the capabilities of conventional SEL systems. A considerable amount of work has been done on learning concepts from imperfect inputs, e.g., learning from examples with noise (e.g., see chapter 5). Related efforts have been concerned with learning concepts that lack precise definition and/or are context dependent (e.g., see chapter 3).

Another major extension of empirical methods addresses the problem of employing more BK in the process of inductively creating concept descriptions. The motivation is that people, due to their prior knowledge, can often create plausible inductive hypotheses from a few or just one instance. For example, if one sees a single window of a particular style in a tall skyscraper, then one does not need to look at other windows to hypothesize that all the windows in that building are of this style. The reason is that we know that windows in a building typically are made in the same style. As another example, consider a person who deceptively misinforms others about something really important. It usually would not take more than one such instance for not trusting that person in the future. Again, this is because of a common belief, that if a person lied once, it is likely that this person may continue such behavior, and because trusting such a person would carry a very high risk. Thus, by involving prior knowledge, one can create plausible inductive hypotheses from very little input information, contrary to some beliefs about inductive learning.

Also, in many applications, it is important to discover relationships that go beyond associations between inputs and outputs. In such applications, it is important to search for relationships that involve higher level concepts than those defined in the inputs, to generate and employ abstract attributes and relations, and/or to determine causal explanations of the observations. Any process of theory formation requires much background knowledge in addition to observational data.

To this end, some researchers started to work on *constructive induction*, which is a term for characterizing inductive processes that engage significant amount of BK (Michalski, 1983; see also Muggleton and Buntine, 1988; Rouveirol and Puget, 1989). Such BK may be in the form of expert-given domain knowledge rules, logical implications and equivalences, abstract concept definitions, heuristic procedures for generating new concepts, goal-oriented criteria for evaluating the importance of created knowledge, and other. This knowledge is used in the conventional, deductive manner, thus, constructive induction includes a large component of deductive inference. Equipped with appropriate BK, constructive inductive systems can change the representation of the problem or invent new attributes or concepts. As described in (Michalski, 1990; see also next section), constructive induction involves "reverse reasoning" or "tracing backward" of certain implicative rules, which are either domain-independent (tautological implications) or domain-dependent (representing domain knowledge). When domain-independent rules are primarily involved (specifically, the falsity-preserving generalization rules), then constructive induction reduces to empirical induction. When certain domain-dependent implicative relationships are "traced backward," then such induction becomes abduction (see next section). There are over 50 publications listed in the MLB in the area of constructive induction, abduction and representation change.

Other classes of empirical learning systems include parametric and heterogeneous systems. In parametric systems, the learning process involves a modification of certain parameters or weights associated with predefined structures (networks, equations, production rules, etc.). Learning in heterogeneous systems involves both a direct modification of knowledge structures and a modification of the parameters associated with these structures. The most popular and important representative of parametric systems are neural nets and connectionist systems. In those systems, the learning process typically involves a modification of the strength of the connections between units in a statically or dynamically defined network. All units often perform the same general transformation, and therefore it is easy to build very large networks of that kind. It is important to note that a modification of the strengths of connections in a neural net can lead to a change in the knowledge structure. This structural change, however, is indirect and implicit, rather than direct and explicit, like in symbolic systems. The most explored subset of heterogeneous systems are genetic algorithms and classifier systems. In those systems, the modification of the structures is done either by random changes (mutation), or semi-random changes (e.g., crossover), rather than by explicit rules of inference, like in typical symbolic systems. The weights assigned to individual production rules represent their importance or effectiveness in performing the assigned task.

Recent years have witnessed a remarkable renaissance of research on learning in neural nets. There is a rapidly growing interest in exploring their properties and potential applications. Since these systems employ a general and uniform knowledge representation, and typically use little background knowledge, it is easy to implement them and apply experimentally to a wide spectrum of problems. As they require very little guidance from a teacher, they are very appealing to many researchers.

A major limitation of neural nets and genetic algorithms is the difficulty of introducing to them large amounts of domain specific knowledge, and *explicitly* exploiting that knowledge or any feedback information in the learning process. To explain the latter, suppose that a neural network or genetic algorithm gets feedback that some example was incorrectly classified. To take care of the mistake, the system modifies its knowledge representation by stepwise corrections, rather than by an explicit analysis of the reasons for the mistake. This seems to explain why such systems tend to exhibit relatively slow rates of learning. Another weakness is the lack of transparency of the results of learning. The knowledge acquired by neural networks or conventional genetic algorithms is not in the form that people can easily understand. The comprehensibility principle has not been viewed as a major issue in implementing such systems. For that reason, they are sometimes called *subsymbolic learning systems.*

The lack of transparency is not necessarily always a problem. There are many application domains, which do not require that the knowledge learned be easy to understand. For example, it is not important to understand the control algorithm of a robot, as long as it can move its hand to the given destination and within a defined space. This weakness is a problem in areas where people need to understand the knowledge underlying the system's behavior, e.g., in diagnostic, advisory or planning systems. It can be pointed out that a genetic algorithm could potentially be applied with a high level symbolic knowledge representation (such a method would not be classified as subsymbolic any more). So far, little has been done in this direction, and therefore it is difficult to predict how successful such methods will be. Some recent experimental studies seem to suggest that in tasks involving learning "human-type" concepts from examples, symbolic empirical methods might be better than subsymbolic ones both in terms of the error rate and the complexity of the descriptions learned (Wnek et al, 1990).

The MLB lists a selection of about 50 papers on learning in connectionist systems (neural nets) and about 60 papers on learning using genetic algorithms. This book includes review chapters on subsymbolic learning systems (chapters 20 and 21).

The primary goal of the symbolic and sub-symbolic empirical learning methods, as well as the knowledge intensive constructive induction systems, is to create a "fundamentally new" knowledge from the given input information. By "fundamentally new" is meant knowledge that cannot be deduced from the knowledge already known. Therefore, such systems can be classified as *synthetic* learning systems.

Synthetic systems can be contrasted with *analytic learning* systems, whose primary goal is to analyze and transform the knowledge already possessed into more effective or "operational" form. Analytic systems rely on large amounts of prior knowledge and use deduction as the primary inference. Because such inference is truth-preserving, the results of analytic learning are as valid as the background knowledge and the input information supplied to them. Thus, if one can assume that the BK and the input facts are correct, the results of learning need not to be validated. This feature makes analytic systems very appealing to many researchers.

Last several years has seen a major and rapid growth of research in the area of analytic learning. In this area, the most active research was on explanation-based learning (EBL), which is concerned with explaining an observed example in terms of the learner's background knowledge, and then using this explanation for creating a more effective or "operational" concept description. In the "pure" EBL, the background knowledge must be complete, consistent and tractable, so that the program can deduce a consistent and unique explanation of the input example from it.

Determining such a consistent, complete and tractable BK, and handcrafting it to the program may require a substantial effort, and thus it is often not easy to apply EBL in a practical setting. Also, domain knowledge is frequently incomplete or not totally correct, and one needs to apply an inductive method to improve it or to fill in the holes. Therefore, a recent trend has been to combine EBL methods with inductive techniques in order to cope with such situations.

The above efforts on extending analytic methods by adding to them inductive learning capabilities seem to be a mirror image of research on constructive induction that extends empirical methods by introducing to them more domain knowledge and more powerful deductive capabilities. One may also note that EBL or other analytic methods do not address the problem of learning abstract concept descriptions or specifications, as these are assumed to be given to the system. For such problems, inductive methods are necessary. MLB lists about 170 publications on explanation-based learning and related methods. A selection of research in this area is presented in Part 3, chapters 10-14.

