NEAR EAST UNIVERSITY

Faculty of Engineering

Department of Computer Engineering

IMPLEMENTATIONS OF NEURAL NETWORKS

Graduation Project
COM- 400

Student: Ramiz Musallam Salama

Supervisor: Assoc.Prof.Dr. Adnan Khashman

Nicosia - 2002

AKNOWLE

First, I would like to thank my super
invaluable advice and for the generosi
project and through

Second I would like to express my gratitude
scholarship that made the work possible, and it is my privilege to
Senol Bektas for standing beside us and helping us.

As I would like to thank my family for giving me the chance to complete my academic study and specially I would like to thank my parents for supporting me and giving me the opportunity to achieve my goal in life.

Finally I want to thank all my friends who helped and advised me during my preparation to the graduation project.

ABSTRACT

Technology of Neural Network gives a computer system an amazing capacity to actually learn from input data. Artificial neural networks have provided solutions to problems normally requiring human observation and thought processes. Neural network simulations appear to be a recent development. However, this field was established before the advent of computers, and has survived at least one major setback and several eras. The computing world has a lot to gain front neural networks. Their ability to learn by example makes them very flexible and powerful.

The most basic components of neural networks are modelled after the structure of the brain. Some neural network structures are not closely to the brain and some does not have a biological counterpart in the brain. However, neural networks have a strong similarity to the biological brain and therefore a great deal of the terminology is borrowed from neuroscience. Given this description of neural networks and how they work, what real world applications are they suited for? Neural networks have broad applicability to real world business problems. In fact, they have already been successfully applied in many industries, science, medicine, Manufacturing, and Sports.... Etc.

Neural networks are trained by repeatedly presenting examples to the network. Each example includes both inputs and outputs. The network tries to learn each of your examples in turn, calculating its output based on the inputs you provided. So, A.N.N. can be trained to solve the most difficult problems in many applications and especially in medicine.

Artificial Neural Network has its own mark in many fields beside medicine, such as it goes in military and image analysis, business, industry and many other fields. Thus, neural network have been applied to medicine to help the physician of handling huge quantities of data to diagnose diseases, drug development and others.

TABLE OF CONTENTS

INTRODUCTION

Chapter one aimed to show the background of the neural networks and what the reasons and the benefits of using neural networks. And also how the neural networks improved to be new technology in the present and the future. Neural network is defined as massively parallel-distributed processor that has a natural propensity for storing experiential knowledge and making it available for use. It resembles the brain in two respects: (1) Knowledge is acquired by the network through a learning process, and (2) Interneuron connection strengths known as synaptic weights are used to store the knowledge.

Neural network simulations appear to be a recent development. However, this field was established before the advent of computers, and has survived at least one major setback and several eras. Many important advances have been boosted by the use of inexpensive computer emulations. Neural networks learn by example. They cannot be programmed to perform a specific task. The development of true Neural Networks is a fairly recent event, which has been met with success. The future of Neural Networks is wide open, and may lead to many answers and/or questions.

Chapter two describes the architectures and basic components of neural networks; the most basic components of neural networks are modelled after the structure of the brain. Some neural network structures are not closely to the brain and some does not have a biological counterpart in the brain. However, neural networks have a strong similarity to the biological brain and therefore a great deal of the terminology is borrowed from neuroscience. Neural networks are named after the cells in the human brain that perform intelligent operations. The brain is made up of billions of neuron cells. Each of these cells is like a tiny computer with extremely limited capabilities; however, connected together, these cells form the most intelligent system known. Neural networks are formed from hundreds or thousands of simulated neurons connected together in much the same way as the brains neurons.

The term 'architecture' has been much abused in the history of mankind. It has many meanings depending on whether you are talking about buildings, inside of computers or neural networks among other things. Even in neural networks, the term architecture and what we have been referring to as 'type' of neural network are used interchangeably. So when we refer to such and such architecture, it means the set of possible interconnections (also called as topology of the network) and the learning algorithm defined for it. Also Learning algorithms, which were considered for a single perception, linear Adeline, and multiplayer perception, belong to the class of supervised learning algorithms. Two basic groups of unsupervised learning algorithms and related self-organizing neural networks, namely: Hebbian Learning, Competitive Learning. Networks such as the one just described are called artificial neural networks (ANNs), in the sense that they represent simplified models of natural nerve or neural networks.

Chapter three shows the fields where the Neural Networks can be applied, Neural Networks are performing successfully where other methods do not, recognizing and matching complicated, vague, or incomplete patterns Neural networks have been applied in solving a wide variety of problems. The most common use for neural networks is to project what will most likely happen. There are many applications of Neural Networks that can be applied in the real world. Although one may apply neural network systems for prediction, diagnosis, planning, monitoring, debugging, repair, instruction, and control, the most successful applications of neural networks are in categorization and pattern recognition.

A number of real applications can also be found in the Neuro Forecaster package. Based on these successful applications, it is therefore evident that the neural network technology can be applied to many real-world problems especially those related to -business, financial and engineering modelling.

Chapter four will be specialized in the medical applications and describes some fields where we can find the N.N. in medicine, whether in medical diagnostic aides, biochemical analysis or in medical image analysis.

CHAPTER ONE

BACKGROUND OF NEURAL NETWORK

1.1 Overview

Neural networks have a large appeal to many researchers due to their great closeness to the structure of the brain, a characteristic not shared by more traditional systems. In an analogy to the brain, an entity made up of interconnected neurons, neural networks are made up of interconnected processing elements called units, which respond in parallel to a set of input signals given to each. The unit is the equivalent of its brain counterpart, the neuron. A neural network consists of four main parts:

- Processing units, where each unit has certain activation level at any point in time.
- Weighted interconnections between the various processing units, which determine how the activation of one unit leads to input for another unit.
- An activation rule, which acts on the set of, input signals at a unit to produce anew output signal, or activation.
- Optionally, a Learning rule that specifies how to adjust the weights for a given input/output pairs.

1.2 What is a Neural Network?

An Artificial Neural Network (ANN) is an information-processing paradigm that is inspired by the way biological nervous systems, such as the brain, process information. The key element of this paradigm is the novel structure of the information processing system. It is composed of a large number of highly interconnected processing elements (neurones) working in unison to solve specific problems. ANNs, like people, learn by example. An ANN is configured for a specific application, such as pattern recognition or data classification, through a learning process. Learning in biological systems involves adjustments to the synaptic connections that exist between the neurones. This is true of ANNs as well.

1.3 Historical background

Neural network simulations appear to be a recent development. However, this field was established before the advent of computers, and has survived at least one major setback and several eras.

Many importand advances have been boosted by the use of inexpensive computer emulations. Following an initial period of enthusiasm, the field survived a period of frustration and disrepute. During this period when funding and professional support was minimal, important advances were made by relatively few researchers. These pioneers were able to develop convincing technology, which surpassed the limitations identified by Minsky and Papert. Minsky and Papert, published a book (in 1969) in which they summed up a general feeling of frustration (against neural networks) among researchers, and was thus accepted by most without further analysis. Currently, the neural network field enjoys a resurgence of interest and a corresponding increase in funding [1].

1.4 Historical background in detail

The history of neural networks that was described above can be divided into several periods:

First Attempts: There were some initial simulations using formal logic. McCulloch and Pitts (1943) developed models of neural networks based on their understanding of neurology. These models made several assumptions about how neurons worked [2]. Their networks were based on simple neurons, which were considered to be binary devices with fixed thresholds. The results of their model were simple logic functions such as "a or b" and "a and b". Another attempt was by using computer simulations. Two groups (Farley and Clark, 1954; Rochester, Holland, Haibit and Duda, 1956). The first group (IBM researchers) maintained closed contact with neuroscientists at McGill University. So whenever their models did not work, they consulted the neuroscientists. This interaction established a multidiscilinary trend, which continues to the present day

Promising & Emerging Technology: Not only was neroscience influential in the development of neural networks, but psychologists and engineers also contributed to the progress of neural network simulations. Rosenblatt (1958) stirred considerable interest and activity in the field when he designed and developed the Perceptron. The Perceptron had three layers with the middle layer known as the association layer[3].

This system could learn to connect or associate a given input to a random output unit. Another system was the ADALINE (ADAptive LInear Element), which did Widrow and Hoff (of Stanford University) develop in 1960. The ADALINE was an analogue electronic device made from simple components. The method used for learning was different to that of the Perceptron, it employed the Least-Mean-Squares (LMS) learning rule[4].

Period of Frustration & Disrepute: In 1969 Minsky and Papert wrote a book in which they generalized the limitations of single layer Perceptrons to multilayered systems. In the book they said: "...our intuitive judgment that the extension (to multilayer systems) is sterile". The significant result of their book was to eliminate funding for research with neural network simulations. The conclusions supported the disenchantment of researchers in the field. As a result, considerable prejudice against this field was activated.

Innovation: Although public interest and available funding were minimal, several researchers continued working to develop neuromorphically based computational methods for problems such as pattern recognition. During this period several paradigms were generated which modern work continues to enhance.Grossberg's (Steve Grossberg and Gail Carpenter in 1988) influence founded a school of thought which explores resonating algorithms. They developed the ART (Adaptive Resonance Theory) networks based on biologically plausible models. Anderson and Kohonen developed associative techniques independent of each other. Klopf (A. Henry Klopf) in 1972 developed a basis for learning in artificial neurons based on a biological principle for neuronal learning called heterostasis. Werbos (Paul Werbos 1974) developed and used the back-propagation learning method, however several years passed before this approach was popularized. Back-propagation nets are probably the best known and widely applied of the neural networks today. In essence, the back-propagation nets [5]. Is a Perceptron with multiple layers, a different thershold function in the artificial neuron, and a more robust and capable learning rule. Amari (A. Shun-Ichi 1967) was involved with theoretical developments: he published a paper, which established a mathematical theory for a learning basis (error-correction method) dealing with adaptive patern classification. While Fukushima (F. Kunihiko) developed a step wise trained multilayered neural network for interpretation of handwritten characters. The original network was published in 1975 and was called the Cognitron.

Re-Emergence: Progress during the late 1970s and early 1980s was important to the re-emergence on interest in the neural network field[6]. Several factors influenced this movement. For example, comprehensive books and conferences provided a forum for people in diverse fields with specialized technical languages, and the response to conferences and publications was quite positive. The news media picked up on the increased activity and tutorials helped disseminate the technology. Academic programs appeared and courses were inroduced at most major Universities (in US and Europe). Attention is now focused on funding levels throughout Europe, Japan and the US and as this funding becomes available, several knew commercial with applications in industry and financial institutions are emerging.

Today: Significant progress has been made in the field of neural networks-enough to attract a great deal of attention and fund further research. Advancement beyond current commercial applications appears to be possible, and research is advancing the field on many fronts. Neurally based chips are emerging and applications to complex problems developing. Clearly, today is a period of transition for neural network technology.

1.5 Why use neural networks?

Either humans or other computer techniques can use neural networks, with their remarkable ability to derive meaning from complicated or imprecise data, to extract patterns and detect trends that are too complex to be noticed. A trained neural network can be thought of as an "expert" in the category of information it has been given to analyse. This expert can then be used to provide projections given new situations of interest and answer "what if" questions.
Other advantages include:

Adaptive learning: An ability to learn how to do tasks based on the data given for training or initial experience.

Self-Organization: An ANN can create its own organization or representation of the information it receives during learning time.

Real Time Operation: ANN computations may be carried out in parallel, and special hardware devices are being designed and manufactured which take advantage of this capability.

Fault Tolerance via Redundant Information Coding: Partial destruction of a network leads to the corresponding degradation of performance. However, some network capabilities may be retained even with major network damage.

1.6 Neural networks versus conventional computers

Neural networks take a different approach to problem solving than that of conventional computers. Conventional computers use an algorithmic approach i.e. the computer follows a set of instructions in order to solve a problem. Unless the specific steps that the computer needs to follow are known the computer cannot solve the problem. That restricts the problem solving capability of conventional computers to problems that we already understand and know how to solve. But computers would be so much more useful if they could do things that we don't exactly know how to do.

Neural networks process information in a similar way the human brain does. The network is composed of a large number of highly interconnected processing elements(neurones) working in parallel to solve a specific problem. Neural networks learn by example. They cannot be programmed to perform a specific task. The examples must be selected carefully otherwise useful time is wasted or even worse the network might be functioning incorrectly. The disadvantage is that because the network finds out how to solve the problem by itself, its operation can be unpredictable.

On the other hand, conventional computers use a cognitive approach to problem solving; the way the problem is to solve must be known and stated in small unambiguous instructions. These instructions are then converted to a high-level language program and then into machine code that the computer can understand. These machines are totally predictable; if anything goes wrong is due to a software or hardware fault.

Neural networks and conventional algorithmic computers are not in competition but complement each other. There are tasks are more suited to an algorithmic approach like arithmetic operations and tasks that are more suited to neural networks. Even more, a large number of tasks require systems that use a combination of the two approaches (normally a conventional computer is used to supervise the neural network) in order to perform at maximum efficiency.

1.7 Where are Neural Networks being used?

Neural networks are performing successfully where other methods do not, recognizing and matching complicated, vague, or incomplete patterns. Neural networks have been applied in solving a wide variety of problems.

The most common use for neural networks is to project what will most likely happen. There are many areas where prediction can help in setting priorities. For example, the emergency room at a hospital can be a hectic place, to know who needs the most critical help can enable a more successful operation. Basically, all organizations must establish priorities, which govern the allocation of their resources. Neural networks have been used as a mechanism of knowledge acquisition for expert system in stock market forecasting with astonishingly accurate results. Neural networks have also been used for bankruptcy prediction for credit card institutions.

Although one may apply neural network systems for interpretation, prediction, diagnosis, planing, monitoring, debugging, repair, instruction, and control, the most successful applications of neural networks are in categorization and pattern recognition. Such a system classifies the object under investigation (e.g. an illness, a pattern, a picture, a chemical compound, a word, the financial profile of a customer) as one of numerous possible categories that, in return, may trigger the recommendation of an action (such as a treatment plan or a financial plan.

A company called Nestor, have used neural network for financial risk assessment for mortgage insurance decisions, categorizing the risk of loans as good or bad. Neural networks has also been applied to convert text to speech, NETtalk is one of the systems developed for this purpose. Image processing and pattern recognition form an important area of neural networks, probably one of the most actively research areas of neural networks.

An other of research for application of neural networks is character recognition and handwriting recognition. This area has use in banking, credit card processing and other financial services, where reading and correctly recognizing handwriting on documents is of crucial significance. The pattern recognition capability of neural networks has been used to read handwriting in processing checks, the amount must normally be entered into the system by a human. A system that could automate this task would expedite check processing and reduce errors. One such system has been developed by HNC (Hecht-Nielsen Co.) for BankTec.

One of the best-known applications is the bomb detector installed in some U.S. airports. This device called SNOOPE, determine the presence of certain compounds from the chemical configurations of their components.

In a document from International Joint conference, one can find reports on using neural networks in areas ranging from robotics, speech, signal processing, vision, character recognition to musical composition, detection of heart malfunction and epilepsy, fish detection and classification, optimization, and scheduling. One may take under consideration that most of the reported applications are still in research stage.

1.8 Where are Neural Networks applicable?

Neural networks cannot do anything that cannot be done using traditional computing techniques, **BUT** they can do some things, which would otherwise be very difficult In particular, they can form a model from their training data (or possibly input data) alone.

This is particularly useful with sensory data, or with data from a complex (e.g. chemical, manufacturing, or commercial) process. There may be an algorithm, but it is not known, or has too many variables. It is easier to let the network learn from examples.

Neural networks are being used:

In investment analysis:

To attempt to predict the movement of stocks currencies etc., from previous data. There, they are replacing earlier simpler linear models.

In signature analysis:

As a mechanism for comparing signatures made (e.g. in a bank) with those stored. This is one of the first large-scale applications of neural networks in the USA, and is also one of the first to use a neural network chip.

In process control:

There are clearly applications to be made here: most processes cannot be determined as computable algorithms. Newcastle University Chemical Engineering Department is working with industrial partners (such as Zeneca and BP) in this area.

In monitoring:

Networks have been used to monitor

The state of aircraft engines. By monitoring vibration levels and sound, early warning of engine problems can be given.

British Rail have also been testing a similar application monitoring diesel engines.

In marketing:

Networks have been used to improve marketing mailshots. One technique is to run a test mailshot, and look at the pattern of returns from this. The idea is to find a predictive mapping from the data known about the clients to how they have responded. This mapping is then used to direct further mailshots.

1.9 Where are neural networks going?

A great deal of research is going on in neural networks worldwide. This ranges from basic research into new and more efficient learning algorithms, to networks, which can respond to temporally varying patterns (both ongoing at Stirling), to techniques for implementing neural networks directly in silicon. Already one chip commercially available exists, but it does not include adaptation. Edinburgh University have implemented a neural network chip, and are working on the learning problem.

Production of a learning chip would allow the application of this technology to a whole range of problems where the price of a PC and software cannot be justified.

There is particular interest in sensory and sensing applications: nets, which learn to interpret real-world sensors and learn about their environment.

1.10 Examples of Real-life Applications

Neural networks can be used in virtually any situation where the objective is to determine an unknown variable or attribute from known observations or registered measurements (i.e., various forms of regression, classification, and time series), where there is a sufficient amount of historical data, and where there actually exists a tractable underlying relationship or a set of relationships (networks are relatively noise tolerant). In addition, neural networks can be used for exploratory analysis by looking for data clustering (Kohonen networks).

A comprehensive discussion of theoretical considerations related to the issue of when neural network applications are most likely to be successful can be found in the chapter on neural networks in the StatSoft Electronic Statistics Textbook (available on the StatSoft web site). The following list includes a selection of representative examples that by no means exhaust all areas where neural networks can be used.

Optical Character Recognition, including Signature Recognition (e.g., a company has developed a device which identifies signatures, using not just appearance but also pen-velocity while signing, which makes it more difficult to perpetrate fraud).

Image Processing (e.g., a system was developed which scanned images of London subway stations, and could tell if the station was Full, Empty, Half-Full etc. and was invariant across light conditions and presence/absence of trains).

Financial Time Series Prediction (e.g., LBS Capital Management claims to have significantly improved trading performance using Multilayer Perceptrons to predict stock prices).

Credit Worthiness (a classic problem - decide whether someone is a good credit risk, based on questionnaire information).

Bulk mail targeting (i.e., identify customers who are more likely to respond positively to a mail-out, based on database information).

Detection and evaluation of medical phenomena (e.g., detection of epileptic attacks, estimation of prostate tumor size).

Condition monitoring of machinery (e.g., detecting when something has gone wrong with a machine based on vibration or acoustic signatures, so that preventative maintenance can be scheduled).

Speech synthesis from text (e.g., the famous early experiment was *Nettalk*, which learned to produce phonemes from written text).

Chaotic Time Series Prediction (a number of researchers have demonstrated good prediction capability on chaotic time series data).

Process control (e.g., monitoring industrial process machinery and continuously adjusting control parameters).

Engine management systems (estimating fuel consumption from sensor measurements and adjusting - a form of process control).

Language analysis (e.g., using unsupervised techniques to identify key phrases, words, etc. in native South American languages).

1.11 What Can Neural Networks Be Used For?

Neural networks constitute a powerful tool for data mining. Data mining has become quite popular recently and really involves the extraction of knowledge from information. Organizations have more and more data from which they need to extract key trends in order to run their businesses more efficiently and improve decision-making.

Applications of neural networks are numerous. Many receive their first introduction by reading about the applications of the techniques in financial market predictions. Several well-known investment groups make claims that at least some of their technical analysis of financial markets and portfolio selection is performed with neural networks.

Other successful applications of the techniques include: analysis of market research data and customer satisfaction, industrial process control, forecasting applications, and credit card fraud identification. Mellon Bank installed a neural network credit card fraud detection system and the realized savings were expected to pay for the new system in six months. A number of other banks are also using neural network - based systems to control credit card fraud. These systems are able to recognize fraudulent use based on past charge patterns with greater accuracy than other available methods.

