Design of the Library for the Framework

Contents

- Introduction
- Structure of the System
- Generating Knowledge Base
- Learning Algorithm for ANN
- Neuro-Fuzzy Classification Architecture
- Conclusion
3.1 Introduction

This chapter focuses on design of the library of the framework developed in this research. The design forms a base for development of a library that includes source codes for artificial neural network and fuzzy logic. The resulting works as a core component of the interactive framework to automatically develop three different types of systems in the respective areas of artificial neural network, fuzzy (type 2) logic and neuro-fuzzy advisory systems. Furthermore the chapter also depicts in detail development of design and source codes for the components of artificial neural network, fuzzy logic (type 2). The library is designed and developed in such a manner that it generates a framework that incorporates the facility of hybridizing the two components to produce expert neuro-fuzzy advisory system in the given domain area.

The chapter discusses logical view of the neuro-fuzzy library in its introduction section. Further it describes the structure of the system, generation of knowledge base, and neuro-fuzzy architecture along with learning algorithms. The objective of this chapter is to present detail structure and background of the system which is presented in this thesis. In order to achieve this objective a detail study is done on how the neuro–fuzzy methodology is used to extract fuzzy rules and sets from numerical input and output data of artificial neural network. Hence, it was necessary to measure efficiency of neuro-fuzzy approach and its decisive role in developing expert systems while designing the library. During literature survey in chapter 2 we discussed about different major kind of neuro-fuzzy systems and models used to develop these systems; all these models had basic aim to incorporate learning mechanism of neural network to that of fuzzy inference. The operation performed by the system is expressed as linguistic fuzzy expression and learning scheme of neural network is used for the training of the system. Hence we can divide the neuro-fuzzy system into two core parts namely fuzzy logic and artificial neural network as shown in Figure 3.1. With intersection of these two core components, neuro-fuzzy system hybridization is obtained.
3.2 Structure of the System

The origin of the neural fuzzy inference is to incorporate the artificial neural network properties such as learning and parallelism into fuzzy inference system (used widely for expert systems, decision support systems, neuro-fuzzy controllers and predictive systems). Neuro-fuzzy inference system realizes the fuzzy inference as shown by JSR Jang[13] and benergi[9]. We can observe that the architecture of the system is parallel and fuzzy inference exploits the same learning algorithm used artificial neural network, for example back propagation. The fuzzy inference can be implemented into different manner, one is to use one artificial neural network that realize the whole fuzzy inference mechanism for the specified system (neuro-fuzzy) and the second is to use separate artificial neural network for each of the specified fuzzy rules of the system (fuzzy-neural) and the final result is fuzzy inference of several combined result of artificial neural networks. Since the second approach requires significant amount of time and also computational cost increases in such kind of problem domain, most probably the first approach is followed. In this thesis, first approach is studied in detailed and then implemented in the framework which acts as core basic technology for each of its outputted system. Lin and lee[1] worked on neuro–fuzzy system their approach replaced the weighted sum of the McCulloh-pitts neuron by applying corresponding fuzzy function for the weighted sum. But they were unable to provide any training laws for the system which was later developed by Sun[2]. As mentioned above neuro-fuzzy system consists of a fuzzy system and artificial neural network. The fuzzy inference system can be either fuzzy inference block that processes linguistic information for the corresponding artificial neural network or in a vice-versa manner a neural network can drive the parameter for the fuzzy inference block. This mechanism can be induced on the type of problem in the study. The framework presented in thesis allows all the three kind
of functionalities for development of neuro-fuzzy system for the problem under study. Another common approach for the neuro-fuzzy system is to fuzzify the learning algorithm for artificial neural network as shown by Huntsberger et al [24], Tsao et al.[5] chung and lee[6], Vuorimaa[17], sajja[16] and various other researchers working in the field of neuro-fuzzy system.

### 3.3 Generating Knowledge Base

Fuzzy rules compromise of the fundamental part of the knowledge base in any fuzzy inference system. Researchers have proposed a number of variations to rules depending upon the usage of the system and its derivation. The rules which are used for neuro-fuzzy system are discussed as under:

Fuzzy rules for neuro-fuzzy systems: A fuzzy inference with two input and single output can be categorized in four rules using mamdani fuzzy model are described as under:

1. If X is small & Y is small then Z is negative large
2. If X is small & Y is large then Z is negative small
3. If X is large & Y is small then Z is positive small
4. If X is large & Y is large then Z is positive large

Where X, Y and Z are the linguistic variables. X and Y assume value large and small. The fuzzy output for the variable Z is defined separately for each of the inputted rule. In the first rule consequence of the rule is negative large, consequence of second rule is negative small, consequence of the third rule is positive small. The antecedent part of the fourth rule uses the large value for both X and Y, when firing rule strength is calculated by measuring the distance between input value and the reference value[21]. The consequence is positive large. Where the neuro-fuzzy systems are used as expert systems or classifiers the fuzzy rules have IF THEN form.
We can divide input variables to several linguistic variables which may be commonly used for several fuzzy rules. Partitions are created depending upon fuzzy rules and fuzzy membership functions. There are mainly two kinds of partitioning schemes one is fixed partitioning here fuzzy sets cannot be tuned and second is adaptive partitioning scheme where the fuzzy sets can be tuned. The main advantage is membership functions are flexible and can be fine tuned according to the requirement. Hence fuzzy rules are interesting features that produces complex rule bases.

