JOURNAL OF MATHEMATICAL PSYCHOLOGY **41**, 287–292 (1997)
ARTICLE NO. MP971164

BOOK REVIEW

Exploring the Diversity of Artificial Neural Network Architectures

Simon Haykin. **Neural Networks: A Comprehensive Foundation.**
New York: MacMillan, 1994. v + 696 pp. $69.95. ISBN 0-02-352761-1.

Reviewed by Richard M. Golden

Simon Haykin is Professor of Electrical and Computer Engineering at MacMaster University, Hamilton, Canada. In 1967, Professor Haykin received his Ph.D. degree in Electrical Engineering from the University of Birmingham, England. His research interests include nonlinear dynamics, neural networks, adaptive filters, and their applications. Haykin was elected Fellow of the Royal Society of Canada. He was awarded the McNaughton Gold Medal, IEEE (Region 7), in 1986. His textbook *Adaptive Filter Theory* (Prentice Hall, 1991) is widely used in engineering departments throughout the nation. He is also the Editor for the Wiley–Interscience series *Adaptive and Learning Systems for Signal Processing, Communications, and Control*. Richard M. Golden is Associate Professor of Cognitive Science in the Cognition and Neuroscience Program in the School of Human Development, University of Texas at Dallas. His primary research interests are mathematical models of higher level cognitive processes and the mathematical analysis of high-dimensional nonlinear artificial neural network models. He has recently completed the book *Mathematical Methods for neural Network Analysis and Design* (MIT Press, 1996).

An artificial neural network (ANN) system is a mathematical algorithm which represents information and processes information according to methods which have been inspired by current knowledge of brain structure and brain function. Artificial neural networks have been used by neuroscientists to build both quantitative and qualitative models of neural systems. They have been used by psychologists to build models of behavior. In addition, ANN systems have been used by engineers to solve specific engineering problems.

Most ANN systems can be characterized as consisting of a collection of computing *units* (analogous to neurons) which are interconnected via a set of *connection strengths* (analogous to synaptic junctions). Each computing unit's state is represented as a real number which is usually referred to as the unit's *activity level* (analogous to a unit's firing frequency). The connection strengths are the parameters of the ANN system, which are adjusted by the *learning algorithm* as a result of specific experiences. Thus, an ANN system's knowledge of the world is represented not only in the system's architecture but also in terms of the specific values of its connection strength parameters. Typically, information flow is viewed in a highly distributed manner so it is usually convenient to introduced the concept of an *activation pattern vector*, which is simply a list of the activation levels of all units in the network.

Units whose only function is to update their activation levels according to incoming information (external to the ANN system) are referred to as *input units*. Similarly, units whose only function is to generate activation levels which are available to other systems for interpretation and the generation of action are referred to as *output units*. Units which are neither input units nor output units are referred to as *hidden units*. If the input units of a network are connected to a set of output units, then this is an example of a *feedforward* network. Another common example of a feedforward network is a network where the input units are connected to the hidden units, and then the hidden units are connected to the output units. If a network architecture is not a feedforward network, then it is defined as a *recurrent* network architecture because it has feedback loops.

In addition to the wide range of network architectures which are potentially available, there is an equally wide range of learning algorithms which have been developed to train ANN systems. The classical Hebbian learning rule states that when two units are simultaneously activated, the connection strength between them should be incremented so that the likelihood of activation of one unit should be increased when the other unit's activation level is increased. Error-correction learning rules use an *error signal* generated by the network's output and the desired response of the network as the basis for incrementing connection strengths among units. Typically, in an error-correction learning ANN system, small changes to the connection strengths are made when the error signal is small and large changes to the connection strengths are made when the error signal is large.

This brief introduction barely touches upon the wide diversity of existing ANN systems which are currently available but it does make the point that any attempt to

0022-2496/97 $25.00

write a relatively complete and comprehensive integrated review of the field of ANN systems is a challenging task. Haykin, however, has managed to meet this challenge with a comprehensive, nonsuperficial, and readable introduction to artificial neural networks. Because Haykin assumes the reader has approximately the equivalent of an undergraduate degree in engineering, he is able to cover topics at a much more rapid rate than other introductions to artificial neural networks. Haykin is also relatively sensitive to issues concerned with the psychological and neurophysiological motivation behind the basic ANN system architectures.

