RESEARCH ARTICLE OPEN ACCESS

Systematic Study of Fuzzy, Neural Network and Neurofuzzy Systems

K .Geetha [1] and Dr. S. Santhosh Baboo[2]
[1] Department of Information Technology, Dr.N.G.P.Arts and Science College, Coimbatore
[2] Department of Computer Science, D.G.Vaishnav College, Chennai

Abstract
The techniques in artificial intelligence are used in almost all the fields where human reasoning and uncertainties can be effectively modeled. The popular techniques in AI are fuzzy logic and neural networks which can be used either separately or applied together. When they are used in combined way, they are called Neuro-Fuzzy Systems. The reasons to combine these two paradigms come out of the difficulties and inherit limitations of each isolated paradigm. This paper shows uniqueness in each technology and compares with others to show the need for hybrid system (Neuro fuzzy).It gives an insight of different concepts how they are utilized in various fields to yield optimized solution.
Keywords: Fuzzy Set, Fuzzy Logic, Fuzzy Rules, Neural Networks, Neural Network Architecture, Learning Methods, Neuro-Fuzzy System

I. INTRODUCTION

The conventional approaches to knowledge representation lack the means for representation the meaning of fuzzy concepts. The development of fuzzy logic was motivated in large measure by the need for a conceptual framework which can address the issue of uncertainty and lexical imprecision. Fuzzy logic provides an inference morphology that enables approximate human reasoning capabilities to be applied to knowledge-based systems. The theory of fuzzy logic provides a mathematical strength to capture the uncertainties associated with human cognitive processes, such as thinking and reasoning.

For example, when one is designing an expert system to mimic the diagnostic powers of a physician, one of the major tasks is to codify the physician's decision-making process. The designer soon learns that the physician's view of the world, despite her dependence upon precise, scientific tests and measurements, incorporates evaluations of symptoms, and relationships between them, in a "fuzzy," intuitive manner: deciding how much of a particular medication to administer will have as much to do with the physician's sense of the relative "strength" of the patient's symptoms as it will their height/weight ratio. While some of the decisions and calculations could be done using traditional logic, we will see how fuzzy systems affords a broader, richer field of data and the manipulation of that data than do more traditional methods.

II. THEORY OF REASONING

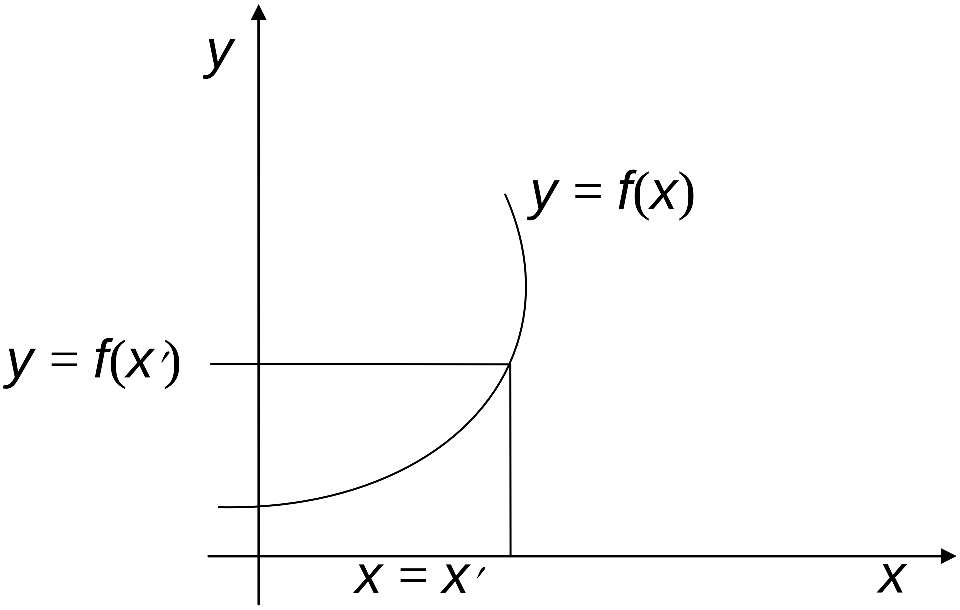
[7]Zadeh introduced the theory of approximate reasoning in 1979. This theory provides a powerful framework for reasoning in the face of imprecise and uncertain information. This theory is the representation of propositions as statements assigning fuzzy sets as values to variables.