Another example of using neural networks to improve decisions is in medical diagnosis. A neural network can be shown a series of case histories of patients, with a number of patient characteristics, symptoms, and test results. The network is also given the diagnosis for the case from the attending physician. The network can then be shown information regarding new patients and the network will provide a diagnosis for the new cases. This essentially creates a system containing the expertise of numerous physicians, which can be called upon to give an immediate, real-time initial diagnosis of a case to medical personnel [7].

1.12 Should I consider Neural Networks?

When approached with a proposal to apply a neural network, how should a business manager evaluate the proposal? Does this new capability offer real benefits, or is this the latest example of trendy approaches and buzzwords? Most importantly, are these techniques practical or are they academic approaches that are not practical or cost effective?

Given a steady increase in successful applications, neural networks are for real and offer substantial benefits. The technical details of neural networks are beyond the scope of this article, but successful applications share certain common characteristics that may be easily understood. First, there will exist interrelationships between the explanatory factors that are used to estimate the factor we don't know -- the outcome. Having interrelationships in the data means that two or more factors work together to predict model outcome. For example, a chemical process in a production facility may be dependent on temperature and humidity. These two factors combine to affect the outcome of the process. The second condition in which neural networks excel is when there is a non-linear relationship between the explanatory factors and the outcome. This simply means that the nature of the relationship between the factors and the outcome changes as the factors take on different values, which is the norm for everyday problems.

In regards to the trendiness issue, yes, neural networks are presently trendy -- at least in some circles. However, the need to improve processes by doing things better and cheaper is more important than ever in today's competitive business climate. Likewise, the desire to develop computer systems that can learn by themselves and improve decision-making is an ongoing goal of information technology. The neural network techniques we use today may not remain with us. However, the goal of developing computers that learn from past experience and lead to better business decisions will remain a high priority. Neural networks now represent one of the best practices in achieving this goal. Furthermore, continued achievements toward this goal are likely to be inspired or generated from these technologies.

The answer to the question of whether these approaches are practical and cost effective is a definitive "yes", although finding documented proof of this can be a challenge. It is true that the techniques are relatively new and that experience with these techniques is not as extensive as with traditional techniques. A great deal has been published about the technical approaches, the mathematics, and the learning rules. However, little has been written about the practical application of neural networks. It would be highly unlikely for you to find a source describing the application of neural networks to your specific problem. However, there is not a dearth of successful applications. Look at it this way - how likely would it be for you to share specifics of key information learned about your markets or business with your competitors?

The fact remains, however, that neural networks are proving their worth everyday in a wide variety of business applications, and saving their users time and money in the process.

1.13 When to Consider a Neural Network

Neural networks should be applied in situations where traditional techniques have failed to give satisfactory results, or where a small improvement in modeling performance can make a significant difference in operational efficiency or in bottom-line profits. Direct marketing is an excellent example of where a small improvement can lead to significant results. The response rate on direct marketing campaigns is usually quite low. A five percent response rate is often considered very good. By reviewing the demographic data on those that respond it may be possible to identify characteristics that would produce a 6% response rate. If a neural network is used to analyze the demographic characteristics and a 7% response rate is produced, then the cost of the direct mail campaign can be reduced while maintaining the same desired level of positive response from prospects.

An individual wanting to investigate this emerging technology and explore ways in which it can improve his/her organization is advised to consult with neural network practitioners who have experience in developing and implementing models for use in commercial applications. Z Solutions will be glad to discuss this with you.

The bottom line is that any manager interested in getting more useful information from available data should consider neural network technology as an option. They can be used by aggressive organizations to focus available resources more effectively, thus gaining a valuable competitive edge.

1.14 What are the advantages of Neural Networks?

- Neural networks can be retrained using additional input variables.
- Once trained, they are very fast.
- Due to increased accuracy, results in cost saving.
- They deal with the non-linearity in the world in which we live.
- They handle noisy or missing data.
- They create their own relationships amongst information – no equation!
- They provide general solutions with good predictive accuracy.

1.15 What are the Disadvantages of the Neural Networks?

- No set rules for network selection.
- Needs expertise in training the network.

1.16 Summary

A brief background about the A.N.N. since aneuronphysiologist and a mathematician that was in the 1943 and may be before and passing through the years 1950's , 1960's... till present days and A.N.N is in an extremely enormous development.

A Neural Network depends on its action to the imitation of the brain so, the biologist trying to simulate the action and the reaction of the brain, just as the human learns from example here also the N.N learns by example.

A neural network can be used in many fields specifically in these where traditional techniques cannot be applied, it can also handle the abnormal tasks where it has to be fed by too much data such as the radar system and the sonar system in the submarine under the water.

With all these developments in neural networks but it are also can not match the human kind's brain in pattern recognition or even an animal brain when it faces a similar problem.

CHAPTER TWO

ARCHITECTURES OF NEURAL NETWWORKS

2.1 Overview

Your network consists of many neurons, grouped into layers. The first layer is the input neurons, which take in your input data. The hidden neurons learn how the inputs combine to produce the desired results; these neurons do the real work of the network. The output neurons translate the network results for you.

The connections, which can be thought of as lines between the layers, are what get corrected during training. BrainMaker strengthens some connections and weakens others, so that the next time example data is presented the neural network will output a more correct answer

2.2 Network Architecture

The manner in which the neurons of a neural network are structured is intimately linked with the algorithm used to train network.

In general we may identify four different classes of network architectures:

2.2.1 Single-Layer Feedforward Networks

A layered neural network is a neurons organized in the form of layers. In the simplest form of a layered network, we just have an input layer of source nodes that project onto an output layer of neurons (computation nods), but not vice versa.

In other words, this network is strictly of a feedforward type. It is illustrated in fig. 2.1. For the case of four nodes in both the input layers. Such a network is called a single-layer network, with the designation "single layer" referring to the output layer of source nodes, because no computation is performed there. A linear associative memory is an example of a single-layer neural network.

In such an application, the network associates an output pattern (vector) with an input pattern (vector), and information is stored in the network by virtue of modification made to the synaptic weights of the network.

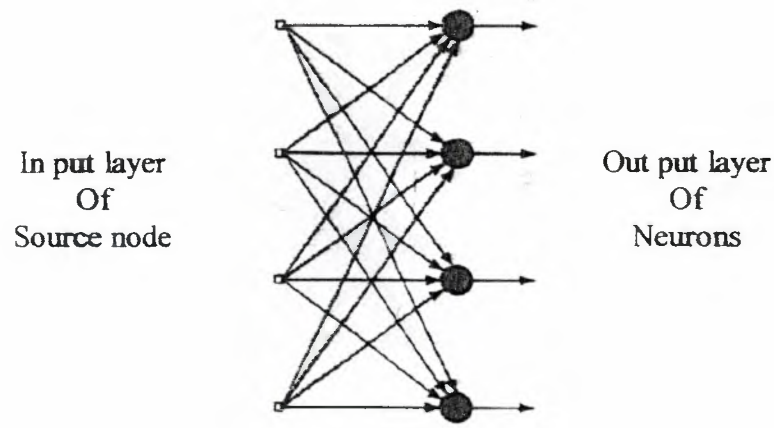

Figure 2.1. Feedforward network with a single layer of neurons.

2.2.2 Multilayer Feedforward Networks

The second class of a feedward neural network distinguishes itself by the presence of one or more hidden layers, whose computation nodes are correspondingly called hidden neurons or hidden units. The function of the hidden is to intervene between the external input and the network output. By adding one more hidden layer, the network is enabled to extract higher-order statistics, for (in a rather loose sense) the network acquires a global perspective despite is local connectivity by virtue of the extra set of synaptic connections and the extra dimension of neural interaction. The ability of hidden neurons to extract higher-order statistics is particularly valuable when the size of the input layer is large.

The source node in the input layer of the network supply respective elements of the activation pattern (input vector), which constitute the input signals applied to the neurons (computation nodes) in the second layer (i.e., the first hidden layer). The output signals of the second layer are used as input to the third layer, and so on for the rest of the network. Typically, the neurons in each layer of the network have as their inputs the output signals of the preceding layer only.

The neural network of Fig. 2.2. is said to be fully connected in the sense that every node in each layer of the network is connected to every other node in the adjacent forward layer. If, however, some of the communication links (synaptic connections) are missing from the network, we say that the network is partially connected. A form of partially connected multilayer feedforward network of particular interest is a locally connected network.

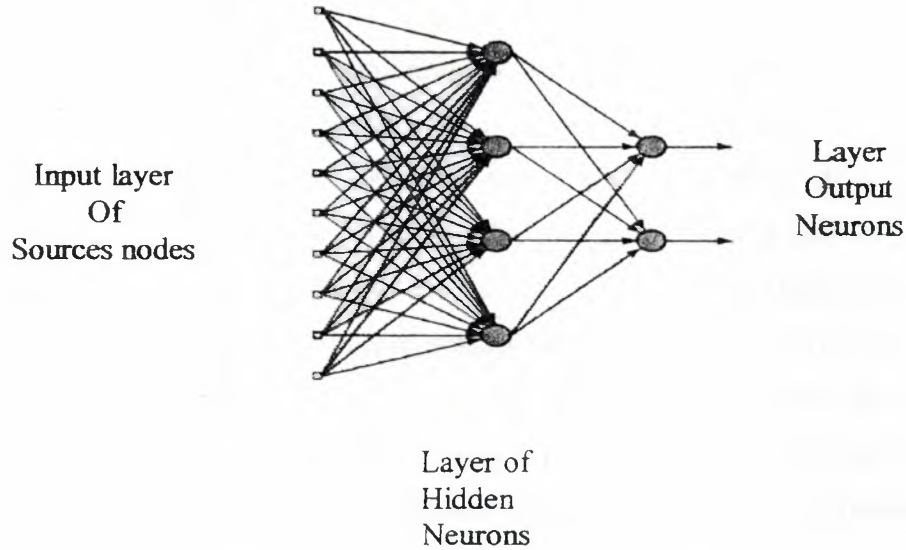

Figure 2.2. One hidden layer and output

2.2.3 Recurrent Networks

A recurrent neural network distinguishes itself from a feedforward neural network in that it has at least one feedback loop. For example, a recurrent network may consist of a single layer of neurons with each neuron feeding its output signal back to the inputs of all the other neurons, as illustrated in the architectural graph of Fig 2.3.

In the structure depicted in this figure there are no self-feedback loops in the network; self-feedback refers to a situation where the output of a neuron is fed back to its own input. The recurrent network illustrated in fig. 2.3 also has no hidden neurons.

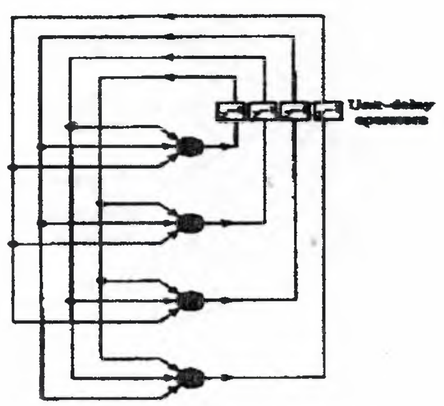

Figure 2.3. Recurrent network with no self-feedback loops and no hidden neurons

2.2.4 Lattice Structures

A lattice consists of a one-dimensional, two-dimensional, or higher-dimensional array of neurons with a corresponding set of source nodes that supply the input signals to the array; the dimension of the lattice refers to the number of dimensions of the space in which the graph lies. The architectural graph of Fig. 2.4 depicts a one-dimensional lattice of 3 neurons fed from a layer of 3 source nodes. The hidden layer learns to recode (or to provide a representation for) the inputs. More than one hidden layer can be used. The architecture is more powerful than single-layer networks: it can be show that any mapping can be learned, given two hidden layers (of units). The units are a little more complex than those in the original perception: their input/output graph is

As a function:

Y = 1 / (1+ exp (-k. (&sum W_{in} * X_{in}))

The graph shows the output for *k=0.5,1, and 10,* as the activation varies from-10 to 10

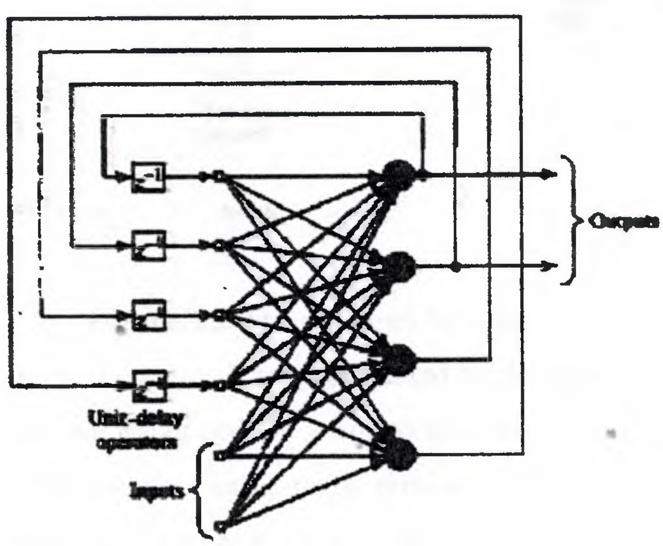

Figure 2.4. **Input layer of source nodes**

2.2.5 Typical Radial Basis Function Architecture:

Like BP, RBF nets can learn arbitrary mappings: the primary difference is in the hidden layer.

RBF hidden layer units have a receptive field, which has a *center:* that is, a particular input value at which they have a maximal output. Their output tails off as the input moves away from this point.

2.3 The Artificial Neuron

The basic unit of neural networks, the artificial neurons, simulates the four basic functions of natural neurons. Artificial neurons are much simpler than the biological neuron; the figure below shows the basics of an artificial neuron.

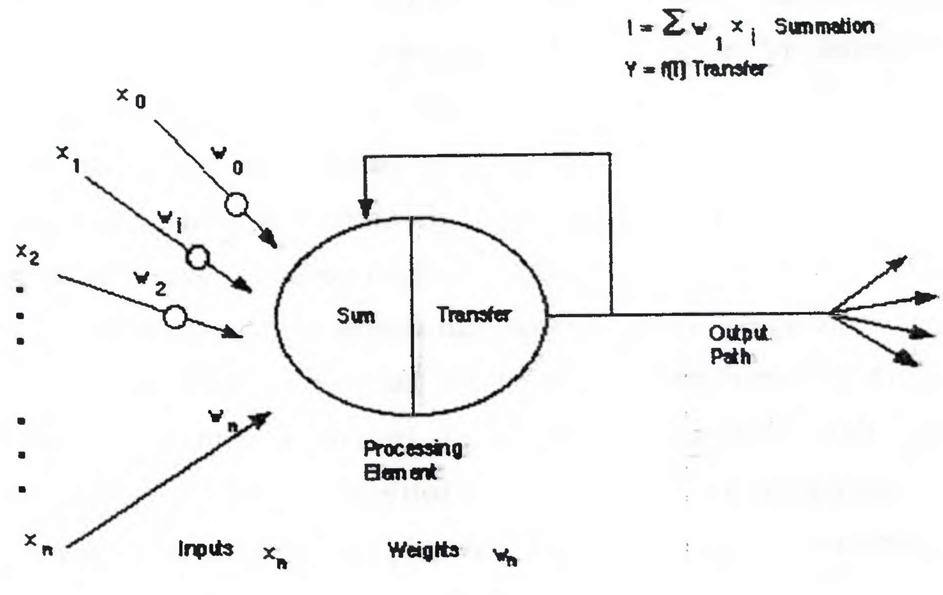

Figure 2.5. The Artificial Neuron

Note that various inputs to the network are represented by the mathematical symbol, x(n). Each of these inputs are multiplied by a connection weight, these weights are represented by w(n). In the simplest case, these products are simply summed, fed through a transfer function to generate a result, and then output. Even though all artificial neural networks are constructed from this basic building block the fundamentals may vary in these building blocks and there are differences.

2.4 Building A Neural Network

Since 1958, when psychologist Frank Rosenblatt proposed the "Perceptron," a pattern recognition device with learning capabilities, the hierarchical neural network has been the most widely studied form of network structure [8]. A hierarchical neural network is one that links multiple neurons together hierarchically. The special characteristic of this type of network is its simple dynamics. That is, when a signal is input into the input layer, it is propagated to the next layer by the interconnections between the neurons. Simple processing is performed on this signal by the neurons of the receiving layer prior to its being propagated on to the next layer. This process is repeated until the signal reaches the output layer completing the processing process for that signal. The manner in which the various neurons in the intermediary (hidden) layers process the input signal will determine the kind of output signal it becomes (how it is transformed). As you can see, then, hierarchical network dynamics are determined by the weight and threshold parameters of each of their units. If input signals can be transformed to the proper output signals by adjusting these values (parameters), then hierarchical networks can be used effectively to perform information processing. Since it is difficult to accurately determine multiple parameter values, a learning method is employed. This involves creating a network that randomly determines parameter values. This network is then used to carry out input-to-output transformations for actual problems. The correct final parameters are obtained by properly modifying the parameters in accordance with the errors that the network makes in the process. Quite a few such learning methods have been proposed. Probably the most representative of these is the error back-propagation learning method proposed by D. E. Rumelhart et al. in 1986. This learning method has played a major role in the recent Neurocomputing boom [9].

The back-propagation paradigm has been tested in numerous applications including bond rating, mortgage application evaluation, protein structure determination, backgammon playing, and handwritten digit recognition. Choosing the right methodology, or backpropagation algorithm, is another important consideration. In working with the financial applications, many have found that the back-propagation algorithm can be very slow. Without using advanced learning techniques to speed the process up, it is hard to effectively apply backpropagation to real-world problems. Overfitting of a neural network model is another area, which can cause beginners difficulty. Overfitting happens when an ANN model is trained on one set of data, and it learns that data too well. This may cause the model to have poor generalization abilities - the model may instead give quite poor results for other sets of data. For an in-depth coverage of other neural network models and their learning algorithms, please refer to the Technical Reading at the end of this User's Guide, the Technical Reference (sold separately), those papers listed in the Reference, or any other reference books on neural networks and relevant technology.

2.5 Biological Neural Networks

Models of our own brains, and nerve cell motivate neural networks architectures. Although the knowledge of the brain is limited, we do have much detailed anatomical and physiological information. The basic anatomy of an individual nerve cell (also known as the neuron) is known, and the most important biochemical reactions that govern its activities have been identified.

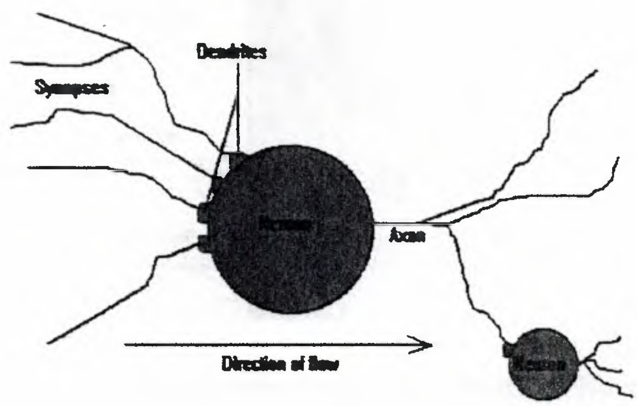

Figure 2.6. A simple neuron cell

The biological brain is an incredibly complex system of more than a 100 billion neurons of different types (not all) highly interconnected with each other via synapses of which there are more than a 150 billion. There is a set of synapses coming into each neuron which communicate with it through it's private dendrites, and each neuron also have an axon out of which it delivers it's messages to other neurons. It is also known that the human brain performs an average of 100 operations per second. Action potentials are fired from each neuron to others (depending on the task the brain is performing), which are electric pulses whose intensity level varies.

In comparison Artificial Neural Nets (ANN), like their biological equivalents, consist of processing elements called *n*eurons or units, and connections between them called synapses or just connections. Associated with each connection is a weight that simulates the synaptic gap. Instead of pulse trains, most ANNs use analogue values as a means of communication. Floating point numbers in software simulations usually represents these.