There has been a growing understanding that future learning systems should not be centered around single learning strategy in acquiring knowledge, but should combine several strategies in a goal-oriented fashion. In this context, there is an interest in moving from single-strategy-oriented systems to multistrategy systems (Michalski, 1990). Most work in this direction has so far been concentrated on developing learning systems that integrate empirical and explanation-based learning methods, and building large-scale cognitive architectures. MLB lists about 65 publications in this area. A selection of research is described in chapters 15-19.

In view of the proliferation of different learning methods and paradigms, there has been a growing interest in experimental investigation and comparison of various learning methods. The MLB lists about 60 publications in this area. Several chapters of this volume report results of various experimental comparisons of learning methods.

A significant amount of activity and major progress have also occurred in the area of computational theory of machine learning algorithms. A large portion on this work has been concerned with probably approximately correct (PAC) concept learning methods, originally introduced by Valiant (see chapter 22). There is a considerable and growing interest in this area, as evidenced by 120 papers listed in MLB. Although results of this research add to the general understanding of learning processes, its range of applicability is still arguable (Buntine, 1989; Bergadano and Saitta, 1989). A selection of research in this area is in chapters 22 and 23.

Concluding, one might ask about the expected future role of the symbolic versus subsymbolic approaches in machine learning. It appears that both these approaches will be useful for various classes of applications. Ultimately, however, machine learning systems will have to be able to acquire and use vast amounts of diverse human knowledge, perform all kinds of inference, and be able to explain their knowledge to people. These types of qualities and functions seem to be easier to implement in symbolic systems. Therefore, some authors believe that in the long run, it may well be that the symbolic systems rather than sub-symbolic will play a "preeminent" role in the field of machine learning (Michalski and Littman, 1989).

3. Synthetic versus Analytic Learning

As mentioned above, learning processes can be classified on the basis of their main goal into synthetic and analytic. Synthetic learning aims primarily at creating new or better knowledge, and analytic learning aims primarily at reformulating given knowledge into a better form. Synthetic learning employs induction as the primary inference, and analytic learning employs deduction as the primary inference. To explain the difference between these two classes of learning processes, one needs to clarify the meaning of these two types of inference.

Inductive inference has been often misunderstood, and different authors have defined it in different ways. One view is that induction is merely an empirical generalization of examples without using much prior knowledge. Another view is that induction includes every inference process under uncertainty, i.e., any inference that is not strictly deductive (e.g., Holland et al, 1986). Both views are somewhat extreme. The first one is clearly inconsistent with the mainstream scientific thoughts on this subject, which go back to Aristotle (see reference), while the second seems to be overly general. Our view is that induction is simply a process opposite of deduction. While deduction is a derivation of consequents from given premises, induction is a process of hypothesizing premises that entail given consequents.

The above is a very simplified characterization of these inference types, which does not explain the role of the reasoner's background knowledge and other important issues involved in these processes. Let us then analyze these processes in a more detailed way.

Consider a relationship stating that P and BK entails C:

$$P \;\&\; BK \vdash C \qquad (1)$$

where P is a premise, BK stands for the relevant reasoner's background knowledge and C is a consequent. If the reasoner observes P (e.g., it is an newly perceived fact), then using its prior knowledge the reasoner can conclude by deduction that C must be true. A weak form of deduction is when the entailment does not always hold, an observation matches P only partially, or BK is not totally certain. On the other hand, if the reasoner observes C, then it may hypothesize P, because it would entail (explain or generalize) the observation. This is an inductive inference. Its type depends on the nature and the role of BK, particularly, on whether the BK includes only domain-independent knowledge (e.g., tautological implications), domain-dependent knowledge (reflecting the properties observed in the world), or both.

Based on these distinctions, inductive processes can be classified into three classes:

• *Empirical induction*, in which the system creates an inductive hypothesis on the basis of the given facts, using primarily domain independent BK (tautological implications or valid logical statements). A particularly important instance of empirical induction is empirical reasoning from particulars to universals, i.e., *empirical inductive generalization.* It can be viewed as a special form of reasoning from effects to premises, which is guided by "generalization rules" (Michalski, 1983). These are *domain-independent* falsity-preserving rules, which reverse certain tautological implications. Domain-dependent knowledge plays only a supportive role, for example, that of providing the constraints on the set of possible attribute values, specifying relations that hold among these attribute values and influencing the preference criterion.

Consider, for example, a tautological implication (a domain-independent rule): $(\forall x \in S, P(x)) \Rightarrow (P(e), e \in S)$, which states that if a property holds for all instances then it must hold for some particular instance. Tracing such a rule backward, that is, concluding that P holds for all x from the set by observing that it holds for one or more elements of this set is an empirical induction.The plausibility of such induction depends on how *typical* event *e* is in the set S (Collins and Michalski, 1989), and on the general knowledge about the property P. For example, if one is very happy with a car made by a particular company, then one may conclude that all cars made by this company are good and recommend the company to a friend. The strength of this conclusion will depend on the degree to which the car can be viewed as typical of this company, and on the degree of belief that the quality of a product is conserved across all products of a company. If the latter belief is non-existent or weak, then the conclusion is highly empirical. If the belief represents a relatively strong background knowledge, then the inference becomes a form of constructive induction (see later). Finally, if the belief represents proven knowledge, then the inference becomes an ordinary deduction.

• *Abduction,* a form of reasoning introduced by Peirce in his classic and influential treatise "Elements of Logic" (see the reference under Peirce). Abduction, as defined by Peirce (also referred to by him as *retroduction*), is "the operation of adopting an explanatory hypothesis that would account for all the facts or some of them." In Peirce's abduction such a hypothesis can be obtained by tracing backward rules that express relationships among facts about the domain, i.e., *domain-dependent* rules. For example, consider a (domain-dependent) rule: $\forall x \in \{\text{locations}\}$ $[\text{fire}(x) \Rightarrow \text{smoke}(x)]$. Tracing such a rule backward, i.e., hypothesizing that there may be fire, if one observes smoke is abduction.

Usually, domain knowledge does not include many rules that point out to the same consequent. For that reason Peirce and other workers on abduction usually ignore the issue of the preference criterion for choosing an explanatory hypothesis. This issue would arise, however, when more than one candidate hypothesis can be generated abductively. Therefore, the general formulation of induction (see eq. 2, below) includes the preference criterion as an important component.

• *Constructive induction (knowledge-based induction),* in which creating a hypothesis may involve both the domain-dependent and the domain-independent BK. In this formulation, constructive induction includes both constructive inductive generalization, which uses background knowledge rules to create higher level generalizations and abduction that produces explanations. The domain knowledge can be used both deductively and inductively, i.e., for forward and backward tracing of implicative relationships. For example, suppose that the BK of a learner includes a rule that a high degree automation in a company leads to a high reliability of the product. If one learns that a particular company has completely automated automobile production, and observes a car of that company that is highly reliable, then one may hypothesize that all cars of this company are very reliable (this would be a forward tracing of the BK rule; the example is not actually needed, but it helps remove the possibility that the automation was not implemented properly). On the other hand, if one sees a highly reliable car, one may create a hypothetical explanation that the company that produced it may be highly automated (this would be a backward tracing of the BK rule).

