Uma Rani et al. International Journal of Recent Research Aspects ISSN: 2349-7688, Special Issue: Conscientious and Unimpeachable Technologies 2016, pp. 40-44

Neural Network Applications in Design Process of Decision Support System

Uma Rani, Surjeet Dalal

Department of Computer Science & Engineering, SRM University, Sonipat, Haryana

Abstract: **This paper is an introduction to artificial neural networks. The different types of neural are explained and demonstrated and a detailed historical background is provided. The connection between the artificial and the real thing is alsoinvestigated and explained. This paper describe the basicbiological neuron and the artificial computational model, outline network architectures and learning laws in the design of the decision support system. The ANN model provide a fast, flexible and strong predictive ability for selecting the product innovation development project.**

Keywords: Decision support system, Neural network, Decision taking,

I. Introduction to Neural Networks

Artificial neural network is an information processing paradigm that is inspired by the way biological nervous system such as 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 (neuron) working in unison to solve specific problems.ANNs like people, lean by example. An ANN is configured for a specific application, such as pattern recognition through learning process. Learning in biological systems involves adjustment to the synaptic connections that exist between the neuron.

<u>Why artificial neural networks?</u>

The long course of evolution has given the human brain many desirable characteristics not present in Von Neumann or modern parallel computers. These include massive parallelism, distributed representation and computation, learning ability, generalization ability, adaptivity, inherent contextual information processing, fault tolerance, and low energy consumption. It is hoped that devices based on biological neural networks will possess some of these desirable characteristics. Modern digital computers outperform humans in the domain of numeric computation and related symbol manipulation.

However, humans can effortlessly solve complex perceptual problems (like recognizinga man in a crowd from a mere glimpse of his face) at such a high speed and extent as to dwarf the world's fastest computer. Why is there such a remarkable difference in their performance? The biological neural system architecture is completely different from the von Neumann architecture. This difference significantly affects the type of functions each computational model can best perform.

Numerous effortsto develop "intelligent"programs based on vonNeumann's centralized architecture have not resulted in general-purposeintelligent programs. Inspired by biological neural networks, ANNs aremassively parallel computing systems consisting of an extremely large numberof simple processors with anyinterconnections. ANN models attempt to use some "organizational" principles believed to be used in the human agesof Neural Networks.

<u>Biological neural networks</u>

A *neuron* (or nerve cell) is a special biological cell that processes information. It is composed of a cell body, or *soma*, and two types of out-reaching tree-like branches: the *axon* and the *dendrites*.

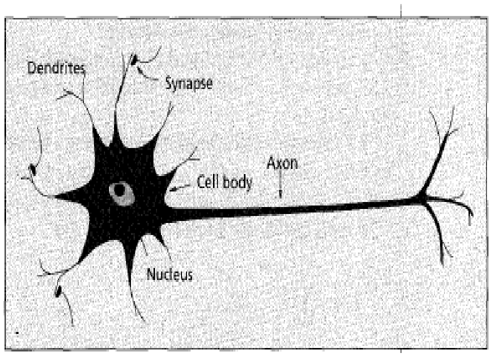
The cell body has ac nucleus that contains information about hereditary traits and plasma that holds the molecular equipment for producingmaterial needed by the neuron. A neuron receives signals (impulses) from other neurons through its dendrites (receivers) and transmits signals generated by its cell body along the axon (transmitter), which eventually branches into strands and sub strands. At the terminals of thesestrands are the *synapses*.

A synapse is an elementary structureand functional unit between two neurons (an axonstrand of one neuron and a dendrite of another), When theimpulse reaches the synapse's terminal, certain chemicalscalled neurotransmitters are released. The neurotransmittersdiffuse across the synaptic gap, to enhance or inhibit,depending on the type of the synapse, the receptor neuron'sown tendency to emit electrical impulses.

The synapse'seffectiveness can be adjusted by the signals passing throughit so that the synapses can *learn* from the activities in whichthey participate. This dependence on history acts as amemory,which is possibly responsible for human memory

II. Learning:

The ability to learn is a fundamental trait of intelligence. Although a precise definition of learning is difficult to formulate, a learning process in the ANN context can be viewed as the problem of updating



network architecture and connection weights so that a network can efficiently perform a specific task. The network usually must learn the connection weights from available training patterns.Performance is improved over time by iteratively updating the weights in the network. ANNs' ability to automatically *learnfrom examples* makes them attractive and exciting. Instead of following a set of *rules* specified by human experts, ANNs appear to learn underlying rules (like input-output relationships) from the given collection of representative examples. This is one of the major advantages of neural networks over traditional expert systems. To understand or design a learning process, you must first have a model of the environment in which a neural network operates, that is, you must know what information is available to the network. We refer to this model asa learning paradigm. Second, you must understand hownetwork weights are updated, that is, which *learning rules*govern the updating process. A *learning algorithm*refersto a procedure in which learning rules are used for adjustingthe weights.

