Artificial Intelligence: Connectionist and Symbolic Approaches

Ron Sun

CECS, University of Missouri-Columbia, Columbia, MO 65211, USA

rsun@cecs.missouri.edu

573-884-7662

November 7, 2000

Abstract

In this article, the two competing paradigms of artificial intelligence, connectionist and symbolic approaches, are described. It is pointed out that no single existing paradigm can fully address all the major AI problems. Each paradigm has its strengths and weaknesses. This situation points to a need to integrate these two existing paradigms.

Perhaps the most significant feature of current artificial intelligence research is the co-existence of a number of vastly different and often seriously conflicting paradigms, competing for the attention of the research community (as well as research funding). In this article, two competing paradigms of artificial intelligence, the connectionist and the symbolic approach, will be described. Brief analysis and criticism of each paradigm will be provided, and possible integration of the two will also be discussed as a result of the analysis on their respective shortcomings.

1 The Two Paradigms

The two main competing paradigms in artificial intelligence can be summarized as follows:

- The traditional symbolic paradigm (Newell and Simon, 1976). The field of AI, since its inception, has been conceived mainly as the development of models using symbol manipulation. The computation in such models is based on explicit representations that contain symbols organized in some specific ways and aggregate information is explicitly represented with aggregate structures that are constructed from constituent symbols and syntactic combinations of these symbols.

- The more recently established connectionist paradigm (McClelland and Rumelhart, 1986; Smolensky, 1988). The emergence of the connectionist paradigm largely resulted from various dissatisfactions with symbol manipulation models, especially in their inability to handle flexible and robust processing in an efficient manner. The connectionist paradigm aims at massively parallel models that consist of a large number of simple and uniform processing elements interconnected with extensive links, that is, artificial neural networks and their various generalizations. [1] In many connectionist models, representations are distributed throughout a large number of processing elements (in correspondence with the structure of such models). Sometimes the constituent symbolic structures of aggregate information are embedded in a network and difficult to identify. Due to their massively parallel nature, such models are good at flexible and robust processing, and show promise at dealing with some tasks that have been difficult for the symbolic paradigm.

It should be emphasized that up until now, no single existing paradigm can fully address all the major AI problems. Each paradigm has its strengths and weaknesses, and excels at certain tasks while falls short at some others. This situation, in a way, indicates the need to integrate these existing paradigms somehow.

[1] The development of such models was inspired in part by biological neural networks, and thus they are also referred to as artificial neural networks or simply neural networks.

2 Symbolic AI

The *physical symbol system hypothesis* introduced by Newell and Simon (1976) clearly articulated the tenets of symbolic AI. They defined a *physical symbol system* as follows:

> A physical symbol system consists of a set of entities, called symbols, which are physical patterns that can occur as components of another type of entity called an expression (symbol structure). Thus a symbol structure is composed of a number of instances (or tokens) of symbols related in some physical way (such as one token being next to another).

They further claimed that symbols can designate arbitrarily: "a symbol may be used to designate any expression whatsoever"; "it is not prescribed a priori what expressions it can designate." "There exist processes for creating any expression and for modifying any expression in arbitrary ways". Based on that, they concluded: "A physical symbol system has the necessary and sufficient means for general intelligent action", which is the (famed) *physical symbol system hypothesis.*

The physical symbol system hypothesis has spawned (and was used to justify) enormous research effort in traditional AI (and in Cognitive Science). This approach (classical symbolism) typically uses discrete symbols as primitives, and performs symbol manipulation in a sequential and deliberative manner.

2.1 Representation and Search

Two fundamental ideas that originated in the earliest days of AI are search and representation, which have played central roles in symbolic AI.

