Neural Networks and Structured Knowledge: Knowledge Representation and Reasoning

FRANZ J. KURFESS
Department of Computer and Information Sciences, New Jersey Institute of Technology, Newark, NJ 07102
franz@cis.njit.edu

Abstract. This collection of articles is the first of two parts of a special issue on "Neural Networks and Structured Knowledge." The contributions to the first part shed some light on the issues of knowledge representation and reasoning with neural networks. Their scope ranges from formal models for mapping discrete structures like graphs or logical formulae onto different types of neural networks, to the construction of practical systems for various types of reasoning. In the second part to follow, the emphasis will be on the extraction of knowledge from neural networks, and on applications of neural networks and structured knowledge to practical tasks.

Keywords: neural networks, knowledge representation, structured knowledge reasoning, connectionism, symbol processing, hybrid systems

1. Introduction

The ability to exchange and preserve knowledge has had a major impact on the fate of humanity, and arguably is a major factor in the evolutionary success of the human species. And for thousands of years, thinkers have sought for ways to improve communication of knowledge, its representation and preservation, and for innovations to increase the descriptive power of whatever methods and mechanisms have been in use to deal with knowledge [1]. In the history of mankind, the most successful knowledge representation and communication mechanism has been spoken and written language. Especially written language, with the potential to be preserved over timespans that surpass that of humans, has had a major impact on the preservation of knowledge. Three particular variations of written language have been especially successful: logographic writing systems, syllabic and alphabet-based ones. Logographic systems use a symbol or icon for one word, and sometimes the symbol carries a pictorial resemblance to the object represented; an example of such a system is Chinese. In syllabic scripts, a symbol stands for one syllable, and words are composed from the symbols representing the respective syllables. One of the major problems especially with logographic systems is the huge number of symbols that is needed to represent knowledge: in a pure logographic script, a separate symbol is required for every word in the language. This problem is not an issue with alphabetic scripts, which use a relatively small set of symbols, the alphabet, and composes syllables and words as sequences of these basic symbols. The expression of words by sequences of symbols in a natural way leads to the representation of sentences as sequences of words, augmented by punctuation marks for easier processing. From a knowledge representation perspective, an important aspect of such a writing system is its compositionality: Smaller units can be composed according to certain rules into larger ones. Independent of the writing system, the symbols need to be interpreted by the user, and thus serve as the carriers of information and knowledge. These skills of reading and writing have been very important for the preservation and distribution of human knowledge.

Over the last fifty years or so, most of the research into the utilization of computers for dealing with knowledge has been performed in the domain of Artificial Intelligence. The most influential and commercially successful approaches to knowledge processing are based on the representation and manipulation of knowledge as sequences of symbols, which ultimately have to be interpreted by humans in order to gain access to the knowledge contained therein. This representation of knowledge is governed by syntactical rules which clearly specify the permitted configuration of symbol sequences. The manipulation of knowledge relies on inference rules based on or derived from mathematical logic, which again provide concise instructions about permissible operations on sequences of symbols. Due to its heavy reliance on symbols, this family of approaches is frequently referred to as symbolic or symbol-oriented knowledge processing. In addition to the fundamental problem of symbols and their associated meanings mentioned above, symbol-oriented approaches suffer from a number of additional conceptual and technical problems: Similarity in internal representation does not imply the similarity of the corresponding objects or concepts, and vice versa; a small error in the representation or processing can have severe consequences; the computation time and space requirements for similar tasks can be vastly different. These and some other considerations strengthen the case for an alternative representation mechanism, frequently termed sub-symbolic, indicating that there are important issues to be dealt with at a level below symbols. In many cases, neural networks serve as the underlying computational mechanism for this alternative approach, and a lot of this research has been performed under the term connectionism [2–8]. One of the ideas common to many of these approaches is that of distributed representation: an item is not represented by one single symbol or sequence of symbols, but by the combination of many small representational entities, often referred to as microfeatures. The concept 'apple', for example, would not be represented as a string of characters, but as an entity that has the properties 'fruit', 'edible', 'round', ('yellow' or 'red' or 'green'), and other characteristics of apples. Such representation schemes have some favorable properties like similarity-based access, fault tolerance, quick response time, etc. On the other hand, their internal workings are usually not easy to inspect, formal aspects like correctness or completeness are difficult or impossible to assess, and most existing systems are research prototypes. This makes them complementary to the symbol-oriented ones, and in fact a whole class of hybrid approaches incorporating the favorable aspects of both symbol-oriented as well as subsymbolic approaches has been investigated over the last years [9–12].