CHAPTER-BY-CHAPTER OVERVIEW

Chapter 1: *Overview of ANN systems.* Chapter 1, unlike most engineering texts on ANN systems, begins with a brief discussion of the human brain and neurophysiology. With this background, Haykin uses this discussion to introduce an artificial neural network as a massively parallel learning machine. The generic artificial neuron is then introduced. The generic artificial neuron essentially computes a weighted sum of the activity levels of the other neurons in the system and then uses that weighted sum to update its activity level. Haykin also introduces a very useful and important notational scheme: a signal-flow graph block diagram notation for representing artificial neural networks. Haykin uses this notation throughout his textbook to graphically, yet formally, define the structure of a variety of artificial neural network architectures. Haykin also gives a useful overview of the various types of network architectures and discusses the important issue of incorporating prior knowledge into network architectures in order to improve generalization performance.

Chapter 2: *Major learning paradigms and rules.* Chapter 2 introduces the major learning paradigms (unsupervised, supervised, and reinforcement learning) as well as some basic learning rules: Hebbian learning, competitive learning, error-correction learning, and Boltzmann learning. The neural and/or psychological underpinnings of each rule are discussed so that the motivation being the original proposals of these fundamental learning rules is appreciated by the reader. Near the end of Chapter 2, Haykin correctly emphasizes that the environment of an artificial neural network is usually conveniently modeled as a stochastic process. Such a formulation allows one to formally view an artificial neural network as a type of statistical pattern recognition algorithm.

Chapter 3: *Linear associative memory systems.* Chapter 3 introduces the concept of a *linear associative memory* system which can learn to map input vectors into output vectors via a linear transformation. Chapter 3 also emphasizes and explores the differences between two basic types of associative memory systems: *autoassociative memory* (where the system attempts to reconstruct missing information and/or enhance degraded information in the input vector) and *heteroassociative memory* (where the system attempts to map an input vector into some specific output vector). The use of both Hebbian and error-correction learning rules for constructing linear associative and linear autoassociative memory systems are discussed.

Chapter 4: *The Perceptron.* Chapter 4 discusses the classic perceptron artificial neural network which is a quasi-linear pattern classifier. The perceptron algorithm is defined, and a version of the Perceptron Convergence Theorem is stated and proved. Chapter 4 also discusses a closely related linear pattern classifier known as the maximum-likelihood Gaussian classifier. In a thoughtful comparison of these two superficially similar algorithms, Haykin emphasizes that the perceptron learning rule is a nonparametric learning rule designed to partition the stimulus space such that each class of stimuli is separated by a hyperplane (i.e., a line in a two-dimensional stimulus space). This is the *linearly separable condition.* On the other hand, the maximum-likelihood Gaussian classifier assumes that the data from each pattern class are best modeled as a multivariate Gaussian distribution (and hence are not linearly separable). For the maximum-likelihood Gaussian classifier, one then finds the hyperplane which maximizes the likelihood of the observed data.

Chapter 5: *The LMS Algorithm.* Chapter 5 discusses the LMS (least-mean-square) algorithm (also commonly known as the Widrow–Hoff learning rule). First the Wiener–Hopf equations for linear optimum filtering are presented. The solution to these equations is a linear filter known as the Wiener filter, which minimizes the mean-squared error between the filter's observed and desired (target) output. This solution requires one to invert an autocorrelation matrix. The method of steepest descent optimization algorithm is then introduced as an alternative procedure for deriving the Wiener filter. Next, the LMS algorithm is introduced as a stochastic type of steepest descent algorithm (which may be realized as a linear adaptive filter), and relevant results concerning its behavior are briefly discussed.