Suppose there are two interactive variables $x \in X, y \in Y$ and the causal relationship between x and y is completely known. (i.e.) that y is a function of x, $y = f(x)$.

This leads to the inferences as Premise $y = f(x)$, Fact $x = x'$, Consequence $y = f(x')$

This inference rule says that if $y = f(x)$, $\forall x \in X$ and observation shows that $x = x'$ then y takes the value $f(x')$.

Fig: 1 Simple crisp inference.



More often than not we do not know the complete causal link f between x and y, only we now the values of $f(x)$ for some particular values of x

R_1: If $x = x_1$ then $y = y_1$
also
R_2: If $x = x_2$ then $y = y_2$
also
. . .
also
R_n: If $x = x_n$ then $y = y_n$

Suppose that we are given an $x' \in X$ and want to find a $y' \in Y$ which corresponds to x' under the rule-base.

R_1: If $x = x_1$ then $y = y_1$
also
R_2: If $x = x_2$ then $y = y_2$
also
. . .
also
R_n: If $x = x_n$ then $y = y_n$
Fact: $x = x'$
Consequence: $y = y'$

This problem is frequently quoted as interpolation.

Let x and y be linguistic variables, e.g.”x is high” and”y is small”.

The basic problem of approximate reasoning is to find the membership function of the consequence C from the rule-base $\{R1 \ldots R_n\}$ and the fact A.

R_1: if x is A_1 then y is C_1,
R_2: if x is A_2 then y is C_2,
………………………
R_n: if x is A_n then y is C_n
Fact: x is A
Consequence: y is C

Zadeh introduced a number of translation rules which allow us to represent some common linguistic statements in terms of propositions in our language. The translation rules are *Entailment rule, Conjunction rule, Disjunction rule, Projection rule and*
Negation rule

2.1 Fuzzy System

The notion central to fuzzy systems is that truth values (in fuzzy logic) or membership values (in fuzzy sets) are indicated by a value on the range [0.0, 1.0], with 0.0 representing absolute Falseness and 1.0 representing absolute Truth. For example, the statement:

"Kumar is old."

If Kumar age was 75, we might assign the statement the truth value of 0.75. The statement could be translated into set terminology as follows:

"Kumar is a member of the set of old people."

This statement would be rendered symbolically with fuzzy sets as:

mOLD(Kumar) = 0.75

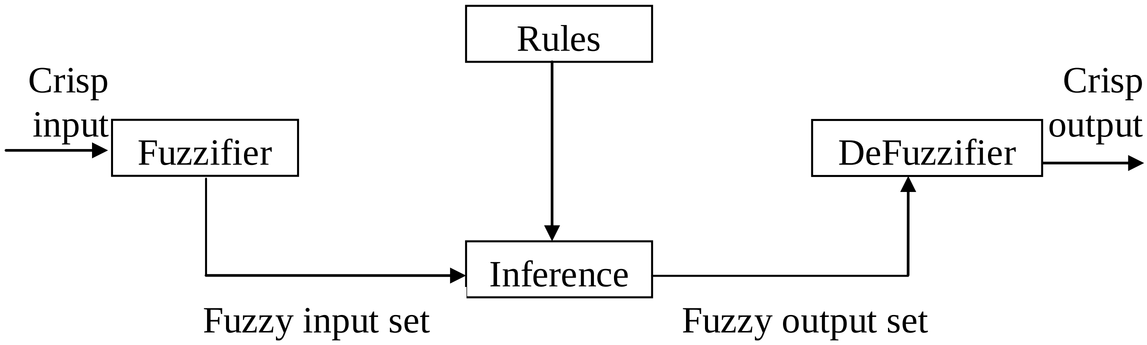
where m is the membership function, operating in this case on the fuzzy set of old people, which returns a value between 0.0 and 1.0.