This special issue of "Applied Intelligence" deals with the usage of neural networks for knowledge representation and manipulation purposes. It consists of six contributions investigating various knowledge-representation and reasoning mechanisms based on neural networks. A companion issue [13] will concentrate on the issues of extracting knowledge from neural networks, and practical applications of knowledge processing based on neural networks.

In the remainder of this editorial, I will review the terms data, knowledge, and information as used in this context and discuss some issues of knowledge representation and the respective operations. This is followed by a brief preview on the individual contributions, and how they fit into the overall context of knowledge representation and reasoning with neural networks.

2. Data, Knowledge, and Information

It is important to clarify the terminology used, and in the following Section I will present some attempts at defining the terms "data", "knowledge", and "information". There are not many definitions of knowledge specifically targeted at and suitable for the representation and processing of knowledge with computers, and most definitions from general dictionaries are too broad for our particular context here. The definition attempts are followed by a short clarification with particular emphasis on aspects that are especially important for the purpose of using neural networks as computational tools for the representation and processing of knowledge.

2.1. Data

Definitions for the term "data:"

1. The first definition is from the Collins COBUILD dictionary [14], and describes the term quite well for our purposes [14]: *Information, usually in the form of facts or statistics that you can analyse, or that you use to do further calculations*.
2. The second one from the Infopedia encyclopedia [15] is somewhat more comprehensive, but also captures the essential aspects:
   (a) factual information (as measurements or statistics) used as a basis for reasoning, discussion, or calculation;

(b) information output by a sensing device or organ that includes both useful and irrelevant or redundant information and must be processed to be meaningful;
(c) information in numerical form that can be digitally transmitted or processed.

Important aspects: Out of the three terms illuminated in this section, this one is relatively easy to characterize in our context. For computer science in general, the term is used to describe items that are used in computations. Data structures describe generic elements with a clearly defined internal structure, specific properties, and procedures for manipulation. The input and output of programs in general, and of neural networks in particular, are also often referred to as data. A common feature is that data typically are arranged into a rigid, simple structure (e.g., tables or arrarys). It is interesting to note that most definitions of the term data rely on the term information.

2.2. Knowledge

Definitions of the term "knowledge:"

1. The following characterization given by Kasabov [16] refers to knowledge representation and processing in a system: *Concise presentation of previous experience which can be interpreted in a system.*
2. In one of the very influential papers on knowledge and computers, Newell [17] defines knowledge with respect to agents and on the basis of rationality: *Whatever can be ascribed to an agent, such that its behaviour can be computed according to the principle of rationality*. Newell's principle of rationality states that "If an agent has knowledge that one of its action will lead to one of its goals, then the agent will select that action" [17].
3. The following definition is from the Infopedia encyclopedia [15]:

   (a) the fact or condition of knowing something with familiarity gained through experience or association;
   (b) acquaintance with or understanding of a science, art, or technique
   (c) the fact or condition of being aware of something; the range of one's information or understanding; the circumstance or condition of apprehending truth or fact through reasoning: cognition; the fact or condition of having information or of being learned;
   (d) the sum of what is known: the body of truth, information, and principles acquired by mankind.

Important aspects: The definitions for knowledge given by Newell and Kasabov both rely on the intuition of the reader, and Newell's is circular by referring to rationality, which refers back to knowledge [18].

