Computer Engineering and Intelligent Systems
ISSN 2222-1719 (Paper) ISSN 2222-2863 (Online)
Vol.7, No.9, 2016 www.iiste.org

A Study on Neural Network Architectures

K Ishthaq Ahamed
Faculty, Department of CSE, G. Pulla Reddy Engineering College, Kurnool, AP, India

Dr. Shaheda Akthar
Faculty of Computer Science, Govt. College for Women, Guntur, AP, India

Abstract
With the growing emphasis on autonomy, intelligence and an increased amount of information required by businesses, traditional processing technology can only cope through faster hardware with more complex customized software. The traditional computation techniques of programming were not capable enough to solve "hard" problems like pattern recognition, prediction, compression, optimization, classification and machine learning. In order to solve such problems, an interest towards developing intelligent computation systems became stronger. To develop such intelligent systems, innumerable advances have been made by the researchers. An artificial neural network is a data processing system consisting of a huge number of simple, highly interconnected processing elements in an architecture inspired by the structure of the cerebral cortex portion of the brain. Hence, neural networks are often capable of doing things which humans or animals do well but which conventional computers often do poorly. These artificial neurons are pigeonholed on the basis of architecture, training or learning method and activation function. The neural network architecture is the arrangement of neurons to form layers and connections scheme formed in between and within the layers. Neural network architectures are broadly classified into feed-forward and feedback architectures that further contain single and multiple layers. The feed-forward networks provide a unidirectional signal flow whereas in the feedback networks the signals can flow in both the directions. These neural network architectures are trained through various learning algorithms for producing most efficient solutions to computation problems. In this paper, we present neural network architectures that play a crucial role in modelling the intelligent systems.
Keywords: Artificial Neural Network, feed-forward networks, feedback networks .

1. Introduction

The human brain is a composite computing system capable of thinking, remembering, and solving problems. There have been a number of attempts to emulate the brain functions with a computer model, and generally these have involved the simulation of a network of neurons, commonly called neural networks. The brain contains approximately 100 billion neurons that are densely interconnected with one thousand to ten thousand connections per neuron.

A neuron is the fundamental cellular unit of the brain's nervous system. It is a simple processing unit (soma) that receives and combines signals from other neurons through input paths called dendrites which contain synaptic junctions. If the combined signal from all the dendrites is strong enough, the neuron "fires", producing an output signal along a path called the axon. The axon splits up and connects to thousands of dendrites (input paths) of other neurons through synapses (junctions containing a neurotransmitter fluid that controls the flow of electrical signals) located in the dendrites. Transmission of the signals across the synapses is electro-chemical in nature, and the magnitudes of the signals depend upon the synaptic strengths of the synaptic junctions. The strength or conductance (the inverse of resistance) of a synaptic junction is modified as the brain "learns". In other words, the synapses are the basic "memory units" of the brain.

The computer simulation of this brain function usually takes the form of artificial neural systems which consists of many artificial neurons, usually called processing elements. These processing elements are analogous to the neuron in that they have many inputs (dendrites) and combine (sum up) the values of the inputs. This sum is then subjected to a nonlinear filter usually called a transfer function, which is usually a threshold function or a bias in which output signals are generated only if the output exceeds the threshold value.

The output of a processing element (axon) branches out and becomes the input to many other processing elements. These signals pass through connection weights (synaptic junctions) that correspond to the synaptic strength of the neural connections. The input signals to a processing element are modified by the connection weights prior to being summed by the processing element. There is an analogy between a processing element and an operational amplifier in an analog computer in which many inputs are summed. The potentiometer settings on the amplifier inputs correspond to the connection weights and the output of the operational amplifier goes through some sort of nonlinear function generator.

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 [1]. They have emerged in the past few years as an area of unusual opportunity for research, development and application to a variety of real world problems. Neural network consists of processing elements that form networks with weighted functions for every input. These elements are generally arranged into a sequence of layers with several connections between them. The structure of neural networks has three types of layers: an **input layer** that receives data from external sources, **hidden layer** that performs computation on the basis of function provided, and an **output layer** that generates output based on the input provided.