### 3.4 Learning Algorithms for ANN

Artificial neural network incorporates different kind of learning mechanisms to learn the training patterns. Following are the most commonly used training algorithms for learning:

**Back propagation**: This is the most commonly used algorithm in most of neuro-fuzzy systems. The back propagation has gradient descent approach for multi layer perceptron problem. Its main objective is to look for minimum for the error function in the weight space using gradient descent approach. The combination of the weights which minimizes the error function is considered to be a solution for the learning problem as shown by D. Nauck et.al.[3]. As the gradient of the error function is to be calculated at each iteration step, it is necessary to ensure the continuity and differentiability of the error function. Sigmoid is a differentiable function hence it is popularly used with back propagation algorithm. The sigmoid function with range 0 to 1, $S_c : \mathbb{IR} \rightarrow (0,1)$ is defined by $S_c(x) = 1/(1 + e^{-cx})$, where $c$ is a constant that can be selected arbitrarily. Since the function sigmoid is differentiable function it is also continuous and hence it is the most suited for back propagation algorithm. Hence various neuro-fuzzy system uses this gradient descent algorithm[7]. Normally the users first supply fuzzy parameters through interactive fuzzy interface and the rules are then applied which are processed by the back propagation algorithm. The only considered drawback with this problem is that it is very slow in convergence near the minima[20], which can be avoided under controlled condition.
Majority of neuro-fuzzy system uses backpropagation algorithm, hence the framework developed using the library will have backpropagation learning method as its main methodology for different kind of neuro-fuzzy system it can generate.

**Learning Vector Quantization:** It is a supervised version of vector quantization that can be used when we have appropriate training set data. This learning technique uses the class information to reposition the vectors slightly, so as to improve the quality of the classifier decision regions. It is generally used in pattern classification problems.

**Orthogonal Least Squares:** The orthogonal least square method is basically used for radial basis function networks. Wang and mendel[18] used it to allocate the so called fuzzy basis function. The neuro-fuzzy system that uses fuzzy basis function works similar to a radial basis function network where the rule firing strength is normalized. The initial fuzzy function is constructed of many fuzzy basis functions as given input output pairs from which the most optimal are selected in the final system using the orthogonal least squares (OLS) approach.

**Other Hybrid Learning Algorithms:** Researcher Nie and Linkens[12] integrated fuzzy inference system with a radial basis function networks (RBF). The algorithm proposed by them was divided into two phases. In the first phase self organizing map (SOM) is used to determine the functional units of radial basis function in input space. Each unit has fixed amount of width allocated to it and new units are generated based on heuristic which are later associated with fuzzy rules. In the second phase a gradient descent method is employed to adjust weights of the output singletons optimally.

Lin and Lee[1] have also proposed hybrid learning method that uses kohenen SOM algorithm but with a modification that the center of membership functions are obtained by dividing each of the input and output linguistic variable in predetermined order of cluster. Fuzzy rules are deduced by training data set fed to the artificial neural network, and then center and width of fuzzy sets are trained using any gradient descent algorithm.
Another approach was to modify the k-means algorithm as shown by carpenter et.al.[7],[8], the method use to linearly relate the distance of input and output of the map. The output singleton and center of membership function are trained by gradient descent algorithm.

ANFIS as shown by jang[14] and jang[15] uses a combination of the least square method and gradient descent learning approach. The system is initialized with various membership function and fuzzy rule base. The learning mechanism comprises of two separate passes. In forward pass the consequent are determined by the least square method and in the backward pass antecedent are updated by gradient descent algorithm in backward pass[4].

### 3.5 Neuro-Fuzzy Classification Architecture

The traditional neuro-fuzzy system can be implemented as five layer structure as shown in Figure 3.2 which is common architecture for most of the neuro-fuzzy inference system.

**Fuzzification Layer:** This layer consists of linguistic variables and uses the fuzzy rules as mentioned in section 3.3 while generating a knowledge base. The crisp inputs are fuzzified by using the membership functions of the linguistic variables. Usually triangular, trapezoidal or bell shaped membership functions are used. Researchers Takagi et.al[10] and Horikawa et.al [19] have modified and developed new membership function for the optimal output for the problem under specific domain. The fuzzy rules are evaluated and the inference is applied on to one of the node of artificial neural network. These nodes are further simulated by artificial neural network by using algorithm like backpropagation for training purpose.

