A&A manuscript no.
(will be inserted by hand later)

Your thesaurus codes are:
13(03.12.2 - 03.13.4 - 08.06.3)

ASTRONOMY
AND
ASTROPHYSICS
17.10.1996

HNS - a hybrid neural system and its use for the classification of stars

M. Klusch[1] **and R. Napiwotzki**[2]

[1] Institut für Informatik und Praktische Mathematik der Universität Kiel, Olshausenstr. 40, W-2300 Kiel, Germany
[2] Institut für Theoretische Physik und Sternwarte der Universität Kiel, Olshausenstr. 40, W-2300 Kiel, Germany

received October 8, 1992; accepted May 23, 1993
printed in: Vol. 276, pp. 309-319

Abstract. A novel hybrid neural approach for a fully automated spectral and luminosity classification of stars is presented. The hybrid neural system (HNS) integrates a neural classifier and a semantic network used for similarity based reasoning and conceptual knowledge representation, respectively. For neural classification the HNS currently uses data based on the Strömgren $uvby\beta$ photometric system. The semantic net is designed in accordance to the MK classification of stars. In this paper, the structure, functionality and application of the hybrid system are presented. The demonstrated functional capabilities, performance and results of stellar classification of the HNS show significant improvements compared to conventional astronomical techniques. After knowledge aquisition is once completed, the system classifies stellar objects very fast, reliable and without any need for pre-classification of them. In particular the HNS is also able to compare classes of stars without forcing the user to give any raw input data and special knowledge about relations between classes of stars. Moreover, this new hybrid approach offers a variety of applications in other areas.

Key words: Methods: data analysis — Methods: numerical — Stars: fundamental parameters of

1. Introduction

Automated procedures are the only practical way of classifying the enormous amount of digitized data produced by machine scans of objective-prism spectra. LaSala (1988) and Kurtz & LaSala (1991) reported on an ongoing project for an automated classification of stars using a pattern recognition technique. But since every star must be compared to many standard stars the application of this procedure is heavily time consuming. Another basic attempt was presented by Flynn & Morrison (1990). They developed a semi-automated procedure using quasi-photometric indices derived from objective-prism spectra, but their method is restricted to the small and specific range of red giant stars. Smriglio et al. (1986) developed an automated two-dimensional spectral classification from multicolour photometry based on a statistical procedure. These classification methods are fundamentally different from the hybrid neural approach presented in this paper. Moreover moderately better results were obtained by the latter. To our knowledge there exists no other fully automated method applicable to the spectral and luminosity classification of stars without any restriction or using special a priori knowledge.

Furthermore recent works on galaxy classification already explored a variety of automated procedures. In spite of several attempts to classify galaxies by deterministic algorithms, morphological classification "remains a process dependent on the eyes of a handful of dedicated individuals" (Storrie-Lombardi et al. 1992). In order to resolve this problem, Storrie-Lombardi et al. investigated an automatic morphological classification of galaxies by the use of a backpropagation neural network. In comparison to the method for automated galaxy classification presented in Laubertz & Valentijn (1989) they obtained a better and more reliable classification. However, only 64% of the tested galaxies were correctly classified. Burda & Feitzinger (1992) attempted to classify galaxies using pattern recognition methods, but their results still heavily depend on the details of the input image. Odewahn et al. (1992) employed a backpropagation neural network to perform the image classification only for a simple automated star-galaxy separation.

In this article the problem of a fully automated and highly reliable spectral and luminosity classification of stars is solved by a new hybrid neural system (HNS) introduced and established by Klusch (1991). Numerous investigations on neuronal networks have shown that these adaptive systems based on the paradigm of neural processing are better at solving inherent problems of many practical applications than others, which mainly rely on symbolic computing. This holds especially for applications where an exclusively symbolic representation of conceptual knowledge and rule-based inference under centralized control as used in conventional expert systems is insufficient or even impossible. One obvious and natural solution for this dualism in representation and processing of knowledge is an integration of both types of systems. Thus the idea of the hybrid approach is to take the advantages of each of them. Recent works on hybrid neural systems such as Hendler (1989), Tarr et al. (1989), Rabelo et al. (1990), Rich (1990), Hruska et al. (1991), Hudli et al. (1991), Bergerson & Wunsch (1991) and Lee et al. (1991) confirm that this approach is very promising for numerous applications. All hybrid neural sytems are instances of a hybrid synthesis of neural classification and symbolic knowledge representation. Thus, the purpose of this work is twofold: presenting the architecture and functionality of the new hybrid neural system HNS and on the other hand providing the HNS as a tool for a reliable, fully automated classification of stars, thereby demonstrating its applicability to this classification task.

The architecture of the presented hybrid neural system HNS integrates a backpropagation neural network and a semantic net. Thus it is designed for arbitrary similarity based reasoning, especially comparing named classes of objects without the necessity of specific numerical input by the user. For the purpose of spectral and luminosity classification of stars the neural net of the HNS currently uses photometric data. Additionally, the semantic net provides a symbolic representation of knowledge about the properties of stars due to their spectral and luminosity type. It is also partially responsible for executing the comparison of named classes of stars with respect to some similarity learned by the neural net. Providing the direct comparison of classes without any given numerical input is a new issue of hybrid neural systems and offers also new facilities for astronomical considerations. Another main advantage of the HNS is that a fully automated classification of stars in a very fast, more reliable and inherently fault-tolerant way is achieved. While working with the HNS there is no need for any pre-classification of stars or prior knowledge about relations between classes of stars. Indeed, solely for knowledge aquisition the neural net has once to be trained supervised.

Instead of the backpropagation neural net of the HNS one can also imagine the use of unsupervised neural nets such as the Carpenter/Grossberg classifier (Lippmann 1987) or Kohonen feature maps (Kohonen 1988a,b) for classification tasks. Moreover, methods for image or pattern classification which do not rely on neural computing can be broadly grouped into two major approaches. The first is the supervised classification, which is based on some discrimination function using several image features. This requires a priori information about the image classes to be recognized. A widely employed discrimination function is that of linear maximum likelihood because of its easy implementation. The second approach is the unsupervised clustering of image data which requires no a priori information. Cluster analysis is viewed as a process of partitioning an image into groups such that patterns belonging to the same group are more similar to each other than are patterns belonging to different groups. Many clustering algorithms are available and widely used in practice to classify multispectral image data (He & Wang 1992, Venkateswarlu & Raju 1992). Due to the fact that several significant features which are used to classify the original image can be pre-selected in the training stage with a priori knowledge of the classes to be recognized, the supervised approach is more frequently quoted than the unsupervised approach. There are statistical methods of tackling the problem of pattern classification which may be superior to neural nets in particular cases, and there are hybrids such as the probabilistic neural nets of Specht (1990). But a further discussion of these methods is not the purpose of this paper. For an introductory survey on classical methods for pattern classification see Duda & Hart (1973) and Ullman (1973).

