Journal of Artificial Intelligence, Machine Learning and Soft Computing
Volume 1, Issue 1, January-June 2016

Role of Artificial Neural Networks in Soft Computing: A Survey

Er. Sunny Thukral
DAV College, Amritsar
E-mail: *sunny_thukral@yahoo.co.in*

Abstract

Companies have been collecting data for decades, and building massive warehouses for data storage. Even though this data is available, very few companies have been able to realize its actual value. The question these companies are asking is how to extract this value. For this, there are many technologies available to Soft Computing practitioners, including Artificial Neural Networks, Regression, and Decision Trees. Many practitioners are wary of Neural Networks due to their black box nature, even though they have proven themselves in many situations. This paper is an overview of artificial neural networks in computer industry with respect to soft computing.

Keywords: *Artificial Neural Network (ANN), Neural Network Topology, Advantages and Applications.*

INTRODUCTION

The study of the human brain is thousands of years old. With the advent of modern electronics, it was only natural that this would be applied in the said study, the result being development of artificial neural networks. The first step toward artificial neural networks came in 1943 when Warren McCulloch, a neurophysiologist, and a young mathematician, Walter Pitts, wrote a paper on how neurons might work. They modeled a simple neural network with electrical circuits. Neural networks, with their remarkable ability to derive meaning from complicated or imprecise data, can be used to extract patterns and detect trends that are too complex to be noticed by either humans or other computer techniques. A trained neural network can be thought of as an "expert" in the category of information it has been given to analyse.

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 (neurons) 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 be solved 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 that are more suited to an algorithmic approach like arithmetic operations, and some tasks that are more suited to neural networks. In fact, a large number of tasks require systems that use a combination of the two approaches (wherein normally a conventional computer is used to supervise the neural network) in order to perform at maximum efficiency.

WORKING OF NEURAL NETWORK

An artificial neural network (ANN), often just called a "neural network" (NN), is a mathematical model or computational model based on biological neural networks, in other words, is an emulation of biological neural system [2]. It consists of an interconnected group of artificial neurons and processes information using a connectionist approach to computation. In most cases an ANN is an adaptive system that changes its structure based on external or internal information that flows through the network during the learning phase.

