International Journal of Engineering and Applied Sciences (IJEAS)
ISSN: 2394-3661, Volume-4, Issue-5, May 2017

Artificial Neural Networks

Usha Kumari, Dr. Gaurav Aggarwal

Abstract— **The explanation and demonstration of the several types of neural networks are given, applications of neural networks like ANNs in medicine are described, and the detailed historical backgrounds are provided. The relationship between the artificial and the actual thing is also thoroughly investigated and explained. Ultimately, the involved mathematical models are shown and explained.**

Index Terms—**Neural, network, ANN., Neurocomputing**

I. INTRODUCTION

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 noble structure of the information processing system. It is composed of a large number of highly interconnected processing elements (neurons) 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 neurons. This is applicable in ANNs as well.

II. HISTORY

First Attempt: The research article of McCulloch and Pitts which was published in 1943 was taken as the beginning of Neurocomputing. It demonstrated that even simple types of neural networks can perform any arithmetic or logical function. It was widely read and had massive influence. Other researchers, Norbert Wiener and von Neumann, wrote a book and research paper in which they suggested that the research into the design of brain-like or brain-inspired computers might be interesting. In 1949 Hebb wrote a book entitled 'The Organization of Behaviour' which introduced the idea that classical psychological conditioning is ubiquitous in animals as it is a property of individual neurons. Not that the idea was exclusive, but Hebb took it further than anyone before him had by proposing a specific learning law for the synapses of neurons. Hebb then used this learning law to build a qualitative explanation of some experimental results from psychology. Even though there were many other people examining the issues regarding the neurocomputing in the 1940s and early 1950s, their work was more like setting the stage for later developments than actually causing those developments. The construction of first neurocomputer (the Snark) by Marvin Minsky in 1951 happened in this era. The Snark did operate successfully from a technical stand point but it never actually carried out any particularly interesting information processing functions.

Usha Kumari, Research Scholar**,** Jagannath University NCR, Haryana, India

Dr. Gaurav Aggarwal Associate Professor & HOD (Dept. of CSE) Jagannath University NCR, Haryana, India

Promising & Emerging Technology: The 1950s and 1960s: The First Golden Age of Neural Networks Mark I perceptron, the first successful neuro-computer was developed during 1957 and 1958 by Frank Rosenblatt, Charles Wightman, and others. That is why Rosenblatt is known as the founder of Neurocomputing. He was primarily interested in pattern recognition. Besides inventing the perceptron, Rosenblatt also wrote an early book on Neurocomputing, 'Principles of Neurodynamics'. Slightly later than Rosenblatt, but similar to him, was Bernard Widrow. Widrow, working with his graduate students (most notably Marcian E. "Ted" Hoff, who later went on to invent the microprocessor) developed a different type of neural network processing element called ADALINE, which was equipped with a powerful new learning law which, unlike the perceptron leaning law, is still in widespread use. Widrow and his students applied the ADALINE successfully to a large number of toy problems, and produced several films of their successes. Besides Rosenblatt and Widrow, there were a number of other people during the late 1950s and early 1960s who had substantial success in the development of neural network architectures and implementation concepts. Notwithstanding the considerable success of these early Neurocomputing researchers, the field suffered from two glaringly obvious problems. First, the majority of researchers approached the subject from a qualitative and experimental point of view. This experimental emphasis resulted in a significant lack of rigor and a looseness of thought that bothered many established scientists and engineers who established the field. Second, an unfortunate large fraction of neural networks researchers were carried away by their enthusiasm in their statements and their writings. For example, there were widely publicized predictions that artificial brains were just a few years away from development, and other incredible statements. Besides the hype and general lack of rigor, by the mid 1960s researchers had run out of good ideas. The last chapter of this era was a campaign led by Marvin Minsky and Seymour Papert to discredit neural network research and divert neural network research funding to the field of "Artificial Intelligence". The campaign was waged by the means of personal persuasion by Minsky and Papert and their allies, as well as by limited circulation of unpublished technical manuscript (which was further published in 1969 by Minsky and Papert as the book Perceptrons).