We will show that the presented hybrid system improves and extends the functionality of the existing systems for a (semi-)automated spectral and luminosity classification of stellar objects within astronomy. The HNS obviously includes the use for classification of galaxies mentioned above.

The remainder of this paper is organized as follows. In section 2 we present the hybrid architecture of the HNS and its functionality. The main results of using the HNS for the spectral and luminosity classification of stars and comparing named classes of stars are summarized in section 3. Section 4 presents a short discussion of the hybrid system with respect to other astronomical classification methods and an outlook on future work.

2. The hybrid neural system HNS

The functional capabilities of the HNS are determined by the hybrid synthesis of neural classification using adaptive learning and a semantic network used for symbolic knowledge representation. Therefore this system integrates two components of artificial intelligence which rely on neural processing and symbolic computing, respectively. A short description of the global structure and behavior of the system is presented in Section 2.1, the used architectures of the neural net and semantic net of the HNS are described in 2.2 and the essential algorithms for the integration and communication between both nets are denoted in 2.3. For details see Klusch (1991).

2.1. Functionality

The structure of the HNS is given by a neural classifier and a semantic network. Conceptual knowledge about the Morgan-Keenan (MK) classification of stars is symbolically represented in an appropiate class hierarchy of the semantic net. Knowledge about similarity of stellar objects in respect to their photometric indices is distributed over weighted links of the neural net which were learned supervised during training by samples.

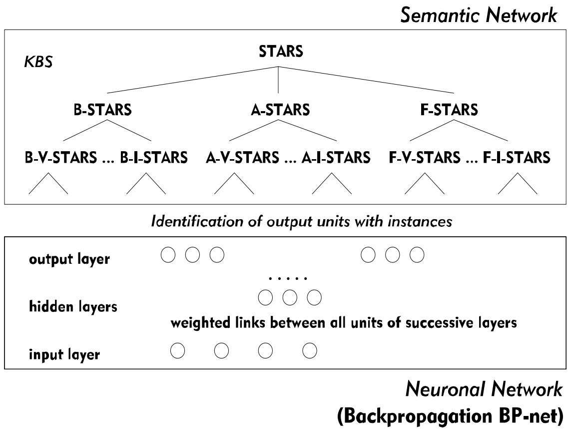

Fig. 1. Global structure of HNS

The behavior and working method of the system can be considered with respect to knowledge acquisition and similarity based reasoning. An intrinsic property of the hybrid system is the independent generation and initialization of the semantic net and neural net by the user. A compatibility check for both components is part of the internal integration of the HNS. This integration is hidden from the user interface as it is for any internal communication between neural and semantic net. The knowledge acquisition of the HNS is described as follows. The required inputs for representing conceptual knowledge about stars are

- a configuration file which contains all information about the semantic net such as number of star classes, number of stars of a class,
- a correlation table including all magnitudes of stars with respect to their luminosity and spectral class,
- some reference tables containing additional descriptive information such as spectral characteristics or any other criterions.

Initially an object-oriented class hierarchy with simple inheritance (is-a hierarchy) will be generated and initialized by reading respective files including tables mentioned above. All files can be manipulated externally. For this purpose the user has to observe the file formats. The constructed static class hierarchy of the semantic net is given by the known MK classification which obviously induces the same is-a hierarchy of stars with respect to their spectral and luminosity class. Independent from a desired initialization of the generated semantic net, leading to the actual state of the knowledge base of stars (KBS), the neural net can be constructed and trained over a set of sample patterns by the user. The set of training patterns includes the associations between measured photometric indices and respective MK class. Any unknown classification can be mapped to a special class CL_Unknown. Based on the user-defined layered feed-forward neural net all patterns will be learned using the known backpropagation algorithm. The system accepts all assignments of associations to be learned, the topology and parameters of the neural net determined by user. The neural architecture is intitialized according to that and a compatibility check finds out if an identification with an additionally initialized semantic net is feasible. Before working with both parts of the system it is possible to test the created neural net extensively using a test-tool which is part of the implemented HNS. Using the system there exist two classification methods: bottom up (BU) and top return (TR). Both modes depend on the given user input as shown in Figs. 2.1 and 2.1.

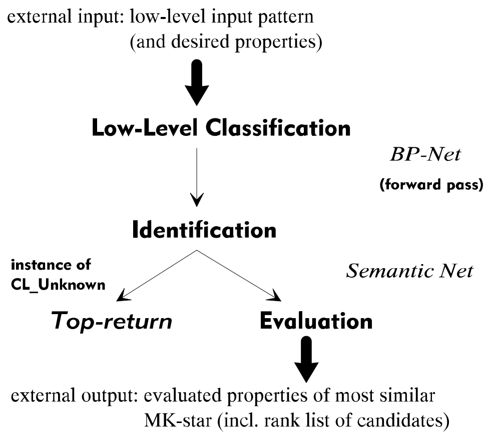

Fig. 2. Bottom up classification

Low-level classification determines the most similar class associated with the given special numerical input. An immediately following *identification* finds out the respective class in the actual overlayed semantic net. The *evaluation* corresponds to an access to the symbolically coded information in the knowledge base of stars which belong to the classified class such as property values. *Spreading-activation* gets the most similar class to a named one using all weights of the last neural layer. Comprehensive examples for both classification methods of the HNS are given in Table 1 and 2.