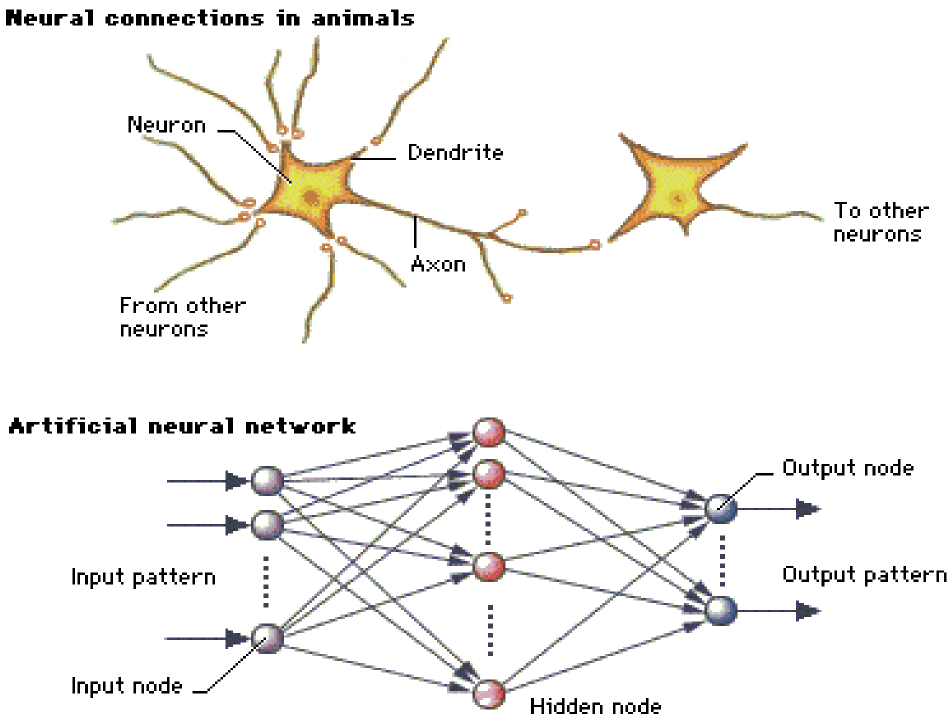

Figure 1: Representation of Artificial Neural Network

Artificial Neural Networks are relatively crude electronic models based on the neural structure of the brain.The brain basically learns from experience. It is natural proof that some problems that are beyond the scope of current computers are indeed solvable by small energy efficient packages. This brain modelling also promises a less technical way to develop machine solutions. This new approach to computing also provides a more graceful degradation during system overload than its more traditional counterparts. These biologically inspired methods of computing are thought to be the next major advancement in the computing industry. Even simple animal brains are capable of functions that are currently impossible for computers. Computers do rote things well, like keeping ledgers or performing complex math. But computers have trouble recognizing even simple patterns much less generalizing those patterns of the past into actions of the future. Now, advances in biological research promise an initial understanding of the natural thinking mechanism. This research shows that brains store information as patterns. Some of these patterns are very complicated and allow us the ability to recognize individual faces from many different angles. This process of storing information as patterns, utilizing those patterns, and then solving problems encompasses a new field in computing [3]. This field, as mentioned before, does not utilize traditional programming but involves the creation of massively parallel networks and the training of those networks to solve specific problems. This field also utilizes words very different from traditional computing, words like behave, react, self-organize, learn, generalize, and forget. Whenever we talk about a neural network, we should more popularly say ―Artificial Neural Network (ANN). ANNs are computers whose architecture is modelled after the brain. They typically consist of hundreds of simple processing units which are wired together in a complex communication network. Each unit or node is a simplified model of real neuron which sends off a new signal or fires if it receives a sufficiently strong Input signal from the other nodes to which it is connected.

TRAINING OF ARTIFICIAL NEURAL NETWORK

Once a network has been structured for a particular application, that network is ready to be trained. To start this process the initial weights are chosen randomly. Then, the training or learning begins. There are two approaches to training - supervised and unsupervised. Supervised training involves a mechanism of providing the network with the desired output either by manually "grading" the network's performance or by providing the desired outputs with the inputs. Unsupervised training is where the network has to make sense of the inputs without outside help. The vast bulk of networks utilize supervised training. Unsupervised training is used to perform some initial characterization on inputs. However, in the full blown sense of being truly self-learning, it is still just a shining promise that is not fully understood, does not completely work, and thus is relegated to the lab. A neural network has to be configured such that the application of a set of inputs produces (either 'direct' or via a relaxation process) the desired set of outputs. Various methods to set the strengths of the connections exist. One way is to set the weights explicitly, using a priori knowledge. Another way is to 'train' the neural network by feeding it teaching patterns and letting it change its weights according to some learning rule. We can categorize the learning situations as follows:

Supervised Training

In supervised training, both the inputs and the outputs are provided. The network then processes the inputs and compares its resulting outputs against the desired outputs. Errors are then propagated back through the system, causing the system to adjust the weights which control the network. The set of data which enables the training is called the "training set." During the training of a network the same set of data is processed many times as the connection weights are refined. This could be because the input data does not contain the specific information from which the desired output is derived. Networks also don't converge if there is not enough data to enable complete learning. Ideally, there should be enough data so that part of the data can be held back as a test [1]. Many layered networks with multiple nodes are capable of memorizing data. To monitor the network to determine if the system is simply memorizing its data in some non-significant way, supervised training needs to hold back a set of data to be used to test the system after it has undergone its training. If a network simply can't solve the problem, the designer then has to review the input and outputs, the number of layers, the number of elements per layer, the connections between the layers, the summation, transfer, and training functions, and even the initial weights themselves. Those changes required to create a successful network constitute a process wherein the "art" of neural networking occurs. Another part of the designer's creativity governs the rules of training. There are many laws (algorithms) used to implement the adaptive feedback required to adjust the weights during training. The most common technique is backward-error propagation, more commonly known as back-propagation. These learning techniques are explored in greater depth later in this report.