As another example of combining empirical generalization with deduction, suppose one notices several students working late into the night in the AI Laboratory. A conclusion that all students in the laboratory are hard working is a constructive induction. It involves a step of empirical generalization (from several students to all) and a weak deduction (from working late into the night to being hardworking). As an example of combining empirical generalization and abduction, suppose that someone observed smoke coming out from different parts of a building. By constructive induction one may conclude that the whole building is on fire (reasoning from different parts of the building to the whole building is empirical generalization; reasoning from smoke to fire is abduction).

It should be mentioned, that the original formulation of constructive induction (Michalski, 1983) emphasized using background knowledge for developing new concepts or attributes, beyond those supplied in the input. However, depending on the type and the way background knowledge is used, the new concepts so created can serve both for creating generalizations or explanatory hypotheses. Therefore, the idea of constructive induction can be viewed generally as an inference that includes constructive generalization, abduction or a combination of the two.

Finally, one needs to mention that induction is usually an underconstrained problem, and there may be many possible hypotheses generalizing or explaining given facts. Thus, a general formulation of induction is that, given a consequent C (an observational statement, partial knowledge, etc.) and background knowledge BK, the reasoner searches for a premise P (a hypothesis) such that P& BK strongly or weakly entails C, which we write:

$$P \;\&\; BK \mid> C \qquad (2)$$

and which satisfies a *preference criterion*. This criterion (also called "bias") specifies extra-logical conditions for selecting a hypothesis among other candidate hypotheses, i.e, the ones that satisfy eq. (2). These conditions may include, e.g., a preference for a simpler hypothesis, a more plausible one (based on the learner's BK), the least costly to use, a hypothesis that is expressed in a certain form, or a combination of such criteria.

4. A Multicriterion Classification of Learning Processes

Learning processes can be classified from many different viewpoints, for example, the type of strategy used, the research orientation, the type of knowledge representation, the application areas, etc. These classifications have been described and discussed in (Michalski, Carbonell and Mitchell, 1983 and 1986). Here, we will propose a classification, which is based on a few interrelated criteria, and which is intended to help the reader to get a general view of the whole field. As with any classification, its validity can be judged by the degree it illustrates important distinctions among various categories.

The presented categories should not be viewed as having precisely delineated sharp borderlines, but rather as labels of central tendencies that can be continuously transformed from one to another. The criteria for classification include the primary purpose of the learning method, the type of input information, the type of primary inference employed, and finally, the role of the learner's prior knowledge in the learning process. This classification is based on the paper (Michalski, 1990).

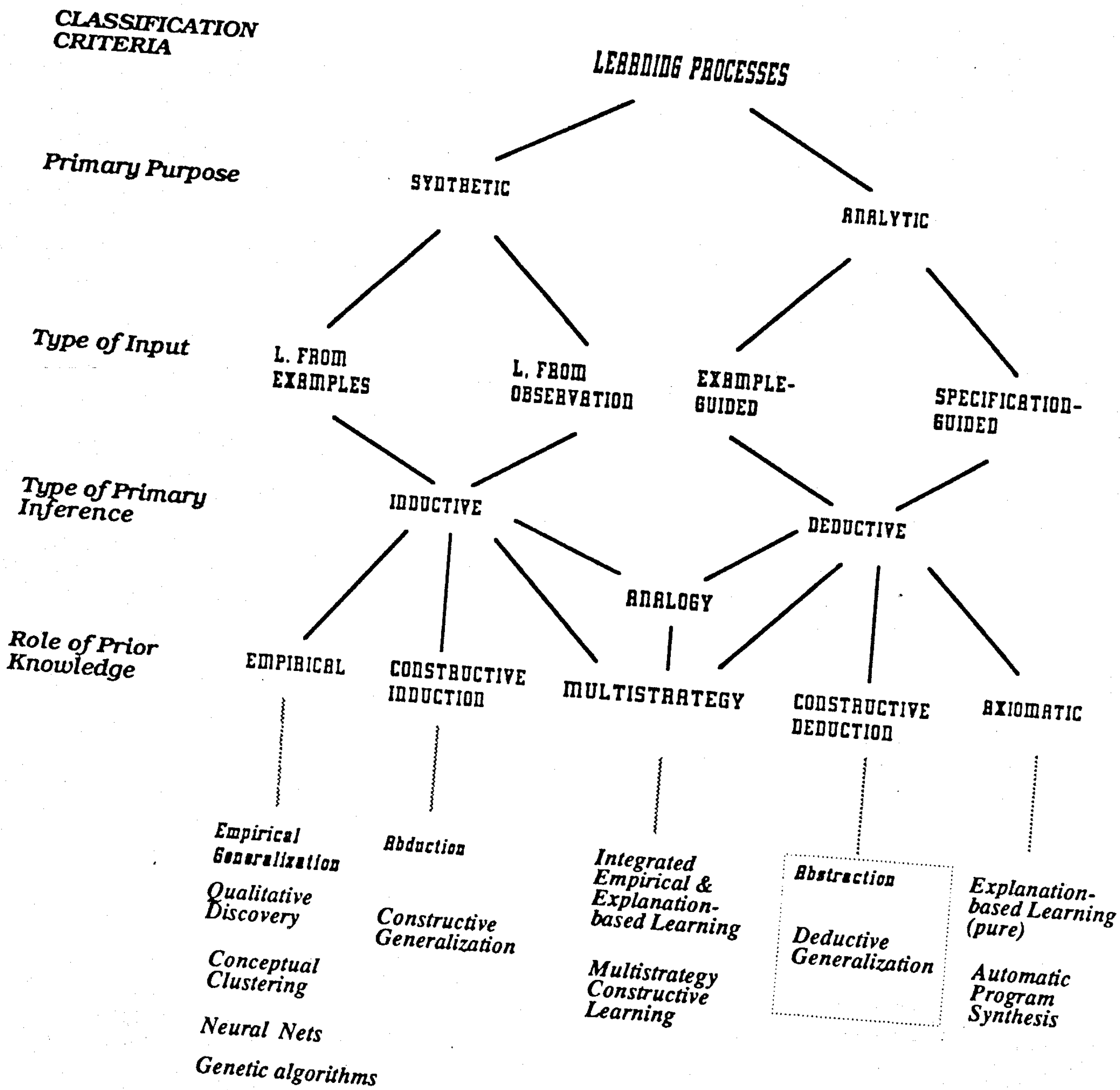

Figure 1: A multicriteria classification of machine learning methods.

As mentioned above, from the viewpoint of the primary purpose, learning methods can be classified into synthetic and analytic. The primary purpose of synthetic methods is to create new or better knowledge, while the primary purpose of the analytic methods is to transform or organize the prior knowledge into a better form according to some goal.

If the input to a synthetic learning method are examples classified by an independent source of knowledge, for example, a teacher, an expert, a simulation model, etc., then we have learning from examples (e.g., see chapter 3). When the input are facts that need to be classified or organized into a knowledge structure by the learner itself, then we have learning from observation. An example of the latter is conceptual clustering (e.g., Stepp and Michalski, 1986). Connectionist methods, genetic algorithms and qualitative discovery have been classified into empirical inductive methods. This is because they rely on relatively small amounts of BK, and their primary inference type is inductive. This inference, however, can be executed in an explicit way, as in typical symbolic methods, or in an implicit way, as in subsymbolic methods.