The next step in establishing a complete system of fuzzy logic is to define the operations of *EMPTY, EQUAL, COMPLEMENT (NOT), CONTAINMENT, UNION (OR), and INTERSECTION (AND).*

2.2 Fuzzy Logic

A fuzzy logic system (FLS) can be denied as the nonlinear mapping of an input data set to a scalar output data. A FLS consists of four main parts: fuzzier, rules, inference engine, and defuzzier. These components and the general architecture of a FLS is shown in Figure 2

Fig: 2 Fuzzy Logic Systems



The process of fuzzy logic is a crisp set of input data are gathered and converted to a fuzzy set using fuzzy linguistic variables, fuzzy linguistic terms and membership functions. This step is known as fuzzification. Afterwards, an inference is made based on a set of rules. Lastly, the resulting fuzzy output is mapped to a crisp output using the membership functions, in the defuzzification step.

2.2.1. Linguistic Variables

Linguistic variables are the input or output variables of the system whose values are words or sentences from a natural language, instead of numerical values. A linguistic variable is generally decomposed into a set of linguistic terms.

Example: Consider age of a person, Let age (a) is the linguistic variable which represents person's age. To qualify the age, terms such as "young" and "old" are used in real life. These are the linguistic values of age. Then, age (a) = {very young, young, old, very old} can be the set of decompositions for the linguistic variable age. Each member of this decomposition is called a linguistic term and can cover a portion of the overall values of the age.

2.2.2. Membership Function

Membership functions are used in the fuzzification and defuzzification steps of a FLS, to map the non-fuzzy input values to fuzzy linguistic terms and vice versa. A membership function is used to quantify a linguistic term. There are different forms of membership functions such as triangular, trapezoidal, piecewise linear, Gaussian, or singleton. The most common types of membership functions are triangular, trapezoidal, and Gaussian shapes.

2.3 Fuzzy Rules

In a FLS, a rule base is constructed to control the output variable. A fuzzy rule is a simple IF-THEN rule with a condition and a conclusion. In Table 1, sample fuzzy rules for the healthy life system in Figure 2 are listed. Table 2 shows the matrix representation of the fuzzy rules for the said FLS. Row captions in the matrix contain the values that age can take, column captions contain the values for target group, and each cell is the resulting command when the input variables take the values in that row and column. For instance, the cell (3, 4) in the matrix can be read as follows: If age is old and target is very old then command is yoga.

Table 1: Fuzzy Rules

Fuzzy Rules	
1	IF (age is old OR very old) AND (target is old age group) THEN command is yoga
2	IF (age is young OR very young) AND (target is young age group) THEN command is exercise
3	IF (age is young) AND (target is young age group) THEN command is aerobics

Table 2: Fuzzy matrix example

Age / Target	very young	young	old	very old
very young	exercise	aerobics	yoga	yoga
young	exercise	aerobics	exercise	yoga
old	exercise	aerobics	yoga	yoga
very old	yoga	yoga	yoga	yoga

2.3.1 Fuzzy Set Operation

The evaluations of the fuzzy rules and the combination of the results of the individual rules are performed using fuzzy set operations. The operations on fuzzy sets are different than the operations on non-fuzzy sets. The mostly used operations for OR and AND operators are max and min, respectively. After evaluating the result of each rule, these results should be combined to obtain a final result. This process is called inference. The results of individual rules can be combined in different ways.

2.4 Defuzzification

After the inference step, the overall result is a fuzzy value. This result should be defuzzified to obtain a final crisp output. This is the purpose of the defuzzifier component of a FLS. Defuzzification is performed according to the membership function of the output variable.

