Modular Neuro-Fuzzy Networks Used in Explicit and Implicit Knowledge Integration

Ciprian-Daniel NEAGU[1], Vasile PALADE[2]

[1]Laboratorio di Intelligenza Artificiale e Robotica, Politecnico di Milano
Piazza L. da Vinci 32, I-20133 Milano, Italy
Neagu@fusberta.elet.polimi.it
[2]Oxford University, Computing Laboratory,
Wolfson Building, Parks Road, Oxford, OX1 3QD, UK
Vasile.Palade@comlab.ox.ac.uk

Abstract

A framework of new unified neural and neuro-fuzzy approaches for integrating implicit and explicit knowledge in neuro-symbolic systems is proposed. In the developed hybrid system, training data set is used for building neuro-fuzzy modules, and represents implicit domain knowledge. On the other hand, the explicit domain knowledge is represented by fuzzy rules, which are directly mapped into equivalent neural structures. Three methods to combine the explicit and implicit knowledge modules are proposed.

1. Introduction

In recent years, the hybrid neural systems have drawn an increasing research interest. This approach has been successfully used in various areas, such as speech/natural language understanding, robotics, medical diagnosis, fault diagnosis of industrial equipment, and financial applications (Kosko 1992, Rocha 1992, Takagi 1994, Sima and Cervenka 1997, Wermter and Sun 2000). The reason for studying hybrid neural systems is based on successful applications of subsymbolic knowledge-based systems, particularly the neuro-fuzzy networks (Pedrycz, Kandel, and Zhang 1997) and the advantages of symbolic knowledge-based systems. In hybrid systems, connectionist tools can be interpreted as hardware, and fuzzy logic as software implementation of human reasoning: modular structures of explicit and implicit knowledge build homogenous inductive and deductive learning and reasoning systems.

The fundamental concepts and methods used in our approach are based on the neuronal fuzzy model MAPI (Rocha 1992). Three specific methods are presented, based on fuzzy operators, supervised or unsupervised gating networks. Steps for building the hybrid system, called NEIKeS (Neural Explicit and Implicit Knowledge-based expert System, Neagu et al. 2001) are described.

The implicit knowledge is defined as a connectionist module-based representation of learning data. The explicit knowledge module of the hybrid system is implemented as hybrid fuzzy neural networks, and has the role to adjust the performances of implicit knowledge modules.

2. Neuro-Fuzzy Knowledge Representation

The last ten years have produced a tremendous amount of research on both symbolic and connectionist fields. In connectionist systems, unlike symbolic models, learning plays a central role. The directions of research explored both, high-level connectionism (applied to natural language processing or commonsense reasoning (Sun 1994)) and hybrid systems (Khosla and Dillon 1997).

The two approaches can be used in complementary way. The hybrid intelligent systems combine connectionist and symbolic features. In such systems, the learner first inserts symbolic information of some sort into a neural network. Once the domain knowledge has a neural representation, training examples are used to refine the initial knowledge. Finally, it processes the output for given instances and, using specific methods (Omlin and Giles 1996, Benitez, Castro, and Requena 1997, Neagu and Palade 2000, Palade, Neagu and Patton, 2001), extracts symbolic information from trained network, to explain the computed outputs and to interpret the refined connectionist knowledge.

The connectionist integration of explicit knowledge and learning by examples appears to be the natural solution of developing connectionist intelligent systems. Explicit and implicit rules should be represented in a neural manner using (Buckley and Hayashi 1995, Fuller 1999) fuzzy (FNN) or hybrid neural networks (HNN), MLP (multilayer perceptron, Rumelhart and McClelland 1986), or neuro-fuzzy nets (NEFCLASS, Nauck and Kruse 1998). While fuzzy logic provides the inference mechanism under cognitive uncertainty, neural nets offer the advantages of learning, adaptation, fault-tolerance, parallelism and generalization.

