INTERNATIONAL JOURNAL OF INTELLIGENT CONTROL AND SYSTEMS
VOL.12, NO.2 JUNE 2007 142-149

Hierarchical Knowledge Representation: Symbolic Conceptual Trees and Universal Approximation

Luis Fernando de MINGO, Fernando ARROYO, Miguel Angel DÍAZ and Juan CASTELLANOS

Abstract— **This paper presents a practical example of a system based on neural networks that permits to build a conceptual hierarchy. This neural system classifies an input pattern as an element of each different category or subcategory that the system has, until an exhaustive classification is obtained. The proposed neural system is not a hierarchy of neural networks, it establishes relationships among all the different neural networks in order to transmit the neural activation when an external stimulus is presented to the system. Each neural network is in charge of the input pattern recognition to any prototyped class or category, and also of transmitting the activation to other neural networks to be able to continue with the classification. Therefore, the communication of the neural activation in the system depends on the output of each one of the neural networks, so as the functional links established among the different networks to represent the underlying conceptual hierarchy.**

Index Terms— **Neural Networks, Function Approximation, Hierarchical Architecture.**

1. Introduction

CONCEPTS and categories are being objects to study in practical Psychology and in Artificial Intelligence (AI) years ago. It deals with the research of how the knowledge and meaning are represented in the memory [4]. Why are concepts important?. Concepts or conceptual categories are stable representations, stored in the memory, that permit to treat different samples as members of a same class; without them we will be slaves of particular [5].

From a psychologist point of view, when dealing with objects, situations and actions as members of conceptual categories, the perceived reality is mentally being divided into different groups. A structure is imposed to the world, dividing mentally the reality [3], [4]. Categories seem to be ruled by the cognitive economy principle, this is, to extract essential things from the known world and to represent them in the most economical way in order to minimize the cognitive effort.

According to Collins and Quillian [2], [3], the importance of a concept is not the concept by itself, it is determined by the set of relationships that are established with other different concepts. These relationships can be of two types: Subset and Property. The former type expresses the specialization of some concepts into others, it produces the categorization, allowing to mentally structure the perceived reality.

What could be added from a connectionist point of view to the concept representation in memory? [6]. Why could not it be thought that the concepts are supported by some small sets of neural assemblies? These ones form a mix of specialized neural networks concerning the samples recognition task and they determined if a sample belongs or is classified into a concept.

Concepts are not isolated into the human cognitive system. They are immersed into a hierarchical structure that facilitates, among others, classification tasks to the human cognitive system. This conceptual hierarchy expresses a binary relationship of inclusion defined by the following criterions:

- Inclusion criterion: Each hierarchy node determines a domain included into domain of its father node. Each hierarchy node determines a domain that includes every domain of its son nodes.
- Generalisation-Specialisation criterion: Every node in the hierarchy, has differentiating properties that make it different from its father node, if it exists, and from the others son nodes of its father, if they exist.

This Inclusion relation makes possible to keep the maximal economy criterion in the conceptual hierarchy. The defined inclusion relation makes also possible to represent a conceptual hierarchy as an acycled directed graph in which nodes represent concepts and edges the inclusion relation among them. Edges connecting nodes establishes that the father nodes are situated in the upper hierarchical level of its son nodes. Obviously, concepts are generated in a natural manner and it will not be constrained by these formal restrictions. However, many of the classifications learnt during our school years are built using these rules. In this paper, we will focus our work in this kind of hierarchies, which are learnt with a teacher or in a supervised manner.

Next sections deal with the representation of any conceptual hierarchy using a set of neural networks, permitting the complete or exhaustive classification of an input. Exhaustive classification means that an input belongs, or not, to each one of the supported concepts by the system and to their specializations. The hierarchical architecture is controlled using neural network outputs, that is, depending on the output a different neural network is activated.

2. Problem Description

LET D domain of any samples. Let H a conceptual hierarchy to the elements of the domain D. Let T the

L.F. de Mingo and M.A. Díaz are with Dept. Organización y Estructura de la Información, Escuela Universitaria de Informática - Universidad Politécnica de Madrid, Crta. de Valencia km. 7 - 28031 Madrid, Spain. {lfmingo,mdiaz}@eui.upm.es

F. Arroyo is with Dept. Lenguajes, Proyectos y Sistemas Informáticos, Escuela Universitaria de Informática - Universidad Politécnica de Madrid, Crta. de Valencia km. 7 - 28031 Madrid, Spain. farroyo@eui.upm.es

J. Castellanos is with Dept. Inteligencia Artificial, Facultad de Informática - Universidad Politécnica de Madrid, Boadilla del Monte - 28660 Madrid, Spain. jcastellanos@fi.upm.es