An artificial neural network (ANN) is a well organized information processing system that has characteristics similar to that of a biological neural network. It consists of largely interlinked processing elements known as units or nodes or neurons. These elements usually run in parallel and are connected to each other using connection links. The connection links have weights that contain data related to input signal. Neurons use this data to solve a specific problem. The behaviour of a neural network is described by their capability of learning, analysing and generalizing training data having resemblance to that of a human brain. These networks consist of largely interconnected processing neurons that are inherently parallel in nature [2].
The below figure represents the difference between a biological and artificial neuron:

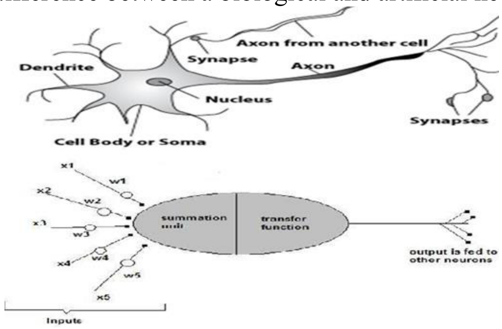

Figure 1: A biological versus artificial neuron

From Fig.1 it is clear that, a biological neuron accepts the inputs through dendrites, processes them in the cell body. They convert the processed inputs into outputs using axons. The electrochemical contact between the neurons is provided by the synapses. Whereas, an artificial neuron takes the Inputs through input neurons, calculates the summation of the inputs and compares them with the activation function being used during the learning stage.

2. Learning methods of ANN:

The two major characteristics which differentiate neural networks from artificial intelligence and traditional computing are learning by example and distributed associative memory [2].

2.1. Learning by example:

The neural networks have the ability to learn by an example. They cannot be organized to perform a particular task. The examples that are used for learning should be selected carefully such that it doesn't lead to wastage of time or incorrect working of the network. Instead of programming, the system is developed through learning.

2.2 Distributed associative memory:

The neural network structure can be largely distributed and associated in parallel. A unit of knowledge is distributed across all the weighted links in the network. Upon training a network it is provided with an input, the network then chooses the nearest match to that input in its memory and generates an output that is similar to the full output.

3. Characteristics of ANN

These characteristics enable Artificial Neural Networks to generate solutions for typical problems that are difficult to manage by the traditional ways of problem solving.

3.1 The Network Structure:

The Network Structure of ANN should be simple and easy. There are mainly two types of arrangements recurrent and non-recurrent structure. The Recurrent Structure is also known as Auto associative or Feedback Network and the Non Recurrent Structure is also known as Associative or feed forward Network. In feed forward Network, the signal travel in one direction only but in Feedback Network, the signal travel in both directions by introducing loops in the net.

3.2 Parallel Processing Ability:

ANN is introduced to extend the concept of parallel processing in the computer field. Parallel Processing is done by the human body in human neurons are very difficult but by using basic and simple parallel processing methods can be implement it in ANN like Matrix and some matrix calculations.

3.3 Distributed Memory:

ANN is very massive system so single place memory or centralized memory cannot satisfy the requirement of ANN system so in this state we need to store data in weight matrix which is form of long term memory because information is stored as patterns throughout the network structure.

3.4 Fault Tolerance Ability:

ANN is a very composite system so it is important that it should be a fault tolerant. Because if any part becomes fail it will not disturb the system as much but if the all parts fails at the same time the system will fails completely.

3.5 Collective Solution:

ANN is an interconnected system the output of a system is a collective output of various input so the result is summation of all the outputs which comes after processing various inputs. The Partial answer is worthless for any user in the ANN System.

3.6 Learning Ability:

In ANN most of the learning rules are used to develop models of processes, while adopting the network to the changing environment and discovering useful knowledge. These Learning methods are Supervised, Unsupervised and Reinforcement Learning.