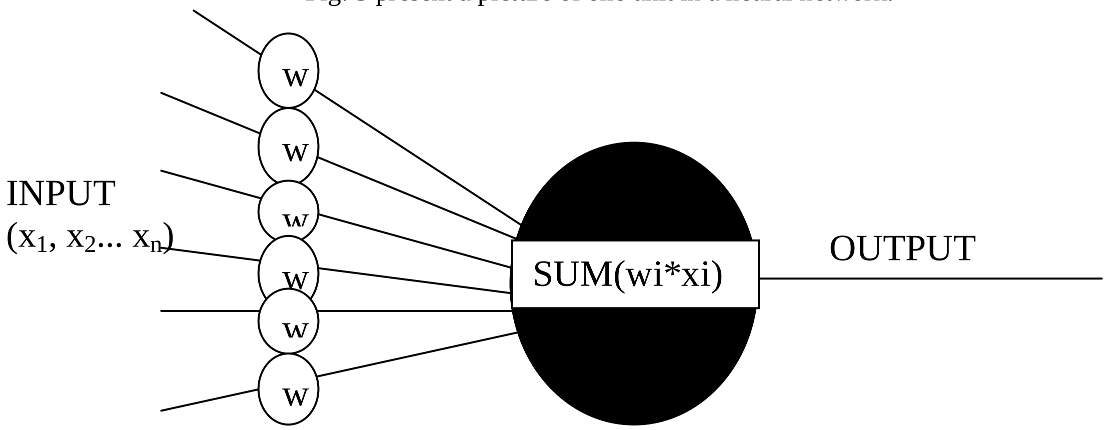
2.5 Limitations of Fuzzy system

In general, knowledge acquisition is difficult and also the universe of discourse of each input variable needs to be divided into several intervals, applications of fuzzy systems are restricted to the fields where expert knowledge is available and the number of input variables is small. To overcome the problem of knowledge acquisition, neural networks are extended to automatically extract fuzzy rules from numerical data.

III. NEURAL NETWORKS

Inspired by the structure of the brain, a neural network consists of a set of highly interconnected entities, called *nodes* or *units*. Each unit is designed to mimic its biological counterpart, the neuron. Each accepts a weighted set of inputs and responds with an output. A neural network is composed of such units and weighted unidirectional connections between them. In some neural nets, the number of units may be in the thousands. The output of one unit typically becomes an input for another. There may also be units with external inputs and/or outputs

Fig: 3 present a picture of one unit in a neural network.



Let $X = (x_1, x_2... x_n)$, where the xi are real numbers, represent the set of inputs presented to the unit U. Each input has an associated weight that represents the strength of that particular connection. Let $W = (w_1,w_2,..., w_n)$, with w_i real, represent the weight vector corresponding to the input vector X. Applied to U, these weighted inputs produce a net sum at U given by
$S = SUM (w_i * x_i) = W.X.$

The weights in most neural nets can be both negative and positive. Another common form for an activation function is a *threshold function*: the activation value is 1 if the net sum S is greater than a given constant T, and is 0 otherwise.

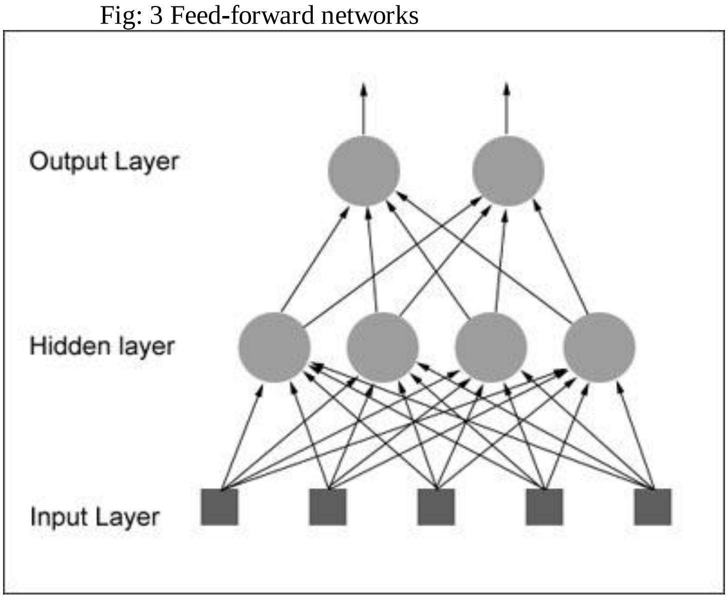
3.1 Neural Network Architecture

ANN is defined as data processing system consists of large number of interconnected processing elements in an architecture which represents the cerebral cortex of brain. The widely used architecture in neural networks is:

3.1.1. Feed-forward networks

Feed-forward ANNs (fig,3) allow signals to travel one way only; from input to output. There is no feedback (loops) i.e. the output of any layer does not affect that same layer. Feed-forward ANNs tend to be straight forward networks that associate inputs with outputs. They are extensively used in pattern recognition. This type of organization is also referred to as bottom-up or top-down

Fig: 3 Feed-forward networks



3.1.2.Feedback or Recurrent network

Feedback networks can have signals travelling in both directions by introducing loops in the network. Feedback networks are very powerful and can get extremely complicated. Feedback networks are dynamic; their 'state' is changing continuously until they reach an equilibrium point. They remain at the equilibrium point until the input changes and a new equilibrium needs to be found. Feedback architectures are also referred to as interactive or recurrent, although the latter term is often used to denote feedback connections in single-layer organizations.

3.1.3. Network layers

The commonest type of artificial neural network consists of three groups, or layers, of units: a layer of "**input**" units is connected to a layer of "**hidden**" units, which is connected to a layer of **"output**" units. (see Fig.3)

- The activity of the input units represents the raw information that is fed into the network.
- The activity of each hidden unit is determined by the activities of the input units and the weights on the connections between the input and the hidden units.
- The behavior of the output units depends on the activity of the hidden units and the weights between the hidden and output units.

This simple type of network is interesting because the hidden units are free to construct their own representations of the input. The weights between the input and hidden units determine when each hidden unit is active, and so by modifying these weights, a hidden unit can choose what it represents.
We also distinguish single-layer and multi-layer architectures. The single-layer organization, in which all units are connected to one another, constitutes the most general case and is of more potential computational power than hierarchically structured multi-layer organizations. In multi-layer networks, units are often numbered by layer, instead of following a global numbering.

3.2 Learning Methods

All learning methods used for adaptive neural networks can be classified into two major categories:

3.2.1. Supervised learning which incorporates an external teacher, so that each output unit is told what its desired response to input signals ought to be. During the learning process global information may be required. Paradigms of supervised learning include error-correction learning, reinforcement learning and stochastic learning.
An important issue concerning supervised learning is the problem of error convergence, ie the minimization of error between the desired and computed unit values. The aim is to determine a set of weights which minimizes the error. One well-known method, which is common to many learning paradigms, is the least mean square (LMS) convergence.

3.2.2. Unsupervised learning uses no external teacher and is based upon only local information. It is also referred to as self-organization, in the sense that it self-organizes data presented to the network and detects their emergent collective properties. Paradigms of unsupervised learning are Hebbian learning and competitive learning.
From Human Neurons to Artificial Neuron either aspect of learning concerns the distinction or not of a separate phase, during which the network is trained, or a subsequent operation phase. We say that a neural network learns off-line if the learning phase and the operation phase are distinct. A neural network learns on-line if it learns and operates at the same time. Usually, supervised learning is performed off-line, whereas unsupervised learning is performed on-line.

IV. FUZZY VS NEURAL NETWORKS

Every intelligent technique has particular computational properties (e.g. ability to learn, explanation of decisions) that make them suited for particular problems and not for others.

For example, while neural networks are good at recognizing patterns, they are not good at explaining how they reach their decisions. Fuzzy logic systems, which can reason with imprecise information, are good at explaining their decisions but they cannot automatically acquire the rules they use to make those decisions. Hybrid systems are also important when considering the varied nature of application domains. Many complex domains have many different component problems, each of which may require different types of processing.