There are three types of neuron, input, output, and hidden. Input, and output neurons form the nodes at which data enters or leaves the network; hidden neurons, as their name implies, are internal to the network [10].

A modern ANN, then, consists of these elements joined by connections, and is normally represented by fig.2.7 similar to that shown below.

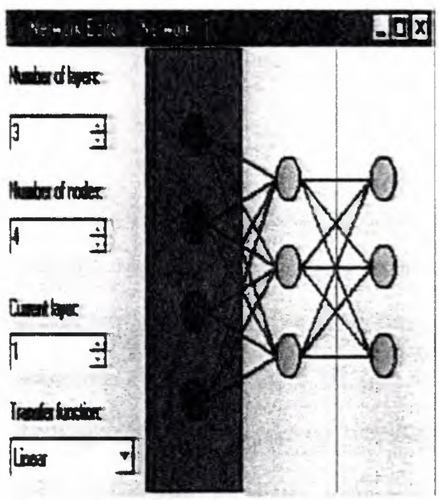

Figure 2.7. A diagrammatic representation of Artificial Neural Network

2.6 Neural Network Learning

Among the many interesting properties of a neural network, the property that is of primary signification is the ability of the network to learn from its environment, and to improve its performance through learning; the improvement in performance takes place over time in accordance with some prescribed measure. A neural network learns about its environment through an iterative process of adjustments applied to its synaptic weights and thresholds. Ideally, the network becomes more knowledgeable about its environment after each iteration of the learning refers to computer models that improve their performance in signification ways based upon data.

The results to this date are revealing robust and elegant solutions. Artificial neural systems are unlike artificial intelligence programs. Artificial intelligence programs use deductive reasoning to apply known rules to situations to produce outputs. Each new situation may require that another rule be implemented. The programs can become quite large and complicated in an attempt to address all possible situations. Artificial neural systems, however, automatically construct associations based upon the results of known situations. For each new situation, the neural system automatically adjusts itself and eventually generalizes it. Artificial neural networks are computer programs that simulate biological neural network [11]. In order to process vague, noisy, or incomplete information, researchers are turning to biological neural systems as a model for a new computing paradigm. Biological neural systems process this type of information seemingly effortlessly. Neuroscientists have learned much about biological neural systems in recent decades and engineers are using this information to create artificial neural systems in the laboratory.

2.6.1 Supervised Learning

This is usually performed with feedforward nets where training patterns are composed of two parts, an input vector and output vector, associated with the input and output nodes respectively. A training cycle consists of the following steps. An input vector is presented at the inputs together with a set of desired responses, one for each node, at the output layer. A forward bass is done and the errors or discrepancies, between the desired and actual response for each node in the output layer, are then used to determine weight changes in the net according to the prevailing rule. The term 'supervised' originates from the fact that the desired signals on individual output nodes are provided by an external 'teacher'. The best-known examples of this technique occur in the backpropagation algorithm, the delta rule and preceptron. See Figure2.8.

2.6.1.1 Supervised Learning divided into two parts:

1- Feedback nets: -

A-Back propagation through time

B-Real time recurrent learning

C-Recurrent extended kalman filter

2- Feed forward-only net: -

A-Perceptron

B-Adeline

C-Time delay neural network

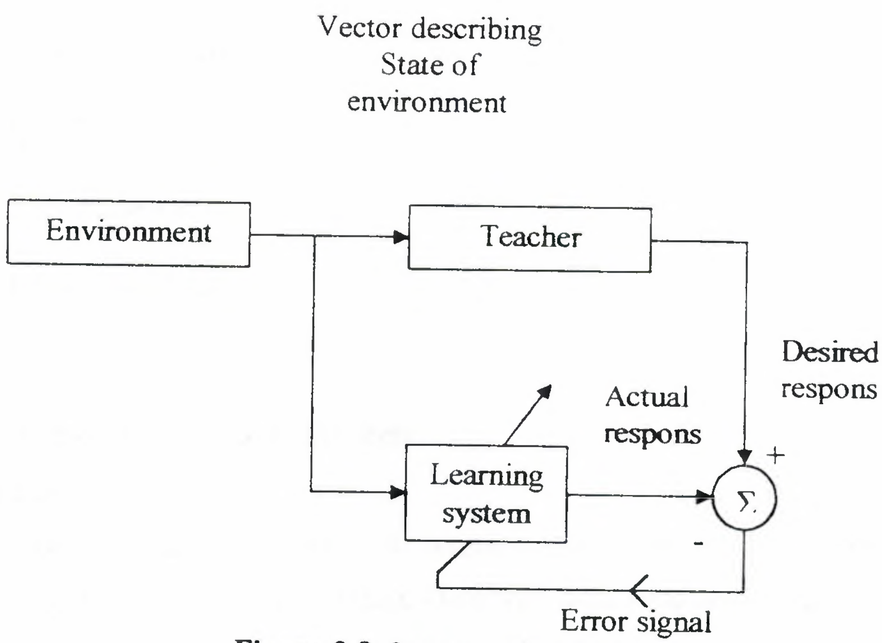

Figure 2.8. Supervised Learning

2.6.2 Unsupervised learning

This is usually found in the context of recurrent and competitive nets. There is no separation of the training set into input and output pairs. Typically a training cycle will consist of the following steps: a vector is applied to the visible nodes (or in the case of competitive learning, the input nodes); the net is allowed to reach equilibrium; weight changes are made according to some prescription. It is the amalgamation of input-output pairs, and hence the disappearance of the external supervisor providing target outputs, that gives this scheme its name. This kind of learning is sometimes referred to as self-organization. See Fig. 2.9

2.6.2.1 Unsupervised divided into two parts:

1-Feedback nets:

A-Discrete hop filed

B-Analog adaptive resonance theory

C-Additive gross berg

2 -Feed forward –only nets

A-Learning matrix

B-Linear associative memory

C-Counter propagation

2.6.3 Applications for unsupervised nets

Clustering data:

Exactly one of a small number of output units comes on in response to an input.Reducing the dimensionality of data: Data with high dimension (a large number of input units) is compressed into a lower dimension (small number of output units).Although learning in these nets can be slow, running the trained net is very fast - even on a computer simulation of a neural net.

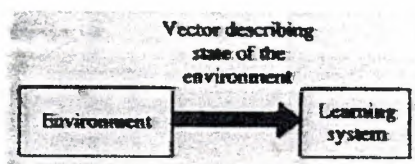

Figure 2.9. Unsupervised learning

2.6.4 Hebbian learning

Hebb's postulate of learning is the oldest and most famous of all learning rules; it is named in honor of the neuropsychologist Hebb (1949) [12]. Quoting from Hebb's book, the organization of behavior (1949,p.62):

When an axon of cell A is near enough to a cell B and repeatedly or persistently takes part in firing it, some growth process or metabolic changes take place in one or both cells such that A's efficiency as one of the cells firing B is increased.

Hebb proposed this change as a basis of associative learning (at the cellular level), which would result in an enduring modification in the activity pattern of a spatially distributed "assemble of never cells."

2.7 Perceptrons.

The most influential work on neural nets in the 60's went under the heading of 'perceptrons' a term coined by Frank Rosenblatt . The perceptron (figure 2.10) turns out to be an MCP model (neuron with weighted inputs) with some additional, fixed, pre--processing. Units labeled A1, A2, Aj, Ap are called association units and their task is to extract specific, localised featured from the input images. Perceptrons mimic the basic idea behind the mammalian visual system. They were mainly used in pattern recognition even though their capabilities extended a lot more.

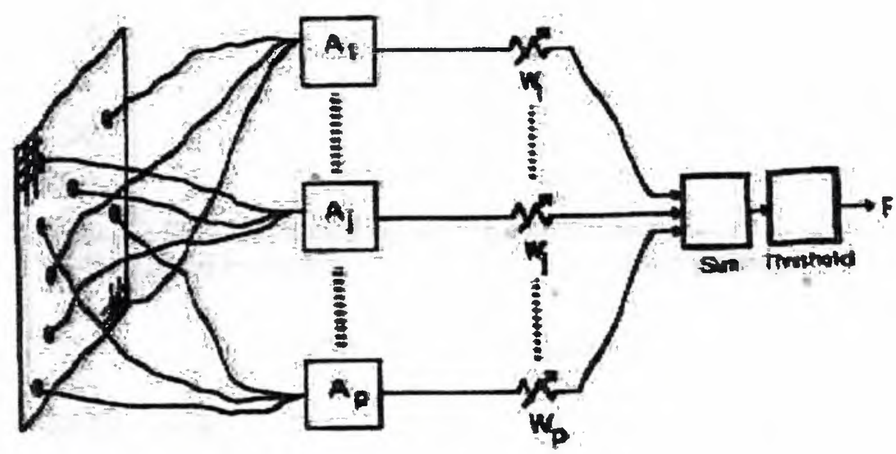

Figure 2.10 The Perceptron

In 1969 Minsky and Papert wrote a book in which they described the limitations of single layer Perceptrons. The impact that the book had was tremendous and caused a lot of neural network researchers to loose their interest. The book was very well written and showed mathematically that single layer perceptrons could not do some basic pattern recognition operations like determining the parity of a shape or determining whether a shape is connected or not. What they did not realised, until the 80's, is that given the appropriate training, multilevel perceptrons can do these operations.

2.8 Competitive learning II. - Limitations and Applications

2.8.1 The Geometric Analogy

This is a useful way of thinking about complex neural systems. Each pattern can represent a point in space, the pattern 1 0 1 is coordinate 1,0,1 in 3 dimensional space. Similarly, the pattern 1 0 1 0 is the coordinate 1,0,1,0 in 4D space. (Space with more than three dimensions are called "hyperspaces".) The weights of the units also represent points in space. In fact, the weights of the units define a point on the surface of a hypersphere (a sphere with 4+ dimensions). This is because the weights must add up to 1 or another constant. Obviously, we cannot imagine spaces with 4+ dimensions but the same relationships are true for spaces with only 2 dimensions. As a result, we can represent the state of a system graphically. Diagram I shows the state of a system before training. Figure 2.11 shows the same system after training.

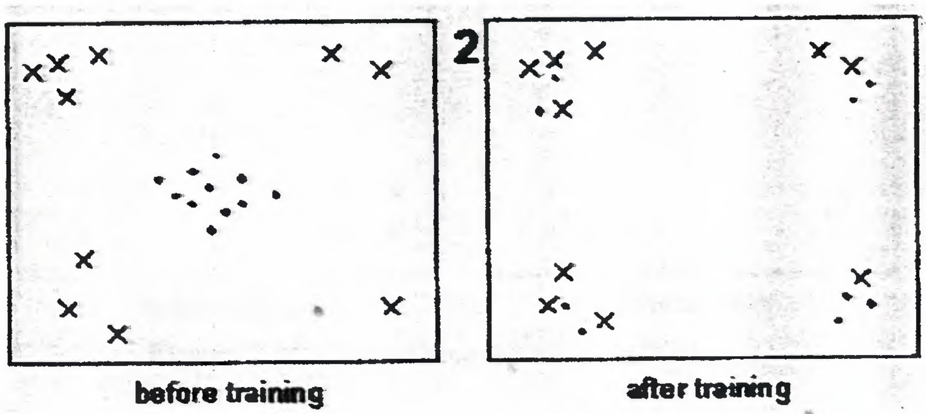

Figure 2.11. Competitive learning II

2.8.2 Note the following:

1. The activation of a unit is inversely proportional to its distance. The nearer a unit to a pattern, the higher its activation will be.

2. When a unit learns, it moves to a point closer to the pattern it is learning. The distance the unit moves is determined by the learning rate. A high learning rate means

that the unit will move quickly towards the pattern.

3. According to the definition of competitive learning, only the unit nearest a pattern will learn that pattern.

4. The distance between patterns shows their similarity. Two very close patterns (e.g. the digits "5" and "3") are very similar, whereas two far patterns (digits "8" and "1") are dissimilar. Exactly what "similarity" means for competitive learning is discussed below.

2.8.3 Classification

Competitive learning can be used to classify a set of patterns/data into a number of output units, one unit responds to each pattern. This is the case in the example program: there are ten units and ten patterns (digits 0-9). If the number of units is less than the number of patterns (as it will be in almost all real world situations) the system will find clusters (groups) of similar patterns: Each unit will respond to a number of similar (i.e., close) patterns. The Figure below shows a possible classification of sixteen patterns by four units [13].

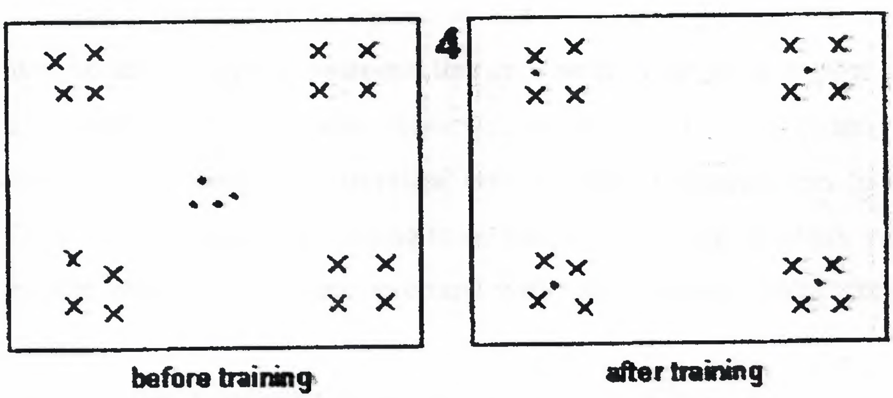

Figure 2.12. Competitive learning classification

2.9 Similarity

What does "similarity" mean? When we say two things are similar, we mean that they have a number of common properties. For example, my cat is similar to another cat, but dissimilar to a table. The types of properties we consider are things like "has whiskers", "is furry", "is living", "is a piece of furniture", etc.

Unfortunately, competitive learning has only a very basic idea of similarity. Two patterns are similar only if they overlap: i.e., if many of the same input units are active for each pattern. This means that competitive learning (in the basic form) will be very bad at classifying visual patterns (which for us, as human beings, are the same or similar) if:

- Pattern is a different size from the learnt pattern.
- Pattern is in a different position in the input.
- Pattern is rotated.

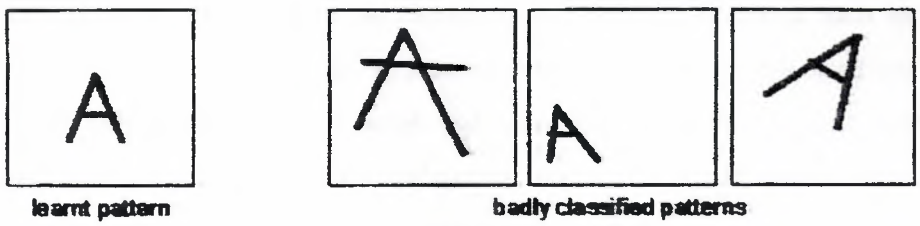

Figure 2.13. Competitive learning similarity

Another example: consider the problem of classifying lines into either horizontal or vertical. We have a system with 64 inputs (an 8x8 array) and two outputs. We want one output unit to be active when a horizontal line is present in the input and the other to be active when a vertical line is present. However, competitive learning cannot learn this. This is because no horizontal line overlaps with another horizontal line. In fact, every horizontal line is more similar (according to competitive learning) to every vertical line. This is because every vertical line overlaps with one element (pixel, etc) of every horizontal line.

The problem is that a competitive learning system in the basic form (with one layer) has no way of learning higher order features. In the case of visual pattern recognition such features are things like "curve", "edge", "junction", etc. Competitive learning must decide what a pattern is ("1", " 2", "3", etc) using only the most primitive low order features. These features, (represented by the input units) mean things like "point at (X, Y)". The system has no way of representing how these points are joined.

A typical classical AI (Artificial Intelligence) solution to this problem uses an

2. 9.1 Algorithm of Similarity

1. Take an array of points
2. Find lines in this array of points
3. Find junctions, curves, edges etc, that tell us how the lines are connected
4. Identify this pattern of connections, the letter "A"

We first find patterns in low order features (points to lines) and then find higher order features by finding patterns within these features (lines to junctions, edges, etc). This suggests that we can solve this problem by using a multi layer network (see Figure 2.10 last notes) where the units in the hidden layer represent features such as "lines", "junctions", etc. This is done in a famous system called "neocognitron" ("new thought"), but this system is very large and complex. However, it uses competitive learning and can learn to recognize handwriting under the conditions above. A three-layer network can also solve the horizontal/vertical problem.

2.10 Benefits of Neural Networks

From the above discussion, it is apparent that a neural network derives its computing power through, first, its massively parallel distributed structure, and second its ability to learn, and therefore generalise; generalisation refers to neural network producing reasonable outputs for inputs not encountered during learning (training). These two information-processing capabilities make it possible for neural networks to solve complex problems. The use of neural networks offers the following properties and capabilities:

Nonlinearity: A neuron is basically a nonlinear device. Consequently, a neural network, made up of an interconnection of neurons, is itself nonlinear. Moreover, the nonlinearity is a special kind os sense that it is distributed throughout the network. Nonlinearity is a highly important property, particularly if the underlying physical mechanism responsible for the generation of an input signal (eg...speech signal) is inherently nonlinear.

Input-output Mapping: A popular paradigm of learning called supervised learning involves the modification of the synaptic weights of a neural network by applying a set of labelled training samples or task samples. Each example consists of a unique input signal, and the corresponding desired response. The network is presented an example picked at random from the set, and the synaptic weights (free parameters) of the network are modified. So as to minimise the difference between the desired response, and the actual response of the network produced by the input signal in accordance with an appropriate statistical criterion. The training of the network is repeated for many examples in the set statistical criterion. The training of the network is repeated for many examples in the set until the network reaches a steady state.

Adaptivity: Neural networks have a built-in capability to adapt their synaptic weights to changes in the surrounding environment. In particular a neural network trained to operate in a specific environment can be easily retrained to deal with minor changes in the operating environmental conditions. Moreover, when it is operating in a nonstationary environment (i.e.. one whose statistics change with time), a neural network can be designed to change its synaptic weights in real time. The natural architecture of a neural network for pattern classification, signal processing, and control application, coupled with the adaptive capability of the network, make it an ideal tool for use in adaptive pattern classification, adaptive signal processing, and adaptive control. As a general rule, it may be said that the more adaptive system is stable, the more robust its performance will likely be when the system is required to operate in a nonstationary environment.

Evidential Response: In the context of pattern classification, a neural network can be designed to provide information not only about which particular pattern to select, but also about the confidence in the decision made. This latter information may be used to reject ambiguous pattern, should they arise, and thereby improve the classification performance of the network.

Fault Tolerance: A neural network, implemented in hardware from, has the potential to be inherently fault tolerant in these in the sense that its performance is degraded gracefully under adverse operating conditions (Bolt, 1992). For example, if a neuron or its connecting links are damaged, recall of a stored pattern is impaired in quality. However, owing to the distributed nature of information in the network, the damage has to be extensive before the overall response of the network is degraded seriously [14].

Uniformity of Analysis and Design: Basically, neural networks enjoy universality as information processors. We say this in sense that the same notation is used in all the domains involving the application of neural networks. This feature manifests itself in different ways.

Neurons, in one form or another, represent an ingredient common to all neural networks.

This commonality makes it possible to share theories, and learning algorithm in different applications of neural network.

2.11 Summary

Many learning laws are in common use. Most of these laws are some sort of variation of the best known and oldest learning law, Hebb's Rule. Research into different learning functions continues as new ideas routinely show up in trade publications. Some researchers have the modeling of biological learning as their main objective. Others are experimenting with adaptations of their perceptions of how nature handles learning. Either way, man's understanding of how neural processing actually works is very limited. Learning is certainly more complex than the simplifications represented by the learning laws currently developed. A few of the major laws are presented as examples.