Chapter 6: *Multilayer feedforward networks.* Chapter 6 introduces multilayer feedforward artificial neural networks and continues to successfully exploit the elegant signal flow digraph notation introduced earlier in the book. Multilayer artificial neural networks are typically nonlinear since the representational power of any given multilayer linear artificial neural network is equivalent to that of a single-layer linear network. The backpropagation learning algorithm is introduced and derived in a manner motivated by the Chapter 5 discussion of the LMS algorithm. Both

nonstochastic (batch mode) and stochastic (pattern mode) versions are discussed as well as stopping criteria. Another useful contribution is the discussion of the computer simulation experiments concerned with the application of backpropagation to specific problems. The computer experiments are fairly effective at introducing key concepts (e.g., initial selection of parameter values and initial selection of number of hidden units) which are necessary for the successful application of backpropagation. For example, in order to avoid "memorizing" the data set and to encourage extraction of the critical discriminating statistical regularities in the data, it is necessary to select the number of hidden units in a multilayer backpropagation network to be neither too small nor too large. The chapter also provides a discussion of problems associated with the convergence of learning algorithms for such networks and some heuristic solutions designed to address these problems. There is a section on methods for "growing" and "pruning" parameters in multilayer networks and another section on how nonlinear optimization methods are relevant to the design of backpropagation learning algorithms. At the end of the chapter, Haykin discusses in some detail two successful applications of multilayer backpropagation networks to the problems of optical character recognition and speech recognition.

Chapter 7: *Interpolation using multilayer feedforward networks.* Chapter 7 begins with a review of Cover's (1965) theorem, which is directly applicable to the analysis of multilayer nonlinear artificial neural networks. Consider a feedforward network with a set of p input units, a set of M hidden units, and a set of output units. Now, select at random a training data set of N p-dimensional activation patterns over the input units. These input unit activation patterns generate N M-dimensional activation patterns over the hidden units. Cover showed that if every subset of m of the N ($m \leqq M$) M-dimensional hidden unit activation patterns is linearly independent (i.e., in *general position*), then (for large M) the probability that the M-dimensional hidden unit activation patterns will be linearly separable is close to 1 for $N > 2M$ and close to 0 for $N < 2M$. Finally, the hidden unit to output unit transformation in most multilayer nonlinear artificial neural networks is usually linear or quasi-linear. Thus, the *capacity* of the feedforward network in this case is defined to be $2M$ pattern vectors (twice the number of hidden units). Note that Cover's theorem provides some justification that a given set of training data (which is not linearly separable in the input space) is likely to be linearly separable in the hidden unit activation vector space. Because this result is applicable to many network architectures (particularly multilayer perceptron and backpropagation networks), it is surprising that the first mention of this result occurs in Chapter 7 as opposed to one of the earlier chapters.

Chapter 7 continues with a discussion of the "interpolation" problem in the context of artificial neural networks. Consider a set of training data consisting of a finite number of input pattern vector/output pattern vector pairs. A feedforward network "solves" the interpolation problem if it can map every input vector in the training data set *exactly* into the output vector determined by the training data. This problem is typically ill posed in the sense that multiple continuous functions (i.e., feedforward networks) can be found which solve the interpolation problem. Thus, one needs to introduce prior knowledge into the solution of the interpolation problem to "select out" the function of interest. Tikhonov's (1963) regularization theory introduces such constraints (which are called "regularization terms") into the interpolation problem. An important result of the regularization theory is that if one is looking for a function which minimizes mean-square error subject to a reasonable smoothness constraint, the resulting parametric form of the function is a weighted sum of multivariate Gaussian probability density functions whose means correspond to the input pattern vectors in the training data set. Haykin also provides a detailed discussion of classical methods for using this result in artificial neural network design and discusses two applications of such networks concerned with the prediction of "chaotic time series" and "adaptive equalization."

Chapter 8: *Classic recurrent artificial neural network for classification.* Chapter 8 discusses the classic Hopfield (1982, 1984), classic Cohen–Grossberg (1983), and classic Boltzmann machine recurrent artificial neural networks. This class of artificial networks consists of a group of units which are symmetrically connected. The units update their activation levels according to a specific rule until the network settles into a stable activation pattern which is interpreted as the network's response. The computing units in the Hopfield (1982, 1984) and Cohen–Grossberg (1983) networks are special cases of the generic artificial neuron reviewed in Chapter 1. The computing unit in the Boltzmann machine is very similar to the generic artificial neural neuron but has stochastic response properties as well. All of these networks can be shown to be "energy-minimization" networks that are implicitly seeking the minimum of a quadratic objective function. The classical Hebbian learning rule for storing patterns in the Hopfield (1982) network and the Boltzmann machine learning rule is also discussed. Haykin also briefly mentions the relevance of Markov chains for analyzing the Boltzmann machine's probabilistic activation updating dynamics and a deterministic version of the Boltzmann machine know as "mean-field theory."