Period of Frustration & Disrepute: Despite of Minsky and Papert's exposition of the limitations of perceptrons, research on neural network continued. A great deal of neural network research went on under the headings of adaptive signal processing, pattern recognition, and biological modeling. In fact, many of the current leaders in the field began to publish their work during 1970s. Examples include Amari, Fukushima, Grossberg and Klopf and Gose. These people, along with few others who came in over the next 13 years, put the field of neural network on a firm floor and paved the way for the renovation of the field.

Innovation: In spite of public interest and available funding being minimal, several researchers continued working to develop neuromorph based computation methods for problems such as pattern recognition. During this period several paradigms were generated which modern work continues to enhance & implement.

- 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 implemented the back-propagation learning method, however several years passed before this approach was popularized. Back-propagation nets are probably the most well known and widely applied of the neural networks today. In essence, the back-propagation net. is a Perceptron with multiple layers, a different threshold 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 pattern 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.

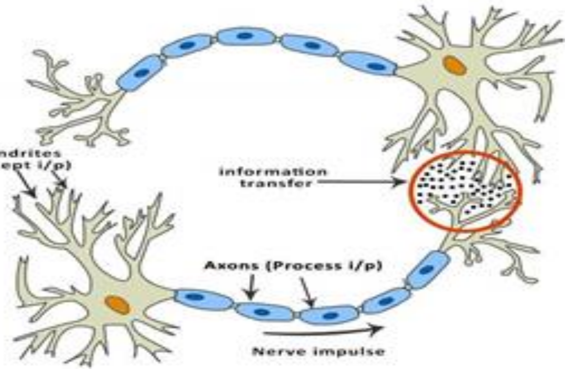
Re-Emergence: It was by the early 1980s that many Neuro-computing researchers became strong enough to begin submitting proposals to explore the development of neuro-computers and of neural network applications. In the years 1983–1986 John Hopfield, an established physicist of worldwide reputation had become interested in neural networks a few years earlier. Hopfield wrote two highly readable papers on neural networks in 1982 and 1984 and these, together with his many lectures all over the world, persuaded hundreds of highly qualified scientists, mathematicians, and technologists to join the emerging field of neural networks. In 1986, with the publication of the "PDP books" (Parallel Distributed Processing, Volumes I and II, edited by Rumelhart and McClelland), the field exploded. In 1987, the first open conference on neural networks in modern times, the IEEE International Conference on Neural Networks was held in San Diego, and the International Neural Network Society (INNS) was formed. In 1988 the INNS journal Neural Networks was founded, followed by Neural Computation in 1989 and the IEEE Transactions on Neural Networks in 1990.

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

Basic Structure of ANNs

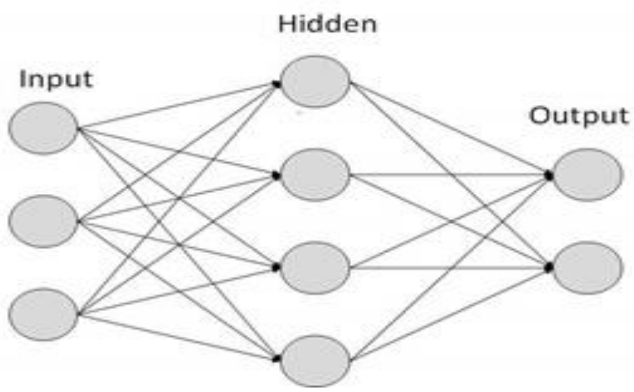
The logic of ANNs is based on the belief that functioning of human brain by making the correct connections, can be copied using silicon and wires as living neurons and dendrites.

Neurons are the 100 billion nerve cells that compose the human brain. They are connected to other thousand cells by Axons. Dendrites accept the stimuli from external environment or inputs from sensory organs. These inputs create electric impulses, which quickly travel through the neural network. A neuron may then transmit the message to other neuron to handle the issue or does not transmit it forward.



ANNs are composed of multiple nodes, which behave same as biological neurons of human brain. The neurons are connected by links and they interact with each other. The nodes can take input data and perform simple operations on the data. The result of these operations is handed over to other neurons. The output at each node is referred as its activation or node value.

