IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

AN ELEMENTARY APPROACH OF ARTIFICIAL NEURAL NETWORK

[1]Aashi Tyagi, [2]Vinita Sharma

[1]M.Tech Scholar, [2]HOD

[1]Computer Science & Engineering,

[1]ABSSIT, Meerut, India

Abstract: In this paper, a brief review of Artificial Neural Network or ANN in the field of Artificial Intelligence is represented. As artificial neural structure is inspired from the biological neuron that is responsible for transformation of signals in human brain, is explained in this paper. ANN is the network models that make machines more intelligent to act more likely or equal to human beings. The most common choice of leading ANN is the decision-making quality. They are adaptive learning models that work on prior data sets. Biological neuron model plays a vital role in the working guidance of ANNs. The approach of these networks is heading more day by day.

Index Terms **- Artificial Intelligence (AI), Artificial Neural Network (ANN), Activation Function (AF), information processing system, neurons, back-propagation algorithm, learning rules.**

I. INTRODUCTION

Artificial Neural Network is defined as an information processing system that is going to implement in machines so that they perform computational tasks and act like human beings. Warren McCulloch and Walter Pits was developed the first artificial neural network in 1943. Artificial neural networks are extensive parallel systems that are interconnected with multiple processors.
In general context, network refers to a group of multiple entities connecting together for the purpose of communication similarly, artificial neural network looks like. It is a network consists up of multiple artificial neurons connected with each other.
Artificial neural network comes under the field of AI. Artificial intelligence is the branch of computer science that accentuates intelligent system of machines. These intelligent systems are designed in such a way that they imitate human behavior to respond smartly, such as decision-making, visual perception and more on by using data sets and learning algorithms.
They are commonly used for classification, regression, prediction accuracy. Non-linear functions are used nowadays, non-linearity i.e. even easier than linear function.

II. PROPOSED BIOLOGICAL MODEL

The phenomenon of artificial neural network is inspired from biological neurons.

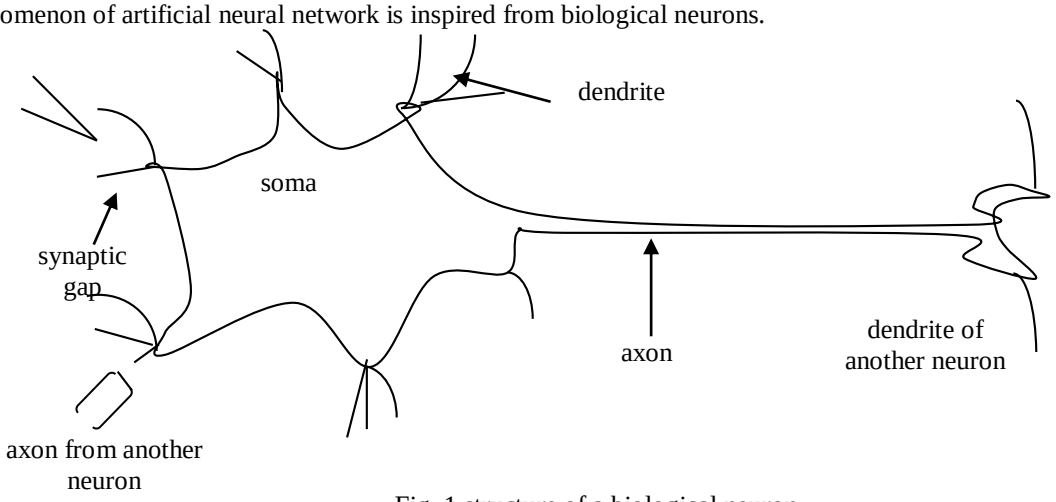


Fig. 1 structure of a biological neuron

According to human nervous system, neurons are the basic building block of the system. They are responsible for transportation of information in human brain. The basic components of a neuron are soma, dendrite, axon, synaptic gap as shown in figure 1. Soma refers to the actual cell body. Dendrites are responsible for combining the input signals receives from many other connected neurons. Furthermore they will pass this information to main cell body, i.e. soma. Axon receives the information from cell body to process at another end of neuron structure, where synaptic gap is the mediocre point in between axon and dendrite of a neuron structure. It may allow them to communicate with more others.

III. LAYERED ARCHITECTURE OF NEURAL NETWORK

The main objective to design a neural network is to generate an error free output by learning that tending towards improving the solutions and proposed a more accurate system.