Let us discuss the idea of search space first. In any problem (to be tackled by AI), there is supposed to be a space of *states* each of which describes a step in problem solving (inference). *Operators* can be applied to reach a new state from a current state. Search techniques adopted in the early days of AI include depth-first search and breadth-first search. In depth-first search, from the current state, the system examines one alternative (a "path") at a time by applying one of the operators to the current state, which leads to a new state. Then from the new state, the same process is repeated, until it reaches a goal state or hits a dead-end where there is no operator that can be applied. In the latter case, the system backs up to a previous state and tries a different alternative. In breadth-first search, however, the system examines all alternatives ("paths") at once by applying each of all the applicable operators. Such exhaustive search techniques are inefficient. To speed up search, many heuristic search algorithms have been proposed. The idea of search space has been applied in all the areas of AI, including in problem solving, natural language processing, robotics and vision, knowledge representation/reasoning, and machine learning (see Russell and Norvig, 1995 for more details regarding search).

Another important idea is representation. It embodies the belief that knowledge should be expressed in an internal form that facilitates its use, corresponding to the requirement of the task to be handled and mirroring the external world (in some way). A variety of symbolic representational forms have been developed over the years in AI, and most of them are used in conjunction with search algorithms for inference. One of the earliest representational forms involves rule-based reasoning, in which discrete rules are used to direct search (inference). Rules are composed of both conditions, which specify the applicability of a rule, and conclusions, which specify actions or outcomes. Rules are modular: ideally each rule can be added to or deleted from a system, without affecting the other parts of the system (modularity may, however, inadvertently hamper computational flexibility and dynamic interaction).

A popular form of rule-based reasoning is the production system, which evolved from some psychological theories that emerged in the 1960's and 1970's. A production system consists of (1) a production rule base (for storing all the rules) and (2) a working memory (for storing initial, intermediate, and final results) and (3) a control structure for coordinating the running of the system. The inference process in a production system can be either forward chaining or backward chaining.

Formal logics constitute an alternative approach in rule-based reasoning, as advocated chiefly by John McCarthy (McCarthy, 1968). They are relatively simple, formally defined languages capable of expressing rules in a rigorous way. Logic inference is performed in formally defined ways that guarantee the completeness and soundness of the conclusions, which can be carried out using a variety of algorithms, beyond simple forward and backward chaining. Formal logics (and most production systems) are restrictive: one needs to have all conditions precisely specified in order to perform one step of inference. Thus they are unable to deal with partial, incomplete or approximate information. There is also no intrinsic way for handling reasoning involving similarity-based processes (Sun, 1994).

Another type of representation aims to capture the aggregate *structures* of knowledge (instead of dispersing such structures as in production systems). Knowledge can be organized in structured chunks, each of which is centered around a particular entity, and each can contain, or be contained in, other chunks. Each chunk contains all the pieces of information regarding a certain entity as well as their interrelations. For instance, a *frame* (as proposed by Marvin Minsky) represents a concept in terms of its various attributes (slots), each of which has a name (label) and a value. By organizing knowledge in frames, all the relevant pieces of information can be accessed in a structured way. A *semantic network* (as proposed by Ross Quillian) consists of a set of nodes, each of which represents a particular (primitive) concept, and labeled links, each of which represents a particular relation between nodes. Semantic networks also allow efficient and effective access of knowledge, albeit in a different way, by following links that go from one concept to all the others that have a particular relation to the original one, through, e.g., "spreading activation". *Scripts* (proposed by Roger Schank and associates) are used for representing prototypical event sequences in stereotypical situations, for example, eating in a restaurant. In such a situation, there is an almost fixed sequence. Scripts help with the efficient recognition and handling of these sequences. The shortcomings of the above types of representations include: (1) structures often need to be determined a priori and hand coded, (2) they are usually fixed and cannot be changed dynamically, and (3) they can become too costly and unwieldy to capture the full extent of complex real-world situations.