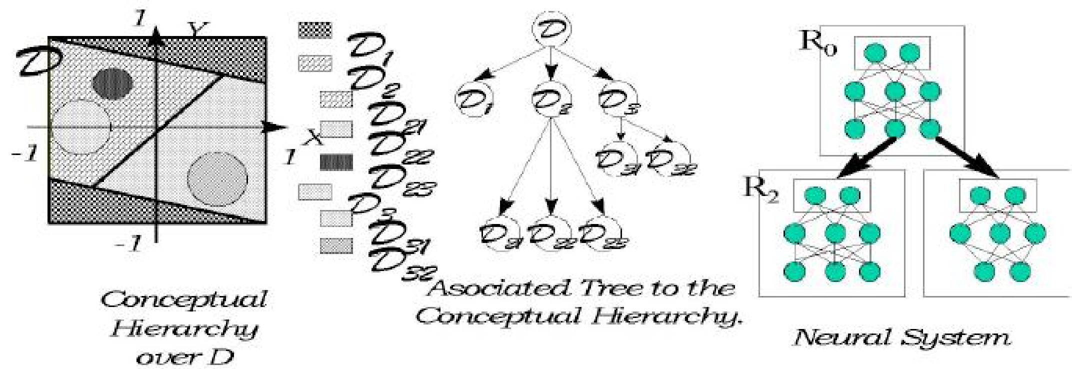


Fig. 1. Conceptual Hierarchy over domain D, and the associated tree.

tree of the conceptual hierarchy H. Each node in the tree T represents a concept in hierarchy H, and an arrow between two nodes implies that the end node is a specialization of the start node.

The nodes belonging to the first level of the tree, associated to the conceptual hierarchy, establish a partition of the input domain D in the following way.

$$D = \bigcup_{i=1}^{N} D_i / \forall i \neq j, D_i \cap D_j = \emptyset \tag{1}$$

Where D_i are the subconcepts corresponding the domain D of a higher level of abstraction in the conceptual hierarchy. Each point or sample in the domain could be classified as belonging to a determined concept of such a level and no other. That is:

$$\forall p \in D, \exists (i,j) \in \{1, \cdots, n\} / p \in D_i \Lambda p \notin D_j, j \neq i \tag{2}$$

Once the classification of the input point $p \in D$ has been established as belonging to one concept of this level, the same classification problem arises; this time concerning the classification of p into the different subconcepts of the concept, provided that the concept had any subconcept, that is, if the following equation is verified.

$$p \in D_i \Lambda D_j = \bigcup_{j=1}^{m} D_{ij}, D_{ik} \cap D_{il} = \emptyset, k \neq l \tag{3}$$

Then, it is verified that:

$$\exists j \in \{1, \cdots\} / p \in D_{ij} \Lambda p \notin D_{ik}, k \in \{1, \cdots\} - \{j\} \tag{4}$$

Therefore, once an input or sample is identified into some pre-established category, into a prototype of a concept belonging to any level of the conceptual hierarchy, the process of classification can end if the recognized concept has not any subconcepts or specializations. Also, it can continue, this case the sample will form a new conceptual category.

It is said that a sample p of the domain D is exhausted classified when the degree of belonging of pattern p to each one of the concepts has been established, for all the concepts in the conceptual hierarchy. Writing this definition in terms of the associated tree, the exhaustive classification of an input $p \in D$ is defined as the path of the tree that goes from the root node to any leave node, in which the classification is ended:

$$D \to D_i \to D_{ik} \to D_{ik \cdots l} \tag{5}$$

With $p \in D_i, p \in D_{ik}, \cdots, p \in D_{ik \cdots l}$.

Next section deals with the construction of a neuronal systems able to solve the problem of exhaustive classification given some samples of a certain domain.

3. Neural Networks Model

THE proposed computational model assumes a dual representation in order to facilitate the information process. One level is related to implement neural circuits in Artificial Neural Networks (Connectionist System) and the other one is related to maintaining of symbolic information [9] about implemented concepts in Artificial Neural Circuits (Symbolic System)[6].

In the Connectionist System, it is possible to answer if a specific input is a member or not of one specific concept or category. Moreover, in [7] it is shown how it is possible to represent a conceptual hierarchy using a supervised connectionist system in such a way that it will permit external sensorial inputs, of given perceptions, to be exhaustively classified as belonging to a whole branch of the conceptual hierarchy. Each neural network, which takes part in the system, is concerned with the association of inputs to a given category and also with the propagation of activity towards others neural networks in order to continue with the input classification. The dual representation supported by this kind of systems makes possible to process information that it is represented in a sub-symbolic or symbolic manner either [8].