Among the many interesting properties of a neural network, the property that is of primary signification is the ability of the network to learn from its environment, and to improve its performance through learning; the improvement in performance takes place over time in accordance with some prescribed measure. A neural network learns about its environment through an iterative process of adjustments applied to its synaptic weights and thresholds. Ideally, the network becomes more knowledgeable about its environment after each iteration of the learning process.

CHAPTER THREE

APPLICATIONS OF NEURAL NETWORKS

3.1 Overview

Neural networks have broad applicability to real world business problems. In fact, they have already been successfully applied in many industries.

But to give more specific examples; ANN are also used in the following specific paradigms: recognition of speakers in communications, Prediction, medicine, Classification, Financial, prediction, Medical diagnosis, Noise filtering, Pattern recognition, Process control, Robotics, Signal analysis, Speech synthesis, Targeted marketing...

Neural networks, with their remarkable ability to derive from complicated or imprecise data, can be used to extract pattern and detect trends that are too complex to be noticed by either human or other computer techniques. Their applications are almost limitless but fall into a few simple categories: Classification, Forecasting, Modeling.

Neural network work is progressing in other more promising application areas.Neural networks can be used in virtually any situation where the objective is to determine an unknown variable or attribute from known observations or registered measurements (i.e., various forms of regression, classification, and time series), where there is a sufficient amount of historical data, and where there actually exists a tractable underlying relationship or a set of relationships (networks are relatively noise tolerant).

3.2 Applications of Neural Networks

Here are some past applications of neural networks (or similiar statistical programs).

Neural Nets have been used to:

- Predict staffing requirements at different times of the year and different conditions. Brooklyn Union Gas Corp predicts in advance the number of crew members who will be needed for service calls based on the time of year, predicted temperature, and day of the week.
- Predict which job a job applicant is best suited for. (Brooklyn Union Gas)

- Predict which customers will pay their bills (Brooklyn Union Gas)
- Spot odd trading patterns (this is how Ivan Boesky, the rogue trader, was caught).
- Predict the properties of chemical mixtures.
- Diagnose diseases (One of our customers trained a net that outdid an expert system in diagnosing smell disorders)
- Predict the stock market, the futures markets, etc.
- Flag faulty parts on an assembly line.
- Regulate industrial processes using inputs from sensors at different points in the process.
- Classify medical ailments (such as hearing losses) and classify living things, and classify cells as cancerous/non-cancerous.
- Predict pollution based on the composition of trash coming into an incinerator.
- Predict Sales
- Predict Costs
- Predict a company's corporate bond rating
- Appraise Real Estate
- Predict the outcome of sports events (such as horse racing).
- Predict Solar Flares
- Predict the length of survival for medical patients with ailments such as cirrhosis of the liver.
- Recognize welds which are most likely to fail under stress
- Test beer: Anheuser-Busch: Identifies the organic contents of its competitors beer vapors with 96% accuracy.
- Predict which prison inmates could benefit from less expensive alternative programs. (Delaware correctional system)

3.3 How BrainMaker Neural Networks work

Neural networks are named after the cells in the human brain that perform intelligent operations. The brain is made up of billions of neuron cells. Each of these cells is like a tiny computer with extremely limited capabilities; however, connected together, these cells form the most intelligent system known. Neural networks are formed from hundreds or thousands of simulated neurons connected together in much the same way as the brain's neurons.

Just like people, neural networks learn from experience, not from programming. Neural networks are good at pattern recognition, generalization, and trend prediction. They are fast, tolerant of imperfect data, and do not need formulas or rules. Neural networks are trained by repeatedly presenting examples to the network. Each example includes both inputs (information you would use to make a decision) and outputs (the resulting decision, prediction, or response).

Your network tries to learn each of your examples in turn, calculating its output based on the inputs you provided. If the network output doesn't match the target output, BrainMaker corrects the network by changing its internal connections. This trial-and-error process continues until the network reaches your specified level of accuracy. Once the network is trained and tested, you can give it new input information, and it will produce a prediction. Designing your neural network is largely a matter of identifying which data is input, and what you want to predict, assess, classify, or recognize [15].

3.4 Manufacturing Applications

3.4.1 The use of neural networks in testing plastic quality

Monsanto is using BrainMaker to predicet the quality of plastics to be used in winshields. The lab tests for brittleness and elasticity take up to 4 hours. Through use of a neural network they were able to save research time, proccessing time, and money.

Comments on the program:

"Very easy to use" -Edwin Nazarian
"Very good. Terrific manuals. My main reason for referring product to others" -Azmi Jaferey
Their network contained 18 different inputs including:
temperature,
length of time baked,
proccesing teqniques,
source of the chemicals,
The network was able to predict the plastic quality within a 10% tolerence.

3.4.2 Neural Networks optimizes IC production by identifying faults

In many chip fabrication lines, an engineer analyzes failures to determine what could have caused failure. At Intel, this problem was previously attacked by an expert system, but this was found to be an inadequate tool, particularly in the case of multiple faults.

Also, the expert system proved incapable of generalizing its knowledge, and completely hopeless with new cases.

Dan Seligson, Ph.D., used BrainMaker to create a neural network that could identify the fabrication problem that caused failures in finished Intel VLSI chips. The neural network was developed using information that was originally gathered for an expert system. The neural network was found to be 99.5% correct in generalizing data it had not seen before, but which was similar to that which it had. It was also found that the neural network was capable of distinguishing data which was unlike any it had seen before (i.e. failures of 3 components in the system, when it had been trained with at most 2 failures).

The original expert system was given the electrical test information from finished chips and the corresponding process control variables. The relationship between these two was determined by numerical experimentation and by simulation of CMOS process and device physics. A responsive surface model (RSM) was used to capture the results of exhaustive set of numerical experiments. Simulation tools were used to generate a data base of paired sets of process variables and electrical test measurements. Rules were generated for the expert system from the e-test data and the corresponding process data.

BrainMaker was trained with the same pairs of e-test and process data. Eighteen e-test measurements were used as neural network inputs, and six process variables were the predicted outputs. A training error tolerance of 10% was used. E-test variables were categorized as one of 5 possible values from lowest to highest in order to determine classes of cases. Every legitimate e-test set must have its origin in a set of process variables so there are 15,625 (5^6) possible classed pairs of input/outputs. Of these, there are 24 classes of 1-fault cases and 240 classes of 2-fault cases. actual continuous-valued numbers were used during training. 1500 examples of 2-fault cases were used, half for training, and half for testing the neural network. These cases were chosen randomly. An additional set of 100 3-fault cases was generated for testing the network's ability further. With an error tolerance of 20% (over the entire range of output values) the network correctly responded to 99.5% of the testing examples, indicating very good generalization. With an error tolerance of 10% fewer than 2% of the examples failed testing [16].

A lookup table was also implemented to solve this problem, but it was found that the network performance was superior. The problem was too large to include every single possible case (for either the expert system, the neural network, or the lookup table), so a sampling of data was used. A lookup table is unable to deal with non-linear changes in data as a neural network can.

3.4.3 Neural Networks performs non-destructive concrete strength testing.

In testing concrete for structural imperfections there are many different methods ranging from the drilling of core samples to the use of radar. The first method is destructive, time consuming, and allows for only a small percentage of the total area, while the second requires expensive equipment and isn't effective when steel reinforcement is present. The National Institute of Standards and Technology (NIST) has developed a non-destructive method for testing the internal structure of concrete.

Nondestructive testing (NDT) methods are used to obtain information about the properties or internal condition of an object without damaging it. Steel balls are dropped onto the concrete surface causing soundwaves, which are reflected by cracks and other imperfections in the concrete. These sound waves can then be collected and analyzed by a neural network to determine the probability of a flaw. NIST has developed a system that used the thickness of the concrete as the base measurement and was able to determine the depth of the flaw to 10% accuracy. (The network was able to test a 0.4m thick slab with a 0.2m flaw and determine that the flaw was 40% to 50% the depth of the slab.)

3.4.4 Neural Networks Optimize Enzyme Synthesis

A neural network has been trained to predict the outcome of a chemical reaction controlled by molar ratios, temperature, pressure, amount of enzyme and stirring speed.Kirk, Barfoed, and Bjorkling at NOVO Nordisk A/S in Denmark used the BrainMaker program to train their neural network to predict the amount of desired product and by-product which would be formed after 22 hours of reaction time.

An excellent correlation between predicted yields and experimental results was found. The neural network saves time and money by predicting the results of chemical reactions so that the most promising conditions can then be verified in the lab, rather than performing a large number of experiments to gain the same information.

Initially 16 experiments were performed to identify the most important parameters controlling the process. The molar ratio between fatty acid and glucoside, reaction temperature, pressure, amount of enzyme, and stirring speed were varied. The synthesis yielded ethyl 6-O-dodecanoyl D-glucopyranoside. This experimental data was used to train the neural network to output the amount of the 6-O monoester and a diester by-product, represented as a percentage of yields.

The neural network had three layers: 5 input layer neurons, 4 hidden layer neurons, and 2 output layer neurons. It was trained using the back propagation algorithm with the sigmoid threshold neuron function.Twelve facts were used to train the network to an accuracy of 96% for the outputs. In only a few minutes, all facts were learned. The trained network was then asked to make four predictions on data it hadn't seen before. The network predictions were compared to experimental observations. Very good correlations were found.The average deviation between the network and the experiments was 4% (percentage of yield), ranging between 2% and 7% difference. These deviations are within the normal experimental error of synthesis [17].

After being tested, the network was put to work evaluating thousands of possible conditions in order to find the most optimum.Using a simple algorithm, a test file was generated containing all of the possible values, totalling 9900 cases.The computer-generated test file contained values for each parameter which were both within and without of the training value's range. The entire file ran through the network in 7 minutes and the predictions were saved in a file. Using a search function, predictions for specified yields were selected. Only three cases were found to predict more than 88% monoester with a less than 4% formation of the diester. One of these cases was tested in the lab and the results were close to experimental observation. The network had predicted 88.1% monoester and the experiment yielded 86.2%. The network predicted 4.0% diester, the experiment yielded 4.8%.

Finally, the 9900 predictions were again searched, but this time with additional restrictions more suitable for large-scale chemical processing. Again, the experimental results were very close to the yields predicted by the network.

3.4.5 Using Neural Networks to Determine Steam Quality

AECL Research in Manitoba, Canada has developed the INSIGHT steam quality monitor, an instrument used to measure steam quality and mass flowrate. Steam Quality and Mass Flowrate is the energy injected into the ground in an oil recovery project, for example.

The improvement obtained by using the trained network was immediately apparent. Using a conventional linear program, the standard error of estimate (RMS of deviations about the ideal line) for steam quality and mass flowrate are 28% and 0.59 kg/s. Using the trained neural network, the standard error was 8.2% and 0.34kg/s.

A common test set of 26 sets of input data was used and the network was trained on an additional 100 facts.

Later, a similar network was trained and tested all of the INSIGHT monitor calibration data obtained to date (i.e. data from tests at four different facilities collected over a period of seven years using a minimum of six to a maximum of nine different monitors). Here, the standard error of estimate for steam quality and mass flowrate were 7.7% and 0.4kg/s, respectively.

Recently AECL has successfully trained a neural network to return methanol, gasoline and water contents from the RF reflectance spectra of mixtures of these three components. Currently they are investigating the application of a neural network to a-spectroscopy and to the interpretation of on-line chemical sensor signals.

3.5 Science

3.5.1Neural Networks Predicts Detrimental Solar Effects

Dr. Henrik Lundstedt of Lund Observatory, Sweden, has trained neural networks to predict solar-terrestrial effects such as disturbances in the earth's magnetic fields. The disturbances have been known to cause blackouts, power plant shutdowns, corrosion in pipelines, disruptions in radio and television transmissions, malfunction of geological survey equipment, satellite tracking problems, and other detrimental effects. Being able to predict these occurrences helps prevent disasters.

The major cause of disturbances on earth is certain behaviors of the sun's solar wind. The solar wind is caused by several things such as coronal mass ejections or CMEs (which can trigger flares), and coronal holes. The neural network inputs consist of 37 known values of solar-terrestrial phenomena such as coronal mass ejections, coronal holes, solar sector boundaries, and proton events. The values are input as changes over the last four days. There are eight output neurons. The first output represents whether geomagnetic activity is expected to be quiet for the next day. The second, third and fourth outputs represent whether the activity is expected to be of a minor, major, or severe storm character. The fifth through eighth outputs predict the same items two days ahead.

The neural networks were trained with seven months of solar data from various US databases from CSSA, Stanford, CA; NOAA/SEL, Boulder, CO; and SacPeak/AFGL, NM. A period of data from June 6 - 21, 1990 was omitted from the training data and used for testing. During that period three major storms and one minor storm occurred. However, the traditional prediction method (NOAA/SEL) predicted no major storms and one minor storm which was in fact one of the major storms. The neural network did much better. It predicted two of the three major storms and the minor storm, and predicted a minor storm for the third major storm [18].

3.5.2 Neural Networks Analysis of Transmembrane-spanning Protein Helices

Dr. George Dombi of Wayne State University has developed neural networks, which generalize common themes found in peptides of 25 amino acid length. The sequences were sorted into two groups: transmembrane or nontransmembrance type. 1751 training examples were used. As a result of training, bacteriohodopsin was examined to determine the position of it 7 transmembrane helices.

Using several training and testing experimental procedures, test results were obtained with up to 98% accuracy. A symbolic rather than numeric representation was used. For example, the alanine position 22 was represented by the input A22 being either on or off.

Neural Networks Recognizes Mosquitoes in Flight

A neural network was trained to recognize two species and both sexes of mosquitoes. The frequency of the wingbeat is unique to each sex of each species. The neural network was given information about the wingbeat frequency and correctly classified the insects with a mean accuracy of 98%. Discriminant analysis had provided an accuracy rate of 84%. Even though the mosquitoes were of very similar species, the neural network had no trouble distinguishing them. Potential uses for this type of network include population/biological studies, pollination studies, evaluation of repellents and attractant, pest control, etc.

Aubrey Moore of the Maui Agricultural Research Station, University of Hawaii, developed this network to assess the feasibility of automatically identifying insects in flight. A photosensor was used to detect fluctuations in light intensity caused by reflections off individual mosquitoes flying through a light beam. Digital recording of the photosensor signals were made with an analog-to-digital recorder. A change in light intensity triggered storage of 512 samples. Each signal was converted to a 256-wide frequency spectrum using a Fast Fourier Transform. One input was assigned for each of the 256 spectrum slices. One output was defined for each of the sex/species combinations for a total of four outputs.

3.5.3 Neural Network Processing for Spectroscopy

StellarNet Inc.'s moniker is "Intelligence from Light" -- an intriguingly cryptic way of describing the spectroscopic technology the Florida firm developed to optically analyse objects and substances. StellarNet's Spectroscope bathes or permeates the sample being investigated with various lights, generating optical patterns called "spectra". Designed to identify the object or substance itself and/or the presence and concentrations of various components, Spectroscope uses a BrainMaker neural network in its SpectraNet application to process the spectral data and make the appropriate recognition in real time.

Operating on PC hardware and using a BrainMaker neural network as the processing engine, SpectraNet performs accurate and detailed analysis in areas such as readout calibration for biomedical, environmental, and aerospace fiber optic monitoring sensors, chemical composition determination, quality assurance, process control, industrial monitoring, production control, and various trouble-shooting operations. In the agricultural area, one StellarNet customer is using SpectraNet's neural network capability to identify and assure proper hydration in recently harvested onions.

SpectraNet automates BrainMaker training on inputs such as units of absorbance, transmittance, reflectance, chemical/biochemical composition, percent concentration, and relative irradiance, while incorporating full analysis capability for absorbance, transmittance, reflectance, or absorbance. While customizing neural net applications into turnkey instruments with various options for data acquisition, processing, and graphical display, SpectraNet will gather known spectral examples, analyse the data using quantitative measurements such as wet chemistry and chromatography, and select the spectral regions for training (based on wavelength start, length, and increments). The data is then fed into the neural network for processing and pattern recognition.

To help automate the data preparation process, StellarNet includes a SNAKE utility in all SpectraNet software packages that allows rapid spectral data configuration for training and testing neural networks.

3.5.4 Using a neural network to predict El Nino

Since January of 1991, a research team at the National Oceanic and Atmospheric Administration in Boulder, Colorado has been training a neural network to predict El Nino. According to head researcher, Dr. Vernon Derr, the purpose of the study was to determine if a neural network could accurately predict warm and cold events in the Pacific ocean, and to compare the prediction capabilities of the neural network to other methods, particularly the Persistence method. According to Dr. Derr, the neural network did surprisingly well.

Researchers defined an El Nino or warm event as a departure of more than 1 standard deviation larger than the long term mean in various regions of the Eastern Pacific ocean. If the standard deviation was 1 standard deviation below the long term mean, it was a cold event. While the Persistence method is often used to make weather and climate predictions, it is unable to forecast change or predict the onset of a new situation. The neural network on the other hand was able to show a correlation between the prediction of El Nino and the actual occurrences of warm and cold events in the Pacific. The neural net proved to be a useful device for predicting out to about six months, and depending on the input data, could possibly be useful to the fishing industry [19].

Researchers used input data found in the Comprehensive Ocean Atmospheric Data Set (COADS). COADS is world wide oceans data giving the sea surface on a monthly basis since 1884. Because warm events occur every five to seven years, and because each event is unique in terms of duration, onset and decay, the statistical character of each even is quite varried. As a result, an event is difficult to predict by any means.

For input, Dr. Derr's team primarily used ship's data from various part of the Pacific Ocean dating back to Mathew Fauntainmaury who was the original oceanographer in the Navy. It includes wind, air temperatures, surface temperatures and southern oscillations which is a comparison of sea surface and pressure between Darwin, Australia and Tahiti. It is a know fact that this difference in pressure occurs during El Ninos, but not before, so it is therefore not useful in predicting them.

One part of the research study was to determine the best set of data. According to Dr. Derr, "the set of data we used to predict things over the last year is probably not the ideal set, and we will be using a different set in the future." Because the team used most of the available data for training, only 10% of the data was left for validation and this remaining 10% may not even encompass a period in which El Nino occurred. The network was trained using using the standard sigmoid transfer function. Using the genetic algorithm method, the team studied learning rates and tolerances to determine the best set for the data set they were using. They also varried the number of hidden neurons to determine the optimum number, but have not yet gone to more than 1 layer; although according to Dr. Derr that is something they want to do in the future.

Testing went as follows: In January of 1991, the team started predicting skill scores -- actually the RMS differences between the actual ocean temperature and the predicted temperature-- for up to six months ahead Then in February of 1991, they again predicted (on the basis of current data) for 1 to 6 months ahead. They continued in this manner up until the present time. According to Dr. Derr, "Those were quite good in the sense that the RMS skill differences were in the order of less than a degree averaged over a long period." However one of the problems was that neither the Persistence method nor the neural net did a thorough job of predicting the onset of the warm or cold event. This fact leads Dr. Derr to speculate that the data was not sufficient for the purpose and that it should include not only at the sea surface temperature in Region 4 of the Pacific, but also at least a nine or ten year the history of it [20].

Dr. Derr plans on concluding his studies at the end of 1993. In the means time he plan to employ a rather unusual validation process. The team will train using all the data they have and then they're going to find skill scores for the same period of time again using all the data they have.

3.5.5 Neural Networks Predicts Rainfall

The need for accurate local rainfall prediction is readily apparent when considering the many benefits such information would provide for river control, reservoir operations, forestry interests, flash flood watches, etc. While the data required to make such predictions has been available for quite some time, the complex, ever-changing relationships among the data and its effect on the probability, much less the quantity, of rain has often proved difficult using conventional computer analysis. The use of a neural network, however, which learns rather than analyzes these complex relationships, has shown a great deal of promise in accomplishing the goal of predicting both the probability and quantity of rain in a local area to an accuracy of 85%.

Using BrainMaker neural network software, Tony Hall (a hydrometeorologist from the National Weather Service in Fort Worth, Texas) has developed such a model. Nineteen meteorological variables (e.g. moisture, lift, instability, potential energy, etc.) were used to develope two networks for quantitative predictions--one for the warm season and and one for the cool season. Two additional networks for probability predictions were also generated. Another completely different program, written in C, was developed to allow both the quantitative and the probability networks to run simultaneously with the results appearing on the same computer monitor.