Many methods to represent knowledge in computers rely on a separation of knowledge items, and relationships between these items. This leads to the representation of knowledge as a graph, with nodes representing the items, and vertices the relationships. The items itself can be atomic, or composed of other items. A prominent example of this graph-based approach are semantic networks, and some early work related to our topic of neural networks and structured knowledge is based on semantic networks [19]. In human knowledge representation, many of the relations expressed explicitly in such graph-based models are hidden, e.g., as associations, or only accessible if triggered by some related event.

2.3. Information

Definitions of the term "information:"

1. This definition is again from Kasabov [16]: *Collection of structured data. In its broad meaning it includes knowledge as well as simple meaningful data*.
2. The following two definitions are from different dictionaries; it is interesting to note that they emphasize rather different aspects of information. The first one is from Collins [20]:

   (a) knowledge acquired through experience or study;
   (b) knowledge of specific and timely events or situations; news;
   (c) the act of informing or the condition of being informed;
   (d) the results derived from the processing of data according to programmed instructions;
   (e) another word for data.

3. This definition is again from Infopedia [15]:

   (a) the communication or reception of knowledge or intelligence;

(b) knowledge obtained from investigation, study, or instruction; intelligence, news; facts, data;
(c) the attribute inherent in and communicated by one of two or more alternative sequences or arrangements of something (as nucleotides in DNA or binary digits in a computer program) that produce specific effects;
(d) a signal or character (as in a communication system or computer) representing data; something (as a message, experimental data, or a picture) which justifies change in a construct (as a plan or theory) that represents physical or mental experience or another construct;
(e) a quantitative measure of the content of information; specif: a numerical quantity that measures the uncertainty in the outcome of an experiment to be performed.

Important aspects: This term has a precise meaning in some specific domains, such as information theory. In our context here, it is sometimes used in a similar way as in information theory to describe the information capacity of neural networks [21–24]. Frequently it is used in rather more generic way, both in discussions of knowledge representation as well as in computer science in general.

For our purposes here, the emphasis will lie on the term knowledge. The term *structured knowledge* indicates that the underlying conceptual notion of knowledge relies on entities to be represented, together with interrelationships between these entities. Frequently this is achieved through the usage of graphs, where nodes stand for the entities, and vertices for the connections. Graphs are often used as the basis for symbol-oriented approaches to knowledge representation and processing, either directly as in semantic networks, or indirectly to visualize certain properties of the representation method. The treatment of knowledge with neural networks sometimes also relies on a graph-based representation scheme; since in this case conceptual entities correspond to individual neurons, these models are often referred to as "localist". This stands in contrast to "distributed" models, where one entity is represented jointly by several neurons, and conversely each neuron contributes to the representation of several nodes.

Structured knowledge also enhances the difference between data and knowledge. As the term "data structure" indicates, there is an underlying structure to data. This structure, however, is usually the same for all the instances of a particular data type, and the essential information is carried by the values of the individual fields inside an instance of the data type. Operations on such data then rely heavily on the pre-defiend structure of a particular data type. Two typical examples are arrays and records: In an array, each element is of the same type, and operations to access or modify elements depend on the arrangement of the elements into columns and rows. In a database, the general structure is often defined through an entity-relationship diagram, and records composed of fields are used to accomodate individual items. In this case, the structure already is much more important than for the array example, since the relationships between the different fields of the record contribute in an essential way to the information contained in a data base. For the representation of knowledge, the importance of relations between individual entities becomes even stronger: instead of using the same relationships for a whole collection of instances, relationships can be established between any pair or set of entities in the domain under consideration. This scheme is much more flexible on one hand, but requires more overhead for storage and processing.

This difference between data and knowledge can be visualized as a pyramid (see Fig. 1). The "raw" data at the bottom may come from sensors, and represent information about the real world, such as measurements, images, or sound. After some initial procesing, they are mapped into data structures such as arrays or records with pre-defined relationships between the individual items. Further processing then may lead to more structured elements such as records in a data base, or frames in a knowledge-base. Once the emphasis shifts from a collection of elements with identical internal structure to individual elements with separate relationships among each other, in our perspective the transition from data to knowledge takes place. At the top of the pyramid finally is meta-knowledge, or knowledge about knowledge, which expresses more abstract information about the relationships between elements at the knowledge level. As indicated by the arrows on the side of the diagram, the level of abstraction and the information density increase from the bottom to the top, and the degree of detail as well as redundancy are higher at the bottom.