In order to represent a conceptual hierarchy into the model, it is assumed that:

- The starting domain of instances and the hierarchy tree are well defined. Hence, the differentiating properties among categories or concepts are known.
- It is possible to build sets of representative instances from every category of the conceptual tree.
- It is possible to choose representatives instances from every category. These instances will be used: (1) for training

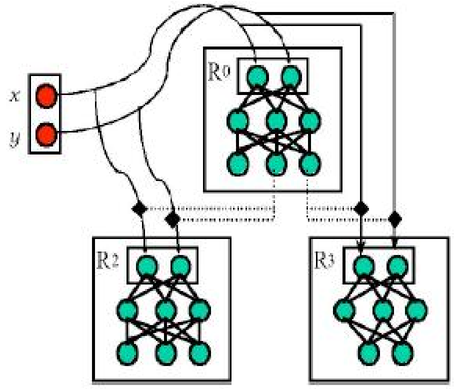


Fig. 2. Neuronal System for the domain D.

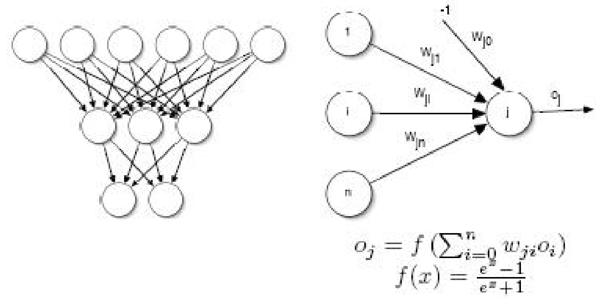


Fig. 3. Multilayer Perceptron Architecture.

the different Artificial Neural Networks that integrate the Connectionist System and (2) for representing essential characteristics of categories in the Symbolic System.

Given an input $p \in D$ to the neural system. To exhaustively classify the input p, a path of the tree is obtained.

$$P = \{D \rightarrow D_i \rightarrow D_k \rightarrow \cdots \rightarrow D_{ik\cdots l}\} \tag{6}$$

It is true that, associated to D and to D_i's there exists a neural network responsible for giving a response concerning to which specialization of the concept the input p belongs to. Except for the end node $D_{ik\cdots l}$ that has no specialization because it is a terminal node.

To begin with, a neural network will be design in order to admit a sample of the domain, subject to study, as an input pattern. Therefore, the number of inputs to such a neural network depend on the dimension of the samples that will be presented to the system: let say that the domain has a valid representation in R^r, so the number of inputs related with the neural networks will be r. The number of output neurons of the neural network will be determined by the number of nodes that the tree T has as a first level, that is, the number of concepts the domain is composed by, n outputs. To identify that an input belongs to some concept the output neuron of the subconcept must be activated, in the path P it will be the i-th neuron.

Till this phase, the input p presented to the system will be propagated through the associated neural network corresponding to the domain D_i and does not to the other neural networks associated to the domains D_j with $i \neq j/i, j \in \{1, \cdots, n\}$. To achieve such exhaustive classification of p, it is continued treating p as a specialization of D_i, until a terminal node is reached, then the classification process of the pattern set is completed.

4. Hierarchical Model

TO establish these partitions over the domain is equivalent to establish a conceptual hierarchy.

Figure 1 shows the conceptual hierarchy over the domain D and its associated tree. Keeping in mind the rules stated in the previous section, the number of input neurons of all the neural networks that build the system is determined, this number is equal to 2 since this is the dimension of the input pattern domain.

Studying the associated tree to the conceptual hierarchy it can be deduced that, besides the root neural network (R_0), two more neural networks are needed in order to be able to manage the specializations of concept $D_2(R_2)$ and concept $D_3(R_3)$. Counting the number of arrows in the concepts, the number of output neurons in each one of the neural networks can be determined.

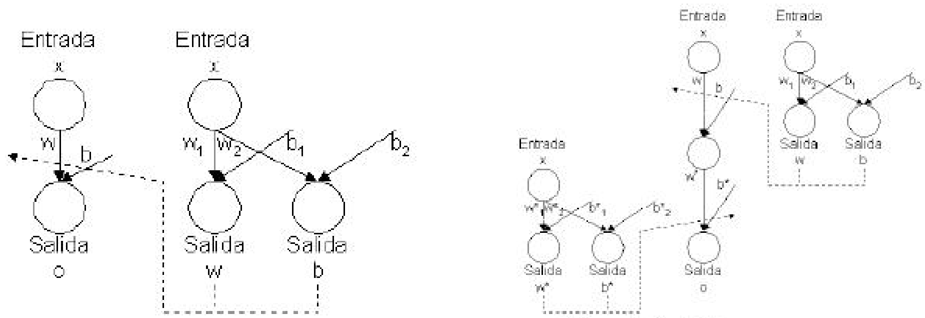
Networks (R_0) and (R_2) have three output neurons, while network (R_3) only has 2 neurons. Next step is to determine the neural network architecture, in this example a Multilayer Perceptron (MLP) has been employed.

Figure 2 shows the established links among all the different neural networks in the system. Note in the figure that an excitatory response could be assimilated from a given output neuron of a neural network with the ability of transmitting current information in the system towards any network set. In this case, it could be said that, the system has neurons whose excitation is the key for transmitting the neural activity through the system.