Some more recent developments in the symbolic paradigm aim to remedy some of the problems with traditional symbolic representation, especially those of logic-based approaches, as discussed earlier. These extensions include Default Logic and Circumscription, among many others. For example, *Default Logic* is proposed to model default reasoning: it deals with beliefs based on incomplete information, which might be modified or rejected later on based on subsequent observations. These logics have shortcomings, including the lack of capabilities for dealing with (1) approximate information, (2) inconsistency, and most of all, (3) reasoning as a complex, interacting process (Sun, 1994). Thus they are somewhat deficient from the standpoint of capturing human reasoning. For an overview of these models, see Davis (1990).

In a totally different vein, Zadeh proposed *fuzzy logic* (see Zadeh, 1988), primarily to capture vagueness or approximate information in linguistic expressions, such as "tall" or "warm", which have no clear-cut boundaries. The basic idea is as follows: for each concept, a set of objects satisfying that concept to a certain degree form a subset, namely a fuzzy subset. This fuzzy (sub)set contains as its elements pairs consisting of an object and its grade of membership, which represents the degree to which it satisfies the concept associated with the set. A fuzzy logic can therefore be constructed with which one can reason about the fuzzy truth values of concepts.

The *probabilistic approach* (especially the Bayesian approach) treats beliefs as probabilistic events and utilizes probabilistic laws for belief combinations (Pearl 1988). Note, however, that human reasoning may not always conform to the assumptions and the laws of probability theory, probably in part because of the complexity of the formal models. In addition, it is not always possible to obtain adequate probability measures in practice.

2.2 Symbolic Learning

Learning is a major issue in AI, in Cognitive Science, and in other related areas. While very sophisticated representations have been developed in symbolic AI, learning is, comparatively speaking, difficult for symbolic AI. This is in part due to the fact that, from its inception, symbolic AI has been centered on representation, not learning.

Nevertheless, there have been some developments of symbolic learning, especially since the late 1980's (see Shavlik and Dietterich, 1990). The majority of work in symbolic machine learning focuses on "batch learning". Typically, the learner is given all the exemplars/instances, positive and/or negative, before learning starts. Most of these algorithms handle the learning of concepts with simple rules or decision trees (Quinlan, 1986). They typically involve (1) a division of all the instances/exemplars into mutually exclusive or overlapping classes (i.e., clustering), and/or (2) the induction of a set of classification rules describing the makeup of a concept (Michalski, 1983; Quinlan, 1986).

More recently, some symbolic learning algorithms have been extended to deal with noisy or inconsistent data and to deal with incremental learning. In addition, inductive logic programming tries to induce more powerful first-order rules from data. Recently, there have also been some reinforcement learning algorithms that handle dynamic sequences, which necessarily involve temporal credit assignment (i.e., attributing properly a success/failure to some preceding steps). (See Shavlik and Dietterich, 1990.)

2.3 Criticisms

Although the symbolic paradigm dominated AI and Cognitive Science for a long while, it has been steadily receiving criticisms from various sources (for example, from Rodney Brooks, John Searle and Hubert Dreyfus; see Dreyfus and Dreyfus, 1987). The critics focused largely on the disembodied abstractness of this approach, especially in relation to interacting with the world (Dreyfus and Dreyfus, 1987). Some, such as John Searle, emphasized the importance of biological substrates of intelligence. Some other more specific criticisms have been identified earlier.

3 Connectionist AI

In the 1980's, the publication of the PDP book (McClelland and Rumelhart, 1986) started the so-called "connectionist revolution" in AI and cognitive science. The basic idea of using a large network of extremely simple units for tackling complex computation seemed completely antithetical to the tenets of symbolic AI and has met both enthusiastic support (from those disenchanted by traditional symbolic

AI) and acrimonious attacks (from those who firmly believed in the symbolic AI agenda). Even today, we can still feel, to some extent, the divide between connectionist AI and symbolic AI, although hybrids of the two paradigms and other alternatives have flourished. However, much of the controversy was the result of misunderstanding, overstatement, and terminological differences.

Connectionist models are believed to be a step in the direction toward capturing the intrinsic properties of the biological substrate of intelligence, in that they have been inspired by biological neural networks and seem to be closer in form to biological processes. They are capable of dealing with incomplete, approximate, and inconsistent information as well as generalization.