3. Contributions to this Part of the Special Issue

This section provides a brief preview of the contributions in this first part of the special issue on "Neural Networks and Structured Knowledge".

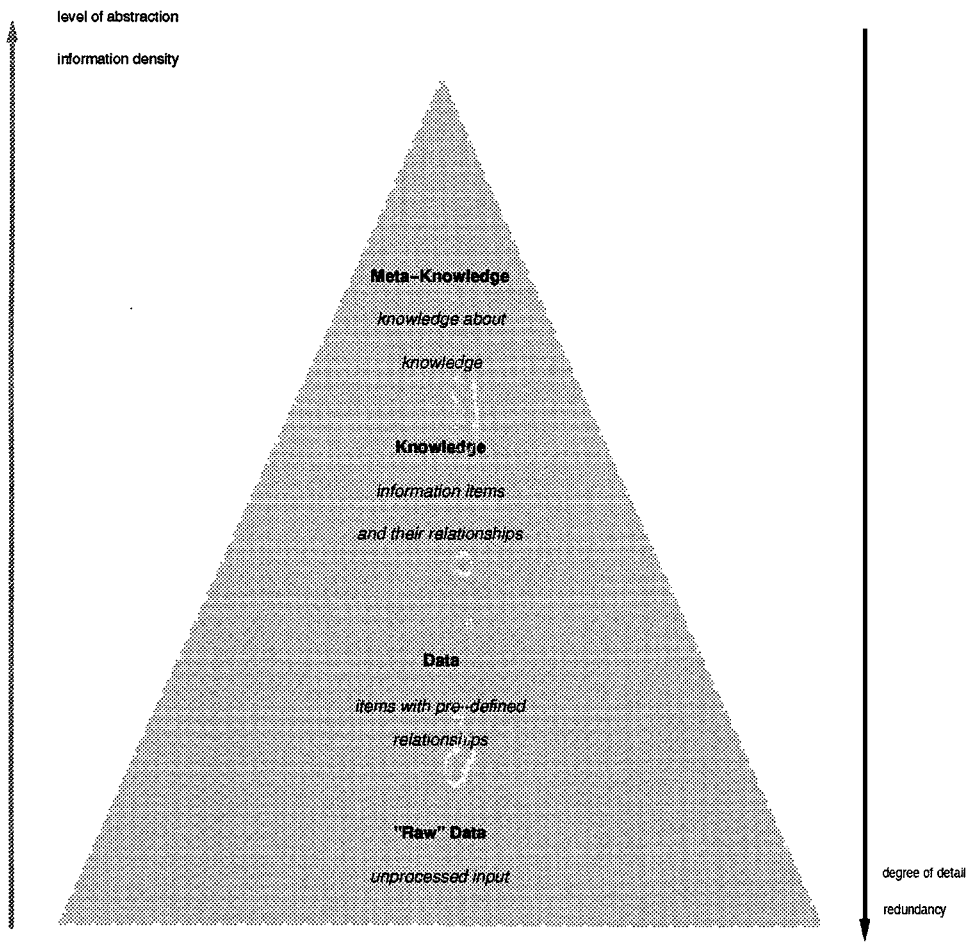

Figure 1. Knowledge pyramid.

3.1. Comparing Structures Using a Hopfield-Style Neural Network

The first contribution by Kristina Schädler and Fritz Wysotzki describes an approach to represent and process labeled graphs in neural networks. Labeled graphs form the basis for many knowledge representation mechanisms, and consequently operations on labeled graphs are very important for processing of structured knowledge. One of the particularly powerful operations is the comparison of two graphs, or graph matching. Unfortunately, this operation is computationally rather expensive, belonging to the class of NP-complete problems. For many applications it is beneficial or necessary to perform approximate graph matching, which checks two graphs not for identity, but for similarity. For this case, the situation is probably even worse because there is no intuitive or commonly agreed upon similarity measure for graphs. This contribution describes a representational scheme for labeled graphs based on Hopfield-style neural networks. This scheme lends itself to a formulation of a similarity measure for the approximate matching task in terms of the minimization of the network's energy function. This similarity function has the additional advantage that user knowledge and preferences about the similarity of graphs can be incorporated by changing the parameters and the dynamics of the network. The approach is evaluated experimentally by applying it to the classification of organic chemical structures and the prediction of their biological activity.