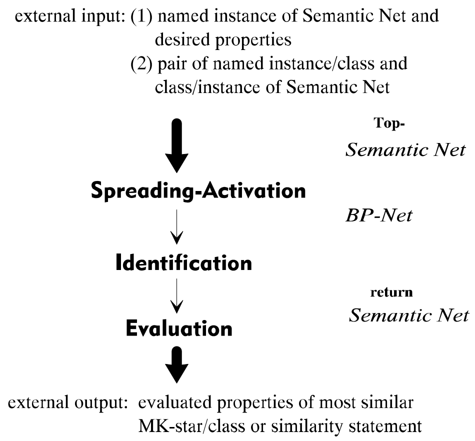

Fig. 3. Top return classification

Table 1. Bottom up (BU) classification

user input:	$[m_1] = 0.102$
	$[c_1] = 0.001$
	$\beta = 2.557$
HNS output:	<rank list of most similar MK classes>
	classified MK class is B0I
user input:	temperature :a mass :p
HNS output:	Evaluate:
	temperature = unknown
	mass = 50 25 (B0I B5I)

:a means actual property value of identified object
:p permitted property values of all respective MK classes

Table 2. Top return (TR) classification

user input:	B0-I radius :a Mvis :a
HNS output:	<rank list of most similar MK classes>
	Similarity with conceptual class B0III detected.
	Evaluate:
	radius = 16
	Mvis = -5
user input:	B0-I A0-III
HNS output:	<rank list of most similar MK classes>
	No similarity with MK class A0III detected.
user input:	BI-STARS BIII-STARS
HNS output:	<rank list of most similar MK classes>
	Similarity with conceptual class BIII-STARS.
user input:	B0-I A-STARS
HNS output:	<rank list of most similar MK classes>
	No similarity with conceptual class A-STARS.

2.2. Architecture

2.2.1. Neural network

The knowledge represented in neural networks is determined by the weights of the connections between the neural units. These units are collected in several layers, leading to a multi-layered neural architecture. Enforcing the weights or weakening them leads to a connection structure, representing the data trained. The unsolved problem of how to develop an optimal neural net for a proposed set of training data is one of the issues which in practice leads to a delay before the system finally works for the desired application. This causes one disadvantage of all hybrid neural systems. In return, once trained, the neural classifier is able to generalize from separated classes of training patterns very fast and in an inherently fault-tolerant matter. Especially, no prior knowledge about any special features of patterns or pre-classification is necessary. Indeed, solely for knowledge aquisition the neural net has to be trained supervised by user-defined class assignments to selected input patterns.

As demonstrated by Storrie-Lombardi et al. (1992), their backpropagation network for morphological classification of galaxies behaves like a Bayesian classifier (see also Yai & Gersho 1990). For a general survey and a more detailed discussion of neural nets used for classification tasks see Lippmann (1987), Kindermann & Linden (1988), Klusch (1991) and Grauel (1992). Especially, Spirkovska & Reid (1992) used higher-order neural networks for a robust position, scale and rotation invariant object recognition and classification.

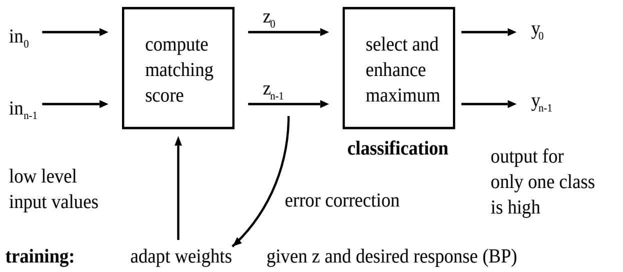

Fig. 4. Neural classifier

The neural net of the HNS uses a version of one well known learning procedures for multi-layered neural networks: the supervised adaptive backpropagation algorithm (Rumelhart & McClelland 1986). A global view of the macrostructure of a backpropagation net is shown in Fig. 5. The purpose of the algorithm is to train the net supervised such that it will approximate the desired classification function in a given precision ϵ. This is done by minimizing the net error defined over all weights and the given training set of sample patterns by computing its gradient at each unit locally. So a gradient descent method is performed in the space of weights. Although it is well-known that this is a rather poor optimization method it is still widely used in backpropagation neural nets for many applications (e.g. Odewahn et al. 1992, Storrie-Lombardie et al. 1992). In accordance with a known improvement of this method, the conjugate gradient descent method, the HNS uses backpropagation with momentum-term and batch version. Thus learning will be done using the generalized delta rule (Rumelhart et al. 1986, Grauel 1992):

$$w_{lij}^{\text{new}} := w_{lij}^{\text{old}} - \alpha \cdot \left(\sum_{s=1}^{n_p} \delta_{li}^{s} \cdot a_{l-1,j}^{s} \right) + \eta \cdot \delta w_{lij}^{\text{old}}$$

with w_{lij} weight of connection between unit u_i at layer l and u_j at layer $l-1$, α learning rate, η momentum parameter, $a_{l-1,j}^{s}$ activity of unit j at layer $l-1$ (forward pass) and δ_{li}^{s} computed gradient of error at unit i at layer l for input pattern s, $\delta w_{lij}^{\text{old}}$ last update of weight w_{lij}, n_p number of patterns for one backpropagation cycle. Using the batch method of backpropagation all update terms will only be computed after one complete cycle with all training patterns. For computation of each units activity we use the well-known (Widrow & Lehr 1990) sigmoid function

$$F_i(net_i) = \frac{1}{1 + e^{-(net_i - \nu_i)}}$$

with $i \in \{1, \ldots, N\}$, ν_i, net_i bias resp. input of unit u_i.
For a short notation of the essential backpropagation algorithm see Appendix A. In contrast to the HNS most implemented improvements of the original backpropagation like the use of second-order methods, the adaptation of learning rate for backpropagation (Salomon 1989) and other approaches (Rehm & Sterzing 1992, Alder et al. 1992) ignore or do not correspond to the required strictly locally neural computing (Hecht-Nielsen 1989, Kohonen 1988a). The need for exclusively local computation in all neural units motivates in particular the Hecht-Nielsen backpropagation architecture based on neurophysiological investigations (Hecht-Nielsen 1989). This special neural architecture is used by the HNS. It is provided that local memory of each neural unit is restricted to only one value. Therefore every unit is divided into two parts. The *main cell* ($s_{l,i}$), computes only the activity of unit $u_{l,i}$. Its associated *weight cells* ($p_{l,i,j}$) are restricted to calculate the weights $w_{l,i,j}$ of the respective neural unit $u_{l,i}$. Each cell is partitioned into the following parts: local memory, receiver (*in*), activity-computation (*act*) and output (*out*). In particular, the weight cells have

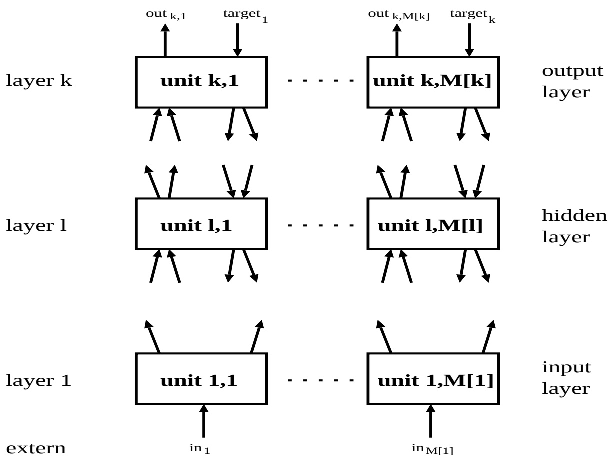

Fig. 5. Macrostructure of backpropagation net.
$k \in \mathbb{N}$: number of layers, $M[i]$: number of units in layer i, in_i: i^{th} component of the input pattern, $\text{out}_{i,j}$: output signal of the j^{th} unit of layer i, target_i: desired i^{th} output signal of layer k.