3.1 Connectionist Learning

Connectionist models excel at learning: Unlike the formulation of symbolic AI which focused on representation, the very foundation of connectionist models has always been learning. Learning in connectionist models generally involve the tuning of weights or other parameters in a large network of units, so that complex computations can be accomplished through activation propagation through these weights (although there have been other types of learning algorithms, such as constructive learning and weightless learning). The tuning is usually based on gradient descent or its approximations. The best known of such learning algorithms is the backpropagation algorithm (McClelland and Rumelhart, 1986).

In terms of task types tackled, connectionist learning algorithms have been devised for (1) supervised learning, similar in scope to aforementioned symbolic learning algorithms for classification rules but resulting in a trained network instead of a set of classification rules; (2) unsupervised learning, similar in scope to symbolic clustering algorithms, but without the use of explicit rules; (3) reinforcement learning, either implementing symbolic methods or adopting uniquely connectionist ones.

Connectionist learning has been applied to learning some limited forms of symbolic knowledge. For example, Pollack (1990) used the standard backpropagation algorithm to learn tree structures, through repeated applications of backpropagation at different branching points of a tree, in an auto-associative manner (which was named auto-associative memory, or RAAM). Since trees are a common symbolic form, this approach is widely applicable in learning symbolic structures. Similarly, Giles and associates (see e.g. Giles and Gori, 1998) used backpropagation for learning finite-state automata, another common symbolic structure. Connectionist learning algorithms combine the advantages of their symbolic counterparts with the connectionist characteristics of being noise/fault tolerant and being capable of generalization. For an overview of both symbolic and connectionist learning, see Shavlik and Dietterich (1990).

3.2 Connectionist Knowledge Representation

Although it is relatively difficult to devise sophisticated representations in connectionist models (compared with symbolic models), there have been significant developments of connectionist knowledge representation. Many so-called "high-level" connectionist models have been proposed that employ representation methods that are comparable with, and sometimes even surpass, symbolic representations, and they remedy some problems of traditional representation methods as mentioned earlier. For an overview of connectionist knowledge representation, see Sun and Bookman (1994).

Let us look into some of these developments in detail. First of all, logics and rules can be implemented in connectionist models in a variety of ways. For example, in one type of connectionist system, inference is carried out by constraint satisfaction through minimizing an error function. The process is extremely slow though. Another type of system, as proposed by Shastri and many others in the early 1990's, uses more direct means by representing rules with links that directly connect nodes representing conditions and conclusions, respectively, and inference in these models amounts to activation propagation. They are thus more efficient. They also deal with the so-called variable binding problem in connectionist networks. Those advanced logics as mentioned earlier that go beyond classical logic can also be incorporated into connectionist models (see, e.g., Sun, 1994).

Aggregate information can also be incorporated into connectionist models. A system developed by Miikkulainen and Dyer (1991) encodes scripts through dividing input units of a backpropagation network into segments each of which encodes an aspect of a script in a distributed fashion. The system is capable of dealing with incomplete (missing) information, inconsistent information, and uncertainty. There are also localist alternatives (such as those proposed by Lange and Dyer in 1989 and by Sun in 1992), in which a separate unit is allocated to encode an aspect of a frame.

Search, the main means of utilizing knowledge in a representation, is employed or embedded in connectionist models. Either an explicit search can be conducted through a settling or energy minimization process (as discussed earlier), or an implicit search can be conducted in a massively parallel and local fashion. Symbolic search requires global data retrieval and is thus very costly in terms of time. Global energy minimization (as in some connectionist models) is also time consuming. Local computation in connectionist models is a viable alternative: Knowledge is stored in a network connected by links that capture search steps (inferences) directly. Search amounts to activation propagation (by following links, similar to semantic networks in a way), without global control, monitoring, or storage.