This leads to the three steps in a fuzzy neural computational process: (1) development of neuro-fuzzy models, and (2) modeling the synaptic connections, which incorporate fuzziness into a neural net, and (3) adjusting the weights through learning, respective mapping algorithms. The system considered is a multi-input single-output fuzzy system (MISO). The MAPI neuron (Rocha 1992) is used to implement the fuzzy neuro-symbolic processing.

2.1. The Implicit Knowledge Modules

We define the *implicit knowledge* as the knowledge represented by neural networks. The representation is based on the numerical weights of the neurons connections. The IKM is a multi-layered neural structure based on an input layer which perform the membership degrees of the current values, a fully connected three-layered FNN2 (Fuller 1999), and a defuzzification layer (fig. 1). The FNN of type 2 (FNN2), implementing IF-THEN fuzzy rules, is characterized by fuzzy inputs and outputs, and crisp weights. The input nodes of the FNN2 are MAPI neurons, parameterized to implement membership functions of the term set for each linguistic input. Jang (1993) proposed similar structures (ANFIS) to approximate the membership functions. The objective for IKM is to learn the fuzzy rules.

2.2. The Explicit Knowledge Modules

We define the *explicit knowledge* as a knowledge base represented by neural networks, which are computationally identical to a fuzzy rules set, and are created by mapping the given fuzzy rules into hybrid neural networks. The fuzzy rule set is described (Buckley and Hayashi 1995) as a discrete fuzzy rule-based system DFRBS. The intrinsic representation of explicit knowledge is based on fuzzy neurons in a MAPI implementation. The numerical weights corresponding to the connections between neurons are computed (Buckley and Hayashi 1995, Fuller 1999) using Combine Rules First Method, or Fire Each Rule Method.

3. Implicit and Explicit Knowledge Integration

The introduction of the modular networks (MNN) into fuzzy systems provides new insights into the integration of explicit and implicit knowledge. MNN is a connectionist architecture that allows each module to exhibit its "opinion" about entries, offering advantages over a single neural network, in terms of learning, and generalization (Haykin 1994, Jacobs 1991, Langari 1993).

The global network (GN) of our approach is a modular structure including two different "points of view" about the problem: the implicit knowledge (a trained neuro-fuzzy network), and the explicit knowledge, (a collection of special neural networks equivalent to some rules proposed by human experts, Neagu and Palade 1999). The IKM is responsible for generalization, and processing noisy cases. The EKM is developed in a top-down manner, mapping available explicit rules in HNN structures.

We propose three strategies to combine IKM and EKM in a global hybrid system: Fire Each Module (FEM), Unsupervised-trained Gating Network (UGN), and Supervised-trained Gating Network (SGN). The first strategy is an adapted Fire Each Rule method (Buckley and Hayashi 1995) for modular networks. The second strategy proposed a competitive-based aggregation of the EKM and IKM outputs, while the third strategy uses a supervised trained layer to process the overall output of the modules.

3.1. Fire Each Module Strategy

The proposed FEM strategy is the simplest mode to integrate IKM and EKM with fuzzy output. A general approach form of this modular structure (fig. 2) is proposed in (Neagu and Palade 1999). After off-line training phase applied to implicit neuro-fuzzy module, the general output of the system is composed as a T-conorm (Zadeh 1983) of fuzzy outputs of each module: the four-layered IKM structure for global network and the EKM (implemented using combine rules first or fire each rule method). The system is viewed as equivalent to a set of given fuzzy rules: the overall output is computed using firing (implicit and explicit) rules first method (Buckley Hayashi 1995, Fuller 1999). The method of combining the membership degrees provided by IKM (ξ_i values, i=1,2,...,m) and EKM (y_i values, i=1,2,...,m), would be done component wise so let:

(1) $y'_i = \text{T-conorm}(\xi_i, y_i)$, i=1,2,...,m,

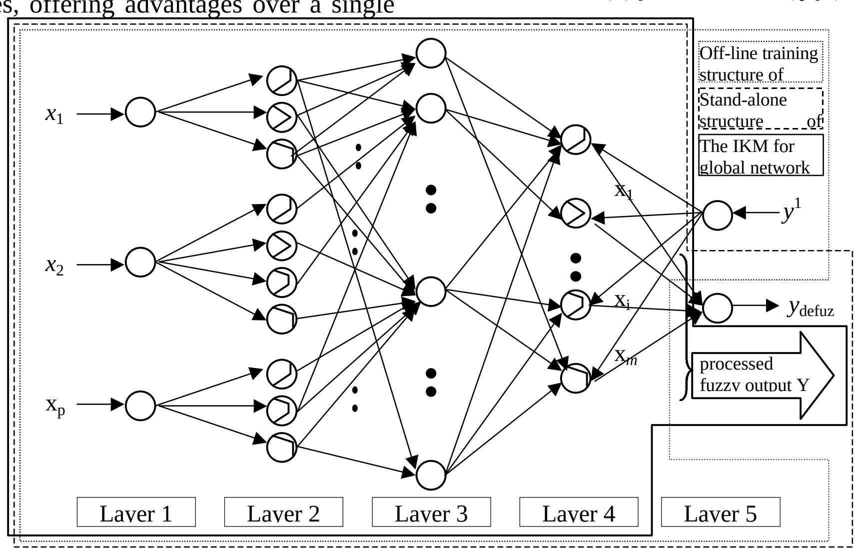

Fig.1. *Implicit Knowledge Module implemented as FNN2/HNN.*

for some aggregating operator (in particular MAX). In the hidden aggregative layer (AL), all the weights are set to one, and the neurons aggregate the specific computed membership degrees ξ_i and y_i as implicit, respective explicit opinion about the current output to be described with B_i-th fuzzy term (where the terms set describing the output is $B=\{B_1,...,B_i,...,B_m\}$). Practically, the inputs for the MAPI deffuzifier describe the shape of the fuzzy output. The final neuron is a MAPI device, which computes the crisp value of the output, using, for example, the center of gravity method. The FEM methodology proposed consists of:

1. Identification of I/O linguistic variables. The variables are represented by fuzzy sets, mapped in MAPI units.
2. The IKM is built and train as a five-layered FNN (the off-line training structure of IKM).
3. From the hidden part of the IKM, the most relevant rules are extracted, using Relative Rule Strength method, Effect Measure Method, Causal Index Method.
4. We construct a set of possible explicit rules in a given problem with the help of a human expert, using both, external rules and those already extracted at step 3, as the most voted and trusty dependencies between inputs and output. All these rules are mapped into EKM. Some explicit rules could have just a part of identified inputs in the rule premise, represented as active neurons, while the rest of the input neurons will be set as inactive.
5. The four-layered IKM and the EKM structures (without the deffuzifier MAPI final neuron) are embedded into the architecture described in fig. 2, for which the combining hidden layer AL and the MAPI deffuzifier are adapted.
6. After an incremental loop sequence based on steps 2 to 5 (which could be used as a knowledge acquisition procedure), the global network is ready to be used.

The incremental loop sequence consisting of steps 2 to 5 could be refined on the basis of combining already given fuzzy rules and training data set as follows. IKM is designed by mapping some external fuzzy rules in the hidden HNN, which further learning with training samples is based on. This way the knowledge is kept at the sub-symbolic level. The main goal of the approach is not just to reduce training period, but also to improve the generalization abilities of the network. The disadvantages consist in both, redistribution of symbolic a priori knowledge (or at least building haloes of initial rules), and necessity of a new refinement of the final incorporated knowledge. This strategy follows the variations of *concept support techniques* (Prem et al. 1993, Wermter and Sun 2000), parameterized to insert a priori knowledge:

- Inserting some rules about a subset of cases of desired input-output mapping, and learning the training samples.
- Inserting the symbolic concepts believed to be relevant for the problem solution and training by supporting the relevant concepts.
- Inserting explicit rules as above, followed by a training phase, in which the used hidden units are different from those designed in first phase.