As described in section 2, inductive learning can be empirical (BK-poor) or constructive (BK-intensive). Empirical induction can be viewed conceptually as "tracing backward" tautological implications, e.g., domain-independent generalization rules. The most common work in this area deals with empirical generalization, which deals with creating general descriptions from specific examples using basically concepts (attributes) that are present in the descriptions of the examples. In contrast, constructive induction is a knowledge intensive induction that uses BK to create new concepts, explanations or high-level characterizations of the input information. The input information can be in the form of low level specific facts or descriptions at higher level of abstraction.

There is an additional classification within inductive methods, which is not shown in the classification, but represents some existing research. A concept description can be matched with examples in a simple and direct way, or can employ a substantial amount of background knowledge and inference. Case-based or exemplar-based methods use past facts as concept characterizations, and employ relatively sophisticated matching procedures, that allow the system to recognize new examples that do not directly match any past example (chapter 4). Such a process can be characterized as a "dynamic" induction that is performed during the matching process (or the recognition process). The method based on the two-tiered concept representation (chapter 3), uses a general concept description, and employs a matching procedure, that can potentially include any kind of inference.

As mentioned earlier, abduction can be conceptually classified as a form of induction, which "traces backward" certain domain-dependent knowledge rules. Such a characterization can be illustrated, for example, by an abductive recovery of failed proofs (Cox and Pietrzykowski, 1986; Duval and Kodratoff, 1990). A trivial way to recover from a failure would be to add the needed theorem into the knowledge base. The technique described determines the place where the proof encounters an impasse, and then proposes a minimal hypothesis necessary to achieve the missing step. The minimal hypothesis is determined by analyzing the obtained partial proof and "tracing backward" certain domain-dependent knowledge.

To reformulate its knowledge, an analytic method can be guided by an example or by a knowledge specification. The example-guided methods include explanation-based learning (DeJong, 1981; DeJong and Mooney, 1986), explanation-based-generalization (Mitchell et al., 1986), and explanation-based specialization (Minton, 1986; Minton et al., 1987). The specification-guided methods include automatic program synthesis, operationalization and advise taking (e.g., Biermann, Guiho and Kodratoff, 1984).

Explanation-based learning is classified as an example of an axiomatic method, because it is based on a pure deductive process utilizing complete and consistent BK. This knowledge plays a role analogous to axioms in formal theories. The automatic program synthesis is a process of transforming a program specification into an executable algorithm. While specification tells the system what relations should hold between the inputs and outputs, and algorithm specifies how these relations can be achieved. This process of automatic program synthesis uses truth-preserving deductive techniques to insure the logical equivalence between the program and its specification (Biermann et al., 1984).

In synthetic methods, the primary inference type is inductive, and in analytic methods, the primary inference type is deductive. Learning by analogy has been placed between inductive and deductive learning, because it can be viewed as a combination of both. Learning by analogy involves transferring certain properties of the base knowledge (a concept, a procedure, etc.) into the target knowledge. Such a transfer includes elements of both inductive and deductive learning. To recognize (or discover) an analogy, one needs to detect a similarity or common relations between the base and the target knowledge, and hypothesize that it may extend beyond the compared relations. This is primarily an inductive process. Once these common relations have been determined, then the transfer of properties from the base to the target knowledge is deductive.

For instance, consider an example of a ruthless dictator, say, Ceausescu, who governed by intimidation and deceit, and eventually was ousted in a brutal way. Suppose that there is a person, say, Patrick, a "small time" dictator, who imposes his will on others by scheming and manipulation. The first phase of the analogy is to detect a certain similarity in the attitude and general behavior between the two individuals, although the specifics are very different, and then to hypothesize that the detected similarity may extend to other correlated aspects of the comparison. This is an inductive phase of analogy. The second, deductive phase, is to develop a prediction that Patrick may also end up badly, though on a correspondingly smaller scale.

In machine learning, analogy have been usually used either to learn concepts or to learn problem solving algorithms. Illustrative examples of analogical concept learning are described in (Winston, 1982) and (Burstein, 1986). A theoretical framework and a method for analogical problem solving is presented in (Carbonell, 1986). Because analogical learning combines two fundamental inference types, it has a widespread presence and is interwoven into many learning processes.

By multistrategy learning systems are meant systems that integrate several basic strategies and/or paradigms. Many current systems are concerned primarily with integrating empirical generalization and explanation-based learning. Part four presents several examples of recent research in this area. Future multistrategy systems will undoubtedly integrate also other strategies and paradigms, including possibly also combinations of symbolic and subsymbolic learning methods (Michalski, 1990). Such multistrategy systems will be capable of adapting the learning strategy to the task at hand, and cope with learning problems involving various kinds of imperfection of BK and/or the input data, and by that will be applicable to a wide range of learning situations.

As the famous precedent of the Mendeleev classification of elements has shown, a possible effect of a taxonomy is helping to identify or introduce a new field even before its actual appearance. Such a field may be suggested by trying to balance a taxonomy. The concept of *constructive deduction* and its subdivision to abstraction and deductive generalization (the dotted rectangle) seems to play such a role, as it identifies a general research area and relates it to other areas.

By constructive deduction is meant a knowledge-based process of transforming descriptions from one representation space or language to another, which preserves information important for an assumed goal. Abstraction is classified as a constructive deduction, which transforms a description at a high level of detail to a description at a low level of detail, while preserving the truth of the relations and/or properties relevant to the assumed goal. In other words, while reducing the total information content of the original description, abstraction preserves the information important to performing an implicitly or explicity defined goal. Depending on the goal, a given description can be abstracted in many different ways. Each such process is essentially deductive, as it is not supposed to introduce or hypothesize any information that is not contained in the initial description or information source, or which cannot be deductively inferred from it using the learner's BK. The difference between constructive deduction and what we call axiomatic deduction is that the former emphasizes a change in the representation space or language, and may use a variety of knowledge transformations, rather then strictly logic-based formal methods. Although constructive deduction is essentially truth-preserving, it may include also elements of inductive inference and approximate reasoning.

Abstraction is sometimes confused with generalization. To see the difference, note that generalization transforms descriptions along the set-superset dimension and may be falsity-preserving, as in the case of inductive generalization, or truth-preserving, as in the case of deductive generalization (Michalski and Zemankowa, 1990; see also below). In contrast, abstraction transforms descriptions along the level-of-detail dimension, and is truth-preserving with regard to the characteristics of the entity(ies) important for the assumed goal. While generalization often uses the same description space (or language), abstraction often involves a change in the representation space (or language). The reason why generalization and abstraction are frequently confused may be attributed to the fact that many processes include both of them.

Deductive generalization is concerned with making generalizations that are logical deductions from the base knowledge. It differs from abstraction as it moves from considering a set to considering a superset, and typically uses the same representation formalism. For example, transforming a statement " George Mason University is in Fairfax" to "George Mason University is in Virginia" is a deductive generalization. In contrast, changing a high resolution digitized satellite image of Fairfax into a low resolution image is a simple form of abstraction. A more sophisticated abstraction would be to use the high resolution image and appropriate BK to create a map of Fairfax, which emphasizes important (according to the goal) aspects of the area. Research on problem representation, transformation of problem representation spaces, determination of a representative set of attributes, deductive transformation of a knowledge base, and related topics can be classified under the rubric of constructive deduction.

