Bridging the gap between subsymbolic and symbolic techniques: A pragmatic approach

G. STAMOU*, D.VOGIATZIS* and S. STROVE**
* Department of Electrical and Computer Engineering
National Technical University of Athens
Heroon Polytexneiou 9, Athens, 157 73
GREECE
gstam@softlab.ntua.gr, dimitrv@central.ntua.gr,

** Department of Medical Physics and Biophysics
University of Nijmegen
Geert Grooteplein 21, 6525 EZ Nijmegen,
THE NETHERLANDS
sybert@mbfys.kun.nl

Abstract: We present a survey of methods for passing from subsymbols to symbols. First, we present a definition of the following notions: Subsymbols, Symbols and Hybrid Systems, as well as a taxonomy of Intelligent Systems. Then, pure Artificial Intelligent (AI) as well as pure Computational Intelligence (CI) techniques for subsymbolic to symbolic mapping are exposed. Next, we present a taxonomy of hybrid systems; according to that there are three categories of hybrid systems: Combined Intelligent Systems, Transformational Intelligent Systems and Coupled Intelligent Systems. Hybrid systems are a very promising alternative to pure AI and pure CI techniques for mapping subsymbols to symbols. Finally, we elaborate on the Hybrid Intelligent Systems as well as on the approach, which we follow in Physta research project, which is a subcategory of Hybrid Systems.

Key-Words: Neurosymbolic Integration, Neural Networks, Symbols

1 A Pragmatic Framework

A framework for the terminology and categorization of intelligent systems as tools for modeling the brain or as machines that are characterized by an intelligent behaviour has been proposed by Bezdec [7]. The main distinction in Bezdec's categorization is made on the basis of the abstraction level of the processing involved in each intelligent system: *symbolic* or *subsymbolic* (numerical). Historically, these two main approaches are represented by *Artificial Intelligence* (AI) and *Neural Networks* (NN), respectively. The NN approach together with other subsymbolic techniques (fuzzy logic, genetic algorithms etc) has lately been referred to as *Computational Intelligence* (CI).

Knowledge representation in artificial intelligence systems involves the use of symbols. We will adopt the following definition of symbols: a symbol represents abstract notion as well as relevant properties of that abstract notion. Symbolic processes transform symbol structures into other symbol structures that are relevant in a specific context. Although heuristic and linguistic knowledge used by experts can be the basis on which AI systems are constructed, the symbol grounding problem, i.e. the problem of how the symbols acquire meaning, remains one of the main problems of AI [14]. Roughly speaking, this is the top-down view of the problem of subsymbolic to symbolic mapping. Several methodologies (as fuzzy set theory, probabilistic reasoning etc.) have been used in order to support the conventional AI systems in order to bridge the gap between symbols and subsymbols. For example, fuzzy expert systems or decision trees are using numerical data in order to determine the symbolic knowledge.

On the other hand, in computational intelligent systems, knowledge is represented in a more opaque, subconceptional manner. The subsymbolic representation is usually defined by the origin of the information (e.g. sensors or database), rather than by the content of the information. Moreover, in a CI system with multiple streams of information, knowledge is

usually stored in a distributed fashion. A main objective of mapping subsymbolic information into a symbolic representation is to find an abstract representation of the symbol or object, which is invariant with respect to various features (e.g. invariant with respect to position and orientation). Moreover, the representation should be such that direct links are possible to properties of the object or symbol and such that the representation can easily be used in symbolic reasoning. This problem can be viewed as the bottom-up case of the subsymbolic to symbolic mapping problem. The main approach of computational intelligence is the connectionist approach. Biologically inspired neural networks that try to model the human brain and artificial neural network models that try to emulate the intelligent behaviour of humans are the basic methods of computational intelligence. Lately, supporting methodologies like fuzzy set theory have been used in order to improve the symbolic representation and processing capabilities of NNs.