Every link is associated with weight. ANNs are capable of learning by altering weight values. The following illustration shows a simple ANN –



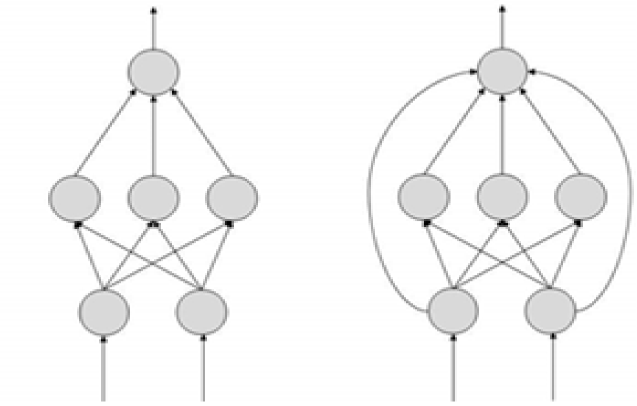
A. Types of Artificial Neural Networks

We have two types of Artificial Neural Network topologies

− FeedForward and Feedback.

1) FeedForward ANN

The information flow is unidirectional. A unit sends information to other unit and that other unit does not receive any information. No feedback loops exist. Such ANNs are used in pattern generation/recognition/classification. They have fixed inputs and outputs.

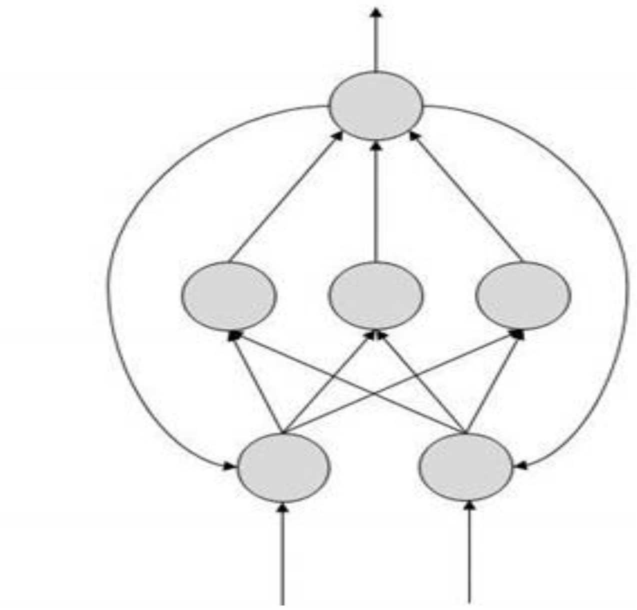


2) FeedBack ANN

In these, feedback loops are allowed. Such ANNs are used in content addressable memories.

B. Machine Learning in ANNs

ANNs have an ability to learn and they need to be trained. There are several learning strategies for that−



- Supervised Learning – There is a role of a teacher who is scholar compared to the ANN. The teacher feeds some example data about which the teacher already knows the answers.

For example, pattern recognition. The ANN comes up with guesses while recognizing. Then the teacher avails the ANN the answers. The network then compares its guesses with the teacher's "correct" answers and makes adjustments according to errors.

- Unsupervised Learning − It is required when there is no example data set with known answers. For an instance, searching for a hidden pattern. In this case, clustering i.e. dividing a set of elements into groups according to some unknown pattern, is carried out based on the existing data sets.
- Reinforcement Learning − This strategy is made on observation. The ANN makes a decision by observing its environment. If the observation is negative, the network adjusts its weights to be capable of making a different decision the next time.

1) Back Propagation Algorithm

It is the training or learning algorithm. Example play a huge role here. If you submit to the algorithm the example of what you want the network to do, it changes the network's weights so that it can produce required output for a particular input after completing the training.

Back Propagation networks are useful for simple Pattern Recognition and Mapping Tasks.

C. Bayesian Networks (BN)

These are also known as Belief Networks or Bayes Nets .They are the graphical structures used to represent the probabilistic relationship among a set of random variables. BNs reason about the uncertain domain.

Each node represents a random variable with specific propositions in these networks. For example, in a medical diagnosis domain, the node Cancer represents the proposition that a patient has cancer. The edges connecting the nodes represent probabilistic dependencies among those random variables. If out of two nodes, one is affecting the other then they must be directly connected in the directions of the effect. The strength of the relationship between variables is measured by the probability associated with each node.