3.2. Unsupervised-trained Gating Network Strategy

The proposed basic configuration consists of two general types of networks: expert networks (implemented by EKM and neuro-fuzzy IKM) and a gating network. A regular modular network considers expert networks competing to learn the training patterns, and the gating network mediating the competition (Haykin 1994, Jacobs, Jordan and Barto 1991, Langari 1993). The proposed modular architecture (fig. 3) uses neural explicit and implicit knowledge modules, and the gating network to vote the best combination of fuzzy terms computed by expert nets, to describe the linguistic output. The EKM and IKM structures are developed and, respectively, trained.

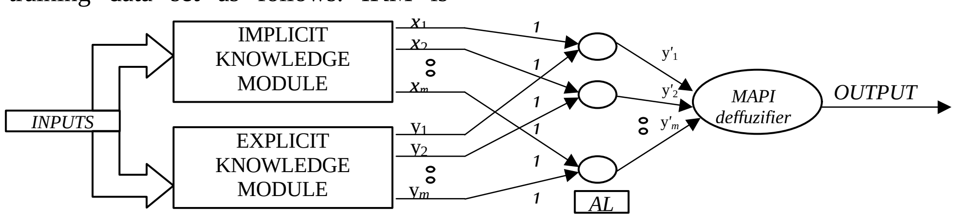

Fig.2. *Integration of explicit and implicit knowledge modules in the global network according to FEM strategy.*

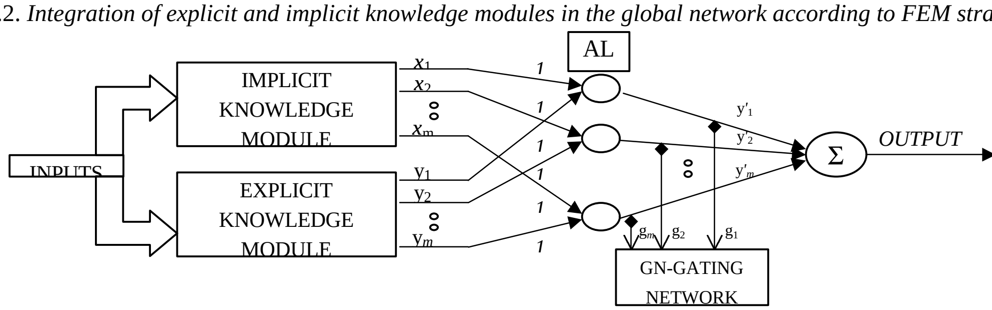

Fig.3. *Integration of explicit and implicit modules using an unsupervised-trained gating network (UGN strategy).*

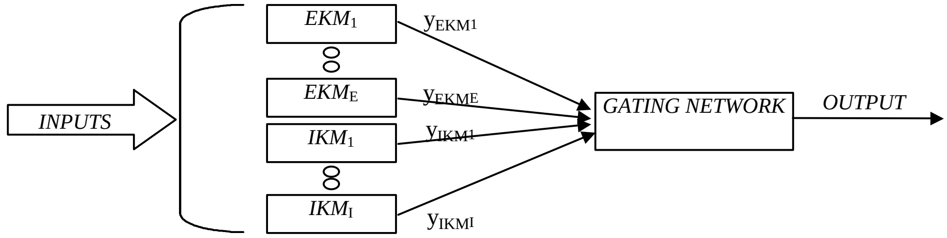

Fig.4. *Integration of explicit and implicit modules using a supervised-trained gating network (SGN strategy).*