The above discussion suggests, that in parallel to multistrategy learning systems, a potentially important research area in machine learning is multirepresentation learning systems (not shown in the classification). Such systems would employ various forms of constructive deduction or induction to create and use representations at different levels of abstraction, and/or apply different description languages in the process of learning. The use of these descriptions and languages would depend on the task at hand and on the application domain. The importance of this area has been acknowledged very early by pattern recognition researchers (Bongard, 1970), as well as by AI researchers (Newell, 1969; Amarel, 1970). Nevertheless, it received relatively little attention during recent years. Among notable exceptions are (Amarel, 1986; Mozetic, 1989)

Summarizing, we have distinguished three pairs of overlapping but different reasoning and learning mechanisms. Each pair contains two opposite processes, which are concerned with different aspects of reasoning and knowledge transformation. Two of these pairs have been relatively well-explored in machine learning: deduction/induction and generalization/specialization. The third pair, which has been relatively less studied, consists of abstraction and its reverse, which we propose to call *concretion* (after Webster's dictionary, which defines it as being a process of concretizing something). These three types of mechanisms can be combined in different ways, giving some classical, well known reasoning mechanisms and some less known. The classical ones include inductive generalization and deductive specialization. Less investigated are inductive specialization, abstraction, deductive generalization, inductive concretion and other.

The above "grand" classification appears to be the first attempt to characterize and relate to each other all major methods and subareas of machine learning within one general scheme. As such, this attempt may suffer from various weaknesses and imprecision, and can be criticized on various grounds. Its primary purpose is to try to help the reader, especially a novice in this field, to view different learning mechanisms and paradigms as parts of a one general structure, rather than as a collection of unclearly related components and research efforts. By analyzing this classification, the reader may be stimulated to improve it or to develop a new, more adequate one.

5. A Brief Review of the Chapters

The book is organized into six parts. **Part One** gives an introduction and a discussion of some general issues in the field of machine learning. After the overview presented here, in **chapter 2,** Schank and Kass discuss various aspects of creativity. They view the creativity is an essential part of human (or machine) intelligence, and propose an algorithm for implementing it. The algorithm relies on *explanation patterns* (XPs), that are a form of scripts for creating explanations. In their view, the creativity lies in the use of XPs in unexpected situations, where new analogies can be drawn. This chapter is followed by Ganascia's **commentary**, which elaborates several topics and presents some critical views. This commentary, as every subsequent commentary, discusses the chapter and gives some flavor of the discussion that followed its original presentation at the International Meeting on Advances in Learning (IMAL).

Part Two presents some new developments in the area of empirical learning methods. It starts with **chapter 3,** which presents of a new approach to learning concepts that lack precise definition and are context-dependent. Such concepts are called *flexible*. The method employes the idea of two-tiered representation, in which the meaning of a concept depends on two components: an explicit part defined by the *base concept representation* (BCR), and an implicit part characterized by the *inferential concept interpretation* (ICI). The BCR (the 1st tier) expresses the general, typical and easy-to-define concept meaning, and the ICI (the 2nd tier) defines allowable modifications of the typical meaning, the matching procedures, context-dependency, and describes special cases. In matching an instance with a concept, the ICI may employ, in general, deductive, analogical or inductive inference. Early experiments with this methodology have shown that by shifting a large part of the concept meaning from the BCR to the ICI, the amount of memory needed for concept representation can be greatly reduced, without decreasing its performance accuracy on new examples. This chapter is followed by Stepp's **commentary**.

Bareiss, Porter and Wier, in **chapter 4**, define exemplars of a concept whose features are presented together with their functional explanation. Providing such explanations becomes one of the main tasks of the teacher, in contrast to approaches in which the role of the teacher is to provide examples and/or required relevant domain knowledge. The concepts are described in a form that represents a blend of an extensional definition, that defines the concept by all its instances, and an intentional definition that gives a general recognition rule for the concept. The method stores all concepts examples together with their prototypicality. This chapter is followed by Holte's **commentary**.

Next, in **chapter 5**, Quinlan describes certain improvements to the well-known ID3 learning method. One is a mechanism for exploring the probability that a case belongs to more than one class. Another involve taking into account the noise in the attribute values, and an introduction of "soft" thresholds for cases in which the attribute values are continuous. Each learning cycle consists of building a decision tree, followed by a pruning of the obtained tree. The pruned tree contains only subtrees which cannot be replaced by leaves without significantly increasing the predicted error rate of the tree. Each tree growing and pruning cycle is repeated several times on randomly chosen subsets of the training set. The most promising pruned tree is chosen as the final outcome of learning.

Next, in **chapter 6**, Falkenheiner and Michalski describe a method for integrating quantitative and qualitative discovery in the system ABACUS. This system formulates a set of equations characterizing data, and determines descriptions stating the applicability conditions for these equations. The performance of ABACUS is illustrated by applying it to some classical discovery problems, such as the discovery of the law of ideal gases. It was also applied to the analysis of chemical compounds in order to predict the distance between the atoms of bimetallic compounds, given the values of other attributes. The relations generated this way are beyond the present explanatory abilities of chemistry.

In **chapter 7**, Carbonell and Gill describe how a planning system can learn by re-planning upon its failures to achieve a goal. The main issues addressed are acquiring and refining control knowledge, augmenting an incomplete domain theory, and refining an incorrect domain theory. Three types of failures are discussed. The first failure happens when achieving a goal violates a precondition, the second occurs when an operator fails to apply, and the third occurs when a postcondition is not satisfied. Depending on the kind of failure, a different recovery strategy is applied.

In **chapter 8**, Pazzani describes a failure-driven learning system that integrates explanation-based learning with empirical learning. The goal of the system is to create heuristics for fault diagnosis which capture the information implicit in the device models. The method explains why a heuristic does not apply in a certain case, and corrects the heuristic so to avoid proposing an erroneous fault hypothesis. This work is applied to a spacecraft failure correction.

This part ends with **chapter 9**, in which Hanson presents a system called WITT, which implements a form of conceptual clustering. The system has three components. The first selects the seeds, around which initial clusters are created. The second component applies three operators: object selection, new seed selection, and seed merging. The third component is an information metric by which the system evaluates the cohesion and distinctiveness of a cluster. WITT uses the most representative instance of a cluster as a prototype, and allows clusters to share some of their instances (polymorphy).

Part Three gives an account of various analytical learning methods. In **chapter 10**, Mitchell, Mahadevan and Steinberg describe a learning apprentice system, called LEAP. The authors define a learning apprentice as an interactive knowledge-based consultant, which improves its knowledge by observing and analyzing the activities of the users during their normal interaction with the system. When a user provides an unexpected (to the system) solution, at first, the system tries to prove or disprove the validity of this solution. When the user's solution is proven to be valid, then the system generalizes the trace of its proof into an explanation of this validity. This explanation then becomes a new rule, which is introduced into the system for future use. This chapter is followed by Brazdil's **commentary**.

Next, in **chapter 11**, Shavlik and DeJong (Gerald) study the problem of detecting recursion in explanations. This requires an analysis of the explanation structures and the detection of repeated, inter-dependent applications of the rules. The detection of such recursive structural patterns is implemented in their system BAGGER. The system is illustrated by applying it to a block-world problem.