The activation of any output neuron belonging to any neural network is interpreted as the recognition of a given input to the concept that the neuron represents. Besides, if the concept of the hierarchy has specializations, then the system will propagate the input to some neural network in order to continue with the classification process, this is what is called propagation of the neuronal activity through the system, until the exhaustive classification of the input is reached.

5. Numeric Representation

THANKS to a neural network, see figure 3, it is possible to predict, analyzing historical series of datasets just as with these systems but there is no need to restrict the problem or use Fourier's transform. A defect common to all those methods it is to restrict the problem setting certain hypothesis that can turn out to be wrong. We just have to train the neural network with hystorical series of data given by the phenomenon we are studying[11]. Calibrating a neural network means to determinate the parameters of the connections (synapsis) through the training process. Once calibrated there is need to test the netowrk efficiency with known datasets, which has not been used in the learning process. There is a great number of Neural Networks[12] which are substantially distingushed by: type



$$
\begin{aligned}
o &= wx + b = \\
&= (w_1 x + b_1)x + w_2 x + b_2 = \\
&= w_1 x^2 + (b_1 + w_2)x + b_2
\end{aligned}
$$

$$
\begin{aligned}
o &= w^*(wx + b) + b^* = \\
&= (w_1^* x + b_1^*)[(w_1 x + b_1)x + w_2 x + b_2] + = \\
&+ w_2^* x + b_2^* = \\
&= Ax^3 + Bx^2 + Cx + D
\end{aligned}
$$

Fig. 4. ENN architectures and output expressions.

of use, learning model (supervised/non-supervised), learning algorithm, architecture, etc.

Multilayer perceptrons (MLPs) are layered feed forward networks typically trained with static backpropagation. These networks have found their way into countless applications requiring static pattern classification. Their main advantage is that they are easy to use, and that they can approximate any input-output map. In principle, backpropagation provides a way to train networks with any number of hidden units arranged in any number of layers. In fact, the network does not have to be organized in layers any pattern of connectivity that permits a partial ordering of the nodes from input to output is allowed. In other words, there must be a way to order the units such that all connections go from earlier (closer to the input) to later ones (closer to the output). This is equivalent to stating that their connection pattern must not contain any cycles. Networks that respect this constraint are called feed forward networks; their connection pattern forms a directed acyclic graph or dag.

Jordan and Elman networks extend the multilayer perceptron with context units, which are processing elements that remember past activity. Context units provide the network with the ability to extract temporal information from the data. In Elman networks, the activity of the first hidden layer are copied to the context units, while the Jordan network copies the output of the network. Jordan and Elman networks combine the past values of the context unit with the present input x to obtain the present net output. The Jordan context unit acts as a so called lowpass filter, which creates an output that is the weighted (average) value of some of its most recent past outputs.

Time lagged recurrent networks are MLPs extended with short term memory structures. Most real-world data contains information in its time structure. Yet, most neural networks are purely static classifiers. TLRNs are the state of the art in nonlinear time series prediction, system identification and temporal pattern classification.

Symbolic information can be represented in this hierarchical model so far. Also numeric information can be represented. Let's supose a function to approximate, then some neural networks models can be used. Next section introduces axo-axonic neural networks (ENN) [10] that are able to perform a similar approach to Taylor series.

TABLE I

NUMBER HIDDEN LAYERS VS. DEGREE OF OUTPUT POLYNOMIAL

Hidden Layers	Degree $P(x)$	Output Polynomial
0	2	$o = a_2x^2 + a_1x + a_0$
1	3	$o = a_3x^3 + a_2x^2 + a_1x + a_0$
...	...	...
n	$n+2$	$o = \sum_{i=0}^{n+2} a_i x^i$

5.1. *Function Approximation*

A function can be approximated with a given error using a polynomial $P(x) = \hat{f}(x)$ with a degree n. The error $f(x) - P(x)$ can be measured in such a way that in order to find a suitable approximation (error lower than a known threshold) it is only needed to compute sucessive derivates of function $f(x)$ until a certain degree n.

Enhanced Neural Networks behave as n-degree polynomial approximators depending on the number of hidden layer in the architecture. In order to obtain such behavior all activation functions of the net must be lineal function $f(x) = ax + b$.

As shown in figure 4 and output equations, the number of hidden layers can be increased in order to increase the degree of the output polynomial, that is, the number n of hidden layers control, in some sense, the degree $n+2$ of output polynomial of the net.

Table I shows how the degree of the output polynomial increases according to the number of hidden layers in the net.

The only condition that the learning algorithm must verified is that weights must be adjusted to values related with the sucesive derivates of function $f(x)$ that pattern set represents. Usually such function is unkown therefore, if the network converges with a low mean squared error then all weights of the net have converged to the derivates of function $f(x)$ (the pattern set unkown function), and such weights will gather some information about the function and its derivates that the pattern set represents.