The gating network is trained with the constrain to have as many output neurons as there are fuzzy terms chosen to describe the linguistic variable Y as the output of global network. The membership degrees provided by both, IKM structure (ξ_i, i=1,2,...,m), and EKM structure (y_i, i=1,2,...,m), are aggregated according to the eq. (1) by MAPI neurons acting as MAX T-conorm (aggregation layer AL). The trained gating network is a distribution model of EKM/IKM membership degrees. The gating network is a single layer of m output neurons (Hashem 1997), each one having m inputs. The activation function of its output neurons is a *softmax* transformation. In the training process of gating network, for each vector $[x'_1..x'_m]$ processed by AL, the activation g_i of the i^{th} output neuron is related to the weighted sum of the inputs applied to that neuron. Consequently, the activations of the output neurons in gating network are nonnegative and sum to one:

$$(2)\ \ 0 \le g_i \le 1 \text{ and } \sum_{i=1}^{m} g_i = 1 .$$

The additional advantage gained by using the gating network is the implicit defuzzification of the overall output:

Proposition 1: Let $[x_1,...,x_p]$ be the current input of the system and $[y'_1,...,y'_m]$ be the current output of the aggregation layer and let consider the gating network trained with unsupervised *compet* algorithm (Hagan and Beale 1996). Then the overall output y of UGN, computed by a *softmax* transformation, is a crisp value, the defuzzified output of the model.

Proof: Let's consider the output of the system as a first order Sugeno model (the consequent part of rules is a linear regression model, Sugeno and Kang 1988, Takagi 1994):

$$(3)\ R_i\text{: IF } X_1 \text{ is } A_{i1} \text{ AND } X_2 \text{ is } A_{i2} \text{ AND ... AND } X_p \text{ is } A_{ip}$$

$$\text{THEN } y'_i = \sum_{j=1}^{p} b_{ij} x_j ,$$

where A_{i1}, A_{i2},..., A_{ip} are fuzzy sets having associated matching functions μ_{Ai1}, μ_{Ai2},..., μ_{Aip}, b_{ij} are real-valued parameters, y'_i is the local output of the model due to rule R_i, i=1,2,...,m. The total output of the Sugeno model is a crisp value defined by the weighted average:

$$(4)\ \ y = \frac{\sum_{i=1}^{m} h_i y'_i}{\sum_{i=1}^{m} h_i} .$$

The weight h_i implies the overall truth value of the premise of rule R_i for current input, and is calculated as:

$$(5)\ h_i = (\mu_{Ai1}(x_1) \wedge \mu_{Ai2}(x_2) \wedge ... \wedge \mu_{Aip}(x_p)),$$

where ^ is a conjunctive T-norm.

Let's now consider the input vector $[x_1…x_p]$ applied to the system (fig. 3). Then, the output of the system is:

$$(6)\ \ y = \sum_{i=1}^{m} g_i y'_i .$$

The expressions of the outputs of Sugeno model (3) and our proposed structure (6) are similar: each rule in Sugeno model could be an explicit rule into EKM, or a particular IKM way, involving one hidden neuron.

As a consequence of the proof for the *Proposition 1*, the relative weight of the i^{th} neuron in GN will be then:

$$(7)\ \ gi = \frac{h_i}{\sum_{i=1}^{m} h_i} .$$

In conclusion, GN, proposed to combine the outputs of the aggregating layer, acts as a special defuzzifier.

The methodology proposed to build the global network architecture is partially similar to the FEM methodology:

1. Steps 1 to 4 are similar to those described for FEM.
2. The four-layered IKM and the EKM (without deffuzifier) are embedded into the architecture (fig. 3), for which the combining hidden layer AL and the deffuzifier MAPI-based unit are adapted.
3. The gating network is trained (by *competitive* algorithm) using the AL outputs computed for the training data set.
4. After the knowledge acquisition, the global network is ready to be used as a classifier/prediction tool: the final crisp output is a *softmax* transformation (6).