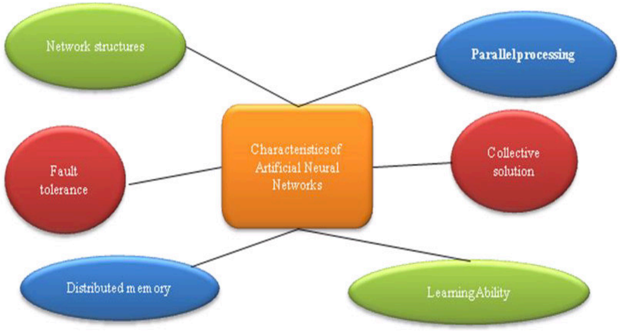

Figure 2: Characteristics of Neural Networks

4. Neural Network Models

4.1 Neuron:

The first artificial neuron was the Threshold Logic Unit (TLU), or Linear Threshold Unit, first proposed by Warren McCulloch and Walter Pitts in 1943. The model was specifically targeted as a computational model of the "nerve net" in the brain. The model describes a neuron as binary processing unit. The neuron is either fired and not fired its output signal and is controlled by a threshold logic or activation function. The model neuron, like its biological counterpart, has number of inputs and an output. At each input to the neuron, there is a weight. The weight acts like an input synaptic resister [3]. To calculate if a neuron will fire output signal, first the sum of multiplication between each input and the input's synaptic weight, is calculated. The activation function, then calculated using the sum value from the first step. The process is illustrated in the figure 3.

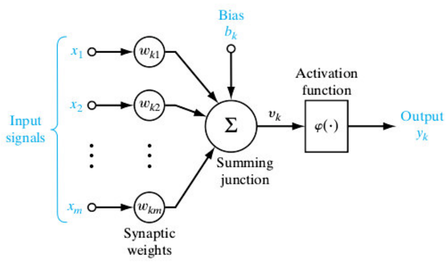

Figure 3: Neuron

A single-input neuron is shown in Fig. 3. The scalar input x is multiplied by the scalar weight w to form w*p*, one of the terms that is sent to the summer. The other input, 1, is multiplied by a bias *b* and then passed to the summer. The summer output n often referred to as the net input, goes into a transfer function φ which produces the scalar neuron output a (sometimes "activation function" is used rather than transfer function and offset rather than bias).

From Fig. 3, both w and b are both adjustable scalar parameters of the neuron. Typically the transfer function is chosen by the designer and then the parameters w and b will be adjusted by some learning rule so that the neuron input/output relationship meet some specific goal. The transfer function in Fig.3 may be a linear or nonlinear function of n. A particular transfer function is chosen to satisfy some specification of the problem that the neuron is attempting to solve.

The below Fig.2 shows architecture of single-input neuron:

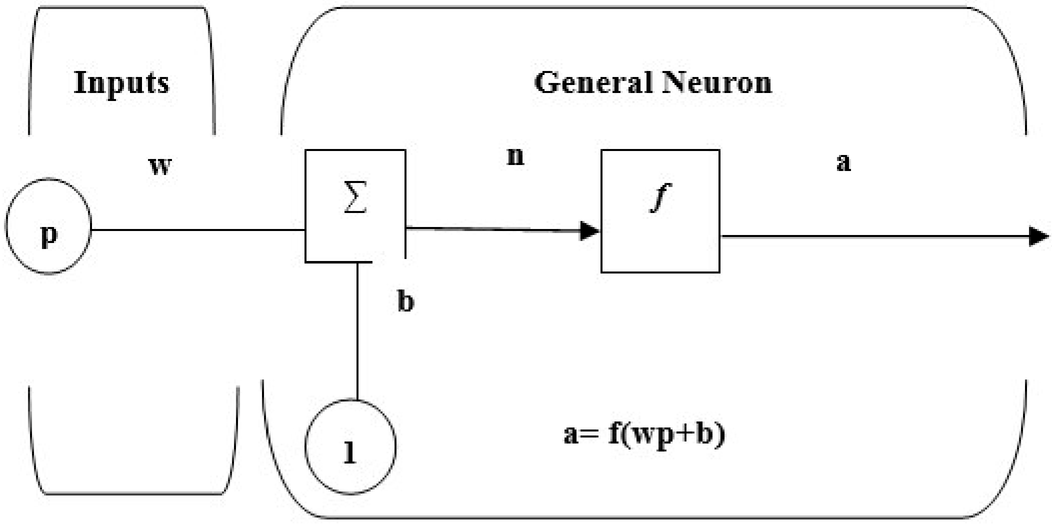

Figure 4: Single-Input Neuron

If we relate the artificial neuron structure with the biological neuron, the weight **w** is similar to strength of synapse, summation represents the cell body, the signal on axon is represented by neuron output **a** and the activation function.

4.2 Perceptron:

An artificial neuron called a perceptron. Perceptron were developed in the 1950s and 1960s by the scientist Frank Rosenblatt, inspired by earlier work by Warren McCulloch and Walter Pitts. Today, it's more common to use other models of artificial neurons and in much modern work on neural networks, the main neuron model used is one called the sigmoid neuron [4].

A perceptron takes several binary inputs, x1,x2,…, and produces a single binary output:

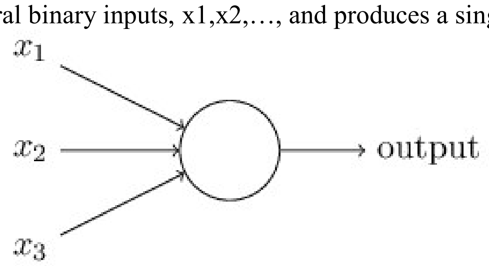

Figure 5: Perceptron

Figure 5 shows the perceptron has three inputs, x1,x2,x3. In general it could have more or fewer inputs. Rosenblatt proposed a simple rule to compute the output. He introduced weights, w1,w2,… real numbers expressing the importance of the respective inputs to the output. The neuron's output, 0 or 1, is determined by whether the weighted sum $\sum_{i=0}^{n} x_i w_{ij}$ less than or greater than some threshold value. Just like the weights, the threshold is a real number which is a parameter of the neuron.

To put it in more precise algebraic terms:

$$\text{output} = \begin{cases} 1 \text{ if threshold} \geq 0 \\ 0 \text{ if threshold} < 0 \end{cases} \quad \ldots\ldots(1)$$

4.3 Single-Layer Perceptrons

The single-layer perceptrons shown in figure 6 was among the first and simplest learning machines that are trainable. In Haykin's book (1999), perceptron denotes the class of two-layer feed forward networks, 1) whose first-layer units have fixed function with fixed connection weights from the inputs, and 2) whose connection weights linking this first layer to the second layer of outputs are learnable. The model of training in perceptrons is supervisory, because the steps in the algorithm involve the comparison of actual outputs with desired outputs associated with the set of training patterns. An input layer of source nodes can project onto an output layer of neurons, but not vice versa [5]. The LMS algorithm can be used in the supervisory training.

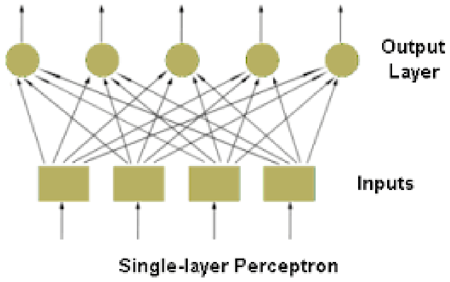

Figure 6 : Single Layer Perceptron

4.4 Multi-Layer Feed-forward Network

The Multi-Layer feed-forward neural network is also known as multilayer perceptron and the most popular network architecture in use today. As mentioned earlier, the leftmost layer in this network is called the input layer, and the neurons within the layer are called *input neurons*. The rightmost or *output* layer contains the *output neurons*, or, as in this case, a single output neuron. The middle layer is called a *hidden layer*, since the neurons in this layer are neither inputs nor outputs.