The only constraint on the arcs in a BN is that you cannot return to a node simply by following directed arcs sufficing for the BNs to be called Directed Acyclic Graphs (DAGs).

BNs are capable of handling multivalued variables simultaneously. The BN variables are composed of two dimensions –

Range of prepositions

Probability assigned to each of the prepositions.

Consider a finite set X = {X1, X2, …,Xn} of discrete random variables, where each variable Xi may take values from a finite set, denoted by Val(Xi). If there is a directed link from variable Xi to variable, Xj, then variable Xi will be a parent of variable Xj showing direct dependencies between the variables.

The BN is ideal for aggregating prior knowledge and observed data. It can be used to learn the causal relationships and understand various problem domains and to predict future events, even in the case of missing data.

Building a Bayesian Network

A Bayesian network can be built by a knowledge engineer. Following are the number of steps the knowledge engineer needs to take while building it.

Example problem – *Lung cancer.* A patient has been suffering from breathlessness. He visits the doctor, suspecting that he has lung cancer. The doctor knows that other than the lung cancer, there are various other possible

diseases the patient might be suffering from, such as tuberculosis and bronchitis.
Collect Relevant Information of Problem-

- Is the patient a smoker? If yes, then high chances of cancer and bronchitis.
- Is the patient exposed to air pollution? If yes, what sort of air pollution?
- Take an X-Ray. Positive X-ray would indicate either TB or lung cancer.

Identify Interesting Variables

The knowledge engineer tries to find the answer of the following questions −

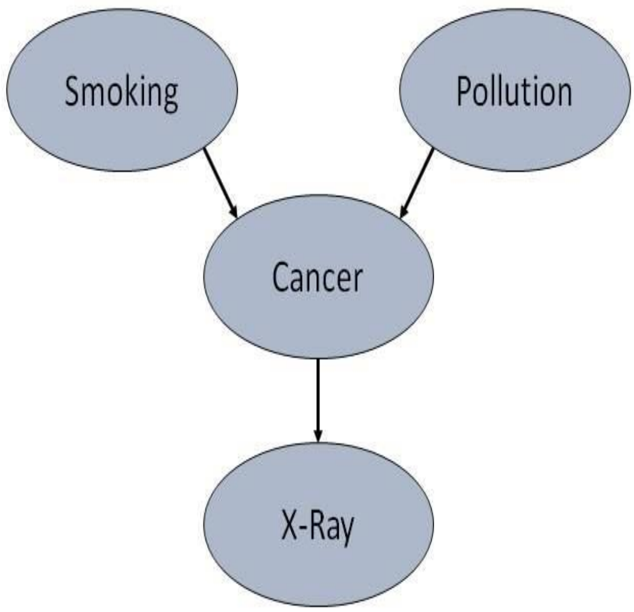
- Which nodes to represent?
- What values can they take? In which state can they be?

Let us consider nodes for with only discrete values for now. The variable must accept exactly one of these values at a time.
Most general types of discrete nodes are −

- Boolean nodes − They represent propositions, taking binary values-TRUE (T) and FALSE (F).
- Ordered values − A node Pollution might represent and take values from {low, medium, high} describing degree of a patient's exposure to pollution.
- Integral values − A node called Age might represent patient's age with possible values from 1 to 120. Even at this early stage, modelling choices are being made.

Possible nodes and values for the lung cancer example −
Create Arcs between Nodes

Topology of the network should include qualitative relationships between variables.
For an instance, what causes a patient to have lung cancer? - Pollution and smoking. Then add arcs from node Pollution and node Smoker to node Lung-Cancer.
Similarly, if the patient has lung cancer, then X-ray result should be positive. So, add arcs from node Lung-Cancer to node X-Ray.



Specify Topology

Conventionally, BNs are presented in the way that the arcs point from top to bottom. The set of parent nodes of a node X is given by Parents(X).
The Lung-Cancer node has two parents (reasons or causes)- Pollution and Smoker, while node Smoker is an ancestor of node X-Ray. Similarly, X-Ray is a child (consequence or effects) of node Lung-Cancer and successor of nodes Smoker and Pollution.