3.3. Supervised-trained Gating Network Strategy

The proposed structure contains expert networks represented by a defined number of EKMs and IKMs solving various sub-problems of the main task, and a supervised trained network mediating their outputs' combination. After training, the expert networks compute different functions, each of them mapping different regions of the input space. Each defuzzified output of expert networks is an input for the final layer. The supervised training process of the final network assures a weighted aggregation of expert networks' outputs with respect to their specialization (fig. 4). The methodology proposed to build the global network architecture consists of:

1. Identification of input and output linguistic variables. The variables are represented by fuzzy sets, mapped in MAPI units. The IKM modules are represented as five-layered fuzzy neural networks and/or MLP networks.
2. The most relevant rules are extracted from the IKM.

3. A set of possible explicit rules about the problem is developed, with the help of a human expert. Each rule is mapped into specific EKM structure. A MAPI-based defuzzifier computes the crisp output of EKMs.
4. The IKMs and EKMs are embedded into SGN (fig. 4).
5. The gating network is supervised trained.
6. The output crisp value is computed as a classification of the best combination of each expert network's behavior.

5. Results and Conclusions

The proposed structures and methods argue the use of connectionist systems in symbolic processing. Since the presented EKMs are identical to Discrete Fuzzy Rule-based Systems (Rocha 1992, Buckley and Hayashi 1995), the homogenous integration of explicit rules and training data sets permits better cover of the problem domain.

We applied with good results the proposed approaches for the prediction of the daily NO_2 maximum concentration for a single representative measuring station (Neagu et al. 2001). The comparison between the different structures used in our study is presented in fig. 4. The constraint of the size of neural networks is solved by the modularity paradigm. EKMs represent explicit rules identified by expert or refined from IKM structures; IKMs are useful for complex problems described by (noisy) data.

The EKM and IKM combination encourages compact solutions for problems described by both, data sets distributed in compact domains in the hyperspace, and isolated data, situated in intersection of compact sub-domains or inhomogeneous intervals. After training, different expert networks compute different functions mapping different regions of the input space. The proposed methods are already applied with promising results in short term air quality prediction (NEIKES: Neagu et al. 2001) and toxicity and carcinogenicity modeling and prediction.

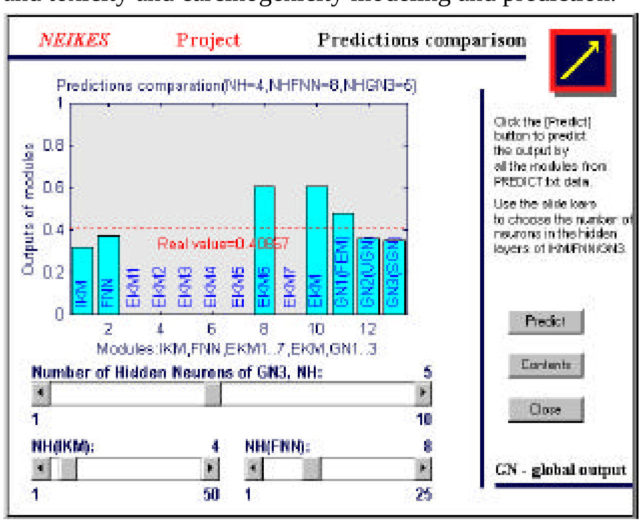

Fig.5. *Comparison between predictions of IKM, EKM and integrating modules for air quality prediction data set.*

References

J.M. Benitez, J.L. Castro, I. Requena, 1997. Are ANNs Black Boxes?, *IEEE Trans on Neural Networks* 8(5):1157-1164.

J.J. Buckley and Y. Hayashi, 1995. Neural nets for fuzzy systems, *Fuzzy Sets and Systems* 71:265-276.

R. Fuller, 1999. *Introduction to Neuro-Fuzzy Systems*, Advances in Soft Computing Series: Springer-Verlag: Berlin.

M.Hagan, M.Beale, 1996. *Neural Network Design*, PWS: Boston.