The remaining three chapters of this part show how analytic learning can be applied to various specific problems. **Chapter 12**, by Prieditis, describes an application of explanation-based methods to program generation, generally referred to as automatic programming. In this case, the training example is a "weak" algorithm that is used to solve a problem. The method first generates a trace of its solution. Then it generalizes the trace by including iterative macro-operators in the places where the trace repeats several times the same kind of actions. The novelty of this approach lies in generalizing trace structures, rather than generalizing instantiated operators.

In **chapter 13**, Vrain describes how a domain theory, expressed in the form of first order logic theorems, can be used to improve the reliability of generated generalizations. It amounts to the addition of a theorem prover to an induction system. The paper concentrates on the way the theorem prover is used as a feature finder during the detection of the similarities among examples.

The author also discusses a method for reducing the number of possible generalizations by using negative examples, and a method for incremental learning.

In **chapter 14**, the last chapter of this part, Hirsh describes a method, called ROE (reasoning about operationality in EBL), which is concerned with operationality of descriptions generated by explanation-based generalization. The method, in addition to generalizing proofs that some property holds for a given training instance, determines the general conditions of operationality for the results. The user is required to provide general knowledge about operationality in the domain of interest, and, from some examples of operationality criteria, the system will find operational rules of operationality.

Part Four describes several efforts on integrating different learning strategies. In **chapter 15**, Lebowitz describes the system UNIMEM, which empirically discovers similarities among examples. These empirical generalizations may represent purely coincidental effects. The author proposes to use background knowledge to explain the obtained generalizations, and choose on that basis the most appropriate one. This chapter is followed by Rendell's **commentary**.

Kodratoff in **chapter 16** describes how to construct explanations from a trace of the proof that a training example is an instance of some concept. The notion of "explanation" is generalized so that it includes also processes of detecting similarities among concept examples, i.e., empirical concept learning. In the latter case, a generalization is viewed as a form of an "explanation" why an example is a concept instance. It is then shown how to learn from past explanations to explain new instances, and how to learn from failures to explain. These processes include recovering from error in proof, learning from unrecognized positive examples and from incorrectly recognized negative examples. An account of how to use human experts' explanations is also given. This chapter is followed by Stepp's **commentary**.

Next, in **chapter 17**, Bergadano and Giordana propose a methodology for integrating empirical and analytical methods for learning concept descriptions. The method involves a search for concept descriptions in a tree of logic descriptions, ranking from the most general to the most specific. The tree is tested on a set of positive and negative examples. This search is guided by several domain independent criteria, such as consistency and completeness, limiting the size of the obtained expressions, and an understandability criterion. This chapter also suggests methods to use background knowledge as a major domain dependent search heuristic in order to expand the specialization tree toward a promising description or, conversely, to prune the branches that are not in agreement with domain theory.

In **chapter 18**, Wilkins presents an apprentice system, ODYSSEUS, which, in contrast with LEAP (chapter 11), can handle situations in which the domain theory is incapable of producing an explanation of a given training instance. It is assumed that the domain knowledge is incomplete or erroneous, and the learning process involves making improvements to this knowledge. The improvements are made by completing failed explanations using general-purpose meta-rules.

In **chapter 19**, Tecuci and Kodratoff describe DISCIPLE, a multistrategy system capable of learning with imperfect background knowledge. If the BK is complete and consistent, the system executes standard explanation-based learing to improve its performance. Otherwise, the system compensates for its imperfect knowledge by interacting with the user, and employing a combination of explanation-based learning, analogical learning and empirical learning. Depending on the degree of BK imperfection, such a process may lead to the improvement of the system's competence, or to the improvement of both the competence and performance.

Part Five gives an overview of research in the area of subsymbolic learning methods. Hinton, in **chapter 20**, provides a comprehensive characterization of the work on connectionist approaches, and explains how to construct internal representations for difficult learning problems. The chapter describes the advantages and disadvantages of various connectionist methods, such as the least mean square procedure, the back-propagation method, Boltzman machines, competitive learning and others. The description of what is a generalization in a neural network is also presented.

In **chapter 21**, De Jong (Kenneth) describes basic aspects of genetic algorithms, and presents them as a means for achieving robust learning systems. In a learning mode, a genetic algorithm searches the space of legal structural changes to determine situations which achieve a desired behavioral change. The search is done using certain operators, such as crossover, mutation and inversion. The chapter discusses the relevance of various features of adaptive systems to machine learning.

Part Six presents two types of formal approaches to machine learning. In **chapter 22**, Haussler presents an improvement over the Mitchell's version space method in the spirit of the Valiant's approach. Instead of being concerned only with hypotheses that are consistent with examples, the method also looks for hypotheses that are sufficiently "close" to the target concept. Such an approach allows the author to define the notion of convergence of the learning method. The compexity of the method is analyzed.

Finally, in **chapter 23**, Rivest and Schapire explore an approach to learning by a deterministic finite-state automaton that operates in an abstract environment. They assume that the automaton has little built-in knowledge of its environment. To reduce the complexity of learning, they limit the set of possible states of the automaton to a carefully selected subset. The admissible states contain a sequence of "actions," followed by a "sensation." The automaton predicts the sensation that may be associated with a given sequence of actions, and this allows it to define and select most informative tests. The system is illustrated by applying it to several simple made-up problems.

5. Frontier problems

In the conclusion, we would like to discuss several areas of machine learning that seem to be of particular challenge and importance in future research. One such area is concerned with the development of methods that can utilize different learning strategies depending on the task at hand, i.e., building multistrategy learning systems. The importance of this area is that real-world situations rarely present problems that can be handled by a neat, single-strategy approach. There have been several learning systems developed that combine two or more strategies. Typically they combine simple empirical learning and explanation-based learning. Most of these systems integrate different strategies in a rather inflexible and an a priori defined way. A natural development of this research could be in the direction of a goal-based integration of a spectrum of machine learning strategies. An attempt in this direction is presented in (Michalski, 1990).

With the growing sophistication of learning programs, there is an interest in increasing the transfer of machine learning programs from university laboratories to the real world, where they can be applied to problems of practical significance. This trend is of great consequence, as it challenges researchers to test their research in the context of real-life problems, and may lead to economically important applications. Such applications may, in turn, generate new interesting research problems, not to mention the needed support for further research.

An exposure of a learning system to real-world problems will require further advancement of learning systems in many areas, such as handling erroneous or missing data, learning imprecise and context-dependent concepts, employing sophisticated methods for plausible inference, and causal reasoning, drawing complex analogies, and using quantitative and/or qualitative models in a learning process. Some applications will require systems that are capable of analyzing and understanding past behavior of a complex system, and then use this understanding to predict future behavior. Thus, there is a need for developing advanced prediction systems.

In view of the above applications, there will also likely be a need for developing multirepresentational learning systems. Such systems employ knowledge representations in different goal-oriented forms, expressed in different structures, and at different levels of abstraction. At present, most existing empirical learning methods, both symbolic and subsymbolic, tend to be oriented toward only single attribute-based representations. Concept instances are described in terms of attribute-value pairs, and concept descriptions as simple structures (rules, trees, etc.) which utilize these attributes. The strong interest in such systems has been probably due to the fact that such representations are sufficient in many simple practical applications, and because many methods do not use much background knowledge. As application problems become more complex, there will be an increased need to use much richer, structural and multiform knowledge representations. In fact, if a learning program needs to use an extensive amount of BK, it will undoubtly need to employ a structural rather than attributional representation.