Results to date have been outstanding. In the quantitative model, five categories were used to group the rain fall data (0.01 to 0.49 inches, 0.5 to 0.99 inches, 1.0 to 1.99 inches, etc.) Different tolerances were allowed for each range. For example, the tolerance for the first category was +/-0.2 inches while the tolerance for the higher categories ranged from 0.25 to 0.5 inches. Predictions for the quantitative models have been accurate in a range of 74% to 100% for the five categories with an overall accuracy of 83%.

The probability model used the criteria that a prediction of 30% probability or higher had to correspond to a rainfall of 0.10 inches or more. Otherwise the network output would be considered in error. The accuracy achieved to date for this model is 94% which, when combined with the quantitative results, gives an overall accuracy of 85%.

Sensitivity analysis was performed on the input variables to determine which had the most effect on the output. This will allow the developers to refine the models and improve the accuracy. Since there are six additional sites in Texas that will be included in future studies, means of further automating both the data gathering and BrainMaker operations are being investigated to improve the cost and allow the technology to be used more economically [21].

Although only two years of training and testing data were available, the results achieved to date are believed to be reliable and consistent enough to be used for forecasting guidance. Since this was the original goal of the project, the use of BrainMaker neural networks to predict local rainfall is now expanding to locations in other parts of the country.

3.6 Stocks, Commodities and Futures

3.6.1 Neural Networks and Technical Analysis of Currencies

Mr. James O'Sullivan, of O'Sullivan Brothers Investments, Ltd. (Connecticut) has been successfully using many BrainMaker (California Scientific Software) neural networks on a daily basis for three years to do financial forecasting. He earned $250,000 in one month using neural networks to advise him on his New York Stock Exchange seat trades. Some of his networks are 88-90% accurate in their predictions, according to Mr. O'Sullivan. He uses an automated neural network system that monitors more than twenty different financial markets on a daily basis.

Mr. O'Sullivan has some unusual designs which act more as detectors of specific market conditions, rather than as exact price predictors. He combines the neural network data with other data from his technical analysis software to produce an automated report about a certain market. He gets his data live via satellite from Data Broadcasting and puts it into a charting and technical data module. He has pre-programmed the various algebraic manipulations to be performed on his data before BrainMaker files are built. He does moving averages, changes from period to period, and a few proprietary operations. He runs new data through the system and produces the one-page report in about thirty seconds. He says at least 80% of his decision-making is based on neural network predictions [22].

Mr. O'Sullivan has not fully disclosed his neural network designs to us, but his basic insights are still quite valuable. The key is to ask the neural network the right kind of questions. He asks questions such as "What is the probability of the product (or market) going up 0.618 standard deviations?" and "What percentage of the time does it go up that much?" He also asks questions about the directional behavior of the market and at what price the product (or market) is likely to take off in one direction or the other.

Mr. O'Sullivan's neural networks output several different things such as predicted prices, limits, and directional thrust. One neural network outputs the probability of a certain price occurring the next time period. Another neural network produces best stop price and best target price for long and short positions. Other neural networks produce directional indicators for three market energies. Another predicts the level at which the market is likely to take off.

In one design, the network is given various market conditions as input. The training output is the likelihood of various changes in price. For example, his neural network is told during training that, given similar market conditions, the closing price goes up at least 0.313 standard deviations above the prior day 90% of the time, at least 0.618 deviations 80% of the time, and at least 1 deviation 70% of the time.

An interesting phenomenon of the market is that when a change starts occurring in one direction or the other, there is a point at which it is very likely to continue moving in that direction for several time periods. Once a price reaches that level, there is a reduced risk to buy or sell (whichever is appropriate). Mr. O'Sullivan calls the network that predicts this price level his Risk Barometer network. He uses all the neural networks trained for a specific market when making decisions. For example, if the long term trend is up, the Fast Movement network is a large positive number, his Risk Barometer says 233.092, and the NYSE is at 250, it could indicate an overreacted market that will reverse itself soon.

3.6.2 Predict Bond prices with neural network software

G. R. Pugh & Company (Cranford, NJ) has been using a BrainMaker neural network trained on three-to-four years of historical data with an XT-compatible PC to help forecast the next year's corporate bond prices and ratings of 115 public utilities companies. "An XT is more than sufficient; it's a fast program," company president Mr. George Pugh notes. Learning to use the program and create a neural network from scratch took only two days. The network trained itself to predict bond prices in about four hours.

G. R. Pugh & Company does consulting to predict bond prices for the public utility industry. He maintains databases with financial and business information on the companies advises with business forecasts and credit risk assessments and predicts the financial and operating health of these companies. His expertise is also used by the brokerage industry. He advises clients by forecasting on the selection of good corporate bonds. His clients need to know more accurately which bonds represent good investments for their customers. Both increases and decreases in bond value provide the potential for profitable investment.

Mr. Pugh announced that predicting bond prices with BrainMaker neural network software has been more successful than discriminant analysis and forecasting methods he has used, and even a little better than a person could do. "Discriminant analysis methods are good for getting the direction of lively issues, but neural networks pick up the subtle interactions much better," he explains. The network categorizes the ratings with 100% accuracy within a broad category and 95% accuracy within a subcategory. The mathematical method of discriminant analysis was only 85% accurate within a broad category. (Bonds are rated much like report cards, with broad category ratings such as A, B, C, etc. A subcategory could be A+, for example).

According to Mr. Pugh, "BrainMaker was able to pick up some of the interplays in the inputs that statistical analysis couldn't get." The network makes a significant contribution to his analysis. "The network allows me to pick up things that are not obvious with typical analysis," he says.

Moreover, nearly all of the network's difficulties were found to be associated with companies that were experiencing a particularly unusual problem (such as regulatory risk) or had an atypical business relationship (such as being involved in a large sale and lease-back transaction). Ratings also tend to be subjective; financial items are not the only things considered by the rating companies. These influences were not represented in the training facts and this makes predictions difficult [23].

The trained network forecasts next year's Standard & Poor's and Moody's corporate bond ratings (both are industry standards) from the previous year's S & P and Moody's ratings and 23 other measures of each company's financial strength, such as income, sales, returns on equity, five-year growth in sales, and measures of investment, construction, and debt load. Each of these factors is assigned its own input neuron, and each company's ratings for next year are the outputs of the network.

3.6.3 Predict the S&P 500 Index Neural Network Software

A highly rated investment firm (Clearwater, FL) manages more than 200 million dollars in investments. LBS relies almost exclusively on computer techniques to guide its decisions in predicting the S&P 500 Index. The firm is a forerunner in using neural network software to recognize patterns and predict indexes and trends for financial forecasting.

The latest approach in forcasting used by LBS integrates an expert system with a neural network to make the most efficient use of the talents of each. The expert system provides rules which govern the application of the neural network to the prediction. For example, if the expert system says the market is trending and the neural network forecasts the S&P will go up, then a buy signal is generated.

In predicting the S&P 500 index the neural network is trained with only recent market data (less than five years' worth) because it was found that the actual behavior of the market 25 years ago was not the same as it is today. Commonly available indicators are used such as the ADX, MACD, stochastics, DOW, volume, etc. The BrainMaker neural network "window" was found to be most effective at five market days for predicting the S&P 500 index. It was speculated that every weekday may have a certain "tone" to it, so that all Mondays tend to behave similarly. By presenting five day intervals as historical input data while outputting forcasts for five days in advance, the neural network deals only with the same weekday for each prediction.

The neural network trained by LBS predicts the S&P 500 with an average accuracy of 95%. This statistic was obtained by testing the network on hundreds of days it had never seen before. The network is retrained every night with the most current information to keep its behavior in agreement with the current behavior of the market.

3.6.4 Predicting Stock Prices using Neural Network Software

Warren Buffett is a pillar of the financial world, and with good reason. He has parlayed his theories on investing and market analysis into a substantial fortune, while others have used his advice to build their own highly successful investment portfolios. Some, too, have crunched Buffett's investment formulas, or something like them, into a suite of computer programs that produce an electronic version of the Buffett genius. Walkrich Investment Advisors, a consulting firm out of Cape Girardeau, Missouri, uses BrainMaker Neural Networks to do just that -- produce an investment tool (WRRAT) based loosely on Buffett's ideas and BrainMaker neural networks in predicting stock prices

How well does WRRAT perform in stock price prediction? From January '95 to January '96, a portfolio made up of WRRAT's 20 most underpriced stocks would have seen an average advance of 32.63%, compared to the S&P's 31.93% gain over the same period. More recently, WRRAT's 20 most underpriced stocks have enjoyed a 44.40% gain from January '96 to February '96, compared to the S&P's 38.65%.

How does WRRAT's forecasting compare to the flesh and blood Buffett? From July '95 to February '96, shares in Berkshire Hathaway, Buffett's holding company, have gained an average of 28%. With WRRAT's 1995 average advance of 32.63%, in financial forecasting Walkrich and BrainMaker can compete with the best.

Walkrich uses a BrainMaker neural network to determine the average premium (discount) the market is currently allocating to particular industries, and then uses that standard in an industry-by-industry neural network analysis designed to determine which stocks are trading below their market value. In that analysis, the neural network will appraise each stock, giving a price estimate (based on price/earnings, price/book and dividend yield) which is adjusted for size, industry, exchange listing and institutional influence. The neural net's per-stock price estimate is then compared to the corresponding industry average, producing a calculated measure of each stock's relative value -- in short, whether the stock is being underpriced or overpriced by the market.

3.6.5 A User Friendly Neural Network Trading System

Stock Prophet is a general purpose trading system development tool employing BrainMaker neural network technology to automatically combine multiple indicators into a single clear buy/sell signal. It can be applied to stocks, mutual funds, futures and other financial instruments.

3.6.6 Stock Prophet Highlights

Stock Prophet aids traders by consolidating multiple intermarket factors into a clear trading signal. Many market analysts have a repertoire of favorite indicators, but decision-making is difficult due to conflicting indications of market direction. Stock Prophet employs neural network technology to automatically combine multiple indicators into a single clear buy/sell signal. It does this by providing straightforward development of trading systems based on the artificial intelligence neural network technique as well as conventional technical analysis. The result is "institutional class" technical/quantitative analysis capability for the astute investor. Highlights of Stock Prophet are:

- Applicable to Stocks, Commodities, Mutual Funds and Other Markets.
- Scientific Simulation Shows Extraordinary Profit Potential.
- Clear Signals Given Days and Weeks Before Trade Execution Date. This is in contrast to essentially all technical indicators which are late due to use of smoothing techniques.
- Complete Trading System can be Designed, Trained, and Tested for Profitability Within a Small Fraction of an Hour.

- Over 35 Indicators Plus Indicators of Other Indicators for an Explosive Number of Composite Indicators for Preprocessing.
- Helps to Select the Best Indicators by Analyzing Your Choice of Indicators for Ability to Predict Market Trend Using a Multiple Correlation Technique.
- Convenient Intermarket Capability Gives You the Edge.
- Automation MACRO Allows Easy Updates of Indicators.
- Provides Efficient Interface with BrainMaker and Can Export to EXCEL.
- Computrac/Metastock, Telescan, and ASCII Data Formats Supported.

Stock Prophet's Forecasting System

As nearly all-neural network afficianados agree, the most difficult step in operating a neural network is gathering and preprocessing voluminous, high-quality data. Neural networks, as powerful as they are, depend on applicable data in sufficient amounts, and in an appropriate format, to work their magic.

Stock Prophet automates much of the preprocessing needed to format data for BrainMaker while allowing the user to incorporate a wide array of well-known technical indicators [24].

In a 1995 issue of Technical Analysis of Stocks and Commodities, Technical Editor John Sweeney noted that neural net integration is a major feature of Stock Prophet, saying that the user can "skip developing complex rules (and redeveloping them as their effectiveness fades) . . . just define the price series and indicators you want to use, and the neural network does the rest."

Moreover, as Sweeney goes on to say, "a major benefit of the neural network is that you don't have to define specific trading rules. Instead, the neural network derives the rules during training from the data. When BrainMaker sends an indicator back from its processing, the only rule required is above zero it's a buy, and below zero it's a sell."

Stock Prophet offers a full complement of technical indicators (on-balance volume, open interest, MACD, split volume, acceleration, etc) or the user can implement his or her own indicators by creating them in another program and importing them via ASCII files. Additionally, Stock Prophet's data manipulation features include detrending, summation, limiting values, scaling, Fourier transformations, and biasing. Several of these indicators can be applied to previously generated indicators, thus increasing data preprocessing options. Many of these options (particularly the neural net data preprocessing features) can be automated through Stock Prophet's macro capability.

The value of a Stock Prophet/BrainMaker combination is summed up concisely by Stocks and Commodities' Sweeney: Stock Prophet's unique feature in developing trading signals is that it's neural net indicator is a prediction of future trend, published in advance of the trade date . . . if you get good signals 10 days in advance of the trade, you're going to be one delighted camper! If you could imagine that, try this program out.

3.7 Pattern Recognition Applications

3.7.1 Neural Network Recognizes Voice Mail

By now, everyone is familiar with voice mail technology. You call a business and a voice directs you to use your touch-tone phone to direct your call or to leave a voice message. Of course if you don't have a touch-tone phone, the current voice mail technology isn't accessible, and you need to wait for the operator to help you - unless the system contains a neural network [25].

By using BrainMaker to train neural networks for speech recognition, Dr. Mark Ortner of Compass Technology in Sarasota, Florida has developed P.C. based software that will make voice recognition technology affordable to small and medium sized businesses around the world. Soon, you'll be able to reach the voice mail box of the desired party by phone or fax whether you have touch-tone phone or not.

In 1992, Octel Communications, the world's largest provider of voice and fax information processing and services, acquired Compass Technology. Currently, Dr. Mark Ortner is revolutionizing information processing. So far, Dr. Ortner has trained a network using 2500 facts and 28 words, including the numbers "zero" through "nine", the words, "yes" and "no", and the names of various departments within the company. The degree of recognition accuracy ranges from 90-97%.

The current application is for voice recognition of an extension. Dr. Ortner collects "voice data" by having a variety of callers dial an extension and enter a "mailbox" located at the "voice training center." The callers' words are run through a normalizer (confidential in nature), which creates a BrainMaker output file (fact file). That information is then trained on a neural net.

According to Dr. Ortner, the advantages of this kind of system are far-reaching. In the United States, only about 37% of the telephones are not touch-tone. However in most other countries, touch-tone is nonexistent. By providing a trainable voice recognition system, the purchaser could record the words used at their location, train the network, then run it. A caller could say, "Extension 230" and the system would make the correct phone transfer. Touch-tone is no longer necessary.

Dr. Ortner's network has 400 input neurons, 107 hidden neurons and, at the present time, the output layer has 28 neurons. This will eventually change as more words are added. The output of the network is the recognition of the spoken word which is drawn from a symbol table.

Dr. Ortner uses BrainMaker for training the neural net, but wrote the software that actually runs the neural net. (BrainMaker Professional comes with this code as part of the Runtime License). Since the largest system Compass Technology handles is 32 ports, this kind of voice recognition system is ideal for small and medium sized businesses.

Once this system is on the market, Dr. Ortner will turn his attention to developing a phonic-based system. The caller will be able to pronounce a word and have the system convert it directly to text. This would be a big advantage to the deaf. A deaf person could read a voice message as it is printed on screen or print a message back and have it translated into voice.

3.7.2 Neural Networks Provide Context for OCR

Neural networks offer a general-purpose solution to pattern recognition problems. They are able to generalize much better than traditional programs and can run faster. Neural networks are not limited to any set of characters, and can learn to recognize just about anything, even things like tools, mechanical parts, aircraft, and cancerous cells.

Neural networks are also useful in determining context in conjunction with traditional OCR applications. For example, when reading a book or journal a neural network can look at the words and tell you if it's reading a title, an author, a publisher, or a date. It has been difficult to get traditional programs to quickly provide such contextual information [26].

Electronic Data Publishing, Inc. (Brooklyn, NY) has incorporated a neural network into its OCR/database system. The system reads documents such as journals and papers, and places information into a database for later retrieval into reports or catalogs. The neural network classifies the material read in from an OCR program into categories such as author, title, abstract, publisher or date, so that it can be tagged and stored in a database for later retrieval. "The neural network has saved $20,000 of labor costs in the first two months and allows the same number of people to get four times as much data through the system," said Ken Blackstein, designer of the neural network. The printed material contains too many variations in the data to be effectively classified using a Prolog decision tree. The neural network approach was chosen for its ability to generalize well when given ample data.

This neural network is one of the largest, most successful designs known. The 1440-input, 20-output network was trained with 200 megabytes of data using BrainMaker running on the BrainMaker accelerator board. After roughly 100 training runs, the neural network converged to 96% accuracy on all training examples. In the three months of use with new data, the neural network has made no errors.

Prior to being read by a scanner, the material is photocopied, perhaps enlarged, and cleaned up by people who may also use a felt pen to block out extraneous printed material. The printed pages are then scanned into a PC with the OmniPage (Caere Corporation) OCR program under the Windows environment. The overall system is depicted in figure 1. The words are then processed through the Soundex algorithm which reduces the number of characters and produces a "word" which is similar to a phoneme.(1) This helps the neural network to generalize, because nearly identical printed words such as "Johnson," "Jonson," and "Johnsen," will appear the same to the neural network. This also reduces the number of inputs to the neural network because Soundex "words" are comprised of fewer characters than English words. The design is similar to Sejnowski's famous "NetTalk", except that a full line of text is input rather than seven characters, and the output is a classification rather than a phoneme for speech production.

The output of the neural network is used to place the text into database developed with Netware (Novell, Inc.). Currently, medical literature is on-line with 600 megabytes of data, which is roughly equivalent to 200,000 pages of printed information. Electronic Data Publishing, Inc. has plans for an Engineering database, which would require the training of another neural network that understands engineering terms.

3.7.3 Chaos, Strange Attractors and Neural Networks Plots

Take the last 200 years' data on cotton production. Plot a point, which is one years' production versus the next years'. You get data points scattered all over the screen like stars at night. If you were to plot a LOT of points (without lines connecting them) you get a shape, like a donut. The points seem to fall on or near a circle. This is a Strange attractor.

In a Normal or Real attractor, you get dense collection of points in the middle and spreading out fading out. The price has equilibrium, the production has an equilibrium, represented by the dense collection around a single point. A Strange attractor is an attractor for which there is not an equilibrium point.

There is no math currently that explains the plot of something versus something else which produces the donut. The presence of a Strange attractor means you're dealing with a chaotic system. A chaotic system is a nonlinear feedback system. In the chaotic cotton production system, what you learn by seeing the Strange attractor is that there is some sort of a feedback mechanism, there is an analytic solution to what the system is doing and there is feedback around the analytic solution [27].

You get Strange attractors when you look at the population of foxes over the years as it grows and shrinks. This is chaotic, rather than random. In a random system, you get points scattered all over with no shape whatsoever and there is no underlying mechanism, therefore no way to predict anything. In a chaotic system there is an underlying mechanism with nonlinearity and feedback. It is believed by some that because there is an underlying mechanism analytic approaches can be used to make predictions.

With BrainMaker Professional, you can make plots to find Strange attractors. In NetMaker you put cotton price in a column, cotton price shifted down by one in another, plot one on the X and one on the Y. Plot lots of months worth of data. You will see a donut, a Strange attractor, which indicates an underlying mechanism with nonlinearity and feedback. If you discover the underlying math that explains this, please call us immediately.

3.7.4 Neural Networks Recognize Chemical Drawings

Pattern recognition is a commonly encountered problem when computers are required to get information the from the physical world around them. It may be easy enough to get a digital picture via a camera or a scanner into a computer file, but how does the computer know what the data means? Recent advances in commercially available optical character recognition software have provided some affordable solutions, particularly when fonts are similar and the material is relatively clean. Blind people can even purchase a scanner and software, which will read aloud to them. However, there are still real limits to what most commercial software can recognize. Most have difficulty when the print is sloppy, small or varies considerably. None offer the ability to recognize arbitrary shapes, symbols or graphics.