parts for weight-computation and update-memory . The output value $p_{l,i,0}$ is identical with the bias of its respective main cell $s_{l,i}$. Therefore any computation and storage is indeed exclusively local! A similar architecture can be found in Fausset (1990).
For more details on how the forward and the backward pass of the backpropagation algorithm work within the proposed architecture and its complexity of computation see Klusch (1991).

2.2.2. Semantic Network

For representing conceptual knowledge about stars a semantic net is designed in accordance with the MK classification. The set of all stars is fracted into several subsets based on spectral and luminosity type of included stars. Every subset represents one conceptual MK class and a star can be seen as a member or instance of it. The root of the resulting class hierarchy includes all stars described by the properties given in the correlation table (cf. Section 2.1). In derived classes all stars are described additionally by their properties with respect to the type of classification the special class stands for. A formal description of the used semantic net by a frame model is given in Klusch (1991). This model is implemented in an object-oriented language (C++). One part of the class hierarchy of the semantic net is shown in Fig.6. The respective knowledge base of stars (KBS) can be considered as an extended implementation of this semantic net. As mentioned in Section 2.1 an user-defined initialization of the KBS implies that of the given semantic net and determines the actual state of the KBS. In the current version of the KBS, stars with spectrum B, A, F (in subtypes 0 and 5) and luminosity type V, III, I are stored. The property values of all 18 classes depend on the manipulation of respective data files by user as mentioned in Section 2.1. In the currently implemented version of the HNS it is impossible to change the internal structure of the KBS, in particular the class hierarchy of the semantic net, externally. An overview of the initialization of the KBS is shown in Fig. 7.
In general the semantic net is used as a basic structure for information retrieval with respect to the MK class classified by the neural net and moreover for providing a starting point for the spreading-activation algorithm (TR mode). Examples of using the TR mode of the HNS were given in Section 2.1, for an explanation of spreading-activation cf. next section. The benefits of the hybrid synthesis of the semantic net with the neural net are a fast and comfortable information retrieval with respect to the MK class actually classified by the neural net and on the other hand providing a facility for the comparison of named MK classes without forcing the user to give any specific numerical input.

2.3. Integration and communication algorithms

The hybrid character of the HNS bases mainly on the integration of the neural and semantic net in order to dispose the functionality of these components. According to the possibility of independent generation and initialization of both nets the integration module of the HNS blends out all classes of the semantic net which are incomparable with those used by the neural net. The identification of classified neural output units with attached instances of MK classes of the semantic net is the presupposition for both modes of the HNS informally introduced in Section 2.1.

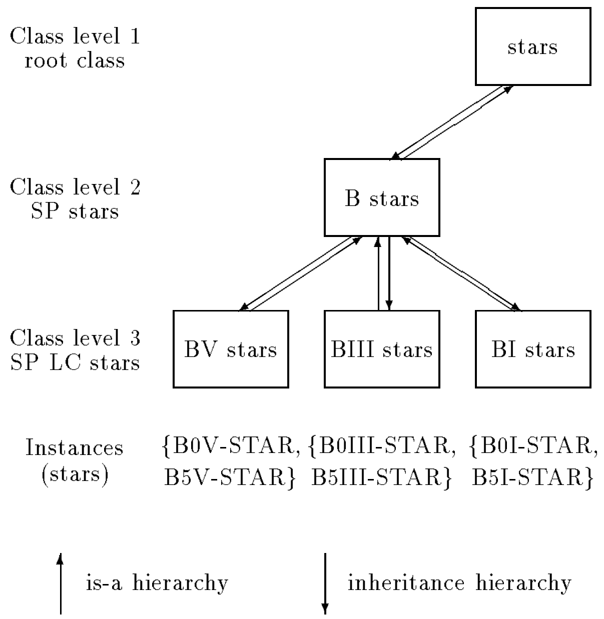

Fig. 6. Part of the class hierarchy

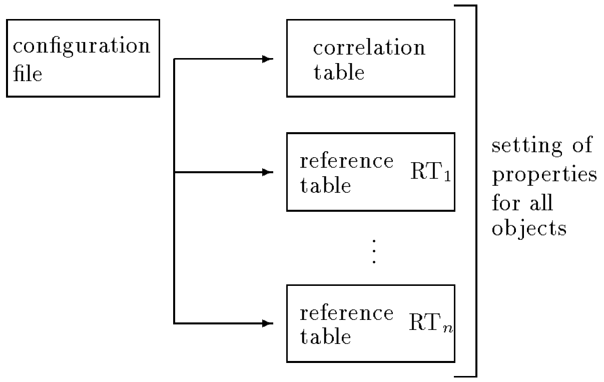

Fig. 7. Initialization of KBS

Since we have introduced the structure and use of the neural net in Section 2.2.1 it is obvious that the performance of the classification method BU bases in principle on the performance of the forward pass of the neural net, identification of respective class in the semantic net and final evaluation of properties. The more interesting classification method TR bases intrinsically on a spreading activation algorithm to determine similarities between two different MK classes of the semantic net using the strengths of neural connections. A formal description how this spreading activation works in the HNS is presented in Klusch (1991). See also the example for the TR classification in Section 2.1.
Spreading activation starts within the semantic net with assigning symbolic activation to the respective instance of desired conceptual class of stars. Now the attached neural output unit, $s_{k,id}$, spreads the activation through its backward connections to the units at the previous layer. From these units in turn activation passes back to all units at the last layer except the initial unit. The resulting maximum of top return activities

$gain_i$ of main cells $s_{k,i}$ leads to the class of stars most similar to the given one by identifying the instance to which the unit is attached. The activity $gain_i$ is computed by

$$\sum_{j=1}^{M[k-1]} (p_{k,id,j}.weight/M[k-1]) * (p_{k,i,j}.weight/M[k])$$

with $i \neq id$.
Hence, the immediate identification of respective class and use of the class hierarchy of the semantic net leads to an answer for the required comparison of named classes with respect to some similarity detected by the neural net. The essential notation for both communication algorithms is given in Appendix B.