Conditional Probabilities

Now quantify the relationships between connected nodes by specifying a conditional probability distribution for each node. As only discrete variables are considered here, this takes the form of a Conditional Probability Table (CPT).
Firstly, for each node we need to look at all the possible combinations of values of those parent nodes. Each such combination is called an instantiation of the parent set. For each distinct instantiation of parent node values, we need to assign the probability that the child will accept.
For an instance, the Lung-Cancer node's parents are Pollution and Smoking. The possible values they take are = {(H,T), (H,F), (L,T), (L,F)}. The CPT specifies the probability of cancer for each of these cases as <0.05, 0.02, 0.03, 0.001> respectively.

Each node will have conditional probability associated with them as follows −

Smoking
P(S = T)
0.30

Pollution
P(P = L)
0.90

Lung-Cancer		
P	S	P (C = T|P, S)
H	T	0.05
H	F	0.02
L	T	0.03
L	F	0.001

X-Ray	
C	X = (Pos | C)
T	0.90
F	0.20

D. Applications of Neural Networks

They can perform tasks that are easy for a human but difficult for a machine −

- Aerospace − Autopilot aircrafts, aircraft fault detection.

- Automotive − Automobile guidance systems.
- Military − Face recognition Weapon orientation and steering, target tracking, object discrimination, signal/image identification.
- Electronics − Code sequence prediction, IC chip layout, chip failure analysis, machine vision, voice synthesis.
- Financial − Real estate appraisal, loan advisor, mortgage screening, corporate bond rating, portfolio trading program, corporate financial analysis, currency value prediction, document reading, credit application evaluating.
- Industrial − Manufacturing process control, product design and analysis, quality inspection systems, welding quality analysis, paper quality prediction, chemical product design analysis, dynamic modelling of chemical process systems, machine maintenance analysis, project bidding, planning, and management.
- Medical − Cancer cell analysis, EEG and ECG analysis, prosthetic design, transplant time optimizer.
- Speech − Speech recognition, speech classification, text to speech conversion.
- Telecommunications − Image and data compression, automated information services, real-time spoken language translation.
- Transportation − Truck Brake system diagnosis, vehicle scheduling, routing systems.
- Software − Pattern Recognition in facial recognition, optical character recognition, etc.
- Time Series Prediction − Predictions on stocks and natural calamities.
- Signal Processing − Process an audio signal and filter it appropriately in the hearing aids.
- Control − Make steering decisions of physical vehicles.
- Anomaly Detection – Since ANNs are expert at recognizing patterns, they can also be trained to obtain an output when something undesirable occurs that mismatches the pattern.

1) Advantages:

- A neural network can perform tasks which a linear program cannot perform.
- Due to their parallel nature, even when an element of the neural network fails, it can continue without any problem.
- A neural network need not be reprogrammed as it learns itself.
- It can be implemented in an easily without any problem.
- As the adaptive, intelligent systems, neural networks are robust and expert at solving complex problems. They are efficient in their programming and the scientists agree that the advantages of using ANNs outweigh the risks.
- It can be implemented in any type of application.

2) Disadvantages:

- The neural networks need training to operate.
- High processing time is required for large neural networks.

The architecture of a neural network is different from the architecture of microprocessors. So, they have to be emulated.

REFERANCE

[1] https://en.wikibooks.org/wiki/Artificial_Neural_Networks/History
[2] https://www.elprocus.com/artificial-neural-networks-ann-and-their-types/
[3] http://www.psych.utoronto.ca/users/reingold/courses/ai/cache/neural4.html
[4] https://cs.stanford.edu/people/eroberts/courses/soco/projects/neural-networks/History/history1.html
[5] https://en.wikipedia.org/wiki/Types_of_artificial_neural_networks
[6] https://www.tutorialspoint.com/artificial_intelligence/artificial_intelligence_neural_networks.htm
[7] http://www2.chemie.uni-erlangen.de/publications/ANN-book/publications/ACS-41-94.pdf
[8] https://cs.stanford.edu/people/eroberts/courses/soco/projects/neural-networks/Architecture/usage.html