3.2. *Massively Parallel Probabilistic Reasoning with Boltzmann Machines*

A special type of graphs, namely Bayesian networks, and their representation together with operations for probabilistic reasoning, is the topic of the second paper, by Petri Myllymäki. In his contribution, he describes the mapping of a Bayesian network to a Boltzmann machine, which is a stochastic neural network architecture especially well suited for massively parallel evaluation. This evaluation is also known as simulated annealing, and allows the approximate solution of NP-hard optimization problems in a computationally efficient way. A straightforward usage of simulated annealing as global optimization method on Bayesian networks is possible, but very slow. The author proposes a hybrid Bayesion-neural system, which utilizes the Bayesian network for the construction of the model as seen by the user, and the Boltzmann machine for the parallel performance of probabilistic reasoning. Thus the system allows for the construction of neural models from expert knowledge on one hand, and the efficient evaluation of Bayesian reasoning on the other hand. Simulations show that one particular advantage of the system is its scalability: Conventional algorithms suffer from combinatorial explosion when the network size is increased, which is not the case for the massively parallel algorithm. The speedup gained from this parallelization, however, depends on the availability of suitable massively parallel hardware.

3.3. *Approximating the Semantics of Logic Programs by Recurrent Neural Networks*

Reasoning according to first-order predicate logic is the topic of the third paper. In this contribution, the authors Steffen Hölldobler, Yvonne Kalinke and Hans-Peter Störr investigate the relationship between the semantics of first-order logic programs and recurrent networks that approximate the fixed point of the meaning function of such a program. Similar as in Myllymäki's approach, a symbol-oriented method can be mapped onto a neural network with favorable properties like approximate computations, parallel execution, or the possibility to apply learning techniques like backpropagation. In the other direction, the neural network can be converted into a logic program, allowing inspection of or rule extraction from the neural network. In contrast to similar previous approaches by the authors and others, which are essentially limited to propositional logic this article discusses an extension to the mapping between a class of first order logic programs and three-layered feedforward networks. This mapping, at present, is mainly a theoretical result, and does not represent a direct practical solution since it would require the representation of infinitely many elements. This problem, however, may be overcome by using fixed-length distributed representations such as recursive auto-associative memories [25] and their extensions to include labels [26] (see also a forthcoming contribution by Sperduti in the second part of this special issue), holographic reduced representations [27] spattercoding [28], or multiplicative binding [29].

3.4. *The Connectionist Inductive Learning and Logic Programming System*

Knowledge representation and reasoning via logic programming is also one corner point of a massively parallel computational model proposed in the fourth article by Artur S. d'Avila Garcez and Gerson Zaverucha. The propositional logic program is translated into a neural network that can be trained with examples, and then reconverted into a revised logic program. The mapping algorithm in fact is based on previous work by Hölldobler et al. [30], but uses a different type of neural networks. With these bipolar semi-linear neurons, the resulting network still is a massively parallel model for logic programming, but it can also perform inductive learning based on a logic program as background knowledge and backpropagation as learning algorithm. This network is capable of performing both inductive learning from examples and deductive reasoning. The method has been applied to DNA classification problems as test cases, and the results show that it is comparable to or better than any other system investigated by the authors.