To exhibit advanced learning capabilities, a learning program - synthetic or analytic - needs to draw upon a substantial amount of both factual and general knowledge. Future learning systems will then require a connection to large databases and knowledge bases. The CYC project (Lenat, 1989) is an example of a long-term effort to develop such a computer-manipulable large scale knowledge base. In this context, an important research area is an application of machine learning to the analysis and extraction of knowledge from large databases. This application will involve a construction of "intelligent" interfaces between existing large databases and the machine learning systems.

In view of a large proliferation of machine learning projects, there is also an strong interest in experimental and analytical comparative studies of different methods and paradigms. At present, basic paradigms and areas of research concentration include: symbolic inductive learning, analytic methods, integrated systems, neural nets and genetic algorithms. Each of these paradigms can be subdivided to various subareas. For example, symbolic inductive learning includes empirical concept learning from examples, discovery systems, examplar-based learning, conceptual clustering and constructive induction. Different paradigms and methods seem to be primarily oriented toward a different area of applicability, and their relative merits and limitations are not clearly understood.

In the context of building large-scale learning systems that utilize significant background knowledge, it is important to assure high instructability of the learning systems. This means that they should be combined with advanced knowledge acquisition systems. This way, the knowledge can be economically introduced to a learning system without always involving a learning process. This direction of research calls for an increased collaboration between machine learning and knowledge acquisition communities.

Finally, future machine learning systems should exhibit high level capabilities for interacting with humans and employing for that purpose multimedia representations. They should be capable of explaining what they have learned, and what they still need to learn. This implies that learning systems should have the ability to perform metaknowledge reasoning, generate learning tasks by themselves, to ask questions and know how to utilize different answers, exhibit advanced explanatory facilities, and perform causal and model-based reasoning. In perfoming such processes, the systems might employ visual or auditory representations.

Acknowledgments

The authors thank Kejitan Dontas, David Littman, Eugene Norris, Gheorghe Tecuci, Gail Thornburg and Jianping Zhang for useful comments and suggestions. This research was supported in part by the Office of Naval Research under grants No. N00014-88-K-0397 and No. N00014-88-K-0226, and in part by the Defense Advanced Research Projects Agency under the grant administered by the Office of Naval Research No. N00014-K-85-0878.

References

Amarel S., "On the Representation of Problems and Goal-Directed Procedures for Computers,", in *Theoretical Approaches to Non-Numerical Problem Solving*, Banerji R.B. and Mesarovic M. (Eds.), Springer Verlag, Heidelberg, 1970.

Amarel S., "Program Synthesis as a Theory Formation Task: Problem Representations and Solution Methods," in *Machine Learning: An Artificial Intelligence Approach, Volume II*, R.S. Michalski, J.G. Carbonell, T.M. Mitchell (Eds.), pp. 499-569, Morgan Kaufmann 1986.

Aristotle, Posterior Analytics, in *Great Books of the Western World*, vol.8, R.M. Hutchins (Ed.), Encyclopedia Britannica, Inc. 1987.

Bergadano, F., Saitta, L., "On the Error Probability of Boolean Concepts Descriptions," *Proceedings of the fourth European Working Session on Learning*, Morik K. (Ed.), pp 201-211, Pitman, 1989.

Biermann A. W., Guiho G., Kodratoff Y., *Automatic Program Construction Techniques*, MacMillan Publishing Company, New York 1984.

Bongard M., *Pattern Recognition*, Spartan Books, Washington D.C., 1970.

Buchanan, B.G., Feigenbaum, E. A., and Sridharan, N. S., Heuristic Theory Formation: Data Interpretation and Rule Formation, *Machine Intelligence*, Vol. 7, pp. 267-290, Halsted Press, Wiley, 1972.

Buntine, W., "A Critique of the Valiant Model," *Proceedings of the Eleventh International Conference on Artificial Intelligence*, pp. 837-842, Detroit , August 1989.

Burstein, M. H., "Concept Formation by Incremental Analogical Reasoning and Debugging," in *Machine Learning: An Artificial Intelligence Approach, Volume II*, R.S. Michalski, J.G. Carbonell, T.M. Mitchell (Eds.), pp. 351-369, Morgan Kaufmann 1986.

Carbonell, J. G., "Derivational Analogy: A Theory of Reconstructive Problem Solving and Expertise Acquisition," in *Machine Learning: An Artificial Intelligence Approach, Volume II*, R.S. Michalski, J.G. Carbonell, T.M. Mitchell (Eds.), pp. 371-392, Morgan Kaufmann 1986.

Collins, A. and Michalski, R.S., "The Logic of Plausible Reasoning: A Core Theory," *Cognitive Science*, Vol. 13, pp. 1-49, 1989.

Cox, P.T., Pietrzykowski, T., "Causes for Events: Their Computation and Applications" *Proceedings of the Eighth International Conference on Automated Deduction*, Oxford, 1986.

DeJong G., "Generalizations Based on Explanations," *Proceedings of the Seventh International Conference on Artificial Intelligence*, pp. 67-69, Vancouver, August 1981.

DeJong G., Mooney R. "Explanation-Based Learning: An Alternative View," *Machine Learning Journal*, Vol. 1, No. 2, pp. 145-175, 1986.

Duval B., Kodratoff Y. "A Tool for the Management of Incomplete Theories: Reasoning about Explanations" in *Machine Learning, Meta-Reasoning and Logics*, P. Brazdil and K. Konolige (Eds.), Kluwer Academic Press, 1990, pp. 135-158.

Holland J. H., Holyoak K. J., Nisbett R. E., Thagard P. R. *Induction: Processes of Inference, Learning and Discovery,* The MIT Press, Cambridge MA, 1986.

Lenat, D.B., "Automated Theory Formation in Mathematics," *Proceedings of the Fifth International Joint Conference on Artificial Intelligence,* pp. 833-842, 1977.

Lenat, D.B., "The Role of Heuristics in Learning by Discovery: Three Case Studies," in *Machine Learning: An Artificial Intelligence Approach,* Michalski R.S., Carbonell J.G., and Mitchell T.M. (Eds.), pp. 243-306, Morgan Kaufmann 1983.

Lenat, D.B. "When will Machines Learn?," *Machine Learning Journal,* Vol. 4, No 3/4, 1989.

Michalski, R. S., "Discovering Classification Rules Using Variable-valued Logic System VL1," *Proceedings of the Third International Conference on Artificial Intelligence,* pp. 162-172, Stanford University, 1973.

Michalski, R.S., "A Theory and Methodology of Inductive Learning," in *Machine Learning: An Artificial Intelligence Approach,* Michalski, R.S., Carbonell, J.G., Mitchell, T.M. (Eds.), pp. 83-134, Morgan Kaufmann 1983.

Michalski R.S., "Understanding the Nature of Learning: Issues and Research Directions," in *Machine Learning: An Artificial Intelligence Approach, Volume II,* R.S. Michalski, J.G. Carbonell, T.M. Mitchell (Eds.), pp. 3-25, Morgan Kaufmann 1986.

Michalski, R.S., "Multistrategy Constructive Learning: Toward a Unified Theory of Learning," *Reports of Machine Learning and Inference Laboratory,* MLI90-1, Center for Artificial Intelligence, George Mason University, 1990 (an earlier version of this paper appeared in the Proceedings of the ONR Workshop on Knowledge Acquisition, Arlington, VA, November 6-7, 1989).