The advantage of connectionist knowledge representation is that such representation can not only handle symbolic structures but goes beyond them by dealing with incompleteness, inconsistency, uncertainty, approximate information, and partial match (similarity) and by treating reasoning as a complex dynamic process. However, developing representation in highly structured media such as connectionist networks is inherently difficult.

4 Hybrid Models

Given the different emphases and strengths of connectionist and symbolic systems, it seems plausible that combining them would be a promising avenue for developing more robust, more powerful, and more versatile systems. The need for such systems has been slowly but steadily growing. There has been a great deal of research recently, which leads to so-called hybrid systems.

The relative advantages of connectionist and symbolic models have been amply argued for — see e.g. Waltz and Feldman (1986), Smolensky (1988), and Sun (1994). The computational advantage of the combination thereof is thus relatively easy to justify: Naturally, we want to take advantages of both types of models and their synergy. See e.g. Dreyfus and Dreyfus (1987), Sun and Bookman (1994), and Sun and Alexandre (1997) for further justifications.

There are many important issues to be addressed in developing hybrid connectionist-symbolic systems. These issues concern architectures, learning, and various other aspects. First, hybrid models likely involve a variety of different types of processes and representations. Multiple heterogeneous mechanisms interact in complex ways. We need to consider ways of structuring these different components; in other words, we need to consider *architectures*, which thus occupy a more prominent place in this line of research. Second, although purely connectionist models, which constitute part of any hybrid system, are known to excel in their learning abilities, hybridization makes it more difficult to perform learning. In a way, hybrid systems inherit the difficulty with learning from the symbolic side and mitigate to some extent the advantage that (purely) connectionist models have in terms of learning.

4.1 Architectures and Representations

We divide systems into two broad categories: *single-module* and *multi-module* architectures. Among single-module systems, along the representation dimension, there can be the following types (Sun and Bookman, 1994): symbolic, localist (with one distinct node for representing each concept; see, for example, Lange and Dyer, 1989; Shastri and Ajjanagadde, 1993) and distributed (with a set of non-exclusive, overlapping nodes for representing each concept; see, for example, Pollack, 1990). Among multi-module systems, we can distinguish between *homogeneous* and *heterogeneous* systems. Homogeneous systems are similar to single-module systems, except they contains several replicated copies of the same structure, each of which can be used for processing the same set of inputs, to provide redundancy for various reasons.

Heterogeneous multi-module systems are more interesting. This category constitutes the true hybrid systems. As an example, CONSYDERR (Sun, 1994) belongs to this category. It consists of two levels: the top level is a network with localist (symbolic) representation, and the bottom level is a network with distributed representation; concepts and rules are diffusely represented in the bottom level by sets of feature units overlapping each other. This is a similarity-based representation, in which concepts are "defined" in terms of their similarity to other concepts in these representations. The localist network is linked with the distributed network by connecting each node in the top level representing one concept to all the feature nodes in the bottom level representing the same concept. Through a 3-phase interaction between the two levels, the model is capable of both rule-based and similarity-based reasoning with incomplete, inconsistent and approximate information, and accounts for a large variety of seemingly disparate patterns in human reasoning data.

A variety of distinctions can be made here. First of all, a distinction can be made in terms of *representations* of constituent modules. In heterogeneous multi-module systems, there can be different combinations of different types of constituent modules: for example, a system can be a combination of localist modules and distributed modules (as in CONSYDERR discussed above) or it can be a combination of symbolic modules and connectionist modules (either localist or distributed; for example, as in SCRUFFY described by Hendler in Barnden and Pollack (1991). Some of these combinations can be traced to the ideas of Smolensky (1988), who argued for the dichotomy of conceptual and subconceptual processing.