3.5. *Advances in SHRUTI—A Neurally Motivated Model of Relational Knowledge Representation and Rapid Inference Using Temporal Synchrony*

The knowledge representation and inference mechanism described by Lokendra Shastri in the fifth contribution, SHRUTI, relies on synchronous firing of neurons for the the propagation of activity in a network of nodes and links. In contrast to the previous approaches, which emphasize the formal correspondence between symbol-oriented representation and processing, and the respective neural counterparts, the goal of SHRUTI is to demonstrate that a neurally plausible network is capable of substantial knowledge representation and reasoning tasks. In this article, the author describes more recent enhancements and developments over the original model, e.g., as described in [31, 32]. Some of these enhancements are of a more fundamental nature, such as the improvements of the expressiveness by allowing negated facts, inconsistent beliefs, or evidential facts and rules, whereas others are targeted towards more efficient evaluation and better usability. In addition to its computational aspects, the neurally inspired architecture of SHRUTI allows some predictions about the nature of reflexive reasoning and aspects of working memory in humans.

3.6. A Hybrid Architecture for Situated Learning of Reactive Sequential Decision Making

In the sixth and final article of this first part, Ron Sun, Todd Peterson, and Edward Merrill describe an approach that combines localist and distributed representations into the hybrid model CLARION. Whereas various aspects, applications, and enhancements of the CLARION model have been described in previous publications (see the article by Sun et al. for references), this paper concentrates on reactive sequential decision tasks. Such tasks are especially important for autonomous agents that need to make quick decisions about actions to take on the basis of currently available perceptual information together with background knowledge. One task, the simulated navigation of an underwater vessel through a minefield towards a target location, is a complex and realistic one developed by the Naval Research Lab. These experiments show that the hybrid model clearly outperforms a version that utilizes only procedural knowledge acquired through reinforcement learning. The experiments also expose a significant similarity with human performance on the same task, thus demonstrating the cognitive validity of the architecture.

4. An Outlook to the Second Part: Rule Extractions and Applications

The second part [13] of the collection of articles on neural networks and structured knowledge will be devoted to the issue of knowledge extraction from neural networks, and to applications of neural-network based approaches to structured knowledge. Knowledge extraction from neural networks is interesting from two perspectives: It allows for a limited inspection of the "contents" of a neural network, thus making the foundations for the response of a network to a given input more comprehensible to humans. It is also the basis for many approaches to hybrid systems, combining neural networks with symbol-oriented approaches like expert systems or theorem provers. An existing set of rules, for example, can be refined by converting it into a neural network and training the neural network with example data, thus modifying the given rules so that they also fit the example data.

The viability of a novel approach for knowledge processing with computers should eventually be checked by its application to realistic problems. Whereas some of the contributions in this first part describe applications briefly, the emphasis lies on the presentation of the approach itself. The second part will contain contributions that concentrate on the application of neural networks to tasks requiring the processing of structured knowledge, such as the recognition of handwritten digits, or the prediction of structural properties for chemical compounds.

Acknowledgments

The papers published in these two special issues have been selected from around forty contributions submitted in response to the call for papers. I would like to thank all contributors for their efforts, especially those whose contributions could not be accepted here due to space restrictions. My thanks also go to the more than 150 referees. Without their help it would have been impossible to put this collection together. Many of them offered valuable suggestions for improving the quality and presentation of the reviewed contributions.

This set of two special issues is an outcome of a number of activities pursued over the last few years. Most directly related is a workshop on "Neural Networks and Structured Knowledge" held during the European Conference on Artificial Intelligence (ECAI '96) in Budapest, Hungary [9]. Similar workshops and symposia took place in combination with other conferences like the International Joint Conference on Artificial Intelligence (IJCAI '95) in Montreal [10], the German Conferences on Artificial Intelligence in Berlin (KI '93) and Saarbrücken (KI '94) [33–35] the Fall Schools on Connectionism and Neural Networks

(HekoNN '94 and '95) [36, 37], and the MIX '97 Fall Symposium on Hybrid Systems organized by Wolfgang Ertel and Bertram Fronhöfer. I would like to thank the attendees of these workshops, the authors of papers, and of course the organizers.

Most of my work in this area was performed during my employment at the University of Ulm, in the Neural Information Processing Department directed by Prof. Günther Palm, with partial funding from the German Ministery for Research and Technology. While at New Jersey Institute of Technology, related activities have been funded by the State of New Jersey in the SBR program.