Recent studies in pattern recognition with neural networks have been sponsored by the US Post Office to read ZIP codes. Even though they are primarily interested in hand-written digits, the techniques developed are general. Feature extraction from bitmaps is the biggest problem. An approach for feature extraction uses Fourier descriptors of the items to be recognized. One such application, described here, reads a chemical drawing (comprised of characters and graphics) and translates it into a chemical structure database [28].

Compounds are described in two ways: as a chemical drawing of connected atoms, or as a list of atoms and their connections in a connection table. A connection table can be easily stored on computer, but most printed sources such as books, journals and papers use the more easily recognized drawings. The connection tables uniquely define compounds and can be used to index information in a database. When chemical compound descriptions are placed in a database with other information they can be used for patent searches, environmental studies, toxicology studies, and precursor searching, for example.

Fein-Marquart Associates, Inc. has developed a program which automatically reads printed chemical drawings and translates them into connection tables in a database. The old approach required manual computation of the connection table. Commercially available optical character recognition programs were not able to read the chemical drawings because many use a very small print (6 and 8 point) and there are graphic elements present as well as standard English characters.

The system was developed by Fein-Marguart and uses a neural network trained with BrainMaker Professional to recognize the printed characters and graphics. The system has a 98% recognition success rate according to Joe McDaniel, Senior Staff Member at Fein-Marquart. The chemical drawings are read into a PC from a scanner, some mathematical processing is performed to provide Fourier descriptors which are then fed into a neural network for recognition and translation into bonds and atomic symbols. The output of the neural network is formatted into a connection table and transmitted to a host computer database.

Fourier descriptors are computed by tracing the outline of a character to create a concave hull. This data is stored as a list of x and y coordinates. If one views the x portion of the data as the real and the y as the imaginary portion of a complex data pair, and then performs a Fourier transform on the list, the result will be a list of complex data points representing frequency. Straight lines or big curves can be interpreted from low frequency data, and corners, serif and end-of-lines from high frequency data. Characters and graphics have frequency magnitude and phase signatures which can be recognized by the neural network.

Low frequency data can be interpreted as straight lines or big curves, and high frequency data as corners, serif and end-of-lines. Characters and graphics have frequency magnitude and phase "signatures" which can be recognized by the neural network. The neural network is given the frequency information as input and is trained to translate information into bonds and atomic symbols.

The output of the neural network is formatted into a connection table and transmitted to a host computer database. When chemical compound descriptions are placed in a database with other information, they can be used for patent searches, environmental studies, toxicology studies, and precursor searching.

3.7.5 Decoding Algorithms and Predicting Sequences with Neural Networks

The ability to predict data sequences is important in data transmission to provide error correction. Certain algorithms can predict repetitive code with good accuracy, but fail in the presence of noisy code sequences.

Mr. James Johnson of Netrologic, Inc. (Dayton, OH) trained a BrainMaker neural network on noisy data and was able to predict code sequence accuracy from 62% to 93%, depending upon the initial conditions and the presence or absence of noise. Higher accuracy could probably be obtained by training a network with a wider variety of training samples.

The network was given an input of 100 bits generated using this algorithm:

b(a) = b(a-3) XOR b(a-31) where 32^{2} a ^{2} 100.

The network was asked to predict what the 101st bit should be in that sequence with no explicit knowledge of how the string was formed. The equation used to generate the bits contained a 31-bit random seed. A set of 1,000 training facts was generated to train a back propagation net. The first data sets were generated with sets of correlated data; that is, five sets of 100 bits were generated using the algorithm above and a 31-bit seed that was identical except that it was shifted right one additional position for each subsequent set of data to generate five separate sets of 100 bits. Then a new random 31-bit seed was generated and five more correlated 100-bit sets were produced.

The network learned all of the 1,000 training sets to within 10%. A test set was generated of 500 sets of 100 strings. The network got 468 out of 500 correct [29].

3.8 Sports Applications

3.8.1 Predicting Thoroughbreds Finish Time with Neural Networks

When Don Emmons' neural network picked the winning horse in 17 out of 22 thoroughbred races at Detroit Race Course, he was astounded that a $195 program (BrainMaker) running on a PC could do so well. "I am amazed at the ability of software to consistently include the winning horse in three of its picks," said Don.

Designing a neural network is largely a matter of defining the problem well in your own mind. The most difficult aspects are deciding what information you're going to use and gathering it.

There are several known methods of successfully predicting horserace winners with neural networks BrainMaker Professional provides a program which automates the design of "competition" networks such a horserace predictors. The program "Compete" designs, tests and runs a network based upon comparisons of all the competing items, comparing them two at a time. The one that wins the largest number of 2-item comparisons is the overall winner. Each item is rated for its overall likelihood of winning.

Another design approach which can be used with standard BrainMaker uses the full number of competing items as input, such as ten horses. There are two difficulties with this. First, not all races have the same number of horses running. Second, There will be a lot of inputs and outputs, making it more difficult to train.

Don Emmon's design approaches the problem by looking at each horse individually and predicting how well it might do in the race. A separate neural network is trained for each horse with past performance information. Then the neural networks are given current information and the ones with the best results are selected as likely winners. When Don selected three horses as the possible winner of the race, 77% of the time one of them was the winner [30].

Information may be gathered from the Racing Form or from a computerized service such as the Equine Line of Jockey Club Information Systems, Inc. in Lexington, Kentucky. BrainMaker will read several different databases or a plain text file. Each horse has its own file and every past race is on a different line in the file. Don started out with four pieces of information: post position, the number of horses in the race, the horse's finish time and the track record. An input file might look like this:

```
        post_position #horses finish_time  track_record
race#1        4           8        106.10       101.05
race#2        2           8        115.34       101.05
race#3        6           7        127.22       101.05
(etc.)
```

A little work was done with the numbers before the network was trained. The finish time was divided by the track record to provide a common denominator between all horseraces. Linear regressions were done on sets of three consecutive races in order to rate the recent performance as improving, staying the same or getting worse. The final design included the post position, the number of horses, the finish time as a percent of track record, and the change in recent performance as inputs. The output was the predicted finish time of the horse's next race expressed as a percentage of track records.

```
             post position--> | one     |
         number of horses--> | horse   | --->predicted time
              finish time--> | neural  |
    change in performance--> | network |
```

NOTES: Times are expressed as a percentage of track record.The change in performance is precalculated with a linear regression equation is used to produce a slope which describes the horse's recent performances as improving, not changing or getting worse.

While the network is being trained, the inputs (the left portion of the diagram) represent the information from a past race. The output is what the horse did at its next race. At least eight races worth of information are needed for training. The program presents the races to the network one a time, over and over, until the network learns what the horse has done in the past. Don typically let the training continue until the network output numbers which were 95% accurate. The training can be done just before the race since it takes about five minutes to complete. After the network is trained, it can be used to predict an upcoming race Information about the horse's last race is input, and the output is a prediction of what the horse will do at this (the next) race.

In order to yield the best results from this network, a race selection system was created:

1. No races with maiden horses or allowance races are used.
2. No races with more than nine horses are used.
3. Every horse must have at least eight past performances
4. Best odds for profit are with races that have at least a 37% longshot to favorite ratio.

3.9 Business, Management, and Finance

3.9.1 Neural Netorks Predicts Gas Index Prices

Dr. Al Behrens of Northern Natural Gas in Nebraska has developed a neural network that predicts next month's gas price change with an average accuracy of 97%. Northern Natural Gas is a regulated wholesaler of natural gas. They must develop and file a rate for gas based on the volume-weighted average cost of gas. Prices and terms are specified in contracts. Being able to predict costs provides a valuable piece of planning data.

The monthly price is sometimes tied to an index such as those published in Inside FERC and Natural Gas Week. The price is a function of many factors, including recent market activity, seasonal factors, weather, etc. Dr. Behrens used seven inputs to the neural network, which included some past information.

3.9.2 Maximize Returns on Direct Mail with Neural Network Software

Microsoft, a leading computer software developer based in Redmond, Washington, is using BrainMaker neural network software to maximize returns on direct mail. Each year, Microsoft sends out about 40 million pieces of direct mail to 8.5 million registered customers. Most of these direct mailings are aimed at getting people to upgrade their software or to buy other related products. Generally, the first mailing includes everyone in the database. The key is to send the second mailing to only those individuals who are most likely to respond .

Company spokesman Jim Minervino when asked how well BrainMaker neural network software had maximized their returns on direct mail responded, "Prior to using BrainMaker, an average mailing would get a response rate of 4.9%. By using BrainMaker, our response rate has increased to 8.2%. The result is a huge dollar difference that brings in the same amount of revenue for 35% less cost!" [31].

To get a BrainMaker neural network to maximize returns on direct mail, several variables were fed into the network. The first objective was to see which variables were significant and to eliminate those that were not. Some of the more significant variables were:

- Recency - the last time something was bought and registered, calculated in number of days. It is known facts that the more recently you've bought something, the better the chance you're going to buy more.
- First date to file - the date an individual made their first purchase. This is a measure of loyalty. The longer you've been a loyal customer, the better the chance is you're going to buy again.
- The number of products bought and registered.
- The value of the products bought and registered - figured at the standard reselling price.
- Number of days between the time the product came out and when it was purchased. Research has shown that people who tend to buy things as soon as they come out are the key individuals to be reached.

Additional variables include information taken from the registration card including yes/no answers to various questions - scored with either a one or zero - areas of interest like recreation, personal finances, and such personal information as age, and whether an individual is retired or has children. Microsoft also purchased data regarding the number of employees, place of employment, as well as sales and income data about that business. While Microsoft has designed this neural network for their own specific needs, some of these inputs could be applied to any network.Prior to training, the information taken from the response cards was put into a format the network could use and yes/no responses were converted to numeric data. Minimums and maximums were also set on certain variables.Initially, the network was trained with about 25 variables. To make sure the data was varied, it was taken from seven or eight campaigns and represented all aspects of the business including the Mac and Windows sides, from high and low price point products.

The model trained for about seven hours before it stopped making progress. At that point, variables that didn't have a major impact were eliminated. This process was repeated. Currently the model is based on nine inputs. Jim Minervino explains some of the other training considerations: "During training I used 'modify size' and I used 'prune neurons'; as training completes, I used 'add neuron', and we did an experiment with 'recurrent operations' although in the net model we ended up using the default."The output was a quantitative score from zero to one indicating whether an individual should receive or should not receive a second mailing. Minervino found that anybody scoring above .45 was more responsive to the mailing than anybody below.The neural network was tested on data from twenty campaigns with known results not used during training. The results showed repeated and consistent savings. An average mailing resulted in a 35% cost savings.

3.9.3 Credit Scoring with Neural networks software

According to research conducted by Herbert L. Jensen, Ph.D., an Ernst & Young Research Fellow at California State University Fullerton, "building a neural network capable of analyzing the credit worthiness of loan applicants is quite practical and can be done quite easily."

The credit scoring neural network was trained on no more than 100 loan applications yet achieved a 75-80% success rate. One day's work by an operator familiar with the BrainMaker software package was required to build, train and test the credit scoring neural network. Except for showing a greater bias towards approving weak loan applications, the neural network's loan classification rate was identical to that achieved using a commercial credit-scoring scheme.

The input data for the credit scoring with Brainmaker neural network software study consisted of information typically found on loan applications. The outcomes of those loans were classified as delinquent, charged-off, or paid-off. The actual outputs from the network were 0 to 1 ratings for the three alternatives.

Once the network was built, it was subjected to two training trials. In the first trial, the data was arranged in random order and the first 75 applications were used to train the network. The remaining 50 applications were then evaluated using the trained network. The network misclassified 10 of the 50 applications in the sample for an 80% success rate. In short, the network favored approving loan applications. More traditional and much more costly, credit scoring method used by 82% of all banks, resulted in a 74% success rate. The credit scoring method proved to be more conservative than the neural network in granting credit

In the second trial, the data was rearranged in different random order and the first 100 applications were used to train the network. The remaining 25 applications were then evaluated using the trained network. The network misclassified 6 of the 25 applications in the sample for a 76% success rate. Classifications of good loans as bad and of bad loans as good were equal at 12% each. The credit scoring method for this sample of 25 applications also misclassified 6 of the 25 applications.

3.9.4 Real Estate Appraisal with Neural Networks

Neural networks can be used to predict the sale price of a home. The information provided by the neural network helps appraisers make assessments, helps sellers determine appropriate asking prices, and helps homeowners decide if improvements would be cost-effective. As the neural network designer, your primary responsibilities are to clearly define the problem and present the data in such a way that the network can find patterns. Once this is accomplished, training the network is mostly a fine-tuning process.

The Problem

Traditional methods for determining the value of real estate include appraisal by a certified expert, computer-assisted appraisal and, of course, the actual sale price. The problems inherent with these valuation methods are the inconsistency between appraisers, the inability of machines to consider more than rules and mathematical formulas, and the effects of changing market conditions.

Neural networks do not fall victim to these problems. When applied to real estate appraisal, neural networks are able to predict the actual sale price of properties with 90% accuracy. Neural networks perform better than multi-variate analysis, since networks are inherently nonlinear. They can also evaluate subjective information, such as a neighborhood rating, which is difficult to incorporate into traditional mathematical approaches. Richard Borst, a Senior Vice President at Day & Zimmerman, Inc., the nation's leading provider of mass appraisal services to state and local governments, has successfully trained a neural network to appraise real estate in the New York area. His network incorporates eighteen data items, which include the number of dwelling units, fireplaces, plumbing fixtures, square feet of living area, age, months since last sale, and air conditioning. He uses 217 sales records from 1988 and 1989 with prices ranging from $103,000 to $282,000. His network was trained on a 386 using BrainMaker Professional v2.5 (California Scientific Software: Nevada City, CA).

The Data

The data used in Mr. Borst's network, collected by the mass appraisal firm, Cole-Layer-Trumble, represent sales from a single area. The data chosen are similar to what an appraiser would examine to make an assessment. The table below lists all variables used in the original network design and the range of possible values for each. All values are continuous except two, heating type and neighborhood group. These two inputs represent categories, but since each has only two possible values, they don't need to be divided into separate inputs.

3.9.5 Neural Network Red-Flags Police Officers With Potential For Misconduct

The Chicago Police Department has used BrainMaker to forecast which officers on the force are potential candidates for misbehavior. The Department's Internal Affairs Division used neural networks to study 200 officers who had been terminated for disciplinary reasons and developed a database of pattern-like characteristics, behaviors, and demographies found among the 200 police officers.

BrainMaker then compared current Department officers against the pattern gleaned from the 200-member control group and produced a list of officers who, by virtue of matching the pattern or sharing questionable characteristics to some degree, were deemed to be "at risk."

This particular application has been highly controversial, drawing criticism from several quarters - the most vocal being Chicago's Fraternal Order of Police. William Nolan, the Order's president, has made Orwellian references, saying the Department's program seems like "Big Brother." Scientific American, Playboy, New Scientist, and Law Enforcement News have all done articles on the ethical implications of the Chicago P.D.'s program with mixed reviews.

The C.P.D. Internal Affairs Division, however, was pleased with the results. After BrainMaker studied the records of the 12,500 current officers (records that included such information as age, education, sex, race, number of traffic accidents, reports of lost weapons or badges, marital status, performance reports, and frequency of sick leaves) the neural network produced a list of 91 at-risk men and women. Of those 91 people, nearly half were found to be already enrolled in a counseling program founded by the personnel department to help officers guilty of misconduct. The I.A.D. now intends to make the neural network a supplement to the counseling program because, as Deputy Superintendent Raymond Risely said, the sheer size of the Chicago police force makes it "pretty much impossible for all at-risk individuals to be identified [by supervisors]."

Terry Heckart, a graduate student at Ohio's Bowling Green State University, recommended neural networks to Chicago's Internal Affairs Division. Heckhart told the Division officials that the software could be effective for two reasons: one, as the number of variables increase in the application, the output reliability increases; secondly, neural nets can deal with missing data. That, says Risley, "was really the key to solidifying our interest."

"We're very pleased with the outcome," Risley says, "We consider it much more efficient and capable of identifying at-risk personnel sooner than command officers might be able to do. The old method just can't compete with it."

The Chicago Police Department stresses that the program utilizing BrainMaker has no punitive ramifications. Risley notes that "it's not disciplinary . . . it's an opportunity for an officer who is moving in the wrong direction to rehabilitate himself . . . if an officer refuses to participate, nothing happens to him."

Despite the ethical discussion raging over whether a neural network should be used to monitor human beings, the program cannot be accused of being subjective and personally biased as "manned" programs often are. Clearly, the software can hold no personal grudges and seeks only to dispassionately identify patterns and characteristics that could spell trouble. The alternative system, being human based, cannot avoid subjectivity and bias on some level. It is worthy of note that the Fraternal Order of Police "vehemently opposed" the Department's old system for that very reason.

To counterbalance the inherent "dispassion" of the neural network, the Department closely examined the net's findings to ensure that officers who are clear anomalies, and thus don't warrant being on the list, are removed from consideration. This combination of objective technology and subjective humanity does not necessarily spell perfection, but it does signify a promising move in that direction.

Currently, we are told, the Chicago Police Department does not use BrainMaker to forecast problems with officers. The program was apparently terminated due to its controversial nature.

3.9.6 Managing Jury Summoning with Neural Networks

The Intelligent Summoner from MEA (Norristown, PA) allows a courthouse to dramatically reduce the number of jurors called for potential service, saving an average of 25-40% of the cost of jurors. Courts often waste money by calling (and paying for) more potential jurors to show up and wait around than are needed for the cases being heard. The Montgomery Court House in Norristown, PA saves $70 million annually using this system.

The Intelligent Summoner determines the number of jurors needed for the next day at a specific court house. The program is custom-tailored for a specific courthouse by MEA with information about that courthouse. At least one year's worth of past information about the courthouse is needed for custom-tailoring. The information can be entered into the Intelligent Summoner system and includes dates, judges, types of cases, and number of jurors used at a particular court house. A file is written to disk which is then sent to MEA.

MEA then creates and trains a neural network using BrainMaker, which learns the needs of that courthouse. The trained neural network is sent back to the courthouse and the file is read onto the PC. From then on, the system is ready for use at that courthouse by anyone with typing skills.

To use the system, a daily survey of tomorrow's judges, trial types, and size of jury panel is input with the data-entry portion of the program. The program will immediately provide the total number of jurors that should be called for tomorrow.

3.9.6 Forecasting Required Highway Maintenance with Neural Networks

We've all driven on a road that is full of pot holes or cracks. You can barely hold your commuter cup and youre anxious to get around that big semi so you can get into the smooth lane. But then you ask yourself, didn't they just fix this road last summer? Chances are you're right. But experienced highway maintenance engineers are hard to find, and as a result, the appropriate treatment isnt always selected.

Professor Awad Hanna at the University of Wisconsin in Madison has taken the guess work out of the maintenance and repair process by training BrainMaker to become a maintenance expert. If a seasoned professional isn't available, a recent college graduate and a computer program can do the job with a high degree of confidence. Since there is no mathematical formula to solve this kind of problem, its an ideal application for BrainMaker.

Professor Hanna trained the neural network with information provided by experts who can tell with a high degree of accuracy (confidence) which type of concrete is better than another for a particular problem. These experts were given a variety of situations and asked to provide various treatments. Professor Hanna then trained using the back propagation method on 1 hidden layer. Currently Professor Hanna is developing a simple program to be used with BrainMaker that will take the input from the user and produce the most appropriate output based on previous experience provided by these senior people.

Some of the inputs include qualitative values for temperature and volume of a particular piece of pavement. Due to lack of funds, the number of input values was limited to 10. The output is the pavement treatment associated with a degree of confidence. For example, the recommended treatment might be chip seal with a confidence of 8 out of 10. Because there are so many variables, rarely is there a situation that occurs with 100% confidence.