Another distinction that can be made among heterogeneous multi-module systems is in terms of the *coupling* of modules: a set of modules can be either loosely coupled or tightly coupled. In loosely coupled situations, modules communicate with each other, primarily through message passing, shared memory locations, or shared files, as in, for example, SCRUFFY (see Hendler's chapter in Barnden and Pollack, 1991). Such loose coupling enables some loose forms of cooperation among modules. One form of cooperation is in terms of pre/postprocessing vs. main processing: while one or more modules take care of pre/postprocessing, such as transforming input data or rectifying output data, a main module focuses on the main part of the task. Another form of cooperation is through a master-slave relationship: while one module maintains control of the task at hand, it can signal other modules to handle some specific aspects of the task. For example, a symbolic expert system, as part of a rule, may invoke a neural network to perform a specific classification or decision making. Yet another form of cooperation is the equal partnership of multiple modules. In this form, the modules (the equal partners) can consist of (1) complementary processes, such as in SOAR/ECHO (Johnson et al in Sun and Alexandre, 1997), or (2) multiple functionally equivalent but structurally and representationally different processes, such as in CLARION (Sun and Peterson, 1998), or (3) they may consist of multiple differentially specialized and heterogeneously represented experts each of which constitutes an equal partner in accomplishing a task.

In tightly coupled systems, on the other hand, the constituent modules interact through multiple channels (e.g., various possible function calls), or may even have node-to-node connections across two modules, such as CONSYDERR (Sun, 1994) in which each node in one module is connected to a corresponding node in the other module. There are a variety of forms of cooperation among modules, in ways quite similar to loosely coupled systems.

4.2 Learning

Learning, which can include (1) learning the content (knowledge) in a hybrid model or (2) learning and developing the model architecture itself, is a fundamental issue that is clearly difficult. However, learning is indispensable if hybrid systems are ever to be scaled up. Over the years, some progress on learning has been made. While some researchers have tried to extend connectionist learning algorithms to learn complex symbolic representations, others have instead incorporated symbolic learning methods. For example, Sun and Peterson (1998) presented a two-module model CLARION for learning sequential decision tasks, in which symbolic knowledge is extracted on-line from a reinforcement learning connectionist network and is used, in turn, to speed up connectionist learning and to facilitate transfer. The work showed not only the synergy between connectionist and symbolic learning, but also that symbolic knowledge can be learned autonomously on-line, from subsymbolic knowledge, which is very useful in developing autonomous agents.

Learning methods that may be applied to hybrid systems include gradient descent and its many variations (extending typical connectionist learning algorithms), Expectation-Maximization and its many instantiations (including hidden Markov model algorithms), search algorithms, evolutionary algorithms, and heuristic methods (such as decision trees or rule induction; see Shavlik and Dietterich, 1990). Some of these methods may be combined with others (as in Sun and Peterson, 1998), which likely results in improved learning. There are a variety of other learning approaches being proposed also, including many rule extraction or insertion algorithms.

There is a sense that future advance in this area is dependent on progress in the development of new learning methods for hybrid systems and the integration of learning and complex symbolic representations. As mentioned above (see Sun and Peterson, 1998), symbolic representation and reasoning may well emerge from subsymbolic processes through learning, and thus an intimate and synergistic combination of symbolic and subsymbolic learning processes should be pursued.

4.3 Discussion

Despite the diversity of the approaches discussed thus far, there is an underlying common theme: bringing together symbolic and connectionist models to achieve a synthesis of the two seemingly radically different paradigms.

There is no doubt that we need to invest substantial efforts in comparing and analyzing various paradigms, especially the symbolic paradigm and the connectionist paradigm, to reveal their respective strengths and limitations, as well as underlying assumptions, which can lead to better understanding and more rapid advancement of this field. Although significant advances have been achieved so far, this process is still on-going today, and it appears that it will not be concluded until a better understanding is achieved of the nature of intelligence in computational terms.

In the meantime, it is an appealing idea that representation and learning techniques from both symbolic processing models and connectionist network models shall be brought together to tackle problems that neither type of model alone can apparently handle very well.