Michalski, R. S. and Chilausky, R., "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," *Policy Analysis and Information Systems,* vol. 4, No. 2, (Special Issue on Knowledge Acquisition and Induction), June 1980.

Michalski, R.S., Stepp R.E., "Learning from Observation: Conceptual Clustering," in *Machine Learning: An Artificial Intelligence Approach,* Michalski, R.S., Carbonell, J.G., Mitchell, T.M. (Eds.), pp. 331-363, Morgan Kaufmann 1983.

Michalski, R.S., Carbonell, J.G., Mitchell, T.M. (Eds.), *Machine Learning: An Artificial Intelligence Approach,* Morgan Kaufmann 1983.

Michalski, R.S., Carbonell, J.G., Mitchell, T.M. (Eds.), *Machine Learning: An Artificial Intelligence Approach, Volume II,* Morgan Kaufmann 1986.

Michalski, R.S., Littman, D. C., "Future Directions of AI in a Resource-Limited Environment", *Proceedings of IJCAI-89 Workshop on Knowledge Discovery in Databases,* Detroit, August 1989.

Michalski, R. S. and Zemankova, M., "What is Generalization? An Inquiry into the Concept of Generalization and its Types," *Reports of Machine Learning and Inference Laboratory,* George Mason University, MLI90-7, 1990.

Minton, S. "Improving the Effectiveness of Explanation-Based Learning," *Proceedings of the Workshop on Knowledge Compilation*, Oregon State University, Corvallis, OR, 1986.

Minton, S. Carbonell, J.G., Etzioni, O., Knoblock, C.A., Kuokka, D.R., "Acquiring Effective Search Control Rules: Explanation-based Learning in the PRODIGY System," *Proc. 4th International Machine Learning Workshop*, pp. 122-133, Univ. California, Irvine, 1987.

Mitchell T., Utgoff P., Banerji R., "Learning by Experimentation, Acquiring and Refining Problem-solving Heuristics," in *Machine Learning: An Artificial Intelligence Approach,* R.S. Michalski, J.G. Carbonell, T.M. Mitchell (Eds.), pp. 163-190, Morgan Kaufmann 1983.

Mitchell T. M., Keller R. M., Kedar-Cabelli S. T. "Explanation-Based Generalization: A Unifying View," *Machine Learning Journal*, Vol. 1, No. 1, pp. 47-80, 1986.

Mozetic I., "Hierarchichal Model-Based Diagnosis," *Reports of Machine Learning and Inference Laboratory*, MLI 90-2, George Mason University, 1989.

Muggleton S., Buntine R., "Machine Invention of First Order Predicates by Inverting Resolution," *Proceedings of 5th International Machine Learning Workshop*, pp 339-352, Morgan Kaufmann, 1988.

Newell A., "Heuristic Programming III: Structured Problems," in *Progress in Operation Research,* Vol. 3, Aronofsky J. S. (Ed.), John Wiley, New York, 1969.

Peirce C. S., Elements of Logic, in *Collected papers of Charles Sanders Peirce* (1839-1914), CH. Hartshone and P. Weiss (Eds), The Belknap Press Harvard University Press, Cambridge, MA, 1965.

Quinlan, J.R., "Discovering Rules from Large Collections of Examples: A Case Study," *Expert Systems in Microelectronic Age*, Michie, D. (Ed.), Edinburgh University Press, Edinburgh, 1979.

Rouveirol C., Puget J.F., "A Simple Solution for Inverting Resolution", *Proceedings of the fourth European Working Session on Learning,* pp 201-211, Pitman, 1989.

Samuel, A. L., Some Studies in Machine Learning Using the Game of Checkers, *IBM Journal of Research and Development*, No. 3, pp.211-229, 1959.

Winston, P.H., "Learning Structural Descriptions from Examples," Ph.D. thesis, MAC TR-76, Massachusetts Institute of Technology, September 1970.

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

Winston, P. H. "Learning New Principles from Precedents and Exercises," *Artificial Intelligence 19*, pp. 321-350, 1982.

Wnek, J., Sarma, J., Wahab, A.A. and Michalski, R.S., "An Experimental Case Study in Concept Learning Comparing Symbolic, Neural Net and Genetic Algorithm Methods," *Reports of Machine Learning and Inference Laboratory*, MLI 89-1, George Mason University, 1990.

MACHINE LEARNING: AN ARTIFICIAL INTELLIGENCE APPROACH

Eds. by

Y. Kodratoff
R. S. Michalski

Vol. III, San Mateo, CA. Morgan Kauffman Publ., June 1990.

11/25/89

PREFACE

As the field of machine learning (ML) is enjoying an unprecedented growth and attracts many new researchers, there is a need for regular summaries and comprehensive reviews of its progress. This volume is a sequel to volumes I (1983) and II (1986), and presents a sample of ML research representative of the period between 1986 and 1989.

One noteworthy characteristic of that period is that a much larger than before portion of research has been done outside of U.S.A., particularily in Europe. In reflection of this fact, the book contains a significant number of non-USA contributions. Another novelty is that this volume covers topics not covered at all or covered only sparcely by previous volumes, such as the connectionist learning methods, genetic algorithms, and computational learning theory.

To provide a comprehensive representation of research, this volume has drawn upon several sources. Most of the chapters are directly invited contributions by leading researchers in the field. Several chapters are updated and extended versions of invited presentations at the International Meeting on Advances in Learning (IMAL) held in Les Arcs, France in July 1986. These chapters are accompanied by commentaries prepared by their discussants at the meeting. Finally, few chapters are based on papers selected from among those presented at the 4th and 5th International Machine Learning conferences, held at the University of California at Irvine in June 1987, and the University of Michigan at Ann Arbor in June 1988, respectively.

For a more complete coverage of the progress of the field, the reader is referred to relevant journals, in particular, Machine Learning Journal, Artificial Intelligence Journal and Artificial Intelligence Magazine, and to the proceedings of various conferences. Among the most relevant such conferences are international machine learning conferences ('87, '88, '89), the meetings of the American Association for Artificial Intelligence (AAAI '86, '87 and '88), workshops on Computational Learning Theory (COLT '88 and '89), the workshop on Explanation-based Learning ('88), International Conferences on Genetic Algorithms ('87 and '89), conferences on Neural Nets, the European Working Sessions on Learning (EWSL '87, '88 and '89), the European Congresses on Artificial Intelligence (ECAI '86 and '88), the International Joint Conferences on Artificial Intelligence (IJCAI '87 and '89) and International Workshop on Tools for Artificial Intelligence (1989).

The bibliography at the end of the book provides a comprehensive guide to these and related publications. It contains over 1000 entries and refers to publications in all major ML subareas for the period 1985-1989. All the entries are indexed, using a classification of ML publications into 17 categories .

It is the editors' pleasant duty to thank all those who helped in the preparation of this book. Our deep gratitude goes to all the contributors for their efforts to write the chapters in a highly comprehensive and easy-to-read manner. We are very grateful to the reviewers, whose help was indispensible in securing the high quality of the volume.

Special thanks go to DIGITAL-EUROPE and the London office of the U.S. Army. These organizations sponsored IMAL, which gave the first impetus for the idea of this volume. The editors also acknowledge the help and technical support extended to them by the faculty, staff and research assistanst of the Center for Artificial Intelligence at George Mason University and by the French National Research Center (CNRS).

RSM & YK

11/25/1989

MACHINE LEARNING:
An Artificial Intelligence Approach Volume III

CONTENTS