3. Empirical examination

The success of the HNS depends greatly on that of the neuronal classification. Since there is no general solution for the problem of how to design an optimal neuronal network for a given application, the only possibility of testing the efficiency is an empirical examination. In Section 3.1 we shortly introduce the known relation between stellar classification and Strömgren $uvby\beta$ photometric indices. The main results for both classification methods (BU and TR) with respect to a given topology of the neural net are presented in Section 3.2 while the influence of different values for learning parameters on the learning process is summarized in Section 3.3.

3.1. Stellar classification

We used Strömgren $uvby\beta$ photometric indices from the Hauck & Mermilliod (1985) catalogue for the MK classification of stars by the neural net of the HNS. Photometry fulfills the required conditions: the data is significant and standardized by the use of standard stars. These stars can be (and are) used for the determination of fundamental stellar parameters, such as effective temperature $T_{\rm eff}$ and surface gravity g or stellar luminosity L. Flynn & Morrison (1990) described the necessary steps for the calculation of quasi-photometric indices from objective-prism plates. After this work is done the HNS is applicable to the survey data in its present form. Alternatively one can derive photometric colours from exposures of photographic plates with different filters, like it was used by Smriglio et al. (1986) for spectral classifications.
Measured and published Strömgren colours are $(b-y)$, m_1, c_1, and β or, corrected for interstellar reddening, $(b-y)_0$, m_0, c_0 (β is reddening free by its definition). c_0 (measures the Balmer jump of hydrogen at 3646 Å) and β (measures the equivalent width of the Balmer line H_β) are used to determine the fundamental parameters $T_{\rm eff}$ and g closely related to the spectroscopic Morgan-Keenan type and the luminosity class. In the region of the A0 stars an ambiguigity appears. It is necessary to define new indices a_0 and r which are linear combinations of the previously mentioned indices. Further details are given in e,g, Golay (1974). The neural net of the HNS can solve this problem by learning the decision regions also for stars of MK class A0 correctly. We decided to use the reddening free indices as input data for adapting the neural net of the HNS. These are combinations of indices that compensate for the effect of interstellar reddening, namely

$$[m_1] = m_1 + 0.33(b-y)$$
$$[c_1] = c_1 - 0.19(b-y)$$

and the reddening free β.
The behavior of the HNS will now be examined concerning the optimum choice of the topology and learning parameters for its neural net.

3.2. Topology

The number and arrangement of the units in layers determine the topology of the neural net. For the examination of each topology a test run for each of our training sets was necessary. For all test runs we investigated the behavior of backpropagation networks with three layers and fixed values for the learning parameters: $\alpha = 0.5$, $\eta = 0.5$. Initial values for the weights were randomly chosen from the interval $[-0.5, 0.5]$.
We have to distinguish between the BU (bottom up) mode and the TR (top return) mode. The classification method of the TR mode is not comparable with that of the BU mode. A star is considered as successfully classified by the BU classification if the class identified as the most likely class to a given numerical input is in accordance with the given MK class. In particular this means that both, spectral and luminosity, classes must be correctly recognized by the neural net! Moreover, a star is considered as successfully classified by the TR classification if the identified class is identical with the second candidate of the rank list obtained by the BU classification. Therefore the TR classification rate indicates especially the relation between the classification of the BU method and spreading activation. Our calculated classification rate is the simple percentage relation between the summarized number of correctly classified test stars for each class and the cardinality of the whole test set. Thus the term classification rate denotes the widely used percent correct recognition rate. For reasons of space limitation we refer the interested reader for a more detailed presentation of the performance of the HNS to Klusch (1991).
In a first run (RUN 1A) we tested networks for the MK classes B0V, B0III, B0I, B5V, B5III, B5I, A0V, A0III, A0I. In a second run (RUN 2) we added the classes A5V, A5III, A5I, F0V, F0III, F0I. As training sets we used the averaged values of photometric indices for the standard stars of each class (taken from Morgan et al. 1943). In an additional run (RUN 1B) we tested the same MK classes as in RUN1A but now using the original measured values to investigate the influence of random errors in the training sets. For each run we tested about 30 (RUN 1A/B) and 50 (RUN 2) stars with accurate spectral classifications from Morgan et al. (1943). These tested stars are different from the standard stars trained during the learning phase. Since the number of input units is equal to the number of photometric indices, we only varied the number of units of the hidden layer. For RUN 1A/B we investigated the behavior of neural networks with 4, 5, 6, 7, 8, 9, and 10 hidden units; for the larger sample of RUN 2 we used 7, 8, 9, 12, 14, 15, and 16 units. For all runs we used neural nets which have to learn for each class just one corresponding input pattern! The system will probably yield good or even better results also without this restriction.
The results of RUN 1A are given in Table 3. We found the networks with 8 and 5 hidden units to produce the highest classification rates for BU (93%) and TR (66%) method, respectively. The network with 5 hidden units is a reasonable compromise with a classification rate of 90%/66% for the BU/TR method. The behaviour of the network in RUN 1B (cf. Table 4) is similar to RUN 1A, but with always moderatly lower classifica-

Table 3. Results of RUN 1A. The classification rate, the corresponding number of backpropagation cycles $N_{\rm BP}$ and ϵ are given for the BU and the TR mode.

Units	Bottom up (BU)			Top return (TR)		
	Rate (%)	$N_{\rm BP}$	ϵ	Rate (%)	$N_{\rm BP}$	ϵ
4	83	5886	1.8	60	5886	1.8
5	90	4492	1.7	66	4492	1.7
6	76	2908	1.3	50	2212	1.5
7	86	3133	1.3	63	1421	1.7
8	93	3809	1.2	56	1753	1.5
9	80	4045	1.0	56	807	2.3
10	90	2032	1.4	53	6259	0.9

Table 4. Results of RUN 1B.

Units	Bottom up (BU)			Top return (TR)		
	Rate (%)	$N_{\rm BP}$	ϵ	Rate (%)	$N_{\rm BP}$	ϵ
4	83	2499	1.7	70	1265	2.2
5	86	7548	1.1	66	3839	1.5
6	70	2487	1.3	46	3817	0.9
7	70	1473	1.6	46	1473	1.6
8	86	4179	0.8	60	2331	1.3
9	70	1536	1.7	43	1174	2.0
10	76	2943	1.2	53	857	2.3

tion rates. This seems to be caused by the noisy input for the trained stars of RUN 1B.