Training is not just a technique. It involves a "feel," and conscious analysis, to insure that the network is not over trained. Initially, an artificial neural network configures itself with the general statistical trends of the data. Later, it continues to "learn" about other aspects of the data which may be spurious from a general viewpoint. When finally the system has been correctly trained, and no further learning is needed, the weights can, if desired, be "frozen." In some systems this finalized network is then turned into hardware so that it can be fast. Other systems don't lock themselves in but continue to learn while in production use.

Un-supervised or Adaptive Training

The other type of training is called unsupervised training. In unsupervised training, the network is provided with inputs but not with desired outputs. The system itself must then decide what features it will use to group the input data. This is often referred to as self-organization or adaption. At the present time, unsupervised learning is not well understood. This adaption to the environment is the promise which would enable science fiction types of robots to continually learn on their own as they encounter new situations and new environments. Life is filled with situations where exact training sets do not exist. Some of these situations involve military action where new combat techniques and new weapons might be encountered. Because of this unexpected aspect to life and the human desire to be prepared, there continues to be research into, and hope for, this field. Yet, at the present time, the vast bulk of neural network work is in systems with supervised learning. Supervised learning is achieving results.

Reinforcement Learning

This type of learning may be considered as an intermediate form of the above two types of learning. Here the learning machine does some action on the environment and gets a feedback response from the environment. The learning system grades its action good (rewarding) or bad (punishable) based on the environmental response and accordingly adjusts its parameters.

NEURAL NETWORK ARCHITECTURE

Neural networks essentially comprise of three pieces: the architecture or model; the learning algorithm; and the activation functions. Neural networks are programmed or "trained" to ". . . store, recognize, and associatively retrieve patterns or database entries; to solve combinatorial optimization problems; to filter noise from measurement data; to control ill-defined problems; in summary, to estimate sampled functions when we do not know the form of the functions." It is precisely these two abilities (pattern recognition and function estimation) which make artificial neural networks (ANN) so prevalent a utility in data mining [1]. As data sets grow to massive sizes, the need for automated processing becomes clear. With their "model-free"

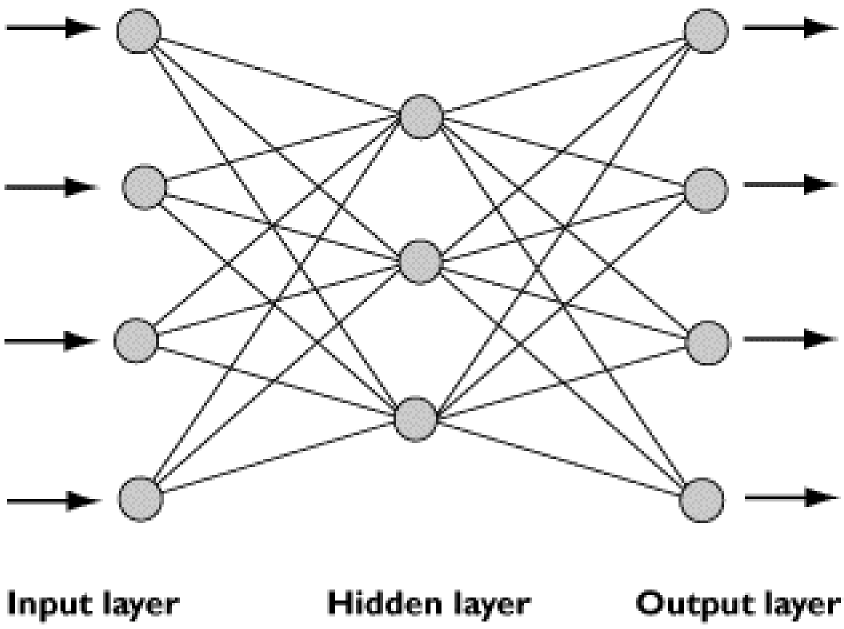

Figure 2: Feed Forward Neural Network Model

estimators and their dual nature, neural networks serve data mining in a myriad of ways.