**Defuzzification Layer:** Before the artificial neural network processes the input from fuzzification layer are converted into crisp values as the machine learning requires crisp values for processing. The defuzzifier includes components which consist of Defuzzification methods like Centroid, Mean, Median and Mode of Maximum, Bisector etc. It also has type reducer that has mechanism to interoperate between
type 1 and type 2 fuzzy systems. The defuzzifier is also used again, if the required output is in crisp format.

**Combination of Firing Strength:** If the output of the neural network has more than one consequence then this layer performs the operation of evaluating the firing strength of a particular node based on a threshold value, to decide which of the output broad categories obtained from the artificial neural network has more priority. On the basis of priority the fuzzy inference (advice) can be generated in the particular domain area.

**Re-Fuzzified Output:** In this layer rule bases for type 1 and type 2 fuzzy inferences of the system are applied before the final output is fed back to the system. If the required output is desired in crisp format then further defuzzification has to be made to convert into crisp logic. But most of modern day expert systems require output in human understandable format, hence the output produced through fuzzy inference mechanism should be considered. This output is feedback to the interactive fuzzy interface that communicates with the end user.

![Figure 3.2: Architecture of Typical Neuro-Fuzzy System](image)

The membership functions (MF) and defuzzification method employed by various neuro-fuzzy systems are listed as under Table 3.1.
### Table 3.1: Architectural Properties of Various Neuro-Fuzzy Systems

<table>
<thead>
<tr>
<th>Neuro-Fuzzy System</th>
<th>Antecedent MF</th>
<th>Consequent MF</th>
<th>Defuzzification Method</th>
<th>Self Constructing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vuromiaa[17]</td>
<td>Triangular</td>
<td>Singleton</td>
<td>Weighted Average</td>
<td>Yes</td>
</tr>
<tr>
<td>Lin[1]</td>
<td>Bell Shaped</td>
<td>Bell Shaped</td>
<td>Center of Area, Weighted Average</td>
<td>Yes</td>
</tr>
<tr>
<td>Jang[14]</td>
<td>Bell Shaped</td>
<td>Singleton</td>
<td>Weighted Average</td>
<td>No</td>
</tr>
<tr>
<td>Kohenen[23]</td>
<td>Trapezoidal, Bell Shaped</td>
<td>Singleton</td>
<td>Weighted Average</td>
<td>No</td>
</tr>
<tr>
<td>Bezdek[11]</td>
<td>Bell Shaped</td>
<td>Singleton</td>
<td>Weighted Average</td>
<td>Yes</td>
</tr>
<tr>
<td>Kosko[22]</td>
<td>Triangular</td>
<td>Triangular</td>
<td>Center of Area</td>
<td>No</td>
</tr>
<tr>
<td>Horikawa[19]</td>
<td>Bell Shaped</td>
<td>Singleton, Monotonic</td>
<td>Weighted Average, Tsukamoto</td>
<td>No</td>
</tr>
<tr>
<td>Nauck[3]</td>
<td>Monotonic</td>
<td>Monotonic</td>
<td>Tsukamoto</td>
<td>No</td>
</tr>
<tr>
<td>Nie[12]</td>
<td>Bell Shaped</td>
<td>Singleton</td>
<td>Weighted Average</td>
<td>Yes</td>
</tr>
<tr>
<td>Wang[16]</td>
<td>Bell Shaped</td>
<td>Singleton</td>
<td>Weighted Average</td>
<td>Partial</td>
</tr>
<tr>
<td>Benerji[9]</td>
<td>Triangular</td>
<td>Triangular</td>
<td>Local Mean of Maximum</td>
<td>No</td>
</tr>
</tbody>
</table>

3.6 Conclusion

From the architecture surveyed of existing neuro-fuzzy systems, it is deduced that most of the neuro-fuzzy systems uses a gradient descent learning algorithm. The fuzzy rules and fuzzy sets are fine tuned using fuzzy inference system that uses
various membership functions as required by the problem under study. Hence while designing the library for neuro-fuzzy system, it is ensured that proper usage of gradient descent algorithm merged with fuzzy inference with proper membership functions. Hence this gives a valid support for development of a generic library that has both the components of neuro-fuzzy systems. The library should have various gradient descent algorithm like back propagation along with fuzzy membership function. Here to strengthen the library of the architecture of the neuro-fuzzy system, the scope of fuzzy system is extended to type 2 fuzzy system. Type 2 fuzzy sets are much more flexible compared to conventional fuzzy sets, hence it easy for linguistic variables to ascertain values which are very vague or unclear for the system under the consideration. Type 2 fuzzy sets posses their own membership function and inference engine. An extra component known as type reduce is added to inter operate between type 1 and type 2 fuzzy sets. Thus the generic library contains method for learning algorithms and various membership functions for fuzzy inference to generate different types of neuro-fuzzy systems for the given problem area.

In coming chapters the organization and development of the library along with development of an interactive framework that uses the library to generate different types of neuro-fuzzy system is discussed.

References


Chapter 3: Design of the Library for the Framework

Design of Neuro-Fuzzy Advisory System Using Type 2 Fuzzy Logic