Learning laws describe the weight vector for the ith processing unit at time instant (t+1) in terms of weight vector at time instant (t) as follows:

$$w_i(t+1) = w_i(t) + \Delta\, w_i(t)$$

Main learning paradigms:

- supervised,
- unsupervised,
- reinforcement.

Supervised:

In supervised learning, orlearning with a "teacher," the network is provided with acorrect answer (output) for every input pattern. Weightsare determined to allow the network to produce answersas close as possible to the known correct answers.Reinforcement learning is a variant of supervised learningin which the network is
provided with only a critique on the correctness
of network outputs, not the correctanswers themselves.

Unsupervised:

Unsupervised learning, orlearning without a teacher, does not require a correctanswer associated with each input pattern in the trainingdata set. It explores the underlying structure in the data,or correlations between patterns in the data, and organizespatterns into categories from these correlations.

Reinforcement:

In computer science, reinforcement learning is a sub-area of machine learning concerned with how an *agent* ought to take *actions* in an *environment* so as to maximize some notion of long-term *reward*. Reinforcement learning algorithms attempt to find a *policy* that maps *states* of the world to the actions the agent ought to take in those states.Reinforcement learning differs from the supervised learning problem in that correct input/output pairs are never presented, nor sub-optimal actions explicitly corrected. Further, there is a focus on on-line performance, which involves finding a balance between exploration (of uncharted territory) and exploitation (of current knowledge).

Basic Learning laws:

- Hebb's Law
- Perceptron Law
- Delta Law
- Widrow and Hoff Law
- Correlation Law
- Instar Law
- Outstar Law

Hebb's Law

In this law the change in the weight vector is given by

$$\Delta\, w_i = \eta\, f\,(w^t{}_i a)\; a$$

Therefore, thejth component of $\Delta\, w_i$ given by

$$\Delta\, w_{ij} = \eta\, f\,(w_i^T\; a)\; a_j$$
$$= \eta\, s_i\, a_j \qquad \text{for } j = 1,2,\ldots\ldots,M$$

Where s_i is the output signal of the ith unit. The law states that the weight increment is proportional to the product of the input data and the resulting output signal of the unit. This law requires weight initialization to small random values around $w_{ij} = 0$ prior to learning. This law represents an unsupervised learning.

Perceptron Law:

In this rule the change in the weight vector is given by

$$\Delta\, w_i = \eta\, [b_i - \mathrm{sgn}(w_i^T\, a)]a$$

where sgn(x) is sign of x. therefore, we have

$$\Delta\, w_{ij} = \eta[b_i - \mathrm{sgn}(w_i^T\, a)]a_j$$

for j-1, 2,…..M

This law is applicable only for bipolar output functions f (.). This is a supervised learning law, as the law requires a desired output for each input. In implementation, the weights can be initialized to any random initial values. The weights converge to the final values eventually by repeated use of the input-output pattern pairs, provided the pattern pairs are representable by the system.

Delta Law:

The change in the weight vector is given by

$$\Delta\, w_i = \eta\, [b_i - f(w_i^T a)]\; f(w_i^T a)\; a$$

where f(x) is the derivative with respect to x. Hence,

$$\Delta\, w_{ij} = \eta\, [b_i - f(w_i^T\, a)]f(w_i^T)a_j$$
$$= \eta\, [b_i\;\, -s_i]f(x_i)a_j$$

for j = 1,2,…,M

This law is valid only for a differentiable output function, as it depends on the derivative of the output function f (.). it is a supervised learning law since the change in the weight is based on the error between the desired and the actual output values for a given input. In implementation, the weights can be initialized to any random values as the values are not very critical. The weightsconverge to the final values eventually by repeated use of the input-output pattern pairs. Inimplementation; the weights can be initialized to any random values as the values are not very critical. The weights converge to the final values eventually by repeated use of the input-output pattern pairs.