The optimum classification rates are reached for different precisions ϵ of approximation, i.e. the net error $\leq \epsilon$ (cf. Section 2.2.1), for different networks. In Fig 8 we show an example for the dependence of the classification rate on ϵ in RUN1A. It is demonstrated that the classification rate is not necessarily improved for a smaller ϵ.

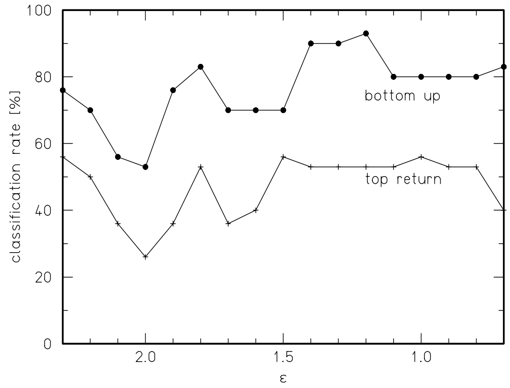

Fig. 8. Dependence of the classification rate on ϵ for a neural net with 8 hidden units in RUN 1A.

In nearly all test runs the classification rate of the TR method was below that of the BU method. But we want to point out that in most cases the MK class obtained by the TR method as the most similar class to a given one makes sense from the astrophysical point of view, even if it differs from the class obtained by the BU method as the second candidate. For example, the class most similar to B0I, B0III and B0V obtained by the TR method was B0III, B0V and B0III, respectively. For further investigations on this topic see Klusch (1991).

It is remarkable that for the BU classification no misclassifications concerning the temperature class were observed! In a few cases only the luminosity classes were confused. But this behavior is also reported for the photometric classification done by Smriglio et al. (1986) and classifications obtained in the conventional way (cf. Hoffleit 1982).

For Results of RUN 2 high classification rates for the BU method were obtained for networks with 7 and 8 (88%) and with 9 and 15 (85%) units. The best result (46%) for the TR method was achieved with 7 hidden units. This rate is even smaller than that obtained for the TR method in RUN 1A/B. The dependence of classification rates on ϵ in RUN2 is shown in Fig. 9.

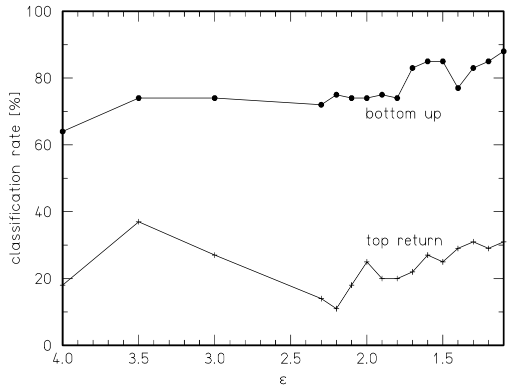

Fig. 9. Dependence of the classification rate on ϵ for a neural net with 8 hidden units in RUN 2.

3.3. Learning parameters

Here we will shortly demonstrate the influence of the learning parameters α and η on the learning process for a given net configuration. This is done by observing the direction and velocity of convergence (the path on the error surface in the space of weights) with respect to different combinations of values for α and η. We used the optimum configuration of RUN 1A (8 hidden units). Calculations are restricted to be done reaching $\epsilon = 1.5$. Results are presented in Table 5. From comparison with the former results presented in Section 3.2 it is obvious that these are not the best ones.

- The run with $\alpha = 0.9$ and $\eta = 0.5$ is the fastest, but oscillations in the net error are occuring during the iterations.
- For a given value of α an increase in η results in increasing oscillations. Oscillations are not necessarily damped by increasing η values!
- For all investigated values of α the fastest convergence is achieved with $\eta = 0.5$.
- The number of BP cycles increases with decreasing α.

– But: in many cases the run with the fastest convergence will not give the best results.

Table 5. Results of runs with different learning parameters. Given are the learning parameters α and η, the classification rates for BU and TR mode, and the number $N_{\epsilon=1.5}$ of BP cycles necessary to reach $\epsilon = 1.5$.

α	η	Classification rates		$N_{\epsilon=1.5}$
		BU	TR	
0.9	0.0	63	40	1841
0.9	0.5	63	50	1593
0.9	0.9	66	46	1861
0.5	0.0	80	40	2837
0.5	0.5	70	56	1725
0.5	0.9	83	56	2401
0.1	0.0	66	40	11990
0.1	0.5	80	40	2696
0.1	0.9	73	60	3283

4. Discussion

We compare the HNS with astrophysical methods used for the MK classification or the determination of equivalent physical parameters like effective temperatures and surface gravities:

1. the classical, visual MK classification from photographic spectra,
2. the program of Moon (1985) for the determination of temperature and gravity by means of the Strömgren $uvby\beta$ photometry,
3. the automated photometric classification in the Vilnius system (Smriglio et al. 1986) and
4. the program for classification of objective-prism spectra presented in LaSala (1988) and Kurtz & LaSala (1991).

These classification methods differs fundamentally from the hybrid neural approach introduced in this work. The HNS is the first system which uses a neural net for the MK classification of stars. Moreover, the hybrid synthesis of the neural and semantic net within the HNS offers a new approach for comparing named classes of stars without any numerical input and on the other hand an integrated, comfortable information retrieval.

The *classical MK classification* of stars uses photographic plates with a number of objective prism spectra taken at a single exposure. Classification is done by visually comparing the observed spectra with standard spectra. One basic catalogue of standard stars is that of Morgan et al. (1943). This method is very effective with respect to observing time, because one can take spectra of many stars with one exposure. But it is very ineffective with respect to manpower, because it is necessary to classify every star by visual comparison with the standards. This is also a source of errors and inconsistencies between different authors. The "Bright Star Catalogue" of Hoffleit (1982) reports discrepant classifications of several stars: about 40 repeatedly observed stars classified with luminosity class V by one author are classified to have luminosity class III or even II by equally reputed authors! Some examples are given below:

Star	Classification	
	Hoffleit (1982)	HNS
HR 114	A7III, F2V	F0V
HR 308	F4II, F4III, F4V	F5III
HR 769	A0III, B9.5V	A0III
HR 5320	B1.5III, B2Ib	B0III

Every reliable, fully automated classification method like the HNS is a significant improvement of the visual MK classification of stars.