The units each perform a biased weighted sum of their inputs and pass this activation level through an activation function to produce their output, and the units are arranged in a layered feed forward topology. The network thus has a simple interpretation as a form of input-output model, with the weights and thresholds (biases) the free parameters of the model. [6]

The following Fig.7 represents Multi-layer feed-forward architecture:

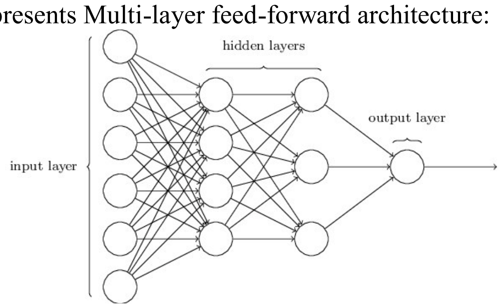

Figure 7: Multi-Layer Feed-forward Network (Multi-Layer Perceptron)

The layer consists of summing units, activation functions, bias **b,** weight matrix **W**, and output vector. Each component of the input **P** is connected to each neuron through weight matrix **W**. Each neuron has an activation function ***f***, bias **b** and an output. The number of inputs to a layer can be different from the number of neurons. The neurons in a layer can have different activation functions by combining two of networks each network can generate some outputs. The input components enter the network through the weight matrix **W**.

While the design of the input and output layers of a neural network is often straightforward, there can be an art to the design of the hidden layers. In particular, it's not possible to sum up the design process for the hidden layers with a few simple rules of thumb. Instead, neural networks researchers have developed many design heuristics for the hidden layers, which help people get the behavior they want out of their nets. For example, such heuristics can be used to help determine how to trade off the number of hidden layers against the time required to train the network [7].

Neural networks where the output from one layer is used as input to the next layer are called *feedforward* neural networks. This means there are no loops in the network - information is always fed forward, never fed back. If we did have loops, we'd end up with situations where the input to the σ function depended on the output. That'd be hard to make sense of, and so we don't allow such loops.

4.5 Multi-Layer Feedback network: (Recurrent Network)

On the other hand, artificial neural networks in which feedback loops are possible. These models are called multi-layer feedback networks or recurrent neural networks. The idea in these models is to have neurons which fire for some limited duration of time, before becoming quiescent. That firing can stimulate other neurons, which may fire a little while later, also for a limited duration. That causes still more neurons to fire, and so over time it get a cascade of neurons firing. Loops don't cause problems in such a model, since a neuron's output only affects its input at some later time, not instantaneously.

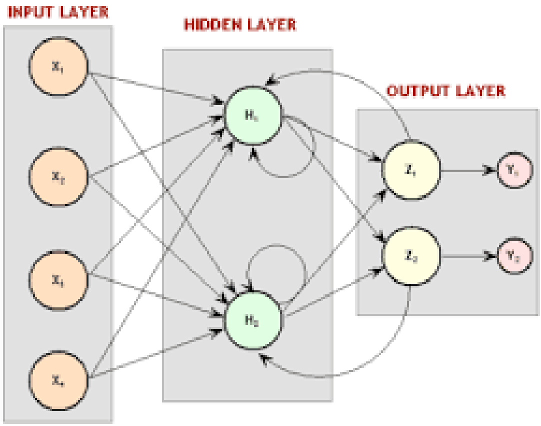

Figure 8: Multi-Layer feedback Network (Recurrent Network)

Recurrent neural networks has very rich temporal and spatial behaviors, such as stable and unstable fixed points and limit cycles, and chaotic behaviors. These behaviors can be utilized to model certain cognitive functions, such as associative memory, unsupervised learning, self-organizing maps, and temporal reasoning [8].