Chapter 9: *Unsupervised learning using the Hebbian learning rule.* Chapter 9 introduces unsupervised learning algorithms which are based upon the Hebbian learning rule. First, Linsker's (1986) artificial neural network model of

self-organized learning in the visual system is reviewed. Linsker's model may be described as a feedforward network architecture where first the connections in the first layer of weights mature, then those is the second layer of weights mature, and so on. The learning rule is a variation of the Hebbian learning rule where the magnitude of the individual weights in the network is bounded during the growth process. Not only does the model develop units which learn to detect features in its environment, but it also manages to qualitatively account for the development of various types of feature detection neurons in the visual cortex. After a brief review of principal component analysis, Haykin notes that a linear neuron model with Hebbian learning is a "maximal eigenfilter" that is implicitly extracting the first principal component from its statistical environment. Haykin then extends the discussion to closely related linear networks which are also implicitly doing a principal component analysis of the data.

Chapter 10: *Unsupervised topographic map learning algorithms.* Chapter 10 introduces the topic of topographic map ANN learning algorithms by discussing the existence of topographic maps (e.g., somatosensory or retinotopic) in the human brain. Briefly, a topographic map ANN learning algorithm is an unsupervised learning algorithm which acquires feature detection units with experience and whose feature detection units which process similar inputs tend to be physically clustered together. Haykin reviews the classic Kohonen (1982) SOFM (self-organizing feature mapping) algorithm, which learns according to the following two-stage process. First, allow all current feature detection units to "compete" via a lateral-inhibition-type process in order to decide which feature detection unit is responding maximally to the training stimulus. Second, teach the feature detection unit and its *physical neighbors* to respond more strongly to that training stimulus in the future. The SOFM algorithm illustrates that these two principles, when correctly instantiated in an ANN, provide a satisfying explanation of how topographic maps in the nervous system might arise through unsupervised learning processes. Haykin also explicitly shows how the SOFM algorithm may be interpreted as a learning vector quantization (LVQ) algorithm. Vector quantization algorithms have been developed in the fields of communication and information theory for the purposes of data compression and bandwidth compression. Near the end of the chapter hierarchical vector quantization schemes are also briefly discussed.

Chapter 11: *Unsupervised learning as average mutual information maximization.* Chapter 11 is concerned with unsupervised learning systems that seek a set of parameters (i.e., connection strengths) that maximize the average mutual information between the learning system's response to a stimulus and that stimulus. After providing a good introduction to information theory and defining the concept of average mutual information, Haykin subsequently uses the average mutual information concept to view a variety of ANN systems as algorithms that seek a set of weights that maximize average mutual information between the input and output of the ANN system architecture. Some the network architectures discussed include a single linear neuron with additive input noise, a self-organizing feature map learning algorithm, and an ANN system for processing radar images.

Chapter 12: *Modular ANN systems.* Chapter 12 discusses the use of modular ANN systems where the computation of the network can be partitioned into multiple, relatively independent "expert" modules. The outputs of the expert modules are then channelled into an integration module which learns to compute an appropriate weighted sum of the outputs of the expert modules in addition to learning appropriate functions for those modules. The presentation of this methodology within the context of a statistical pattern recognition methodology following Jordan and Jacobs (Jordan & Jacobs, 1992) is done in a straightforward and insightful manner.

Chapter 13: *Learning algorithms for temporal regularity detection.* Chapter 13 discusses the problem of processing dynamic incoming information using finite impulse response (FIR) models of connection weights. Thus, a generic FIR artificial neuron updates its activation level using a sum of the convolution of the impulse response of each connection weight with the time-dependent input signal. A methodology for developing a discrete-time version of a FIR neuron is also developed so as to be suitable for computer simulation modelling work. FIR multilayer feedforward and recurrent backpropagation networks are then developed using the concepts discussed earlier in the chapter. Chapter 13 also discusses other important categories of backpropagation recurrent networks such as the "backpropagation through time" algorithm and the "real-time recurrent network" algorithm.