Artificial neural network design is a layered architecture that contains number of processing nodes settled in the different layers of network structure, interconnected with each other. In technical terms, neurons can be treated as nodes.

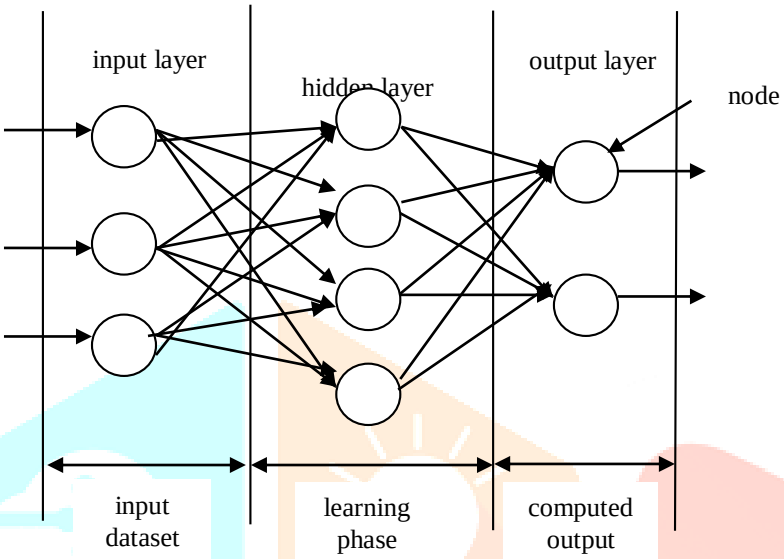


Fig. 2 layered neural network structure having one hidden layer

It is called layered due to presence of multiple layers basically input, hidden and output layers. Each and every neural network structure must have one input and one output layer. On the same hand, number of hidden layers may vary according to the type of problem.

Layers of neural network are mainly classified into three categories:-

1) Input layer: It is responsible for getting input as maintaining the data set values, and passes these values to next hidden layer through weight vectors.
2) Hidden layer: It is responsible for processing of data based on chosen algorithmic approach. Weight vector plays an important role in traversing of data as they represent the weightage of that particular input in computing the desired output.
3) Output layer: It is used to represent the output of the network design. These computed output values are further compared with threshold values to check out the error rate. This layer holds either true or false value, as it refers to decision-taking one.

Similar working of layered architecture inspired by biological neuron model: -

Input layer behaves like dendrites, as it combines and further responsible for sending the input values. For actual computation of network elements, Hidden layer is effectively responsible. It is designed by keeping the cell body in mind. Output layer works like axons with synaptic gaps. Weight vector, i.e. synaptic gaps in terms of biological neuron model actually allows the interconnection between neurons. Elements of a single neuron are:-

- Input Vector: the input value for that particular neuron.
- Weight vector: respective weight value that is associated with that particular neuron for interconnection.

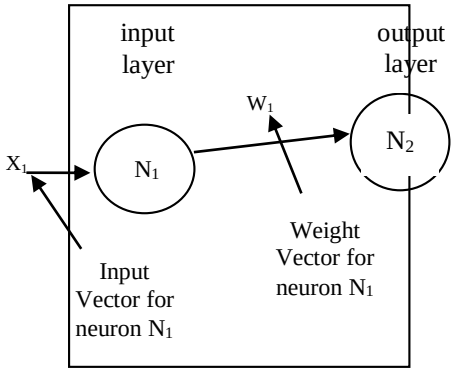
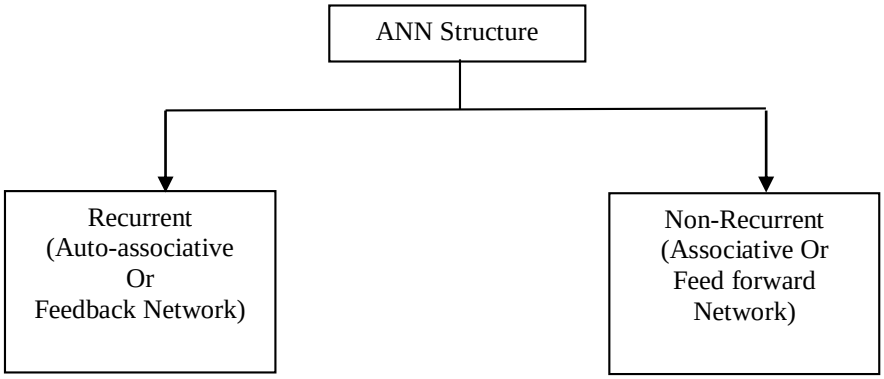