Although each of the subsymbolic and symbolic framework can specify intelligent systems by itself, the limits of each category are not clear and become increasingly vague. The emergent need for intelligent systems that can work in both the symbolic and the subsymbolic framework leads to the unification of AI and CI approaches. Hence, the perspective that the basic method of CI is the NN approach and that the basic method of AI is the traditional symbolic processing (expert systems, rule-based systems) is mainly due to historical reasons. The main issue in the research on intelligent systems is now how the symbolic description of mental processes, in terms of rules and representations in the province of conventional AI, can be related to a subsymbolic description in terms of brain mechanisms (the top-down approach). The same question is how the subsymbolic processes of neurons, synapses and interconnections of a conventional artificial neural network can be related to the symbolic description of human logic and behaviour (the bottom-up approach). The perspective of PHYSTA is based on the bottom-up approach. One of the principal reasons that neural networks have been considered a useful vehicle for such a development is that there is an existence evidence for the solution of creating such a system (and of course answering the above questions), that of the human being.
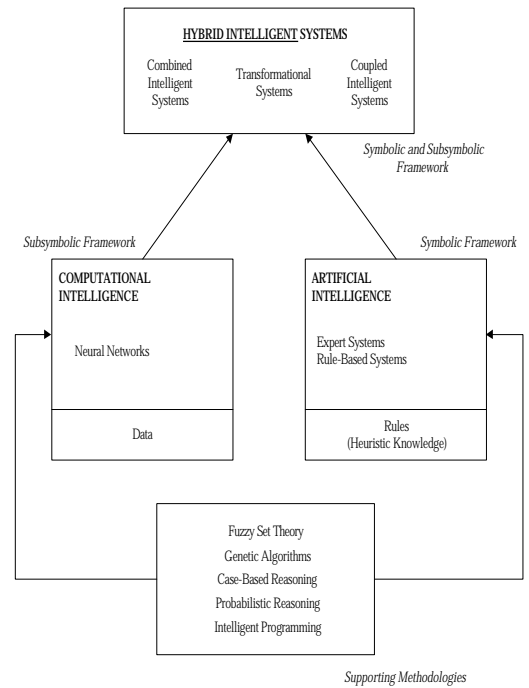

Figure Error! Unknown switch argument. **Intelligent Systems taxonomy**

The need for combination of AI and CI methods arises from the need for both symbolic and subsymbolic processing. This combination leads to hybrid systems. Hybridization is done not only in the sense of integrating independent AI and CI methods in order to construct an overall system, but also in the sense of the theoretical unification of AI and CI ideas (Figure 1). In the following, we will adopt a taxonomy of hybrid intelligent systems, as shown in Figure 2. This taxonomy is done on the basis of previous work by (see [16],[25]) The basic innovation with respect to the latter lies in the use of a different terminology and in the distinction of hybrid intelligent systems into three categories. The first category, Combined Intelligent Systems, comprises systems that use NNs as tools for symbolic processing (top-down approach) and NNs that use AI concepts in order to support symbolic processing (bottom-up approach). AI and CI methodologies are combined to construct a combined intelligent system. The second category, Transformational Intelligent Systems, use CI and AI techniques to transform symbolic representations to subsymbolic and vice versa. The main operations provided by Transformational Intelligent Systems are rule insertion, rule extraction and rule refinement. The most important point is that the CI and AI methods used for this operation are stand-alone and based on conventional approaches, in the corresponding

framework. The third category, Coupled Intelligent Systems, integrate AI and CI modules to produce a system with the ability to work at both symbolic and subsymbolic level. The AI and CI components are independent and communicate in a predefined way that characterize the kind of processing involved. Four types of coupling can be distinguished, namely chainprocessing, subprocessing, metaprocessing and coprocessing (see [16]).
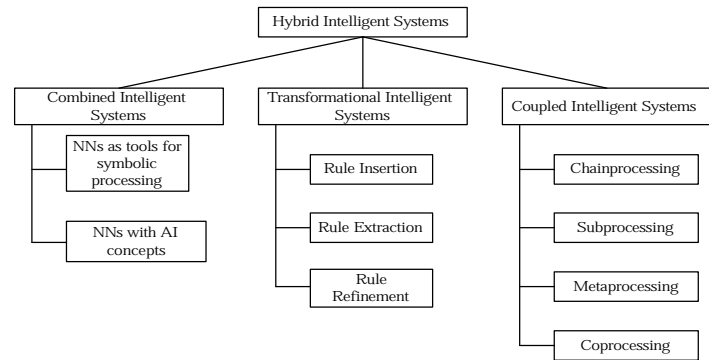