While Professor Hannas research is focused on a Midwestern area that experiences cold, ice and snow, and is based on the input of experts from this area only, his methodology could be applied to any geographic location. If human experts are not available to provide input, routine maintenance data from any Department of Transportation can be used instead. According to the Professor, Usually there is some kind of historical record of work that has been done in particular sections of road over the last few years as well as an annual evaluation of the Riding Comfort Index.

The Riding Comfort Index is a rating of how comfortable you are on a particular section of the highway. The smoothest, best road would score 10 out of 10. A bumpy road for example would score a 5. Roads are measured before treatment and again a year later. If a year after treatment the road is still scores high it means the treatment was a good one. If it the score is low it means the treatment wasnt really appropriate.

To validate his results, Professor Hanna is seeking funding to test his program. His testing will include more highway locations, different types of cracks and bumps, and data collected at different times of the year. He plans to record various sections of a particular highway (For example I 94 between miles 101 and 102); classify the cracks found there (severe, medium or light); look at external factors (annual daily traffic, air temperature, amount of snow removal, wind factor); input all facts into the software, and obtain recommendation for the ideal treatment .

BrainMakers transfer function will be used to determine the exact confidence -- based on information provided in the training phase.

3.10 Control Applications for Neural Network Systems

These applications are typically found in a manufacturing plant or service center.

- Predictive control
- Process optimization
- Instrumentation sensors
- Process diagnostics
- Product formulation

3.11 Applications by Functions for Neural Network Systems

- Classification
- Financial prediction
- Medical diagnosis
- Noise filtering
- Pattern recognition

- Process control
- Robotics
- Signal analysis
- Speech synthesis
- Targeted marketing
- Vision

3.12 Applications by Industry for Neural Networks

- Chemical processing
- Credit
- Financial
- Food and beverage
- Materials processing
- Marine
- Medical
- Power generation
- Pulp and paper

3.13 Networks for Data Association

The previous class of networks, classification, is related to networks for data association. In data association, classification is still done. For example, a character reader will classify each of its scanned inputs. However, an additional element exists for most applications. That element is the fact that some data is simply erroneous. Water stains might have rendered credit card applications unreadable. The scanner might have lost its light source. The card itself might have been filled out by a five year old. Networks for data association recognizes these occurrances as simply bad data and they recognize that this bad data can span all classifications.

3.14 Networks for Data Conceptualization

Another network type is data conceptualization. In many applications data is not just classified, for not all applications involve data that can fit within a class, not all applications read characters or identify diseases. Some applications need to group data that may, or may not be, clearly definable. An example of this is in the processing of a database for a mailing list of potential customers. Customers might exist within all classifications, yet they might be concentrated within a certain age group and certain income levels. Also, in real life, other information might stretch and twist the region, which contains the vast majority of potential buyers. This process is data conceptualization. It simply tries to identify a group as best as it can.

3.15 Networks for Data Filtering

The last major type of network is data filtering. An early network, the MADALINE, belongs in this category. The MADALINE removed the echoes from a phone line through a dynamic echo cancellation circuit. More recent work has enabled modems to work reliably at 4800 and 9600 baud through dynamic equalization techniques. Both of these applications utilize neural networks, which were incorporated into special purpose chips.

3.15.1 Character Recognition

Character recognition is another area in which neural networks are providing solutions. Some of these solutions are beyond simply academic curiosities. HNC Inc., according to a HNC spokesman, markets a neural network based product that can recognize hand printed characters through a scanner. This product can take cards, like a credit card application form, and put those recognized characters into a database. This product has been out for two and a half years. It is 98% to 99% accurate for numbers, a little less for alphabetical characters. Currently, the system is built to highlight characters below a certain percent probability of being right so that a user can manually fill in what the computer could not. This product is in use by banks, financial institutions, and credit card companies.

Odin Corp., according to a press release in the November 4, 1991 Electronic Engineering Times, has also proved capable of recognizing characters, including cursive. This capability utilizes Odin's propriatory Quantum Neural Network software package called, QNspec. It has proven uncannily successful in analyzing reasonably good handwriting. It actually benefits from the cursive stroking.

The largest amount of research in the field of character recognition is aimed at scanning oriental characters into a computer. Currently, these characters require four or five keystrokes each. This complicated process elongates the task of keying a page of text into hours of drudgery. Several vendors are saying they are close to commercial products that can scan pages.

3.15.2 Image (data) Compression

A number of studies have been done proving that neural networks can do real-time compression and decompression of data. These networks are auto associative in that they can reduce eight bits of data to three and then reverse that process upon restructuring to eight bits again. However, they are not losses. Because of this losing of bits they do not favorably compete with more traditional methods.

3.15.3 Pattern Recognition

Recently, a number of pattern recognition applications have been written about in the general press. The Wall Street Journal has featured a system that can detect bombs in luggage at airports by identifying, from small variances, patterns from within specialized sensor's outputs. Another article reported on how a physician had trained a back-propagation neural network on data collected in emergency rooms from people who felt that they were experiencing a heart attack to provide a probability of a real heart attack versus a false alarm. His system is touted as being a very good discriminator in an arena where priority decisions have to be made all the time.

Another application involves the grading of rare coins. Digitized images from an electronic camera are fed into a neural network. These images include several angles of the front and back. These images are then compared against known patterns, which represent the various grades for a coin. This system has enabled a quiek evaluation for about $15 as opposed to the standard three-person evaluation, which costs $200. The results have shown that the neural network recommendations are as accurate as the people-intensive grading method.

Yet, by far the biggest use of neural networks as a recognizer of patterns is within the field known as quality control. A number of automated quality applications are now in use. These applications are designed to find that one in a hundred or one in a thousand part that is defective. Human inspectors become fatigued or distracted. Systems now evaluate solder joints, welds, cuttings, and glue applications.

3.16 Summary

The most common use for neural networks is to project what will most likely happen. There are many applications where prediction can help in setting priorities. For example, the emergency room at a hospital can be a hectic place. To know who needs the most time critical help can enable a more successful operation. Basically, all organizations must establish priorities, which govern the allocation of their resources. This projection of the future is what drove the creation of networks of prediction. A network that can classify could be used in the medical industry to process both lab results and doctor-recorded patience symptoms to determine the most likely disease. Other applications can separate the "tire kicker" inquiries from the requests for information from real buyers.

In data association, classification is still done. For example, a character reader will classify each of its scanned inputs. However, an additional element exists for most applications. That element is the fact that some data is simply erroneous. Water stains might have rendered credit card applications unreadable. The scanner might have lost its light source. The card itself might have been filled out by a five year old. Networks for data associations recognize these occurrences as simply bad data and they recognize that this bad data can span all classifications.

CHAPTER FOUR

MEDICAL APPLICATIONS

4.1 Overview

One important area of application of neural network is the medical field. Such applications range from statistical techniques for interpretation epidemiological data, to predictive techniques for performing diagnosis, to systems, which interpret real-time signals from intensive-care units, and report any unusual conditions.

Artificial Neural Networks (ANN) is currently a 'hot' research area in medicine and it is believed that they will receive extensive application to biomedical systems in the next few years. At the moment, the research is mostly on modeling parts of the human body and recognizing diseases from various scans (e.g. cardiograms, CAT scans, ultrasonic scans, etc.).

Neural networks are ideal in recognizing diseases using scans since there is no need to provide a specific algorithm on how to identify the disease. Neural networks learn by example so the details of how to recognize the disease are not needed. What is needed is a set of examples that are representative of all the variations of the disease. The quantity of examples is not as important as the 'quantity'. The examples need to be selected very carefully if the system is to perform reliably and efficiently.

4.2 When Can Artificial Neural Networks Be Applied to Medicine?

The A.N.N. approach to the analysis of data will see extensive application to biomedical problems in the next few years. It has already been successfully applied to various areas of medicine, such as diagnostic aides, biochemical analysis, image analysis, and drug development.

4.2.1 Medical Diagnostic Aids.

The application of A.N.Ns in diagnosing heart attacks received publicity in the Wall Street Journal when the A.N.N. was able to diagnose with better accuracy than physicians. This application is significant because it was used in the emergency room where the physicians are not able to handle large amounts of data.A commercial product employs A.N.N. technology in the diagnosis of cervical cancer by examining pap smears. In clinical use, this product has proven to be superior over human diagnosis of pap smears.In the United Kingdom, an A.N.N. used in the early diagnosis of myocardial infarction is currently undergoing clinical testing at four hospitals. At the research level, A.N.Ns are used in diagnosing ailments such as heart murmur, coronary artery disease, lung disease, and epilepsy.This technology is also being used in the interpretation of Electrocardiograms (ECG) and Electroencephalograms (EEG).

4.2.2 Biochemical Analysis.

A.N.Ns are used in a wide variety of analytical chemistry applications. In medicine, A.N.Ns have been used to analyze blood and urine samples, track glucose levels in diabetics, determine ion levels in body fluids, and detect pathological conditions such as tuberculosis. At Pacific Northwest National Laboratory, A.N.Ns are being combined with chemical sensor arrays and spectrometers for use in automated chemical analysis.

4.2.3 Medical Image Analysis.

A.N.Ns are used in the analysis of medical images from a variety of imaging modalities. Applications in this area include tumor detection in ultra-sonograms, detection and classification of micro-calcifications in mammograms, classification of chest x-rays, tissue and vessel classification in Magnetic Resonance Images (MRI), x-ray spectral reconstruction, determination of skeletal age from x-ray images, and determination of brain maturation. At Pacific Northwest National Laboratory, A.N.Ns are being developed to examine thallium centigram images of the heart and identify the existence of infarctions. Another project at Pacific Northwest National Laboratory uses A.N.N. technology to aid in the visualization of three-dimensional ultrasonic images.

4.2.4 Drug Development.

Researchers at the National Institutes of Health as well as other institutions have used A.N.Ns as tools in the development of drugs for treating cancer and AIDS. A.N.Ns are also used in the process of modeling biomolecules. At Pacific Northwest National

Laboratory, A.N.N. technology is being incorporated into the software used to model protein molecules.

4.3 Neural networks in medicine

Artificial Neural Networks (ANN) is currently a 'hot' research area in medicine and it is believed that they will receive extensive application to biomedical systems in the next few years. At the moment, the research is mostly on modeling parts of the human body and recognizing diseases from various scans (e.g. cardiograms, CAT scans, ultrasonic scans, etc.).

Neural networks are ideal in recognizing diseases using scans since there is no need to provide a specific algorithm on how to identify the disease. Neural networks learn by example so the details of how to recognize the disease are not needed. What is needed is a set of examples that are representative of all the variations of the disease. The quantity of examples is not as important as the 'quantity'. The examples need to be selected very carefully if the system is to perform reliably and efficiently.

4.3.1 Modelling and diagnosing the Cardiovascular System

Neural Networks are used experimentally to model the human cardiovascular system. Building a model of the cardiovascular system of an individual and comparing it with the real time physiological measurements taken from the patient can achieve diagnosis. If this routine is carried out regularly, potential harmful medical conditions can be detected at an early stage and thus make the process of combating the disease much easier.

A model of an individual's cardiovascular system must mimic the relationship among physiological variables (i.e., heart rate, systolic and diastolic blood pressures, and breathing rate) at different physical activity levels. If a model is adapted to an individual, then it becomes a model of the physical condition of that individual. The simulator will have to be able to adapt to the features of any individual without the supervision of an expert. This calls for a neural network.

Another reason that justifies the use of ANN technology is the ability of ANNs to provide sensor fusion, which is the combining of values from several different sensors. Sensor fusion enables the ANNs to learn complex relationships among the individual sensor values, which would otherwise be lost if the values were individually analyzed.

In medical modeling and diagnosis, this implies that even though each sensor in a set may be sensitive only to a specific physiological variable, ANNs are capable of detecting complex medical conditions by fusing the data from the individual biomedical sensors.

4.3.2 Electronic noses

ANNs are used experimentally to implement electronic noses. Electronic noses have several potential applications in telemedicine. Telemedicine is the practice of medicine over long distances via a communication link. The electronic nose would identify odours in the remote surgical environment. These identified odours would then be electronically transmitted to another site where a door generation system would recreate them.

4.3.3 Instant Physician

An application developed in the mid-1980s called the "instant physician" trained an autoassociative memory neural network to store a large number of medical records, each of which includes information on symptoms, diagnosis, and treatment for a particular case. After training, the net can be presented with input consisting of a set of symptoms; it will then find the full stored pattern that represents the "best" diagnosis and treatment.

4.4 Neural Network for Breast Cancer Diagnosis

Breast cancer diagnosis has been approached by various machine-learning techniques for many years. This paper describes two neural network based approaches to breast cancer diagnosis, both of which have displayed good generalization.The first approach is based on evolutionary artificial neural networks. In this approach, a feedforward neural network is evolved using an evolutionary programming algorithm. Both the weights and architectures (i.e., connectivity of the network) are evolved in the same evolutionary process. The network may grow as well as shrink.

The second approach is based on neural network ensembles. In this approach, a number of feedforward neural networks are trained simultaneously in order to solve the breast cancer diagnosis problem cooperatively. The basic idea behind using a group of neural networks rather than a monolithic one is divide-and-conquer. The negative correlation training algorithm we used attempts to decompose a problem automatically and then solve them.

4.5 A Computer Program for Neural Network Aided Diagnosis of Inherited Metabolic Diseases

We have developed a prototype computer program, MetaNet, that uses a combination of artificial neural networks and knowledge-based expert systems to assist in the diagnosis of inborn errors of metabolism in children.Results of amino acid analysis data of normal children, and of patients diagnosed with a number of amino acid and organic acid abnormalities were used as inputs to train the neural network component of the program. To diagnose new cases, plasma or urinary amino acid results are entered. The knowledge-based expert system then asks questions of the user regarding the presence or absence of common clinical and/or biochemical abnormalities.

Using both the amino acid data and the answers to the questions, the MetaNet program integrates the output of the neural network and the results of the knowledge-based expert system to yield a provisional diagnosis.The diagnostic output is accompanied by a numerical *belief vector*, which indicates the degree of confidence of the program in the diagnosis. Altering any of the input variables followed by reprocessing of the data generates a new diagnostic output and a revised belief vector. This allows analysis of the importance of any input variable to the proposed diagnosis. The neural network component consists of eight, three-layer neural networks that are trained using a back-propagation approach. Analysis of the hidden layers following training of the neural network revealed both expected and novel, unexpected connections between specific diagnoses and clusters of amino acids. Such data may be used as a guide for future investigation of the contribution of the metabolism of specific amino acids to amino acid disorders.

4.6 Investigation of the use of Neural Networks for Computerised Medical Image Analysis.

Advances in clinical medical imaging have brought about the routine production of vast numbers of medical images that need to be analyzed. As a result an enormous amount of computer vision research effort has been targeted at achieving automated medical image analysis. This has proved to be an elusive goal in many cases. The complexity of the problems encountered has prompted considerable interest in the use of neural networks for such applications. However, many reports of such work have been unsatisfactory in that often only qualitative results are reported, or only few patient cases are used. This thesis presents a study of the use of neural networks and computer vision for medical image analysis, which aims to quantitatively investigate and demonstrate the potential of neural networks in such an application. A medical image analysis problem was selected which would facilitate this. The problem chosen was the automatic detection of acoustic neuromas in MR images of the head.

Since neural networks excel at statistical pattern recognition tasks a broadly bottom-up approach to the problem was adopted. Neural networks were utilized for `intelligent' tasks, which were supported by more conventional image processing operations in order to achieve the objectives set. The prototype system developed as a result of the study achieved a100% sensitivity and 99.0% selectivity on a dataset of 50 patient cases.

4.7 Tumor Diagnosis Using Backpropagation Neural Network Method.

For characterization of skin cancer, an artificial neural network (A.N.N.) method has been developed to diagnose normal tissue, benign tumor and melanoma. The pattern recognition is based on a three-layer neural network fuzzy learning system. In this study, the input neuron data set is the Fourier Transform infrared (FT-IR) spectrum obtained by a new Fiberoptic Evanescent Wave Fourier Transform Infrared (FEW-FTIR) spectroscopy method in the range of 1480 to 1850 cm-1 [32]. Ten input features are extracted from the absorbency values in this region. A single hidden layer of neural nodes with sigmoids activation functions clusters the feature space into small subclasses and the output nodes are separated in different nonconvex classes to permit nonlinear discrimination of disease states. The output is classified as three classes: normal tissue, benign tumor and melanoma. The results obtained from the neural network pattern recognition are shown to be consistent with traditional medical diagnosis. Input features have also been extracted from the absorbency spectra using chemical factor analysis. These abstract features or factors are also used in the classification.

4.8 Traditional Difficulties in handling Medical Data

Medical data typically requires a large amount of preprocessing in order to be useful. There is numeric and textual data interspersed. Frequently different symbols are used with the same meaning; "male" may be denoted as "M", "m", 0 or a variety of other formats. One medication or condition may be commonly referred to by a variety of names. There is often a redundancy of data; age may appear in several places. Erroneous data is very common; medical terms are frequently misspelled. Finally, medical data is frequently sparse; when a structure is imposed on medical data much of the structure remains empty for a large portion of the population due to the breadth required of any structure.

A robust data preprocessing system is required in order to draw any kind of knowledge from even medium sized medical data sets. The data must not only be cleaned of errors and redundancy but also organized in a fashion, which makes sense for the problem; in this study's context, the data must be organized so that the benefits of using unsupervised Neural Networks may be maximized.

4.8.1 Organizing Medical Data

Standard techniques were employed to clean erroneous and redundant data; for example, "GI Bleeding", "Gi Bleed" and "Gastrointestinal Bleeding" were all mapped to "Gastrointestinal Bleeding". If "Albuterol" appeared more than once for a patient, multiple entries were discarded.

The data set used contains only data at the leaves; each leaf node is present in only one or two tuples on average. This poses two problems: any conclusions formed would be statistically insignificant and the level of computation required for such an analysis would be exceedingly high due to the existence of roughly ten thousand different leaf nodes. To alleviate these problems the data was processed at the root level of each tree. At the root level each tree collapses into much fewer nodes; fourteen root level drugs, sixteen root level topographies and ten root level morphologies. By constraining all data to the root level the degree of differentiation has been greatly reduced from thousands to 40 (14 + 16 + 10). The trade-off in cost is a great reduction in precision, while a benefit is the possibility of detecting trends within the data at the general level.

As the trees were collapsed to the root level the per tuple data was converted to bipolar format. For every tuple each of the 40 root level nodes was assigned a value of either 1 or -1 depending on whether any data existed for the leaves of that root node. The node is assigned a value of 1 if at least one data value is present at the leaves. The node is assigned a value of –1 if no data is at the leaves. In other words, only existence is preserved; quantity is lost. The end result for each tuple is a 40-dimensional bipolar array. The original data is at most 18-dimensional - containing one to six drugs one to six topographies and one to six morphologies. Each dimension may contain one of thousands of values, and there is no structure to the data due to variable dimensionality and null and repeated values. By contrast, every tuple of the transformed data contains 40 dimensions, each of which may take one of only two values, 1 or -1. The transformed data is much more consistent and lends itself to computationally intensive analysis such as Neural Networks. NeuralWare, Inc., Data Sculptor Manual. Pittsburgh: NeuralWare Technical Publications Group, 1994 [33].

4.9 Classify Breast Cancer Cells with Neural Network Software

A human, who decides the degree of cancer present, traditionally examines breast cancer cells under a microscope. People are inconsistent in these judgments from day to day and from person to person.A BrainMaker neural network that classifies breast cancer cells has been developed.

The system was developed by Andrea Dawson, MD of the University of Rochester Medical Center, Richard Austin, MD of the University of California at San Francisco, and David Weinberg, MD, PhD of the Brigham and Womens' Hospital and Harvard Medical School of Boston. Initial comparisons showed that BrainMaker is in good agreement with human observer cancer classifications.