1) Symmetric Recurrent Network: In symmetric recurrent network, the connections are symmetric, that is, the connection weights from unit i to unit j and from unit j to unit i are identical for all i and j. The widely known Hopfield networks, are a kind of symmetric recurrent networks.[6]

***2)* Asymmetric Recurrent Network:** The dynamic behavior of asymmetric networks includes *limit cycles* and *chaos*, and these networks are capable of storing or generating temporal sequences of spatial patterns. Chaos in a recurrent neural network is characterized by a time evolution that progresses through a set of distorted patterns in a notably irregular manner. In this network the initial conditions are supplied by input vector **p.**

3) Fully Recurrent Network: The main example of implementation of feedback is the classical fully recurrent neural network, i.e. a single layer of neurons fully interconnected with each other or several such layers [10]. They are very general architectures which can model a large class of dynamical systems, but on specific problems simpler dynamic neural networks which make use of available prior knowledge can be better.

4) Locally Recurrent Network: The newest approach to the temporal processing by neural networks is realized by Locally Recurrent Neural Networks (LRNNs) or Local Feedback Multi-Layer Networks (LF-MLN).

Recurrent neural nets have been less influential than feedforward networks, in part because the learning algorithms for recurrent nets are (at least to date) less powerful. But recurrent networks are still extremely interesting. They're much closer in spirit to how our brains work than feedforward networks. And it's possible that recurrent networks can solve important problems which can only be solved with great difficulty by feedforward networks. However, to limit our scope, in this book we're going to concentrate on the more widely-used feedforward networks [9].

Conclusion

Thus, from this paper it is evident that the neural network architectures play a very important role in modelling the intelligent systems. The feed-forward and feedback architectures make use of various learning algorithms and paradigms for obtaining the outputs. They act as a backbone of the entire learning process which provides the end user with the desired result. For this purpose the neural network architectures are trained through various learning algorithms for producing most efficient solutions to computation problems.

References

1. Koushal Kumar, Gour Sundar Mitra Thakur, 2012. Advanced Applications of Neural Networks and Artificial Intelligence: A Review. I.J. Information Technology and Computer Science, 2012, 6, 57-58.
2. Robert E. Uhrig, “Introduction to Artificial Neural Networks”, IEEE Transactions on Industrial Electronics, Control, and Instrumentation, vol. 1, pp. 33-37, 1995.
3. A. Jain, J. Mao, and K. Mohiuddin, “Artificial neural networks: A tutorial,” Computer, vol. 29, no. 3, pp.31–44, Mar. 1996.
4. S.Haykin, Neural Networks A Comprehensive Foundation, 2nd Edition, Prentice Hall, 2000.
5. G. E. Hinton, S. Osindero, and Y. Teh. A fast learning algorithm for deep belief nets. Neural Computation, 18:1527–1554, 2006.
6. Walter H. Delashmit and Michael T. Manry, “Recent Developments in Multilayer Perceptron Neural Networks”, Proceedings of the 7th Annual Memphis Area Engineering and Science Conference, MAESC,

2005, pp. 1-3.
7. L. Medsker and L. Jain, Recurrent Neural Networks Design and Applications. London, U.K.: CRC Press, 2001.
8. A.B.Karthick Anand Babu, 2012. Design and Development of Artificial Neural Network Based Tamil Unicode Symbols Identification System. IJCSI International Journal of Computer Science Issues, Vol. 9 Issue 1, No 2, January 2012.
9. Humayun Karim Sulehria, Hopfield Neural Networks—A Survey. Proceedings of the 6th WSEAS Int. Conf. on Artificial Intelligence, Knowledge Engineering and Data Bases, Corfu Island, Greece, February 16-19, 2007.
10. K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators", Neural Networks, vol. 2, no. 5, pp. 359–366, 1989.