Figure 2. Hybrid Intelligent Systems

2 Review of conventional AI techniques for mapping features to symbols

In this section, some methods for solving the complete reduction problem of AI systems are reviewed. In expert systems, knowledge is represented in the form of rules that are used to carry out tasks usually performed by human experts [10]. The basis of such rules is the theory of propositional logic which uses propositional variables (true/false) and truth-functional propositional connectives, including conjunction, disjunction, negation, implication and logical equivalence. If axioms and rules of inference are provided, a sequence of inferential rules results in a proof.

However, deterministic reasoning cannot solve many real life problems; thus probabilistic reasoning is required. Reasoning under uncertainty may be achieved by using a probabilistic network (see [19]). Another interesting generalization of AI systems that is very effective with noisy and uncertain data is that of fuzzy knowledge-based systems. Fuzzy set theory and fuzzy logic are mathematical theories widely used in artificial intelligence [21]. Fuzzy rule-based systems and fuzzy expert systems are AI systems with the ability of mapping subsymbolic to symbolic knowledge with the aid of fuzzy mathematics [40].

There are also many methods for rule induction. One of the earliest is the decision tree, which is a directed acyclic graph comprising internal and terminal nodes. All internal nodes contain splits, which test the value of a mathematical expression. Each terminal node of a decision tree has a class label associated with it. The nodes are connected by edges. The construction algorithm tries to find a rule at each node which splits the data according to some criterion giving rise to a decision tree with an exponential number of nodes at each level. The criterion is usually such that entropy or a related quantity is minimized at each node. The result is a decision tree, which gives a good performance (usually with regard to classification) and which provides insight into the performance, since the rules at the decision nodes are known. These rules usually are defined in terms of if-then rules.

Learning and adaptation are important characteristics of intelligent systems that play a central role in subsymbolic to symbolic mapping via classification. EITHER [28] and Labyrinth-k [36] are two symbolic learning algorithms. The input consists of a (probably incorrect) domain theory and a set of data (i.e. empirical knowledge). The output is a refined domain theory. Another interesting method, lately used to enrich AI systems with the ability of self-construction is the Genetic Programming (GP). GP is actually a method of automatic programme induction and bears a resemblance to the theory of natural selection [22]. The word "programme" should be considered here in the broadest possible sense, to include computer programs but also any symbolic structure

3 Review of conventional NN techniques for mapping features to symbols

Artificial neural networks are universal approximators and operate in a subsymbolic framework. They have been widely used for classification problems. Classification can be viewed as a first way of relating subsymbolic to symbolic information. Ideas of mapping features to symbols can be found in the basic structure of typical neural networks (such as multi-layer perceptrons, self-organizing maps etc). However, this mapping takes place to a limited extent, since

traditional neural networks do not support methods for symbolic processing. With the aid of clustering and classification some general information is mined concerning the grouping of numerical data. On the other hand, methods for solving the problem of mapping features to symbols have to prove useful in terms of symbol understanding and processing. For this aim, the problem of mapping features to symbols with the aid of NNs is connected with the problem of rule extraction from already trained NNs.

The topic of rule extraction in neural network research has raised a recent increase of interest driven by the need to develop artificial systems with near-human capabilities (although this is still remote) and analyse the manner in which humans achieve rule extraction; also by the need to understand deficits caused by brain injury as well as help build artificial systems. A review of the whole area of these various aspects of rule formation is given in [6].