The other conventional classification method by *Strömgren $uvby\beta$ photometry* is used for the determination of basic stellar parameters. This is done by using photometric calibrations. These calibrations define relations between photometric indices and quantities like effective temperatures and surface gravities (or, alternatively, luminosities). A two-dimensional grid for the determination of temperature and gravity was published by Moon & Dworetsky (1985). As shown by Castelli (1992) and Napiwotzki et al. (1993), a good accuracy can be achieved with this method. Moon (1985) published a FORTRAN-program for the analysis of photometric data. But for the use of this program an approximate pre-classification is necessary. In doubtful cases the MK class can be determined interactively. An automatic pre-classification only by means of the photometric data is difficult. In contrast to that the HNS is able to learn this classification without any prior knowledge about relations between colours and photometric indices. We also think that the combination of both methods would be a promising project.

A *statistical* procedure for the MK classification of stars based on the *Vilnius multicolour photometry* was developed by Smriglio et al. (1986). Smriglio et al. (1990) present classification rates of this procedure analogous to the BU classification rates of the HNS. Their results are nearly as reliable as those obtained by the HNS. But nevertheless their most seriously problem is the luminosity classification of stars which is shown to be solved by the HNS. In contrast to their classification rate of about 80% the HNS provides of about 90%.

Similar to the traditional MK classification the method of LaSala (1988) and Kurtz & LaSala (1991) uses the objective-prism spectra of stars. It is based on a *weighted metric distance technique*, known as a simple pattern recognition method. A major disadvantage is the need for comparing the spectrum (containing lots of spectral points) of every investigated star with many standard stars. Obviously this procedure is a very time consuming one. Their comparison with visual spectral classifications showed results moderatly poorer than that obtained by the HNS. This may be caused by the different input data. Besides, in their status report (Kurtz & LaSala 1991) only an example limited to one spectral class is given.

We have shown that the HNS obviously offers significant improvements for the MK classification of stars compared to astronomical techniques used to date. One main advantage of the HNS is the fully automatic classification of stars. After an eventually time consuming determination of an optimal neural net, the individual classification with both modes (BU and TR) of the HNS is very fast and yields high classification rates. Another advantage is the comparison of two named classes of stars without giving any raw input data (TR classification) which is very useful for determining some similarity of both! For this the HNS only uses knowledge it adapts from the user during a learning phase. We believe that this is also a new and promising issue on comparing classes of stars.

For further works on applications of the HNS we mention the use for spectroscopic surveys (like the Palomar Green survey or the Hamburg Schmidt survey) of the sky. Moreover, the system is not yet tested for working with photographic objective prism data used by the classical method. Adapting the HNS for this purpose is not very difficult. The main work, that is left to do, is to reduce the data on a not too large set of significant and homogeneous data points. A first progress in this direction was reported by Flynn & Morrison (1990). In addition, considering the very useful facility of the TR mode of the HNS further research on spreading activation would be very promising. We provide a novel, elegant and successful hybrid approach for classification, which can cover a wide range of applications not only for astronomical purposes.

Acknowledgements. The authors thank Prof. Dr. D. Klusch, J. Wappler, T. Prey, Dr. S.D. Jordan, Dr. K. Werner, Dr. N. Junkes, H.-J. Hagen for reviewing earlier drafts of this paper and providing constructive comments on its representation.

References

Alder, M., Lim, S.G., Hadingham, P., Attikionzel, Y., 1992, "Improving neural network convergence", Pattern Recognition letters 13(5)

Bergerson, K., Wunsch, D., 1991, "A commodity trading model based on a neural network expert system hybrid", Proc. IEEE IJCNN-91, Vol.1

Burda, P., Feitzinger, J.V., 1992, A&A 261, 697

Castelli, F. 1991, A&A 251, 106

Duda, R., Hart, P.E., 1973, "Pattern classification and scene analysis", Wiley & Son

Fausset, D., 1990, "Strictly local backpropagation", Proc. IEEE IJCNN-90, Vol.3

Flynn, C., Morrison, H.L., 1990, AJ 100, 1181

Gallant, S., 1988, "Connectionist expert system", Commun. of ACM, Vol.31(2)

Grauel, A., 1992, Neuronale Netze, BI-Wiss.Verl., Mannheim

Griffith, R.L., 1982, "Three principles of representation for semantic networks", ACM Trans.on Database Systems

Golay, M., 1974, "Introduction to astronomical photometry", Reidel Publ. Comp.

Hauck, B., Mermilliod, M. 1985, A&AS 60, 61

He, D.-C., Wang, L., 1992, "Unsupervised textural classification of images using the texture spectrum", Pattern Recognition, Vol.25(3)

Hecht-Nielsen, R., 1989, "Theory of the backpropagation neural network", Proc. IEEE IJCNN, Vol.1

Hendler, J.A., 1989, "Problem solving and reasoning: a connectionist perspective", in: Connectionism in Perspective, Elsevier Sci. Publ.

Hoffleit, D. 1982, "The Bright Star Catalogue", Yale University Observatory, New Haven, Connecticut, USA

Hruska, S., Kuncicky, D., Lacher, R., 1991, "Hybrid learning in expert networks ", Proc. IEEE IJCNN-91, Vol.2

Hudli, A.V., Palakal, M.J., Zoran, M.J., 1991, "A neural network based expert system model", Proc. IEEE Int. Conf. on Tools for AI

Kindermann, J., Linden, A., 1988, "Pattern Completion and Classification with backpropagation", Arbeitspapier GMD, Nr.329, Vol.8

Klusch, M., 1991, "Entwicklung und Implementierung eines hybrid-neuralen Systems HNS zur Klassifikation von Sternen", Diplom thesis, Institut für Informatik, CAU Kiel

Kohonen, T., 1988a, "Introduction to Neural Computing", Neural Network, Vol.1, Pergamon Journals

Kohonen, T., 1988b, "Self-organizing and associative memory", Springer Berlin

Kruschke, J.K., Movellan, J.R., 1991, "Benefits of gain: speeded learning and minimal hidden layers in Backpropagation Networks", IEEE Trans. SMC, Vol.21(1)

Kurtz, M.J., LaSala, J., 1991, "Toward an automated spectral classification", in: Objective-prism and other surveys, eds. A.G.D. Philip & A.R. Upgren, Van Vleck Obs. Contrib. No. 12, Davis Press, Schenectady, New York, p. 133

LaSala, J., 1988, "A program for automated two-dimensional classification of objective-prism spectra", in: Astronomy from large databases – scientific objectives and methodological approaches, eds. F. Murtagh & A. Heck, ESO Conf. Workshop Proc. No. 28, p. 127

Lauberts, A.,Valentijn, E.A., 1989,"The Surface Photometry Catalogue of the ESO-Uppsala Galaxies",ESO