Fig. 3 Representation of a single neuron with respective input vector and weight vector

Figure 3 represents the input vector X_1 and weight vector W_1 for neuron N_1. Weight vector W_1 is responsible for transfer of this input vector further towards the neurons of next layer. Neural network that contains only one input and one output layer is simple referred as perceptron, single layer perceptron.

IV. NEURAL NETWORK STRUCTURE

Artificial neural network has two types of structures:-



1) Recurrent Structure:

Recurrent Structure referred as Auto associative or Feedback Network. In this, data travels in both directions by introducing loops in the network. The processing nodes also contain the feedback of prior data sets. It works with variable length of inputs/outputs. They are called recurrent as they work in directed cycle and because of performing computation in sequential manner.
Example: - competitive networks, Hopfield network etc.

2) Non-Recurrent Structure:

Non-Recurrent Structure is come to known as Associative or Feed forward Network. In this, data travels in one direction only. No loops are there, data is just forwarding towards the next layer with no option of backtracking. Non-recurrent network consider fixed length of input/output values.
Examples: - radial basis function nets, single and multi-layer perceptron, etc.

V. ACTIVATION FUNCTION

Activation function exhibits a great variety and has a strong effect over the behavior of neural network. It may also know as squashing function as it squashes the permissible amplitude range of output signal to some finite value or limiting the amplitude of the output of a neuron. It is basically used to decide whether a neuron is fired (activated) or not. They have the capability to improve the learning of the patterns in data. They are capable of justifying their use in the hidden layers of the neural networks. When the AF is placed after the hidden layers, it converts the learned linear mappings into non-linear forms for propagation.
Process of transmission of data may have two states: - excitatory state and inhibitory state. Inhibitory state is one that inhibits the neuron, i.e. gets down and excitatory state refers to firing of neuron, i.e. taken up. Activation functions are used to control the outputs of neural networks by computing the respective inputs and biases.

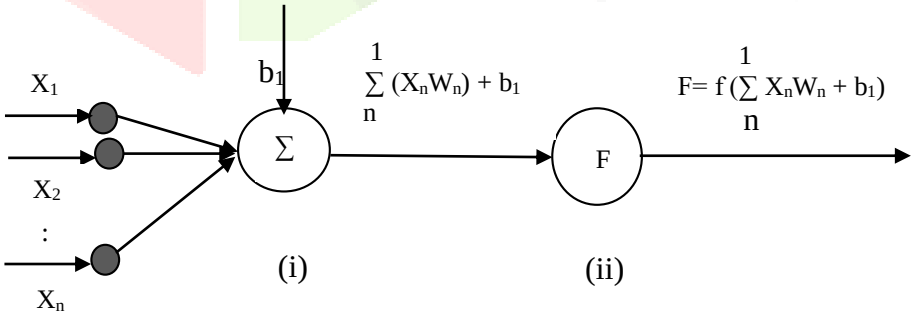


Fig. 4 summation (i) and transformation (ii) functions of an artificial neuron

From Figure 4, once the input vectors and corresponding weight vectors are assigned to neurons. Summation function (i) is taken up the sum of product of inputs with respective weights. Bias (b_1) is a constant value that is used to set activation function. Transformation function (ii) is the activation function that transforms the weighted sum to the output. Activation functions such as binary step function, tanH, ReLU, sigmoid and so on are examples of transformation functions.

VI. TRAINING NEURAL NETWORK

Working with neural network may include three basic steps:-

i) Constructing the neural network, getting the input vectors and respective weight vectors for computation task with their corresponding test values.
ii) Train the network by iteration process using prior data sets, i.e. known outputs.
iii) Trained network is used for specific purposes, as it may also learn by itself.

Learning rules are basically implemented to improve network's performance. There are three specified learning rules to train the neural network models: -

1) Supervised Learning:

In supervised learning, input and output vectors are well known, i.e. they have prior datasets. It works with training data. During training of an neural network under this learning rule, output is produced after processing of defined input vectors and then compared with the target output vector in order to achieve optimal performance. Respective weight vectors are then adjusted if there is some error found while achieving the target output.