Chapter 14: *Liapunov methods for analyzing ANN dynamical systems.* Chapter 14 provides a nice introduction to methods for investigating the long-term behavior of continuous-time dynamical systems using Liapunov function methods. The concept of a Liapunov function is introduced for proving statements about the behavior of a continuous-time dynamical system in the vicinity of an equilibrium point, and then the theorem is applied to the analysis of several classical network architectures such as the Hopfield (1984) model, the Cohen–Grossberg (1983) model, and a continuous-time version of the brain-state-in-a-box model. Pineda's recurrent backpropagation learning algorithm is also discussed but its placement in this chapter

is not well motivated, and the author might consider moving the description of this algorithm to Chapter 13.

Chapter 15: *VLSI methods for implementing ANN systems.* Chapter 15 discusses VLSI (very-large-scale integrated) circuitry implementations of artificial neural networks such as the Boltzmann machine, the generic artificial neuron, and more realistic network models of portions of the nervous system.

Appendices (*especially Appendix B on stochastic approximation*). The Appendices of Haykin's book also contain notable contributions. Appendix B, in particular, provides a nice summary of a method for doing stochastic convergence analyzes of a large class of artificial neural network learning algorithm using a stochastic approximation theorem developed by Ljung (1977) and Kushner and Clark (1978). Haykin also provides some nice examples regarding how the theorem may be applied. The other appendices review pseudo-inverse matrix theory (Appendix A), statistical thermodynamics (Appendix C), and the relevance of Brownian motion models for studying the asymptotic behavior of artificial neural network learning algorithms (Appendix D).

COMPARISON OF HAYKIN'S BOOK TO RELATED TEXTBOOKS

Textbooks for psychology and neuroscience graduate students. Currently, there is an exceptionally wide variety of textbooks on ANN systems suitable for research and/or teaching in the areas of psychology and neuroscience. The textbooks by Levine (1991), McClelland and Rumelhart (1986), Quinlan (1991), and especially Anderson (1995) are well suited for graduate students in psychology and neuroscience who have relatively weak backgrounds in linear algebra and multivariate calculus. These textbooks (with the exception of Levine's book) tend to discuss a relatively limited number of network architectures at a relatively leisurely pace and make efforts to minimize the mathematical presentation. I have personally found Anderson's (1995) book to be most appropriate for undergraduate and graduate psychology students who have weak mathematics backgrounds. I would also recommend Anderson's (1995) book over Haykin's (1994) book to a student who has no knowledge of linear algebra and no knowledge of calculus.

Textbooks for mathematical psychology students. On the other hand, for graduate students in the field of mathematical psychology, Haykin's book is probably one of the best possible introductions to the wide range of artificial neural network architecture available. Haykin's book is highly recommended. The only real competitor to Haykin's book, in my opinion, is the book by Hassoun (1995), which also discusses a wide range of network architectures from an engineering perspective. Either of these books would be an excellent choice for an advanced introductory course designed to introduce mathematically sophisticated graduate students to the wide range of available ANN systems architectures.

The book by Golden (1996) differs from the books by Haykin and Hassoun in that it is designed to teach students how to do rigorous mathematical analyzes of high-dimensional nonlinear artificial neural networks. To be sure, several chapters in Haykin (Chapters 11, 14, Appendix B) are devoted to general mathematical theorems and principles applicable to analyzing and designing wide classes of artificial neural networks. However, this is not the emphasis of Haykin's (or Hassoun's) book. Both Haykin and Hassoun are trying to introduce readers (in a maximally efficient manner) to the wide range of network architectures available, while simultaneously providing "hooks" to the literature for readers interested in issues of mathematical analysis and design of ANN systems. Golden's (1996) book, however, emphasizes specific *mathematical methods* for ANN system analysis and design and analyzes the same group of ANN system architectures from the perspective of different branches of mathematics. Accordingly, Golden's (1996) book is organized not by classes of network architectures but by branches of mathematics (specifically, dynamical systems theory, optimization theory, and statistical pattern recognition); it covers *fewer* network architectures than either Haykin's or Hassoun's book and is generally *more* mathematically rigorous. Either Hassoun's or Haykin's book would be an excellent prerequisite for a course based on Golden's (1996) book concerned with the mathematical analysis and design of artificial neural networks.