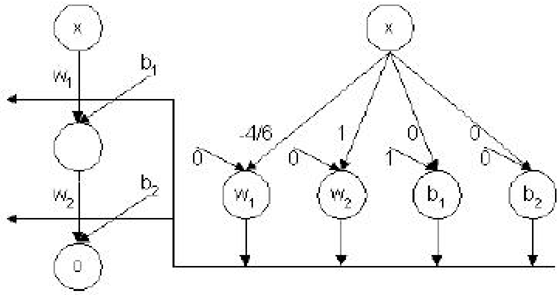


Fig. 5. Approximation of $f(x) = sen(x)cos(x)$ with a one hidden layer.

As an example, function $f(x) = sen(x)cos(x)$ can be approximated using equation obtained from *Taylor* series, with a given point $a = 0$. Such equation can be reduced to $\tilde{f}(x) = x - \frac{4}{6}x^3$, using a polynomial $P(x)$ degree 3. This is a mathematical approach, but what happens if such function is the pattern set to an enhanced neural network mentioned before?.

A one hidden layer neural network must be used in order to obtain a 3-degree polynomial as the output expression. Figure 5 shows such architecture, after the training stage, the final configuration is shown. Output equation of the net is $o = x - \frac{4}{6}x^3$, equivalent equation with $\tilde{f}(x)$.

The approximation error using net in figure 5 can be computed using equation mentioned before, and therefore $MSE \leq |e(x)|$. Such approximation is not the only one nor the best one, but it can be computed theoretically in order to provide the net some initial weights in order to speed up the learning process and to obtain a better approximation that the initial one with a lower error ratio. In sumary, Enhanced Neural Networks can be initialized to some weights computed using the Taylor Series of the function that the pattern set defines and after this initial stage the learning algorithm must be applied in order to achieved the best solution (the one that improves the Taylor Series error).

Figure 6 shows the surface computed by a net as the number of hidden layers is increased. The mean squared error is decreasing as the number of hidden layers goes up. This figure shows that this kind of neural net is very suitable when approximating functions, a given function or a function defined by the pattern set.

5.1.1) Non-Linear Activation: According to previous ideas, *linear ENNs* are better than linear *MLPs*, or at least, they are able to generate complex regions in order to divide the output space. When working with a *MLP*, only hyperplanes can be obtained. And moreover, the degree of the output equation increases according to the number of hidden layers.

In order to obtain a functional basis, one constraint must be made. It consists on implementing the network architecture with lineal *PE*s except the output neurons of assistant network. These neurons must have an activation function $g(x)$ which is used to computed the functional basis as the application of $g(x)$ to a non lineal combination of inputs. Figure 7 shows an example of a functional basis and the main network ourput.

Depending on the activation function of output neurons belonging to assistant network, the main network will output an approximation function based on non lineal combination of elements belonging to the basis. That is if a sinusoidal activation function is implemented, then a cuasi-Fourier approximation is computed by the network; is a Ridge activation function is implemented, then a cuasi-Ridge approximation is computed and so on.

Main advantage of this new approximation method is that is absolutely easy to implement. And moreover, a global approximation to all the pattern set is perform. This way, if there are enough input patterns, then the generalization error will be minimized if there are enough learning iterations.

5.2. An example

ENN obtain best performance than classical models (MLP, RBF, etc...) in order to approximate functions. But they used a global approximation schema along the whole pattern set, and in some points the mean squared error is high compared to Taylor series.

Hierarchical model explained before can be used in order to perform a global-specific approximation, that is, a general net that performs a global approximation can fire up another net that performs an approximation in a given interval. This second net can fire up another net in a narrower interval an so on (see figures 2 and 8).

6. A SYMBOLIC PRACTICAL CASE OF USE

IN order to check the System capabilities, we have use the Zoo database from the UCI Machine Learning Repository[1]. This database has the following characteristics: each record has 18 attributes, the animal name, 15 Boolean attributes corresponding to different particular values of attributes for each instance and 2 numeric attributes (one for number of legs and other for the class each animal belongs). The database has 101 instances of different classes of animal. In order to process with categorization, the database has been modified and the instances have been rearranged into a conceptual tree of categories. The figure 9 shows the conceptual tree for this case of study.