One of the simplest feed forward neural networks (FFNN), such as in Figure 2, consists of three layers: an input layer, hidden layer and output layer. In each layer there are one or more processing elements (PEs). PEs are meant to simulate the neurons in the brain and this is why they are often referred to as neurons or nodes. A PE receives inputs from either the outside world or the previous layer. There are connections between the PEs in each layer that have a weight (parameter) associated with them [2]. This weight is adjusted during training. Information only travels in the forward direction through the network - there are no feedback loops.

The simplified process for training a FFNN is as follows:

- Input data is presented to the network and propagated through the network until it reaches the output layer. This forward process produces a predicted output.

- The predicted output is subtracted from the actual output and an error value for the networks is calculated.

- The neural network then uses supervised learning, which in most cases is back propagation, to train the network. Back propagation is a learning algorithm for adjusting the weights. It starts with the weights between the output layer PE's and the last hidden layer PE's and works backwards through the network.

- Once back propagation has finished, the forward process starts again, and this cycle is continued until the error between predicted and actual outputs is minimized.

APPLICATIONS OF NEURAL NETWORK

The various real time application of Artificial Neural Network are as follows:

- Function approximation, or regression analysis, including time series prediction and modelling.

- Call control - answer an incoming call (speaker-ON) with a wave of the hand while driving.

- Classification, including pattern and sequence recognition, novelty detection and sequential decision making.

- Skip tracks or control volume on your media player using simple hand motions- lean back, and with no need to shift to the device- control what you watch/ listen to.

- Data processing, including filtering, clustering, blind signal separation and compression.

- Scroll Web Pages, or within an eBook with simple left and right hand gestures, this is ideal when touching the device is a barrier, such as when hands are wet, gloves are dirty, etc.

- Application areas of ANNs include system identification and control (vehicle control, process control), game-playing and decision making (backgammon, chess, racing), pattern recognition (radar systems, face identification, object recognition, etc.), sequence recognition (gesture, speech, handwritten text recognition), medical diagnosis, financial applications, data mining (or knowledge discovery in databases, "KDD").

- Another interesting use case is when using the Smartphone as a media hub; a user can dock the device to the TV and watch content from the device - while controlling the content in a touch-free manner from afar.

ADVANTAGES

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

- An ANN can create its own organisation or representation of the information it receives during learning time.

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

- Pattern recognition is a powerful technique for harnessing the information in the data and generalizing it. Neural nets learn to recognize the patterns which exist in the data set.

- The system is developed through learning rather than programming. Neural nets teach themselves the patterns in the data, freeing the analyst for more interesting work.

- Neural networks are flexible in a changing environment. Although neural networks may take some time to learn a sudden drastic change, they are excellent at adapting to constantly changing information.

- Neural networks can build informative models whenever conventional approaches fail. As neural networks can handle very complex interactions, they can easily model data which is difficult to model with traditional approaches such as inferential statistics or programming logic.

- Performance of neural networks is at least as good as classical statistical modeling, and better on most problems. The neural networks build models that are more reflective of the structure of the data in significantly lesser time.

CONCLUSION

In this paper we discussed about the artificial neural network, its working along with training phases of an Artificial Neural Network. There are various advantages of Artificial Neural Network over conventional approaches. Today, discussions on neural network are occurring everywhere. Their prospects seem very bright as nature itself is the proof that this kind of structure works. Yet, its future, indeed the very key to the whole technology, lies in hardware development. Currently the advances in neural network development are proving that the principal is valid.

REFERENCES

1. Bradshaw, J.A., Carden, K.J., Riordan, D., 1991. Ecological ―Applications Using a Novel Expert System Shell‖. Comp. Appl. Biosci. 7, 79–83.
2. Lippmann, R.P., 1987. An introduction to computing with neural nets. IEEE Accost. Speech Signal Process. Mag., April: 4-22.
3. N. Murata, S. Yoshizawa, and S. Amari, ―Learning curves, model selection and complexity of neural networks,‖ in Advances in Neural Information Processing Systems 5, S. Jose Hanson, J. D. Cowan, and C. Lee Giles, ed. San Mateo, CA: Morgan Kaufmann, 1993, pp. 607-614