Lee, J., Hwang, J., Davis, D., Nelson, A., 1991, "Integration of neural nets and decision tree classification for automated cytology screening", Proc. IEEE IJCNN-91, Vol.1

Lippmann, R.P., 1987, "An introduction to computing with neural nets", IEEE ASSP Magazine, Vol.4

Moon, T.T. 1985, Comm. Univ. London Obs. 78

Moon, T.T., Dworetsky, M.M., 1985, MNRAS 217, 305

Morgan, W.W., Keenan, P.C., Kellman, E., 1943,"An atlas of stellar spectra", University of Chicago Press, Chicago, Illinois USA

Napiwotzki, R., Schönberner, D., Wenske, V., 1993, A&A, in press

Odewahn, S.C., Stockwell, E.B., Pennington, R.L., Humphreys, R.M., Zumach, W.A., 1992, AJ 103, 318

Rabelo, L.C., Alptekin, S., Kiran, M., 1990, "Synergy of artificial neural networks and knowledge-based systems for intelligent FMS scheduling", Proc. IEEE IJCNN, Vol.1

Rehm, W., Sterzing, V., 1992, "Ein evolutionstheoretisches Optimierungsverfahren für Multi-Layer-Perceptrons", Informationstechnik it 5/92

Rich, E., 1990, "Expert systems and neural networks can work together", IEEE Expert, Vol.10

Rumelhart, D.E., McClelland, J.L., 1986, "PDP: Explorations in the Microstructure of Cognition", MIT Press, Vol.1

Rumelhart, D.E., McClelland, J.L., Hinton, E., 1986, "A general framework for PDP", in: Parallel Distributed Processes, Vol. 1, MIT Press

Salomon, 1989, "Adaptive Regelung der Lernrate bei Backpropagation", Forschungsbericht FB-Informatik TU Berlin

Samad, T., 1990, "Hybrid distributed/local connectionist architectures" Proc. IEEE Neural Networks

Shastri, L., 1988, "Semantic Networks: an evidential formalization and its connectionist realization", Morgan Kaufmann Publ., Los Altos(CA), Pitman, London

Smriglio, F., Boyle, R.P., Straizys, V. et al., 1986, A&AS 66,181

Smriglio, F., Dasgupta, A.K., Nandy, K., Boyle, R.P., 1990, A&A 228, 399

Specht, D., 1990, "Probabilistic neural networks", Neural Networks, 3/90

Spirkovska, L., Reid, M.B., 1992, "Robust position, scale and rotation invariant object recognition using higher-order neural networks", Pattern Recognition, Vol.25(1)

Storrie-Lombardi, M.C., Lahav, O., Sodre, L., Storrie-Lombardie, L.J., 1992, MNRAS 259, 8P

Tarr, G.L., Rogers, S.K., Kabriesky, M., 1989, "Hybrid neural networks for tactical target recognition", Proc. IEEE Neural Networks, Vol.1

Ullman, J.R., 1973, "Pattern recognition techniques", Butterworths

Venkateswarlu, N. B., Raju, P.S.V.S.K., 1992, "Fast ISODATA clustering algorithms", Pattern Recognition, Vol.25(3)

Widrow, B., Lehr, M.A., 1990, "30 years of adaptive neural networks: perceptron, madaline and backpropagation", Proc. IEEE, Vol.78/9

Yai, E., Gersho, A., 1990, "Maximum a posteriori decision and evaluation of class probabilities by Boltzmann perceptron", Proc. IEEE IJCNN-90, Vol.1

Appendix

A. Backpropagation algorithm

int *bpcycles* //number of desired backpropagation cycles
int k //number of layers
begin
Weight initialization;
Generation of input patters and target outputs;
for $i = 1$ **to** *bpcycles*
{
for $s = 1$ **to** n_p
{
for $l = 1$ **to** k
Compute activities on actual input pattern s;
Compute error on input pattern s;
for $l = k$ downto 2
Compute errors and weight variations for units at layer l;
}
Compute net error (sum of pattern errors);
if (exit criterion satisfied) **then stop**
Updating of weight matrix [batch version]
}
end.

B. Communication algorithms

For a notation of both communication algorithms the following definitions are used.

Define sets:
$\mathcal{S}$ set of all strings
$\equiv_{\rm str}$ binary operator which denotes the identity on strings.
INST := set of all instances of all conceptual classes of the semantic net;
ODescr := $\{< \text{classK} >_j \,\|1 \le j \le OD, < \text{classK} > \in \mathcal{S}\}$ $\cup \{< \text{classUK} >\}$,
set of internal object descriptions,the class denoted by $< \text{classUK} >$ is the class CL_Unknown mentioned in Section 2.1;
OD cardinality of set ODescr;
CDescr := $\{< \text{classN} >_j \,\|1 \le j \le M[k], < \text{classN} > \in \mathcal{S}\}$
set of class descriptions, externally set by user (while generating neuronal net) associated to the output units;
$k \in \mathbb{N}$: number of layers
$M[i]$: number of units in layer i
$\mathbf{in}_i$: i^{th} component of the input pattern,

Spreading activation/evaluation algorithm (TR mode):

begin
Get high-level input // e.g. $< \text{classK} >$, properties
for $n = 1$ **to** OD // find starting point for TR
if ($< \text{classK} >_n \equiv_{\rm str} < \text{classN} >_{id}$)
{ *Spreading-activation(id)*; **break;** }
Compute maximal gain$_j$ $= \max\{gain_i \| 1 \le i \le M[k]\}$;
Identification(j); // identify instance for resp. output unit
Evaluate desired properties; // retrieve property values of instance
end.

Classification/evaluation algorithm (BU mode):

begin
Get low-level indices in_i and desired properties;
for $i = 1$ **to** $M[1]$
$s_{1,i}.in = s_{1,i}.act = s_{1,i}.out = in_i$;
Forward pass;
Compute maximal $s_{k,j}.out$; // $j \in \{1, ...M[k]\}$
$Identification(j)$;

if desired by user (class CL_Unknown classified):
{
for $i = 1$ **to** $M[k]$
if($< \text{classUK} >_s \equiv_{\rm str} < \text{classN} >_i$)
{ *Spreading-activation(i)*; **break;** }
Compute maximal gain$_j$ $= \max\{gain_i \| 1 \le i \le M[k]\}$;
Identification(j);
}

Evaluate desired properties;
end.

This article was processed by the author using Springer-Verlag LaTeX A&A style file 1990.