The Artificial Neural Networks chosen to represent the connectionist component for every non-terminal concept is a Multilayer Perceptron with two layers. They have been trained with the Backpropagation algorithm. Every Perceptron in the system has the same number of input neurons in order to propagate adequately the sub symbolic information in the Connectionist System. The output layer has as many output neurons as son nodes have the concept in the category tree. The activation of any output neuron belonging to any neural network in the System is interpreted as the recognition of a given input to the concept that the output neuron represents. Besides, if the concept of the hierarchy has specializations, then the system will propagate the input to the appropriate neural network in order to continue with the classification process, until the exhaustive classification of the input is reached. This process of categorization is called propagation of the neural activity through the system. Each Perceptron has been trained with the appropriate instances from each category

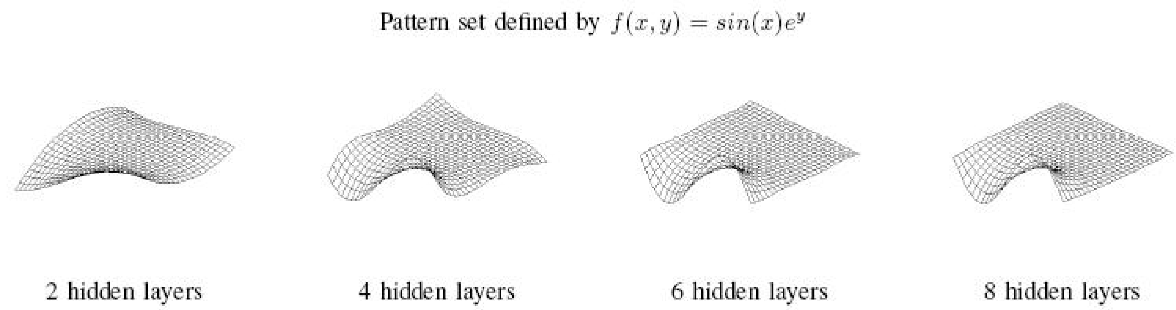


Fig. 6. Surface approximation depending on the number of hidden layers

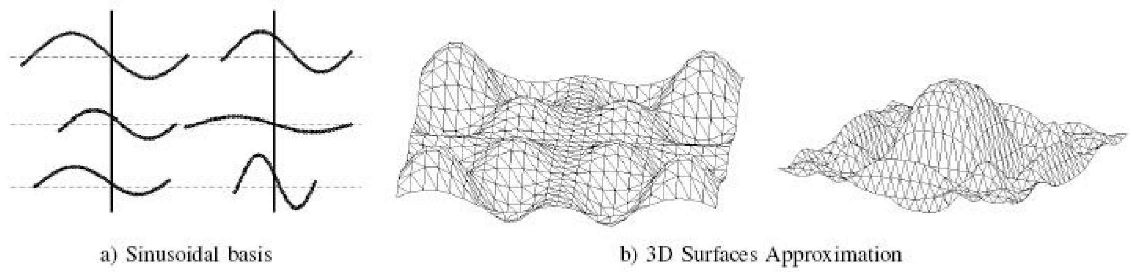


Fig. 7. Approximation with sinusoidal activation functions using basis of figure a).

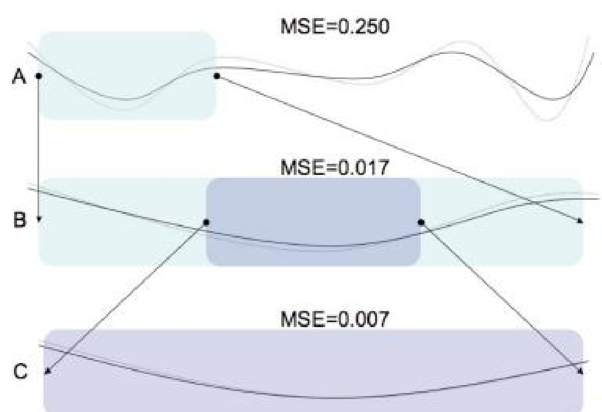


Fig. 8. Hierarchical Network Approximation. A) First net with a global approximation, B) Second net with an interval approximation and C) Last net with a narrow interval approximation.

TABLE II
SYSTEM OUTPUT

```
• Attributes:
    – Aquatic: 0
    – Fins: 0
    – Tail: 0
    – Toothed: 0
    – Domestic: 0
    – Eggs: 1
    – Milk: 0
    – #Legs: 0
    – Hair: 0
    – Feathers: 0
    – Predator: 1
    – Breathes: 0
    – Catsize: 0
    – Venomous: 0
    – Backbone: 0
    – Airborne: 0
  ANIMAL INVERTEBRATE NOINSECT NOVENOMOUS
```

independently of the other ones. In order to implement the Connectionist System, it has been needed 12 Artificial Neural Circuits, one per each category or concept with specialisations; in our case of study the Neural circuit is constituted by only one Artificial Neural Network.

The symbolic component of every concept has been designed in order to keep the maximal economy criterion. Therefore, communes attributes of categories have been placed at the higher level of the hierarchy in which they can be placed. Therefore, the Symbolic System has 27 categories or concepts, which maintain the symbolic information of the system in a distributed manner. The instances are directly associated to the categories without specializations, because they are responsible for maintaining such a kind of information in the System.

The System answers to new inputs giving an exhaustive classification for them. The exhaustive classification process presents the complete path through the conceptual tree of categories, to which the input belongs. This classification process is produced by the Connectionist System. Moreover, due to the duality existing in the System, it is possible to recover the input attributes from the symbolic representation of categories.