Neural networks have been widely used for classification problems; classification could be viewed as a subsymbolic to symbolic mapping. Feedforward networks, like multi-layer perceptrons (MLPs) [15], are appropriate for various classifications tasks, although it is not clear that they can be useful for subsymbolic to symbolic mapping purposes. A more useful approach is that of tree neural networks (TNNs) ([30, 13]). The main idea of these networks is the combination of decision trees with feedforward neural networks using a small MLP for each decision node of the decision tree. An interesting extension of decision trees is described in [5]. Another idea following the same rationale, is the *hierarchical mixture of experts* (HME) architecture [18] which is a tree-structure with gating networks on the links and expert networks on the leaves.

Another useful approach are the probabilistic neural networks (PNN) [33, 8]. The network structures are similar to those of multilayer perceptrons; the primary difference is that the sigmoid activation function is replaced by an exponential one. Key advantages of PNNs are that training requires only a single pass and that decision surfaces approach the Bayes-optimal decision boundaries as the number of training samples grows. An interesting generalization of probabilistic networks is proposed by Kuo et al. in [23] using concepts from fuzzy set theory. The pRAM model is a stochastic neural network, generating strings of random bits, whose joint probability distribution is modifiable through learning algorithms, in terms of conditional probabilities of any bit given the values of certain others [4].

Using fuzzy set theory and fuzzy logic principles the network becomes more flexible, more robust and can be described at a higher symbolic level. Fuzziness can be introduced either in neuron models or in the training algorithm.

4 Hybrid AI and NN techniques for mapping features to symbols

Next we explore the *Combined Intelligent Systems* which is the first sub-category of the hybrid systems. The earliest examples in this category are the Neural expert systems which enrich neural networks with heuristics. Those heuristics allow neural networks to cope with incomplete information, to explain conclusions and to generate questions for unknown inputs (see [12, 32]).

The *KBANN* [38] integrates propositional calculus with neural learning. A set of rules designating a domain theory that is partially correct is integrated into a feed-forward neural network. The neural network is trained, thus refining the rules. Furthermore, it is possible to integrate a deterministic finite state automaton in a recurrent neural network (see [27]). Rule extraction for feed-forward neural nets can be achieved with the *KT* [11], which produces propositional logic rules. A novel method for the classification of structures by neural networks has been proposed by Sperduti and Starita [35]. The authors propose a generalization of a recurrent neuron, whereby it is facile to map graphs to a neural network. Another idea is the LRAAM model [32, 35], which can be used in conjunction with a classifier network to classify labeled structures.

The use of NNs in order to realize the key concepts of a fuzzy logic system enriches the system with the ability of learning and improves the subsymbolic to symbolic mapping. Neural network realization of basic operations of fuzzy logic, such as fuzzy complement, fuzzy intersection and fuzzy union, has been proposed [29,17]. A connectionist approach of fuzzy inference has been proposed by Keller et al. [20]. The issue of identifying the fuzzy rules and tuning the membership functions of fuzzy sets using neural networks and training algorithms has been widely studied. Another approach to this problem related to fuzzy control has been proposed by Lin and Lee [24] (chapter 19), who introduced the fuzzy

adaptive learning control network (FALCON) to study hybrid structure-parameter learning strategies. Structure learning algorithms are used to find appropriate fuzzy rules and parameter learning algorithms are used to fine tune the membership functions and other parameters of the fuzzy inference system.

Several methods are described next which aim at extracting, inserting or refining propositional rules in trained neural networks, all these methods fall in the transformational intelligent systems category. Some of these methods are *decompositional*, in the sense that the focus is on extracting rules at the level of individual (hidden and output) units [3]. Other rule extraction methods are usually classified as *pedagogical*, in the sense that the network is treated like a black box, whereby rules are extracted.

RULEX [1,2] is a technique of rule extraction from a trained CEBP (Constrained Error Back-Propagation Network). The derived rules belong to the class of propositional logic. The algorithm for rule production is quite efficient since it is based on a direct interpretation of the parameters of the CEBP network. Furthermore, it is possible to refine a set of rules by inserting them into the CEBP network, training the network and then extracting them. There are also methods for extraction of propositional rules from a trained multilayered perceptron when considered as a black box (see [37]) or at the level of individual neurons (see [31]). In a parallel line of research a finite state automaton has been extracted from a trained networks (see [27]).