GENERAL SUMMARY AND CONCLUSIONS

Overall, Haykin's book provides a good, accurate, comprehensive introduction to artificial neural networks and is highly recommended for students with some mathematics background. The book manages to successfully touch upon many important and representative network architectures yet manages to do this in an insightful and nonsuperficial way. Considering the wide range of network architectures in the book, this makes the book a valuable and important contribution to knowledge in this area. Haykin's book provides a nice "snapshot" of the state-of-the-art in artificial neural network modelling in the mid-1990s. The author has a tendency to avoid interpreting and integrating results using his own perspectives, but rather provides a relatively realistic picture of specific network architectures as they have been originally presented in the literature. Such an approach clearly has its advantages and disadvantages. In particular, the reader is not

"biased" by the author's perspectives but on the other hand (without such biases) the novice to the field may find Haykin's discussion to be not only comprehensive but overwhelming as well. I have had Haykin's book on my bookshelf for several years now, and I expect to find his book to be a valuable resource for many years to come.

REFERENCES

Anderson, J. A. (1995). *Practical neural modeling*. Cambridge, MA: MIT Press.

Cohen, M., & Grossberg, S. (1983). Absolute stability of global pattern formation and parallel memory storage by competitive neural networks. *IEEE Transactions on Systems on Man and Cybernetics*, **SMC-13**, 815–826.

Cover, T. M. (1965). Geometrical and statistical properties of systems of linear inequalities with applications in pattern recognition. *IEEE Transactions on Electronic Computers*, **EC-14**, 326–334.

Golden, R. M. (1996). *Mathematical methods for neural network analysis and design*. Cambridge, MA: MIT Press.

Hassoun, M. H. (1995). *Fundamentals of artificial neural networks*. Cambridge, MA: MIT Press.

Haykin, S. (1991). *Adaptive filter theory*. Englewood Cliffs, NJ: Prentice Hall.

Haykin, S. (1994). *Neural networks*: *A comprehensive foundation*. New York: Macmillan.

Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences*, *USA*, **79**, 2554–2558.

Hopfield, J. J. (1984). Neurons with graded response have collective computational properties like those of two-state neurons. *Proceedings of the National Academy of Sciences*, *USA*, **81**, 3088–3092.

Jordan, M. I., & Jacobs, R. A. (1992). Hierarchies of adaptive experts. In J. E. Moody, S. J. Hanson, & R. A. Lippmann (Eds.), *Advances in neural information processing systems* (Vol. 4, pp. 985–992). San Mateo, CA: Morgan Kaufmann.

Kohonen, T. (1982). Self-organized formation of topologically correct feature maps. *Biological Cybernetics*, **43**, 59–69.

Kushner, H. J., & Clark, D. S. (1978). *Stochastic approximation methods for constrained and unconstrained systems*. New York: Springer-Verlag.

Levine, D. S. (1991). *Introduction to neural and cognitive modelling*. Hillsdale: Erlbaum.

Linsker, R. (1986). From basic network principles to neural architecture. *Proceedings of the National Academy of Sciences*, *USA*, **83**, 7508–7512, 8390–8394, 8779–8783.

Ljung, L. (1977). Analysis of recursive stochastic algorithms. *IEEE Transactions on Automatic Control*, **22**, 551–575.

McClelland, J. L., & Rumelhart, D. E. (1986). A distributed model of human learning and memory. In *Parallel distributed processing*: *Vol. 2*. *Psychological and biological models* (pp. 170–215). Cambridge, MA: MIT Press.

Quinlan, P. (1991). *Connectionism and psychology*. Chicago: Univ. of Chicago Press.

Tikhonov, A. N. (1963). On solving incorrectly posed problems and method of regularization. *Doklady Akademii Nauk USSR*, **151**, 501–504.