For example, if we present to the system information from an animal (e.g. clam) and this animal has never been presented to the system, the system produces the following response, see table II.

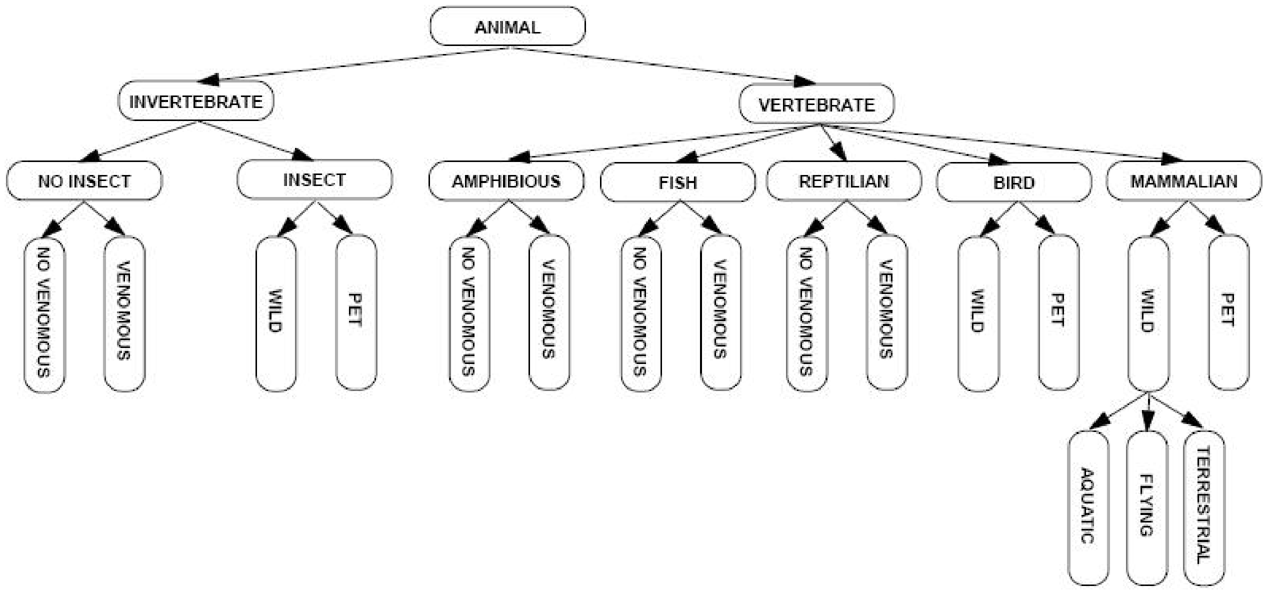


Fig. 9. Conceptual Tree for our Case of Study

TABLE III
NEURAL NETWORK PERFORMANCE

File	Samples	Net	Hidden	Perf.	MSE
1	3024	R_0	3	99.9997	0.000154
2	1532	R_1	15	99.997	0.003124
3	1275	R_2	20	99.997	0.003639
Whole System				99.041	

The new inputs presented to the System can be added to the Symbolic System be mean of this recovering process of symbolic information. Hence, if we add information from new instances to the System, it will add such information to the appropriate non-specialised category.

6.1. Learning and Performance of Neural Networks and System

Neural networks that build the system have been trained using the back propagation algorithm, which is based on the delta rule. Table III shows the results obtained training these neural networks:

Performance of the system is 99% over the input training set. The number of patterns misclassified was 29, and all of them were bound patterns, that is, were patterns maybe belonging to more than one class. Table 2 shows the results of the system with a test set.

Performance of the system is 97% over the test set. Same as previous set, the misclassified patterns correspond to those belonging to more than one class.

7. CONCLUSION

THIS paper presents a Hierarchical Connectionist - Symbolic System, which permits to insert and recover information hierarchically structured following a priory predetermined hierarchy of classification using the generalization properties from Connectionist Systems. The system not only exhaustively classifies but also recovers symbolic information from the categories of the tree, which permits to automatically insert new inputs in the appropriate non-specialised node of the hierarchy. What the System really does, is to insert new information non-previously known increasing the information about categories in the System.

This paper proposes a connectionist system, inspired by the neurobiologist point of view, formed with a set of Artificial Neural Networks. This system is able to exhaustively classify an input to the system. This property gives the system the possibility to classify inputs as belonging, not only to one category, but also to any category that is a specialization and to continue until a real concept is achieved. That is the way the system is a tool when dealing complex classification problems, due to any domain features and to any information about the elements of the domain

The performance of the whole system is as good as the performance of the individual neural networks by themselves. At least in the pattern sets that have been tested. This kind of connectionist system could help in the processing of abstract information within some global system. Given an input to the system, a neural network activity turns up and this activity is propagated through the system in a total controlled way by the system.