Narazaki in [26] deals with the problem of rule extraction in a novel way: he bases his method on the function approximated by the network, rather than considering connection weights. Finally, a method called TREPAN, whereby propositional rules are extracted from a neural network with the aid of a decision tree, has been proposed in [9].

6 Conclusion - Perspective

The above described rich taxonomy and characterization of algorithms and representations is the basis for the development of a theory and methodology in the PHYSTA Project. Moreover, PHYSTA will investigate a different perspective. According to this, the rule extractor is not an external agent but a rule located in the mentioned hierarchy, i.e. a piece of neural network by itself. Thus, rather than an external subroutine, we are looking for an inner loop priming symbols from synapses.

References

[1] Andrews R. and Geva S., Rule extraction from a constrained error back propagation MLP. *Proc. 5th Australian Conference on Neural Networks*, Brisbane, Queensland, 1994, pp. 9-12.

[2] Andrews R. and Geva S., Inserting and extracting knowledge from constrained error back propagation networks. *Proc. 6th Australian Conference on Neural Networks*, Sydney, 1995

[3] Andrews R., Diederich J. and Tickle A.B., Survey and critique of techniques for extracting rules from trained artificial neural networks. *Knowledge-based Systems*, 8 (6), 1995,pp.373-389

[4] Apolloni B., de Falco D. and Taylor J.G., pRAM lay-out optimization. *Neural Networks*, 10, 1997, pp1709-1716

[5] Apolloni B., Zamponi G. and Zanaboni A.M., Learning fuzzy decision trees, *Neural Networks*, to appear. 1998.

[6] Bapi R.S., Bugmann G., Levine D.S. and Taylor J.G., Analysing the Executive Function of the Prefrontal System: Towards a Network Theory. *Behaviour & Brain Sciences* (submitted), 1998

[7] Bezdec J.C., *What is Computational Intelligent.* In Computational Intelligence Imitating Life, Zurada, Marks and Robinson (eds), IEEE Press, New York, 1994.

[8] Burrascano P., Learning vector quantization for the probabilistic Neural Network. *IEEE Trans. Neural Networks*, 2(4): 1991, pp 458-461.

[9] Craven M. and Shavlik J., Extracting tree-structured representations of trained networks. In Touretzky, D., Mozer, M., & Hasselmo, M., editors, *Advances in Neural Infromation Processing Systems* (volume 8). MIT Press, Cambridge, M.A., 1996

[10] Dean T., Allen J. and Aloimonos J., *Artificial intelligence: theory and practice.* Benjamin/Cummings Publishing Company, Redwood City (CA), USA, 1995

[11] Fu L., Rule Generation from Neural Networks. *IEEE Transaction on Systems, Man and Cybernetics*, Vol. 24, No. 8, 1994,pp. 1114-1124.

[12] Gallant S., Connectionist Expert Systems. *Communications of the ACM*, Vol. 31, No 2, 1988

[13] Guo H. and Gelfand S.B. Classification trees with neural network feature extraction. *IEEE Trans. Neural Networks*, 3(6): 1992, pp. 923-933.

[14] Haugeland J. , The nature of plausibility of cognitivism. *Behaviour and Brain Science* 2, 1978, pp. 215-260.

[15] Haykin S. , *Neural Networks - A Comprehensive Foundation*, MacMillan, 1994

[16] Hilario M., An overview of strategies for neurosymbolic integration. *In Connectionist-Symbolic Processing*: From Unified to Hybrid Approaches, R. Sun and F. Alexandre ed., Lawrence Erlbaum.,1997

[17] Hsu L.S., Teh H.H., Wang P.Z., Chan S.C. and Loe K.F., Fuzzy neural-logic systems. Proc. *Int. Joint Conf. Neural Networks*, Vol I, 1992, pp. 245-250.