Cancer cells are measured with the CAS-100 (Cell Analysis System, Elmhurst, IL). There are 17 inputs to the neural network, which represent morphometric features such as density and texture. There are four network outputs representing nuclear grading. The cancerous nucleus is graded as being well, moderate, or poorly differentiated, or as benign. Correct grade assignments were made between 52% and 89% of the time on cases not seen during training. The lower success rate (for well differentiated) may have been due to the smaller percentage of this type in the training set. In addition, heterogeneity is much lower in well-differentiated tumors. Cancerous nuclei were classified within one grade of the correct grade.

4.10 Neural network Improves Hospital Treatment and Reduces Expenses

A new hospital information and patient prediction system has improved the quality of care, reduced the death rate and saved millions of dollars in resources at Anderson Memorial Hospital in South Carolina. The CRTS/QURI system uses neural networks trained with BrainMaker to predict the severity of illness and use of hospital resources. Developed by Steven Epstein, Director of Systems Development and Data Research, the CRTS/QURI system's goal is to provide educational information and feedback to physicians and others to improve resource efficiency and patient care quality.

The first study showed that the program was directly responsible for saving half a million dollars in the first fifteen months even though the program only included half of the physicians and three diagnoses. Since then, the number of diagnoses and physicians included in the program has increased. The quality of care has improved such that there are fewer deaths, fewer complications, and a lower readmission rate. Expenses have been reduced by fewer unnecessary tests and procedures, lowered length of stays, and procedural changes. The reported success has motivated several other hospitals to join in the program and has provided the impetus to begin a quality program with the state of South Carolina.

Individually trained neural networks learn how to classify and predict the severity of illness for particular diagnoses so that quality and cost issues can be addressed fairly. After attempts to use regression analysis to predict severity levels for several diagnoses failed, Epstein turned to the BrainMaker program for a new approach and taught his neural networks to classify and predict severity with 95% accuracy. The neural networks are also used to predict the mode of discharge - routine through death - for particular diagnoses.

Training information is based upon the length of stay in the hospital, which has a direct relationship to the severity of the illness (acuity). The neural network uses variables of seven major types: diagnosis, complications/comorbidity, body systems involved (e.g., cardiac and respiratory), procedure codes and their relationships (surgical or nonsurgical), general health indicators (smoking, obesity, anemia, etc.), patient demographics (race, age, sex, etc.), and admission category.Three years of patient data was chosen for training. There were approximately 80,000 patients and 473 primary diagnoses. For a given diagnosis, about 400 to 1000 cases were used for training. Two neural networks for each diagnosis were trained - one to predict the use of resources and the other to predict the type of discharge. For a single diagnosis network, there are 26 input variables and one output variable.

4.11 Neural Network predicts functional recovery

The Arcon Group provides accurate predictions of the functional recovery of patients over the Internet. These individual data based predictions are displayed in the form of line-graphs and delivered to clinical personnel in a few seconds. The predictions lower hospital length-of-stay, improve subacute and home care outcomes, and significantly reduce the cost of patient care. They have proven invaluable for Quality Improvement, Resource Utilization and managing care.

Arcon's FACT system predictions are derived from extensive research in the area of Rehabilitation Medicine and the broad and detailed medical database that resulted from it. The methodology incorporates state-of-the-art predictive power of Artificial Neural Nets and global, instantaneous communication over the Internet.

In client hospitals where FACT is currently operating, length-of-stay has dropped thirty percent within the populous geriatric diagnosis related groups (DRG's) where functional recovery is a key determinant of hospital discharge.

The founder and President, Loren M. Fishman M.D., Corporate Vice-President, Victor Oppenheimer Vice President of Legal and Business Affairs, Marc L. Bailin, Esq., and other Arcon Group personnel develop purely data-driven tools valuable for improving institutional effectiveness and efficiency, such as Arcon's FACT system.

Based in New York City and Cambridge, Massachusetts, Arcon's advisory board includes medical, educational, business and legal professionals of international distinction. With its comprehensive mastery of emerging technologies, Arcon provides the health care industry with an accurate forecast of the course of functional recovery that is totally objective and yet sensitive to each individual's uniqueness.

4.12 Diagnose Heart Attacks with Neural Network Software

When a patient complaining of chest pains is received by the emergency room, it is no simple matter to diagnose a heart attack. Merely examining the patient and performing an electrocardiogram (EKG) is not often enough. If a patient is suspected of having experienced a heart attack, several blood samples are drawn over the next 4 to

48 hours. Patients with heart tissue damage will have various cardiac enzymes appear in their blood. There is a characteristic pattern of the change in enzyme levels during the short period after a heart attack. Using all three techniques (EKG, exam, and blood analysis), a doctor can diagnose and treat heart attack patients. Neural network methods were found to correlate closely with expert human analysis, providing another opinion doctors can use to make a correct and timely diagnosis.

A physician at St. Joseph Mercy Hospital in Michigan designed a neural network that recognizes cases of acute myocardial infarction (AMI, commonly called heart attack) using the cardiac enzyme data from series of tests on patients. The input consisted of two sequential cardiac enzyme tests and the number of hours between the tests. The output was "1" if the patient had a heart attack and "0" if the patient did not. The network was trained with 185 examples from 47 patients using blood tests that were not more than 48 hours apart. There were a total of 21 inputs and 1 output as shown below. The network was trained to a 10% error tolerance on all training data.

The neural network was then tested on 53 new sets of data. The data represented sets of serial cardiac enzyme data for ten patients with AMI and eight patients without AMI. Neural network outputs of less than 10% probability of AMI were classified as no-AMI cases. Outputs of at least 90% probability of AMI were classified as AMI cases. Outputs between 10% and 89% were interpreted as ambiguous or uncertain.

The neural network's diagnosis was then compared to three experts. One evaluated patients on the basis of ECHO/EKG changes. Another used the cardiac enzyme data. A third used autopsy reports. The network agreed with 100% of the AMI cases diagnosed by the cardiac enzyme expert, and 93% of the non-AMI cases. The 7% difference occurred where the network was uncertain. The network agreed with 86% of the AMI cases diagnosed by the EKG expert, and 33% of the non-AMI cases. In one case the EKG data was misleading due to multiple past heart attacks. In another case the network was uncertain. The network agreed with the autopsy results in 92% of the AMI cases, and 67% of the non-AMI cases. In one case the networks were uncertain, and in another the heart had been damaged but by another cause.

4.13 Neural Network Orders Medical Laboratory Tests for ER

When a patient appears at the emergency room door it is sometimes an hour until a doctor can see him or her. It may be another hour until the lab can do the ordered tests. In order to save patient waiting time, Dr. Steven Berkov of Kaiser Hospital in Walnut Creek, California, developed a neural network program that can order the lab tests as soon as the patient is admitted. Up to 38 labs the neural network can order tests [34].

Not only does the neural network save up to two hours of patient waiting time, it can reduce expenses. When the pilot system was tested, it reduced the number of tests that were ordered by 10-15%.Dr. Berkov says it could save half a million dollars a year. The neural network is able to reduce the number of tests for two reasons. First, medical records are used as examples for training the neural network. It can be determined which tests were actually necessary in retrospect, so the neural network can be trained to order only the pertinent tests. Second, nurses had been given the blanket permission to order tests and they tended to order even more than doctors.

The neural network has 67 inputs that include patient demographics and symptoms. This information is gathered when the patient is admitted and placed in the medical record. There are 38 outputs, each representing a different test that might be ordered. The pilot system neural network was trained on 250 patients from past hospital medical records.

When the pilot system was tested on new patients it was found to be about 95% accurate, according to Dr. Berkov. Most of the time the neural network would order most of the necessary tests. Sometimes it did not order enough, but Dr. Berkov explained that usually the doctor would only need to call the lab and order another test on the already collected specimen.Dr. Berkov is working on a neural network that is even more robust and focused. He is waiting for funds from the hospital to pursue this project on a larger scale and implement it in the hospital.

4.14 Classifying Psychiatric Patients for Care with Neural Networks

Dr. George Davis of the Augusta Mental Health Center (Augusta, ME) has trained a BrainMaker neural network, which predicts the length of, stay (LOS) for psychiatric patients' [35]. His system (available through Psybernetics, Inc. Augusta, ME) allows state hospitals and private facilities to determine which patients would benefit most from short stays and which require long-term (thirty days or more) hospitalization.

The system has the potential of providing annual savings of $100,000 to a 300 bed private facility, and up to $600,000 for a tertiary (state) facility. Separating short term from long term stay patients upon admission rather than after some period of observation saves time and money. Fewer inappropriate hospitalizations occur, which not only saves the state money, but also allows the short-term patient to benefit from community settings and support systems, and reduces the psychosocial stigma of hospitalization. In addition, there is a lessened burden on the legal system and law enforcement agencies, and less paperwork. Short-term patients who require hospitalization are more likely to be admitted to a general hospital because they may still retain insurance benefits.

The neural network performs better than traditional approaches in predicting the length of stay (LOS). Only 8-30% of the variance in LOS could be correlated to a combination of demographic, diagnostic and clinical variables. By comparison neural networks were able to explain 39-86% of the variance. There are 48 or 49 kinds of input data used to train two different neural networks. The inputs include basic demographics, admission history, family support systems, ability to care for self, diagnosis information, etc.

There are four output neurons. The outputs define four classes of LOS: 1) less than 1 week, 2) greater than 1 week but less than thirty days, 3) greater than thirty days but less than six months, 4) greater than six months but less than one year. Four networks were trained using 500-600 cases, two each on two different years of annual data. In this way, the predicted LOS for particular patients could be compared between two years at the institution, which had undergone major organizational changes.

Psychiatric diseases are the most difficult to predict. In addition, varying standards and funding policies make care more susceptible to chaos and difficult to compare between locations. Neural networks provide a means to predict the effectiveness of care for a specific location. Several neural networks can be trained with patient data from particular time periods, which will provide a method of judging the effectiveness of changing policy, procedure, or available resources. For example, if several "typical" patient cases are created or particular troublesome cases are selected, these can be run through the different networks to determine which changes in treatment would be most beneficial.

4.15 Diagnosing Giant Cell Arteritis with Neural Network

Five doctors have trained a neural network using the American College of Rheumatology (ACR) database of patients with vasculitis. The ACR has developed standards for classifying a number of rheumatic diseases. In addition to traditional classification approaches, other methods have been used such as decision trees, linear discriminate function analysis, logistic regression, and neural networks.

For the classification of Giant Cell Arteritis (GCA) of patients in the ACR database, all approaches have been used and compared. BrainMaker was trained on this set of patients, with the ACR diagnosis standards for comparison reasons. The inputs to the neural network were eight ACR predictor variables: 1) age greater than 50, 2) new localized headache, 3) temporal artery tenderness or decrease in a temporal artery pulse, 4) polymyalgia rheumatica, 5) abnormal artery biopsy, 6) erythrocyte sedimentation rate greater than 50mm/hour, 7) scalp tenderness or nodules, and 8) claudication of the jaw, tongue or on swallowing. If the predictor variable was present, a 1 was input. If the variable was not present, a 0 was input. The output was a 1 if the patient was diagnosed as having GCA or a 0 if not.

There were 807 patients in the database, 214 with GCA and 593 with other forms of vasculitis. The 807 patients were broken into three groups for neural network design and testing. One group of 80 or 81 patients was set aside for testing. A second group of 200 patients was set aside for monitoring the training (testing while training). A third group of 526 patients was set aside for training. Ten sets of these triplet groups were created using a different set of 80 or 81 patients each time. Ten different neural networks were trained on slightly different training groups.

After training, each network was tested on its corresponding testing set. In this way, the networks would test each and every case in the database without having seen the case during training. The trained networks correctly classified 94.4% of the testing cases that had GCA and 91.9% of the cases that did not have GCA.

4.16 Summary

In this chapter which is specialized in a Neural Network in medicine. No one denies that Neural Network has played a very essential rule in medicine as well as in many other applications. So here N.N. in order for the physician to hold or to deal with large quantities of data, the N.Ns has been placed instead of him, so it can handle these quantities and other samples of data and then it will not be that difficult to diagnosis the diseases and tumors as will.

Neural Networks also gives facilities to physicians to access the record of the past patients data, which that may help the hospitals to minimize the excessive costs from unnecessary laboratory testing and ineffective patient treatment. N.N. also has been used to develop the drugs for treating the cancer and other tumors.

CONCLUSION

First chapter presented the background as well as the history of the neural networks, which shows the developments and improvements through previous years. And also the benefits and advantages that made the neural network new technology in order to be used in many successful applications.

In addition, the future of the neural networks is wide open and may lead to many answers or questions. Because it is technology gives a computer system an amazing capacity to actually learn from input data. Artificial neural networks have provided solutions to problems normally requiring human observation and thought processes. Either humans or other computer techniques can use neural networks, with their remarkable ability to derive meaning from complicated or imprecise data, to extract patterns and detect trends that are too complex to be noticed. And also the Neural Networks have many advantages such as Adaptive learning, Self-Organisation, Real Time Operation, and Fault Tolerance via Redundant Information Coding . . . etc.

In chapter two the architectures of neural networks were described Networks such as the one just described are called artificial neural networks (ANNs), in the sense that they represent simplified models of natural nerve or neural networks. The basic processing element in the nervous system is the neuron. Neural networks process information in a similar way the human brain does. The network is composed of a large number of highly interconnected processing elements (neurones) working in parallel to solve a specific problem. The commonest type of artificial neural network consists of three groups, or layers. Of units: a layer of "input" units is connected to a layer of 'hidden" units, which is connected to a layer of"output" units.

Obviously the idea of designing the neural networks referred to the human brain behaviours. Therefore neural networks learn by example. They cannot be programmed to perform a specific task. The examples must be selected carefully otherwise useful time s wasted or even worse the network might be functioning incorrectly. A trained neural network can be thought of as an "expert" in the category of information it has been given to analyse. Some neural network structures are not closely to the brain and some does not have a biological counterpart in the brain. However, neural networks have a strong similarity to the biological brain and therefore a great deal of the terminology is borrowed from neuroscience.Neural networks are sometimes called machine-learning algorithms, because changing of its connection weights (training) causes the network to learn the solution to a problem. The strength of connection between the neuron is stored as a weight-value for the specific connection. The system learns new knowledge by adjusting these connection weights. This method is proven highly successful in training of multilayered neural nets. The learning ability of a neural network is determined by its architecture and by the algorithmic method chosen for training. The training method usually consists of one of two schemes: Unsuppressed learning, supervised learning. There is a variety of learning laws, which are in common use. These laws are mathematical algorithms used to update the connection weights.

Chapter three was aimed to show the fields, where the neural networks can be applied. So many applications of the neural network either in the real world or predicted applications., also used in the following specific paradigms: recognition of speakers in communications; diagnosis of hepatitis; recovery of telecommunications from faulty software; interpretation of multimeaning Chinese words; undersea mine detection; texture analysis; three-dimensional object recognition; handwritten word recognition; and facial recognition.Neural networks are trained by repeatedly presenting examples to the network. Each example includes both inputs (information you would use to make a decision) and outputs (the resulting decision, prediction, or response). Your network tries to learn each of your examples in turn, calculating its output based on the inputs you provided. Neural networks are good at pattern recognition, generalisation, and trend prediction. They are fast, tolerant of imperfect data, and do not need formulas or rules.

Chapter four was the final chapter; it is aimed to determine the applications of neural network specifically in. medicine whether in medical diagnostic aids, biochemical analysis. Also described the developing the drugs using A.N.Ns.

REFERENCES

[1] Galatea, "Exploration of model of thinking and consciousness Neural Network named (Helen)".

[2] Frank Rosenblatt (psychologist) "Purposed the Perceptron", 1958.

[3] Rumelhart D.E. "Purposed the error back-propagation learning method", 1986.

[4] Nippon Steel Crop "built a blast furnace operation control support system that makes use of ANN", 1992.

[5] Daiwa Securities Co. & NEC "Apply Neural Network to learn and recognize of stocks price".

[6] Nippon Oil Co. " The first applying of neural networks to facilities diagnosis".

[7] Adnan Khashman, Lecture Notes of COM420, "Neural Network", N.E.U.

[8] J.Anderson, "A simple Neural Generating an Interactive Memory" 14:197-220,1972.

[9] Aihara, K., T. Takabe, M. Toyda. 1990. Chaotic Neural Networks. Physics Letters A144: 333-340.

[10] Schi\oler, H. and Hartmann, U. (1992) "Mapping Neural Network Derived from the Parzen Window Estimator", Neural Networks, 5, 903-909.

[11] An Introduction to Neural Computing. Alexandria, and Morton, H. 2nd edition, http://www.calsci.com/intro ToNN.html

[12] "Introduction to Neural Networks by Jeannette Lawrence", http://www.calsci.com/intro ToNN.html

[13] Generation5: Artificial Intelligence Repository-Introduction-, http://library.thinkquest.org/1824/nnintro.shtml

[14] Neural Networks at pacific Northwest National Laboratory, http://www.emsl.pnl.gov:2080/docs/cie/neural/neural.homepage.html

[15] Eberhart RC and Dobbins RW (1990) Early Neural Network Development History: The Age of Camelto. IEEE Engineering in Medicine and Biology 9, 15-18.

[16] Jagielska I. (1993), "The application of neural networks to business information systems", Proceedings of 4th Australian Conference on Information Systems, Brisbane, pp. 565-574.

[17] Windsor C. G. & Harker A. H. (1990), "Multi-variate financial index prediction - a neural network study", Proceedings of International Neural Network Conference, Paris, France, pp. 357-360.

[18] Wang H. A. & Chan A. K.-H. (1993), "A feedforward neural network model for Hang Seng Index", Proceedings of 4th Australian Conference on Information Systems, Brisbane, pp. 575-585.

[19] Ling C. S. (1993), "Choosing the right neural network model for trading", Proceedings of First Symposium on Intelligent Systems Applications, Singapore, pp. 26-33.

[20] Eberhart R. C. & Dobbins R. W. (1990), Neural Network PC Tools - A Practical Guide, San Diego: Academic Press.

[21] C.H. Messom, C.J. Hinde, A.A. West, D.J. Williams. 1992. "Designing Neural Networks for Manufacturing Process-Control Systems." Proceedings of the 1992 IEEE International Symposium on Intelligent Control, Glasgow, Scotland.

[22] Industrial Applications of Neural Networks (research reports Esprit, I.F.Croall, J.P.Mason.

[23] "Intelligent Manufacturing Trough Neural Networks - Monitoring of Machining Processes." Intelligent Autonomous Systems IAS-3, Vol.? Ch. 72, pp. 328-337.

[24] Makridakis et al. (1983), Forecasting: "Methods and Applications", New York: Wiley.

[25] Neural Networks at Pacific Northwest National Laboratory,
http://www.emsl.pnl.gov:2080/docs/cie/neural/neural.homepage.html

[26] Artificial Neural Networks in Medicine,
http://www.emsl.pnl.gov: 2080/docs/cie/techbrief/NN.techbrief.ht

[27] DARPA Neural Network Study (October, 1987-February, 1989). MIT Lincoln Lab. Neural Networks, Eric Davalo and Patrick Naim.

[28] MeDavid Brown. 1994. "Artificial Neural Networks for Biomedical Applications." The Workshop of Computer-aided. Diagnosis in Medical Imaging, Georgetown University Medical Center, Washington, DC, (5-6 July 1994).

[29] H. Burke. 1994. "A Computerized Prediction System for Cancer Survival that Uses an Artificial Neural Network." First World Congress on Computational Medicine, Public Health and BioTechnology, Austin, TX, 25-28 April.

[30] Data and Analysis Center for Software "Artificial Neural Network Technology", 1992 http:// www.docs.dtic.mil/techs/neural/neural.litle.html

[31]. Academic writing. From the world wide web: "http://rodin.wustl.edu/center.html".

[32]. Academic writing. From the world wide web: " http://www.pnl.gov/medical/".

[33]. Academic writing. From the world wide web: "http://www.emsl.pnl.gov:2080/".

[34]. Writing papers. From the world wide web: "http://www.ee.umd.edu/medlab/papers/".

[35]. Academic writing. From the world wide web: "http://www.calsci.com/".