ACKNOWLEDGMENT

The authors would like to thank the spanish agency *Ministerio de Ciencia y Tecnología* its support via the project TIC2003-09613-c03-03 - *Artificial Neural Networks and Networks of Evolutionary Processors.*

REFERENCES

[1] Blake, C., Keogh, E. and Merz, C. J.: *UCI Repository of machine learning databases.* [http://www.ics.uci.edu/mlearn/MLRepository.html].

Irvine, CA: University of California, Department of Information and Computer Science. 1998.

[2] Collins, A. M. and Quillian, M. R.: *Retrieval time from semantic memory*. Journal of Verbal Learning and Behavior, 8,240-247. 1969.

[3] Quillian, M. R. Semantic Memory. E. Minsky (Ed.): *Semantic Information Processing*. Cambrige, Mass.: MIT Press. 1968

[4] Tulving, E. Episodic and Semantic Memory. E. Tulving and W. Donaldson (Eds.): *Organization of Memory*. New York: Academic Press. 1972

[5] Wallace, J.G.; Silverstein, R.B.; Bluff, K. and Pipingas, A.: *Semantic Transparency, Brain Monitoring and Evaluation of Hybrid Cognitive Architectures*. Connection Science, Vol. 6, N. 1, 1994.

[6] F. Arroyo, J. Castellanos, C. Luengo, L.F. Mingo: *Frames Neural Networks for Cognitive Process*. Proceedings of SCI?97 ?World Multiconference on SYSTEMICS, CYPERNETICS AND INFORMATICS. Caracas July 1997, pp. 288-295. 1997.

[7] F.Arroyo, J. Castellanos, C. Luengo, L.F. Mingo: *Connectionist Systems for a Hierarchical Knowledge Representation*. Proceedings of ICNNB98 - 1998 International Conference on Neural Networks and Brain. Beijing, pp. 471-474. 1998.

[8] F. Arroyo, C. Luengo, F. Mingo, J. Castellanos: *Evolving Connectionist and Symbolic Processes in a Hierarchical Memory Model*. Proceedings of World Multiconference on Systemics, Cybernetics and Informatics. SCI'99 and ISAS'99, Orlando, Florida, 1999, vol 8, pp. 255-261. 1999.

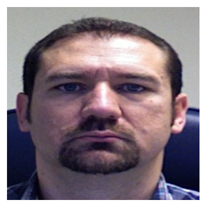
[9] Levon Aslanyan, Luis de Mingo, Juan Castellanos, Vladimir Ryazanov, F. Chelnokov, Alexander Dokukin: *On Logical Correction of Neural Network Algorithms for Pattern Recognition*. International Journal on INFORMATION THEORIES & APPLICATIONS. Vol. 14. No. 2. ISSN 1310-0513. Pp.: XX-XX. 2006.

[10] Luis Mingo, Levon Aslanyan, Juan Castellanos, Miguel Diaz, and Vladimir Riazanov. *Fourier Neural Networks: An Approach with Sinusoidal Activation Functions*. International Journal on INFORMATION THEORIES & APPLICATIONS. Vol. 11. No. 1. ISSN 1310-0513. Pp.: 52-55. 2004.

[11] Anderson, James A. and Edward Rosenfield (1988). Neurocomputing: Foundations of Research Cambridge, MA: MIT Press ISBN: 0-262-01097-6.

[12] Anderson, James A. (1995). An Introduction to Neural Networks Cambridge, MA: MIT Press ISBN: 0-262-01144-1.

Fernando Arroyo Ph.D. on Artificial Intelligence with a position of full professor at the *Universidad Politécnica de Madrid*. He has a great number of international publications in leading journals concerning Membrane Computing and Neural Networks.



Miguel Angel Díaz Full professor at the *Universidad Politécnica de Madrid*. His research interests lie in the area of membrane computing, especially in networks of evolutionary processors, but also in the area of neural networks.He has participated in several local projects. He has some papers published in visible journals and in proceedings from notable conferences (WSEAS, etc.)

Luis F. de Mingo Contracted professor at the *Universidad Politécnica de Madrid* since 1998 and Ph.D. on Artificial Intelligence. His research interests lie in the area of learning models, in particular in the theoretical parts of Neural Networks and Pattern Recognition, but also in the area of Networks of Evolutionary Processors. He has participated in several *INTAS* projects and some local projects. Member of the editorial board of IJITA, and international reviewer of KDS, ITECH conferences. Active member of European Molecular Computation Consortium.

Juan Castellanos Full professor at the *Universidad Politécnica de Madrid* with a Ph.D. on Artificial Intelligence. His research interests lie in the area of bio-computing, in particular in the theoretical parts of DNA computing and membrane systems, but also in the area of neural networks. He has more than sixty papers in these branches of bioinformatics, many of them published in visible journals and in proceedings from notable conferences (7th, 8th International Meeting on DNA Based Computers, etc.)