WidrowAndHoff Law:

The change in the weight vector is given by

$$\Delta w_i = \eta [b_i - w_i^T a]a$$

Hence

$\Delta w_{ij} = \eta [b_i - w_i^T a]a_j$

for j =1, 2,……, M

This is a supervised learning law and is a special case of the delta learning law, where the output function is assumed linear, i.e.

$$f(x_i) = x_i$$

In this case the change in the weight is made proportional to the negative gradient of the error between the desired output and the continuous activation value which is also the continuous output signal due to linearity of the output function. This is also called the Least Mean Squared (LMS) error learning law. In implementation, the weights may be initialized to any values. The input-output pattern pairs data is applied several times to achieve convergence of the weights for a given set of training data.

Correlation Learning Law:

The change in the weight vector is given by

$$\Delta w_i = \eta\, b_i\, a$$

Therefore

$$\Delta w_{ij} = \eta\, b_i a_j \quad \text{for } j=1, 2,\ldots,M$$

this is a special case of the hebbian learning with the output signal(s_i) being replaced by the desired signal(b_i). But the Hebbian learning iasn unsupervised learning, whereas the correlation learning is a supervised learning, since it uses the desired output value to adjust the weights. in the implementation of the learning law, the weights are initialized to small random values close to zero.

i.e.W_{ij} =0.

InstarLearminglaw:

This is relevant for a collection of neurons, organized in a layer. All the inputs are connected to each of the units in the output layer in a feed forward manner. For a given input vector a, the output from each unit i is computed using the weighted sum w_i^T a. the unit k that gives maximum output is identified. that is

$$w_k^T a = \max(w_i^T a)$$

Then the weight vector leading to the kth unit is adjusted as follows:

$$\Delta w_k = \eta\, (a - w_k)$$

Therefore,

$$\Delta w_{kj} = \eta\, (a_j - w_{kj}) \quad \text{for } j = 1,2,\ldots,M$$

The final weight vector tends to represent a group of input vectors within a small neighbored. This is a case of unsupervised learning. In implementation, the values of the weight vectors are initializes to random values prior to learning, and the vector lengths are normalized during learning.

Outstar Learning Law:

The outstar learning law is also related to a group of units arranged in a layer. In this law the weights are adjusted so as to capture the desired output pattern characteristics. The adjustment of the weights is given by

$$\Delta w_{jk} = \eta\, (b_j - w_{jk}) \quad \text{for } j = 1,2,\ldots,M$$

where the kth unit is the only active unit in the input layer. The vector b =$(b_1,b_2,\ldots,b_M)^T$ is the desired response firm layer of M units. The outstar learning is a supervised learning law, and it is used with a network of instars to capture the characteristics of the input and output patterns for data compression. In implementation, the weight vectors are initialized to zero prior to learning.

Artificial intelligence in decision support system

Intelligence is the ability to think and understand instead of doing things by instinct or automatically. The basic ideas of intelligence are the studying thought processes of humans, dealing with representing and duplicating those processes via machines (e.g., computer, robots), and exploring the behaviour by a machine but performed by human being. Artificial Intelligence (AI) study is how to make computers do things at which, at the moment people are better, some of intelligent behaviors in a computer system are:

- Learn and understand from experience
- Conclude in situation where exist fuzziness and uncertainty
- Use knowledge and experience to manipulate the environment
- Think and reasoning
- Understand and infer in ordinary, rational ways.
- Respond quickly and successfully to new situations.

Intelligent abilities and behaviors integrate with computer system will produce an intelligent machine. The machine should help humans to make decision, to search for information, to control complex objects. In order to develop intelligent computer system, we have to capture, organize and use human expert knowledgein some areas of expertise, upgrade the computational power of the system's brain with the sophistication of algorithms using sensory processing, world modeling, behavior generation, the amount of information and values the system has stored in its memory.

In system development, some AI features that can be used to develop an intelligent system are:

- Symbolic processing which is non-algorithmic methods of problem solving
- Heuristics which is intuitive knowledge or rules of thumb, learned from experience.
- Inference that includes reasoning capabilities that can build higher-level knowledge from existing heuristics

(from facts and rules using heuristics or other search approaches)

- Machine learning that allows system to adjust their behavior and react to changes in the outside environment.