Back-propagation is an algorithm that is used to train neural networks. Mode of learning is broadly categorized into batch mode and sequential mode. Batch mode of learning offers the processing of data sets after traversing of all patterns. Sequential mode of learning is an iterative process. It allows working on pattern by pattern basis, i.e. finding out the errors while reviewing the pattern and sorts them first before going to the next pattern. Most of the network works with sequential mode. Reviewing the network may include passing the input data through network and calculating the error factor as every neuron has made their calculations. After forwarding the first phase, sorting of error is taken place before further pass of pattern by back propagating the network. The weights of the connected nodes are updated by learning algorithms until error rate becomes negotiable. Gradient Descent is one of the steps of this algorithm, to update weight values for the respective network. The process of weight updation repeats for a number of times till error rate becomes lesser or negotiable.

Computing the total error rate: -

Error factor is calculated at output layer.

$$E_{total} = \sum \frac{1}{2} (\text{Target} - \text{Observed})^2 \qquad (6.1)$$

This is the formula to calculate the error factor, where:

"Target" is the expected/actual output value, i.e. known prior and "Observed" is the computed output value of the network, i.e. being calculated. If the calculated output is quite varying from the target output then we have to back propagate the network structure.

It may be further categorized into classification and regression. Examples of supervised learning algorithms are linear regression, nearest neighbor, decision tree, gaussian naive bayes and so on.

2) Unsupervised Learning:

In unsupervised learning, input data does not have labels associated with it, there is no assurance of clear labeled datasets. Unlike supervised learning rule, there is no target output vector for training of network model. It works with unstructured data.

It may be further categorized into association and clustering. Examples of unsupervised learning algorithms are K-means clustering, deep adversarial network and so on.

3) Reinforcement Learning:

In reinforcement learning, there is no training datasets are available. Network model has to learn from its existing experience and make decisions on variable time. It may allow network models to achieve optimal performance and automatically acquire the ideal properties.

It may be further categorized into positive and negative reinforcement. Examples of reinforcement learning algorithms are Q-learning, deep Q network and so on.

VII. APPLICATION AREAS

Neural networks are used for various purposes. A variety of objectives are there while working with neural networks:-

- To determine a specific class that an input belongs to and so on.
- To achieve more accurate system by performing fast.
- To obtain a cheaper solution as compared to the conventional one.
- They are used to learn real-valued target functions.

Application areas of neural networks are object recognition, spell checking, computer software, classification, clustering, phrase detection, medical diagnosis, marketing, space and aeronautics' etc.

VIII. CONCLUSION AND FUTURE WORK

Computer science make machines/systems more intelligent by implementing learning rules that further make them respond and think like human brains. Artificial neural networks are capable of robust performance. They are adaptive learning models that work on prior data sets. These intelligent systems are become more powerful and essential in the world of today. The major key areas of artificial neural network to work on in future are development of network process, training of neural network with large data sets.

REFERENCES

[1] Volume 2, Issue 10, October 2012 ISSN: 2277 128X "A Comprehensive Study of Artificial Neural Networks" Vidushi Sharma Sachin Rai Anurag Dev *MTech, GGSIPU MCA, GGSIPU MCA, and GSSIPU*

[2] International Journal of Scientific Research, Volume 3, Issue 6, June-2012 ISSN 2229-5518 "Backpropagation Algorithm: An artificial Neural Network Approach for Pattern Recognition" by Dr. Rama Kishore, Taranjit Kaur

[3] S. Haykin. "Neural Networks, a Comprehensive foundation", 2nd edition. Prentice Hall, 1999.

[4] Yeremia, Hendy, et al. "Genetic algorithm and neural network for optical character recognition." Journal of Computer Science 9.11 (2013): 1435.

[5] International Journal of Advanced Research in Computer Science and Software Engineering Volume 3, Issue 8, August 2013 ISSN: 2277 128X.

[6] Biological Foundations of Psychology "Introducing the neuron" Lumen, Boundless psychology

[7] Lynne E. Parker [Material in these notes was gleaned from various sources, including E. Alpaydin's book Introduction to Machine Learning, MIT Press, 2004; and T. Mitchell's book Machine Learning, McGraw Hill, 1997.]