References

1. C. Van Doren, "*A history of knowledge*," *Ballantine Books*, 1991.
2. J.A. Feldman, "Connectionist models and parallelism in high level vision," *Computer Vision, Graphics, and Image Processing*, vol. 31, pp. 178–200, 1985.
3. G. Hinton, "Connectionist learning procedures," Technical Report CMU-CS-87-115, Carnegie-Mellon University, Pittsburgh, PA 15213, 1987.
4. D.E. Rumelhart, G.E. Hinton, and J.L. McClelland, "A general framework for parallel distributed processing," *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, Bradford Books: Cambridge, MA, vol. 1, 1986.
5. J.A. Fodor and Z.W. Pylyshyn, *Connectionism and cognitive architecture: A critical analysis*, Elsevier Science Publishers B.V.: Holland, 1988.
6. P. Smolensky, "On the proper treatment of connectionism," *Behavioral and Brain Sciences*, vol. 1, pp. 1–74, 1988.
7. R. Pfeifer, Z. Schreter, F.F. Soulie, and L. Steels (Eds.). *Connectionism in Perspective*, Amsterdam, 1989. Elsevier North Holland.
8. S. Addanki, "Connectionism," in *Encyclopaedia of Artificial Intelligence*, edited by S.C. Shapiro, Wiley: New York, pp. 268–274, 1992.
9. F.J. Kurfeß (Ed.), "*Neural Networks and Structured Knowledge*," *European Conference on Articial Intelligence (ECAI '96) Workshop Proceedings*, Budapest, 1996. European Coordinating Committee for Artificial Intelligence (ECCAI).
10. R. Sun and F. Alexandre (Eds.), *Connectionist-Symbolic Integration*, Lawrence Erlbaum, 1997.
11. L. Medsker, *Hybrid Neural Network and Expert Systems*, Kluwer: Boston, 1994.
12. S. Goonatilake and S. Khebbal. *Intelligent Hybrid Systems*, John Wiley & Sons, 1995.
13. F.J. Kurfeß. Special issue on "Neural Networks and Structured Knowledge: Rule Extraction and Applications" (guest editor). *Applied Intelligence*, vol. 11, no. 2, 1999.
14. J. Sinclair (Ed.), *Collins Cobuild English Language Dictionary*, Collins: London, Glasgow, 1987.
15. M.-Webster, *Infopedia 2: The Ultimate Multimedia Encyclopedia and Reference Library*, Softkey/Merriam-Webster, 1994. CD-ROM.
16. N.K. Kasabov, *Foundations of Neural Networks, Fuzzy Systems, and Knowledge Engineering*, MIT Press, 1996.
17. A. Newell, "The knowledge level," *Artificial Intelligence*, vol. 18, pp. 87–127, 1982.
18. A.G. Hoffmann, *Paradigms of Artificial Intelligence*, Springer, 1998.
19. L. Shastri and J.A. Feldman, "Evidential reasoning in semantic networks: A formal theory," *IJCAI '85*, pp. 465–474, 1985.
20. P. Hanks (Ed.), *Collins Dictionary of the English Language*, William Collins Sons & Co. Ltd: London, Glasgow, 1979.
21. H. Bosch and F.J. Kurfeß, "Incomplete interconnectivity in neural associative memories," *Neural Networks*, vol. 11, No. 5, pp. 869–876, July 1998.
22. H.J. Bentz, M. Hagström, and G. Palm, "Information storage and effective data retrieval in sparse matrices," *Neural Networks*, vol. 2, no. 4, pp. 289–294, 1989.
23. G. Palm, "Rules for synaptic changes and their relevance for the storage of information in the brain," in *Cybernetics and Systems Research*, edited by R.Trappl, North Holland, 1982.
24. G. Palm, "On the information storage capacity of local learning rules," *Neural Computation*, vol. 4, no. 5, pp. 703–711, 1992.
25. J.B. Pollack, "Recursive distributed representations," *Artificial Intelligence*, vol. 46, pp. 77–105, 1990.
26. A. Sperduti, "Labeling RAAM," Technical Report TR-93-029, International Computer Science Institute, Berkeley, CA, 1993.
27. T.A. Plate, "Distributed representations and nested compositional structure," Ph.D. Thesis, Department of Computer Science, University of Toronto, 1994.
28. P. Kanerva, "Binary spatter-coding of ordered k-tuples," in *Artificial Neural Networks—ICANN Proceedings*, edited by C. von der Malsburg, Vj. von Seelen, J.C. Vorbruggen, and B. Sendhoff, vol. 1112 of *Lecture Notes in Computer Science*, Springer: Berlin, pp. 869–873, 1996.
29. R.W. Gayler, "Holographic networks are hiking the foothills of analogy," *Neural Computing Surveys*, vol. 2, 1999.
30. S. Hölldobler and Y. Kalinke, "Towards a new massively parallel computational model for logic programming," in *Combining Symbolic and Connectionist Processing*, edited by M. Hilario, ECAI'94 Workshop W 14, Amsterdam, pp. 68–77, August 1994.
31. V. Ajjanagadde and L. Shastri, "Rules and variables in neural nets," *Neural Computation*, vol. 8, no. 1, pp. 121–134, 1991.
32. L. Shastri and V Ajjanagadde, "From simple associations to systematic reasoning: A connectionist representation of rules, variables, and dynamic bindings using temporal synchrony," *Behavioral and Brain Sciences*, vol. 16, pp. 417–494, 1993.
33. G. Paaß and F.J. Kurfeß (Eds.), *Wissensverarbeitung mit neuronalen Netzen (Knowledge Processing with Neural Networks)*, number 221 in GMD-Studien, Schloß Birlinghoven, 53757 Sankt Augustin, Germany, September 1993. Gesellschaft für Mathematik und Datenverarbeitung (GMD). Workshop KI'93.
34. G. Paaß and F.J. Kurfeß, "Wissensverarbeitung mit neuronalen Netzen," in *Grundlagen und Anwendungen der Künstlichen Intelligenz-17. Fachtagung für Künstliche Intelligenz (KI'93)*, edited by O. Herzog, T. Christaller, and D. Schütt, Informatik aktuel, Subreihe Künstliche Intelligenz, Springer Verlag: Berlin, pp. 217–225, 1993.
35. F.J. Kurfeß and G. Paaß (Eds.), *Integration Neuronaler und Wissensbasierter Ansätze*, number 242 in GMD-Studien, D-53754 Sankt Augustin, Gesellschaft für Informatik (GI), Gesellschaft für Mathematik und Datenverar-beitung (GMD). Workshop