One such problem is the modeling of human cognition, which requires dealing with a variety of cognitive capacities. Several researchers (e.g., Smolensky, 1988; Dreyfus and Dreyfus, 1987) have consistently argued that cognition is multi-faceted and better captured with a combination of symbolic and connectionist processes. Many methods and frameworks reviewed above share the belief that connectionist and symbolic methods can be usefully integrated, and such integration may lead to significant advances in our understanding of cognition. It thus appears that hybridization is a theoretically sound approach, in addition to being a practically expedient approach.

5 References

Barnden, J.A. and Pollack, J.B. (eds.), (1991). *Advances in Connectionist and Neural Computation Theory*, Hillsdale, NJ: Lawrence Erlbaum Assoc.

E. Davis, (1990). *Representations of Commonsense Knowledge.* Morgan Kaufmann, San Mateo, CA.

Dreyfus, H. and S. Dreyfus, (1987). *Mind Over Machine*, The Free Press. New York.

Giles, L. and Gori, M. 1998. *Adaptive Processing of Sequences and Data Structures.* New York: Springer.

Lange, T. and M. Dyer, (1989). High-level inferencing in a connectionist network. *Connection Science*, 1:181–217.

J. McCarthy, (1968). Programs with Common Sense. In: M. Minsky (ed.), *Semantic Information Processing*, MIT Press, Cambridge, MA.

J. McClelland, D. Rumelhart, and the PDP Research Group, (1986). *Parallel Distributed Processing: Explorations in the Microstructures of Cognition*, MIT Press, Cambridge, MA.

R. Michalski, (1983). A theory and methodology of inductive learning. *Artificial Intelligence.* Vol.20, pp.111-161.

R. Miikkulainen & M. Dyer, (1991). Natural language processing with modular PDP networks and distributed lexicons. *Cognitive Science.* 15(3). pp.343-399.

A. Newell and H. Simon, (1976). Computer science as empirical inquiry: symbols and search. *Communication of ACM.* 19. 113-126.

J. Pearl, (1988). *Probabilistic Reasoning in Intelligent Systems.* Morgan Kaufmann, San Mateo, CA.

Pollack, J. (1990). Recursive distributed representation. *Artificial Intelligence.* 46 (1-2). 77-106.

M. Quillian, 1968. Semantic memory. in M. Minsky, (ed.) *Semantic Information Processing*, 216-260. MIT Press, Cambridge, MA.

R. Quinlan, (1986). Inductive learning of decision trees. *Machine Learning.* 1, 81-106.

S. Russell and P. Norvig, (1995). *Artificial Intelligence: A Modern Approach.* Prentice-Hall, Englewood Cliffs, NJ.

J. Shavlik and T. Dietterich, (1990). *Readings in Machine Learning.* Morgan Kaufmann. San Mateo, CA.

L. Shastri and V. Ajjanagadde, (1993). From simple associations to systematic reasoning: A connectionist representation of rules, variables and dynamic bindings. *Behavioral and Brain Sciences*, 16(3), 417-494.

P. Smolensky, (1988). On the proper treatment of connectionism. *Behavioral and Brain Sciences*, 11(1), 1-74.

R. Sun, (1994). *Integrating Rules and Connectionism for Robust Reasoning.* John Wiley and Sons, New York, NY.

R. Sun and F. Alexandre (eds.), *Connectionist Symbolic Integration.* Lawrence Erlbaum Associates, Hillsdale, NJ. 1997.

R. Sun and L. Bookman, (eds.) *Computational Architectures Integrating Neural and Symbolic Processes.* Kluwer Academic Publishers, 1994.

Sun, R. and T. Peterson, (1998). Autonomous learning of sequential tasks: experiments and analyses. *IEEE Transactions on Neural Networks*, Vol.9, No.6, pp.1217-1234.

A. Turing, (1950). Computing machinery and intelligence. *Mind*, Vol.59, 433-460.

Waltz, D. and J. Feldman (eds.), (1986). *Connectionist Models and Their implications*, Ablex. Norwood, NJ.

L. Zadeh, (1988). Fuzzy Logic, *Computer*, Vol.21, No.4, 83-93