[18] Jordan M.I. and Jacobs R.A., Hierarchical mixtures of experts and the EM algorithm. *Neural Computation* Vol.6: 1994, pp. 181-214.

[19] Jordan M.I., Ghahramani Z., Jaakkola T.S. and Saul L.K. , An introduction to variational methods for graphical models. In: Jordan, MI (ed.), *Learning in graphical models*, Kluwer Academic Publishers, 1998

[20] Keller J.M., Yager R. R. and Tahani H., Neural Network implementation of fuzzy logic. *Fuzzy Sets and Syst*. 45: 1992, pp 1-12.

[21] Klir J. and Bo Yuan , *Fuzzy Sets and Fuzzy Logic: Theory and Applications*, Prentice Hall, 1995

[22] Koza J., *Genetic Programming : On the Programming of Computers by Means of Natural Selection* (Complex Adaptive Systems), MIT press, 1992

[23] Kuo Y.H., Kao C. I. and Chen J. J., A Fuzzy Neural Network model and its hardware implementation. *IEEE Trans. Fuzzy Syst*., Vol.1, No.3, 1993, pp. 171-183

[24] Lin C.T. and Lee C.S. , *Neural fuzzy systems-A Neuro-Fuzzy Synergism to Intelligent Systems, Prentice-Hall*, New York., 1995

[25] Medsker L.R., *Hybrid Intelligent Systems*, Kluwer Academic Publishers, 1995

[26] Narazaki H., Watanabe T. and Yamamoto M., Reorganizing Knowledge in Neural Networks in Data Classification Problems. *IEEE Transactions on Systems Man and Cybernetics-Part B: Cybernetics*, Vol. 26, No. 1, 1996.

[27] Omlin C.W. and Lee Giles C., Constructing Deterministic Finite-State Automata in Recurrent Neural Networks. *Journal of ACM*, Vol. 43, No. 6, 1996 ,pp.937-972

[28] Ourston D. and Mooney R.J., Theory refinement combining analytical and empirical methods. *Artificial Intelligence* 66, 1994, pp. 273-310.

[29] Pao Y.H., *Adaptive Pattern Recognition and Neural networks*, Reading, Addison-Wesley, 1989

[30] Sankar A. and Mammone R. J., Neural tree networks. In R. J. Mammone and Y. Y. Zeevi, eds., *Neural Networks*, Vol. 3, No. 6, pp. 465-467, 1991.

[31] Setiono R. and Liu H., Understanding Neural Networks via Rule Extraction. *Procceedings of the 14th International Joint Conference on Artificial Intelligence*, 1995, pp. 480-485.

[32] Sima J., Neural Expert Systems. *Neural Networks* Vol. 8, No.2, 1995, pp. 261-271.

[33] Specht D.F. , Probabilistic Neural Networks. *Neural Networks*, Vol. 3, No. 1: 1990, pp. 281-302.

[34] Sperduti A., Starita A. and Goller C., Learning distributed representations for the classification of terms. *In Proceedings of the International Joint Conference of Artificial Intelligence*, 1995, pp. 509-515.

[35] Sperduti A. and Starita A., Supervised Neural Networks for the Classification of Structures. *IEEE Transactions on Neural Networks*, Vol. 8, No.3, 1997.

[36] Thomson K., Langley P. and Iba W. , Using background knowledge in concept formation. *In Proc. Eighth International Machine Learning Workshop*, Evanston, IL., 1991

[37] Thrun S., Extracting Rules from Artificial Neural Networks with Distributed representations. *In Advances in Neural Information Systems* 7, 1995

[38] Towell G.G., Shavlic J.W. and Noordewier M.O., Refinement of approximately correct domain theories by knowledge-based neural networks. *Proceedings of the Eighth National Conference on Artificial Intelligence*, San Jose, CA, AAAI Press, 1990, pp. 177 - 182.

[39] Yager R.R., Expert systems using fuzzy logic. In R.R. Yager and L. A. Zadeh eds., *An introduction to fuzzy logic applications in intelligent systems*, Kluwer Academic Publishers, 1992.