at the KI'94 Conference, Saarbrücken, Germany, September, 1994.
36. I. Duwe, F.J. Kurfeß, G. Paaß, and S. Vogel (Eds.), *Konnektionismus und neuronale Netze—Beiträge zur Herbstschule HeKoNN 94*, number 242 in GMD-Studien, D-53754 Sankt Augustin, Oktober 1994.
37. F.J. Kurfeß, "Wissensverarbeitung mit neuronalen Netzen," in G. Dorffner, K. Möller, G. Paaß, and S. Vogel (Eds.), *Konnektionismus und neuronale Netze—Beiträge zur Herbstschule HeKoNN'95*, GMD-Studien, D-53754 Sankt Augustin, pp. 211–223, Oktober 1995. Gesellschaft für Mathematik und Datenverarbeitung (GMD).

Franz J. Kurfess is the director of the Software Engineering Lab at the Computer and Information Sciences Department, New Jersey Institute of Technology (NJIT). His research activities are centered around knowledge management systems, in particular hybrid systems combining various methods for storing, processing, accessing, and presenting knowledge. Before joining NJIT, he worked in the areas of hybrid systems, neural networks, and parallel inference mechanisms at the University of Ulm, Germany, the International Computer Science Institute in Berkeley, California, and the Technical University in Munich, Germany.