III. DSS

An application uses to support decision making is usually known as DSS and can be categorized into three categories which are passive DSS, active DSS and proactive DSS. Passive DSS is a traditional DSS with functionalities to react as a personalized decision support built-in knowledge, no content and only for static user preference. Besides that, the components of passive DSS are Data warehouse, OLAP and rule-based. The second category of DSS is active DSS which is known as a personalized decision support with learning capability, no content and for static user preference. Expert system, adaptive DSS, knowledge-based system (KBS) are categorized as part of Intelligent DSS (IDSS). Finally, the third category is proactive DSS, which known as Ubiquitous Computing Technology-based DSS (ubiDSS) which contains decision making and context aware functionalities. This type of DSS has mobility, portability and pro-activeness capabilities.

IV. DATA

The data consist of 87 product innovation projects introduced by entrepreneurial firm in Thailand. To assess the validity of the project performance and avoid potential bias, we asked the contact people and other company executives to assess the project's performance and classify the project as either a success or a failure using the following criteria: The project should be considered a success if the new product has been completely launched and gained an expected market share; the project should be considered a failure if the new product failed the product testing, or manufacturing, or launching. Network's input layer has 23 nodes, corresponding to an independent variable. All inputs are measured on a 0-to-10 scale. The output of the networks is dichotomous success, or failure. The target during training is a dichotomous variable that represents a self-selected successful or unsuccessful project as determined by the informant. This MFN model simply predicts project success or failure.

V. CONCLUSION

Developments in ANNs have stimulated a lot of enthusiasm and criticism. Some comparative studies are optimistic, some offer pessimism. For many tasks, such as pattern recognition, no one approach dominates the others. The choice of the best technique should be driven by the given application's nature. We should try to understand the capacities, assumptions and maximally exploit their complementary advantages to develop better intelligent systems. such an effort may lead to synergistic approach that combines the strengths of ANNs with other technologies to achieve significantly better performance for challenging problems. It is clear that the communication and cooperative work between researchersworking in ANNs and other disciplines will not only avoid repetitious work but will stimulate and benefit individual disciplines.

This objective of this study is to develop the model concerning factors influencing the successful product innovation and the utilizing of artificial intelligent methodologies and applications. We conclude that firm's innovation capabilities, firm's new product development capability and external competitive environment are three groups of factors that influence the successful product innovation. In searching for the alternative and more effective tools to the statistic analysis method traditionally used, we selected the Artificial Neural Network (ANN) model, which is particularly useful for modeling underlying patterns in data through a learning process. We developed the model using the ANN and trained with 23 inputs and data from 87 product innovation projects to recognize patterns consistent with success and failure. From the measure of the predictive ability of the model through the variance measurement (MSE) and the accuracy of the predictive ability (Kappa Statistic), the resulting strong prediction ability of the network is recognized.

VI. REFERENCES

[1]. DARFA Neural Network Study, FCEAInt'l Press, Fairfax,Va., 1988

[2]. J. Hertz, A. Krogh, and R.G. Palmer, Introduction to the "theory ofNeural Computation, Addison-Wesley, Reading, Mass., 1991.

[3]. S. Haykin, Neural Networks: A Comprehensive Foundation, MacMillan College Publishing Co., New York, 1994.

[4]. W.S. McCulloch and W. Pitts, "A Logical Calculus of Ideas Immanent in Nervous Activity," Bull. Mathematical Bio-physics, Vol. 5, 1943, pp. 115-133.

[5]. R. Rosenblatt, Principles of Neurodynamics, Spartan Books, New York, 1962.

[6]. M. Minsky and S. Papert, Perceptrons:AnIntroduction to Computational Geometry, MIT Press, Cambridge, Mass., 1969.

[7]. J.J. Hopfield, "Neural Networks and Physical Systems with Emergent Collective Computational Abilities," in Roc.Nat'l Academy of Sciences, USA 79, 1982, pp. 2, 5542,558.

[8]. P. Werbos, "Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences," PhD thesis, Dept. of Applied Mathematics, Harvard University, Cambridge, Mass., 1974.

[9]. D.E. Rumelhart and J.L. McClelland, ParallelDistributedRocessing: Exploration in the Microstructure of Cognition, MIT Press, Cambridge, Mass., 1986.

[10]. J.A. Anderson and E. Rosenfeld, Neurocomputing: Foundations ofResearch, MIT Press, Cambridge, Mass., 1988.

[11]. S. Brunak and B. Lautrup, Neural Networks, Computers with Intuition, World Scientific, Singapore, 1990.