S. Haykin, 1994. *Neural Networks: A Comprehensive Foundation*, IEEE Press: New Jersey.

S. Hashem, 1997. Optimal Linear Combinations of Neural Networks, *Neural Networks* 10(4):599-614.

R.A. Jacobs, M.I. Jordan, A.G. Barto, 1991. Task decomposition through competition in a modular connectionist architecture: the what and where vision tasks, *Cognitive Science* 15:219-250.

R.J.S. Jang, 1993. ANFIS: Adaptive-network-based fuzzy inference system, *IEEE Trans.Syst.,Man, and Cyb.* 23:665-685.

R. Khosla, T. S. Dillon, 1997. *Engineering Hybrid MAS*, Kluwer.

B. Kosko, 1992. *Neural Networks and Fuzzy Systems*, Prentice-Hall, Englewood Cliffs: NJ.

R. Langari, 1993. Synthesis of nonlinear control strategies via fuzzy logic, in Procs. of American Control Conf.: 1855-1859.

D. Nauck, R. Kruse, 1998. NEFCLASS-X: A Neuro-Fuzzy Tool to Build Readable Fuzzy Classifiers. *BT Tech. J.*16(3):180-192.

C.-D.Neagu,V.Palade,1999. Fuzzy Computing in a Multi Purpose Neural Network Implementation, Procs. Int'l Conf. 6th Fuzzy Days in Dortmund, B. Reusch, ed, Springer Verlag, 697-700.

C.-D. Neagu, V. Palade, 2000. An interactive fuzzy operator used in rule extraction from neural networks, *Neural Networks World Journal* 10(4):675-684.

C.D.Neagu, N.Avouris, E.Kalapanidas, V.Palade, 2001. Neuro-symbolic Integration in a Knowledge-based System for Air Quality Prediction, *Applied Intelligence*, Kluwer (forthcoming).

C.W. Omlin, C.L. Giles, 1996. Extraction of Rules from Discrete-Time Recurrent Neural Networks, *Neural Networks* 9:41-52.

V. Palade, C.-D. Neagu, R.J. Patton, 2001. Interpretation of Trained Neural Networks by Rule Extraction, Procs. of Int'l Conf. 7th Fuzzy Days in Dortmund: 152-161.

W. Pedrycz, A. Kandel, Y.-Q. Zhang, 1997. Neurofuzzy Systems, *International Handbook on Fuzzy Sets and Possibility Theory* (D. Dubois, H. Prade eds.): 311-380, Kluwer Academic.

E. Prem, M. Mackinger, G. Dorffner, G. Porenta, H. Sochor, 1993. Concept Support as a Method for Programming ANN with Symbolic Knowledge, TR-93-04, OEFAI.

A.F. da Rocha, 1992. *Neural Nets: A Theory for Brains and Machines*, Lecture Notes in AI: Springer-Verlag.

D.Rumelhart,J.McClelland,1986.*Parallel Distributed Processing, Explanations in the Microstructure of Cognition*, MIT Press.

J.Sima, J.Cervenka,1997. Neural Knowledge Processing in Expert Systems, TR V-735, ICS, Acad. of Sciences, Czech Rep.

M. Sugeno, G.T. Kang, 1988. Structure identification of fuzzy model, *Fuzzy Sets and Systems* 28:15-33.

R. Sun,1994. CONSYDERR: A Two Level Hybrid Architecture for Structuring Knowledge for Commonsense Reasoning, Int'l Symp on Integr. Knowledge and Neural Heuristics: 32-39.

H. Takagi, 1994. Cooperative system of neural networks and fuzzy logic and its application to consumer products, *Ind Apps of Fuzzy Control&Intell Systems*, Van Nostrand Reinhold: NY.

S. Wermter, R. Sun, 2000. *Hybrid Neural Systems*, Springer Verlag: Heidelberg.

L.A. Zadeh, 1983. The role of fuzzy logic in management of uncertainty in expert systems, *Fuzzy Sets&Sys:* 11(3):199-227.