If there is a complex application which has two distinct sub-problems, say a signal processing task and a serial reasoning task, then a neural network and an expert system respectively can be used for solving these separate tasks. The use of intelligent hybrid systems is growing rapidly with successful applications in many areas including process control, engineering design, financial trading, credit evaluation, medical diagnosis, and cognitive simulation. In theory, neural networks, and fuzzy systems are equivalent in that they are convertible, yet in practice each has its own advantages and disadvantages. For neural networks, the knowledge is automatically acquired by the back propagation algorithm, but the learning process is relatively slow and analysis of the trained network is difficult (black box).

But since, in general, knowledge acquisition is difficult and also the universe of discourse of each input variable needs to be divided into several intervals, applications of fuzzy systems are restricted to the fields where expert knowledge is available and the number of input variables is small. To overcome the problem of knowledge acquisition, neural networks are extended to automatically extract fuzzy rules from numerical data.

V. NEUROFUZZY SYSTEM

The combination of fuzzy logic and neural networks constitutes a powerful means for designing intelligent systems. Domain knowledge can be put into a neuro-fuzzy system by human experts in the form of linguistic variables and fuzzy rules. When a representative set of examples is available, a neuro-fuzzy system can automatically transform it into a robust set of fuzzy IF-THEN rules, and thereby reduce the dependence on expert knowledge when building intelligent systems. [5][6]A neuro-fuzzy system is essentially a multi-layer neural network, and thus it can apply standard learning algorithms developed for neural networks, including the back-propagation algorithm (Lin and Lee, 1991; Nauck et al, 1997;). When a training input-output example is presented to the system the back-propagation algorithm computes the system output and compares it with the desired output of the training example. The difference (also called the error) is propagated backwards through the network from the output layer to the input layer. The neuron activation functions are modified as the error is propagated. To determine the necessary modifications, the back propagation algorithm differentiates the activation functions of the neurons The strength of neuro-fuzzy systems involves two contradictory requirements in fuzzy modeling: interpretability versus accuracy. In practice, one of the two properties prevails. The neuro-fuzzy in fuzzy modeling research field is divided into two areas: linguistic fuzzy modeling that is focused on interpretability, mainly the Mamdani model; and precise fuzzy modeling that is focused on accuracy, mainly the Takagi-Sugeno-Kang (TSK) model.

SUMMARY

This paper gives an introduction to concepts in fuzzy system, neural networks and a overview about neuro-fuzzy system. Fuzzy logic provides effective tools for modeling uncertainty in human reasoning, while neural networks are effective in training the data sets. Neuro-fuzzy systems combine the advantages of both paradigms for the purpose of classification, acquisition of knowledge to yield an optimal solution to the given problems.

References

[1] J. J.Buckley. “Fuzzy Complex numbers”,Fuzzy sets system.pp.333-345,1989.

[2] S.Abe,M.S.Lan. “Fuzzy rules extraction directly from numerical data for function approximation”.IEEE Trans.Syst.Man Cyben.,25(1),pp.119-129,1995.

[3] M. F. Azeem, M. Hanmandlu, N. Ahmad.”Generalization of adaptive neuro-fuzzy inference systems”, IEEE Trans.Neural Networks 11(6), pp.1332-1346, 2000.

[4] E.Czogala and J.Leski.”Neuro-Fuzzy Intelligent Systems, Studies in Fuziness and Soft Computing”,Springer Verlag,Germany,2000.

[5] T.C.Lin, C.S.Lee,”Neural Network Based Fuzzy Logic Control and Decision System”, IEEE Transactions on Computers, 1991, Vol.40, no.12.pp.1320-1336.

[6] D.Nauck,R.Kurse,”Neuro-Fuzzy Systems for function Approximation”,4th International workshop Fuzzy-Neuro Sytems,1997.

[7] L.A.Zadeh,”Fuzzy Sets”, Information and Control, 1965, Vol.8, pp.338-353.

[8] Fuzzy control programming. Technical report, International Electrotechnical Commission, 1997.

[9] J. Mendel. Fuzzy logic systems for engineering: a tutorial. Proceedings of the IEEE, 83(3):345-377, Mar 1995.